Backup And Recovery Approaches Using Aws (detailed)

AWS Whitepaper: Backup and Recovery Approaches Using AWS [PDF]

http://media.amazonwebservices.com/AWS_Backup_Recovery.pdf December 09, 2014

Amazon Web Services – Backup and Recovery Approaches Using AWS December 2012 Page 4 of 12 In this case, there is no need to back up the server itself.

Backup and Recovery Approaches
Using Amazon Web Services

December 2012

Simon Elisha

Abstract
Traditional enterprise backup and recovery strategies typically take an agent-based approach whereby the entire contents of a server are backed up over either the local area network (LAN) or the storage area network (SAN). Traditional architectures have required this approach because replacing failed components is complex, time consuming, and operationally intensive. This has, in turn, created a backup environment that is complex to manage and resource intensive to operate—requiring technologies such as data de-duplication and virtual tape libraries to cope with ever-increasing workloads.

The AWS platform enables a far more lightweight approach to backup and recovery due, in part, to the following characteristics:

- Computers are now virtual abstract resources instantiated via code rather than being hardware-based.
- Capacity is available at incremental cost rather than up-front cost.
- Resource provisioning takes place in minutes, lending itself to real-time configuration.
- Server “images” are available on-demand, can be maintained by an organization, and can be activated immediately.

These characteristics offer you opportunities to recover deleted or corrupted data with less infrastructure overhead.

This paper is intended to describe some of the high-level concepts you can leverage to deliver less complex, lightweight data backup and recovery capabilities.

Protecting Configurations Rather Than Servers

The Amazon Elastic Compute Cloud (Amazon EC21) service enables the backup and recovery of a standard server, such as a web server or application server, so that you can focus on protecting configuration and stateful data—rather than the server itself. This set of data is much smaller than the aggregate set of server data, which typically includes various application files, operating system files, temporary files, and so on. This change of approach means that regular nightly incremental or weekly full backups can take far less time and consume less storage space.

When a compute instance is started in Amazon EC2, it is based upon an Amazon Machine Image (AMI)2 and can also connect to existing storage volumes—for example, Amazon Elastic Block Store (Amazon EBS)3. In addition, when launching a new instance, it is possible to pass “user data”4 to the instance that can be accessed internally as dynamic configuration parameters.

A sample workflow is as follows:

- Launch a new instance of a web server, passing it the “identity” of the web server and any security credentials required for initial setup. The instance is based upon a pre-built AMI that contains the operating system and relevant web-server application (e.g., Apache or IIS).
- Upon startup, a boot script accesses a designated and secured Amazon Simple Storage Service (Amazon S3)5 bucket that contains the specified configuration file(s).
- The configuration file contains various instructions for setting up the server (e.g., web server parameters, locations of related servers, additional software to install, and patch updates).
- The server executes the specified configuration and is ready for service. An open source tool for performing this process, called cloud-init6, is already installed on Amazon Linux AMIs and is also available for a number of other Linux distributions.

Figure 1: Traditional Backup Approach Figure 2: Amazon EC2 Backup Approach

1 http://aws.amazon.com/ec2/
2 http://docs.amazonwebservices.com/AWSEC2/latest/UserGuide/index.html?AMIs.html
3 http://aws.amazon.com/ebs/
4 http://docs.amazonwebservices.com/AWSEC2/latest/UserGuide/index.html?AESDG-chapter-instancedata.html
5

http://aws.amazon.com/s3/
6
https://launchpad.net/cloud-init

In this case, there is no need to back up the server itself. The relevant configuration is contained in the combination of the AMI and the configuration file(s). So the only components requiring backup and recovery are the AMI and configuration file(s).

Consider a web farm of 10 servers, each with an operating system and related configuration files of 5 GB per server—requiring 50 GB of storage and network capacity to do a full backup. (Web content is typically stored in a separate repository that is backed up independently.) Contrast this to the AWS approach where you need to protect only the AMI (of, say, 5 GB) and the relevant configuration files (typically tens of KB). This dramatically reduces the overhead of backup and recovery, eliminates "backup windows," and provides effective version control for the environment.

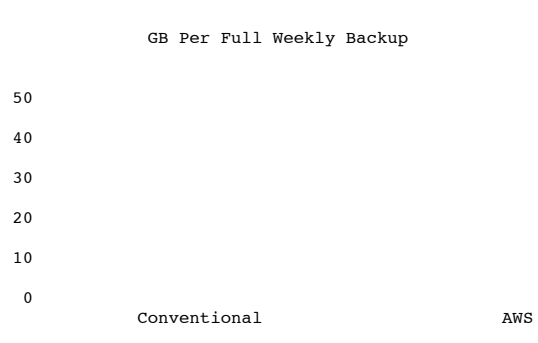


Figure 3: Example reduction in backup data volume

Self-Configuring Instances – Creating Flexibility and Deployment Options
Because you can start and stop instances at will, and have different versions of an application running concurrently, you can leverage more sophisticated and flexible deployment options. The self-configuration of instances enables you to implement techniques such as rolling-upgrades and A/B testing in the environment.

For example, to implement a new version of an application server in the architecture, you can take the following approach:

1. Create a new instance of the application based upon the correct AMI version and relevant configuration files. In our example, we call this "Application Version 2.0."
2. Map "Application Version 2.0" to the relevant load balancer so it is now "in the rotation" of servers available to service a customer request.
3. Once you confirm that "Application Version 2.0" is in production, you can stop or terminate the existing "Application Version 1.0" instances.
4. At this point, the entire application is operating in version 2.0 mode without outages and with simple rollback capability to version 1.0 using the stopped instances.

To take this example further, you may want to utilize A/B testing of new application capabilities or features. In the same way that you were able to introduce a new version of the application server into the architecture for a rolling upgrade, you can use the load balancer to direct certain customers to particular (new version) instances of the application, while the remaining customers continue to use the existing version of the application.

Figure 4: Rolling Upgrade – Adding new version instances

Figure 5: Rolling Upgrade – Decommission and terminate old versions

Other important aspects to consider are security and remediation of compromises. Because instances are easily replaced, you can focus on a strategy of "replace" rather than "repair." This strategy significantly reduces response speed and complexity.

For example, consider a content management system (CMS) that hosts your Internet presence. For some reason, the latest version of the code has not been deployed, and hackers know about and are exploiting a security breach on your site. Forensically analyzing which instances are compromised is time consuming, and trying to "sanitize" each one is often impossible to do with 100% certainty. Instead, you simply terminate the compromised instances and replace them with "fresh" ones. These new instances would then leverage updated configuration files to ensure that the latest patched versions of the software are always deployed. By taking this approach, you eliminate the risk of a security breach that is not completely remediated and guarantee that the new instances are not compromised.

This approach also provides an effective way to "architect for failure," which is a key design pattern when deploying distributed systems at scale. Because you can automatically replace components at will, unexpected failures need not affect service delivery.

Backup and Recovery of the Amazon Machine Image (AMI)
AMIs that you register are automatically stored in your account using Amazon EBS snapshots. These snapshots reside in Amazon S3 and are highly durable7. This means that the underlying storage mechanism for the AMIs is protected from multiple failure scenarios.

It is also possible to share AMIs between separate AWS accounts. Consequently, you can create totally independent copies of the AMI by:

- Sharing the original AMI to another specified AWS account that you control
- Starting a new instance based upon the shared AMI
- Creating a new AMI from that running instance

The new AMI is then stored in the second account and is an independent copy of the original AMI. Of course, you can also create multiple copies of the AMI within the same account.

Backup and Recovery of Configuration Files
Customers use a variety of version management approaches for configuration files, and you can follow the same regime for the files used to configure your Amazon EC2 instances. For example, you could store different versions of configuration files in designated locations and securely control them like any other code. You then back up these code repositories using the appropriate backup cycle (e.g., daily, weekly, monthly) and snapshots to protected locations.

Furthermore, you could use Amazon S3 to store your configuration files, taking advantage of the durability of the service e in addition to backing up the files to an alternate location on a regular basis. Bootstrap approaches are limited only by your imagination. We recommend using AWS CloudFormation templates as you can describe your AWS resources, and any associated dependencies or runtime parameters in a simple JSON file.

Backing Up Database and File Servers
Backing up data for database and file servers differs from the web and application layers. In general, database and file servers contain larger amounts of business data (tens of GB to multiple TB) that must be retained and protected at all times. In these cases, you can leverage efficient data movement techniques such as snapshots to create backups that are fast, reliable, and space efficient.

For databases that are built upon RAID-sets of Amazon EBS volumes (and have total storage less than 1 TB), an alternative backup approach is to asynchronously replicate data to another database instance built using a single Amazon EBS volume. While the destination Amazon EBS volume will have slower performance, it is not being used for data access and can be easily snapshotted to Amazon S3 using the Amazon EBS snapshot capability (see the Snapshot Options for Amazon EBS section).

7 http://aws.amazon.com/s3/ - protecting

Alternative to Backing Up Static Content
If you manage large data sets of static information (e.g., map tiles or web site graphics), you can opt to migrate that data into Amazon S3, which is designed to provide 99.999999999% durability of object storage. This enables the data to be both highly durable while also being served directly from Amazon S3 rather than via web servers—potentially improving application performance.

To protect against logical corruption, you can also use techniques such as object versioning8, MFA Delete9 and simply copying the data to another Amazon S3 bucket.

Snapshot Options for Amazon EBS
Amazon EC2 volumes use Amazon EBS to store block-based data. Examples of this are file systems and databases. Amazon EBS natively enables you to create a snapshot of a volume to Amazon S3 using the AWS Management Console, the command line interface (CLI), or the APIs. Using the console, clicking the Create Snapshot option commences the creation of a snapshot to Amazon S3.

Figure 3 — Creating a snapshot from Amazon EBS using the console.

You can also create the snapshot using the ec2-create-snapshot command.

When you apply these commands to a backup strategy, you protect your data directly to durable disk-based storage. You can schedule and issue the commands on a regular basis, and due to the economical pricing of Amazon S3, you can retain many generations of data. Further, because snapshots are block-based, you consume space only for changed data after the initial snapshot is created.

8 http://docs.amazonwebservices.com/AmazonS3/latest/dev/Versioning.html
9 http://docs.amazonwebservices.com/AmazonS3/latest/dev/UsingMFADelete.html

To restore data from a snapshot, use the console or the CLI command ec2-create-volume to create a new volume from an existing snapshot. For example, to restore a volume to a prior point-in-time backup, you could use the following sequence:

1. Create a new volume from the backup snapshot using the following command:

```
ec2-create-volume –z us-west-1b –snapshot MySnapshotName
```

2. Within the Amazon EC2 instance, un-mount the existing volume (e.g., by using umount in Linux or the Logical Volume Manager in Windows).
3. Detach the existing volume from the instance using the following command:

```
ec2-detach-volume OldVolume
```

4. Attach the new volume that was created from the snapshot using the following command:
```
ec2-attach-volume VolumeID –I InstanceID –d Device
```
5. Remount the volume on the running instance.

This process enables a fast and reliable way to restore full volume data as needed. If you need only a partial restore, you can attach the volume to the running instance under a different device name, mount it, and then use operating system copy commands to copy the data from the backup volume to the production volume.

Amazon EBS snapshots can also be copied between AWS Regions using the Amazon EBS snapshot copy capability via the Console, API or GUI10. This enables data to be protected out of region without having to manage the underlying replication technology.

Creating Consistent or “Hot” Backups

When you back up a system, it is ideal to have the system in a “quiet” state where it is not performing any processing. From a backup perspective, the “ideal” state is a machine that is accepting no traffic—but this ideal is increasingly rare as 24/7 IT operations become the norm.

As such, it is necessary to “quiesce” the file system or database in order to take a “clean” backup. How you do this depends on your database and/or file system—so due diligence is required. To summarize the process for a database:

- If possible, put the database into “hot backup mode.” Alternatively, create a “read replica” copy of the database; this is a copy of the database that is up to date, but runs on a separate instance. Keep in mind that, on AWS, you can run this instance for the duration required to perform the backup and then close it down—saving resources. Also note that there may be a performance impact on the primary database during the existence of the read replica due to additional replication workload.
- Issue the relevant Amazon EBS snapshot commands.
- Take the database out of hot backup mode, or if using a read replica, terminate the read replica instance.

Backing up a file system works similarly, and depends highly on the capabilities of the particular operating system or file system. An example of a file system that can flush its data for a consistent backup is xfs (xfs_freeze). If the file system in question does not support the ability to freeze, you should un-mount it, issue the snapshot command, and

10 http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-copy-snapshot.html

then re-mount the file system. Alternatively, you can facilitate this process by using a logical volume manager that supports freezing of I/O.

Because the snapshot process is fast to execute, and captures a “point in time,” the volumes you are backing up only need be un-mounted for a matter of seconds. This ensures that the backup “window” is as small as possible, and that outage time is predictable and can be effectively scheduled. While the data copy process of creating the snapshot may take longer, the snapshot activity requiring the volume to be un-mounted is very quick. Don’t confuse the two processes when structuring your backup regime.

Backups for Amazon Relational Database Service

The Amazon Relational Database Service (Amazon RDS)11 includes automated backups. This means that you do not need to issue specific commands to create backups of your database.

Amazon RDS provides two different methods for backing up and restoring your DB Instance(s); automated backups and database snapshots (DB Snapshots).

- Automated backups enable point-in-time recovery of your DB Instance. When automated backups are turned on for your DB Instance, Amazon RDS automatically performs a full daily backup of your data (during your preferred backup window) and captures transaction logs (as updates to your DB Instance are made). When you initiate a point-in-time recovery, transaction logs are applied to the most appropriate daily backup in order to restore your DB Instance to the specific time you requested. Amazon RDS retains backups of a DB Instance for a limited, user-specified period of time called the retention period, which by default is one day but can be set to up to thirty-five days. You can initiate a point-in-time restore and specify any second during your retention period, up to the Latest Restorable Time. You can use the DescribeDBInstances call to return the latest restorable time for your DB Instance(s), which is typically within the last five minutes. Alternatively, you can find the Latest Restorable Time for a DB Instance by selecting it in the AWS Management Console and looking in the Description tab in the lower panel of the console.

- DB Snapshots are user-initiated and enable you to back up your DB Instance in a known state as frequently as you wish, and then restore to that specific state at any time. DB Snapshots can be created with the AWS Management Console or by using the CreateDBSnapshot call and are kept until you explicitly delete them with the console or the DeleteDBSnapshot call.

Note that when you restore to a point in time or from a DB Snapshot, a new DB Instance is created with a new endpoint. (If you want to, you can delete the old DB Instance by using the AWS Management Console or a DeleteDBInstance call.) You do this so you can create multiple DB Instances from a specific DB Snapshot or point in time.

Multi-Volume Backups

In some cases, you may stripe data across multiple Amazon EBS volumes using a logical volume manager in order to increase potential throughput. When using a logical volume manager (e.g., mdadm or LVM), it is important to perform the backup from the volume manager layer rather than the underlying devices. This ensures all metadata is consistent and that the various sub-component volumes are coherent. In these cases, you can use the ec2-create-snapshot command for this type of backup with the logical volume manager. You can take a number of approaches to accomplish this, an example being the script created by alestic.com (http://alestic.com/2009/09/ec2-consistent-snapshot).

11 http://aws.amazon.com/rds/

You can also perform backups of this nature from the logical volume manager or file system level. In these cases, using a "traditional" backup agent enables the data to be backed up over the network. When using tools such as Zmanda, NetBackup, or CommVault, it is important to remember that they expect a consistent server name/IP address. As a result, using these tools in concert with instances deployed in a Virtual Private Cloud (VPC)12 is the best method to ensure reliability.

An alternative approach is to create a replica of the primary system volumes that exist on a single large volume. This simplifies the backup process, as only one large volume needs to be backed up, and the backup does not take place on the primary system. However, it is important to ascertain whether the single volume can perform sufficiently to maintain changes during the backup and whether the maximum volume size is appropriate for the application.

Other Backup and Recovery Integration Points

Oracle Backup Using the Oracle Secure Backup Cloud Module to Amazon S3

Database administrators are always seeking efficient ways to protect the data contained in Oracle databases. Oracle has made available the ability to backup data directly from the Oracle database to Amazon S3 buckets. This means that backups take advantage of the economical and durable storage made available by Amazon S3 with native integration into the Oracle database framework and operational procedures using RMAN.

Further information about installation and operation of OSB Cloud Module can be found at http://aws.amazon.com/oracle.

This approach to backup for Oracle enables low-cost, reliable off-premise backup of Oracle databases, and can apply to Oracle databases hosted both on-premise and in Amazon EC2.

Sending On-Premises Backups to Amazon S3

Many backup software vendors now support Amazon S3 as a backup destination (e.g., CommVault Simpana Software Cloud Storage Connector, SecoBackup, and Zmanda). Further, many storage gateways offer integration between existing backup software and Amazon S3 storage (e.g., Nasuni and Riverbed). This is useful in providing an off-site backup that is both durable and cost effective—eliminating the complexity and security risks of off-site tape management.

You can also leverage AWS Direct Connect13 to provide a dedicated link into Amazon S3 over which your data is sent. This provides the potential for both higher dedicated bandwidth and private connectivity.

AWS Storage Gateway14 also provides a useful method to send backups to Amazon S3, enabling seamless data migration between AWS's cloud storage and on-premises applications. AWS Storage Gateway stores volume data locally in your infrastructure, and in AWS. In addition to storage replication, it stores the data as an Amazon EBS Snapshot, which you can use to

12 http://aws.amazon.com/vpc/
13 http://aws.amazon.com/directconnect/
14 http://aws.amazon.com/storagegateway

recover data and present it to your Amazon EC2 instances. This makes recovery processes efficient and repeatable.

Managing Backup Generations and Security

When performing backups on an ongoing basis, it is important to implement effective backup rotation strategies to reduce storage overhead, and to ensure the correct versions of data are maintained as per business requirements. A detailed discussion of backup rotation protocols is beyond the scope of this paper.

Protocols aside, if your data is sensitive, you should encrypt it while it's in transit and at rest as part of the backup process. An interesting solution to the backup rotation and encryption requirement is the s3napback tool: http://dev.davidsoergel.com/trac/s3napback/.

When using Amazon RDS, your backups are created automatically and retained for up to 8 days—enabling recovery of DB Instances to any second within that period up to the last 5 minutes.

Long-Term Data Archival

Many customers have a requirement to retain digital information for long periods of time (e.g., 7 years, 21 years, life of the patient, or indeterminate duration) in a format whereby it can be retrieved when needed, albeit infrequently. This presents a challenge in being able to store large (and continually growing) volumes of information in a manner that is durable, economical, and low-maintenance. The Amazon Glacier15 service is designed to enable customers to efficiently and reliably store unlimited amounts of archival data at low cost, with high durability (i.e., designed to provide average annual durability of 99.999999999%), and for long periods of time. You can choose to retrieve your data anytime within a 3 to 5 hour time window, rather than instantaneously. This enables you to effectively meet the dual (and often conflicting) goals of cost effective long-term storage and near real-time data retrieval.

In Amazon Glacier, data is stored as archives that are uploaded to Amazon Glacier and organized into vaults, which customers can control access to using the AWS Identity and Access Management (IAM)16 service. You retrieve data by scheduling a job, which typically completes within 3 to 5 hours.

Amazon Glacier integrates seamlessly with other AWS services such as Amazon S3 and the AWS storage and database services. Amazon S3 enables you to create lifecycle policies that will archive data to Glacier (and allow retrieval) automatically.17

Customers can integrate Amazon
Glacier into their existing backup and

15
http://aws.amazon.com/glacier
16
http://aws.amazon.com/iam
17
http://docs.aws.amazon.com/AmazonS3/latest/dev/object-archival.html

archive tools and processes such that it represents a new tier of storage useful for any data to be kept for long periods of time. Furthermore, if you have existing tape-based archives, you can migrate them to Amazon Glacier using the AWS Import/Export service18 whereby physical devices can be shipped to AWS for direct ingestion into the relevant Amazon Glacier vaults.

Conclusion
The AWS platform provides new and more flexible options for infrastructure configuration that enable a far more efficient and cost-effective backup and recovery regime for enterprise customers. By evolving certain processes and procedures from current legacy approaches to the state-of-the-art "infrastructure as code" approach, you can achieve the correct level of backup and recovery for your applications while reducing backup infrastructure and complexity.

Further Reading
1. Backup and Storage Webpage - https://aws.amazon.com/backup-storage/

2. Step-by-step video series on how to backup your Oracle Databases to Amazon S3 using Oracle Secure Backup Cloud Module - https://aws.amazon.com/backup-storage/gsg-oracle-rman/

18
http://aws.amazon.com/importexport

Using Amazon Web Services for Disaster Recovery

http://d36cz9buwru1tt.cloudfront.net/AWS_Disaster_Recovery.pdf December 09, 2014

Key steps for recovery: 1. Start your application Amazon EC2 ... following steps outline the different fail-back approaches: Backup ... Amazon Web Services – Using ...

Using Amazon Web Services for Disaster Recovery
October 2014

Glen Robinson, Attila Narin, and Chris Elleman

Contents

Abstract
In the event of a disaster, you can quickly launch resources in Amazon Web Services (AWS) to ensure business continuity. This whitepaper highlights AWS services and features that you can leverage for your disaster recovery (DR) processes to significantly minimize the impact on your data, your system, and your overall business operations. The whitepaper also includes scenarios that show you, step-by-step, how to improve your DR plan and leverage the full potential of the AWS cloud for disaster recovery.

Introduction

Disaster recovery (DR) is about preparing for and recovering from a disaster. Any event that has a negative impact on a company's business continuity or finances could be termed a disaster. This includes hardware or software failure, a network outage, a power outage, physical damage to a building like fire or flooding, human error, or some other significant event.

To minimize the impact of a disaster, companies invest time and resources to plan and prepare, to train employees, and to document and update processes. The amount of investment for DR planning for a particular system can vary dramatically depending on the cost of a potential outage. Companies that have traditional physical environments typically must duplicate their infrastructure to ensure the availability of spare capacity in the event of a disaster. The infrastructure needs to be procured, installed, and maintained so that it is ready to support the anticipated capacity requirements. During normal operations, the infrastructure typically is under-utilized or over-provisioned.

With Amazon Web Services (AWS), your company can scale up its infrastructure on an as-needed, pay-as-you-go basis. You get access to the same highly secure, reliable, and fast infrastructure that Amazon uses to run its own global network of websites. AWS also gives you the flexibility to quickly change and optimize resources during a DR event, which can result in significant cost savings.

This whitepaper outlines best practices to improve your DR processes, from minimal investments to full -scale availability and fault tolerance, and shows you how you can use AWS services to reduce cost an d ensure business continuity during a DR event.

Recovery Time Objective and Recovery Point Objective

This whitepaper uses two common industry terms for disaster planning:

Recovery time objective (RTO) 1 – The time it takes after a disruption to restore a business process to its service level, as defined by the operational level agreement (OLA). For example, if a disaster occurs at 12:00 PM (noon) and the RTO is eight hours, the DR process should restore the business process to the acceptable service level by 8:00 PM.

Recovery point objective (RPO) 2 – The acceptable amount of data loss measured in time. For example, if a disaster occurs at 12:00 PM (noon) and the RPO is one hour, the system should recover all data that was in the system before 11:00 AM. Data loss will span only one hour, between 11:00 AM and 12:00 PM (noon).

A company typically decides on an acceptable RTO and RPO based on the financial impact to the business when systems are unavailable. The company determines financial impact by considering many factors, such as the loss of business and damage to its reputation due to downtime and the lack of systems availability.

IT organizations then plan solutions to provide cost-effective system recovery based on the RPO within the timeline and the service level established by the RTO.

Traditional DR Investment Practices

A traditional approach to DR involves different levels of off-site duplication of data and infrastructure. Critical business services are set up and maintained on this infrastructure and tested at regular intervals. The disaster recovery environment's location and the source infrastructure should be a significant physical distance apart to ensure that the disaster recovery environment is isolated from faults that could impact the source site.

At a minimum, the infrastructure that is required to support the duplicate environment should include the following:

- Facilities to house the infrastructure, including power and cooling.

- Security to ensure the physical protection of assets.
- Suitable capacity to scale the environment.
- Support for repairing, replacing, and refreshing the infrastructure.
- Contractual agreements with an Internet service provider (ISP) to provide Internet connectivity that can sustain bandwidth utilization for the environment under a full load.
- Network infrastructure such as firewalls, routers, switches, and load balancers.
- Enough server capacity to run all mission-critical services, including storage appliances for the supporting data, and servers to run applications and backend services such as user authentication, Domain Name System (DNS), Dynamic Host Configuration Protocol (DHCP), monitoring, and alerting.

1 From http://en.wikipedia.org/wiki/Recovery_time_objective
2 From http://en.wikipedia.org/wiki/Recovery_point_objective

AWS Services and Features Essential for Disaster Recovery

Before we discuss the various approaches to DR, it is important to review the AWS services and features that are the most relevant to disaster recovery. This section provides a summary.

In the preparation phase of DR, it is important to consider the use of services and features that support data migration and durable storage, because they enable you to restore backed-up, critical data to AWS when disaster strikes. For some of the scenarios that involve either a scaled-down or a fully scaled deployment of your system in AWS, compute resources will be required as well.

When reacting to a disaster, it is essential to either quickly commission compute resources to run your system in AWS or to orchestrate the failover to already running resources in AWS. The essential infrastructure pieces include DNS, networking features, and various Amazon Elastic Compute Cloud (Amazon EC2) features described later in this section.

Regions

Amazon Web Services are available in multiple regions around the globe, so you can choose the most appropriate location for your DR site, in addition to the site where your system is fully deployed. AWS has multiple g eneral purpose regions in the Americas, EMEA, and Asia Pacific that anyone with an AWS account can access. Special-use regions are also available for government agencies and for China. See the full list of available regions here.

Storage

Amazon Simple Storage Service (Amazon S3) provides a highly durable storage infrastructure designed for mission-critical and primary data storage. Objects are redundantly stored on multiple devices across multiple facilities within a region, designed to provide a durability of 99.999999999% (11 9s). AWS provides further protection for data retention and archiving through versioning in Amazon S3, AWS multi-factor authentication (AWS MFA), bucket policies, and AWS Identity and Access Management (IAM).

Amazon Glacier provides extremely low-cost storage for data archiving and backup. Objects (or archives, as they are known in Amazon Glacier) are optimized for infrequent access, for which retrieval times of several hours are adequate. Amazon Glacier is designed for the same durability as Amazon S3.

Amazon Elastic Block Store (Amazon EBS) provides the ability to create point-in-time snapshots of data volumes. You can use the snapshots as the starting point for new Amazon EBS volumes, and you can protect your data for long-term durability because snapshots are stored within Amazon S3. After a volume is cre ated, you can attach it to a running Amazon EC2 instance. Amazon EBS volumes provide off-instance storage that persists independently from the life of an instance and is replicated across multiple servers in an Availability Zone to prevent the loss of data from the failure of any single component.

AWS Import/Export accelerates moving large amounts of data into and out of AWS by using portable storage devices for transport. AWS Import/Export bypasses the Internet and transfers your data directly onto and off of storage devices by means of the high-speed internal network of Amazon. For data sets of significant size, AWS Import/Export is often faster than Internet transfer and more cost effective than upgrading your connectivity. You can use AWS Import/Export to migrate data into and out of Amazon S3 buckets and Amazon Glacier vaults or into Amazon EBS snapshots.

AWS Storage Gateway is a service that connects an on-premises software appliance with cloud-based storage to provide seamless and highly secure integration between your on-premises IT environment and the storage infrastructure of AWS.

AWS Storage Gateway supports three different configurations:

Gateway-cached volumes – You can store your primary data in Amazon S3 and retain your frequently accessed data locally. Gateway-cached volumes provide substantial cost savings on primary storage, minimize the need to scale your storage on-premises, and retain low-latency access to your frequently accessed data.

Gateway-stored volumes – In the event that you need low-latency access to your entire data set, you can configure your gateway to store your primary data locally, and asynchronously back up point-in-time snapshots of this data to Amazon S3. Gateway-stored volumes provide durable and inexpensive off-site backups that you can recover locally or from Amazon EC2 if, for example, you need replacement capacity for disaster recovery.

Gateway-virtual tape library (gateway-VTL) – With gateway-VTL, you can have an almost limitless collection of virtual tapes. You can store each virtual tape in a virtual tape library (VTL) backed by Amazon S3 or a virtual tape shelf (VTS) backed by Amazon Glacier. The virtual tape library exposes an industry standard iSCSI interface that provides your backup application with on-line access to the virtual tapes. When you no longer require immediate or frequent access to data contained on a virtual tape, you can use your backup application to move it from its VTL to your VTS to further reduce your storage costs.

Compute

Amazon Elastic Compute Cloud (Amazon EC2) provides resizable compute capacity in the cloud. Within minutes, you can create Amazon EC2 instances, which are virtual machines over which you have complete control. In th e context of DR, the ability to rapidly create virtual machines that you can control is critical. To describe every feature of Amazon EC2 is outside the scope of this document; instead; we focus on the aspects of Amazon EC2 that are most relevant to DR.

Amazon Machine Images (AMIs) are preconfigured with operating systems, and some preconfigured AMIs might also

include application stacks. You can also configure your own AMIs. In the context of DR, we strongly recommend that you configure and identify your own AMIs so that they can launch as part of your recovery procedure. Such AMIs should be preconfigured with your operating system of choice plus appropriate pieces of the application stack.

Availability Zones are distinct locations that are engineered to be insulated from failures in other Availability Zones. They also provide inexpensive, low-latency network connectivity to other Availability Zones in the same region. By launching instances in separate Availability Zones, you can protect your applications f rom the failure of a single location. Regions consist of one or more Availability Zones.

The Amazon EC2 VM Import Connector virtual appliance enables you to import virtual machine images from your existing environment to Amazon EC2 instances.

Networking

When you are dealing with a disaster, it's very likely that you will have to modify network settings as you r system is failing over to another site. AWS offers several services and features that enable you to manage and modify network settings.

Amazon Route 53 is a highly available and scalable Domain Name System (DNS) web service. It gives developers and businesses a reliable, cost-effective way to route users to Internet applications. Amazon Route 53 includes a number of global load-balancing capabilities (which can be effective when you are dealing with DR scenarios such as DNS endpoint health checks) and the ability to failover between multiple endpoints and even static websites hosted in Amazon S3.

Elastic IP addresses are static IP addresses designed for dynamic cloud computing. However, unlike traditional static IP addresses, Elastic IP addresses enable you to mask instance or Availability Zone failures by programmatically remapping

your public IP addresses to instances in your account in a particular region. For DR, you can also pre -allocate some IP addresses for the most critical systems so that their IP addresses are already known before disaster strikes. This can simplify the execution of the DR plan.

Elastic Load Balancing automatically distributes incoming application traffic across multiple Amazon EC2 instances. It enables you to achieve even greater fault tolerance in your applications by seamlessly providing the load-balancing capacity that is needed in response to incoming application traffic. Just as you can pre-allocate Elastic IP addresses, you can pre-allocate your load balancer so that its DNS name is already known, which can simplify the execution of your DR plan.

Amazon Virtual Private Cloud (Amazon VPC) lets you provision a private, isolated section of the AWS cloud where you can launch AWS resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways. This enables you to create a VPN connection between your corporate data center and your VPC, and leverage the AWS cloud as an extension of your corporate data center. In the context of DR, you can use Amazon VPC to extend your existing network topology to the cloud; this can be especially appropriate when recovering enterprise applications that are typically on the internal network.

Amazon Direct Connect makes it easy to set up a dedicated network connection from your premises to AWS. In many cases, this can reduce your network costs, increase bandwidth throughput, and provide a more consistent n etwork experience than Internet-based connections.

Databases

For your database needs, consider using these AWS services:

Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate, and scale a relational database in the cloud. You can use Amazon RDS either in the preparation phase for DR to hold your critical data in a database that is already running, or in the recovery phase to run your production database. When you want to look at multiple regions, Amazon RDS gives you the ability to snapshot data from one region to another, and also to have a read replica running in another region.

Amazon DynamoDB is a fast, fully managed NoSQL database service that makes it simple and cost-effective to store and retrieve any amount of data and serve any level of request traffic. It has reliable throughput and single-digit, millisecond latency. You can also use it in the preparation phase to copy data to DynamoDB in another region or to Amazon S3. During the recovery phase of DR, you can scale up seamlessly in a matter of minutes with a single click or API call.

Amazon Redshift is a fast, fully managed, petabyte-scale data warehouse service that makes it simple and cost-effective to efficiently analyze all your data using your existing business intelligence tools. You can use Amazon Redshift in the preparation phase to snapshot your data warehouse to be durably stored in Amazon S3 within the same region or copied to another region. During the recovery phase of DR, you can quickly restore your data warehouse into the same region or within another AWS region.

