HITACHI
Inspire the Next

Best Practices for Pentaho and Amazon Web Services

HITACHI
Inspire the Next

Change log (if you want to use it):

Date	Version	Author	Changes

Contents

Overview 1

Scope 1

Best Practices and Recommendations for AWS 2

Running Pentaho Components in AWS 2

Virtual Networks 2

Extending Corporate WAN to include AWS 3

Network Security Groups and Subnets 3

Pentaho Repository Database 4

Pentaho Server VM and Storage 4

EC2 Considerations 5

Server and Repository Maintenance 5

Known Issues 6

Repository Database – Aurora 6

Elastic Load Balancer (ELB) and Auto Scaling Groups 6

Related Information 6

Finalization Checklist 7

This page intentionally left blank.

Overview

This document covers some best practices on the installation of Pentaho's Server and Client products on Amazon Web Services (AWS), and gives an overview of the server, network, and storage architecture recommended to run Pentaho.

Pentaho products are accompanied by detailed Installation Instructions that explain how to configure servers, storage, and databases to host the Pentaho products. However, these installation instructions do not capture the data center environments in which the hardware and storage might be hosted. This best practice document complements the product installation instructions by showing how the server and database configurations can be applied specifically in an AWS elastic compute cloud (EC2) environment.

Scope

The primary focus of this document is the infrastructure architecture to support deployment of Pentaho's server and client products on AWS EC2. It does not focus on integrating Pentaho with other AWS services such as Elastic MapReduce, Redshift, or QuickSight.

We recommend you have a network architect design the logical network for Pentaho products, spanning AWS and your own locations, accounting for scale and security and complying with your unique technology standards and business requirements.

The information in this document covers the following versions:

Software	Version(s)
Pentaho	7.x, 8.x

The Components Reference in Pentaho Documentation has a complete list of supported software and hardware.

Table 1: Pentaho Products and Descriptions

Product Category	Details
Pentaho Server	Installed on server equipment and runs unattended, typically 24x7. It executes Pentaho jobs and stores shared definitions of data stores, reports, and so on. Users typically do not log into the server directly.
Pentaho Server Repository	Database used by Pentaho Server to store definitions and contexts/preserve states necessary to run the server.
Pentaho Client Tool	Installed on client equipment like desktops or laptops, where users can interact directly with the tools, allowing users to create reports, define scheduled jobs, and perform other actions. Client tools can also interact with the server to do tasks like store report definitions or schedule background jobs.

Best Practices and Recommendations for AWS

The following methods are our recommended best practices for deploying the Pentaho Server, Server Repository, and client tools to AWS EC2.

- Running Pentaho Components in AWS
- Virtual Networks
- Pentaho Repository Database
- Pentaho Server VM and Storage
- Server and Repository Maintenance

Running Pentaho Components in AWS

You can create maximum usability and efficiency by following these recommendations:

- **Run Pentaho Server in EC2**: Pentaho Server should be deployed to an EC2 virtual machine (VM) if you already use EC2 and your data sources to be accessed by Pentaho are in AWS.
- **Run Pentaho Repository Database in RDS**: The Pentaho Repository database should be deployed to Relational Database Services (RDS), optionally to an EC2 VM. See the Pentaho Repository Database section for further details.
- **Do Not Run Pentaho Client Tools in AWS**: Users interact directly with the Pentaho client tools, typically through a GUI, so latency can cause issues. Therefore, local installation on the user's workstation itself, or on a server that has minimal network latency for the user can increase the speed of this interaction, since users' workstations are usually not in EC2 itself.

Virtual Networks

Use a separate Virtual Private Cloud (VPC) to host the Pentaho products, sorted by development lifecycle stage. For example:

- **VPC 1**: Pentaho Production
- **VPC 2**: Pentaho Development
- **VPC 3**: Your business applications
- **VPC…*n***: Any other needs

Using this method simplifies firewall design and provides separation of concerns and of duties.

We recommend you pair the Pentaho VPC with each VPC that contains a data source or application with which Pentaho needs to communicate if your VPCs are in the same region. Configure the VPCs to intercommunicate using VPN or a corporate network as outlined in Amazon's Multiple Region Multi-VPC Connectivity guide if your VPCs are in different regions.

Extending Corporate WAN to include AWS

We recommend you create a virtual network between the WAN and AWS if Pentaho needs access to any resources on your corporate wide-area network (WAN). Although Pentaho traffic can be plainly routed over the public internet, this has the following disadvantages:

Table 2: Disadvantages of Using Public Internet

Feature	Disadvantages of Using Public Internet
Service Level Agreements (SLA)	On a local network, bandwidth availability, low latency, and connectivity cannot be guaranteed by AWS.
Security	Not all protocols used during data transfer are encrypted.
Complexity	All customer sites and all applicable endpoint devices must be configured to talk over the internet to Pentaho in AWS. This includes routing and firewall configuration and maintenance.
Throughput	Bandwidth and latency are limited by the internet.

