# Table of Contents

## Contents

About this Guide.................................................................................................................. 4  
Prerequisite Documents ..................................................................................................... 4  
SAP Notes .......................................................................................................................... 4  
Introduction ......................................................................................................................... 6  
Types of SAP Landscape Deployments on AWS ............................................................... 6  
  All-on-AWS ....................................................................................................................... 6  
    Diagram of All-On-AWS Topology .................................................................................. 6  
    Additional considerations for All-on-AWS Topology ...................................................... 7  
  Hybrid Topology ................................................................................................................ 7  
    Diagram of Hybrid Topology ......................................................................................... 8  
    Additional Considerations for the Hybrid Topology ....................................................... 9  
SAP Technical Operations on Amazon EC2 ....................................................................... 9  
Starting and Stopping the SAP System ............................................................................. 9  
SAP Licenses ....................................................................................................................... 9  
  High Availability .............................................................................................................. 10  
    Amazon EC2 Instance Failure ....................................................................................... 10  
    EBS Failure ................................................................................................................... 10  
    Disaster Recovery ......................................................................................................... 10  
      Multi Availability Zone (AZ) ...................................................................................... 10  
      Multi Region .............................................................................................................. 12  
    Amazon EC2 Reserved Instances .............................................................................. 12  
Security and User Administration ...................................................................................... 13  
  Network Configuration and Security .............................................................................. 13  
  Sample SAP Hybrid Deployment .................................................................................... 14  
    Amazon VPC with two subnets ..................................................................................... 15  
    DNS configuration ........................................................................................................ 18  
    Amazon Identity and Access Management (IAM) ......................................................... 18  
Monitoring ......................................................................................................................... 18  
Software Logistics ............................................................................................................ 19
About this Guide

This document provides guidance on the special considerations that must be taken into account when operating SAP solutions on Amazon Web Services (AWS). This document should be used as an AWS specific supplement to the standard Technical Operations for SAP Netweaver guide that can be found at:

http://help.sap.com/saphelp_nw70ehp1/helpdata/en/48/0dd91ad6013d1be10000000a42189d/frameset.htm

This guide will cover the following areas for operating SAP solutions on AWS:

- Overview of the Landscape deployment options
- SAP Licenses
- Performance Monitoring
- System Copy
- AWS Network Configuration and Security
- SAProuter and SAP Service Network configuration
- STMS Configuration
- Patching SAP Systems on-AWS
- Printing

Backup and Recovery of SAP systems on AWS are covered in a separate guide. This guide can be found at: http://aws.amazon.com/sap

- Please note that any denomination of SAP or AWS products in this guide is purely exemplary and does not overrule according SAP release notes. The latter are reference for SAP software, which is supported or not. Important support notes are listed below in section “SAP Notes”.

Prerequisite Documents

Before reading this guide please read the SAP on AWS Implementation Guide that can be found at http://aws.amazon.com/sap for information on installing SAP systems on the AWS Cloud. The SAP on AWS Implementation Guide provides information on:

- AWS Overview
- AWS Glossary
- AWS Networking Overview and Options
- Special considerations when installing SAP systems on AWS
- Sizing and Performance
- Creating Instances, EBS Volumes, etc.
- IO performance / EBS striping
- Preparing SUSE / Red Hat instances for installation of SAP
- VPC Overview and How to Create (link)
- Reference to AWS SAP

SAP Notes

Please read the following SAP notes before proceeding with this guide.
<table>
<thead>
<tr>
<th>Note #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1588667</td>
<td>SAP on AWS: Overview of related SAP Notes and Web-Links</td>
</tr>
<tr>
<td>1656099</td>
<td>SAP on AWS: Supported SAP, DB/OS and AWS EC2 products</td>
</tr>
<tr>
<td>1656250</td>
<td>SAP on AWS: Support prerequisites</td>
</tr>
</tbody>
</table>
Introduction

Types of SAP Landscape Deployments on AWS

SAP products can be implemented resulting in a number of different landscape topology patterns

- **Standalone**
  Under this category fall smaller systems like sandbox, POC, demo or training systems. These types of systems are usually not interconnected with other systems. Typically, the lifetime of these systems is limited and thus causes only minor operational effort.

- **All-on-AWS**
  These are landscapes, where all (i.e. more than one) systems reside in one cloud location. In- and outbound (WAN) connections are limited to user frontends and remote support connections. Inter-system connectivity can be assumed to work with LAN latency and LAN throughput.

- **Hybrid**
  These are landscapes which are distributed across on-premise and cloud locations. In between the two location types, only WAN connectivity can be assumed. Compared to LAN, higher latency and lower throughput as well as high variations have to be expected. Reasonable operations can be expected challenging.

Each non-trivial landscape topology (i.e. All-on-AWS and Hybrid) requires further solution planning, which will be discussed in the following sections.