You can also install and run your choice of database software on Amazon EC2, and you can choose from a variety of leading database systems.

For more information about database options on AWS, see Running Databases on AWS.

Deployment orchestration

Deployment automation and post-startup software installation/configuration processes and tools can be used in Amazon EC2. We highly recommend investments in this area. This can be very helpful in the recovery phase, enabling you to create the required set of resources in an automated way.

AWS CloudFormation gives developers and systems administrators an easy way to create a collection of related AWS resources and provision them in an orderly and predictable fashion. You can create templates for your environments and deploy associated collections of resources (called a stack) as needed.

AWS Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, and Docker. You can deploy your application code, and AWS Elastic Beanstalk will provision the operating environment for your applications.

AWS OpsWorks is an application management service that makes it easy to deploy and operate applications of all types and sizes. You can define your environment as a series of layers, and configure each layer as a tier of your application. AWS OpsWorks has automatic host replacement, so in the event of an instance failure it will be automatically replaced. You can use AWS OpsWorks in the preparation phase to template your environment, and you can combine it with AWS

CloudFormation in the recovery phase. You can quickly provision a new stack from the stored configuration that supports the defined RTO.

Security and compliance

There are many security-related features across the AWS services. We recommend that you review the Security Best Practices whitepaper. AWS also provides further risk and compliance information in the AWS Security Center. A full discussion of security is out of scope for this paper.

Example Disaster Recovery Scenarios with AWS
This section outlines four DR scenarios that highlight the use of AWS and compare AWS with traditional DR methods. The following figure shows a spectrum for the four scenarios, arranged by how quickly a system can be available to users after a DR event.

Figure 1: Spectrum of Disaster Recovery Options

AWS enables you to cost-effectively operate each of these DR strategies. It's important to note that these are just examples of possible approaches, and variations and combinations of these are possible. If your application is already running on AWS, then multiple regions can be employed and the same DR strategies will still apply.

Backup and Restore
In most traditional environments, data is backed up to tape and sent off -site regularly. If you use this method, it can take a long time to restore your system in the event of a disruption or disaster. Amazon S3 is an ideal destination for backup data that might be needed quickly to perform a restore. Transferring data to and from Amazon S3 is typically done through the network, and is therefore accessible from any location. There are many commercial and open-source backup solutions that integrate with Amazon S3. You can use AWS Import/Export to transfer very large data sets by shipping storage devices directly to AWS. For longer-term data storage where retrieval times of several hours are adequate, there is Amazon Glacier, which has the same durability model as Amazon S3. Amazon Glacier is a low-cost alternative starting from $0.01/GB per month. Amazon Glacier and Amazon S3 can be used in conjunction to produce a tiered backup solution.

AWS Storage Gateway enables snapshots of your on-premises data volumes to be transparently copied into Amazon S3 for backup. You can subsequently create local volumes or Amazon EBS volumes from these snapshots.

Storage-cached volumes allow you to store your primary data in Amazon S3, but keep your frequently accessed data local for low-latency access. As with AWS Storage Gateway, you can snapshot the data volumes to give highly durable backup. In the event of DR, you can restore the cache volumes either to a second site running a storage cache gateway or to Amazon EC2.

You can use the gateway-VTL configuration of AWS Storage Gateway as a backup target for your existing backup management software. This can be used as a replacement for traditional magnetic tape backup.

For systems running on AWS, you also can back up into Amazon S3. Snapshots of Amazon EBS volumes, Amazon RDS databases, and Amazon Redshift data warehouses can be stored in Amazon S3. Alternatively, you can copy files directly into Amazon S3, or you can choose to create backup files and copy those to Amazon S3. There are many backup solutions that store data directly in Amazon S3, and these can be used f rom Amazon EC2 systems as well.

The following figure shows data backup options to Amazon S3, from either on-site infrastructure or from AWS.

Figure 2: Data Backup Options to Amazon S3 from On-Site Infrastructure or from AWS.

Of course, the backup of your data is only half of the story. If disaster strikes, you'll need to recover your data quickly and reliably. You should ensure that your systems are configured to retain and secure your data, and you should test your data recovery processes.

The following diagram shows how you can quickly restore a system from Amazon S3 backups to Amazon EC2.

Figure 3: Restoring a System from Amazon S3 Backups to Amazon EC2

Key steps for backup and restore:
1. Select an appropriate tool or method to back up your data into AWS.
2. Ensure that you have an appropriate retention policy for this data.
3. Ensure that appropriate security measures are in place for this data, including encryption and access policies.
4. Regularly test the recovery of this data and the restoration of your system.

Pilot Light for Quick Recovery into AWS
The term pilot light is often used to describe a DR scenario in which a minimal version of an environment is always running in the cloud. The idea of the pilot light is an analogy that comes from the gas heater. In a gas heater, a small flame that's always on can quickly ignite the entire furnace to heat up a house.

This scenario is similar to a backup-and-restore scenario. For example, with AWS you can maintain a pilot light by configuring and running the most critical core elements of your system in AWS. When the time comes for recovery, you can rapidly provision a full-scale production environment around the critical core.

Infrastructure elements for the pilot light itself typically include your database servers, which would replicat e data to Amazon EC2 or Amazon RDS. Depending on the system, there might be other critical data outside of the database that

needs to be replicated to AWS. This is the critical core of the system (the pilot light) around which all other infrastructure pieces in AWS (the rest of the furnace) can quickly be provisioned to restore the complete system.

To provision the remainder of the infrastructure to restore business-critical services, you would typically have some pre-configured servers bundled as Amazon Machine Images (AMIs), which are ready to be started up at a moment's notice. When starting recovery, instances from these AMIs come up quickly with their pre-defined role (for example, Web or App Server) within the deployment around the pilot light. From a networking point of view, you have two main options for provisioning:

- Use Elastic IP addresses, which can be pre-allocated and identified in the preparation phase for DR, and associate them with your instances. Note that for MAC address-based software licensing, you can use elastic network interfaces (ENIs), which have a MAC address that can also be pre-allocated to provision licenses against. You can associate these with your instances, just as you would with Elastic IP addresses.
- Use Elastic Load Balancing (ELB) to distribute traffic to multiple instances. You would then update your DNS records to point at your Amazon EC2 instance or point to your load balancer using a CNAME. We recommend this option for traditional web-based applications.

For less critical systems, you can ensure that you have any installation packages and configuration information available in AWS, for example, in the form of an Amazon EBS snapshot. This will speed up the application server setup, because you can quickly create multiple volumes in multiple Availability Zones to attach to Amazon EC2 instances. You can then install and configure accordingly, for example, by using the backup-and-restore method.

The pilot light method gives you a quicker recovery time than the backup-and-restore method because the core pieces of the system are already running and are continually kept up to date. AWS enables you to automate the provisioning and configuration of the infrastructure resources, which can be a significant benefit to save time and help protect against human errors. However, you will still need to perform some installation and configuration tasks to recover the applications fully.

Preparation phase

The following figure shows the preparation phase, in which you need to have your regularly changi ng data replicated to the pilot light, the small core around which the full environment will be started in the recovery phase. Your less frequently updated data, such as operating systems and applications, can be periodically updated and stored as AMIs.

Figure 4: The Preparation Phase of the Pilot Light Scenario

Key steps for preparation:

1. Set up Amazon EC2 instances to replicate or mirror data.
2. Ensure that you have all supporting custom software packages available in AWS.
3. Create and maintain AMIs of key servers where fast recovery is required.
4. Regularly run these servers, test them, and apply any software updates and configuration changes.
5. Consider automating the provisioning of AWS resources.

Recovery phase

To recover the remainder of the environment around the pilot light, you can start your systems from the AMIs within minutes on the appropriate instance types. For your dynamic data servers, you can resize them to handle production volumes as needed or add capacity accordingly. Horizontal scaling often is the most cost-effective and scalable approach to add capacity to a system. For example, you can add more web servers at peak times. However, you can also choose larger Amazon EC2 instance types, and thus scale vertically for applications such as relational databases. From a networking perspective, any required DNS updates can be done in parallel.

After recovery, you should ensure that redundancy is restored as quickly as possible. A failure of your DR environment shortly after your production environment fails is unlikely, but you should be aware of this risk. Continue to take regular backups of your system, and consider additional redundancy at the data layer.

The following figure shows the recovery phase of the pilot light scenario.

Figure 5: The Recovery Phase of the Pilot Light Scenario.

Key steps for recovery:

1. Start your application Amazon EC2 instances from your custom AMIs.
2. Resize existing database/data store instances to process the increased traffic.
3. Add additional database/data store instances to give the DR site resilience in the data tier; if you are using Amazon RDS, turn on Multi-AZ to improve resilience.
4. Change DNS to point at the Amazon EC2 servers.
5. Install and configure any non-AMI based systems, ideally in an automated way.

Warm Standby Solution in AWS

The term warm standby is used to describe a DR scenario in which a scaled-down version of a fully functional environment is always running in the cloud. A warm standby solution extends the pilot light elements and preparation. It further decreases the recovery time because some services are always running. By identifying your business-critical systems, you can fully duplicate these systems on AWS and have them always on.

These servers can be running on a minimum-sized fleet of Amazon EC2 instances on the smallest sizes possible. This solution is not scaled to take a full-production load, but it is fully functional. It can be used for non-production work,

such as testing, quality assurance, and internal use.

In a disaster, the system is scaled up quickly to handle the production load. In AWS, this can be done by adding more instances to the load balancer and by resizing the small capacity servers to run on larger Amazon EC2 instance typ es. As stated in the preceding section, horizontal scaling is preferred over vertical scaling.

Preparation phase

The following figure shows the preparation phase for a warm standby solution, in which an on-site solution and an AWS solution run side-by-side.

Figure 6: The Preparation Phase of the Warm Standby Scenario.

Key steps for preparation:

1. Set up Amazon EC2 instances to replicate or mirror data.
2. Create and maintain AMIs.
3. Run your application using a minimal footprint of Amazon EC2 instances or AWS infrastructure.
4. Patch and update software and configuration files in line with your live environment.

Recovery phase

In the case of failure of the production system, the standby environment will be scaled up for production load , and DNS records will be changed to route all traffic to AWS.

Figure 7: The Recovery Phase of the Warm Standby Scenario.

Key steps for recovery:

1. Increase the size of the Amazon EC2 fleets in service with the load balancer (horizontal scaling).
2. Start applications on larger Amazon EC2 instance types as needed (vertical scaling).
3. Either manually change the DNS records, or use Amazon Route 53 automated health checks so that all traffic is routed to the AWS environment.
4. Consider using Auto Scaling to right-size the fleet or accommodate the increased load.
5. Add resilience or scale up your database.

Multi-Site Solution Deployed on AWS and On-Site

A multi-site solution runs in AWS as well as on your existing on-site infrastructure, in an active-active configuration. The data replication method that you employ will be determined by the recovery point that you choose. For more information about recovery point options, see the Recovery Time Objective and Recovery Point Objective section in this whitepaper.

In addition to recovery point options, there are various replication methods, such as synchronous and asynchronous methods. For more information, see the Replication of Data section in this whitepaper.

You can use a DNS service that supports weighted routing, such as Amazon Route 53, to route production traffic to different sites that deliver the same application or service. A proportion of traffic will go to your infrastructure in AWS, and the remainder will go to your on-site infrastructure.

In an on-site disaster situation, you can adjust the DNS weighting and send all traffic to the AWS servers. The capacity of the AWS service can be rapidly increased to handle the full production load. You can use Amazon EC2 Auto Scaling to automate this process. You might need some application logic to detect the failure of the primary database services and cut over to the parallel database services running in AWS.

The cost of this scenario is determined by how much production traffic is handled by AWS during normal operation. In the recovery phase, you pay only for what you use for the duration that the DR environment is required at full scale. You can further reduce cost by purchasing Amazon EC2 Reserved Instances for your “always on” AWS servers.

Preparation phase

The following figure shows how you can use the weighted routing policy of the Amazon Route 53 DNS to route a portion of your traffic to the AWS site. The application on AWS might access data sources in the on-site production system. Data is replicated or mirrored to the AWS infrastructure.

Figure 8: The Preparation Phase of the Multi-Site Scenario.

Key steps for preparation:

1. Set up your AWS environment to duplicate your production environment.
2. Set up DNS weighting, or similar traffic routing technology, to distribute incoming requests to both sites. Configure automated failover to re-route traffic away from the affected site.

Recovery phase

The following figure shows the change in traffic routing in the event of an on-site disaster. Traffic is cut over to the AWS infrastructure by updating DNS, and all traffic and supporting data queries are supported by the AWS infrastructure.

Figure 9: The Recovery Phase of the Multi-Site Scenario Involving On-Site and AWS Infrastructure.

Key steps for recovery:

1. Either manually or by using DNS failover, change the DNS weighting so that all requests are sent to the AWS site.
2. Have application logic for failover to use the local AWS database servers for all queries.
3. Consider using Auto Scaling to automatically right-size the AWS fleet.

You can further increase the availability of your multi-site solution by designing Multi-AZ architectures. For more information about how to design applications that span multiple availability zones, see the Building Fault-Tolerant Applications on AWS whitepaper.

AWS Production to an AWS DR Solution Using Multiple AWS Regions

Applications deployed on AWS have multi-site capability by means of multiple Availability Zones. Availability Zones are distinct locations that are engineered to be insulated from each other. They provide inexpensive, low-latency network connectivity within the same region.

Some applications might have an additional requirement to deploy their components using multiple regions; this can be a business or regulatory requirement.

Any of the preceding scenarios in this whitepaper can be deployed using separate AWS regions. The advantages for both production and DR scenarios include the following:

- You don't need to negotiate contracts with another provider in another region
- You can use the same underlying AWS technologies across regions
- You can use the same tools or APIs

For more information, see the Migrating AWS Resources to a New Region whitepaper.

Replication of Data

When you replicate data to a remote location, you should consider these factors:

- Distance between the sites – Larger distances typically are subject to more latency or jitter.
- Available bandwidth – The breadth and variability of the interconnections.
- Data rate required by your application – The data rate should be lower than the available bandwidth.
- Replication technology – The replication technology should be parallel (so that it can use the network effectively).

There are two main approaches for replicating data: synchronous and asynchronous.

Synchronous replication

Data is atomically updated in multiple locations. This puts a dependency on network performance and availability. In AWS, Availability Zones within a region are well connected, but physically separated. For example, when deployed in Multi-AZ mode, Amazon RDS uses synchronous replication to duplicate data in a second Availability Zone. This ensures that data is not lost if the primary Availability Zone becomes unavailable.

Asynchronous replication

Data is not atomically updated in multiple locations. It is transferred as network performance and availability allows, and the application continues to write data that might not be fully replicated yet.

Many database systems support asynchronous data replication. The database replica can be located remotely, and the replica does not have to be completely synchronized with the primary database server. This is acceptable in many scenarios, for example, as a backup source or reporting/read-only use cases. In addition to database systems, you can also extend it to network file systems and data volumes.

We recommend that you understand the replication technology used in your software solution. A detailed analysis of replication technology is beyond the scope of this paper.

AWS regions are completely independent of each other, but there are no differences in the way you access them and use them. This enables you to create DR processes that span continental distances, without the challenges or costs that this would normally incur. You can back up data and systems to two or more AWS regions, allowing service restoration even in the face of extremely large-scale disasters. You can use AWS regions to serve your users around the globe with relatively low complexity to your operational processes.

Failing Back from a Disaster

Once you have restored your primary site to a working state, you will need to restore your normal service, which is often referred to as a "fail back." Depending on your DR strategy, this typically means reversing the flow of data replication so that any data updates received while the primary site was down can be replicated back, without the loss of data. The following steps outline the different fail-back approaches:

Backup and restore

1. Freeze data changes to the DR site.
2. Take a backup.
3. Restore the backup to the primary site.
4. Re-point users to the primary site.
5. Unfreeze the changes.

Pilot light, warm standby, and multi-site

1. Establish reverse mirroring/replication from the DR site back to the primary site, once the primary site has caught up with the changes.
2. Freeze data changes to the DR site.

3. Re-point users to the primary site.

4. Unfreeze the changes.

Improving Your DR Plan

This section describes the important steps you should follow to establish a strong DR plan.

Testing

After your DR solution is in place, it needs to be tested. You can test frequently, which is one of the key advantages of deploying on AWS. “Game day” is when you exercise a failover to the DR environment, ensuring that sufficient documentation is in place to make the process as simple as possible should the real event take place. Spinning up a duplicate environment for testing your game-day scenarios is quick and cost-effective on AWS, and you typically don’t need to touch your production environment. You can use AWS CloudFormation to deploy complete environments on AWS. This uses a template to describe the AWS resources and any associated dependencies or runtime parameters that are required to create a full environment.

Differentiating your tests is key to ensuring that you are covered against a multitude of different types of disasters. The following are examples of possible game-day scenarios:

- Power loss to a site or a set of servers
- Loss of ISP connectivity to a single site
- Virus impacting core business services that affects multi-sites
- User error that causes the loss of data, requiring a point-in-time recovery

Monitoring and alerting

You need to have regular checks and sufficient monitoring in place to alert you when your DR environment has been impacted by server failure, connectivity issues, and application issues. Amazon CloudWatch provides access to metrics about AWS resources, as well as custom metrics that can be application–centric or even business-centric. You can set up alarms based on defined thresholds on any of the metrics and, where required, you can set up Amazon SNS to send alerts in case of unexpected behavior.

You can use any monitoring solutions on AWS, and you can also continue to use any existing monitoring and alerting tools that your company uses to monitor your instance metrics, as well as guest OS stats and application health.

Backups

After you have switched to your DR environment, you should continue to make regular backups. Testing backup and restore regularly is essential as a fall-back solution.

AWS gives you the flexibility to perform frequent, inexpensive DR tests without needing the DR infrastructure to be “always on.”

User access

You can secure access to resources in your DR environment by using AWS Identity and Access Management (IAM). With IAM, you can create role-based and user-based security policies that segregate user responsibilities and restrict user access to specified resources and tasks in your DR environment.

System access

You can also create roles for your Amazon EC2 resources, so that only users who are assigned to specified roles can perform defined actions on your DR environment, such as accessing an Amazon S3 bucket or re-pointing an Elastic IP address.

Automation

You can automate the deployment of applications onto AWS-based servers and your on-premises servers by using configuration management or orchestration software. This allows you to handle application and configuration change management across both environments with ease. There are several popular orchestration software options available. For a list of solution providers, see the AWS Partner Directory.3

AWS CloudFormation works in conjunction with several tools to provision infrastructure services in an automated way. Higher levels of abstraction are also available with AWS OpsWorks or AWS Elastic Beanstalk. The overall goal is to automate your instances as much as possible. For more information, see the Architecting for the Cloud: Best Practices whitepaper.

You can use Auto Scaling to ensure that your pool of instances is appropriately sized to meet the demand based on the metrics that you specify in AWS CloudWatch. This means that in a DR situation, as your user base starts to use the environment more, the solution can scale up dynamically to meet this increased demand. After the event is over and usage potentially decreases, the solution can scale back down to a minimum level of servers.

Software Licensing and DR

Ensuring that you are correctly licensed for your AWS environment is as important as licensing for any other environment. AWS provides a variety of models to make licensing easier for you to manage. For example, “Bring Your Own License” is possible for several software components or operating systems. Alternately, there is a range of software for which the cost of the license is included in the hourly charge. This is known as “License included .”

“Bring your Own License” enables you to leverage your existing software investments during a disaster. “License included” minimizes up-front license costs for a DR site that doesn’t get used on a day-to-day basis.

If at any stage you are in doubt about your licenses and how they apply to AWS, contact your license reseller.

Conclusion

Many options and variations for DR exist. This paper highlights some of the common scenarios, ranging from simple backup and restore to fault tolerant, multi-site solutions. AWS gives you fine-grained control and many building blocks to build the appropriate DR solution, given your DR objectives (RTO and RPO) and budget. The AWS services are available on-demand, and you pay only for what you use. This is a key advantage for DR, where significant infrastructure is needed quickly, but only in the event of a disaster.

This whitepaper has shown how AWS provides flexible, cost-effective infrastructure solutions, enabling you to have a more effective DR plan.

3 Solution providers can be found at http://aws.amazon.com/solutions/solution-providers/

Further Reading

- Amazon S3 Getting Started Guide: http://docs.amazonwebservices.com/AmazonS3/latest/gsg/
- Amazon EC2 Getting Started Guide: http://docs.amazonwebservices.com/AWSEC2/latest/GettingStartedGuide/
- AWS Partner Directory (for a list of AWS solution providers): http://aws.amazon.com/solutions/solution-providers/
- AWS Security and Compliance Center: http://aws.amazon.com/security/
- AWS Architecture Center: http://aws.amazon.com/architecture
- Whitepaper: Designing Fault-Tolerant Applications in the AWS Cloud
- Other AWS technical whitepapers: http://aws.amazon.com/whitepapers

Document Revisions

We've made the following changes to this whitepaper since its original publication in January, 2012:

- Updated information about AWS regions
- Added information about new services: Amazon Glacier, Amazon Redshift, AWS OpsWorks, AWS Elastic Beanstalk, and Amazon DynamoDB
- Added information about elastic network interfaces (ENIs)
- Added information about various features of AWS services for DR scenarios using multiple AWS regions
- Added information about AWS Storage Gateway virtual tape libraries

4 Approaches of Backup and DR Techniques using Amazon Cloud

http://blog.blazeclan.com/4-approaches-backup-disaster-recovery-explained-amazon-cloud/ December 09, 2014

For a better understanding of Amazon Web Services read white papers on AWS features. Acknowledgements: Backup and Disaster Recovery by Glen Robinson, Ianni Vamvadelis ...

What can I expect to learn from this blog?

This blog offers an introduction to the 4 types of Backup and DR solutions that AWS has to offer. Enterprises use a mixture of these techniques. If you understand the basics of cloud computing, and want to know how you can protect your data from a disaster, to ensure business continuity; you have come to the right place.

A disaster can be defined as any phenomenon that disrupts business continuity. It is not only any natural calamity such as an earthquake or a flood, it can be any hardware or software failure, a network or power outage, physical damage to a building like fire or human error.

Our data is the most precious asset that we have and protecting it is our top priority. Creating backups of our data to an off shore data center, so that in the event of an on premise failure we can switch over to our backup, is a prime focus for business continuity. As AWS says, âDisaster recovery is a continual process of analysis and improvement, as business and systems evolve.Â For each business service, customers need to establish an acceptable recovery point and time, and then build an appropriate DR solution.â

Backup and DR on Cloud reduces costs by half as compared to maintaining your own redundant data centers. And if you think about it, itâs really not that surprising. Imagine the kind of cost you would entail in buying and maintaining servers and data centers, providing secure and stable connectivity and not to mention keeping them secure. You would also be under utilizing severs; and in times of unpredictable traffic rise it would be strenuous to set up new ones. To all these cloud provides a seamless transition reducing cost dramatically.

4 standard Approaches of Backup and Disaster Recovery using Amazon Cloud:Â

Â 1. Backup and Recovery: To recover your data in the event of any disaster, you must first have your data periodically backed up from your system to AWS. Backing up of data can be done through various mechanisms and your choice will be based on the RPO (Recovery Point Objective- So if your disaster struck at 2 pm and your RPO is 1 hr, your Backup & DR will restore all data till 1 pm.) that will suit your business needs. AWS offers AWS Direct connect and Import Export services that allow for faster backup. For example, if you have a frequently changing database like say a stock market, then you will need a very high RPO. However if your data is mostly static with a low frequency of changes, you can opt for periodic incremental backup. Once your backup mechanisms are activated you can pre-configure AMIs (operating systems & application software). Now when a disaster strikes, EC2Â (Elastic Compute Capacity) Â instances in the Cloud using EBS (Elastic Block Store) coupled with AMIs can access your data from the S3 (Simple Storage Service) buckets to revive your system and keep it going.

2. Pilot Light Approach: The name pilot light comes from the gas heater analogy. Just as in a heater you have a small flame that is always on, and can quickly ignite the entire furnace; a similar approach can bethought of about your data system. In the preparatory phase your on premise database server mirrors data to data volumes on AWS. The database server on cloud is always activated for frequent or continuous incremental backup. This core area is the pilot from our gas heater analogy. The application and caching server replica environments are created on cloud and kept in standby mode as very few changes take place over time. These AMIs can be updated periodically. This is the entire furnace from our example. If the on premise system fails, then the application and caching servers get activated; further users are rerouted using elastic IP addresses to the ad hoc environment on cloud. Your Recovery takes just a few minutes.

3. Warm Standby Approach: This Technique is the next level of the pilot light, reducing recovery time to almost zero. Your application and caching servers are set up and always activated based on your business critical activities but only a minimum sized fleet of EC2 instances are dedicated. The backup system is not capable of handling production load, but can be used for testing, quality assurance and other internal uses. In the event of a disaster, when your on premise dataÂ centerÂ fails, two things happen. Firstly multiple EC2 instances are dedicated (vertical and horizontal scaling) to bring your application and caching environment up to production load. ELB and Auto Scaling (for distributing traffic) are used to ease scaling up. Secondly using Amazon Route 53 user traffic is rerouted instantly using elastic IP addresses and there is instant recovery of your system with almost zero down time.

4. Multi-Site Approach:Â Well this is the optimum technique in backup and DR and is the next step after warm standby. All activities in the

preparatory stage are similar to a warm standby; except that AWS backup on Cloud is also used to handle some portions of the user traffic using Route 53. When a disaster strikes, the rest of the traffic that was pointing to the on premise servers are rerouted to AWS and using auto scaling techniques multiple EC2 instances are deployed to handle full production capacity. You can further increase the availability of your multi-site solution by designing Multi-AZ architectures.

The diagram below shows exactly how the Pilot Light Approach works, in the occurrence of a disaster :

For a better understanding of Amazon Web Services read white papers on AWS features.

To see more about our Backup and DR solution and other solutions visitÂ our website.

Backupand Restoreof'SAPSystems 'on Amazon'Web ...

http://awsmedia.s3.amazonaws.com/backup_and_recovery_of_sap_systems_on_aws_for_linux_maxdb_and_db2_v1.6.pdf December 09, 2014

Amazon'Web'Services'Infrastructure ... • SAPnote1377148 66!FAQ:!SAPMaxDB!backup/recovery! Answers!toFrequently!Asked!Questions!on!SAPMaxDB!backup!and!recovery!

Backup and Restore of SAP Systems on Amazon Web Services Infrastructure For MaxDB and DB2 LUW Databases on Linux

Authors:
Amazon Web Services

sap--on--aws@amazon.com

Protera Technologies

http://www.protera.biz

Version:

1.6 – March 2012

Table of Contents

Example
1:
database
backup
to
an
EBS
backup
file
system
.....
14

Example
1a:
full
online
data
and
log
backup
for
MaxDB
.....
15

Create
MaxDB
backup
templates
.....
15

Back
up
the
database
using
Database
Studio
.....
16

Back
up
the
database
transaction
logs
using
Database
Studio
.....
16

Back
up
the
database
using
DBMCLI
.....
16

Back
up
the
database
log
using
DBMCLI
.....
16

Schedule
hourly
automatic
log
backup
using
Database
Studio
.....
16

Create
a
snapshot
to
send
the
backup
to
Amazon

S3
.....
17

Example
1b:
full
online
backup
for
DB2
LUW
.....
17

Steps
to
enable
rollforward
recovery
and
set
up
DB2
logfile
management:
.....
17

Option
1:
back
up
the
database
directly
from
the
SAP
system
.....
18

Option
2:
back
up
the
database
using
the
CLI
.....
18

DB2
transaction
log
file
management
.....
19

Create
a
snapshot
to
send
the
backup
to
Amazon
S3
.....
19

Example
2:
OS
backup
to
S3
using
the
TAR
command
.....
20

Examples
for
restoring
SAP
systems
using
AWS
infrastructure
.....
22

Example
1:
restore
OS

Prerequisite Documents

The following information should be read carefully before continuing with this guide. Especially, this

document cannot serve as a replacement for the MaxDB and DB2 Backup and Restore information

resources that are referenced below.

SAP on Amazon Web Services

- SAP on AWS Implementation Guide http://aws.amazon.com/sap

  Information on deploying SAP systems on AWS Infrastructure

- SAP note 1588667 – SAP on Amazon Web Services (AWS)

  Entry SAP note for Amazon Web Services

- SAP note 1677381 – Backup and Restore guidelines

  Errata and feedback reference for this guide

SAP on

MaxDB

- MaxDB Backup and Restore

SAP SDN page on MaxDB Backup and Restore

- SAP Library – Database Studio – Backing up Databases: Overview

SAP Library documentation on MaxDB Backup and Restore

- SAP note 1377148 ---- FAQ: SAP MaxDB backup/recovery

Answers to Frequently Asked Questions on SAP MaxDB backup and recovery

- SAP note 767598 ---- Available SAP MaxDB documentation

General overview of SAP MaxDB documentation

SAP on DB2 UDB

- Database Administration Guide "SAP on IBM DB2 for Linux, Unix and Windows"

This document provides specific information about the administration of IBM DB2 for

Linux,

UNIX,
and
Windows
(in
the
following
referred
to
as
DB2)
in
an
SAP
environment.
In
addition,
it

provides
references
to
additional
documentation
and
guidelines
as
well
as
recommendations

from
SAP
that
are
only
available
in
this
document.
It
also
helps
you
to
plan,
install,
and
maintain

SAP
systems
and
the
database.

• IBM
DB2
Universal
Database
for
UNIX
and
Windows
--
New
Log
File
Management

This
documentation
describes
the
concepts
of
the
new
DB2
log
file
management
and
how
you

can
migrate
from
the
existing
SAP
DB2
log
file
management
solution
to
the
DB2
V8.2
log
file

management solution.

• A Practical Guide to Backup and Recovery of IBM DB2 for Linux, UNIX and Windows in SAP Environments

Introduction to the basics of backup and recovery for DB2 databases. This guide explains the architecture for backup and recovery, the most relevant commands and discusses the DB2 log file management. In addition, the integration of the DB2 backup and recovery architecture into SAP NetWeaver is described.

Scope of this Document

This document will cover backing up and restoring of

a SAP ECC 6.0 system on MaxDB or DB2 on SLES in the Amazon AWS environment. Steps for performing both OS and Database backups for MaxDB and DB2 are provided.

Amazon Elastic Block Store (EBS) snapshots are point in time images of volumes which are persisted to Amazon S3. These snapshots can be used as the starting point for new Amazon EBS volumes, and protect data for long--term durability. The same snapshot can be used to instantiate as many volumes as desired.

Database and OS backups can be accomplished via provided OS and DBMS

backup tools and in addition

utilizing Amazon EBS snapshots to secure these backups. Complete offline system backups to a so called

Amazon machine Image (AMI) will also be described.

Components for SAP Backup and Restore on AWS infrastructure

Apart from an SAP system on a platform that is supported by SAP on Amazon Web Services (AWS)

infrastructure, the following AWS products are required to perform the backup and restore operations

that are described in this document.

- Amazon Elastic Compute Cloud (Amazon EC2)
- Amazon Elastic Block Storage (EBS)
- Amazon Simple Storage Service (Amazon S3)

- Indirectly:

Amazon Virtual Private Cloud (Amazon VPC)

Amazon Elastic Compute Cloud (EC2)

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. Amazon EC2 presents a virtual computing environment, allowing one to use web service interfaces to launch instances with a variety of pre--imaged operating systems, load them with an own custom application environment and manage network's access permissions. A customized image can be persisted as own Amazon Machine Image (AMI), and can be redeployed to as many or few instances as desired.

Amazon EC2

provides different instance types to meet different computing needs. The specific Amazon

EC2 instance types that are currently supported for SAP application deployments are listed in SAP note

1588667.

Further information on Amazon EC2 can be found at http://aws.amazon.com/ec2/.

Amazon Simple Storage Service (Amazon S3)

Amazon S3 is storage for the Internet. Amazon S3 provides a simple web services interface that can be

used to store and retrieve any amount of data, at any time, from anywhere on the web. The service is

also designed

to provide 99.999999999% durability and 99.99% availability over a given year. These and other Amazon S3 properties make it the ideal storage for enterprise backups.

Further information on Amazon S3 can be found at http://aws.amazon.com/s3/.

Amazon Elastic Block Storage (EBS)

Amazon Elastic Block Store (EBS) provides block level storage volumes for use with Amazon EC2 instances. EBS volumes are off--instance storage that persists independently from the life of an EC2 instance. EBS provides highly available, highly reliable storage volumes that can be attached to a running Amazon EC2 instance and exposed as a device within

the instance.

EBS also provides the ability to create point--in--time snapshots of volumes, which are persisted to Amazon S3. These snapshots can be used as the starting point for new EBS volumes, and protect data for long--term durability. The same snapshot can be used to instantiate as many volumes as required.

Further information on EBS can be found at http://aws.amazon.com/ebs/.