There are two methods to connect the corporate WAN with AWS:

1. **Site to site virtual private network (VPN) connection**: This addresses the security concern by ensuring that all traffic to and from Pentaho and the corporate WAN is IPSec encrypted.
2. **Direct connect**: This addresses the SLA, security, and throughput concerns. If it is designed correctly, it also addresses complexity concerns.

Network Security Groups and Subnets

- For management simplicity and security transparency, subnet Network Security Groups (NSGs) should be used to control traffic to and from the Pentaho servers.
- Place the Pentaho Server in its own NSG and subnet.
- Place the Pentaho Repository in its own NSG and subnets.
- Open only the minimum number of ports, depending on the protocols being used to communicate with your applications and data sources, and from the Pentaho Server to the Repository database. Which specific port numbers to open depends on your database product.

Figure 1 shows the recommended high-level network architecture:

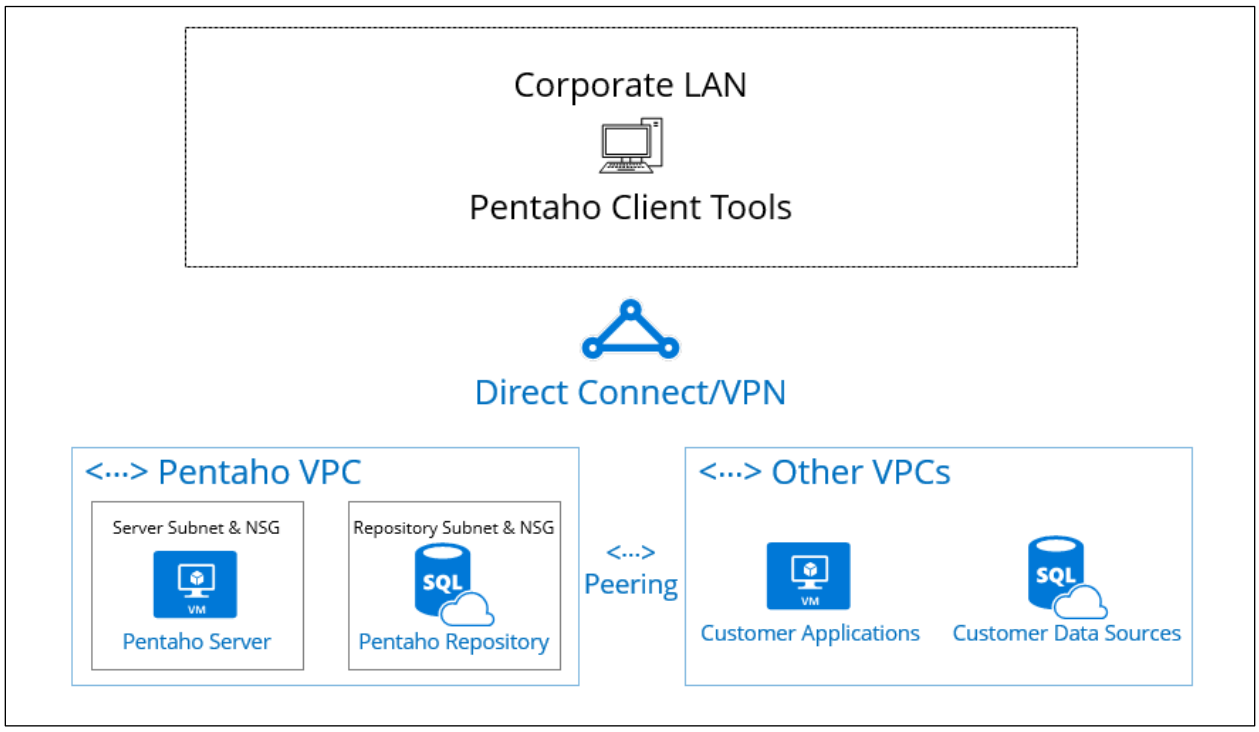

Figure 1: Recommended Network Architecture for Pentaho and AWS

Pentaho Repository Database

Although the Pentaho Repository database can be deployed to an EC2 VM, deploying it in AWS RDS can reduce complexity and cost.

It should be noted that Pentaho does not support Amazon Aurora or MariaDB. The Components Reference in Pentaho Documentation has a complete list of supported database repository types.

Configure the RDS servers (capacity, availability, and performance) as part of the infrastructure architecture, as if you were deploying to an on premises server, or to an EC2 instance.

