

Tableau Server on AWS

Deployment Guidelines and Best Practices

July 2015



© 2015, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Notices

This document is provided for informational purposes only. It represents AWS's current product offerings and practices as of the date of issue of this document, which are subject to change without notice. Customers are responsible for making their own independent assessment of the information in this document and any use of AWS's products or services, each of which is provided "as is" without warranty of any kind, whether express or implied. This document does not create any warranties, representations, contractual commitments, conditions or assurances from AWS, its affiliates, suppliers or licensors. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Contents

Abstract	4
Introduction	4
Tableau Server	4
Getting Started	6
Using the AWS Marketplace	7
Installing from Tableau	8
Securing Tableau Server on AWS	9
Network	10
Access	12
Data	13
Scaling Tableau Server on AWS	15
Failover	16
Load Balancing	17
High Availability	18
Deploying Tableau Server on AWS	19
Conclusion	27
Contributors	27
Notes	27

Abstract

Amazon Web Services (AWS) provides a reliable, scalable, secure, and highly performing infrastructure for the most demanding web applications—an infrastructure that matches IT costs with customer traffic patterns in real time.

As an enterprise business intelligence platform, Tableau Server provides comprehensive and robust capabilities for all aspects of business intelligence. These capabilities are quick and easy to deploy, and require minimal, if any, customizations. The core architecture of Tableau Server runs two of the largest data platforms in the world: Tableau Online and Tableau Public.

This whitepaper provides technical guidance on how to deploy and configure Tableau Server on AWS. Additionally, it outlines key integrations and configurations unique to AWS products that enable you to use Tableau Server to best meet your needs.

Introduction

Organizations need to ensure that they can keep up with rapid changes that are required for global computing infrastructures. Additionally, they need to find ways to deploy and deliver applications from distributed, cloud-based services with the confidence that these applications can deliver a consistent and reliable level of service, and can withstand significant, unpredictable spikes in traffic without missing a beat. The compute resources offered by AWS meet these needs by providing a global computing infrastructure and services that simplify managing that infrastructure.

Tableau Server

Tableau Server provides business intelligence for organizations of all sizes, accessible through a browser or a mobile device. Tableau Server is built to simplify sharing and collaborating on interactive data visualizations by offering the following advantages:

- **Simple User Interface** – Tableau Server makes it easy to find, explore, and interact with analytic dashboards for every type of user. Powerful search capabilities and intuitive navigation controls make discovering content, users, and data sources straightforward.
- **Flexible data architecture** – If you have a fast database, Tableau Server can leverage that speed by maintaining live query connections back to that database. Alternatively, you can use Tableau Server to take in-memory snapshots of a data source (called *extracts*) and physically host that data on the Tableau Server platform.
- **Automatic data and content updates** – Tableau Server can refresh in-memory data extracts based on a set schedule, at specified intervals, or at incremental levels. You can also set alerts when data connections fail or use subscriptions to receive regularly scheduled emails about dashboards and reports.
- **Embedded analytics** – With Tableau Server, you can rapidly embed interactive dashboards within your organization’s existing web portals. Built-in sharing capabilities quickly provide HTML snippets that you can use to place Tableau Server views directly into webpages, SharePoint portals, intranet wikis, and so on.
- **Scalable** – Tableau Server scales with both hardware and memory to support a growing organization. Flexible content management, user permissions, and detailed administrative capabilities make managing a growing Tableau Server platform a straightforward process.
- **Secure** – Tableau Server gives you security permissions at any level you need. With multi-tenancy, you can create multiple sites on the server to separate users and content. You can set individual permissions for projects, dashboards, or even users.
- **Mobile** – You can view a dashboard from anywhere, on any device. All dashboards are automatically optimized for mobile tablets with touch-sensitive UI without requiring any additional authoring or configuration.

In the next section, we’ll discuss some of the AWS services and how you can use them to run Tableau Server in the AWS cloud.

Getting Started

Amazon Elastic Compute Cloud (Amazon EC2) provides resizable compute capacity in the cloud. Viewed from the framework of a traditional infrastructure, Amazon EC2 represents the servers that run your applications, web servers, databases, and, in this case, Tableau Server.