Amazon Virtual Private Cloud (VPC)

Amazon Virtual Private Cloud (Amazon VPC) lets one provision a private, isolated section of the Amazon Web Services (AWS) Cloud, where AWS resources like Amazon EC2 instances and EBS

volumes
can
be

launched
in
a
self--defined
virtual
network.
With
Amazon
VPC,
a
virtual
network
can
be
defined
that

closely
resembles,
and
can
be
connected
securely
to,
the
own
traditional
on--premise
network.

Amazon
VPC
is
not
a
direct
prerequisite
for
the
backup--
and
restore--operations
that
are
described
in
this

guide.
However,
SAP
systems
themselves
are
only
supported
on
Amazon
Web
Services
infrastructure

when
deployed
within
an
Amazon
VPC.

Further
information
on
Amazon
VPC
can
be
found
at
http://aws.amazon.com/vpc/.

Storage
layout
of
SAP
systems
on
EBS
volumes

Foremost,
please
refer
to
the
White
Paper

for SAP Systems on Amazon AWS as a baseline guide on building SAP systems on AWS. The most recent version of this document can accessed at http://media.amazonwebservices.com/Operating%20SAP%20Solutions%20on%20AWS%20White%20Paper.pdf.

It is recommended to separate OS, SAP, DBMS, DB Data and DB transaction log components onto different EBS volumes. In addition, separate file systems will be used to store the following types of backups on MaxDB:

1. OS, SAP and DBMS system backups (binaries, profiles, etc.)
2. Full and/or incremental database data backups
3. Database transaction log backups

This translates to the recommended file system layout for

MaxDB on linux as shown in Table 1.

Table 1: A recommended file system layout for an SAP system on MaxDB on linux

Linux File System	Description	Tag
/	OS root directory	OS--EXE
/sapmnt	SAP system, shared	OS--EXE
/sapdb	MaxDB system	OS--EXE
/usr/sap	SAP system, local	OS--EXE
/sapdb/<SID>/sapdata	Database Data files	DB--DATA
/sapdb/<SID>/saplog	Database Transaction Logs	DB--LOG
/os_exe_backups	OS, SAP and MaxDB system backups	OS--EXE--BACKUPS
/db_data_backups	MaxDB database backups	DB--DATA--BACKUPS
/db_log_backups	MaxDB transaction log backups	DB--LOG--BACKUPS

Note that file systems printed in bold belong to the usual SAP system, and the other file systems

are additional ones to store backups. The tags will be used in the following sections to

distinguish the file system groups.

As DB2 can be configured to have a separate file system to archive its transaction logs

automatically, a file system for transaction log backups like we have on MaxDB is not required

for DB2. Table 2 shows how a recommended file system layout could look like for DB2 on Linux.

Table 2: A recommended file

system
layout
for
an
SAP
system
on
DB2
on
Linux

Linux File System	Description	Tags
/	OS root directory	OS--EXE
/sapmnt	SAP system, shared	OS--EXE
/db2	DB2 LUW binaries, configuration and trace files	DB2--EXE
/usr/sap	SAP system, local	OS--EXE
/db2/<SID>/sapdata1 : /db2/<SID>/sapdata<n>	Database data files	DB2--DATA
/db2/<SID>/log_dir	Active database transaction logs	DB2--LOG
/db2/<SID>/log_archive	Archived database transaction logs	DB2--LOG--ARCHIVE
/os_exe_backups	OS, SAP and DB2 system backups	OS--EXE--BACKUPS
/db2_data_backups	DB2 Database Data Backups	DB2--DATA--BACKUPS

Note that the file system for MaxDB transaction log backups /db_log_backups in Table 1

been replaced by /db2/<SID>/log_archive for DB2 (Table 2). The content of this DB2 file

systems is managed by the DB2 RDBMS itself.

Backup and Restore procedures using AWS infrastructure

Apart from creating a full offline Amazon EC2 machine image, the following generalized procedure will

be used to create more fine--grained and online backups:

1. Create a classical backup to a separate staging file system (on EBS storage)

2. Create

an EBS snapshot of the staging file system

An EBS snapshot is automatically persisted onto highly available Amazon S3 storage. Multiple snapshots of a file system will be stored incrementally, which means that only changed blocks with respect to the previous snapshot will be stored to Amazon S3.

The following sections will apply the above described general procedure to the different backup types for SAP systems on MaxDB or DB2.

SAP on MaxDB backups using AWS Infrastructure

Figure 1 illustrates the previously described backup procedure as applied to an SAP system installed on MaxDB on Linux.

Figure 1: SAP on MaxDB, overview of backup types and procedures using AWS infrastructure

In the figure, each file system tag listed in Table 1 is represented by an EBS volume symbol. The following 3 types of backup sequences are displayed:

1. OS--EXE: backup of the OS--EXE file systems to the OS--EXE--BACKUPS file system, using an OS specific copy program like tar on Linux. Subsequent persistence into Amazon S3 by creating a snapshot of the EBS volume that holds the OS--EXE--BACKUPS file system.

2. DB--DATA: MaxDB COMPLETE or

INCREMENTAL backup to the DB--DATA--BACKUPS file system.

Subsequent persistence onto Amazon S3 by creating an EBS snapshot of the volume that holds

the DB--DATA--BACKUPS file system.

NOTE: DIRECT SNAPSHOTS OF DB--DATA AND DB--LOG VOLUMES ARE ALSO POSSIBLE USING MAXDB I/O

SUSPEND/RESUME FUNCTIONALITY, BUT THESE METHODS ARE NOT WITHIN THE SCOPE OF THIS GUIDE.

3. DB--LOG: MaxDB LOG backup to the DB--LOG--BACKUPS file system. Subsequent persistence into

Amazon S3 by creating an EBS snapshot of the volume that holds the DB--LOG--BACKUPS file

system.

NOTE: MAXDB LOG BACKUPS CAN ALSO BE AUTOMATED USING ITS AUTOSAVE LOG MECHANISM. THIS WILL

BE

DESCRIBED
LATER
IN
THIS
GUIDE.

SAP
on
DB2
backups
using
AWS
Infrastructure

Similar
as
for
MaxDB,
Figure
2
displays
the
data
types
and
its
associated
backups
of
an
SAP
system
on

DB2
on
Linux.

Figure
2:
SAP
on
DB2,
overview
of
backup
types
and
procedures
using
AWS
infrastructure

As
mentioned
before
in
the
section
“Storage
layout
of
SAP
systems
on
EBS
volumes”,
DB2
functionality

is
used
to
manage
transaction
log
archiving
itself
to
the
DB2--LOG--ARCHIVE
file
system,

which also

resides on an EBS volume (Figure 2). Direct EBS snapshots can be made of this file system, so that its

contents are regularly persisted onto Amazon S3 storage. It is recommended to create an EBS snapshot

of the associated volume each time a new transaction log has been archived into the DB2--LOG--ARCHIVE

file system.

Note that EBS snapshots are written incrementally to Amazon S3, differential to the previous snapshot,

which means that previously snapshotted data are not stored over and over again, just once. This makes

the snapshot procedure fast and cost efficient.

Recapitulating for DB2, the following backup procedures can be distinguished:

1. OS--EXE: backup of the OS--EXE file systems to the OS--EXE--BACKUPS file system, using an OS specific copy program like tar on Linux. Subsequent persistence onto Amazon S3 by creating a snapshot of the EBS volume that holds the OS--EXE--BACKUPS file system.

2. DB2--DATA: DB2 full and/or incremental backups to the DB--DATA--BACKUPS file system.

Subsequent persistence onto Amazon S3 by creating an EBS snapshot of the volume that holds the DB--DATA--BACKUPS file system.

NOTE: DIRECT SNAPSHOTS

OF
DB2--DATA
AND
DB2--LOG
VOLUMES
ARE
ALSO
POSSIBLE
LEVERAGING
DB2
I/O

SUSPEND/RESUME
FUNCTIONALITY,
BUT
THESE
METHODS
ARE
NOT
WITHIN
THE
SCOPE
OF
THIS
GUIDE.

3. DB2--LOG--ARCHIVE:
direct
persistence
onto
Amazon
S3
by
creating
an
EBS
snapshot
of
the

volume
that
holds
the
DB2--LOG--ARCHIVE
file
system,
each
time
a
new
DB2
transaction
log
has

been
archived
by
the
DBMS
into
this
file
system.

Restore

For
each
backup
type,
its
last
backup
can
usually
directly
be
restored
from
its
associated
staging
file

system.
If
the
file
staging
system
is
not
accessible
anymore
or
an
older
backup
is
required,
a

new
EBS

volume
can
be
created
out
of
a
snapshot
that
was
created
in
the
past
of
the
parent
EBS
volume.
The

new
EBS
volume
can
then
be
attached
and
mounted
onto
the
Amazon
EC2
instance
where
restore
and

(database)
recovery
is
taking
place.

Common
backup
and
restore
operations
on
Amazon
EC2
instances
and
EBS

volumes

This
section
briefly
documents
some
common
operations
on
Amazon
EC2
instances
and
EBS
volumes

that
are
used
for
backup
and
restore
purposes.
The
operations
are
described
as
if
they
would
be

performed
from
the
graphical
Amazon
Management
Console,

available at

https://console.aws.amazon.com/ec2

However, all operations can be fully automated using the Amazon EC2 web service API and/or Command Line Tools. For more information please visit:

http://aws.amazon.com/documentation/ec2/

Backup: creating a new EBS volume with an empty file system

1) Create a new EBS volume

a. Log in to AWS EC2 Management Console

https://console.aws.amazon.com/ec2

1. On Volumes, click on Create Volume

2. Type the size

3. Select the same availability zone as the AWS instance to be attached

b. Select the volume

c. Click on Attach Volume

1. Select the instance

2. Choose a free device name (write this name down)

2) Create the file system

a. Log in to the instance and create the file system

```
pvcreate /dev/sdX
vgcreate vgbackup /dev/sdX
lvcreate -L <SIZE> -n backups vgbackup
mkfs.ext3 /dev/vgbackup/backups
mkdir -p /backups
mount /dev/vgbackup/backups /backups
```

NOTE: THE COMMAND HAS BEEN PROVIDED AS AN EXAMPLE; YOU CAN USE LVM2 OR DIRECT PARTITIONS TO STORE BACKUPS. THE SELECTION OF THE MOUNT POINT (/BACKUPS) IS ARBITRARY

Backup: creating an EBS snapshot onto Amazon S3 of an EBS volume

1) Make sure that the EBS volume is not written to.

NOTE: IF POSSIBLE, THE FILE SYSTEM(S)

ON
THE
EBS
VOLUME
CAN
BE
DISMOUNTED
TO
ENSURE
THAT
NO
WRITE

I/O
IS
OCCURRING.
WRITE
I/O
TO
A
FILE
SYSTEM
BEING
SNAPPED
CAN
CAUSE
INCONSISTENCIES
ON
THE
SNAPSHOT

COPY.

2) Log
in
to
AWS
EC2
Management
Console
https://console.aws.amazon.com/ec2

3) Go
to
volumes
and
select
the
volume
to
snapshot

4) Click
on
Create
Snapshot

5) Type
the
name
and
description.

NOTE:
CHOOSE
A
UNIQUE
BUT
EASILY
IDENTIFIABLE
NAME
THAT
INCLUDES
A
TIMESTAMP
AND/OR
SEQUENCE

NUMBER

6) Click
on
Yes
Create

7) You
can
monitor
the
progress
on
Snapshot
navigation
menu

Backup:
dismounting
file
system(s)
and
detaching
an
EBS

Volume

a. Log in to the EC2 instance, dismount and remove the volume

```
umount /backups
vgchange vgbackup –a n
vgexport vgbackup
```

b. Log in to AWS EC2 Management Console

```
https://console.aws.amazon.com/ec2
```

c. Go to Volumes and select the volume to remove

d. Click on Detach Volume

e. Click on Yes, Detach on the popup window

f. When detached, click on Delete volume

Backup: creating a full offline Amazon EC2 Amazon Machine Image (AMI)

If the SAP system can be shut down for a period of time, a full image of the system can be created. The Amazon Machine Image (AMI) offline backup

creates a snapshot of each EBS volume and stores it onto Amazon S3 storage. The AMI can be used as a golden image, to spin up new instances in case of:

1. Recovery from a complete source system loss (DR)

   To restore the root volume and bare structure of the system. Typically, the database file systems will be outdated too much to be rolled forward, so these should be restored separately as described in subsequent sections.

2. Set up of new systems

A full AMI

backup should be performed (at least) after the SAP system is installed, but best after each

low--level change of the OS, SAP or DBMS, like for instance:

- Changing the file system layout
- Upgrading OS, SAP or DBMS binaries
- Installing new additional software dependencies

Briefly, the steps to create a full offline AMI are as follows:

1. Stop the SAP and database instances
2. Select the instance and create the AMI from the AWS Management Console

The operating system will be stopped and started again automatically during the process.

3. Monitor the AMI creation until successful

completion

4. Start the database and SAP instances again

The detailed steps will be described next.

Detailed steps to create the Amazon Machine Image (AMI)

1) Log on into the OS and shutdown SAP

```
su - <sidadm>
stopsap all
```

2) Make sure that the SAP and database instances are shut down completely by monitoring the

processes and logs. If the database was not able to shut down due to still active connections,

then issue the following commands:

For MaxDB

```
su – <sidadm>
dbmcli db_offline
```

For DB2

```
su – db2<sid>
db2stop force
```

3) Log in to AWS EC2 Management Console

https://console.aws.amazon.com/ec2

4) Click ‘Instances‘

5) Click on the Instance you wish to create a AMI of

6) Right--mouse click ‘Create Image (EBS AMI)’

7) Type the AMI name and description

8) Click on ‘Create This Image‘

9) A snapshot will be created in parallel for each EBS volume in the Amazon EC2 instance

Snapshot and AMI creation can be monitored within the ‘Snapshots‘ or ‘AMIs‘ section.

As the Amazon EC2 instance will be restarted automatically after EBS AMI creation has finished,

you can also

monitor in parallel to see when the system comes back up again, by for instance

pinging its IP address.

10) Log on into the instance and startup SAP

```
su - <sidadm>
startsap all
```

The AMI should now be available for Amazon EC2 instance deployment in the ‘AMIs’ section.

Examples for backing up SAP System components using AWS infrastructure

The following sections merely provide basic examples for backing up MaxDB and DB2 LUW databases on

AWS infrastructure. Please consult the documentation referenced in section “Prerequisite Documents”

at the beginning of this guide as a complete reference.

In addition, specific references will be provided within the sections wherever suitable.

Example 1: database backup to an EBS backup file system

To recapitulate, the following general procedure will be followed:

1) Online Database backups are performed to EBS volume dedicated for backups

2) Transaction log backups are performed to EBS volume dedicated to offline (archived) DB logs

3) The EBS volumes for DB backups and offline/archived logs are snapshot on a recurring basis to Amazon S3.

The incremental snapshots ensure point in time database recovery in case of disaster.

4) Snapshots

are tagged with descriptions of backup type and time.

Backups are created using the DBMS tools listed in Table 3.

Table 3: DBMS tools used for backup

DBMS Backup Tools

MaxDB MaxDB Database Studio and dbmcli (Command Line Interface)

DB2 LUW DB2 standard commands

Example 1a: full online data and log backup for MaxDB

Make sure the DB--DATA--BACKUPS and DB--LOG--BACKUPS file systems have enough storage space, and

ensure proper read/write permissions for these file systems. The file system names listed in

Table 1 will be used in this example.

If required, please follow steps described in the section “Backup: creating a new EBS volume with an empty file system”.

Apart from the general references mentioned in section “Prerequisite Documents”, specific MaxDB Backup / Recovery examples can also be found at “SAP MaxDB HowTo” on the SAP Community Network (SCN).

Create MaxDB backup templates

a. Start the SAP MaxDB Database Studio

If not installed, it can be downloaded from

http://www.sdn.sap.com/irj/scn/maxdb--downloads

and installed on any computer for remote management

b. Add the server and

database
into
the
landscape
(if
required)

Go
to:
My
Landscape
/
Servers
/
(right
click)
/Add/”Server/Database”)

c. Type
the
server
name
or
ip:
on
“Server
Name:”
and
click
next

d. Select
the
Databases
and
click
on
Finish

e. Log
in
to
the
database
using
the
CONTROL
user

(My
Landscape
/
Servers
/
<servername/ip>/<DB>/(right
click)
/Login)

f. (Right
click)
on
the
database
and
click
on

Administration

g. Go
to
the
Backup
Tab

h. Expand
Templates

i. Create
a
Backup
template
for
FULL
Backup

j. Choose
New…

Name

<Template
Name>
can
be
FULL

Backup
Type
COMPLETE
DATA

Device Type FILE

Backup Tool NONE

Device/File /db_data_backups/<SID>_FULL

Compressed Unselect

k. Click Ok

l. Create a Backup template for LOG Backup

Name <Template Name> can be LOG

Backup Type LOG

Device Type FILE

Backup Tool NONE

Device/File /db_log_backups/<SID>_LOG

Compressed Unselect

Back up the database using Database Studio

1) Start the SAP MaxDB Database Studio

2) Log in to the database using the CONTROL user

(My Landscape / Servers / <servername/ip>/<DB>/ (right click)

/Login)

3) (Right click) on the database and click on

Administration

4) Go to the Backup Tab

5) Expand Templates

6) Right click on the template “FULL” and click on Backup

Back up the database transaction logs using Database Studio

1) Start the SAP MaxDB Database Studio

2) Log in to the database using the CONTROL user

(My Landscape / Servers / <servername/ip>/<DB>/ (right click)

/Login)

3) (Right click) on the database and click on

Administration

4) Go to the Backup Tab

5) Expand Templates

6) Right click on the template “LOG”

and click on Backup

Back up the database using DBMCLI

1) Log in to the Amazon EC2 instance

2) Execute the following commands

```
su - <sid>adm
#start an utility session
dbmcli –d <SID> -U c -uUTL
#start backup using the template FULL
backup_start FULL
```

3) Wait until the backup has completed

Back up the database log using DBMCLI

1) Log in to the Amazon EC2 instance

2) Execute the following commands

```
su - <sid>adm
#start an utility session
dbmcli –d <SID> -U c -uUTL
#start backup using the template FULL
backup_start LOG
```

3) Wait until the backup has completed

Schedule hourly automatic log backup using Database Studio

1) Start the SAP MaxDB Database Studio

2) Log in to the database using the CONTROL user

(My Landscape / Servers

/ <servername/ip>/<DB>/ (right click) /Login)

3) (Righ click) on the database and click on Administration Tasks/Automatic Log Backup

4) Select the template “LOG” from the list

5) Select “Create a log backup every <XX> minutes”

6) Type 60 on the <XX> field

7) Click on Activate

The EBS snapshot creation of the volume that holds the DB--LOG--BACKUPS file system can now be automated through a script, time--synchronized with the automatic log backup.

Create a snapshot to send the backup to Amazon S3

Please

refer to the section “Backup: creating an EBS snapshot onto Amazon S3 of an EBS volume” for the

detailed steps.

It is recommended to tag snapshots with a description like

- “<SID>_MAXDB_DATA_BACKUPS_<YYYY-MM-DD>” for the database data backups
- “<SID>_MAXDB_LOG_BACKUPS_<YYYY-MM-DD-HH-MM-SS>” for the database log backups

Example 1b: full online backup for DB2 LUW

Make sure that you have enabled your DB2 database for rollforward recovery. Rollforward recovery

mode enables you to recover from a database backup to the most recent point in time using archived

database log files and is a prerequisite for taking

DB2 online backups. If you have not configured your DB2 database for rollforward recovery mode, you cannot take online backups. Instead you will need to shutdown the SAP system and DB2 to take offline backups. For production systems, your database must be in rollforward recovery mode. For more details refer to the „Database Administration Guide SAP on IBM DB2 for Linux, UNIX, and Windows”

Steps to enable rollforward recovery and set up DB2 logfile management:

1) Stop SAP and DB2

2) Enable rollforward recovery by updating the database configuration parameter

LOGARCHMETH1:

```
su – db2<sid>
db2 update db cfg for <SID> using logarchmeth1
DISK:/db2/<SID>/log_archive
```

The database will now be placed in backup pending state. A full database backup must be taken.

From this point onwards DB2 will automatically archive log files from the /db2/<SID>/log_dir

filesystem to the /db2/<SID>/log_archive filesystem.

3) Take a full database offline backup

db2 backup database <sid> to "/backups" compress

4) Configure a retention period for your database backups and the corresponding database log

files. For example if you want to keep at least 4 database backups and you want to remove

surplus backups older than 30 days and all corresponding database

log files, configure DB2 in the following way:

```
db2 update db cfg for <sid> using NUM_DB_BACKUP 4
db2 update db cfg for <sid> using REC_HIS_RETENTN 30
db2 update db cfg for <sid> using AUTO_DEL_REC_OBJ ON
```

5) Start SAP

Make sure the DB2--DATA--BACKUPS file system has enough storage space, and proper read/write permissions. The file system names listed in Table 2 will be used for the examples.

If required, please follow steps described in the section “Backup: creating a new EBS volume with an empty file system”.

Please remember to use the references mentioned in section “Prerequisite Documents” as primary documentation. In addition the SAP on DB2 UDB for Unix and Windows (DB6)

forum on the SAP Community Network (SCN) can be consulted for other questions. The following sections merely provide examples and should not be used as a general reference.

Two options to back up the DB2 LUW database will be shown in the following sections:

- Option 1: back up the database directly from the SAP system
- Option 2: back up the database using the CLI

Option 1: back up the database directly from the SAP system

1) Log in to the SAP system with an admin user

2) Execute the transaction /nDBACOCKPIT

3) On the

left screen panel, navigate to

Jobs/DBA Planning Calendar

4) In the calendar area, select any cell representing time older than current time and click on 'Add'

button.

5) On the pop--up window, select 'Database Backup to Device' action, choose 'Online' backup

mode with 'Include Logs' option and enter /db2_data_backups in the 'Device/Directory' field

6) Click on 'Execute Immediately' button

Option 2: back up the database using the CLI

1) Log in to the Amazon EC2 instance

2) Execute the following commands:

```
su – db2<sid>
#start online compressed backup including logs
db2 backup database <sid> online to "/db2_data_backups" compress \
 include logs
```


3) Wait

until the backup has completed.

You will see the following message when the backup has completed:

```
Backup successful. The timestamp for this
backup image is : 20111220221428
```

This message is shown in the following log file, located in the backup directory:

```
<SID>.0.db2<sid>.NODE0000.CATN0000.<datetime stamp>.001
```

4) Backup the database manager configuration required to rebuild the database:

```
su – db2<sid>
cd /db2_data_backups
db2cfexp <SID>_cfg_backup.txt BACKUP
```

5) Backup the DB2 recovery history file:

```
su – db2<sid>
cp /db2/<SID>/db2<sid>/NODE0000/SQL00001/db2rhist.asc \
 /db2_data_backups
```

DB2 transaction log file management

It is recommended to configure the DB2 transaction log management as mentioned in the example

above and described in "IBM DB2 Universal

Database for UNIX and Windows -- New Log File Management”, section 2.2.1.

As from DB2 UDB V9.5, automatic log file retention management can be configured in addition. This is described in the example above and in the Database Administration Guide "SAP on IBM DB2 for Linux, Unix and Windows", section “DB2 V9.5 and Higher Only: Automatic Log File and Backup Retention”.

The EBS volume holding DB2--LOG--ARCHIVE file system should be sent to Amazon S3 on a regular basis by creating a direct snapshot, optimally each time after a transaction log was written into the DB2--LOG--ARCHIVE

file system. The snapshot can be taken directly without dismounting the file system.

Create a snapshot to send the backup to Amazon S3

Please refer to the section “Backup: creating an EBS snapshot onto Amazon S3 of an EBS volume” for the

detailed steps.

It is recommended to tag snapshots with a description like

- “<SID>_DB2_DATA_BACKUPS_<YYYY-MM-DD>” for the database data backups
- “<SID>_DB2_LOG_ARCHIVE_<YYYY-MM-DD-HH-MM-SS>” for the database archive log backups

Example 2: OS backup to S3 using the TAR command

This chapter provides an

example for creating a low--level operating system backup that can be used in

case of a full system loss (DR).

NOTE: THIS OS BACKUP CANNOT BE USED TO RESTORE THE DATABASE, AS DATABASE DATA AND TRANSACTION LOGS ARE

SPECIFICALLY EXCLUDED FROM THE BACKUP. AFTER RESTORING THIS BACKUP, RESTORE AND RECOVERY OF THE DATABASE

SHOULD FOLLOW.

The example procedure is as follows:

1) Ensure you have enough space on the file system /os_exe_backups for the OS backup.

NOTE: DATABASE DATA, TRANSACTION LOGS AND MOUNTED BACKUP FILE SYSTEMS WILL BE EXCLUDED FROM THIS

BACKUP

2) Start the OS backup

a. Logon into the OS start a tar backup on the /os_exe_backups file system

NOTE: THE USE OF COMPRESSION AFFECTS THE CPU UTILIZATION AND BACKUP TIME. CHOOSE TO USE

COMPRESSION OR NOT BY ADDING OR REMOVING THE "----GZIP" PARAMETER OF THE TAR COMMAND

For database type of MaxDB use this script as an example:

```
export       exclusion_file=/os_exe_backups/backup-exclude-dirs.txt
export       backup_file=/os_exe_backups/backup.tar.gz
export       log_file=/os_exe_backups/backup.stdout
export       error_log_file=/os_exe_backups/backup.stderr

#Create the exclusion file
#Exclude OS directories
>$exclusion_file
echo "/tmp" >>$exclusion_file
echo "/proc" >>$exclusion_file
echo "/sys" >>$exclusion_file
echo "/dev" >>$exclusion_file
#Exclude database files
echo "/sapdb/<SID>/sapdata1" >>$exclusion_file
echo "/sapdb/<SID>/saplog" >>$exclusion_file
#Exclude the backup directories
echo "/os_exe_backups" >>$exclusion_file
echo "/db_data_backups" >>$exclusion_file
echo "/db_log_backups" >>$exclusion_file

#Run the backup
cd /os_exe_backups
tar -v --gzip -cf $backup_file / --exclude-from=$exclusion_file >
$log_file 2> $error_log_file

#to monitor
#tail -f $log_file
#tail -f $error_log_file
NOTE:
```

THE COMMAND HAS BEEN PROVIDED AS AN

EXAMPLE, PLEASE TEST AND CHANGE AS REQUIRED.

For database type of DB2 use this script as an example:


```
export        exclusion_file=/os-exe-backups/exclude-dirs.txt
export        backup_file=/os-exe-backups/backup.tar.gz
export        log_file=/os-exe-backups/backup.stdout
export        error_log_file=/os-exe-backups/backup.stderr


#Create the exclusion file
#Exclude OS directories
>$exclusion_file
echo "/tmp" >>$exclusion_file
echo "/proc" >>$exclusion_file
echo "/sys" >>$exclusion_file
echo "/dev" >>$exclusion_file

#Exclude database-related filesystems
echo "/db2/<SID>/sapdata1" >>$exclusion_file
:
echo "/db2/<SID>/sapdata<n>" >>$exclusion_file
echo "/db2/<SID>/log_dir" >>$exclusion_file
echo "/db2/<SID>/log_archive" >>$exclusion_file

#Exclude the backup directories
echo "/os-exe-backups" >>$exclusion_file
echo "/db2-data-backups" >>$exclusion_file

#Run the backup
cd /os-exe-backups
tar -v --gzip -cf $backup_file / --exclude-
from=$exclusion_file > $log_file 2> $error_log_file

#to monitor
#tail -f $log_file
#tail -f $error_log_file


NOTE:
```

THE COMMANDS HAVE BEEN PROVIDED AS AN EXAMPLE, PLEASE TEST AND CHANGE AS REQUIRED.

To send the backup to Amazon S3, create an EBS snapshot of the EBS volume that holds the /os_exe_backups file system.

You can refer to the section "Backup: creating an EBS snapshot onto Amazon S3 of an EBS volume" for the required steps.

It is recommended to tag the snapshot with a description like

"<EC2-INSTANCE-ID>_OS_EXE_BACKUPS_<YYYY-MM-DD>",

where <EC2-INSTANCE-ID> can be retrieved from the EC2 metadata web service through

"GET http://169.254.169.254/latest/meta-data/instance-id"

Linux tools like curl or wget can be used to issue the above HTTP command from a local shell on the Amazon EC2 instance.

More information on using Amazon EC2 instance metadata is available at

http://docs.amazonwebservices.com/AWSEC2/latest/UserGuide/AESDG--chapter--instancedata.html

Examples for restoring SAP systems using AWS infrastructure

Example 1: restore OS from Amazon S3 using TAR

Create an Amazon instance using the “golden backup” AMI of the original system

1) Log in to the AWS EC2 Management Console

https://console.aws.amazon.com/ec2

2) Within AMIs select your “golden backup” AMI and click Launch

The “golden backup” AMI should be created before, as described in the previous section

“Backup: creating a full offline Amazon EC2 Amazon Machine Image (AMI)”

a. On Instance Details

1. Choose an instance

type
that
is
supported
for
SAP

Check
SAP
note
1588667
–
SAP
on
Amazon
Web
Services
(AWS)

2. Select
the
VPC
and
Subnet

3. Click
on
Continue

4. Type
the
IP
Address
(Select
the
same
IP
address
of
the
server
you
want
to
recover)

5. Check
the
Termination
Protection

6. Click
on
Continue

7. On
Name,
type
the
name
of
the
instance

b. On
Create
Key
Pair

1. Select
the
existing
Key
Pairs
from
Choose
From
your
existing
Key
Pairs

2. Click
on
Continue

c. On
CONFIGURE
FIREWALL

1. Choose
an
existing
SG
or
Create
a
new
one

NOTE:
DEPENDING
ON
THE

SAP
INSTANCE
TYPE,
THE
PORTS
REQUIRED
WOULD
BE
TCP/22,

TCP/3200,
TCP/3300,
TCP/3600.

d. On
REVIEW

1. Click
on
Launch

Recreate
the
file
systems
required
for
the
recovery

NOTE:
IT
IS
REQUIRED
TO
HAVE
EBS
SIZES
AT
LEAST
THE
SAME
SIZE
OR
LARGER
THAN
THE
ORIGINAL
SYSTEM.
ALSO
THE

NAMES
OF
THE
MOUNT
POINTS
MUST
BE
IDENTICAL.

1) To
create
a
new
EBS
volume

a. On
Volumes,
click
on
Create
Volumes

b. Type
the
Size
of
the
Volume

c. Select
the
same
Availability
Zone
as
the
Instance

d. Click
on
Yes,
Create

2) To
attach
the
EBS
volume

an instance

a. On Volumes, select the volume

b. Click on Attach Volume

c. Select the AWS instance on Instances

d. Type the Device the EBS will be presented on the Instance (write it down)

e. Click on Yes, Attach

3) To create a file system using LVM

a. Log in to the Amazon EC2 instance

For the MaxDB database type, use this script as an example:

```
#To Create swap
pvcreate /dev/sdX
vgcreate vgswap /dev/sdX
lvcreate -L <size>M -n swap vgswap
mkswap /dev/vgswap/swap
swapon –a
#
pvcreate /dev/sdY
vgcreate vgsapdb /dev/sdY
lvcreate -L <size>M -n sapdb vgsapdb
mkfs.ext3 /dev/vgsapdb/sapdb
mkdir /sapdb
mount /dev/vgsapdb/sapdb /sapdb

#Add the mount points into the /etc/fstab
/dev/vgsapdb/sapdb /sapdb ext3 acl,user_xattr 1 1
/dev/vgswap/swap swap swap defaults 0 0
NOTE:
```

THESE COMMANDS HAVE BEEN PROVIDED AS

AN
EXAMPLE,
YOU
CAN
USE
LVM2
OR
LINUX
PARTITION.