**All-on-AWS**

An All-On-AWS landscape has all SAP systems on the AWS cloud. Some customer systems might still be located on-premise like DNS, Active Directory or LDAP services, File Server, Print Servers and attached Printers and mail and groupware applications.

**Diagram of All-On-AWS Topology**

Figure 1 shows an exemplary All-On-AWS topology with all SAP systems running on AWS. Users access the SAP systems from their corporate network or remote locations from the internet, a VPN or an Amazon DirectConnect connection.

**Systems on AWS**

- **AWS VPC Private Subnet**
  - Development and Production ECC
  - Development and Production Portal
  - Solution Manager

- **AWS VPC Public Subnet (internet accessible)**
  - Development Web Load Balancer
  - Production Web Load Balancer
  - SAProuter
Additional considerations for All-on-AWS Topology

The SAProuter is a required component for a Production SAP landscape to get adequate SAP Support; SAP’s support systems are denoted with sapservx in Figure 1. Instructions on how to setup the SAProuter for on All-on-AWS deployment are provided later in this guide.

Traffic back to any services to any on-premise systems must be enabled through the AWS Internet Gateway. Please refer to the “Network Configuration and Security” section for more information.

Furthermore DNS services should be considered. DNS should be setup in AWS or point back to the on-premise DNS system. The steps to configure DNS on Linux and Windows are provided in section “DNS configuration” later in this guide.

Hybrid Topology

The Hybrid Topology has a mixture of SAP systems on the AWS Cloud and on-premise. A sample topology could include Development systems on AWS and Production systems On-Premise or vice versa.
It is not recommended to distribute SAP systems of a production landscape between AWS Cloud and on-premise data centers, as these are best connected through low-latency networks with least possible network hops, either All-on-AWS or on-premise.

**Diagram of Hybrid Topology**

Figure 2 shows an example Hybrid AWS topology with the production systems running on-premise and Development systems on the AWS cloud.

**On-Premise Systems**
- ECC Prod and Solution Manager with SLD

**On-AWS Systems**
- In private VPC:
  - Dev ECC and Dev Enterprise Portal System
- In DMZ (accessible from public internet):
  - Web Load Balancer (the SAP Web Dispatcher can be utilized for this purpose)
Additional Considerations for the Hybrid Topology

Additional steps need to be considered for SAP’s Transport Management System (STMS) setup. These steps are documented in section “Configuration of STMS for a Hybrid Landscape”.

All systems in AWS DNS lookups should point back to the on-premise DNS system. The steps to configure DNS on Linux and Windows are provided in section “DNS configuration”.

Caution
A distribution of interconnected production systems across on-premises and on AWS, respectively multiple AWS Cloud locations, should not be considered. Synchronous communication over WAN may fail with unwanted effects.

SAP Technical Operations on Amazon EC2

Starting and Stopping the SAP System
By default, Amazon EC2 instance startup or shutdown using the AWS Management Console or API will not automatically start and stop the SAP application and database within the Amazon EC2 instance.

SAP services should be started and stopped on Amazon EC2 just like on the corresponding operating system on classical on-premise infrastructure. If sapstartsrv is started and its user permissions are set up correctly as described in SAP note 927637, the following commands can be used – both on Linux as on Windows – to manually stop and start SAP services:

```
  sapcontrol -nr <instance nr> -function StopService
  sapcontrol -nr <instance nr> -function StartService <SID>
```

To automatically stop and start SAP services and their dependencies while the corresponding Amazon EC2 instance is being stopped or started, appropriate scripts need to be created that correctly link into the shutdown and startup sequences of the operating system.

More information on the SAP Start Service framework can be found at http://help.sap.com/saphelp_nwes72/helpdata/en/b3/903925c34a45e28a2861b59c3c5623/content.htm

SAP Licenses
SAP licensing has been enhanced for Amazon EC2. Please install an SAP kernel with a patch level that includes this enhancement, as described in SAP note 1656099.

The adapted SAP license algorithm will use the Amazon EC2 instance-id as a unique identifier for the generating license’s hardware key. You can find the Amazon EC2 instance-id locally on your Amazon EC2 instance through:

More information on Amazon EC2 instance metadata can be found at:

**High Availability**

The standard method for providing high availability for an SAP system, by protecting the single points of failure (SPOF), is to use a failover cluster. Currently it is not possible to install a traditional failover cluster on AWS, but this does not mean that it is not possible to provide different levels of high availability for an SAP system. The following section describes different failure topologies and possible architectural solutions that can be utilized to provide maximum availability for an SAP system.