Tableau Server system requirements vary based on many factors, including the number of users and target workload.

Tableau Server is available in both 32-bit and 64-bit versions on AWS, but we recommend using a 64-bit version, which is available in the [AWS Marketplace](#).¹ For custom installations, we recommend using one of the following Microsoft Windows 64-bit operating systems:

- Windows Server 2008/Windows Server 2008 R2
- Windows Server 2012/Windows Server 2012 R2

Tableau Server infrastructure requirements are based on the number of users:

Users	CPU	RAM
< 25	4-core	8 GiB
< 100	8-core	32 GiB
> 100	16-core	64 GiB

Note These requirements are provided at the time of document publication and are subject to change. It is highly advised to always check the most recent technical specifications for Tableau Server.²

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Each instance type includes one or more instance sizes, which allow you to scale your resources to the requirements of your target workload. The following table

Tableau doesn't recommend running Tableau Server on a 4-core, 8-GiB machine except for proof of concept studies or really simple tasks.

shows the recommended Amazon EC2 instance types that map to Tableau Server infrastructure requirements:

Users	Instance	vCPU*	RAM (GiB)
< 25	m3.2xlarge	8	30
	c3.2xlarge	8	15
	r3.2xlarge	8	61
< 100	c3.4xlarge	16	30
	r3.4xlarge	16	122
> 100	c3.8xlarge	32	60
	r3.8xlarge	32	244

* Each vCPU is a hyper-thread of an Intel Xeon core for the M3, C3, and R3 instance families. You shouldn't equate a vCPU to a core. Instead, use the operating system to determine the number of cores available. Tableau Server also uses the Windows OS to determine the number of cores, for core-based licenses.

Using the AWS Marketplace

The easiest way to get started is to use the Tableau Server Amazon Machine Image (AMI), which is available in the AWS Marketplace. The Tableau Server AMI is pre-packaged and configured for deployment in a typical environment on AWS.

1. In the [AWS Marketplace](#), search for **Tableau**.
2. Select the Tableau Server product you want to use. When prompted, sign in to your AWS account.
3. Review the product details, and then choose **Continue** when you are ready to deploy Tableau Server on AWS.
4. On the AMI details page, choose the **1-Click Launch** tab.

You can configure all the settings for your instance on this tab. This option allows you to launch only one Amazon EC2 instance at a time.

Note If you prefer to use the Amazon EC2 console to launch your instance, click the **Manual Launch** tab from the AMI details page. This option allows you to launch multiple Amazon EC2 instances.

5. In the **Region** section, select the AWS region where you want to launch your Tableau Server instance.
6. In the **EC2 Instance Type** section, choose your preferred instance type. See the previous section for recommendations.
7. In the **VPC Settings** section, choose the Amazon Virtual Private Cloud (Amazon VPC) that your instance will be launched in.
8. In the **Security Group** section, review the proposed security group settings and either choose from your existing security groups or accept the new proposed security group.
9. In the **Key Pair** section, choose a key pair to associate with the instance. This key is used to connect to your instance over Secure Shell (SSH).
10. Choose **Accept Terms & Launch with 1-Click** to create your new instance.

For step-by-step information about using Tableau Server after launching it successfully from the AWS Marketplace, see <http://www.tableau.com/Amazon-Web-Services/Marketplace>.

Installing from Tableau

You can also download Tableau Server from <http://www.tableausoftware.com/products/server>, and install it on one of the recommended Amazon EC2 instances.

If you're installing Tableau Server in this manner, you'll need to choose the right storage for use with Amazon EC2. **Amazon Elastic Block Store (Amazon EBS)** provides persistent block-level storage volumes for use with EC2 instances in the AWS cloud. You should attach a separate Amazon EBS General Purpose

(SSD) volume with a volume size of at least 100 GiB for installing Tableau Server. This volume size supports a baseline performance of 300 IOPS with the ability to burst to 3,000 IOPS. For more information, see [I/O Credits and Burst Performance](#) in the Amazon EBS product documentation.³

For heavy workloads where the storage subsystem must provide an absolute minimum number of IOPS for performance, you can either create a larger Amazon EBS General Purpose (SSD) volume to get up to 10,000 IOPS with the ratio of 3 IOPS per GiB, or use an Amazon EBS Provisioned IOPS (SSD) volume with the IOPS you need. In either case, we recommend that you use Amazon EBS-optimized instances if you plan to run heavy extracts and processing.