THIS
SCENARIO
WAS
TESTED
WITH
LVM2

For
the
DB2
LUW
database
type,
use
this
script
as
an
example:

```
#To Create swap
pvcreate /dev/sdX
vgcreate vgswap /dev/sdX
lvcreate -L <size>M -n swap vgswap
mkswap /dev/vgswap/swap
swapon –a
#
pvcreate /dev/sdY
vgcreate vgdb2 /dev/sdY
lvcreate -L <size>M -n db2 vgdb2
mkfs.ext3 /dev/vgdb2/db2
mkdir /db2
mount /dev/vgdb2/db2 /db2

#Add the mount points into the /etc/fstab
/dev/vgdb2/db2 /db2 ext3 acl,user_xattr 1 1
/dev/vgswap/swap swap swap defaults 0 0
NOTE:
```

THESE
COMMANDS
HAVE
BEEN
PROVIDED
AS
AN
EXAMPLE,
YOU
CAN
USE
LVM
OR
LINUX
PARTITION.
THIS

SCENARIO
WAS
TESTED
WITH
LVM

Restore
the
OS

1) Create
an
EBS
volume
based
on
an
Amazon
S3
OS
backup
snapshot

a. On
Volumes,

click on Create Volumes

b. Type the Size of the Volume

c. Select the same Availability Zone as the Instance

d. On Snapshot select the latest OS backup

e. Click on Yes, Create

2) Attach the volume to the Instance

3) Mount the file system

a. Log in to the Amazon EC2 instance

b. Mount the /os_exe_backup file system

Example:

```
vgscan
vgimport vgbackup
vgchange vgbackup -a y
mkdir /os_exe_backups
mount /dev/vgbackup/os_exe /os_exe_backups
```

4) Restore the OS

Example:

```
#Backup the current fstab
cp /etc/fstab /etc/fstab.<timestamp>
#Restore the OS
cd /
tar -zxvf /os_exe_backups/backup.tar.gz

#Rewrite the lvm information
vgcfgbackup

#restore the original fstab
cp /etc/fstab.<timestamp> /etc/fstab
#Update the /etc/hosts file and update the
IP if necessary
vi /etc/hosts
```

Reboot the instance

```
shutdown –r –y 0
```

The system is now ready for database restore and recovery.

Example 2: Restoring and Recovering the Database from Amazon S3

In general, restore and recovery of databases requires careful planning and preparation before execution. Take time for root--cause analysis to be better able to identify the most efficient recovery strategy before executing.

For example, data only need to be restored from Amazon S3 if recovery is not possible anymore from data and backups that are already available on the system. If a database

needs to be recovered to the latest possible point in time, always make sure to not overwrite and destroy the latest database transaction logs. As these are typically not yet archived and/or backed up to Amazon S3, the latest transactions contained in these logs could then get lost forever!

This guide does not intend to replace the original backup and restore documentation of the database vendors that was referenced in section “Prerequisite Documents”. As restore and recovery scenarios are most diverse and dependent on the environment and failure cause, it is strongly recommended to follow the

original documentation in case of a real failure.

The following sections describe some basic examples of MaxDB and DB2 database restore and recovery,

to get an initial idea how that works on Amazon infrastructure.

Example 2a: MaxDB restore and recovery from Amazon S3

1) If this is a DR, make sure you have restored the OS as described in section “Example 1: restore OS

from Amazon S3 using TAR”

2) If required, mount the Amazon S3 Database backup on the Amazon EC2

instance

NOTE: NORMALLY, THE MOST RECENT BACKUP SHOULD ALREADY BE AVAILABLE ON THE

AMAZON EC2 INSTANCE

a. Only if required, create a new EBS volume based on an Amazon S3 snapshot backup

1. On Volumes, click on Create Volumes
2. Type the Size of the Volume
3. Select the same Availability Zone as the Instance
4. On Snapshot select the latest OS backup
5. Click on Yes, Create

b. Attach the volume to the Instance

c. Mount the file system

1. Log in to the EC2 instance
2. Mount the /db_data_backups file system

Example if using LVM2:

```
vgscan
vgimport vgbackup
vgchange vgbackup -a y
mkdir /db_data_backups
mount /dev/vgbackup/db_data /db_data_backups
```

3) If required, repeat the previous step, but

now
for
the
/db_log_backups
file
system

NOTE:
NORMALLY
THE
MOST
RECENT
BACKUP
SHOULD
ALREADY
BE
AVAILABLE
ON
THE
EC2
INSTANCE

4) Restore
the
database
using
Database
Studio

a. Logon
into
the
instance
as
root

b. Start
x_server

c. Logon
into
the
database
using
the
CONTROL
user

(My
Landscape
/
Servers
/
<servername/ip>/<DB>/(right
click)
/Login)

d. Set
the
database
in
admin
mode:
(Right
click)
on
the
database
and
click
on

Administration
Tasks/Set
State/Admin

e. Start
the
recovery:
(Right
click)
on
the
database
and
click
on

Administration

Tasks/Recovery…

f. On the Recovery of Database window, click on Recover a medium

g. Select FULL from the template list and then Next

h. Click on Start

i. Click Ok on the in the Confirmation Initialization database window

j. Wait until the restore has completed.

5) Apply transaction logs using Database Studio

a. First restore the database using dbmcli or Database Studio without restarting the database

b. Logon into the database using the CONTROL user (My Landscape / Servers /

<servername/ip>/<DB>/(right click) /Login)

c. Start the recovery: (Right click) on the database and click on

“Administration

Tasks/Recovery…”

d. You can select

if
you
want
to
restore
until
a
specific
time
(Point
in
time
recovery),
in
this

example
we
will
restore
until
the
last
available
log,
so
do
not
select
“Recover
until
a
specific

time.”
Select
Recover
a
medium
and
then
Click
Next

NOTE:
CREATE
AN
EBS
VOLUME
FROM
AN
AMAZONS3
LOG
BACKUP
SNAPSHOT
–
FOLLOWING
THE
LOG
SEQUENCE

–
AND
MOUNT
IT
TO
THE
/DB_LOG_BACKUPS
IF
IT
IS
REQUIRED
TO
APPLY
OLDER
LOGS
AS
THE
ONES
AVAILABLE
ON

THE
SYSTEM.
HOWEVER,
BEFORE
DOING
SO,
MAKE
SURE
TO
HAVE
A
SNAPSHOT
AVAILABLE
OF
THE
LATEST
CONTENT

OF
/DB_LOG_BACKUPS
FILE

SYSTEM,
AS
THAT
MIGHT
BE
REQUIRED
LATER
IN
THE
RECOVERY
PROCESS.

e. Select
LOG

from
the
template
list
and
then
Next

f. Select
the
Log
File
Number
and
Then
Next
(In
this
case
put
the
next
log
after
the
backup)
and

click
Next

g. Click
Start

h. The
recovery
session
will
try
to
recover
all
consecutive
logs
until
it
fails
with
a
missing
log.
You

can
choose
to
restore
more
logs
and
continue…

i. Choose
Start
Database
and
then
Continue

6) Restore
the
database
using
DBMCLI

a. Log
in
to
the
Amazon
EC2
instance
as
root

b. Execute
the

following example commands

```
#Fix permissions
chown sdb:sdba /sapdb/<SID>/sapdata
chown sdb:sdba /sapdb/<SID>/saplog

#logon as <sid>adm
su - <sid>adm

#start the db in ADMIN mode
dbmcli -U c db_admin
```



```
#Logon a recovery session
dbmcli -U c

#List the backup history, and last logs
#take notes of the next log number

backup_history_open
backup_history_list -r last -c
label,action,pages,firstlog,lastlog,media

#If possible, check if the backup is
accessed as expected
recover_check FULL data

#Restore the database

db_connect
recover_start FULL data
```

c. Wait until the restore is completed

To monitor the restore session, use the following command

Example:

```
#logon as <sid>adm
su - <sid>adm

#start restore session session
dbmcli -U c -uUTL -d <SID>
recover_state
```

Monitor the “Pages Transferred” and “Pages Left”

7) Restore the database logs using CLI

a. Before restoring logs, you should have had restored a database without restarting, (see step 1)

or step 3)). The

DB should be in ADMIN mode

b. Logon into the instance as root

c. Execute the following

```
su - <sid>adm

#Logon a recovery session
dbmcli -U c

#List the backup history, and last logs
#take notes of the next log number

backup_history_open
backup_history_list -r last -c
label,action,pages,firstlog,lastlog,media

service_connect
```



```
#Restore the Logs where <XXX> is the next log sequence.
db_connect
recover_start LOG log <XXX>

#If the recovery ends with -8020 error code and you still
have logs to recover that are not listed in the backup
history, you can continue with the following commands,
where <YYY> is the next log to recover, recover log by log
until you restore the latest available log.

recover_replace LOG /backuplog/<SID>_LOG.<YYY>

#Use the following command to review the status of the
database after the restore


db_restartinfo
#If the consistent=1 the database can start
```

d. Use the following commands to start the recovery if you want to restore in point in time

recovery

```
su - <sid>adm

#Logon a recovery session
dbmcli -U c

#List the backup history, and last logs
#take notes of the next log number

backup_history_open
backup_history_list -r last -c
label,action,pages,firstlog,lastlog,media

service_connect

#Restore the Logs where <XXX> is the next log sequence.

db_connect
recover_start LOG log <XXX> UNTIL <date> <time>
#If the recovery ends with -8020 error code and you still
have logs to recover that are not listed in the backup
history, you can continue with the following commands,
where <YYY> is the next log to recover, recover log by log
```

```
until you restore the latest available log.

recover_replace LOG /backuplog/<SID>_LOG.<YYY>

#Note: the database will be put on ONLINE automatically
after the DB is recovered until the time specified
```

8) Start the SAP instance


```
su - <sidadm>
startsap all
```

At this point the SAP instance should be started with no issues

9) After restoring the database, you can remove the EBS if required

a. Logon into the Amazon EC2 instance

b. dismount and remove the volume

Example:

```
umount /db_data_backups
vgchange vgbackup –a n
vgexport vgbackup
```

c. Logon in to the AWS EC2 Management Console

```
https://console.aws.amazon.com/ec2/
```

d. Go to Volumes and select the volume to remove

e. Click on Detach

Volume

f. Click on Yes, Detach on the popup window

g. When detached, click on Delete Volume

Example 2b: DB2 LUW restore and recovery from Amazon S3

1) If this is an DR, make sure you have restored the OS

2) Mount the Amazon S3 Database backup on the instance

a. Create an EBS volume based on an Amazon S3 DB backup snapshot

1. On Volumes, click on Create Volumes

2. Type the Size of the Volume

3. Select the same Availability Zone as the Instance

4. On Snapshot select the latest DB backup

5. Click on Yes, Create

b. Attach the volume to the Instance

c. Mount the file system

1. Logon to the instance using putty, or any other ssh client

2. Mount the backup filesystem

Example:

```
vgscan
vgimport vgbackup
vgchange vgbackup -a y
mkdir /backups
mount /dev/vgbackup/backups /db2_data_backups
```


3) Mount the Amazon S3 Database logs on the instance

NOTE: THIS IS STEP ONLY REQUIRED IF THE AVAILABLE ARCHIVED LOGS ARE NOT SUFFICIENT FOR RECOVERY, OR IF THEY

ARE NOT AVAILABLE ANYMORE

a. Create an EBS volume based on an AmazonS3 log backup snapshot

1. On Volumes, click on Create Volumes

2. Type the Size of the Volume

3. Select the same Availability Zone as the Instance

4. On Snapshot select the latest log backup

5. Click on Yes, Create

b. Attach the volume to the Instance

c. Mount the FS

NOTE: FOR SAFETY REASONS, THE FILE SYSTEM WILL BE MOUNTED TO ANOTHER MOUNT POINT AS THE

ARCHIVE LOG FILE SYSTEM ITSELF.

1. Logon to the instance using putty, or any other ssh client

2. Mount the backup filesystem

Example:

```
vgscan
vgimport vgbackuplog
vgchange vgbackuplog -a y
mkdir /backuplogs
mount /dev/vgbackuplog/backuplogs /backuplogs
```

4) Restore and recover the database using CLI

a. Logon into the

instance as root

b. Execute the following commands

```
# Log on as db2<sid>
su – db2<sid>

# If required, copy additional DB logs to the default log archive location
# For example:
cp –rp /backuplogs/* /db2/<SID>/log_archive

# Start database manager
db2start

# Start the db recovery
db2 recover db <sid> using history file (/db2_data_backups/db2rhist.asc)

# Once DB recovery is done, restore configuration stored outside of DB
cd /db2_data_backups
db2cfimp <SID>_cfg_backup.txt
```

5) Start the SAP system


```
su - <sidadm>
startsap all
```


Backup, Archive, and Restore Approaches Using AWS

http://d0.awsstatic.com/whitepapers/1117%20FINAL_AWS_Whitepaper_-_Backup_Archive_and_Restore_Approaches_Using_AWS__AWS_Legal_Comments_.pdf December 09, 2014

... Archive and Restore Approaches Using AWS ... so has the strategy for using it in backup and recovery ... One of the main advantages of using Amazon Web Services ...

Backup, Archive, and Restore Approaches Using AWS

Pawan Agnihotri
AWS Certified Solutions Architect – Professional
Amazon Web Services

November 2014

Contents

Abstract
Over the past couple of years enterprise data has grown substantially, and the growth is accelerating. The need to protect the data has grown along with it. The increase in data also brings an increase in the complexity of methods used for the backing up the data. Questions such as durability and scalability of the backup solution are now commonplace. A common question is: How does cloud help my backup and archival needs?

This document aims to answer this question and propose some solutions around using cloud to back up your data. It discusses best practices of protecting your data using cloud services from AWS. This guide for backup, archive and restore will assist enterprise solution architects, backup architects, and IT administrators who are responsible for the design and deployment of the data protection for their corporate IT environments.

Introduction
As a backup architect or engineer, you are responsible for backups and archive for your enterprise. You have to manage the infrastructure as well as the backup operations. This may include managing tapes, sending tapes offsite, managing tape drives, managing backup servers, managing backup software, creating backup policies, insuring the backup data is secure, meeting compliance requirements for data retention, and performing restores. Furthermore, cost cutting puts pressure on your budgets and, with business open for more hours, your window to perform the backup is getting smaller.

These are some of the challenges that are faced by backup teams across many enterprises. The legacy environments are hard to scale, you need more tape and tape drives, and more storage capacity to back up the avalanche of data that the business is producing.

For those of you dealing with backups and restores, you may be employing many different systems, processes, and techniques available in the market. Additionally, you may have to support multiple configurations. With AWS, organizations can obtain a flexible, secure, and cost-effective IT infrastructure in much the same way that national electric grids enable homes and organizations to plug into a centrally managed, efficient, and cost-effective energy source. When freed from creating their own electricity, organizations were able to focus on the core competencies of their business and the needs of their customers. Please review some of the terms related to backup and archiving in the appendix which will be used throughout this whitepaper.

Why Use AWS
Amazon Web Services (AWS) is a secure, high-performance, flexible, cost-effective, and easy-to-use cloud computing platform. AWS takes care of the undifferentiated heavy lifting and provides the user with the necessary tools to accomplish the task of backing up the vast amounts of data from a variety of sources.

The first question asked by many customers is about security: Will my data be secure in the cloud? Amazon Web Services takes security very seriously; every service that we launch focuses on security as the foundation. Our storage services like Amazon Simple Storage Service1 (Amazon S3) provide strong capabilities for access control and encryption both at rest and in transit. For encryption at rest, customers can use their own encryption keys2 with the Amazon S3 server side giving them control over their data.

Switching to AWS offers many advantages:

- Durability – Amazon S3 and Amazon Glacier3 are designed for 99.999999999% durability for the objects stored in them.
- Security – AWS provides a number of options for access control and encrypting data in transit and at rest.
- Global Infrastructure – Amazon Web Services are available across the globe so you can back up and store data in the region that meets your compliance requirement.
- Compliance –AWS infrastructure is designed and managed in alignment with regulations, standards and best-practices including (as of the date of this publication) SOC, SSAE 16, ISO 27001, PCI DSS, HIPPA, and FedRamp so you can easily fit the backup solution into your existing compliance regimen.
- Scalability – With AWS, you don't have to worry about capacity. You can scale your consumption up or down as your needs change.
- Lower TCO – The AWS scale of operations drives service costs down and helps lower the overall TCO of the storage. AWS often passes these cost savings on to the customer. As of the date of this publication, AWS has lowered prices 45 times since they began offering web services.

1 http://aws.amazon.com/s3/
2

http://aws.amazon.com/blogs/aws/s3-encryption-with-your-keys/

3 http://aws.amazon.com/glacier/

Backup and Archive

Developing a comprehensive strategy for backing up and restoring data is not a simple task. In some industries, regulatory requirements for data security, privacy, and records retention can be important factors to consider when developing a backup strategy. A good backup process can be defined based on the objectives:

1. Backing up file data
2. Backing up database
3. Backing up machine images

In the following sections we describe the backup and archives approaches based on the organization of your infrastructure. IT infrastructure can broadly be categorized into the following scenarios - Cloud native, on premises, and hybrid.

Cloud Native

This scenario describes a workload environment that exists entirely on AWS. This includes web servers, application servers, databases, Active Directory, monitoring servers, etc. See Figure 1: AWS Native Scenario.

Figure 1: AWS Native Scenario

With all the services in AWS, you can leverage many of the built-in features to accomplish the backup-archive tasks.

Snapshot Options for Amazon EBS

In AWS “file data” can be stored on either Amazon S3 or Amazon Elastic Block Store4 (Amazon EBS) volumes. Let’s take a look at how you can backup data on these.

Amazon Elastic Compute Cloud5 (Amazon EC2) can use Amazon EBS volumes to store block-based data. You can use this block storage for databases, or formatted into any OS-supported file system. Amazon EBS provides the ability to create snapshots (backups) of any Amazon EBS volume and write a copy of the data in the volume to Amazon S3, where it is stored redundantly in multiple Availability Zones. You can create the Amazon EBS snapshot by using the AWS Management Console, the command line interface (CLI), or the APIs. Using the Elastic Block Store Volumes page6 of the Amazon EC2 console, click Actions and then click Create Snapshot to commence the creation of a snapshot that is stored in Amazon S3.

Figure 2: Creating a Snapshot from Amazon EBS Using the Console

You can also create the snapshot using the CLI command ec2-create-snapshot.

When you create a snapshot, you protect your data directly to durable disk-based storage. You can schedule and issue the commands on a regular basis. And due to the economical pricing of Amazon S3, you can retain many generations of data. Further,

4 http://aws.amazon.com/ebs/

5 http://aws.amazon.com/ec2/

6 https://console.aws.amazon.com/ec2/v2/#Volumes

because snapshots are block-based, you consume space only for changed data after the initial snapshot is created.

To restore data from a snapshot, use AWS Management Console, the command line interface (CLI), or the APIs to create a new volume from an existing snapshot.

For example, to restore a volume to a prior point-in-time backup, you could use the following sequence:

1. Create a new volume from the backup snapshot by using the following command:

```
> ec2-create-volume –z us-west-1b –snapshot MySnapshotName
```

2. Within the Amazon EC2 instance, unmount the existing volume (e.g., by using umount in Linux or the Logical Volume Manager in Windows).
3. Detach the existing volume from the instance by using the following command:

```
> ec2-detach-volume OldVolume
```

4. Attach the new volume that was created from the snapshot by using the following command:

```
> ec2-attach-volume VolumeID –I InstanceID –d Device
```

5. Remount the volume on the running instance.

This process is a fast and reliable way to restore full volume data as needed. If you need only a partial restore, you can attach the volume to the running instance under a different device name, mount it, and then use operating system copy commands to copy the data from the backup volume to the production volume.

Amazon EBS snapshots can also be copied between AWS regions using the Amazon EBS snapshot copy capability that is available from the console or command line, as explained in the Amazon Elastic Compute Cloud User Guide.7 You can use this feature to store your backup in another region without having to manage the underlying replication technology.

Creating Consistent or Hot Backups

When you back up a system, the ideal is to have the system in a quiet state where it is not performing any I/O. From a backup perspective, the ideal state is a machine that is

7 http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-copy-snapshot.html

accepting no traffic. But this ideal is increasingly rare as 24/7 IT operations become the norm.

Consequently, it is necessary to quiesce the file system or database in order to make a clean backup. How you do this depends on your database and/or file system, so due diligence is required. To summarize the process for a database:

- If possible, put the database into hot backup mode. Alternatively, create a read replica copy of the database; this is a copy of the database that is up to date but runs on a separate instance. Keep in mind that on AWS you can run this instance for the duration required to perform the backup and then close it down, saving resources.
- Issue the relevant Amazon EBS snapshot commands.
- Take the database out of hot backup mode or, if using a read replica, terminate the read replica instance.

Backing up a file system works similarly and depends highly on the capabilities of the particular operating system or file system. An example of a file system that can flush its data for a consistent backup is xfs (see xfs_freeze).8 If the file system in question does not support the ability to freeze, you should unmount it, issue the snapshot command, and then remount the file system. Alternatively, you can facilitate this process by using a logical volume manager that supports freezing of I/O.

Because the snapshot process is fast to execute and captures a point in time, the volumes you are backing up only need be unmounted for a matter of seconds. This ensures that the backup window is as small as possible and that outage time is predictable and can be effectively scheduled. While the data copy process of creating the snapshot may take longer, the snapshot activity requiring the volume to be un-mounted is very quick. Don't confuse the two processes when structuring your backup regime.

Multivolume Backups

In some cases, you may stripe data across multiple Amazon EBS volumes by using a logical volume manager in order to increase potential throughput. When using a logical volume manager (e.g., mdadm or LVM), it is important to perform the backup from the volume manager layer rather than the underlying devices. This ensures all metadata is consistent and that the various subcomponent volumes are coherent. You can take a number of approaches to accomplish this, an example being the script created by alestic.com.9

8 https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Storage_Administration_Guide/xfsfreeze.html

9 https://github.com/alestic/ec2-consistent-snapshot

You can also perform backups of this nature from the logical volume manager or file system level. In these cases, using a "traditional" backup agent enables the data to be backed up over the network. A number of agent-based backup solutions are available on the internet and the AWS Marketplace.10 It is important to remember that agent-based backup software expects a consistent server name/IP address. As a result, using these tools in concert with instances deployed in a virtual private cloud (VPC)11 is the best method to ensure reliability.

An alternative approach is to create a replica of the primary system volumes that exist on a single large volume. This simplifies the backup process, as only one large volume needs to be backed up, and the backup does not take place on the primary system. However, it is important to ascertain whether the single volume can perform sufficiently

to maintain changes during the backup and whether the maximum volume size is appropriate for the application.

Backing Up Databases

AWS has many options for running databases. You can run your own database on an Amazon EC2 instance or use one of the managed services offering. If you are running your own database on an Amazon EC2 instance, you can back up data to files using database native tools (e.g., MySQL,12 Oracle,13, 14 MSSQL,15 postgresSQL16) or create a snapshot of the volumes containing the data.

Backing up data for database differs from the web and application layers. In general, databases contain larger amounts of business data (tens of GB to multiple TB) in database-specific formats that must be retained and protected at all times. In these cases, you can leverage efficient data movement techniques such as snapshots to create backups that are fast, reliable, and space efficient.

For databases that are built upon RAID sets of Amazon EBS volumes (and have total storage less than 1 TB), an alternative backup approach is to asynchronously replicate data to another database instance built using a single Amazon EBS volume. While the destination Amazon EBS volume will have slower performance, it is not being used for data access, and you can easily send a snapshot to Amazon S3 using the Amazon EBS

10 https://aws.amazon.com/marketplace/

11 http://aws.amazon.com/vpc/

12 http://dev.mysql.com/doc/refman/5.7/en/backup-and-recovery.html

13 https://media.amazonwebservices.com/AWS_Amazon_Oracle_Backups.pdf

14 http://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmbckba.htm#BRADV8003

15 http://msdn.microsoft.com/en-us/library/ms187510.aspx

16 http://www.postgresql.org/docs/9.3/static/backup.html

snapshot capability (see “Snapshot Options for Amazon EBS” earlier in this paper section).

Backups for Amazon Relational Database Service

The Amazon Relational Database Service (Amazon RDS)17 includes automated backups. This means that you do not need to issue specific commands to create backups of your database.

Amazon RDS provides two different methods for backing up and restoring your DB Instance(s); automated backups and database snapshots (DB snapshots).

- Automated backups enable point-in-time recovery of your DB Instance. When automated backups are turned on for your DB instance, Amazon RDS automatically performs a full daily backup of your data (during your preferred backup window) and captures transaction logs (as updates to your DB instance are made). When you initiate a point-in-time recovery, transaction logs are applied to the most appropriate daily backup in order to restore your DB instance to the specific time you requested. Amazon RDS retains backups of a DB instance for a limited, user-specified period of time called the retention period, which, as of the date of this publication, by default is one day but can be set to up to thirty-five days.

  You can initiate a point-in-time restore and specify any second during your retention period, up to the Latest Restorable Time. You can use the DescribeDBInstances API call to return the latest restorable time for your DB instance(s), which is typically within the last five minutes. Alternatively, you can find the Latest Restorable Time for a DB instance by selecting it in the AWS Management Console and looking in the Description tab in the lower panel of the console.

- DB snapshots are user-initiated and enable you to back up your DB instance in a known state as frequently as you wish, and then restore to that specific state at any time. DB snapshots can be created with the AWS Management Console or by using the CreateDBSnapshot API call. The snapshots are kept until you explicitly delete them with the console or the DeleteDBSnapshot API call.

Note that when you restore to a point in time or from a DB snapshot, a new DB instance is created with a new endpoint. (If you want, you can delete the old DB instance by using the AWS Management Console or a DeleteDBInstance call.) You do this so you can create multiple DB instances from a specific DB snapshot or point in time.

17 http://aws.amazon.com/rds/

Backup of the Amazon Machine Image (AMI)

Next we look at “machine images.” AWS stores system images in what are called Amazon Machine Images or AMI for short. These images consist of the template for the root volume required to launch an instance. To save your instance’s as a machine image

you simply backup the root volume as an AMI.

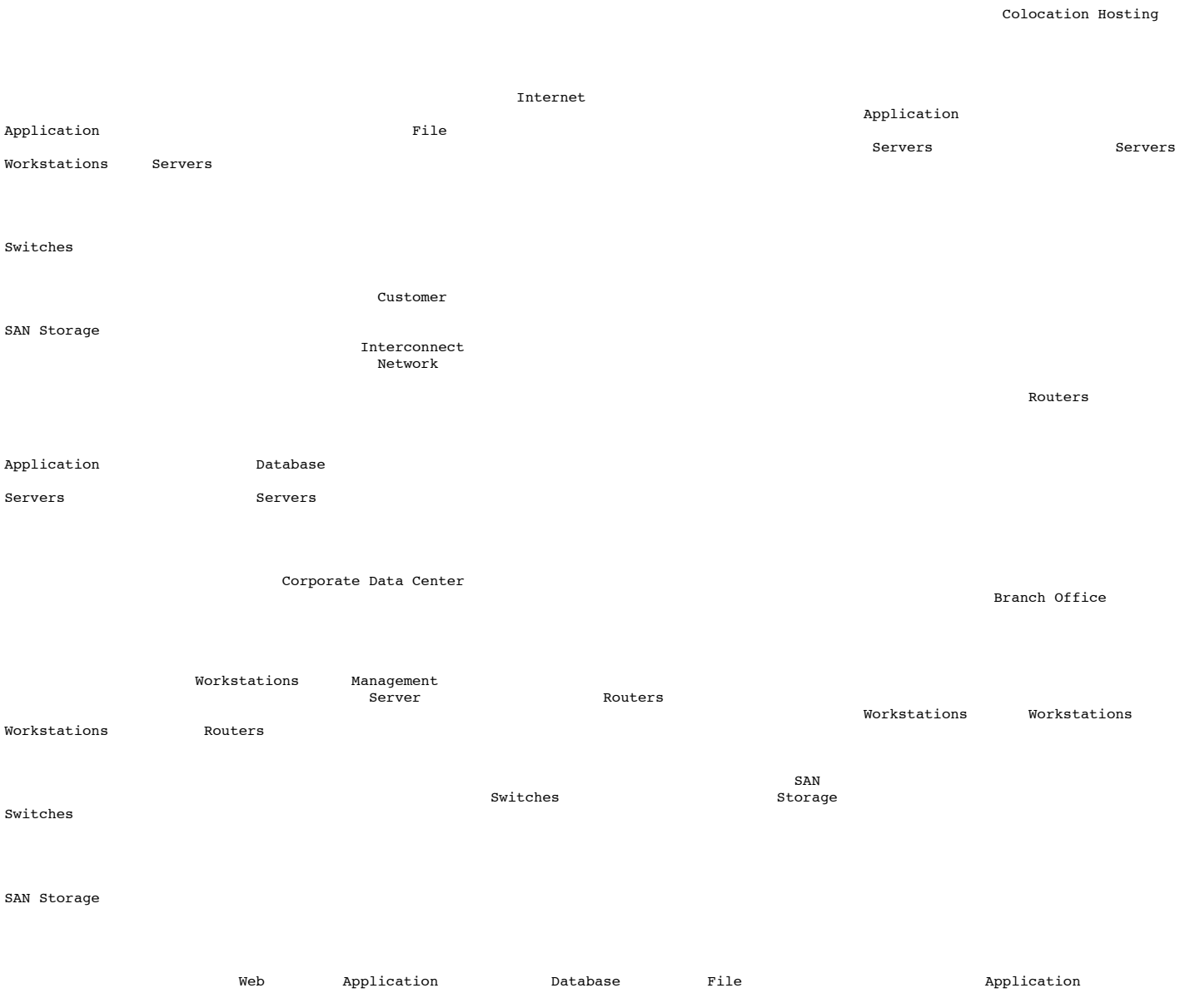
Figure 3: Using AMI to backup and launch an instance

An AMI that you register is automatically stored in your account using Amazon EBS snapshots. These snapshots reside in Amazon S3 and are highly durable. This means that the underlying storage mechanism for the AMIs is protected from multiple failure scenarios.

Figure 4: Using the EC2 console to create a machine image

Once you have created an AMI of your Amazon EC2 instance you can use the AMI to recreate the instance or launch more copies of the instance. It is also possible to copy AMIs from one region to another. Consequently, you can save a copy of a system image to another region.

On Premises
This scenario describes a workload environment with no component in the cloud. All resources, including web servers, application servers, databases, Active Directory, monitoring, and more, are hosted either in the customer data center or colocation. See the following figure.



Database Servers File Servers Servers Servers Servers Servers Servers Servers Servers Servers

Figure 5: On-premises environment

AWS can be leveraged very nicely for this scenario to help with backup and archiving. Using AWS storage services lets you focus on the backup and archiving task, leaving the heavy lifting on the storage side to AWS. With AWS you do not have to worry about storage scaling or infrastructure capacity to accomplish the backup task.

Amazon storage services such as Amazon S3 and Amazon Glacier are natively API based and available via the Internet. This allows backup software vendors to directly integrate their applications with storage solutions provided by AWS as represented in the following figure. You can look at our partner directory18 for the backup software vendors who work with AWS.

The primary solution in this scenario is to use backup and archive software that directly interfaces with AWS through the APIs. Here the backup software is AWS aware and will

18 http://www.aws-partner-directory.com/PartnerDirectory/PartnerSearch?type=ISV

back up the data from the on premises servers directly to Amazon S3 or Amazon Glacier.

Figure 6: Backup connector to Amazon S3 or Amazon Glacier

If your existing backup software does not natively support the AWS cloud, the alternate solution is to use our storage gateway products. AWS Storage Gateway19 is a virtual appliance that provides seamless and secure integration between your data center and AWS's storage infrastructure. The service allows you to securely store data in the AWS cloud for scalable and cost-effective storage. The AWS Storage Gateway supports industry-standard storage protocols that work with your existing applications while securely storing all of your data encrypted in Amazon S3 or Amazon Glacier.

Figure 7: Connecting on-premises to AWS storage

19 http://aws.amazon.com/storagegateway/

AWS Storage Gateway supports three configurations:

- Gateway-cached volumes – You can store your primary data in Amazon S3 and retain your frequently accessed data locally. Gateway-cached volumes provide substantial cost savings on primary storage, minimize the need to scale your storage on premises, and retain low-latency access to your frequently accessed data.
- Gateway-stored volumes – In the event you need low-latency access to your entire data set, you can configure your on-premises data gateway to store your primary data locally, and asynchronously back up point-in-time snapshots of this data to Amazon S3. Gateway-stored volumes provide durable and inexpensive off-site backups that you can recover locally or from Amazon EC2.
- Gateway-virtual tape library (gateway-VTL) – With gateway-VTL you can have a limitless collection of virtual tapes. Each virtual tape can be stored in a virtual tape library backed by Amazon S3 or a virtual tape shelf backed by Amazon Glacier. The virtual tape library exposes an industry standard iSCSI interface, which provides your backup application with online access to the virtual tapes. When you no longer require immediate or frequent access to data contained on a virtual tape, you can use your backup application to move it from its virtual tape library to your virtual tape shelf to further reduce your storage costs.

These gateways act as plug-and-play devices providing standard iSCSI devices, which can be integrated into your backup or archive framework. You can use the iSCSI disk devices as storage pools for your backup software or the gateway-VTL to offload tape-based backup or archive directly to Amazon S3 or Amazon Glacier.

Using this method your backup and archives are automatically offsite (for compliance purposes) and stored on durable media, eliminating the complexity and security risks of off-site tape management.

To offer flexibility to the customer, a multitude of third-party appliances and gateway devices work with Amazon storage services and can be found on our partner network website.20

Hybrid
The two infrastructure deployments addressed up to this point, “cloud native” and “on-premises,” can be combined into a hybrid scenario whose workload environment has infrastructure components in AWS as well as on premises. Resources, including web servers, application servers, databases, Active Directory, monitoring, and more, are

20 http://www.aws-partner-directory.com/

hosted either in the customer data center or AWS. Applications running in the AWS are connected to applications running in the customer premises.