**Amazon EC2 Instance Failure**

- **DB Instance**
  - **Option #1** - DB replication between two Amazon EC2 Instances (e.g. MS SQL Server Mirroring, etc.). To provide additional protection, you can create the secondary database in a second Availability Zone (see Disaster Recovery in next section).
  - **Option #2** - Launch new DB instance from AMI and attach EBS volumes from failed DB instance to new DB instance

- **SAP Central Instance**
  - **Option #1** – Setup standby SAP CI and configure enqueue replication between systems. In case of primary SAP CI failure, manually fail over to the secondary SAP CI. Note that this HA option is only available for distributed (3-tier) SAP system deployments.
  - **Option #2** – Launch new SAP CI instance from AMI

**EBS Failure**

- **DB Instance**
  - DB replication between two Amazon EC2 Instances (e.g. MS SQL Server Mirroring, etc.). To provide additional protection you can create the secondary database in a second Availability Zone (see Disaster Recovery in next section).

- **SAP Central Instance**
  - Same options as with Amazon EC2 Instance Failure (above).
  - **Option #1** – Setup standby SAP CI and configure enqueue replication between systems. In case of primary SAP CI failure, manually fail over to the secondary SAP CI. Note that this HA option is only available for distributed (3-tier) SAP system deployments.
  - **Option #2** – Launch new SAP CI instance from AMI

**Disaster Recovery**

Depending on the requirements of your business AWS provides multiple solutions for addressing Disaster Recovery. To protect your SAP systems against various natural and manmade disasters you can utilize the different AWS Regions and Availability Zones.

**Multi Availability Zone (AZ)**

Each AWS Availability Zone is designed as an independent failure zone. This means that Availability Zones are typically physically separated within a metropolitan region and are in different flood plains. In addition to discrete uninterruptable power source (UPS) and onsite backup generation facilities, they are each fed via different grids from independent utilities to further reduce single points of failure. Availability Zones are all redundantly connected through multiple independent network providers.
Depending on the Recovery Point Objective (RPO) and Recovery Time Objective (RTO) requirements of your business there are two primary Multi AZ topologies that you can utilize to protect your SAP environment in case of an AZ failure.

**Multi AZ Replication**

The SAP database is replicated to a standby database in a secondary AZ by means of log shipping or database replication (Oracle Data Guard, MS SQL Server Mirroring, etc.). If the primary AZ fails, the DB in the secondary AZ is available to take over from the primary DB. You can then directly rebuild your SAP CI and DIs by launching new instances from their most recent AMI backups.

**Multi AZ Backup/Restore**

If the RPO and RTO requirements of the business do not warrant the additional cost of DB replication you can use a simple DB backup/restore between AZs to recover your SAP database in the case of primary AZ failure. Like the Multi AZ replication topology you can quickly rebuild your SAP CI and DIs utilizing AMIs to launch new instances.
Multi Region
If the fault separation provided by multiple Availability Zones does not meet the requirements of your business, then multiple Regions can be utilized for Disaster Recovery.

Multi Region Disaster Recovery requires additional effort compared to Multi AZ Disaster Recovery since AMIs and Snapshots created in a Region are only accessible from within that Region. Currently AWS does not provide tools to replicate these objects between Regions. Also, unlike Availability Zones within the same region, there is no high speed low latency network connection between Regions.

While it is possible to architect a Disaster Recovery solution using multiple Regions the details of such a solution are beyond the scope of this document. If your business requires Multi Region Disaster Recovery you should consult with an AWS Service Provider who is experienced in Multi Region architectures.

Amazon EC2 Reserved Instances
All of the techniques examined above rely on the assumption that you will be able to procure Amazon EC2 Reserved Instances whenever you need them.

Amazon Web Services has massive hardware resources at its disposal, but like any cloud computing provider, those resources are finite. The best way for users to maximize their access to these resources is by reserving a portion of the computing capacity that they require. This can be done through a feature called Reserved Instances.

With Amazon EC2 Reserved Instances, you literally reserve computing capacity in the Amazon Web Services cloud.
For additional information on Reserved Instances please visit http://aws.amazon.com/ec2/#pricing

Security and User Administration
The following sections will describe the methods that AWS offers to configure a secure network through Amazon Virtual Private Cloud (Amazon VPC).

Network Configuration and Security
The AWS Virtual Private Cloud (Amazon VPC) enables you to create a virtual network topology very similar to a typical on-premise network topology. Amazon VPC enables very familiar networking concepts like subnets, routing and firewalls.

By using Amazon VPC with Amazon EC2 you can:

- Logically group your Amazon EC2 instances, and assign them private IP addresses
- Control the outbound traffic from your Amazon EC2 instances (in addition to controlling the inbound traffic to them)
- Add an additional layer of security to your Amazon EC2 instances in the form of network Access Control Lists (ACLs)
- Connect your VPC to your corporate data center and branch offices with a VPN connection, so that you can use the VPC as an extension of your corporate data center network

When operating SAP systems on AWS, just as with operating SAP systems on-premise, you may require some system to be accessible from the Internet (e.g. web servers, web load balancers, SAProuter, external facing Portal, etc.) and some systems to only be accessible from your corporate network (e.g. SAP ERP, SAP BI, SAP SCM, etc). This type of network separation/zoning can be achieved with an AWS VPC utilizing multiple subnets, network access control lists and security groups. Figure 3 below shows an example VPC with both public (Internet accessible) and private (no Internet access) subnets and a physical VPN connection to connect your corporate network to your AWS VPC.