Refer to the [Improve Server Performance](#) section in the Tableau Server documentation for comprehensive information to help you improve performance.⁴

If you're observing high I/O latency, changing the disk driver performance setting to **Better Performance** may give you better results.

Securing Tableau Server on AWS

AWS provides security features that Tableau Server can use to protect your environment, including the following:

- **Amazon Virtual Private Cloud (Amazon VPC)** adds another layer of network security to your environment by creating private subnets.
- **Security Groups** are like built-in firewalls that allow inbound and outbound connections to your network.
- **AWS Identity and Access Management (IAM)** allows specific control over access levels.
- **AWS Direct Connect** allows a dedicated network connection from a corporate network to AWS using industry-standard 802.1Q VLANs.

Enterprise application security has three main components: network, access, and data. In the next sections, we'll look at how you can implement these capabilities in AWS and Tableau Server to enable a single report or dashboard to securely

serve the needs of a broad and diverse user base, including both internal and external users.

Network

Network security for Tableau Server in AWS relies on the use of Amazon VPC security groups with SSL for securing internal and external communications.

Amazon VPC

An Amazon VPC is a distinct, isolated network within the cloud; network traffic within each Amazon VPC is isolated from all other Amazon VPCs. Using an Amazon VPC allows you to create your own network subnets and divide application layers into network subnets for a greater level of control. We recommend that you install and run Tableau Server in a separate subnet within your Amazon VPC so you can configure the network accordingly for access to Tableau Server and other data sets. Figure 1 shows a typical installation of Tableau Server in an Amazon VPC.

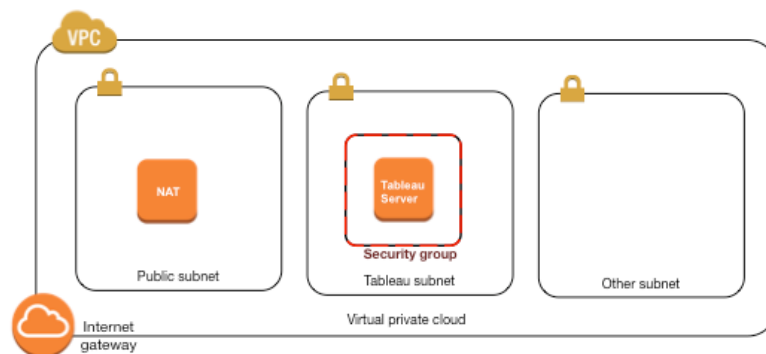


Figure 1: Amazon VPC Network Setup

Security Groups

You can define what types of network traffic can access Tableau Server by using security groups. Amazon EC2 security groups act as a firewall that governs network traffic into and out of Amazon EC2 instances. You can define and assign security groups that are appropriate for your Amazon EC2 instances. By default, EC2 instances are launched with security groups that are in an “allow nothing” state for inbound traffic, but you can make changes to allow the appropriate inbound traffic to the EC2 instance.

Here are the minimum requirements for connections to Tableau Server:

- Connection via RDP (port 3389) using a Remote Desktop client to access and manage the instance and services
- Standard web traffic via HTTP (port 80) and HTTPS (port 443), to view content hosted on, and to publish to, Tableau Server
- Communication between Tableau Server components (see the [Scaling Tableau Server on AWS](#) section later in this paper)

Based on these requirements, the security group for your Tableau Server deployment should enable only three standard ports for inbound traffic: HTTP 80, HTTPS 443, and RDP 3389 for restricting access to your instance. You should also limit remote access (port 3389) from a few hosts, and also limit HTTP and HTTPS traffic to hosts within your corporate network or to a trusted set of clients.