This scenario is emerging as a very common case for enterprise workloads. Many enterprises have data centers of their own while leveraging AWS to augment capacity. These customer data centers are often connected to the AWS network by high capacity network links. For example, with AWS Direct Connect21 you can establish private dedicated connectivity from your premises to AWS.

Figure 8: A hybrid infrastructure scenario

You can leverage AWS to help with backup and archiving for this scenario as well. The techniques you use are a combination of the methods described previously in cloud-native and on-premises solutions.

Hybrid Techniques
If you already have an existing framework that backs up data for your on-premises servers, then it is easy to extend that framework to your AWS resources over a VPN

21 http://aws.amazon.com/directconnect/

connection or AWS Direct Connect. You will install the backup agent on the Amazon EC2 instances and back them up per the existing data protection policies.

Depending on your backup framework setup, you may have a master backup server along with one or more media servers. You may consider moving the master backup server to an Amazon EC2 instance to automatically protect your master backup server against on-premises disasters and have a highly available backup infrastructure.

To manage the backup data flows, you may also consider creating one or more media servers on Amazon EC2 instances. This will help the cloud-based resources backup to a local media target rather than go over the network back to the on-premises environment.

You can also leverage the AWS Storage Gateway or other third-party storage gateways from the AWS Marketplace to connect your backup framework to Amazon storage services. The storage gateways are connected to the media servers allowing data to be securely and durably stored on Amazon S3 or Amazon Glacier.

Figure 9: Leveraging gateways in the hybrid scenario

Use Cases
A use case can help explain the on-premises and hybrid scenarios: Assume that you are managing an environment where you are backing up a mixture of standalone servers, virtual machines, and database servers. This environment has 1,000 servers, and you backup operating system, file data, virtual machine images, and database backups. You have 20 databases to back up, which are a mixture of MySQL, MSSQL, and Oracle. You use “myqldump” to create a database dump file to disk for MySQL backups. Your backup software provides plugins or agents to back up operating system, virtual machine images, data, and MSSQL databases. Additionally this software has tight integration to backup Oracle database using RMAN.

To support the above environment, your backup software has a global catalogue server or master server that controls the backup, archive and restore activities as well as multiple media serves that are connected to disk-based storage and LTO tape drives.

Case 1: As the very first step, you check the vendor site to see if there is a plugin or

built-in support for cloud storage backup and archive. If the software has cloud storage backup options, you can proceed to configure it. Many vendors support Amazon S3 as an option for cloud storage and Amazon Glacier for cloud archive. You can create the target bucket either from within the backup software or use the AWS Management Console to create a bucket in Amazon S3. Next you configure the media servers to create storage pools that use the Amazon S3 bucket. Once the storage pool is configured, the backup software starts using Amazon S3 to store the backup data.

Case 2: If your backup software does not natively support cloud storage for backup or archive, you can use a storage gateway device as a bridge between the backup software and Amazon S3 or Amazon Glacier. If you want to attach disk-based storage to your media server, you can download the gateway — cached volumes storage gateway. If you want to attach tape drives to your media server you can download the gateway—virtual tape library storage gateway. You can download the storage gateway from the AWS Management Console. Once the gateway is downloaded and activated, you can create iSCSI targets, which can be attached to the media servers. The media server sees these iSCSI targets as local disks or tape drives. You can then configure these into the storage pools and used for backups or archives.

Figure 10: Choose and download the appropriate storage gateway

Cloud Paradigms
As cloud-based computing has evolved, so has the strategy for using it in backup and recovery.

Protecting Configurations Rather Than Servers
The Amazon EC2 service simplifies the backup and recovery of a standard server, such as a web server or application server. Traditionally, you would back up the complete server via a central backup server. With Amazon EC2 you can focus on protecting configuration and stateful data, rather than the server itself. This set of data is much smaller than the aggregate set of server data, which typically includes various application files, operating system files, temporary files, and so on. This change of approach means that regular nightly incremental or weekly full backups can take far less time and consume less storage space.

Figure 11: Traditional backup approach

When a compute instance is started in Amazon EC2, it is based upon an AMI and can also connect to existing storage volumes such as Amazon EBS. In addition, when launching a new instance, it is possible to pass “user data”22 to the instance that can be accessed internally as dynamic configuration parameters.

A sample workflow is as follows:

- Launch a new instance of a web server, passing it the “identity” of the web server and any security credentials required for initial setup. The instance is based upon a prebuilt AMI that contains the operating system and relevant web-server application (e.g., Apache or IIS).
- Upon startup, a boot script accesses a designated and secured Amazon S3 bucket that contains the specified configuration file(s).

22 http://docs.amazonwebservices.com/AWSEC2/latest/UserGuide/index.html?AESDG-chapter-instancedata.html

Figure 12: Amazon EC2 Backup Approach

- The configuration file contains various instructions for setting up the server (e.g., web server parameters, locations of related servers, additional software to install, and patch updates).
- The server executes the specified configuration and is ready for service. An open source tool for performing this process, called cloud-init,23 is already installed on Amazon Linux AMIs and is also available for a number of other Linux distributions.

In this case, there is no need to back up the server itself. The relevant configuration is contained in the combination of the AMI and the configuration file(s). So the only components requiring backup and recovery are the AMI and configuration file(s).

Using Storage Fit for Purpose
In traditional storage model you have a choice of SAN or NAS based storage. Amazon Web Services provides many storage options. If your data and workflow need a file- or object-based store then Amazon S3 is the best solution. Amazon S3 is highly durable data store designed for 99.999999999% data durability. The data is internally replicated across multiple data centers. In addition Amazon S3 provides many features such as versioning. With versioning enabled, you can automatically save versions of objects that are overwritten.24 This provides an automatic backup capability for the data stored in

Amazon S3. Amazon S3 also offers data lifecycle management features so you can automatically archive or delete data when certain time criteria are met.

23 https://launchpad.net/cloud-init

24 http://docs.aws.amazon.com/AmazonS3/latest/dev/Versioning.html

Figure 13: Using the console to enable versioning for an S3 bucket

If your data and workflow need long-term retention with low chances of retrieval then Amazon Glacier is the best solution. Amazon Glacier is a storage service optimized for infrequently used data, or "cold data." The service provides durable and extremely low-cost storage with security features for data archiving and backup. With Amazon Glacier, you can store your data cost-effectively for months, years, or even decades. With Amazon S3's lifecycle management, you can automatically move data from Amazon S3 to Amazon Glacier based on the lifecycle policy.

Figure 14: Amazon Glacier storage

If your data and workflow require a file system to store files or database data then Amazon EBS is the best storage option. Amazon EBS provides many features such as high durability and reliability, encryption, provisioned IOPS, and point-in-time snapshots amongst others. The built-in volume snapshot feature is a good option for backing up data.

Automating Infrastructure

One of the main advantages of using Amazon Web Services is that capacity is available to you on demand. You have no need to preprovision your infrastructure for future or backup use. Using tools such as AWS CloudFormation25 and AWS OpsWorks,26 you can automate the build out of your infrastructure, as explained in the Bootstrapping Applications whitepaper.27 With this approach, you can operate your infrastructure as code. You are then not tied to a specific system image that you have to backup. The application can be backed up in code repositories and used to create a full-blown infrastructure at the time it is needed. Anytime you need to create a server, you launch an automated deployment of the application, which creates the infrastructure within minutes to host your application.

Conclusion

The growth in data and an explosion in the creation and use of machine-generated data are increasing the need for robust, scalable and secure backup solutions. At the same time, organizations are struggling to deal with an explosion in retained data for compliance or business reuse. Providing IT teams with services and solutions that are optimized for usability in backup and archival environments is a critical requirement.

Amazon Web Services provides cost-effective and scalable solutions to help organizations balance their requirements for backup and archiving. These services integrate well with new as well as existing technologies the customers are working with today. Gartner has recognized AWS as a leader in providing public cloud storage services28. AWS is well positioned to help organizations move their workloads to the cloud-based platforms that are the next generation of backup.

Notices

25 http://aws.amazon.com/cloudformation/

26 http://aws.amazon.com/opsworks/

27 https://s3.amazonaws.com/cloudformation-examples/BoostrappingApplicationsWithAWSCloudFormation.pdf

28 http://www.gartner.com/technology/reprints.do?id=1-1WWKTQ3&ct=140709&st=sb

customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Appendices
Terms
Some important terms often used in backup and restore discussions

- Archive – A strategy for long-term retention of data for use in case of compliance, regulatory, or historical records requirement.
- Backup – A strategy of copying files or databases for short-term retention for use in case of failure or corruption.
- Backup Frequency – The time between consecutive backups.
- Data Lifecycle Management – The process of managing data information throughout its lifecycle, from requirements through retirement.
- Data Versioning – Maintaining multiple versions of data for backup purposes.
- File / Data Backup – The process of copying individual data files to a backup medium so that they will be preserved.
- Image Backup – An exact copy of a drive or storage device containing the complete contents and structure representing the operating system and all the data associated with it, including the system state and application configurations.
- Off-Site Backups – The process of storing the copy of data in a geographically different location from the source.
- Restore – A process that involves copying backup files from secondary storage (tape, zip disk, or other backup media) to hard disk. A restore is performed in order to return data to its original condition if files have become damaged or to copy or move data to a new location.
- Retention – The amount of time that a given set of data remains available for restore. Some backup products rely on daily copies of data and measure retention in terms of days. Others retain a number of copies of data changes regardless of the amount of time.
- RPO – The maximum tolerable period in which data might be lost from an IT service due to a major incident.
- RTO – The targeted duration of time and a service level within which a business process must be restored after a disaster (or disruption) in order to avoid unacceptable consequences associated with a break in business continuity.

Partner Solutions

- Avere – Hybrid cloud NAS and AWS:
  http://www.averesystems.com/amazon-web-services
- Commvault – Cloud Integration with Amazon Web Services:
  http://www.commvault.com/resource-library/1843/commvault-amazon-web-services-solution-brief.pdf
- CTERA – CTERA Cloud Storage Services Platform and Amazon Web Services:
  http://www.ctera.com/amazon-aws-cloud-storage-platform
- NetApp Riverbed SteelStore™ – cloud storage gateway:
  http://www.riverbed.com/partners/find-a-partner/find-a-partner-tool/aws-partner.html#Cloud_Storage
- Symantec Solutions for Amazon Web Services – Symantec Netbackup Platform:
  http://www.symantec.com/content/en/us/enterprise/fact_sheets/b-nbu-DS-solutions-for-amazon-web-services-21281095.en-us.pdf
- Zmanda – Backup to Amazon S3:
  http://www.zmanda.com/backup-Amazon-S3.html

Symantec Solutions for Amazon Web Services

http://www.symantec.com/page.jsp?id=amazon December 09, 2014

Symantec and Amazon Web Services have teamed up to ... which allows access to either S3 or Amazon Glacier. Through this approach, ... scalable, backup and recovery ...

NetBackup allows customers the freedom of choice for how they want to leverage the cloud for data protection. When it comes to storing data in the cloud, NetBackup can connect to Amazon S3 via cloud connector. In addition, the Amazon Storage Gateway can be presented to NetBackup as a disk target, which allows access to either S3 or Amazon Glacier. Through this approach, customers can dramatically lower storage costs without sacrificing unified management of their information or by creating additional silos.

In some cases, enterprise customers are using Amazon EC2 for hosted workloads, which also need to be backed up. In this scenario, customers can deploy a NetBackup media server in EC2 that can protect all their cloud-based workloads. Data can be stored within EC2 for local access and quick restores, but then moved back to on-premises and to less expensive cloud storage tiers like S3 or Glacier.

For added disaster recovery readiness, the management of the NetBackup environment (master server) can also be located in the cloud, reducing DR costs and eliminating the need to maintain a secondary DR site. Whatever your unique situation, NetBackup provides your enterprise the heterogeneous, scalable, backup and recovery options you need.

Backup Exec 2012 protects virtual and physical environments, simplifies backup and disaster recovery, and offers unmatched recovery capabilities for data and applications. When paired with, Riverbed Whitewater as a WAN accelerated a cloud gateway device, and the AWS cloud, data protection experts can be assured of industry leading on-site data protection, coupled with the safety, security, and cost savings associated with enabling an offsite copy to the AWS cloud.

Symantec Disaster Recovery Orchestrator enables customers to automate and manage the takeover and failback of Microsoft Windows-based applications residing on premises in either physical or virtual machines (VMs) to public cloud such as Amazon Web Services (AWS). Disaster Recovery Orchestrator replicates application data and fully automates end-to-end application recovery. Using Disaster Recovery Orchestrator to target the cloud for disaster recovery can significantly reduce costs while achieving stringent Recovery Time Objectives (RTOs) and Recovery Point Objectives (RPOs).

Customers can now confidently deploy workloads containing confidential data to the Amazon cloud with Symantec's market-leading data loss prevention (DLP) solution. Symantec Data Loss Prevention provides comprehensive coverage and unified management of your confidential data across the Amazon Web Services (AWS) cloud and your on-premises environment.

Symantec Data Loss Prevention is a content-aware data security solution that discovers, monitors and protects confidential data stored across the AWS cloud, including AWS-hosted instances of Microsoft Exchange and Microsoft SharePoint. Unlike other security solutions that provide limited DLP controls, Symantec delivers deep content inspection, sophisticated policy and incident management, and proven scalability and performance. With AWS and Data Loss Prevention, businesses can confidently deploy workloads to the cloud without sacrificing control over of their confidential data.

DISASTER RECOVERY FOR LOCAL APPLICATIONS - Amazon Web Services

http://media.amazonwebservices.com/architecturecenter/AWS_ac_ra_disasterrecovery_07.pdf December 09, 2014

DISASTER RECOVERY FOR LOCAL ... approach involves duplicating ... AWS Storage Gateway securely uploads data to the AWS cloud for cost effective backup and rapid ...

DISASTER RECOVERY

Disaster recovery is about preparing for and recovering from any

W S

event that has a negative impact on your IT systems. A typical

approach involves duplicating infrastructure to ensure the

A ce

availability of spare capacity in the event of a disaster.

enes

FOR LOCAL APPLICATIONS

Amazon Web Services allows you to scale up your infrastructure

f e r

on an as-needed basis. For a disaster recovery solution, this

Reectur

results in significant cost savings. The following diagram shows an

example of a disaster recovery setup for a local application.

h it 2

Arc

C

E

n

zo

et

Sn

a

Fi

C

AM

m

ap les 3

n

r

P

A

sh

e y

V

Int tewa

ot Is

n

zo

s

Ga
a

S
m

B
A

E
n
zo

ion
a

3

rat
m

S

o
A

t
n

es
zo

R

ta
a

Da
m

ay
A

ew
at
G

EB

A
ge

m

S

a
ra

zo
to

n
S
S

W

Co

nn Se

aser
A

ec cu
tio re
n

GaSto

b

ta e 5
te rag
wa e
2
Da Servry)

e
A

cov
W

y
S3
S

(Re

A

m

a

zo

EC

n


A
on
m
2
a
ti
zo
a
n
ic
AW
pl
S
C
VP y
Aprver y)
ase
On
Cl
Seecover
ta b
-p
ou
e wa
Da erveorn)
re
cu s
re
d
Ga
t
(R
S ducti
e
e S up
aclBack
m
6
Pro
O r
ise
t ion
(
s
VP
A
a
m
St
5
c
C
a
pli ver)
zo
or
n
ag
eV
p
A Ser tion
S
4

ol
Co ec
um duco
nn ure
es
(Pr
ec

tio
Da Co
n
ta rp
Ce ora
nt te 1
er
C

Us orp

er or

at

e

System 1 A corporate data center hosts an application consisting of a database server and an application server with local storage for a content management system.

recovery
servers are stored on Amazon Simple Storage
Service
(Amazon S3), a highly durable and cost-effective
data
store. AMIs are pre-configured operating system and

on Amazon Virtual Private Cloud (Amazon VPC). Amazon

VPC lets you provision a private, isolated section of the AWS

cloud where you can recreate your application.

Overview 2 AWS Storage Gateway is a service connecting an on-

application software that are used to create a virtual machine
Amazon
Elastic Compute Cloud (Amazon EC2). Oracle 5 The application and database servers are recreated premises software appliance with cloud-based storage. databases
can directly back up to Amazon S3 using the using Amazon EC2. To restore volume snapshots, you AWS Storage Gateway securely uploads data to the AWS Oracle
Secure Backup (OSB) Cloud Module. can use Amazon Elastic Block Store (EBS) volumes, which cloud for cost effective backup and rapid disaster recovery.
are then attached to the recovered application server.

3 Database server backups, application server volume 4 In
case of disaster in the corporate data center, you can To remotely access the recovered application, you use snapshots, and Amazon Machine Images (AMI) of the
recreate the complete infrastructure from the backups 6 a VPN connection created by using the VPC Gateway.

Restoring a DB Instance to a Specified Time

http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER_PIT.html December 09, 2014

The Amazon RDS automated backup feature automatically creates a backup ... Click on the Use Custom Restore Time radio ... Amazon Web Services, Inc. or its ...

The Amazon RDS automated backup feature automatically creates a backup of your database. This backup occurs during a daily user-configurable 30 minute period known as the backup window. Automated backups are kept for a configurable number of days (called the backup retention period). You can restore your DB instance to any specific time during this retention period, creating a new DB instance.

When you restore a DB instance to a point in time, the default DB security group is applied to the new DB instance. If you need custom DB security groups applied to your DB instance, you must apply them explicitly using the AWS Management Console, ModifyDBInstance API, or the rds-modify-db-instance command line tool once the DB instance is available.

You can restore to any point in time during your backup retention period. To determine the latest restorable time for a DB instance, use the command with the and parameters and look at the value returned in the Latest Restorable Time column. The latest restorable time for a DB instance is typically within 5 minutes of the current time.

The OFFLINE, EMERGENCY, and SINGLE_USER modes are not currently supported. Setting any database into one of these modes will cause the latest restorable time to stop moving ahead for the whole instance.

Several of the database engines used by Amazon RDS have special considerations when restoring from a point in time. When you restore an Oracle DB instance to a point in time, you can specify a different Oracle DB engine, license model, and DBName (SID) to be used by the new DB instance. When you restore a SQL Server DB instance to a point in time, each database within that instance is restored to a point in time within 1 second of each other database within the instance. Transactions that span multiple databases within the instance may be restored inconsistently.

Some actions, such as changing the recovery model of a SQL Server database, can break the sequence of logs that are use for point-in-time recovery. In some cases, Amazon RDS can detect this issue and the latest restorable time is prevented from moving forward; in other cases, such as when a SQL Server database uses the BULK_LOGGED recovery model, the break in log sequence is not detected. It may not be possible to restore a SQL Server DB instance to a point in time if there is a break in the log sequence. For these reasons, Amazon RDS does not support changing the recovery model of SQL Server databases.

Backup and recovery glossary

http://whatis.techtarget.com/reference/Backup-and-recovery-glossary December 09, 2014

Glossary definitions for backup and recovery. ... is an approach to computer storage backup and archiving in which data is ... authentication to protect AWS ...

To read the complete WhatIs.com definition, click on the link.

archive

An archive is a collection of computer files that have been packaged together for backup, to transport to some other location, for saving away from the computer so that more hard disk storage can be made available, or for some other purpose.

backup

Backup is the activity of copying files or databases so that they will be preserved in case of equipment failure or other catastrophe.

backup storage

In computers, backup storage is storage that is intended as a copy of the storage that is actively in use so that, if the storage medium such as a hard disk fails and data is lost on that medium, it can be recovered from the copy.

cold backup

A cold backup, also called an offline backup, is a database backup when the database is offline and not accessible for updating.

CDP

Continuous data protection, also called continuous backup, is a storage system in which all the data in an enterprise is backed up whenever any change is made.

DAT USB drive

A DAT USB drive is a tape drive with digital audio tape (DAT) that can be plugged into a Universal Serial Bus (USB) connection as a simple and relatively low-cost way to back up data routinely, especially on servers.

disk-to-disk-to-tape

Disk-to-disk-to-tape (D2D2T) is an approach to computer storage backup and archiving in which data is initially copied to backup storage on a disk storage system and then periodically copied again to a tape storage system (or possibly to an optical storage system).

failover

Failover is a backup operational mode in which the functions of a system component (such as a processor, server, network, or database, for example) are assumed by secondary system components when the primary component becomes unavailable through either failure or scheduled down time.

hot backup

A hot backup, also called a dynamic backup, is a backup performed on data even though it is actively accessible to users and may currently be in a state of being updated.

iSCSI

Internet SCSI (Small Computer System Interface) is an Internet Protocol (IP)-based storage networking standard for linking data storage facilities developed by the Internet Engineering Task Force (IETF).

LTO

Linear Tape-Open (LTO) is an open-format tape storage technology developed by Hewlett-Packard (HP), International Business Machines (IBM), and Certance.

optical storage

Optical storage is any storage method in which data is written and read with a laser for archival or backup purposes.

RAID

Redundant array of independent disks; originally redundant array of inexpensive disks) is a way of storing the same data in different places (thus, redundantly) on multiple hard disks. By placing data on multiple disks, I/O operations can overlap in a balanced way, improving performance.

restore

In data management, restore is a process that involves copying backup files from secondary storage (tape, Zip disk or other backup media) to hard disk.

restore point

In recent Windows operating systems, a restore point is a saved "snapshot" of your computer's data at a specific time. By creating a restore point, you can save the state of the operating system and your own data so that if future changes cause a problem, you can restore the system and your data to the way it was before the changes were made.

Serial ATA

Serial Advanced Technology Attachment or SATA) is a new standard for connecting hard drives into computer systems. As its name implies, SATA is based on serial signaling technology, unlike current IDE (Integrated Drive Electronics) hard drives that use parallel signaling.

serverless backup

Serverless backup is a method of offloading backup procedures from a server so that the time ordinarily devoted to backup functions can be used to carry out other server tasks.

storage at the edge

Storage at the edge is an expression that refers to data storage and backup routines used in portable and mobile computing.

storage filer

A storage filer is a file server designed and programmed for high-volume data storage, backup, and archiving.

storage snapshot

A storage snapshot is a set of reference markers, or pointers, to data stored on a disk drive, on a tape, or in a storage area network (SAN).

tape backup

Tape backup is the ability to periodically copy the contents of all or a designated amount of data from its usual storage device to a tape cartridge device so that, in the event of a hard disk crash or comparable failure, the data will not be lost.

tape library

In data storage, a tape library is a collection of magnetic tape cartridges and tape drives.

virtual tape

Virtual tape is the use of a special storage device that manages less-frequently needed data so that it appears to be stored entirely on tape cartridges when some parts of it may actually be located in faster, hard disk storage.

SAP HANA on AWS Implementation and Operations Guide

http://awsmedia.s3.amazonaws.com/SAP_HANA_on_AWS_Implementation_and_Operations_Guide.pdf December 09, 2014

SAP HANA Disaster Recovery using System ... The SAP on AWS Backup and Recovery Guide provides guidelines on how to ... Additional recommendations for this approach:

About this Guide

This guide provides best practice guidelines for implementing and operating the SAP HANA Infrastructure Subscription offering on Amazon Web Services (AWS). The intended audience of this guide is SAP customers and partners. This guide is not intended to replace any of the standard SAP HANA documentation. SAP Administration installation guides and notes can be found at:

- SAP Library (help.sap.com) - SAP HANA Administration Guide
- SAP Installation Guides
- SAP Notes

This guide assumes that you have a basic knowledge of Amazon Web Services. If you are new to AWS please read the following guides before continuing with this guide.

- Getting Started with AWS
- What is Amazon EC2?
- SAP on AWS Implementation Guide

Additional SAP on AWS Documentation

In addition to this guide the following SAP on AWS guides can be found at http://aws.amazon.com/sap > Resources

SAP on AWS Operations Guide
The SAP on AWS Operations Guide provides guidelines on the special considerations that must be taken into account when operating SAP environments on AWS.

SAP on AWS High Availability Guide
The SAP on AWS High Availability Guide provides guidelines on how to configure SAP systems on Amazon EC2 in such a way as to be able to protect the application from various single points of failure.

SAP on AWS Backup and Recovery Guide
The SAP on AWS Backup and Recovery Guide provides guidelines on how to backup SAP systems running on AWS. The guide focuses on the essential differences in backing up SAP systems on AWS compared to traditional infrastructure.

Overview of SAP HANA on AWS

AWS and SAP have worked together closely over the past couple of years to make SAP HANA available on the flexible AWS platform. Today there are multiple SAP HANA offerings available on AWS. An overview of the different offerings is

provided in the following section.

SAP HANA Developer Edition

Description	Fully featured SAP HANA virtual appliance on AWS for individual developers
Use Cases	Non-production only Develop, test and demo applications Learning environment
HANA Licensing	Free license provided by SAP for developers
Available from	SAP SCN
EC2 instance types	m2.xlarge / m2.2xlarge / m2.4xlarge
Number of nodes	1
HANA memory	17.1 GiB / 34.2 GiB / 68.4 GiB

SAP HANA One

Description	Fully featured SAP HANA virtual appliance on AWS
Use Cases	Production and non-production Analytics acceleration Data merging Temporary event based analytics Self-service BI Prototypes and proofs-of-concept No connection to SAP-licensed products other than Lumira permitted
HANA Licensing	$0.99 p/hour on-demand license from SAP via the AWS Marketplace
Available from	AWS Marketplace
EC2 instance types	cc2.8xlarge
Number of nodes	1
HANA Memory	60.5 GiB

SAP HANA Infrastructure Subscription

Description	Fully featured SAP HANA virtual appliance on AWS
Use Cases	Production, non-production, POC's, DR All SAP HANA use cases supported for non-production scenarios on single node and multi-node HANA virtual appliances. SAP BW supported in production on single node HANA virtual appliance Multi-node for BW and Business Suite use cases coming soon.
HANA Licensing	Bring-your-own-License. Customers must have a current license for the SAP HANA Database.
Available from	SAP HANA Marketplace
EC2 instance types	cr1.8xlarge
Number of nodes	1-5
HANA Memory	244 GiB / 488 GiB / 732 GiB / 976 GiB / 1.22TiB

Sizing

SAP HANA is offered on AWS in both single node and multi-node configurations with a total of 244, 488, 732, 976, and 1220 GiB RAM. Since HANA is a columnar database it requires less storage to store data compared to a traditional row based RDMS. Data is highly compressed and compression ratios can range from 3:1 to over 10:1 based on the source data and source database.

As far as sizing of the HANA appliance is concerned, main memory is the most important resource. There are various sizing methods depending on the implementation scenario but in general the following methods apply:

To obtain sizing information for a system that has not yet been implemented, use the SAP QuickSizer. Please go to http://service.sap.com/quicksizer for further details. The SAP QuickSizer will provide information on both the SAP HANA In-Memory Database and the SAP NetWeaver application server where applicable.

To migrate an existing SAP NetWeaver BW system from any database platform to HANA, SAP strongly recommends to use the new ABAP sizing report for SAP NetWeaver BW described in SAP note 1736976.

To migrate an already existing Business Suite System to HANA, it's recommended to use SAP note 1872170 to estimate the main memory requirements of the HANA virtual appliance.

Note: Further sizing information is also available in the SAP HANA Administration Guide.

SAP Note #	Description
1736976	Sizing Report for BW on HANA
1637145	SAP BW on HANA: Sizing SAP In-Memory Database
1702409	HANA DB: Optimal number of scale out nodes for BW on HANA
1855041	Sizing Recommendation for Master Node in BW-on-HANA
1793345	Sizing for SAP Suite on HANA
1872170	Suite on HANA memory sizing

Table 1: Common SAP HANA Sizing Notes

If memory requirements for the SAP HANA solution exceed the available memory of a single AWS instance, a scale out solution consisting of multiple instances can be deployed as long as the SAP solution being deployed supports a scale-out configuration.

Solution Architecture

The SAP HANA on AWS Infrastructure Subscription can be deployed in either a single node or multi-node architecture configuration consisting of up to 5 HANA nodes.

AWS Architecture Components

Single and Multi-node deployments automatically provision and deploy and stitch together all the necessary AWS components into a customer's AWS account using AWS CloudFormation. AWS CloudFormation provides an easy way to create and manage a collection of related AWS resources, provisioning and updating them in an orderly and predictable fashion.

The following components are deployed and configured as part of this offering:

- An AWS Virtual Private Cloud (VPC) configured with two subnets, one public, and the other private.
- A NAT instance deployed into the public subnet and configured with an Elastic IP Address (EIP) for outbound Internet connectivity and inbound SSH access.
- A Windows Server deployed in the public subnet with HANA Studio preloaded.
- An Identity and Access Management (IAM) instance role with fine-grained permissions for backup and failure recovery.
- An S3 Bucket where HANA Backups can be stored.
- An SAP HANA System installed with the proper EBS storage volumes configuration for HANA performance needs.

- Pre-configured security groups.
- Single node or multi-node SAP HANA virtual appliances automatically configured per SAP best practices.

Single Node Architecture

Figure 1: Single Node Architecture

Multi-Node Architecture

Multi-node deployments additionally automatically install the worker nodes based on the deployment selection for number of nodes. Worker nodes are also deployed into the same subnet as the Master node.

Figure 2: Multi-Node Architecture

Advanced Configurations

The provisioning process starts on the saphana.com website where the user is required to enter their AWS account. Upon submission, SAP grants access to a private Amazon Machine Image (AMI), which is used during the deployment process. After selecting the number of HANA nodes desired the users browser is redirected to an AWS CloudFormation template depending on the number of nodes selected. At this point a custom CloudFormation template can be substituted instead in order to "customize" the deployment. For example, if a customer already had an existing VPC where they wanted to deploy the solution this could be accomplished by specifying additional parameters upfront. See appendix A for sample custom CloudFormation templates.

Storage Architecture

In order to meet the HPC requirements of SAP HANA, the storage configuration used for SAP HANA on AWS is optimized for both price and performance based on KPI's provided by SAP through the SAP HANA Tailored Datacenter Integration program. As long as the deployment is done using the standard provisioning process through saphana.com and AWS CloudFormation, the storage configuration is built using an SAP supported configuration.

The storage configuration for SAP HANA on AWS is based on Elastic Block Store (EBS) Provisioned IOPS (P-IOPS) volumes. AWS Elastic Block Store (EBS) provides persistent block level storage volumes for use with EC2 instances. EBS volumes are off-instance storage that persists independently from the life of an instance.
Provisioned IOPS volumes offer storage with consistent low-latency performance, and are designed for applications with I/O-intensive workloads such as SAP HANA. Backed by Solid-State Drives (SSDs), provisioned IOPS volumes can achieve single digit millisecond latencies and are designed to deliver within 10% of the provisioned IOPS performance 99.9% of the time. Furthermore, volume striping allows for significant IOPS and throughput performance.

Each Amazon EBS volume is automatically replicated within its Availability Zone to protect from component failures, offering high availability and durability. As such, the production configuration is based on 12 x 200GB x 2000 P-IOPS volumes striped together in a Raid-0 configuration. Each SAP HANA node carries the same EBS configuration regardless of whether it is configured as Master or worker node.

The solution also uses a shared nothing storage concept for the data and log area so a single HANA node failure does not impact the availability of all the storage for the solution. However, the backup and HANA Shared file systems are owned by the HANA Master node and shared via NFS to all worker and standby nodes as per SAP best practices.

Figure 3: EBS Persistence Architecture

> **Tip**
> EBS Standard storage volumes can easily be substituted for non-production and proof of concept environments when storage performance is not critical. See Appendix A for modified CloudFormation templates and instructions.

Deployment

Preparation

1. Create an Amazon Web Services (AWS) account, if needed. – http://aws.amazon.com
2. Choose an EC2 Region to deploy the SAP HANA on AWS solution.

   Amazon EC2 locations are composed of Regions and Availability Zones. Regions are dispersed and located in separate geographic areas. Currently, the BYOL version of HANA on AWS can be deployed in the following AWS regions:

   - US, Northern Virginia (us-east-1)
   - US, Oregon (us-west-2)
   - Ireland, EU (eu-west-1)
   - Tokyo, Japan (ap-northeast-1)

   Note:
   Consider choosing a region closest to your data center and/or corporate network to reduce network latency between systems running on AWS and systems and users on your corporate network.

3. Choose an Availability Zone within the region of your choice.

   Availability Zones are distinct locations within a Region that are engineered to be isolated from failures in other Availability Zones and provide inexpensive, low latency network connectivity to other Availability Zones in the same Region.

   Note:
   In most cases, choosing the first availability zone in your region should be sufficient. For example, in us-east-1, the first availability zone would be us-east-1a. For us-west-2, this would be us-west-2a and so forth.