For detailed documentation how to setup a VPC like the one in Figure 5 please read:

- Amazon Virtual Private Cloud User Guide
  http://docs.amazonwebservices.com/AmazonVPC/latest/UserGuide
- Amazon Virtual Private Cloud Network Administrators Guide
  http://docs.amazonwebservices.com/AmazonVPC/latest/NetworkAdminGuide
Sample SAP Hybrid Deployment
This section shows a sample Hybrid deployment with the SAP production landscape on-premise and the SAP development landscape on-AWS.

In Figure 6 below “AWS Network 1” is a private subnet within the VPC and the systems within this subnet are only accessible from the corporate on-premise network. “AWS Network 2” is a public subnet within the VPC and the system within this subnet can be accessed from the Internet.

The section following Figure 6 provides details of the different subnets, network ACLs and security groups that were created to build the sample deployment.
Note that the used IP addresses, network masks below and security configurations are mere examples that should be adapted to the customer’s network setup and security requirements.

Amazon VPC with two subnets

<table>
<thead>
<tr>
<th>Network</th>
<th>CIDR</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS Network 1</td>
<td>10.160.2.0/25</td>
<td>Internal</td>
</tr>
<tr>
<td>AWS Network 2</td>
<td>10.160.2.128/28</td>
<td>DMZ</td>
</tr>
</tbody>
</table>

Figure 6: SAP Systems in AWS VPC utilizing a DMZ

**Table:**

<table>
<thead>
<tr>
<th>Network</th>
<th>CIDR</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS Network 1</td>
<td>10.160.2.0/25</td>
<td>Internal</td>
</tr>
<tr>
<td>AWS Network 2</td>
<td>10.160.2.128/28</td>
<td>DMZ</td>
</tr>
</tbody>
</table>
Routing tables

AWS Network 1 – non-DMZ systems, can route to on-premise

<table>
<thead>
<tr>
<th>Route</th>
<th>Gateway</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.159.0.0</td>
<td>VPN</td>
<td>Connectivity to On-premise systems</td>
</tr>
<tr>
<td>10.160.2.0/24</td>
<td>Local</td>
<td>Local connectivity in AWS (Network 1 and 2)</td>
</tr>
<tr>
<td>0.0.0.0</td>
<td>Internet Gateway</td>
<td>Internet</td>
</tr>
</tbody>
</table>

AWS Network 2 (DMZ) – cannot route to on-premise

<table>
<thead>
<tr>
<th>Route</th>
<th>Gateway</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.160.2.0/24</td>
<td>Local</td>
<td>Local connectivity in AWS (Network 1 and 2)</td>
</tr>
<tr>
<td>0.0.0.0</td>
<td>Internet Gateway</td>
<td>Internet</td>
</tr>
</tbody>
</table>

Note: The networks CIRD are merely examples.

AWS instances IP Addresses

<table>
<thead>
<tr>
<th>Instance</th>
<th>IP</th>
<th>Elastic IP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Utility Server</td>
<td>10.160.2.4</td>
<td>No</td>
</tr>
<tr>
<td>ECC instance</td>
<td>10.160.2.5</td>
<td>No</td>
</tr>
<tr>
<td>CRM instance</td>
<td>10.160.2.6</td>
<td>No</td>
</tr>
<tr>
<td>BW instance</td>
<td>10.160.2.7</td>
<td>No</td>
</tr>
<tr>
<td>EP instance</td>
<td>10.160.2.8</td>
<td>No</td>
</tr>
<tr>
<td>Web Dispatcher</td>
<td>10.160.2.132</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Security Groups

Staging instance

<table>
<thead>
<tr>
<th>Direction</th>
<th>PORT</th>
<th>Source/Target</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inbound</td>
<td>TCP/3389</td>
<td>0.0.0.0/0</td>
<td>RDP Access</td>
</tr>
<tr>
<td>Outbound</td>
<td>ALL</td>
<td>0.0.0.0/0</td>
<td></td>
</tr>
</tbody>
</table>