Client Access

By default, Tableau Server uses standard HTTP requests and responses. Tableau Server can be configured for HTTPS (SSL) with customer-supplied security certificates. When Tableau Server is configured for SSL, all content and communications between clients are encrypted and use the HTTPS protocol.

When you configure Tableau Server for SSL, the browser and SSL library on the server negotiate a common encryption level. Tableau Server uses OpenSSL as the server-side SSL library, and is pre-configured to use currently accepted standards. Each web browser that accesses Tableau Server via SSL uses the standard SSL implementation provided by that browser.

Internal Communication

There are two aspects to communications between Tableau Server components in a distributed server installation: trust and transmission. Each server in a Tableau Server cluster uses a stringent trust model to ensure that it is receiving valid requests from other servers in the cluster. Trust is established by a whitelist of IP addresses, ports, and protocols. If any of these are invalid, the request is ignored.

Computers in the cluster that are running a gateway process accept requests from third parties (clients), unless a load balancer receives the requests. Servers that

aren't running a gateway process accept requests only from other trusted members of the cluster. With the exception of license validation and repository access, transmission of all internal communication is performed via HTTP.

Refer to the [Tableau Server Administrator Guide](#) for more information about configuring Tableau Server with SSL and enabling trust.⁵

Access

Access security is used to establish the user's identity, to prevent unauthorized access, and to personalize each user's experience. Tableau Server supports three types of authentication:

- **Security Assertion Markup Language (SAML)** – Tableau Server can be configured to use SAML for single sign-on (SSO). In this model, an external identity provider (IdP) authenticates the user's credentials, and then sends a security assertion to Tableau Server with information about the user's identity.
- **Kerberos** – If Tableau Server is configured to use Windows Active Directory authentication and Kerberos is enabled, users can gain access to Tableau Server based on their Windows identities.
- **Trusted authentication** – In trusted authentication, you set up a trusted relationship between Tableau Server and one or more web servers. When Tableau Server receives requests from a trusted web server with a redeemable token or ticket, it assumes that the web server has already handled authentication as necessary.

Tableau Server also provides a personalized experience for users by creating an account for each named user of the system. For more information, see the [Authentication section](#) of the *Tableau Server Administrator Guide*.⁶

AWS Directory Service

AWS Directory Service is a managed service that allows you to connect your AWS resources to an existing on-premises directory such as Microsoft Active Directory (with AD Connector), or to set up a new, stand-alone directory in the AWS cloud (with Simple AD). Connecting to an on-premises directory is easy, and once this connection is established, all users can access AWS resources and applications with their existing corporate credentials.

Using the AWS Directory Service, you can choose to use Active Directory-based authentication instead of local authentication. (Local authentication creates users and assigns passwords using Tableau Server’s built-in user management system.) To set up Active Directory-based authentication, in the configuration step after installing Tableau Server, you must choose Active Directory. It is not possible to switch between Active Directory and local authentication later.

The Active Directory authentication model uses the Microsoft Security Support Provider Interface (SSPI) to sign in your users automatically, based on their Windows user name and password. This creates an experience similar to single sign-on (SSO).

You can also use AWS Directory Services to run Tableau Server in distributed mode with failover support and high availability. Distributed mode requires all the computers in a cluster to be members of the same domain. The Run As User account, which is specified on the primary Tableau Server, must be a domain account in this same domain.

Data

Tableau Server uses native drivers (relying on a generic ODBC adapter when native drivers are not available) to connect to databases whenever possible, for processing result sets, for refreshing extracts, and for all other communications with the database. You can configure the driver to communicate on non-standard ports or use transport encryption, but this type of configuration is transparent to Tableau Server. However, since the Tableau Server-to-database communication is typically behind a firewall, most customers choose not to encrypt this communication.

Connecting to Data Stores in AWS

You can launch AWS resources, such as Amazon Relational Database Service (Amazon RDS), Amazon Elastic MapReduce (Amazon EMR), or Amazon Redshift, into an Amazon VPC. By placing the Tableau Server into the same Amazon VPC as your data stores, you can ensure that your traffic never leaves the Amazon VPC.