   To find the availability zones available in your particular region:
   a. Sign in to the AWS Management Console and open the Amazon EC2 console at https://console.aws.amazon.com/ec2/
   b. From the navigation bar, view the options in the region selector.

Figure 4: Region Selection

   c. After you select a region, you can view your Availability Zones within that region directly on the main EC2 page.

Figure 5: Availability Zones

Tip
In the case of us-east-1 (Virginia) and ap-northeast-1 (Tokyo) there are Availability Zones that do not support VPC. If you receive the message “Value for parameter availability zone is invalid. Subnets can currently only be created in the following availability zones,“ you will need to choose a different availability zone for your deployment.

4. Create a key-pair in your preferred region.

Amazon EC2 uses public–key cryptography to encrypt and decrypt login information. To be able to log into your instances, you must create a key pair. You will use this key pair to log into the Linux instance where HANA is installed using SSH. With Windows instances, use the key pair to obtain the administrator password via the EC2 console and then log in using RDP.
Step-by-step instructions

5. If applicable, request EC2 and/or EBS limit increases.

a. If you plan on deploying more than a single node, please request a limit increase for Elastic Block Store (EBS) Provisioned IOPS volumes here. Request 24,000 x the number of nodes you plan on deploying. For example a 5-node deployment you would add to your current Provisioned IOPS limit 5*24000, so that you would request from AWS 10,000 + 120,000 Provisioned IOPS. 10,000 is the default provisioned IOPS limit. There is no charge associated with extending the limit for Provisioned IOPS.

Figure 6: Sample EBS Limit Increase Request

b. If you plan on deploying more than two SAP HANA nodes, please request a limit increase for the CR1 instance type here. By default, each AWS account starts with a limit of 2.

Figure 7: Sample EC2 Limit Increase Request

Receive SAP HANA Images

1. Navigate to the SAP HANA Marketplace offering.
2. Click Deploy Now.
3. Enter your Amazon Web Services account number and, if known, your SAP customer id.
4. Click OK.
5. Your screen will now look like the following:

Figure 8: Deployment Selection

Deploy the SAP HANA Solution

1. Select the AWS Region and SAP HANA Size.
2. For the SAP HANA Size 244 GiB you can proceed to Launch and skip the following steps.
3. If not already done previously, click on the link “request AWS” to increase the limit for Provisioned IOPS to the specified number. Also request a limit increase for the “High Memory Cluster Eight XL” instance type if deploying more than 2 nodes. Wait until Amazon completes the request (usually within one business day).
4. Click Launch and log into your AWS account if needed.
5. Specify a name of the Stack

Figure 9: Cloud Formation – Choose Stack Name

6. On the next page:
   a. Specify a SID for the HANA System.
   b. Specify a CIDR range that will have SSH access to the NAT instance (TCP/22) and RDP access (TCP/3389) to the HANA Studio instance in the public subnet. A default of 0.0.0.0/0 will allow access from any IP address.

      Important:
      As a security best practice, we recommend you restrict this to your own specific CIDR Range or IP address.

   c. Enter a Master Password. This password will be used to set the password for OS users <sid>adm, sapadm, and the HANA SYSTEM DB user.
   d. Enter the Availability Zone of your choice from step 2.3c above.
   e. Specify the Key-Pair name created in Step 2.4 above.

Figure 10: Cloud Formation – Deployment Parameters

   f. The screen “Add Tags” is optional. Information on how tagging works in AWS can be found here.
   g. Review the information and press Continue to initiate the provisioning. If you receive any warnings about the parameters you entered, use the back selection to go back and fix them.
   h. Wait until the Stack is marked as CREATE_COMPLETED.

7. Monitor the provisioning process.

   a. You will immediately be able to track the status of the deployment process in the description tab of the CloudFormation stack.

Figure 11: Cloud Formation – Overall Deployment Status

b. To see progress of the individual component and system deployments, navigate to the Events tab. Here you can monitor the progress of the entire CloudFormation stack deployment process.

Figure 12: Cloud Formation – Overall Deployment Status

Note:
Single node SAP HANA deployments can take anywhere from 10-15 minutes.
Multi-node SAP HANA deployments will take 10-15 minutes for the master node and an additional 10-15 minutes for all worker nodes as all worker nodes are deployed in parallel.

c. Once the create process is complete you will see the stack marked as CREATE_COMPLETED.

Figure 13: Cloud Formation – Create Complete

d. If you encounter the status message ROLLBACK_IN_PROGRESS or ROLL_BACK_COMPLETE, please see the next section for troubleshooting.

Troubleshooting

Most provisioning errors can be attributed to problems with account limits. If you see a ROLLBACK_IN_PROGRESS or ROLLBACK_COMPLETE status message check the events tab of the failed CloudFormation stack to determine which resource first attributed to the ROLLBACK event.

Figure 14: CloudFormation – Rollback Example

Start from the bottom and scroll up until you see the first CREATE_FAILED event. You may need to scroll to the right to see the actual error message.

Figure 15: CloudFormation – Create Failed

If you get an error that the "instance did not stabilize" (as below) this means you have exceeded your IOPS for the region and need to request an increase.

Figure 16: CloudFormation – Instance Did Not Stabilize

If you get an error “Value for parameter availabilityZone is invalid. Subnets can currently only be created in the following availability zones,“ you will need to choose a different availability zone for your deployment.

Figure 17: CloudFormation – Choose another Availability Zone

Getting Access to SAP HANA

The default network security setup of this solution follows security best practices of AWS. The provisioning logic creates the solution architecture described in the solution architecture section with the SAP HANA instances in a private subnet to restrict direct exposure to the Internet. As such, the SAP HANA instances can only be accessed through instances placed in the public subnet or DMZ layer.

Through this DMZ layer, two methods of access are available.

- HANA Studio Access
  Connect to the Windows Instance using a Remote Desktop Client where HANA Studio has been preloaded.
- OS Level Access
  SSH to the NAT instance and then to the SAP HANA instance(s) using a SSH client of your choice.

Tip
To connect directly to the SAP HANA systems from a corporate network, you can provision an encrypted IPsec hardware VPN connection between your corporate datacenter and your VPC. See http://aws.amazon.com/vpc/ for more details.

HANA Studio Access using the RDP Instance

1. In the output window please note down the Elastic IP address (EIP) of the RDP Instance.

Figure 18: Cloud Formation – RDP Server IP info

2. Get the Windows Administrator Password from the EC2 console.
   - Go to Services -> EC2 -> Instances -> Select your RDP Instance
   - Choose Connect -> Get password
   - Choose or paste in the contents of your private key in the space provided.
   - The password will be decrypted and shown to you.
3. Choose Download Remote Desktop File or connect via an RDP client of your choice.
4. Start HANA Studio and add a System
   - IP Address or hostname of Master Node (imdbmaster)
   - Instance Number: 00
   - User: SYSTEM
   - Password: < your password from 5.3.c >

Figure 19: HANA Studio – SAP HANA Overview

Figure 20: HANA Studio – Landscape Status

Note
We recommend you take a backup at this point. This can be done via HANA Studio for HANA. You can also take complete system image (Amazon Machine Image) through the EC2 console for recovery later.

SSH Access

1. Navigate to Services -> EC2 -> Instances and find your NAT instance and note the public Elastic IP Address.

Figure 21: NAT – Elastic IP Address

2. Using an ssh client of your choice (i.e. Putty or ITerm), ssh into the NAT instance using the key-pair specified during the deployment process.

Note
If your connection times out, you may need to adjust the security group rules for the NAT instance to allow access from your computers IP address or proxy server.

ITerm Example:
- Add private key to authentication agent (ssh-add)
- ssh to NAT instance with –A option to forward the key.
- Note that entries for the servers hosting SAP HANA have already been maintained in /etc/hosts.
- ssh to the SAP HANA server

Figure 22: SSH – ITERM Example

Putty Example:
a. Download Putty.exe, Puttygen.exe, and pageant.exe
b. Load your private key into Puttygen and save as ppk file that putty can use.
c. Execute Pageant.exe, and add your new ppk key. The Pageant process must be running in order for agent forwarding to work.
d. Configure putty with the private key and choose allow agent forwarding.

Figure 23: SSH – Putty Example

e. Save configuration
f. SSH to NAT instance, then to HANA node

Figure 24: SSH – Putty Example Continued

Administration

Start / Stop of EC2 instances running SAP HANA Hosts

At any time one or multiple SAP HANA Hosts can be stopped. Before stopping the EC2 instance of an SAP HANA host, it is recommended to first stop SAP HANA on that instance. When resuming the EC2 Instance, the instance will automatically be started with the same IP address, network, and storage configuration as before.

Creating an Image of a SAP HANA System

There are multiple reasons for creating an image of a SAP HANA System. These include:

- Create a full system backup (OS, /usr/sap, HANA Shared, Backup, Data, Log) via Amazon Machine Image (AMI). Amazon Machine Images are automatically saved in 3 different availability zones within the same region.
- Change Storage Performance
  During instantiation of an Image it is possible to specify EBS performance ranging from EBS Standard to EBS Provisioned IOPS with 4000 IOPS per Volume. The default storage performance for SAP HANA is 2000 IOPS per Volume. Storage performance has a significant impact on AWS infrastructure cost.
- Relocating a HANA system from one region to another.
  This can be done by leveraging Image Copy and specify the new target region. The SAP HANA System can be resumed in the new region

Tip
The SAP HANA system should be in a consistent state before creating an Amazon Machine Image (AMI). This can be accomplished by stopping the SAP HANA Instance before creation or by following instructions in SAP Note 1703435.

Cloning a SAP HANA System

Cloning a SAP HANA System via imaging and re-instantiating is currently only supported for a HANA system with a single Host.

In order to clone a multi-Host SAP HANA deployment:
1. Provision a new SAP HANA system with the same configuration.
2. Perform a data backup of the original system
3. And restore the backup on the new system.

Backup/Recovery

Apart from some examples, this guide does not include detailed instructions how to execute database backups using either native HANA backup/recovery features or 3rd party backup tools. Please refer the standard OS, SAP and SAP HANA documentation or the documentation provided by the backup software vendor. In addition, backup schedules, frequency, and retention periods, are primarily based on your system type and business requirements. Please refer to the standard SAP documentation for guidance on these topics.

Note
Both general and advanced backup and recovery concepts for SAP Systems on AWS can be found in detail in the SAP on AWS Backup and Recovery Guide.

SAP Note #	Description
1642148	FAQ: SAP HANA Database Backup & Recovery
1821207	Determining required recovery files
1869119	Checking backups using hdbbackupcheck
1873247	Checking recoverability with hdbbackupdiag --check
1651055	Scheduling SAP HANA Database Backups in Linux

AWS Services and Components for Backup Solutions

Simple Storage Service (S3) – http://aws.amazon.com/s3
Amazon S3 provides a highly durable storage infrastructure designed for mission-critical and primary data storage. Amazon S3 is designed to provide 99.999999999% durability and 99.99% availability over a given year. Amazon S3 is the center of any SAP backup and recovery solution on AWS.

The deployment process automatically creates a private S3 bucket where SAP HANA backups can be stored off instance to provide more protection and durability. Only the AWS account that is used to create the bucket has access to this bucket. The S3 Bucket follows the naming convention <template-name-randomly_chosen_characters> (for example: node2-hana-s3bucket-gcynh5v2nqs3).

Figure 25: SSH – S3 Bucket Example

Note
Additional S3 buckets can be created if needed through the AWS console or using the AWS command line interface.

AWS Identity and Access Management
AWS Identity and Access Management (IAM) enables you to securely control access to AWS services and resources for your users. Using IAM, you can create and manage AWS users and groups and use permissions to allow and deny their access to AWS resources. You can create roles in IAM, and manage permissions to control which operations can be performed by the entity, or AWS service, that assumes the role. You can also define which entity is allowed to assume the role.

An IAM role allowing access to get/put objects to and from S3 created during the CloudFormation deployment process and is subsequently assigned to each AWS instance hosting SAP HANA master and worker nodes at launch time as they are deployed.

Figure 26: SSH – IAM Role Example

To ensure security using the principle of least privilege, permissions for this role are limited to only actions that are required for backup and recovery functions.

```
{"Statement":[
 {"Resource":"arn:aws:s3::: node2-hana-s3bucket-gcynh5v2nqs3/*",
   "Action":["s3:GetObject","s3:PutObject","s3:DeleteObject", "s3:ListBucket","s3:Get*","s3:List*"],
   "Effect":"Allow"},
 {"Resource":"*","Action":["s3:List*","ec2:Describe*","ec2:AttachNetworkInterface",
  "ec2:AttachVolume","ec2:CreateTags","ec2:CreateVolume","ec2:RunInstances",
  "ec2:StartInstances"],"Effect":"Allow"}]}
```

If additional functions are later desired, the IAM Role can be modified using the AWS Console.

Amazon Glacier – http://aws.amazon.com/glacier
Amazon Glacier is an extremely low-cost storage service that provides secure and durable storage for data archiving and backup. In order to keep costs low, Amazon Glacier is optimized for data that is infrequently accessed and for which retrieval times of several hours are suitable. With Amazon Glacier, customers can reliably store large or small amounts of data for as little as $0.01 per gigabyte per month, a significant savings compared to on-premises solutions. SAP HANA backups can be pushed to Glacier for long-term archival using lifecycle policies.

SAP HANA Backup Destination

The primary difference between backing up SAP systems on Amazon Web Services compared to traditional on-premises infrastructure is the backup destination. The typical backup destination used with on-premises infrastructure is tape. On AWS, instead of storing backups on tape, backups are stored in Amazon S3. There are many benefits to storing backups in Amazon S3 vs. tape. Backups stored in Amazon S3 are automatically stored “offsite” from the source system since data in Amazon S3 is replicated across multiple facilities within the AWS region.

SAP HANA Data backups can be triggered and/or scheduled using SAP HANA studio, SQL commands, or the DBA Cockpit. While log backups are written automatically (unless disabled). The /backup file system has been configured as part of the deployment process.

Figure 27: SSH – File system Layout

The SAP HANA global.ini configuration file has been customized as follows. Database backups go directly to /backup/data/<SID> while automatic log archival files go to /backup/log/<SID>.

```
[persistence]
basepath_shared = no
savepoint_intervals = 300
basepath_datavolumes = /hana/data/<SID>
basepath_logvolumes = /hana/log/<SID>
basepath_databackup = /backup/data/<SID>
basepath_logbackup = /backup/log/<SID>
```

AWS Command Line interface
The AWS Command Line Interface (CLI), which is a unified tool to manage AWS services, has already been installed as part of the base image. Using various commands you are able to control multiple AWS services from the command line directly and automate them through scripts. Access to the S3 bucket is obtained through the aforementioned IAM role assigned to the instance. Using the AWS S3 commands, we can list the contents of the previously created bucket, backup files, and restore files.

```
imdbmaster:/backup # aws s3 ls --region=us-east-1 s3://node2-hana-s3bucket-gcynh5v2nqs3

Bucket: node2-hana-s3bucket-gcynh5v2nqs3
Prefix:
   LastWriteTime Length Name
   ------------- ------ ----
```

Backup Example
1. In the SAP HANA Backup editor, choose “Open Backup Wizard.” Right-clicking the system that you want to back

up and choose “Back Up” can also open the backup wizard.
2. Select destination type “File.” This will back up the database to files in file system specified.
3. Specify the backup destination (/backup/data/<SID>) and the backup prefix.

Figure 28: SSH — Backup Example

4. Chose next, and Finish
5. When the backup is complete a confirmation message will be displayed.
6. Verify the backup files are available at the operating system level.

```
imdbmaster:/backup # ll */*

data/YYZ:
total 1588080
-rw-r--r-- 1 yyzadm sapsys 163840 Oct 28 18:44 COMPLETE_DATA_BACKUP_databackup_0_1
-rw-r--r-- 1 yyzadm sapsys 70443008 Oct 28 18:44 COMPLETE_DATA_BACKUP_databackup_1_1
-rw-r--r-- 1 yyzadm sapsys 1000955904 Oct 28 18:44 COMPLETE_DATA_BACKUP_databackup_2_1
-rw-r--r-- 1 yyzadm sapsys 69292032 Oct 28 18:44 COMPLETE_DATA_BACKUP_databackup_3_1
-rw-r--r-- 1 yyzadm sapsys 101605376 Oct 28 18:44 COMPLETE_DATA_BACKUP_databackup_4_1
-rw-r--r-- 1 yyzadm sapsys 98521088 Oct 28 18:44 COMPLETE_DATA_BACKUP_databackup_5_1
-rw-r--r-- 1 yyzadm sapsys 69488640 Oct 28 18:44 COMPLETE_DATA_BACKUP_databackup_6_1
-rw-r--r-- 1 yyzadm sapsys 136269824 Oct 28 18:44 COMPLETE_DATA_BACKUP_databackup_7_1

log/YYZ:
total 34928
-rw-r--r-- 1 yyzadm sapsys   12288 Oct 28 18:44 log_backup_0_0_0_0.1382985855848
-rw-r--r-- 1 yyzadm sapsys   12288 Oct 28 18:44 log_backup_0_0_0_0.1382985856054
-rw-r--r-- 1 yyzadm sapsys   12288 Oct 28 18:44 log_backup_0_0_0_0.1382985856098
-rw-r--r-- 1 yyzadm sapsys   12288 Oct 28 18:44 log_backup_0_0_0_0.1382985856110
-rw-r--r-- 1 yyzadm sapsys   12288 Oct 28 18:44 log_backup_0_0_0_0.1382985860695
-rw-r--r-- 1 yyzadm sapsys   12288 Oct 28 18:44 log_backup_0_0_0_0.1382985864944
-rw-r--r-- 1 yyzadm sapsys   16384 Oct 28 18:44 log_backup_0_0_0_0.1382985864955
-rw-r--r-- 1 yyzadm sapsys   16384 Oct 28 18:59 log_backup_0_0_0_0.1382986752676
```

7. The next step is to push or synchronize the backup files from the /backup file system to S3 using the AWS S3 CLI.

```
imdbmaster:/ # aws s3 sync backup s3://node2-hana-s3bucket-gcynh5v2nqs3 --region=us-east-1

upload: ../backup/data/YYZ/COMPLETE_DATA_BACKUP_databackup_0_1 to s3://node2-hana-s3bucket-
gcynh5v2nqs3/data/YYZ/COMPLETE_DATA_BACKUP_databackup_0_1
upload: ../backup/data/YYZ/COMPLETE_DATA_BACKUP_databackup_1_1 to s3://node2-hana-s3bucket-
gcynh5v2nqs3/data/YYZ/COMPLETE_DATA_BACKUP_databackup_1_1
upload: ../backup/data/YYZ/COMPLETE_DATA_BACKUP_databackup_2_1 to s3://node2-hana-s3bucket-
gcynh5v2nqs3/data/YYZ/COMPLETE_DATA_BACKUP_databackup_2_1
upload: ../backup/data/YYZ/COMPLETE_DATA_BACKUP_databackup_3_1 to s3://node2-hana-s3bucket-
gcynh5v2nqs3/data/YYZ/COMPLETE_DATA_BACKUP_databackup_3_1
upload: ../backup/data/YYZ/COMPLETE_DATA_BACKUP_databackup_4_1 to s3://node2-hana-s3bucket-
gcynh5v2nqs3/data/YYZ/COMPLETE_DATA_BACKUP_databackup_4_1
upload: ../backup/data/YYZ/COMPLETE_DATA_BACKUP_databackup_5_1 to s3://node2-hana-s3bucket-
gcynh5v2nqs3/data/YYZ/COMPLETE_DATA_BACKUP_databackup_5_1
upload: ../backup/data/YYZ/COMPLETE_DATA_BACKUP_databackup_6_1 to s3://node2-hana-s3bucket-
gcynh5v2nqs3/data/YYZ/COMPLETE_DATA_BACKUP_databackup_6_1
   upload: ../backup/log/YYZ/log_backup_0_0_0_0.1382985855848 to s3://node2-hana-s3bucket-
   gcynh5v2nqs3/log/YYZ/log_backup_0_0_0_0.1382985855848
   upload: ../backup/log/YYZ/log_backup_0_0_0_0.1382985856054 to s3://node2-hana-s3bucket-
   gcynh5v2nqs3/log/YYZ/log_backup_0_0_0_0.1382985856054
   upload: ../backup/log/YYZ/log_backup_0_0_0_0.1382985856098 to s3://node2-hana-s3bucket-
   gcynh5v2nqs3/log/YYZ/log_backup_0_0_0_0.1382985856098
   upload: ../backup/log/YYZ/log_backup_0_0_0_0.1382985856110 to s3://node2-hana-s3bucket-
   gcynh5v2nqs3/log/YYZ/log_backup_0_0_0_0.1382985856110
   upload: ../backup/log/YYZ/log_backup_0_0_0_0.1382985860695 to s3://node2-hana-s3bucket-
   gcynh5v2nqs3/log/YYZ/log_backup_0_0_0_0.1382985860695
   upload: ../backup/log/YYZ/log_backup_0_0_0_0.1382985864944 to s3://node2-hana-s3bucket-
   gcynh5v2nqs3/log/YYZ/log_backup_0_0_0_0.1382985864944
   upload: ../backup/log/YYZ/log_backup_0_0_0_0.1382985864955 to s3://node2-hana-s3bucket-
   gcynh5v2nqs3/log/YYZ/log_backup_0_0_0_0.1382985864955
```

8. Verify the files have been pushed to S3 through the AWS Console or with the “aws s3 ls” command shown previously.

Figure 29: S3 Bucket Contents

> Tip
>
> The S3 sync command will only upload new files that don’t exist in S3. Use a periodic scheduled cron job to sync then delete files that have been uploaded. See note 1651055 for scheduling periodic backup jobs in Linux and extend the supplied scripts with the AWS S3 sync commands.

Restore Example

1. If the backup files are not readily available already in the /backup file system but are in S3, restore the files from S3 using the AWS S3 CLI command “aws --region <region> cp <s3-bucket/path> --recursive <backup-prefix>*”

```
Example:
imdbmaster:/backup/data/YYZ # aws --region us-east-1 s3 cp s3://node2-hana-s3bucket-gcynh5v2nqs3/data/YYZ . --
recursive --include COMPLETE*
download: s3://node2-hana-s3bucket-gcynh5v2nqs3/data/YYZ/COMPLETE_DATA_BACKUP_databackup_0_1 to
./COMPLETE_DATA_BACKUP_databackup_0_1
download: s3://node2-hana-s3bucket-gcynh5v2nqs3/data/YYZ/COMPLETE_DATA_BACKUP_databackup_1_1 to
./COMPLETE_DATA_BACKUP_databackup_1_1
download: s3://node2-hana-s3bucket-gcynh5v2nqs3/data/YYZ/COMPLETE_DATA_BACKUP_databackup_2_1 to
./COMPLETE_DATA_BACKUP_databackup_2_1
download: s3://node2-hana-s3bucket-gcynh5v2nqs3/data/YYZ/COMPLETE_DATA_BACKUP_databackup_3_1 to
./COMPLETE_DATA_BACKUP_databackup_3_1
download: s3://node2-hana-s3bucket-gcynh5v2nqs3/data/YYZ/COMPLETE_DATA_BACKUP_databackup_4_1 to
./COMPLETE_DATA_BACKUP_databackup_4_1
```

```
      download: s3://node2-hana-s3bucket-gcynh5v2nqs3/data/YYZ/COMPLETE_DATA_BACKUP_databackup_5_1 to
./COMPLETE_DATA_BACKUP_databackup_5_1
      download: s3://node2-hana-s3bucket-gcynh5v2nqs3/data/YYZ/COMPLETE_DATA_BACKUP_databackup_6_1 to
./COMPLETE_DATA_BACKUP_databackup_6_1
      download: s3://node2-hana-s3bucket-gcynh5v2nqs3/data/YYZ/COMPLETE_DATA_BACKUP_databackup_7_1 to
./COMPLETE_DATA_BACKUP_databackup_7_1
```

2. Recover the SAP HANA database using the recovery wizard as outlined in the SAP HANA Administration Guide, being sure to specify file as the destination type and the correct backup prefix.

Figure 30: Restore Example

3. When the recovery is complete, resume operation and cleanup backup files from /backup/<SID>/* directories.

SAP Support Access

In some situations it may be necessary to allow an SAP support engineer to access your SAP HANA Systems on AWS. This information serves only as a supplement to the information contained in “Getting Support” section of the SAP HANA Administration guide.

There are a few steps that need to be followed in order to configure proper connectivity to SAP. These steps differ depending on whether you want to leverage an existing remote network connection to SAP or if you are setting up a new connection directly with SAP from systems on AWS.

Support Channel Setup with SAProuter on AWS

When setting up a support to connection to SAP from AWS directly, consider the following steps:

- Configure a specific SAProuter Security Group SAProuter instance, which only allows the required inbound and outbound access to the SAP support network. This should be limited to a specific IP address SAP gives you to connect to along with TCP port 3299.
- The instance that the SAProuter software will be installed on should be launched into a public subnet of the VPC and should be assigned an Elastic IP Address (EIP).
- Install the SAProuter software and create a saprouttab file allowing access from SAP to your SAP HANA systems on AWS.
- Setup the connection with SAP. The type of Internet connection that should be used is Secure Network Communication (SNC), see https://service.sap.com/internetconnection
- Modify the existing SAP HANA security groups to trust the SAProuter Security Group.

Tip
For added security, shut down the AWS instance hosting the SAProuter service when it is not needed for support purposes.

Figure 31: Support Connectivity with SAProuter on AWS

Support Channel Setup with SAProuter on-premises

In many cases a customer will already have a support connection configured between their own datacenter and SAP. This can easily be extended to allow for support of SAP systems on AWS. This scenario assumes connectivity between the customers datacenter and AWS has already been established either by way of a secure VPN tunnel over the internet or by using AWS Direct Connect.

There are a only a few steps to perform to extend this connectivity:

- Ensure the proper saprouttab entries exist to allow access from SAP to resources in the AWS VPC.
- Modify the SAP HANA Security groups to allow access from the on-premises SAProuter IP address.
- Ensure the proper firewall ports are open on the customer gateway to allow traffic to pass over TCP port 3299.

Figure 32: Support Connectivity with SAProuter On-Premises

High Availability / Disaster Recovery

This section outlines number of options for ensuring the SAP HANA system is deployed in a highly available manner. Your particular approach should only be decided after discussions with key stakeholders to understand availability requirements in terms of both recovery point and recovery time objectives (RPO/RTO).

Spare AWS Capacity

Sometimes with on-premises deployments, customers choose to purchase additional hardware to protect the SAP HANA environment in case of a hardware failure. On AWS, this may not be depending on your availability requirements. Instead of failing over to a “standby server,”, you can simply start the failed virtual machine back up again and your virtual machine will be placed on a new physical host. Keep in mind, this solution is not the same as a hot standby as the SAP HANA DB will be unavailable for the time it takes to boot the virtual machine back up. However, if some downtime can be tolerated, this can save a considerable amount of money for your business.

Figure 33: Leveraging Spare Capacity

Additional recommendations for this approach:

Leverage Reserved Instances

Reserved Instances can potentially provide significant cost savings depending on usage model. In addition, Reserved Instances provide a capacity reservation so that you can have confidence in your ability to launch the number of instances you have reserved when you need them.

Health Monitor

We recommend you configure a monitoring solution external to the SAP HANA System that can detect the availability of the SAP HANA Solution. Upon failure detection you can simply script appropriate actions to take based on your scenario and availability requirements.

For Example:

- Check the instance Status and availability using the AWS Command Line Interface (CLI)

```
aws ec2 describe-instance-status --region <region> --instance-ids <instance-id>..<instance-id>
```

- If the state of the instance is stopped, just issue the start-instances command.

```
aws ec2 start-instances --region us-east-1 --instance-ids <instance-id>
```

- If either of the status checks show as failed you may have an impaired host and should restart your instance.
  aws ec2 stop-instances --region us-east-1 --instance-ids <instance-id>

  Tip
  Generally it's best to allow an instance to gracefully shutdown. However, if you have issued a stop command and the instance appears to be stuck in this state you can issue the stop-instances command with the –force flag. This means the instance does not have an opportunity to flush file system caches or file system metadata. If you use this option, you must perform file system check and repair procedures.

SAP HANA High Availability using System Replication – Single Region
SAP HANA now supports system replication, which provides for a continuous update of a secondary set of HANA systems by the primary system. System replication is documented in detail in the SAP HANA Administration guide but in general system replication is configured such that the secondary systems are configured as copies of the primary systems. The number of active hosts in each system must be identical. Each SAP HANA service on the primary HANA instances communicates with its counterpart on the secondary system.

System replication can be configured for either asynchronous or synchronous replication. In synchronous mode, the primary system only commits a transaction after it has received acknowledgment from the secondary system that it has received the changes. This provides immediate consistency and provides the highest protection from data loss. While this works well for primary and secondary systems deployed in close proximity, care should be taken when system replication is configured across longer distances as this could introduce transaction delay in the system.

Within a single AWS region, Availability Zones are engineered to be isolated from failures in other Availability Zones and provide inexpensive, low latency network connectivity to the other Availability Zones in the same region. Furthermore a single VPC can be configured with separate subnets existing in different Availability Zones. These constructs then provide the ability to configure a SAP HANA environment that spans multiple datacenters to serve as a rapid failover solution for not only unplanned downtime but also planned downtime activities such as system upgrades or other maintenance activities.

Figure 34: Multi-AZ System Replication

The process for setting up the additional systems in the secondary availability zone are as follows:

1. Create additional subnets in the VPC where SAP HANA has been deployed leveraging a second availability zone. One subnet should be private for the SAP HANA Database and or SAP Application servers and the other public if you require High Availability for the NAT and RDP instances.

   Note
   If you have connected your VPC to your own Datacenter through a secure VPN connection over the internet or via AWS Direct Connect you may not have the public subnets.

2. Associate the new subnets with the appropriate route tables in the VPC console.
3. Shutdown the primary SAP HANA Database Instance(s) and create full Amazon Machine Images (AMI's) of each instance.
4. Modify the SAP HANA Master and Worker security groups to include the new subnets to allow traffic to pass between primary and secondary HANA nodes.
5. Launch new SAP HANA systems into the new subnet leveraging the recently created AMI's.
6. Once the new systems are up and running, change the hostnames for each new HANA DB instance and update the /etc/hosts file with the proper IP Address/Hostname entry.
7. Change the hostname for the secondary SAP HANA DB Nodes using the HANA Lifecycle Manager (HLM) or command line as described in the SAP HANA Update and Configuration guide.
8. Verify that the new SAP HANA Nodes are up and running.
9. Follow the steps in section 4.1.2.1 of the SAP HANA Administration Guide to configure System Replication.
10. Test failover procedure as documented in the SAP HANA Administration Guide.

SAP HANA Disaster Recovery using System Replication – Multiple Regions
AWS also provides the ability to deploy SAP HANA environments in a multi-region deployment model. AWS Regions are dispersed and located in separate geographic areas. Currently, the BYOL version of HANA on AWS can be deployed in the following AWS regions:

- US, Northern Virginia (us-east-1)
- US, Oregon (us-west-2)
- Ireland, EU (eu-west-1)
- Tokyo, Japan (ap-northeast-1)

This method uses two separate VPC's configured in separate regions with the same number of primary and secondary SAP HANA systems. System Replication is configured using asynchronous mode. This means the primary system commits a transaction when it has been written to the log file of the primary system and sent to the secondary system through the network. It does not wait for confirmation from the secondary system. Therefore transactions are not held up on the primary system as in synchronous mode. This has potential to improve performance but also introduces the possibility of data loss upon failover if not all changes have been transferred or committed on the secondary prior to takeover.

Figure 35: Multi-AZ System Replication

Note
This setup requires advanced configuration and is often influenced by custom requirements by the customer. Please contact saphana@amazon.com for additional help with this scenario.

Security
The AWS cloud infrastructure has been architected to be one of the most flexible and secure cloud computing environments available today. It provides an extremely scalable, highly reliable platform that enables customers to deploy applications and data quickly and securely.

With the AWS cloud, not only are infrastructure headaches removed, but so are many of the security issues that come with them. AWS's world-class, highly secure data centers utilize state-of-the art electronic surveillance and multi-factor access control systems. Data centers are staffed 24x7 by trained security guards, and access is authorized strictly on a least privileged basis. Environmental systems are designed to minimize the impact of disruptions to operations. And multiple geographic regions and Availability Zones allow you to remain resilient in the face of most failure modes, including natural disasters or system failures.

The AWS virtual infrastructure has been designed to provide optimum availability while ensuring complete customer privacy and segregation. For a complete list of all the security measures built into the core AWS cloud infrastructure, platforms, and services, please read our Overview of Security Processes whitepaper.