ECC instance

<table>
<thead>
<tr>
<th>Direction</th>
<th>PORT</th>
<th>Source/Target</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inbound</td>
<td>TCP/22</td>
<td>0.0.0.0/0</td>
<td>OS access</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/3200</td>
<td>0.0.0.0/0</td>
<td>SAP Access (Sap Dispatcher/Sap)</td>
</tr>
<tr>
<td>Direction</td>
<td>PORT</td>
<td>Source/Target</td>
<td>Usage</td>
</tr>
<tr>
<td>-----------</td>
<td>------------</td>
<td>---------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/3600</td>
<td>0.0.0.0/0</td>
<td>Gateway/Message Server</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/8000</td>
<td>0.0.0.0/0</td>
<td>ICM HTTP</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/7210</td>
<td>0.0.0.0/0</td>
<td>MaxDB (used for remote Maxdb administration)</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/21200</td>
<td>0.0.0.0/0</td>
<td>Used only for Sapinst</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/21212</td>
<td>0.0.0.0/0</td>
<td>Used only for Sapinst</td>
</tr>
<tr>
<td>Outbound</td>
<td>ALL</td>
<td>0.0.0.0/0</td>
<td></td>
</tr>
</tbody>
</table>

**BI instance**

<table>
<thead>
<tr>
<th>Direction</th>
<th>PORT</th>
<th>Source/Target</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inbound</td>
<td>TCP/22</td>
<td>0.0.0.0/0</td>
<td>OS access</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/3200</td>
<td>0.0.0.0/0/0</td>
<td>SAP Access (Sap Dispatcher/Sap Gateway/Message Server)</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/3300</td>
<td>0.0.0.0/0/0</td>
<td>SAP Access (Sap Dispatcher/Sap Gateway/Message Server)</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/3600</td>
<td>0.0.0.0/0/0</td>
<td>SAP Access (Sap Dispatcher/Sap Gateway/Message Server)</td>
</tr>
<tr>
<td>Outbound</td>
<td>ALL</td>
<td>0.0.0.0/0</td>
<td></td>
</tr>
</tbody>
</table>

**CRM instance**

<table>
<thead>
<tr>
<th>Direction</th>
<th>PORT</th>
<th>Source/Target</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inbound</td>
<td>TCP/22</td>
<td>0.0.0.0/0</td>
<td>OS access</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/3200</td>
<td>0.0.0.0/0/0</td>
<td>SAP Access (Sap Dispatcher/Sap Gateway/Message Server)</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/3300</td>
<td>0.0.0.0/0/0</td>
<td>SAP Access (Sap Dispatcher/Sap Gateway/Message Server)</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/3600</td>
<td>0.0.0.0/0/0</td>
<td>SAP Access (Sap Dispatcher/Sap Gateway/Message Server)</td>
</tr>
<tr>
<td>Outbound</td>
<td>ALL</td>
<td>0.0.0.0/0</td>
<td></td>
</tr>
</tbody>
</table>

**EP instance**

<table>
<thead>
<tr>
<th>Direction</th>
<th>PORT</th>
<th>Source/Target</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inbound</td>
<td>TCP/22</td>
<td>0.0.0.0/0</td>
<td>OS access</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/3200</td>
<td>0.0.0.0/0/0</td>
<td>SAP Access (Sap Dispatcher/Sap Gateway/Message Server)</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/3300</td>
<td>0.0.0.0/0/0</td>
<td>SAP Access (Sap Dispatcher/Sap Gateway/Message Server)</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/3600</td>
<td>0.0.0.0/0/0</td>
<td>SAP Access (Sap Dispatcher/Sap Gateway/Message Server)</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/8000</td>
<td>0.0.0.0/0</td>
<td>ICM HTTP</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/8001</td>
<td>0.0.0.0/0</td>
<td>ICM HTTP</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/7210</td>
<td>0.0.0.0/0</td>
<td>MaxDB (used for remote MaxDB administration)</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/21200</td>
<td>0.0.0.0/0</td>
<td>Used only for Sapinst</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/21212</td>
<td>0.0.0.0/0</td>
<td>Used only for Sapinst</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/50000</td>
<td>0.0.0.0/0</td>
<td>J2EE HTTP</td>
</tr>
<tr>
<td>Outbound</td>
<td>ALL</td>
<td>0.0.0.0/0</td>
<td></td>
</tr>
</tbody>
</table>
Web Dispatcher instance

On Network 2 or DMZ, this instance has an elastic IP address published to the Internet, so the security is being setup on the security group below.

<table>
<thead>
<tr>
<th>Direction</th>
<th>PORT</th>
<th>Source/Target</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inbound</td>
<td>TCP/22</td>
<td>10.160.2.0/25 (Subnet 1)</td>
<td>OS access</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/80</td>
<td>0.0.0.0/0</td>
<td>HTTP</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/21200</td>
<td>10.160.2.0/25 (Subnet 1)</td>
<td>Used only for Sapinst</td>
</tr>
<tr>
<td>Inbound</td>
<td>TCP/21212</td>
<td>10.160.2.0/25 (Network 1)</td>
<td>Used only for Sapinst</td>
</tr>
<tr>
<td>Outbound</td>
<td>TCP/8101</td>
<td>ERD SG</td>
<td>Connectivity to ERD instance</td>
</tr>
<tr>
<td>Outbound</td>
<td>TCP/50000</td>
<td>ERD SG</td>
<td>Connectivity to ERD instance</td>
</tr>
<tr>
<td>Outbound</td>
<td>TCP/50004</td>
<td>ERD SG</td>
<td>Connectivity to ERD instance</td>
</tr>
<tr>
<td>Outbound</td>
<td>TCP/50004</td>
<td>ERD SG</td>
<td>Connectivity to ERD instance</td>
</tr>
<tr>
<td>Outbound</td>
<td>TCP/53 UDP/53</td>
<td>0.0.0.0/24</td>
<td>DNS lookup</td>
</tr>
</tbody>
</table>