Engage a network architect to configure the network aspects of RDS, as it can become complex especially if using multi-AZ. There are no Pentaho-specific network requirements of the RDS network configuration; it only needs to be able to reach the database on the database product's listening ports.

Pentaho Server VM and Storage

This section applies to designing the EC2 resources that will host Pentaho Server. It does not specify the infrastructure design but does highlight specific server and storage configurations and design fragments that you should incorporate into your overall infrastructure design to allow for the best hosting of Pentaho.

Pentaho's Components Reference contains details on installing Pentaho products and what specifics are required. Pentaho being deployed in EC2 does not affect this list, so choose an operating system that fits best with your overall infrastructure architecture.

EC2 Considerations

An infrastructure architect should work with Pentaho personnel to appropriately size the Pentaho Server virtual machines to ensure adequate performance, capacity, and availability.

Make sure that you have automatic, routine snapshots set up for running EC2 production servers.

EC2-specific concepts and restrictions to be incorporated are:

- **Instance Types**
  - The general purpose M4 and M3 instance types are known to work well with Pentaho servers.
  - Consider the C4, C3, or X1 classes, respectively, if your application has special CPU or memory demands.
  - Choose Hardware Virtual Machine (HVM) instances rather than Paravirtual (PV) ones.
- **Availability**
  - Pentaho products can be deployed to provide High Availability (HA) and Disaster Recovery (DR). As the designs and mechanisms to achieve this are not specific to EC2, you can apply your corporate virtual machine HA and DR standards to Pentaho servers.
  - Pentaho Server should be deployed in an active/passive HA configuration. More information is available at Best Practices Library – Pentaho in High Availability.

Server and Repository Maintenance

These maintenance recommendations will help you keep your Pentaho installation running at peak performance. There are no specific backup or monitoring requirements for Pentaho on AWS, so approach these topics as you would if AWS were not involved.

- **Backups**
  - You should back up the VMs that are run by the Pentaho server products using the same mechanism as your other EC2 VMs.
  - The **RDS repository database** should be backed up using RDS's backup mechanism. There are no RDS backup requirements that are specific to Pentaho on RDS.
  - More information on backing up Pentaho databases can be found at Best Practices – Backup and Recovery.
- **Monitoring**
  - The Pentaho Server and Repository should be monitored in the same way that other VMs and RDS databases are monitored. More information on monitoring Pentaho's processes in environments such as AWS is available at Best Practices – Logging and

Monitoring for Pentaho Servers.

Known Issues

The following issues apply to Pentaho products running in AWS. Make sure to account for them in your infrastructure and network architecture.

Repository Database – Aurora

The Pentaho Server repository database is not currently supported on AWS's Aurora product.

Workaround

Use one of the recommended RDS database products listed in the Components Reference.

Elastic Load Balancer (ELB) and Auto Scaling Groups

Pentaho Servers can run in active/passive load balanced mode, where multiple Tomcat servers run copies of Pentaho Server, but only one is active at any one time. However, ELB Auto Scaling Groups have not been tested as balancers for this design pattern.

Workaround

If you wish to run the Pentaho Server in active/passive mode, use Apache Web Server to load balance incoming traffic as in Best Practices Library – Pentaho in High Availability.

Related Information

Here are some links to information that you may find helpful while using this best practices document:

- Amazon: Multiple Region Multi-VPC Connectivity
- Best Practices – Backup and Recovery
- Best Practices – Logging and Monitoring for Pentaho Servers
- Best Practices Library – Pentaho in High Availability
- Pentaho Components Reference
- Pentaho Installation Instructions

Finalization Checklist

This checklist is designed to be added to any implemented project that uses this collection of best practices, to verify that all items have been considered and reviews have been performed.

Name of the Project:_____

Date of the Review:_____

Name of the Reviewer:_____

Item	Response	Comments
Did you have a network architect design the local network for Pentaho products?	YES_____ NO_____	
Did you use separate VPCs to host the Pentaho products?	YES_____ NO_____	
Did you use NSGs to control traffic for Pentaho servers for management simplicity and security transparency?	YES_____ NO_____	
Did you deploy the Pentaho Database repository in AWS RDS, rather than in EC2 VM?	YES_____ NO_____	
Did you engage a network architect to configure network aspects of RDS to reduce complexity when using a multi-AZ?	YES_____ NO_____	
Did you choose an OS that best fits your overall infrastructure architecture?	YES_____ NO_____	
Did you deploy the Pentaho Server in an active/passive HA configuration?	YES_____ NO_____	
Did you use the backups necessary for server and repository maintenance?	YES_____ NO_____	