When building systems on top of the AWS infrastructure, the security responsibilities are shared between AWS and the customer. AWS secures the datacenters, infrastructure components, on up through the hypervisor layer. It is the

responsibility of the customer and/or a managed service provider employed by the customer to secure the operating system, applications, and restrict access to the deployed instances from a network perspective. More information can be found at http://aws.amazon.com/security/.

Network Security

The default network security setup of this solution follows security best practices of AWS. The provisioning logic creates the solution architecture described in the solution architecture section. The provisioned SAP HANA instances can only be accessed:

1. From the CIDR block specified as “RemoteAccessCIDR” during the provisioning process.
2. By connecting to either the HANA Studio Windows Instance using Remote Desktop Client or the NAT Linux Instance using SSH.
3. Alternatively if a VPN tunnel is provisioned between the customers own data center and AWS, access can be restricted to a known CIDR block.

Identity and Access Management (IAM)

As described previously, this solution leverages an IAM role with least privileged access. It is not necessary or recommended to store SSH keys or secret keys and/or access keys on the provisioned instances.

OS Security

Access to root user on Linux or the Administrator on the Windows RDP instance can only be gained by using the SSH key specified during the deployment process. Amazon Web Services does not store these SSH keys so if you lose your SSH key you can lose access to these instances.

Operating system patches are the responsibility of the customer and should be performed on a periodic basis. The command “zypper up” will update SuSE Linux to the latest patch level available in the SuSE Linux repos on AWS.

Security Groups

A security group acts as a firewall that controls the traffic for one or more instances. When you launch an instance, you associate one or more security groups with the instance. You add rules to each security group that allow traffic to or from its associated instances. You can modify the rules for a security group at any time; the new rules are automatically applied to all instances that are associated with the security group.

The security groups created and assigned to the individual instances created as part of this solution are restricted as much as possible while allowing access to the various functions of SAP HANA. See Appendix B for a complete list of ports and protocols configured as part of this solution.

Additional Security Options

OS Hardening

Some customers would like to lock down the OS configuration further for instance to avoid providing a DB admin with root credentials when logging into an instance.

Please also refer to SAP Notes:
1730999: Configuration changes in HANA appliance
1731000: Unrecommended configuration changes

Disabling HANA Services

HANA Services such as HANA XS are optional and should be deactivated in the case they are not needed. For instructions, see SAP Note 1697613: Remove XS Engine out of SAP HANA Database. In case of service deactivation the TCP ports should also be removed from the SAP HANA AWS Security groups for complete security.

AWS Cloud Trail

AWS Cloud Trail is a recently introduced service, which logs all AWS API calls that are made including the identity of the caller.

Notifications on Access

Notifications on SSH Login to your email address or mobile phone can be setup using AWS SNS or through 3rd party applications.

Summary

Now with AWS you don’t need to wait days, weeks or even months to deploy the infrastructure needed to support your SAP HANA environment. Furthermore, AWS is completely self-service and you only pay for the resources you use. This provides a lot of flexibility for all types of SAP HANA projects and you can quickly convert these to production directly on the AWS platform.

For feedback or questions please contact us at sap-on-aws@amazon.com.

Appendix A: Custom CloudFormation Template Examples

Because the SAP HANA on AWS Infrastructure Subscription is largely based on CloudFormation, the overall solution that is deployed is largely customizable. Keep in mind that the storage configuration and instance type configurations should not be customized if “Production Support” is desired from SAP. Note you must still go through the sign-up process at saphana.com to gain access to the solution before you can use any of the custom CloudFormation templates.

Production templates:
https://s3.amazonaws.com/cf-templates-hana/SAP_HANA_AWS_1.template
https://s3.amazonaws.com/cf-templates-hana/SAP_HANA_AWS_2.template
https://s3.amazonaws.com/cf-templates-hana/SAP_HANA_AWS_3.template
https://s3.amazonaws.com/cf-templates-hana/SAP_HANA_AWS_4.template
https://s3.amazonaws.com/cf-templates-hana/SAP_HANA_AWS_5.template

Same templates with EBS Standard Volumes for non-prod and/or POCs:
https://s3.amazonaws.com/cf-templates-hana/SAP_HANA_AWS_1_EBS_Standard.template
https://s3.amazonaws.com/cf-templates-hana/SAP_HANA_AWS_2_EBS_Standard.template
https://s3.amazonaws.com/cf-templates-hana/SAP_HANA_AWS_3_EBS_Standard.template
https://s3.amazonaws.com/cf-templates-hana/SAP_HANA_AWS_4_EBS_Standard.template
https://s3.amazonaws.com/cf-templates-hana/SAP_HANA_AWS_5_EBS_Standard.template

The following CloudFormation templates provide a significant amount of customization that the default delivery including the ability to specify the following:

- Domain name – The Linux hosts are automatically configured using the domain name specified.
- Hostnames for the Linux hosts where the SAP HANA Master and Worker nodes are deployed.
- VPC-ID of existing VPC where the HANA
- Subnet-ID of existing subnet within the aforementioned VPC where SAP HANA nodes are deployed.
- Private IP Addresses of SAP HANA Virtual machines. These must be valid for the aforementioned Subnet
- Existing IAM Role to be assigned to each virtual machine (i.e. for backup functions)
- Existing Security group to be applied to each instance deployed.
- Placement group (optional)
- No RDP or NAT instance

https://s3.amazonaws.com/cf-templates-hana/SAP_HANA_AWS_1_single_subnet_existing_vpc.template
https://s3.amazonaws.com/cf-templates-hana/SAP_HANA_AWS_1_single_subnet_existing_vpc_EBS_Standard.template
https://s3.amazonaws.com/cf-templates-hana/SAP_HANA_AWS_2_single_subnet_existing_vpc.template

https://s3.amazonaws.com/cf-templates-hana/SAP_HANA_AWS_2_single_subnet_existing_vpc_EBS_Standard.template
Appendix B: Security Group Specifics
The following are the configured inbound and outbound protocols and ports allowed for the various instances deployed as part of this solution.

RDP Security Group

Inbound			
Source	Protocol	Port Range (Service)	Comments
Restricted to CIDR Block specified during the deployment process	TCP	3389 (RDP)	Allow inbound RDP access to Windows instance from your network (over the Internet gateway)
Outbound			
Destination	Protocol	Port Range	Comments
0.0.0.0/0	TCP	1 - 65535	Allow outbound access from RDP server to anywhere

NAT Security Group

Inbound			
Source	Protocol	Port Range (Service)	Comments
Restricted to CIDR Block specified during the deployment process	TCP	22 (SSH)	Allow inbound SSH access to Linux instance from your network (over the Internet gateway)
10.0.0.0/16	TCP	80 (HTTP)	Allow inbound HTTP access only from instances deployed in the VPC
10.0.0.0/16	TCP	443 (HTTPS)	Allow inbound HTTPS access from only instances deployed in the VPC
Outbound			
Destination	Protocol	Port Range	Comments
10.0.1.0/24	TCP	22 (SSH)	Allow SSH access from NAT instance to 10.0.1.0 subnet
0.0.0.0/0	TCP	80 (HTTP)	Allow outbound HTTP access from instances deployed in the VPC to anywhere.
0.0.0.0/0	TCP	443 (HTTPS)	Allow outbound HTTPS access from instances deployed in the VPC to anywhere.

SAP HANA Master & Worker** Security Groups

Inbound			
Source	Protocol	Port Range (Service)	Comments
10.0.1.0/24	TCP	1 - 65535	Communication between instances within private subnet
10.0.1.0/24	TCP	30000 - 30010	Database Internal Communication & SAP Support Access
**10.0.1.0/24	TCP	22 (SSH)	Allow SSH access from other HANA Nodes
10.0.2.0/24	TCP	22 (SSH)	Allow SSH access from NAT instance
10.0.2.0/24	TCP	1128 - 1129	Host Agent Access
10.0.2.0/24	TCP	4300	Access to XSEngine (HTTPS) from 10.0.2.0 subnet
10.0.2.0/24	TCP	8000	Access to XSEngine (HTTP) from 10.0.2.0 subnet
10.0.2.0/24	TCP	8080 (HTTP*)	Software Update Manager (SUM) access (HTTP)
10.0.2.0/24	TCP	8443 (HTTPS*)	Software Update Manager (SUM) access (HTTPS)
10.0.2.0/24	TCP	30015	DB Client Access
10.0.2.0/24	TCP	30017	DB Client Access
10.0.2.0/24	TCP	50013 - 50014	Allow Access for HANA Studio from RDP Instance
Outbound			
0.0.0.0/0	TCP	1 - 65535	Outbound access from HANA Master allowed to anywhere

Amazon Relational Database Service

http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Overview.BackingUpAndRestoringAmazonRDSInstances.html December 09, 2014

In addition to the daily automated backup, Amazon RDS ... since storage engines like MyISAM do not support reliable crash recovery, ... Amazon Web Services, Inc ...

Amazon RDS provides two different methods for backing up and restoring your Amazon DB instances: automated backups and DB snapshots. Automated backups automatically back up your DB instance during a specific, user-definable backup window, and keeps the backups for a limited, user-specified period of time (called the backup retention period); you can later recover your database to any point in time during that retention period. DB snapshots are user-initiated backups that enable you to back up your DB instance to a known state, and restore to that specific state at

any time. Amazon RDS keeps all DB snapshots until you delete them.

A brief I/O freeze, typically lasting a few seconds, occurs during both automated backups and DB snapshot operations on Single-AZ DB instances.

Automated backup is an Amazon RDS feature that automatically creates a backup of your database. Automated backups are enabled by default for a new DB instance.

An automated backup occurs during a daily user-configurable period of time known as the preferred backup window. Backups created during the backup window are retained for a user-configurable number of days (the backup retention period).

The preferred backup window is the user-defined period of time during which your DB instance is backed up. Amazon RDS uses these periodic data backups in conjunction with your transaction logs to enable you to restore your DB instance to any second during your retention period, up to the LatestRestorableTime (typically up to the last five minutes). During the backup window, storage I/O may be suspended while your data is being backed up and you may experience elevated latency. This I/O suspension typically lasts for the duration of the snapshot. This period of I/O suspension is shorter for Multi-AZ DB deployments, since the backup is taken from the standby, but latency can occur during the backup process.

When the backup retention changes to a non-zero value, the first backup occurs immediately. Changing the backup retention period to 0 turns off automatic backups for the DB instance, and deletes all existing automated backups for the instance.

If you don't specify a preferred backup window when you create the DB instance, Amazon RDS assigns a default 30-minute backup window which is selected at random from an 8-hour block of time per region.

The following table lists the time blocks for each region from which the default backups windows are assigned.

Changes to the backup window take effect immediately. The backup window cannot overlap with the weekly maintenance window for the DB instance.

When you delete a DB instance, you can create a final DB snapshot upon deletion; if you do, you can use this DB snapshot to restore the deleted DB instance at a later date. Amazon RDS retains this final user-created DB snapshot along with all other manually created DB snapshots after the DB instance is deleted. All automated backups are deleted and cannot be recovered when you delete a DB instance. Refer to the pricing page for information on backup storage costs.

For more information on working with automated backups, go to Working With Automated Backups.

In addition to the daily automated backup, Amazon RDS archives database change logs. This enables you to recover your database to any point in time during the backup retention period, up to the last five minutes of database usage. Amazon RDS stores multiple copies of your data, but for Single-AZ DB instances these copies are stored in a single availability zone. If for any reason a Single-AZ DB instance becomes unusable, you can use point-in-time recovery to launch a new DB instance with the latest restorable data. For more information on working with point-in-time recovery, go to Restoring a DB Instance to a Specified Time. Multi-AZ deployments store copies of your data in different Availability Zones for greater levels of data durability. For more information on Multi-AZ deployments, see High Availability (Multi-AZ).

Amazon RDS automated backups and DB snapshots are currently supported for all DB engines. For the MySQL DB engine, only the InnoDB storage engine is supported; use of these features with other MySQL storage engines, including MyISAM, may lead to unreliable behavior while restoring from backups. Specifically, since storage engines like MyISAM do not support reliable crash recovery, your tables can be corrupted in the event of a crash. For this reason, we encourage you to use the InnoDB storage engine. If you choose to use MyISAM, you can attempt to manually repair tables that become damaged after a crash by using the REPAIR command ((see: http://dev.mysql.com/doc/refman/5.5/en/repair-table.html). However, as noted in the MySQL documentation, there is a good chance that you will not be able to recover all your data. If you want to take DB snapshots with MyISAM tables, follow these steps: Stop all activity to your MyISAM tables (that is, close all sessions) Lock and flush each of your MyISAM tables Issue a API call, or use the Amazon RDS CLI command . When the snapshot has completed, release the locks and resume activity on the MyISAM tables. These steps force MyISAM to flush data stored in memory to disk thereby ensuring a clean start when you restore from a DB snapshot.

Finally, if you would like to convert existing MyISAM tables to InnoDB tables, you can use alter table command (for example, alter table TABLE_NAME engine=innodb;).

SharePoint 2010: Backup and Recovery Guidelines

http://social.technet.microsoft.com/wiki/contents/articles/5099.sharepoint-2010-backup-and-recovery-guidelines-en-us.aspx December 09, 2014

SharePoint 2010: Backup and Recovery Guidelines. ... Backup using SharePoint Server: Approach two is to use SharePoint Server backup to backup Farm, ...

This blog post describes best practices that you can use to help ensure that backup and recovery operations in Microsoft SharePoint Server 2010 are successful and that the environment is protected against data loss or continuity gaps. The article includes best practices for performance, quality assurance, security, and operational excellence

There are two different approaches for backup and restore

Lets look at each approach in details

Approach one is to use SQL Server backup to back content DBs and Configuration DBs with a complete script for deployment of Web application, Site Collection, Service Provisioning.

a. Create Deployment script of entire Farm. This will include below components of SharePoint

b. Use SQL Server Backup for backing up all Content Database & Configuration Database

c. Backup the entire 14 hive (c:\program files\common files\microsoft shared\web server extensions\14). This is because, frequently you will deploy code to your SharePoint farm, and you will need to restore the supporting physical files for the site to work properly.

d. You need to keep monitoring the size of your content databases. If you start hitting the 50GB mark, think of splitting them up, so the backups are done overnight before users start hitting the database in the morning.

In case of disaster we need to follow below steps to recover/restore content for approach 1

Customizations to SharePoint sites can include the following:

How customizations are deployed, and how changes are made to the Web.config file, have a significant effect on which tools can be used to back up and recover customizations. To provide the greatest opportunity for recovery, we recommend that you deploy customizations by using solution packages and configure the Web.config file by using Central Administration or the SharePoint APIs and object model.

NOTES: site collection does not backup/restore TermSets bound to managed metadata fields, if site collection uses that type of field you have to consider also to export and import Managed Metadata content of the Managed Metadata Service Application. More information on this topic http://social.technet.microsoft.com/wiki/contents/articles/5233.aspx

Approach two is to use SharePoint Server backup to backup Farm, Web application, content DBs and Configuration DBs

a. Create backup script of entire Farm. This will include below components of SharePoint

In case of disaster we need to follow below steps to recover/restore content for approach 1

Backup and restore operations can consume server resources and limit server performance while the operations are running. By following these best practices, you can reduce resource usage and increase the performance of servers and the backup or restore operation

You can follow these best practices to help ensure the quality of the backups of the farm environment and reduce the chances of data loss

Be certain that the system has adequate disk space to accommodate the backup

Routinely test backups and validate their consistency. Run practice recovery operations to validate the contents of the backup and to ensure that you can restore the entire environment. For geographically dispersed environments, prepare for disaster recovery by setting up a remote farm. Then you can restore the environment by using the database attach command to upload a copy of the database to the remote farm and redirect users. Periodically perform a trial data recovery operation to verify that the files are correctly backed up. A trial restoration can expose hardware problems that do not show up with software verifications.

The SharePoint Server 2010 tools do not back up the ULS trace logs. Data in ULS trace logs can be useful for performance analysis, troubleshooting, monitoring compliance with service-level agreements, and legal, regulatory, or business reasons. Therefore, protect this data as part of the routine maintenance. For more information about backing up the ULS logs

To safeguard against loss from a catastrophic event, such as a fire or earthquake, maintain duplicate copies of backups in a separate location from the servers. Doing so can help protect you against the loss of critical data. As a best practice, keep three copies of the backup media, and keep at least one copy offsite in a controlled environment. This should include all backup and recovery materials, documents, database and transaction log backups, and usage and trace log backups

You can use these procedural best practices to help plan and perform backup and restore operations with better documentation, more ease, and greater assurance.

When referring to servers in a different domain, always use fully qualified domain names (FQDN).

When you deploy SharePoint Server 2010, record the accounts that you create, and the computer names, passwords, and setup options that you choose. Keep this information in a safe place.

Prepare for restore testing and disaster recovery by setting up a remote farm. Then you can restore the environment by using the database attach command to upload a copy of the database to the remote farm and redirect users. Similarly, you can set up a standby environment running the same version of software as the production environment so that you can restore the databases and recover documents quickly.

If you want to schedule backups, you can use the Windows Task Scheduler to run them by using a Windows PowerShell script file (*.ps1).

If you are using BLOB storage using the SQL FILESTREAM provider and you back up the content database with that Remote BLOB Store (RBS) defined, both the RBS and the content database will be backed up and restored when you use SharePoint tools or SQL Server tools. We do not recommend that you use RBS with other restore methods.

Please note: Also check out the SharePoint 2010 Best Practice Overview page at http://social.technet.microsoft.com/wiki/contents/articles/8666.sharepoint-2010-best-practices-en.aspx

New Whitepaper: AWS Cloud Security Best Practices

http://blogs.aws.amazon.com/security/post/TxDA6TS0KJK82R/New-Whitepaper-AWS-Cloud-Security-Best-Practices December 09, 2014

... restore the data from backup, ... This approach will help you to customize AWS security controls for your ... By using the various best practices ...

We have just published an updated version of our AWS Security Best Practices whitepaper . We received a ton of feedback from customers on our previous version . You wanted us to provide a holistic and familiar approach to managing the overall information security posture of the organization that’s based on periodic risk assessments when you deploy applications and assets on AWS. Specifically, you asked for:

We decided to structure and model this version of the paper around basic building blocks of designing an Information Security Management System (ISMS) . ISMS is a familiar framework that helps build a collection of information security policies, procedures and processes customized for the organization’s assets. We think that using a widely adopted global security approach that outlines the requirements for information security management systems helps improve your overall security posture. The paper provides a set of best practices on a variety of different security-related topics:

As an example, the table below extends the risk based ISMS approach and maps a recommended protection approach and multiple alternative strategies for data at rest security concerns.

We think this new document structure will make it easier for you to find and understand the information you need. We are constantly launching new AWS services and adding features to our existing services. The number and types of services offered by AWS have increased dramatically. The whitepaper provides a clear description of AWS’ shared responsibility model and discusses the model in depth for different categories of AWS services: Infrastructure Services, Container Services and Abstracted Services. This approach will help you to customize AWS security controls for your organization and help build a more efficient security posture depending on the services you consume.

By using the various best practices highlighted in this whitepaper, you can build a set of security policies and processes for your organization and help you deploy applications and protect data quickly and easily. Like all whitepapers, this whitepaper is a “living document” and we plan to update this whitepaper as we introduce new features and services. We look forward for your feedback.

Creating a Backup and Recovery Plan

http://technet.microsoft.com/en-us/library/cc739288(v=WS.10).aspx December 09, 2014

A rollback strategy The rollback strategy defines how you plan to use backup and recovery procedures to return your pilot or production environment to the ...

The backup and recovery plan establishes guidelines and procedures to prevent problems that might cause data loss or interruptions to your organization’s operations, and to allow recovery as quickly as possible if such events do occur.

Consider planning downtime or outages for the pilot in order to test rollback procedures, and, if applicable, disaster recovery and business continuity plans.

For more information about creating backup plans, see the Storage Technologies Collection of the Windows Server 2003 Technical Reference (or see the Storage Technologies Collection on the Web at http://www.microsoft.com/reskit).

In addition, the backup plan should identify who is responsible for performing backups, and should include the schedule for all periodic backups and periodic testing of backups, as well as instructions for labeling and storing all backup files.

The importance of a backup plan cannot be overstated. When you begin rolling out Windows Server 2003 in the business environment, problems might arise that even the most thorough testing could not reveal. By making regular and reliable backups, you ensure that the team can restore the system to its original state if your pilot rollout process changes or fails.

The recovery plan describes the recovery and rollback process, which allows you to return your production system to whatever earlier state you require. Depending on the severity of a problem encountered in the pilot, you might need to return your production system to a baseline configuration or just roll it back to the state it was in at a particular point in time.

Include the following elements in the recovery plan:

A list of scenarios Analyze all of the systems involved in the rollout and identify the situations, or scenarios, under which problems are likely to occur. Determine which systems might be affected and the functional dependencies among them so you have a clear understanding of the larger impact that a single failure might have. Use these scenarios to create strategies that identify when and how to run backups and the types of recovery for which you need to plan.

A definition of acceptable downtime Define how much downtime your organization can accommodate. If your organization cannot afford for systems to go down during normal business hours, you might plan to roll out the pilot, or parts of it, at night or over a weekend. If systems must be operational at all times, you might plan to deploy servers and desktops on new computers and then quickly replace the old ones, instead of upgrading computers.

A list of critical systems and processes In the event that a failure does occur, you need to know which systems are the most critical and must be brought back online first. If resources such as bandwidth are limited, you need to know which systems have the highest priority and which should not take up network traffic. When evaluating how critical a system or process is, consider factors such as its effect on human health and safety, the legal liability it exposes, the risk to corporate confidentiality, and the cost of replacement.

A recovery strategy Your recovery strategy should define how you will recover data or systems in each of the scenarios you define in the recovery plan. This might include restoring data from backup tapes, switching over to redundant systems, rolling back to previous configurations, or other strategies. Include an additional procedure for recovering from severe data corruption in your directory service if that becomes necessary. By having your recovery strategy in place, you can quickly restore your production environment to the required state so that work can continue with minimal interruption.

A rollback strategy The rollback strategy defines how you plan to use backup and recovery procedures to return your pilot or production environment to the state it was in before changes were made. Specify the criteria that a problem should meet to warrant rolling the environment back to its previous state. For example, you might establish a system for classifying the severity of problems and describe which type of response is warranted by certain levels of severity. Also decide whether you need to have different rollback strategies for different types of problems. For example, you might develop one procedure for backing out the entire pilot if the problem is pervasive and another procedure for backing out specific components if the problem is isolated.

The roles and responsibilities for team members Make sure that every task in the plan is assigned to an appropriate team member, and that that person has the information needed to successfully perform required tasks. Consider including training in the plan.

Have the backup and recovery plan reviewed by the project team and by those responsible for potentially affected systems. After the plan has been approved, test it to ensure that the processes you put in place work as expected.

Backup and Recovery

http://www.thexlab.com/faqs/backuprecovery.html December 09, 2014

... most backup and recovery applications use the terms source and ... At The X Lab we believe in the "belt and suspenders" approach to backup and recovery, ...

Our recommended, comprehensive Backup and Recovery solution consists of: One or both of the following backup and recovery applications: Time Machine® (included with Mac® OS X 10.5 and later). External hard disk drives as the primary backup media. If your Mac has a FireWire® port, we recommend using FireWire over USB 2.0 drives. In creating backups, most backup and recovery applications use the terms source and destination to specify the data to be backed up and where the backup is saved, respectively. For example, if you back up the Macintosh HD volume in your Mac to a partition on a FireWire drive, Macintosh HD is the source and the partition on the FireWire drive is the destination. In recovery, these terms are reversed: the source is the backup itself and the destination is where the saved data will be restored. For example, if Macintosh HD will be restored from a backup saved on a partition on a FireWire drive, the backup is the source and Macintosh HD is the destination. SuperDuper creates Copies. A Copy is an exact duplicate of a disk, partition, or collection of files. Think of a Copy as a genetic clone of the source. Other third-party backup and recovery applications may refer to Copies as duplicates or clones. Copies are primarily used to duplicate a disk, partition, or collection of files to another disk or partition of a size greater than or equal to the amount of data being backed up. Some backup and recovery applications support saving Copies to a folder or a disk image. As Copies are saved in the same structure as the original data, they have an additional property: a Copy of a bootable volume, such as the Mac OS X startup disk, that is saved to a dedicated partition can be used as a startup disk, also known as a bootable duplicate. Since Copies are exact duplicates of the source at the time of the backup, they do not preserve history, such as older versions of files or files that have been deleted from the source. The only files that can be recovered from a Copy are those extant when the backup was created. Updating a Copy generally synchronizes the Copy with the source: new or changed files are copied to the backup and files deleted from the source since the last backup are deleted from the Copy. Time Machine create Snapshots. Snapshots are also an exact duplicate of the data backed up, but with a twist: Snapshots preserve history. The first backup of the source copies everything to the destination. Subsequent backups copy only new or changed files since the last backup. Since Snapshots preserve older versions of files, you can recover earlier versions of files that have been backed up. Depending upon the application, the destination for Snapshots can be a disk, partition, folder, or disk image. Since Snapshots preserve history, the disk serving as the destination will eventually become full. SuperDuper is renowned for its speed, ease of use, and low cost. It excels at duplicating hard drive volumes quickly and painlessly. Recovery of an entire volume is likewise quick and painless. The short learning curve, intuitive interface, low cost, good documentation, excellent support, and proven reliability make SuperDuper a super choice. A free trial is available; purchasing a license enables its incremental backup functions. Why not use Time Machine exclusively? Time Machine is exceptionally innovative and easy to use. Nevertheless, it has a number of shortcomings, especially the inability to create a bootable duplicate of your Mac OS X startup disk. The conservative approach is to use Time Machine as an adjunct to, rather than a replacement for, a third-party backup and recovery application. If you are running Mac OS X 10.5 or later and do not have a comprehensive backup and recovery solution, start with Time Machine, then add SuperDuper as your budget permits. What does The X Lab use? At The X Lab we believe in the "belt and suspenders" approach to backup and recovery, an idea captured by the acronym LOCKSS: Lots Of Copies Keeps Stuff Safe. We use both SuperDuper and Time Machine. We use multiple tools to reduce the possibility that a bug in any one tool could result in data loss. SuperDuper is used to perform multiple,

scheduled, nightly backups. Depending on the Mac in question, we make multiple backups of each drive to multiple, separate FireWire drives: this reduces the possibility of data loss from the failure of any given backup device. Time Machine is used as an adjunct to manually create snapshot backups during the day. The backup and recovery strategy at The X Lab is discussed in detail in the "Backup and Recovery" chapter of our Troubleshooting Mac OS X e-books. Designing and implementing a backup and recovery strategy involves five steps: What backup application will be used? What volume of data will be backed up, both now and in the future? What backup devices will be employed? What provisions will be made for off-site data protection? What backup strategy will be used? Specifically, how will backup devices be employed in combination with the features of the backup and recovery application, such as types of backups, scheduling, and automation to minimize the risk of data loss? These steps are discussed in detail in the "Backup and Recovery" chapter of our Troubleshooting Mac OS X e-books. Special considerations in implementing Time Machine are discussed in the "Time Machine" chapter of our Troubleshooting Mac OS X e-books. Unfortunately, too many folks fail to realize the value of a comprehensive backup and recovery solution until after disaster has struck; by then, it's too late. We believe that using a personal computer without a comprehensive backup and recovery solution is like driving without auto insurance. For help implementing a backup and recovery solution like that employed by The X Lab, see the "Backup and Recovery" chapter of our book Troubleshooting Mac OS X.

Administration at a glance

http://scn.sap.com/docs/DOC-59809 December 09, 2014

Perform data backup and recovery by using simple point and click in the ... guide provides an overview of how to backup SAP systems running on Amazon Web Services.

HANA status, memory, CPU available on status page, along with the disk usage of various important disks.

Perform Following passwords can be easily changed using HANA One Management console. (The password reset is protected using AWS credentials.)

HANA database can be started/stopped using Console.

Perform data backup and recovery by using simple point and click in the HANA One Management Console.

Refer to the HANA One Admin guide for the backup and recovery steps using Studio.

After you have created a backup onto an EBS volume, copy the backup data to Amazon S3. This protect the data against EBS volume failure.

To copy to or retrieve data from Amazon S3 there are assorted command line interfaces that allow you to incorporate into a script. S3cmd is an open source project available under GNU Public License v2 (GPLv2) and is free for both commercial and private use.

Instead of copying and retrieving single files to Amazon S3, you can also create a snapshot of the complete EBS Volume. The snapshot is automatically stored in Amazon S3 and can subsequently be used to create new EBS Volumes that contain the same data that is stored in the snapshot.

The Backup and Recovery of SAP systems on Amazon Web Services guide provides an overview of how to backup SAP systems running on Amazon Web Services. This guide focuses on the essential differences in backing up SAP systems on AWS compared to traditional infrastructure.

Migrate older versions of HANA One to the latest HANA One Rev 52.1 (HANA Upgradable) Migrate HANA contents from HANA Dev Edition to HANA One Rev 52.1 (HANA Upgradable) SAP released HANA One Rev 38, HANA One Rev 48 and HANA Rev 52 before releasing HANA One Rev 52.1. HANA One Rev 52.1 includes Addon Manager, a self-service upgrading tool [blog]. Customers using HANA One Rev 52.1, can always upgrade to the latest HANA version as a self-service option. In addition, when the license key is about to expire, customers can extend a 1 year license key or install a valid license key, if necessary. However, customers using HANA One before HANA One Re 52.1, cannot upgrade HANA or extend the license key. This guide includes the procedure to migrate older releases of HANA One to the latest HANA One, to take advantage of new Addon feature in HANA One Rev 52.1.

With the availability of HANA License Key 1.0 in HANA One Rev 52.1 from September 23 2013, customers can use database backup and recovery (system copy) to migrate their HANA contents from HANA One Rev 48 or HANA One Rev 52 to HANA One Rev 52.1. However, this feature is not applicable to HANA One Rev 38 (1 year license expires in early October 2013). The only option to migrate HANA contents from HANA One Rev 38 to HANA One Rev 52.1 is using HANA export and import utility. Although, these instructions cover the HANA One migration, the same procedure is applicable if customers want to migrate HANA contents to HANA Dev Edition to HANA One.

In HANA One at AWS (an example of a public cloud), a database copy is possible, as it uses only file-based backups. To simplify the migration in the cloud, only the data backup files are considered here. This restores the content exactly as of the point in time at which the data backup was created. Before migrating, the following are required: A database backup of the source system is available. The version of the target system is the same or higher than the source system. Except HANA One Rev 38, all HANA One versions met this criterion. The target system has sufficient disk space and memory. Once a new HANA One Rev 52.1 instance is launched, it meets this criterion. The target system configuration is usable for the recovery of the source system. The type and number of services (for example, indexserver) must be identical in both systems. HANA One is not expected to change any internal configuration and therefore, meets this criterion as well. A license key file is available for the target database. The license key of the source system will not work in the target system. HANA One Rev 52.1 includes HANA One License Version 1.0 since September 23, 2013. Create a backup of your data using HANA Studio or the HANA One management console (Rev 52 or later). Launch a new HANA One Rev 52.1 instance as your new target system Transfer backup files from your source system to your target system. Use HANA Studio (Rev 52) to recover the data only. After recovery is successful, use the Addon feature in HANA One Rev 52.1 to install HANA Rev 52.1 License Key 1.0. Verify the license at your Rev 52.1 instance and content. Note: For a migration example, see the blog post SAP HANA One Migration Via Data Backup and Recovery.

From the HANA One Management Console, select the Addons tab. Select SAP HANA Version you want Download & Install. Once you install a new HANA version, you can not uninstall. Select Install and follow the steps on the screen to continue the install. To check the status of the install or uninstall, select the active installation package under Installing/Uninstalling. Click "Input" button to enter the user input parameters.

For example, enter the SYSTEM DB and hdbadm OS passwords and Submit. To optimally utilize the available memory, SAP recommends to stop and restart the EC2 instance from EC2 management console. You may need to associate Elastic IP after a restart of the EC2 instance.

SAP HANA One Rev 70.0 comes with Lumira Samplers 2.1 (The available packages list is dynamically populated). Follow similar steps to install any packages as above. These packages can be uninstalled.

To postpone any installation, select Ignore. The ignored installations display in the Ignored tab.

Failed installations display in the Failed tab. Select the Failed package for details.

From the Installed tab, select any package, select Uninstall and follow the instructions on the screen.

Cloud-Stored Offsite Database Backups - Oracle | Hardware ...

http://www.oracle.com/technetwork/database/features/availability/twp-oracledbcloudbackup-130129.pdf December 09, 2014

production database. Amazon Web Services addresses this ... Oracle Secure Backup Database Web ... Please refer to Oracle Backup and Recovery Guide to learn more ...