Note: This is just an example; the Source network can be changed on the inbound to match the network security requirements

DNS configuration

Instructions to change DNS configuration on SUSE systems

1. Logon to the instance OS as root
2. Execute the command “yast”
3. Navigate to “Network Devices/Network Settings”
4. Type “Alt-S” keys to modify the “Hostname/DNS”
5. Goto the name Servers and Domain Search List and configure the DNS servers IP
6. Type F10 key to accept the changes
7. Type F9 key to quit Yast

Amazon Identity and Access Management (IAM)

Access the functions of the AWS Console and APIs can be set up for multiple users with different privileges.

For more detailed information, please visit http://aws.amazon.com/documentation/iam/

Monitoring

To receive support for your SAP installation, please carefully follow SAP Note 1656250 - SAP on AWS: Support prerequisites
Software Logistics

SAP Transport (STMS) landscapes are an integral part of SAP production software logistics. A typical and most simple example of a 3-system SAP transport landscape is DEV → QAS → PRD, where repository and customizing changes are created in DEV and tested in QAS before they are delivered to PRD.

The following section will describe a 2-system STMS configuration (DEV → PRD) for a Hybrid landscape deployment, where the Production system is installed on-premise and its Development system On-AWS.

Configuration of STMS for a Hybrid Landscape

To connect the AWS Development ECC system to the on-premise Prod ECC system, the Amazon VPC must be utilized. There will be two transport groups and hence two separate transport directories. STMS will be setup with one transport domain.

In this example we are setting up the on-Premise Production ECC System (ERP) as the domain controller. We add the AWS Cloud Development ECC System (ERD) to this domain, as shown in Figure 7.

Figure 7: one TMS transport domain and two transport groups

Configuration Steps

1. In the domain controller SAP system (in this example ERP), execute transaction STMS as user DDIC in client 000 and enter the following:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>&lt;SID&gt; transport system</td>
<td>ERP transport system</td>
</tr>
<tr>
<td>Transport Domain Name</td>
<td>DOMAIN&lt;SID&gt; (where &lt;SID&gt; is the domain controller SID)</td>
<td>DOMAIN_ERP</td>
</tr>
<tr>
<td>Transport Domain</td>
<td>Transport Domain &lt;SID&gt; (where &lt;SID&gt; is the</td>
<td>Transport Domain ERP</td>
</tr>
<tr>
<td></td>
<td>domain controller SID)</td>
<td></td>
</tr>
</tbody>
</table>
2. Click on ‘Save’ button ( ).
3. System ERP is now configured as domain controller.
4. In the development system (ERD), execute transaction STMS as user DDIC in client 000. By default, the attempt will be made to create new transport domain DOMAIN_ERD and set ERD as domain controller.
5. Click on button (Other Configuration (F6)). Now you can include system ERD to another transport domain by providing host name and system number of domain controller as it shown below:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>System &lt;SID&gt;</td>
<td>System ERD</td>
</tr>
<tr>
<td>Target Host</td>
<td>&lt;Hostname&gt;</td>
<td>PROTSERP00.appcarecloud.com</td>
</tr>
<tr>
<td>System Number</td>
<td>&lt;Instance Number&gt;</td>
<td>00</td>
</tr>
</tbody>
</table>

6. Click on ‘Save’ button ( ). ERD is now waiting to be included into the transport domain DOMAIN_ERP.
7. In system ERP, navigate to transaction STMS and click on ‘System Overview’ button or select Overview \(\rightarrow\) Systems.
8. You will notice that ERD is waiting for inclusion into transport domain.
9. Highlight ERD entry and click on ‘Approve’ button ( ).
10. Confirm the pop-up window by clicking on ‘Yes’ button.
11. New transport group GROUP_ERD will be created.
12. System ERD is successfully added to transport domain.
13. Next step is configuration of transport routes. In system ERP, navigate to transaction STMS and click on ‘Transport Routes’ button ( ) or select Overview \(\rightarrow\) Transport Routes.
14. On the next screen, switch to the change mode.
15. Navigate to menu Configuration \(\rightarrow\)Standard Configuration \(\rightarrow\) Development and Production System.
16. Provide development (ERD) and production system ID (ERP) and hit ‘Enter’.
17. Two routes for standard SAP and custom objects will be generated.
18. Activate configuration by navigating to menu option Configuration \(\rightarrow\) Distribute and Activate
19. Enter configuration description, e.g. ‘Two System Configuration’.
20. Confirm next pop-up window by clicking on ‘Yes’ button.
21. Transport route configuration will be activated.
Backup and Recovery
A separate guide is available that covers SAP Backup and Recovery on AWS. Please visit http://aws.amazon.com/sap to download the latest version of this guide.