An Oracle White Paper
May 2010

Cloud-Stored Offsite Database Backups

1

Introduction

Cloud Computing allows users to tap into a virtually unlimited pool of computing and storage resources over the Internet (the Cloud). Unlike traditional IT, Cloud users typically have little insight or control over the underlying infrastructure, and they must interact with the computing and storage resources via an Application Programming Interface (API) provided by the Cloud vendors. In exchange for those constraints, Cloud users benefit from utility-like costs, scalability, and reliability, as well as the ability to self-provision resources dynamically and pay only for what they use

The ability to back up Oracle Database in the Cloud is a key part of Oracle's Cloud offering. It allows customers to use Storage Clouds, such as Amazon's Simple Storage Service (S3), as their next-generation offsite backup storage destination. Compared to traditional tape-based offsite storage, Cloud backups are more accessible, faster to restore under most circumstances, and more reliable. Cloud backups are also the right protection for databases running within the compute Cloud.

2

Why Backup Storage in the Cloud

Good Disaster Recovery (DR) practice requires keeping usable business-critical backups offsite. Organizations have traditionally implemented this by writing backups to tape and shipping the tapes to be stored offsite. This is costly and operationally complex, requiring hardware, personnel, and sound procedures to ensure that the offsite backups are up-to-date, secure, and able to be recalled and used in the face of disaster. While shipping and secure storage are often outsourced, the IT organization of the enterprise retains the burden of ensuring the integrity of the backups and procedures.

The pricing and operational characteristics of Cloud Storage make it a very compelling alternative to shipping tapes offsite. Cloud storage offers pay-as-you-go, elastic self-provisioning, with low prices 1 per unit storage per unit time, making costs easy to predict, control, and map to the workloads of an organization's IT assets. Good Cloud infrastructure offers storage redundancy, security, availability and scalability with geographic distribution that enables it to absorb a broad range of adverse events with minimal or no loss of availability. These characteristics make it an excellent alternative to writing, shipping and storing tapes in a secure location. Last, but not least, backups are created and updated over the network, with minimal or no operator involvement — drastically simplifying operational procedures.

Amazon Web Services (AWS) is the first Cloud vendor that Oracle has partnered with to enable database backup in the Cloud. Simple Storage Service (S3) is the main storage offering of AWS. S3's simple web-services interface enables applications to store and retrieve any amount of data from anywhere on the Internet. S3 is a highly scalable, reliable, fast, inexpensive data storage infrastructure, and thousands of enterprises small and large rely on it for their production storage needs, from "cold" inexpensive storage to serving rich multimedia in real time to customers worldwide.

Sending backups over the Internet to be stored in the Cloud benefits from the elasticity in capacity and operational expenses typical of Cloud services. It can also simplify your own infrastructure as you no longer need to provide and manage storage (e.g., tapes that need to be rotated, shipped away, etc.).

An important objection to over-the-network Cloud backup is that limited network bandwidths in the public Internet preclude the fast transfer of large data amounts like those of a full backup of a large production database. Amazon Web Services addresses this problem by offering S3 data import and export services, which enable bulk movement of data into and out of S3 by shipping portable disks. For example, , after a disaster, S3 can express mail a portable hard drive containing all the backup data for a given database. This makes the cloud storage offering comparable to storing tapes offsite, especially when it is done as part of a complete backup strategy that includes keeping backups onsite as well as offsite.

1 Storage Cloud benefits from the falling prices of the commodity disks underlying the infrastructure, and the economies of scale of the Cloud operator.

3

Oracle Secure Backup Cloud Module

The Oracle Secure Backup (OSB) Cloud Module enables an Oracle Database to send its backups to Amazon S3. It is compatible with Oracle Database versions 9i Release 2 and above, and it requires a network connection to the Internet, and provisioning the means of payment to Amazon Web Services.2 The Oracle Secure Backup Cloud module can also be used when the database is running within the Amazon Elastic Compute Cloud (EC2), in which case it benefits from the higher internal network bandwidth and no transfer costs into and out of S3.

The Oracle Secure Backup Cloud module is implemented using the Oracle Recovery Manager (RMAN) SBT interface. The SBT interface allows external backup libraries to be seamlessly integrated with RMAN. Consequently, database administrators can continue to use their existing backup tools — Enterprise Manager, RMAN and other scripts, etc. — to perform Cloud backups.

OSB Cloud module is available for Linux 32 and 64, SPARC 64, and Windows 32.

Complete Data Security with Built-in Encryption

Oracle Secure Backup leverages RMAN's ability to encrypt backups to ensure data security. Data security and privacy is particularly important in shared, publicly accessible environments such as the Storage Cloud. While most Storage Cloud vendors provide robust security to ensure that only authorized users can access data, Oracle's encryption of backup data before it leaves your database further mitigates risk of theft or unauthorized access because the backup data remains encrypted both in-transit and at rest in the Cloud.

Compressed Backups for Better Performance
Integration with the Oracle Database engine enables Oracle Secure Backup to identify and skip unused space (blocks) within the database. Users also benefit from RMAN's rich compression capabilities. When transmitting backups over slower networks, such as the public Internet, any reduction in backup size is directly realized as an increase in backup performance.

Database Version Support

The Oracle Secure Backup Cloud Module may be used to back up the following supported versions of Oracle Database: Oracle Database 9i Release 2 or higher, including Oracle Database 11g.

2
The Cloud Backup Module is a part of the Oracle Secure Backup product family, and licensed on a per-RMAN channel basis. Oracle Secure Backup is Oracle's next-generation tape backup management solution and it now provides customers the flexibility to back up data to either tape or the Cloud.

4
Figure 1. Oracle Database backup in the Cloud

Benefits of Oracle Cloud Backup
Oracle's Cloud backup functionality provides advantages over traditional tape-based offsite backups:

- Continuous Accessibility: Backups stored in the Cloud are always accessible – much in the same way local disk backups are. As such, there is no need to call anyone and no need to ship or load tapes before a restore can be performed. Administrators can initiate restore operations using their standard tools (Enterprise Manager, scripts, etc.) just as if the offsite backup was stored locally. This can help make restores faster and reduce down time from days to hours/minutes in many cases. For large databases where shipping a portable disk from the Cloud is required, a restore takes no longer than it would take to have a tape recalled from an offsite location.
- High Reliability: Storage Clouds are disk based and thus inherently more reliable than tapes. Additionally, Cloud vendors typically keep multiple redundant copies of data for availability and scalability purposes. (See AWS's S3 Service Level Agreement and FAQs.)
- Unlimited Scaling and No Up-front Capital Expense: The Cloud provides virtually unlimited capacity with no up-front capital expenditure. Consequently, users need not worry about provisioning adequate tapes or local storage to hold the required backup data. The Cloud scales seamlessly and users pay only for what they use, when they use it.
- Reduced Tape Backup and Offsite Storage Cost: Since Cloud backup reduces or eliminates the need for tapes, this can lead to significant savings in tape backup software licensing/support and offsite tape storage costs.
- Easy Provisioning of Test and Dev Environments: As Cloud Backups are accessible from anywhere via the Internet, they can be used to quickly clone databases to create custom test, development, or QA environments. For instance, Cloud Backups stored in Amazon S3 can be cloned to machines running in Amazon EC2 by running a simple script that is included in the Oracle-provided Amazon Machine Images (AMIs). An AMI is a virtual machine image that allows quick provisioning of a pre-installed and pre-configured Oracle database environment on Amazon EC2.

5
Getting Started with Cloud Backup
This section explains how to provide the means of payment to Amazon to use their storage cloud, and how to obtain and configure the OSB Cloud module with your Oracle Database.

Sign up for Amazon S3

The first step in getting started with the Oracle Secure Backup Cloud module is to sign up for Amazon S3. This can be done by visiting the Amazon S3 website (http://aws.amazon.com/s3). Upon successful registration, users will be provided a pair of access identifiers called the Access Key ID and the Secret Access Key.

Register for an Oracle.com or Oracle Technology Network (OTN) Account

An Oracle.com or OTN account is required to install the Oracle Secure Backup Cloud module. New accounts may be created by visiting the OTN website (http://otn.oracle.com).

Install the Oracle Secure Backup Cloud Module

The next step is to download the Oracle Secure Backup Cloud module install tool from OTN's Cloud webpage, and run it to install and configure Cloud backups. Oracle Amazon Machine Images (AMI) on AWS's EC2 already include this install tool. Therefore, if the database being backed up is running on Amazon EC2, there is no need to download the install tool – it can be found in the /home/oracle/scripts/osbws directory.
The install tool can be invoked as follows (you must supply your OTN and AWS credentials):

```
$java -jar osbws_install.jar -AWSID <AWS ID> -AWSKey <AWS Secret Key>
-otnUser <OTN User ID> -otnPass <OTN Password> -walletDir <Wallet
Directory> -configFile <Cloud Backup Configuration File Name> -libDir
<Location to store Cloud Backup Module/Library> -proxyHost www-
proxy.yourcompany.com –proxyPort <your proxy port>

Oracle Secure Backup Database Web-Service Install Tool
OTN userid is valid.
AWS credentials are valid.
Creating new registration for this S3 user.
Created new log bucket.
Registration ID: 0f0a8aac-dad0-6254-7d70-be4ac4f112c4
```

```
 S3 Logging Bucket: oracle-log-jane-doe-1
 Create credential oracle.security.client.connect_string1
 OSB web-services wallet created in directory /orclhome/dbs/osbws_wallet.
 OSB web-services initialization file /orclhome/dbs/osbwst1.ora created.
 Downloading OSB Web Services Software Library.
 Downloaded 13165919 bytes in 204 seconds.
 Transfer rate was 64538 bytes/second.
```



```
 Download complete.
 Extracted file /orclhome/lib/libosbws11.so
```

Example 1: Running the Cloud Backup Install Tool

Example 1 above shows how the tool automatically performs all the required steps to install and configure the Cloud backup module – downloading the software, creating a wallet containing the user's AWS identifiers, and creating the Cloud backup configuration file. More details on how to run the install tool and the description of all of its arguments can be found in the install tool readme document.

Configure Recovery Manager (RMAN) Settings

This step stores the configuration information for the Cloud Backup module in the RMAN repository so that it does not need to be specified each time a backup is invoked.

```
RMAN> configure channel device type sbt parms
'SBT_LIBRARY=/orclhome/lib/libosbws11.so
ENV=(OSB_WS_PFILE=/orclhome/dbs/osbwst1.ora)';

using target database control file instead of recovery catalog
new RMAN configuration parameters:
CONFIGURE CHANNEL DEVICE TYPE 'SBT_TAPE' PARMS
'SBT_LIBRARY=/orclhome/lib/libosbws11.so
ENV=(OSB_WS_PFILE=/orclhome/dbs/osbwst1.ora)';
new RMAN configuration parameters are successfully stored
```

Example 2: Configuring RMAN

Once the RMAN configuration is completed, Cloud Backups can be performed using the same RMAN commands you usually use. This step is optional but strongly recommended.

Cataloguing and Using Cloud Backups

All Cloud backup operations will be catalogued by RMAN in the same manner as local disk or tape backups are, ensuring a seamless restore/recovery process. When a restore/recover operation is initiated, RMAN and Oracle Secure Backup Cloud module will automatically restore the required data from the Cloud - without requiring any special user intervention.

Cloud Backup Best Practices

Securing Data in the Cloud

Oracle strongly recommends encrypting your Cloud backups. Encrypting backups ensures that your data remains secure and protected against unauthorized access. Please refer to Oracle Backup and Recovery Guide to learn more about the RMAN commands that are used to configure backup encryption. Encryption can also be enabled while scheduling backups in Enterprise Manager.

Optimizing Cloud Backup Performance

As Cloud Backups are sent over the public Internet, performance is dependent on Internet network throughput – typically less than 1 MB/Sec per connection. Additionally, Cloud vendors may throttle sessions to prevent individual users from consuming disproportionate amounts of resources. According to internal tests conducted at Oracle, Amazon S3 limits an individual session's read/write throughput to around 2-3 MB/Sec. However by using the right combination of parallelism and compression, backup speeds of up to 40-50 MB/Sec were attained; test results are summarized in Table 1, below. Some observations follow from these tests:

- Cloud Backups of your on-premise (off-cloud) databases are slower than for databases running on EC2. This is due to public Internet network bandwidth constraints.
- Compression helps overcome the network bandwidth limitations. For a database at Oracle HQ, the use of compression resulted in a 4X gain in backup speed.
- Using parallel streams (RMAN channels) also speeds up Cloud backups – particularly for on-premise databases. As can be seen in Table 1, peak performance for a database at Oracle HQ was achieved with 64 channels.

Oracle recommends the following to optimize the performance of Cloud Backups:

- Use multiple RMAN channels for higher parallelism resulting in full utilization of the network.
- Use multi-section backups. Oracle Database 11g allows multiple channels to back up a single file in parallel, increasing parallelism beyond the number of datafiles to be backed up. For example, the RMAN command to specify backup section size 1 GB is:

```
BACKUP DEVICE TYPE SBT DATABASE SECTION SIZE 1g;
```

- Use the Oracle Database 11g Advanced Compression. Oracle Database 11g Compression is significantly faster and more efficient (in terms of CPU overhead) than pre-11g compression.
- Consider making full database backups once a week and performing incremental backups during the weekdays. This will results in faster backups and may help save significant amount of network bandwidth. Use the RMAN Block Change Tracking feature to optimize the performance of your daily incremental backups.

8

Test Environment	Uncompressed Backup Speed (Network Throughput)	Compressed Backup Speed	Full DB Backup Time (250 GB)	Incremental Backup Time (10% delta)
DB at Oracle HQ (8 x 2 GHz CPU, 16 GB RAM)	10 MBPS (64 RMAN Channels)	40 MBPS (64 RMAN Channels)	2-6 Hours	30 Minutes – 1 Hour
DB within Amazon Cloud (Extra Large EC2 Instance)	35 MBPS (16 RMAN Channels)	50 MBPS + Constrained by CPU (32 RMAN Channels)	2 Hours	< 20 Minutes

Table 1: Cloud Backup Performance

Conclusion

The Oracle Database Cloud Module allows customers to use Amazon's Simple Storage Service (S3) as their offsite backup storage destination. Compared to traditional tape-based offsite storage, Cloud backups are more accessible, faster to restore under most circumstances, and more reliable, while eliminating the overheads associated with maintaining off-site backup operations.
Cloud backups are also the optimal protection for databases running within the compute Cloud.

9

Cloud-Stored Offsite Database Backups
May 2010
Author: Cris Pedregal
Contributing Authors: Bill Hodak, Muthu Olagappan

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

AWS console breach leads to demise of service with “proven” backup plan

http://arstechnica.com/security/2014/06/aws-console-breach-leads-to-demise-of-service-with-proven-backup-plan/ December 09, 2014

Did AWS offer any sort of recovery? ... These people apparently offered a backup and recovery service with a single point of failure, ...

A code-hosting service that boasted having a full recovery plan has abruptly closed after someone gained unauthorized access to its Amazon Web Service account and deleted most of the customer data there.

Wednesday's demise of Code Spaces is a cautionary tale, not just for services in the business of storing sensitive data, but also for end users who entrust their most valuable assets to such services. Within the span of 12 hours, the service experienced the permanent destruction of most Apache Subversion repositories and Elastic Block Store volumes and all of the service's virtual machines. With no way to restore the data, Code Spaces officials said they were winding down the operation and helping customers migrate any remaining data to other services.

"Code Spaces will not be able to operate beyond this point," a note left on the front page of codespaces.com said. "The cost of resolving this issue to date and the expected cost of refunding customers who have been left without the service they paid for will put Code Spaces in a[n] irreversible position both financially and in terms of on going credibility. As such at this point in time we have no alternative but to cease trading and concentrate on supporting our affected customers in exporting any remaining data they have left with us."

A copy of the Code Spaces website still cached on Bing shows that the website promoted its ability to protect customer data from catastrophic events as a key benefit.

"Backing up data is one thing, but it is meaningless without a recovery plan, not only that [but also] a recovery plan—and one that is well-practiced and proven to work time and time again," the cache stated. "Code Spaces has a full recovery plan that has been proven to work and is, in fact, practiced."

Wednesday's advisory said the unauthorized access to the Amazon-hosted Code Spaces data came on the heels of a distributed denial-of-service attack, presumably in an attempt to extort money from the service. Code Spaces officials wrote:

Dear Customers,

On Tuesday the 17th of June 2014 we received a well orchestrated DDOS against our servers, this happens quite often and we normally overcome them in a way that is transparent to the Code Spaces community. On this occasion however the DDOS was just the start. An unauthorised person who at this point is still unknown (All we can say is that we have no reason to think [it's] anyone who is or was employed with Code Spaces) had gained access to our Amazon EC2 control panel and had left a number of messages for us to contact them using a hotmail address Reaching out to the address started a chain of events that revolved [around] the person trying to extort a large fee in order to resolve the DDOS. Upon realisation that somebody had access to our control panel we started to investigate how access had been gained and what access that person had to the data in our systems, it became clear that so far no machine access had been achieved due to the intruder not having our Private Keys. At this point we took action to take control back of our panel by changing passwords, however the intruder had prepared for this and had already created a number of backup logins to the panel and upon seeing us make the attempted recovery of the account he proceeded to randomly delete artifacts from the panel. We finally managed to get our panel access back but not before he had removed all EBS snapshots, S3 buckets, all AMI's, some EBS instances and several machine instances. In summary, most of our data, backups, machine configurations and offsite backups were either partially or completely deleted. This took place over a 12 hour period which I have condensed into this very brief explanation, which I will elaborate on more once we have managed our customers' needs.

The advisory didn't say exactly how the hacker gained entry to the Code Space panel hosted on AWS. People who host their services on Amazon should avail themselves of the full spectrum of multifactor protections.

Running SQL Server Databases in the Amazon Cloud

http://www.mssqltips.com/sqlservertip/3290/running-sql-server-databases-in-the-amazon-cloud--rds-backup-and-restore-part-3/ December 09, 2014

... we will talk about how RDS SQL Server ... You can either configure the time of the backup manually or let AWS ... RDS will set the recovery models ...

In the last installment of this series, we discussed some challenges associated with running a SQL Server instance in Amazon Web Service (AWS) RDS. In this tip, we will talk about how RDS SQL Server databases can be backed up and restored.

Amazon Web Service (AWS) Relational Database Service (RDS) offers two different ways to backup database instances. The first one is when you enable automated backup and the second one is when you take manual snapshots. As with any other AWS service, these methods are accessible via programmatic interfaces like command line tools and various programming language SDKs or via the AWS Management Console. In this tip we will discuss how to back up and restore using the console.

With RDS, DBAs don't have to worry about database backups and backup jobs. This can be set up only once during instance creation time or later and RDS will make sure all databases are backed up on a regular schedule. As we saw in a previous tip, DBAs can't take manual backups of individual databases; nor can they exclude databases from a backup plan. However, this also frees up the DBA, because:

With RDS automated backups, only two options need to be specified. In the following image, I am configuring backup for a new instance:

The Backup Retention Period decides how long you want RDS to keep the automated backups. At the time of this writing (July 2014), this can go back to thirty five days.

You can either configure the time of the backup manually or let AWS decide it for you. The timing in RDS is specified in Coordinated Universal Time (UTC), so you may need to do a bit of calculation to convert the timescale of your choice to UTC. In the image above, my database instance is in ap-southeast-2 (Sydney) region and I prefer it to be backed up at 2:00 AM AEST (Australian Eastern Standard Time). In UTC, that's 4:00 PM the previous day (16:00 hours). With manual configuration, you can also specify the backup duration. This can be anywhere between thirty minutes to three hours.

If you don't specify the backup window, RDS will choose an arbitrary thirty-minute window from an eight-hour block assigned to the RDS region. Every AWS region has a default eight-hour block of time for RDS and RDS can initiate backup any time during that block. The following list shows these blocks of time for different regions.

If you don't want to back up your databases, choose a retention period of zero days. RDS will warn you if you choose so, but will allow you to go ahead and create the instance.

With automated backups, RDS will perform a full backup of the databases once every day during the backup window. It will also perform transaction log backups of all databases. This allows the instance to be restored to any point in time. So, what happens if you change a database's recovery model to simple? Well, RDS will revert it back to full within five minutes of making this change. If you disabled automated backups, RDS will set the recovery models of all databases to simple. Again, if you choose to change them back to full, the changes will be reverted within five minutes.

If you want to stop the backup of an existing instance, you can modify the instance's property and choose a retention period of 0 days. When that happens, RDS will warn you, but let you save your configuration. RDS will then delete every automated backup of the instance. The same thing happens when you delete the instance: RDS will get rid of all previous backups. You have to modify an instance's property to enable backups again.

Restoring a backup allows you to restore the whole instance to a point in time, typically within the last five minutes of current time. The following screenshots show how I am restoring an RDS instance.

I have used a custom restore time here. I could specify the time up to which RDS should apply its database transaction logs.

One thing to notice here is that the restore actually creates a new database instance. You can't overwrite an existing RDS instance with a restore. Also, most options can be changed here: you can change the VPC where the instance would be created, modify its size and so on, but you can't modify the master user name or password.

Once the instance has been restored, it's accessible from the RDS console. Note that it has a new endpoint and it also inherits the source server's backup properties.

RDS snapshots are what the name implies: they are "snapshots" or point-in-time replicas of your database instance. You can create snapshots of RDS instances as many times as you want, whenever you want, and they will be saved persistently in durable storage in AWS. You can restore from a snapshot any time and RDS will create a fully functional instance from it. However, you need to be aware of a few things:

In the image below, I am taking a snapshot of the database instance:

I can provide a name for the snapshot before it's generated:

Once the snapshot has been created, it's found under the "Snapshots" section of the navigation menu:

The "Copy Snapshot" option allows you to copy the snapshot to a different region. That's useful if you are building a multi-region application in AWS and you want to start from an existing "baseline". In the following image, I am copying the RDS snapshot located in the Oregon Region (us-west-2) to Singapore (ap-southeast-1)

Think before you copy snapshots though. Copying the snapshot of a 1 terabyte database instance over the Internet can be an expensive affair both in terms of time and money. You can think about using Amazon CloudFormation for such operations. We will talk about CloudFormation in a later tip.

Restoring from an RDS snapshot is fairly simple as well:

Although you can choose to modify a number of options here, the main thing to watch is the DB Instance Identifier field. You can't restore on top of an existing instance. If you try to do so, you will get an error message like this:

If the instance doesn't exist (deleted previously), you can safely use its name.

I can think of at least two scenarios where snapshots would be useful:

Like most other things in the cloud, RDS backup also comes with a price. Backup storage is a billable item and how you are charged will depend on how long you keep your backups and how often you create snapshots. Amazon provides free backup space of up to 100% of the instance's provisioned storage. Beyond that, charges will start to accrue. What this means is that if you have rolled out a 500 GB-month database instance, Amazon will give you a 500 GB-month backup space free of charge. That space can be used for automated backups as well database snapshots.

This doesn't apply to "idle" instances though: an RDS instance that gets backed up regularly, but has no user or app connecting to it will cost you money. Think about a staging system. You could be backing up a staging system before it goes live. Post roll-out the instance could be running idle with no one connecting to it - everything would have been migrated to the production system by then - and automated backups of the system would be unnecessarily incurring charges.

The main thing to be mindful here is the retention period. It's easy to get carried away and go for the maximum, but ask yourself, do you really need that many backups? Also, delete snapshots once their purpose is over.

We will talk about how to monitor RDS instance usage in a later tip.

In this tip we have tried to look at two different ways of backing up SQL Server RDS instances and restoring them. As with any backup plan, you still need to perform periodic test restores and run integrity checks against the databases. You will also need to formulate a retention policy for your backups. The retention period should create a balance between the company's RPO and the projected budget for RDS service. DBAs need to be mindful about creating a final snapshot before instances are decommissioned.

AWS Storage Gateway jolts cloud-storage ecosystem — Gigaom Research

http://research.gigaom.com/report/aws-storage-gateway-jolts-cloud-storage-ecosystem/ December 09, 2014

The Amazon Web Services (AWS) Storage Gateway, introduced in January, ... Use cases: Backup and disaster recovery: Primary storage, backup, disaster recovery, archive:

Amazon’s recent announcement of the AWS Storage Gateway was a surprise to some and for others just a natural extension of the online retailer’s strategy to grab all of a customer’s data and store it in its Simple Storage Service (S3) cloud.

The Amazon Web Services (AWS) Storage Gateway, introduced in January, is the company’s first foray into the on-premises cloud-storage space. The gateway is intended to be an on-ramp into the Amazon S3 and EC2 (Elastic Compute Cloud) networks for storing file data for data protection and disaster recovery. It is a way for customers to “kick the tires” of cloud-storage appliances that will get them used to working with the cloud before they graduate to a more sophisticated hardware-based appliance that meets their needs for true enterprise-class cloud storage. Implemented as software that installs as an image on a VMware ESXi 4.1 virtual machine, the gateway connects to the network via the iSCSI storage protocol, where it can attach to direct-attached storage (DAS) and network-attached storage (NAS). It supports the CIFS and NFS file protocols. The offering is in beta now, and it has the ability to cache files on-premises on the road map.

Amazon isn’t alone, however. A number of vendors are attacking the on-premises cloud-storage gateway market, among them Nasuni, Nirvanix, StorSimple and TwinStrata. Originally these companies called their products cloud-enablement appliances, thereby drawing attention to the purpose of the product: to enable organizations to put data in the service-provider cloud.

Since the cloud-storage gateways were first introduced in early 2010, the market has evolved. Rather than offering simply software- or hardware-based gateways to cloud storage, vendors are now offering a more complete storage solution, in which the gateway plays not only as an on-ramp to the cloud but also as a local on-premises repository for unstructured data — files, spreadsheets, images — and structured transactional data — databases, ERP and CRM systems. Vendors are also adding enterprise features such as deduplication and encryption to their gateway products and targeting them for not only the online backup and disaster recovery market but also for archive as well as primary storage.

According to recent research from Storage Strategies NOW, an industry analyst firm focusing on storage technologies, cloud storage is being adopted by North American organizations of all sizes. Of 187 respondents, 40 percent of enterprise-size organizations (5,000 to 100,000-plus employees) are deploying cloud storage, followed by 33 percent of midsize organizations (1,000 to 4,999 employees) and 27 percent of small businesses (1 to 999 employees). Further, they are deploying cloud storage as expected for email archives, followed by backup applications and primary-storage applications (front office and database applications).

It shouldn’t be any surprise that small businesses are among the first to adopt cloud storage for backup data and then disaster recovery. These businesses are often strapped with limited budgets and a lack of full-time, skilled IT staff, so they turn to clouds built or used by the value-added resellers (VARs) or managed service providers (MSPs) that manage their networks for support and deployment. In doing so — adopting cloud storage — they can easily and affordably replace shuttling tapes off-site for disaster recovery with cloud-based services that let them implement disaster-recovery strategies.

Table 1. What storage applications are you planning to deploy or have you deployed in the storage cloud? (Select all that apply.)

Source: Storage Strategies NOW, 2011

Midsize businesses are more resistant to adopting cloud storage, except for backup applications, citing in the study that they will wait until this year through 2014 to deploy public-cloud storage. For enterprise-size businesses, between 2012 and 2014, 45.3 percent plan to use the cloud for backup applications, 15.4 percent for archive purposes, 18.2 percent for disaster recovery and 22.7 percent for primary storage.

By contrast to other vendors of cloud-storage gateways, the AWS Storage Gateway looks anemic. It lacks a number of enterprise features such as encryption, where the user holds the encryption keys, deduplication and support for a local data cache. It supports only iSCSI block storage, which makes file-level restorations impossible. And it lacks scale: Volumes up to only 1 TB in size can be created, and they can’t be resized after creation. The AWS Storage Gateway also supports a maximum of only 12 volumes, in contrast to vendors such as Nirvanix that have no restriction on volume size or number of volumes per virtual machine.

In addition, the AWS Storage Gateway is implemented only as a virtual appliance. While a virtual appliance can easily support small installations or branch offices within larger organizations, enterprise users often want a physical appliance that reduces contention for resources and can support primary storage, storage bursting or online transaction processing (OLTP).

Another weakness of the AWS Storage Gateway is the cost for the system. At first blush the $125 per month charge for the gateway appears inexpensive when compared to other vendors, which offer all-inclusive, one-price-per-month or -year services. But when a user cobbles together the AWS Storage Gateway with the Amazon S3 or EC2 clouds, calculates the costs of storing data in the cloud and of getting it out (additional costs such as the transfer-out fees that have been factored into other vendor's offerings), and installs it into a VMware ESX 4.1–only environment, the costs can be steep. Imagine a small business restoring data from the cloud at a cost of $0.12 per GB of data stored.

Vendor lock-in is also inherent with the AWS Storage Gateway. Unlike products from TwinStrata, Nasuni and StorSimple, the gateway can be used with the Amazon S3 and EC2 clouds alone. This makes it difficult, if not downright impossible, for customers to migrate data between clouds run by different vendors if they are unsatisfied with the service. Nirvanix alone, like Amazon, only supports one cloud.

There are many other differences between the AWS Storage Gateway and other vendors' products that we won't discuss here but that you can see in Table 2 and Table 3.

And, finally, don't forget the AWS Storage Gateway is a beta version. Amazon has promised that a future version of its software will allow on-premise caching, where frequently accessed data will remain on local storage and only the entire data set will reside in its cloud.

The cloud-storage gateway as it evolves is being assimilated into many data-protection packages and hardware-based appliances as a feature rather than as a separate product. Although many times these are basic gateways that don't incorporate enterprise features, they are the future of cloud-storage gateways as indicated by these announcements in the recent past.

Last year EMC introduced the EMC Cloud Tiering Appliance, which integrates with its VNX storage array, to enable the tiering of inactive files into the EMC Atmos cloud. EMC also has the EMC Cloud Backup Option for its data-protection product, Networker. CommVault has been shipping a cloud-storage connector for its Simpana data information management software that allows customers to back up and archive data into Amazon S3, Microsoft Azure and Nirvanix clouds. Symantec isn't left out: With NetBackup 7.1, the company introduced a cloud connector for the Nirvanix cloud for backup. With NetBackup 7.5, the company is extending that to AT&T, Rackspace and Amazon S3 clouds.

Appliance vendors are also implanting cloud-enabled appliances. Quantum is expected to offer cloud backup with its DXi deduplication appliances, and Dell will integrate the cloud into its storage products as another tier of storage.

Adding the cloud as another tier of storage is perhaps an unrealized or at least unspoken goal of Amazon's. The company, which at present stores over 762 billion objects in Amazon S3 and processes over 500,000 requests per second, intends to own not a portion of your data but all of it. The AWS Storage Gateway is its first step in accomplishing this goal.

The AWS Storage Gateway doesn't signal the death of the storage gateway for all the reasons given above, but it provides validation that the vendors of gateway products — Nasuni, TwinStrata, Nirvanix and StorSimple — are headed down the right path with feature-rich products and all-inclusive cloud services. The challenge for startups like Nasuni, TwinStrata and StorSimple is to innovate faster than Amazon on features and to continue to partner with multiple cloud-storage providers, thereby offering a choice of storage clouds other than the Amazon-only model.

Adaptive Backup and Recovery

http://www.autonomy.com/html/preserve/adaptive-backup/index.html December 09, 2014

Adaptive Backup and Recovery from HP Autonomy combines ... At HP we are transforming data protection with a new and unique approach called Adaptive Backup and Recovery.

The world of IT and enterprise data protection has transformed rapidly and dramatically with the unabated growth of large volumes of data, a transition to virtualization and cloud deployments, and an increasingly mobile workforce that demands always-on operations. Taking a static, reactive data backup and recovery approach is no longer a fit for the new style of IT. IT directors, CIOs, and infrastructure administrators require a much more intelligent and dynamic approach to deliver not just backup and recovery, but business resiliency.

At HP we are transforming data protection with a new and unique approach called Adaptive Backup and Recovery. Going beyond traditional backup and recovery requirements it combines adaptive intelligence, operational analytics, and application and data awareness to meet the business resiliency requirements of today's highly dynamic, heterogeneous, and complex data centers. HP backup solutions deliver seamless management and federated deduplication backup, based on a single common technology and architecture.

With HP Adaptive Backup and Recovery, your organization can gain: Most optimal use of backup infrastructure and IT resources HP Adaptive Backup and Recovery leverages application and data intimacy and operational analytics to decide when, where, and how to protect data and make it available. At the heart of HP Adaptive Backup and Recovery is an intelligent engine that adapts and adjusts backup priorities to dynamically changing data, applications, and business requirements.

Adaptive intelligence: Automatically foresee, predict, and adapt to infrastructure changes. Industry-leading deduplication: HP StoreOnce-based federated deduplication can be deployed anywhere in the backup stack—application source, backup server, or backup target—to meet varying business needs.

Real-world uses of HP Adaptive Backup and Recovery: Centralize protection of the entire enterprise ecosystem: any device, any source, any environment Leverage operational analytics to gain visibility and control of the backup and recovery process while maximizing backup resource utilization