Data Archiving
The Amazon Simple Storage Service (Amazon S3) provides an ideal platform for archiving SAP data, mainly due to its high durability and availability. Objects can directly be up- and downloaded using the HTTPS and can be encrypted at rest, also by using an own key pair.

Direct integration of Amazon S3 with SAP’s archiving interfaces is available from other SAP partners. To find partners with certified implementations, please query the SAP Partner Information Center, available at: http://www.sap.com/partners/directories/SearchSolution.epx

Certification Category: “Archiving”

For more information on Amazon S3, please visit: http://aws.amazon.com/s3/

System Copy
In general, SAP homogeneous and heterogeneous system copy methods are the same as described in the existing SAP documentation.

However, the following built-in AWS methods can be leveraged as well:

- Clone systems through Amazon Machine Images (AMIs)
- Clone individual EBS volumes through EBS snapshots

Clone systems through Amazon Machine Images (AMIs)
The main procedure to create a system (Amazon EC2 Instance) clone consists of the following two major steps:

1. Create an AMI from the source EC2 Instance
2. Launch a new instance from the AMI to create a new target EC2 Instance

The first procedure to create an AMI from an existing source system works as follows:

1. Shutdown the SAP and database instances cleanly
2. Within the AWS Management Console open the “Instances” pane and select the EC2 instance
3. Select “Instance Actions”, then “Create Image (EBS AMI)”
   Follow the steps of the wizard
4. After the Amazon EC2 instance is available again, startup the database and SAP instances again

The subsequent procedure to launch a new Amazon EC2 Instance from an AMI is as follows:

1. Within the AWS Management Console, open “AMIs” and select your AMI
2. Select “Launch”  
   Follow the steps of the wizard

After the new instance got created, you can log in to it like you do with the source system. As the new EC2 instance got a new IP address assigned, you need to adapt the local hostname to point to that new IP address (e.g. adapt /etc/hosts on linux). After that, the database and SAP instances should be able to start. A new SAP license is required, as the new EC2 instance has a new instance-id.

The described procedure can be automated by leveraging Amazon EC2 APIs and SDKs. For further details, please visit [http://aws.amazon.com/documentation/ec2/](http://aws.amazon.com/documentation/ec2/).

**Clone individual EBS volumes through EBS snapshots**

Instead of cloning a complete source system to create a new target system, an existing target system can be “refreshed” by leveraging EBS Snapshot functionality.

The main procedure consists of the following major steps:

1. Create a consistent EBS Snapshot of the source database EBS volumes
2. Dismount target database volumes on the existing target system
3. Create new volumes from the source database EBS snapshots and mount these as new target database volumes on the existing target system
4. Execute the usual SAP technical post processing steps, e.g.
   a. Rename, recover and startup database
   b. Cleanup database (e.g. source system RFC destinations, planning calendar, etc.)
   c. Startup SAP instance
   d. Execute the usual SAP homogeneous system copy post-processing steps

The described procedure can be automated by leveraging Amazon EC2 APIs, SDKs. For further details, please visit [http://aws.amazon.com/documentation/ec2/](http://aws.amazon.com/documentation/ec2/).

Note: the SAP technical steps above (item 4.) should be carried out by an experienced SAP Technical Consultant who is well acquainted with the methods manually cloning and renaming SAP systems, and should have good background knowledge of the technical SAP customizing to carry out these steps correctly. Many of these steps have already been automated through scripts in SAP customer environments today; such scripts can directly be re-used for SAP deployments on AWS.

**Setting Up Service Connections for SAP Remote Support**

An integral part of setting up the service connections for SAP remote support is the configuration of SAProuter. This will be described in the following sections.

**SAProuter and SAP Service Network Connection**

Below, SAProuter configuration will be described for Hybrid and All-On-AWS SAP landscape deployments.
Hybrid Deployment
In the case of a Hybrid deployment where the SAP production landscape is on-premise and one or more other SAP landscapes is on-AWS, it is recommend that the SAProuter and connection to the SAP Remote Network are kept on-premise. Connective between the on-premise SAProuter and the on-AWS SAP systems can be easily established via the VPN connection between your on-premise network and your VPC.

All-On-AWS Deployment
With an All-On-AWS configuration you must install a SAProuter within a public subnet so the SAProuter can be access from the SAP Support Network. The network connection type that is required to connect an on-AWS SAProuter to the SAP Support Network is the “SNC over Internet connection” type. For additional information on setting the connection to the SAP Support Network please visit: http://service.sap.com/access-support

Steps for configuring SAPRouter
1. Launch an instance within the public subnet of your VPC and assign it an Elastic IP (EIP) address
2. Install and configure the SAProuter software on the new instance as documented on the SAP web site http://help.sap.com/saphelp_nwmobile71/helpdata/en/65/8d09ab5c7e46028f633bb01a09b380/content.htm

Troubleshooting for SAP Web Application Server
The following sections will describe the following topics:

- Applying software patches to SAP systems deployed on AWS
- Tested printer configurations for SAP systems deployed on AWS

Applying SAP Patches to Systems on AWS
The process of applying patches to SAP systems on AWS is very similar to the process for on-premise systems. A few additional considerations can help speed and ease the process of patching systems with SUM (Software Update Manager) on AWS.

Hybrid Deployment
In a typical on-premise configuration a single shared transport directory is configured between all landscapes (i.e. DEV/QAS/PRD). While it is still possible to setup this type of configuration when running systems in a hybrid deployment the increased network latency introduced between on-premise and on-AWS systems may make it impractical to import large patches from a shared transport directory. Since the amount of network latency between an on-premise network and an AWS VPC is dependent on many different factors (i.e. quality and speed of customer Internet connection, proximity to AWS Region, use of AWS DirectConnect, etc.), it is not possible to provide specific guidance on the maximum sized patch that can be applied from a remote transport directly. If you do experience issues importing large patches across on-premise and on-AWS a workload would be to first copy the patch files to the local server and then import the patch from the local server.
Applying OS Patches to Systems on AWS

SLES 11
To initiate a connection to the repository, execute the following commands:
# sces-activate
# zypper ref -s -f

SLES instances must have an elastic IP address and must be able establish outbound connections to the Internet over http or https to be able to connect to the SLES repository.

Printing from AWS to On-Premise Printers
The following configurations will be described for printing from SAP on AWS to on-premise printers:

- Setup of an output device using an on-premise network print spooler
- Setup an output device spooling directly from SAP system to on-premise printer
- Setup of local (LOCL) front-end printing

The examples below are specific for the spooler daemons on the Linux operating systems; the overall procedures will work similar with Windows print spooler services.

Setup of an output device using an on-premise network print spooler
This will print from the SUSE Linux server directly to the network printer through port 515.

Define Printer on Linux Spooler
On Linux we need CUPS (Common Unix Printing System) and LPD (Line Printer Daemon) to print.

The commands to enable LPD on a Linux server are as follows:
1) vi /etc/xinetd.d/cups-lpd
2) change the line "disable = yes" by "disable = no"
3) Save the file /etc/xinetd.d/cups-lpd
4) Run the command: "restart xinetd"
5) Run the command: "service xinetd stop"
6) Run the command: "service xinetd start"

Now LPD is enabled.

CUPS may need to be installed on the SUSE system also if not on the Amazon Machine Image (AMI).

Create the printer definition on the SLES OS
1) Run the command lpstat -t
   Example of the output of the command
   "device for lprt0002: lpd://<print-server> /AUTO"

2) In the hosts file enter the IP address for the printer.
   Example of entry:
   10.159.93.14 hplj2055 HPLj2055

3) To add the printer, use the lpadmin command.
   Example commands for adding an LPD enabled printer. Uses port 515:
Example of adding a JetDirect LaserJet printer. Uses port 9100 (9101 and 9102 if multiport):
lpadmin -p lpri0003 -v socket://10.159.93.14/9100 -P
/usr/share/cups/model/laserjet.ppd.gz -E

Create the printer device in transaction SPAD
Configure the printer in transaction SPAD as follows:

Device Attributes: Device Type = SAPWIN
Host Printer: <name defined on the linux OS>
Destination Host: <IP address of the print server>

Setup an output device spooling directly from SAP system to on-premise printer
This will print from SAP directly to the network printer through port 515.

Create the printer device in SAP
Configure the printer in SPAD in the SAP system as follows:

Device Attributes: Device Type = SAPWIN
Host Printer: <network printer name>
Destination Host: <network printer name>

Setup of local (LOCL) front-end printing
Use a SAPGUI on a network which has network connectivity to the SAP system in AWS.
Ensure the port 32xx (where xx is the instance number) is open on the network. This is the same port SAPGUI communicates over.

Configure the printer in SPAD in the SAP system as follows:

Device Attributes: Device Type = SAPWIN
Access Method: Host Printer = __DEFAULT

The printer will print to the Default printer on the Windows PC.

Closing Remarks
All information provided herein represents knowledge at the time of the document’s writing. In case of unclear or erroneous statements, feedback to sap-on-aws@amazon.com is highly appreciated.