You can use subnets with security groups to launch your resources into different layers but allow them to communicate securely within an Amazon VPC, as illustrated in Figure 2.

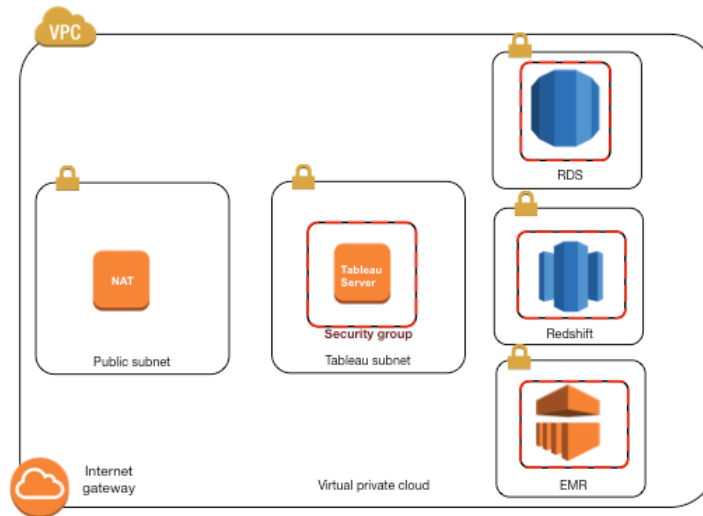


Figure 2: Amazon VPC Network Setup with Data Stores

Connecting to Data Stores Outside of AWS

You can optionally connect your Amazon VPC to your own corporate data center by using an IPsec hardware VPN connection, thus making the AWS cloud an extension of your data center. A VPN connection consists of a virtual private gateway attached to your Amazon VPC and a customer gateway located in your data center.

AWS Direct Connect is a network service that provides an alternative to using the Internet to utilize AWS cloud services. AWS Direct Connect lets you establish a dedicated network connection by using industry-standard 802.1Q VLANs. You can use the same connection to access public resources (such as objects stored in Amazon Simple Storage Service [Amazon S3] using public IP address space) and private resources (such as Amazon EC2 instances running within an Amazon VPC using private IP space), while maintaining network separation between the public and private environments.

Scaling Tableau Server on AWS

Tableau Server is architected to scale up with more CPU cores and memory, and scale out when you add servers. This architecture allows you to maximize the use of compute resources while giving you the ability to scale massively.

With single-server deployment on AWS, all Tableau Server processes run on a single instance, as shown in Figure 3. This is the most basic configuration for Tableau Server.

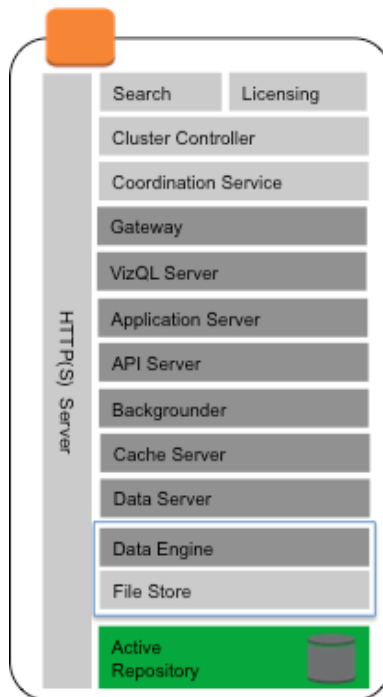


Figure 3: Tableau Server Components on a Single Amazon EC2 Instance

To build in redundancy, you need to add additional servers, Tableau workers (which host copies of the repository), data engines, and other processes. In this model, the primary server should be isolated in its own node, and ideally should run as few of the server processes as possible.

Although you can split an 8-core license on two 4-core machines, we recommend that you scale in increments of 8-core machines only.

The next section provides guidelines for failover support and high availability for this basic configuration.

Failover

To configure a cluster that provides failover support for the data engine and repository processes, you need at least two instances: one for the primary Tableau Server and one for a Tableau worker. The recommended failover configuration requires three instances: one for the primary Tableau Server and two for the workers.

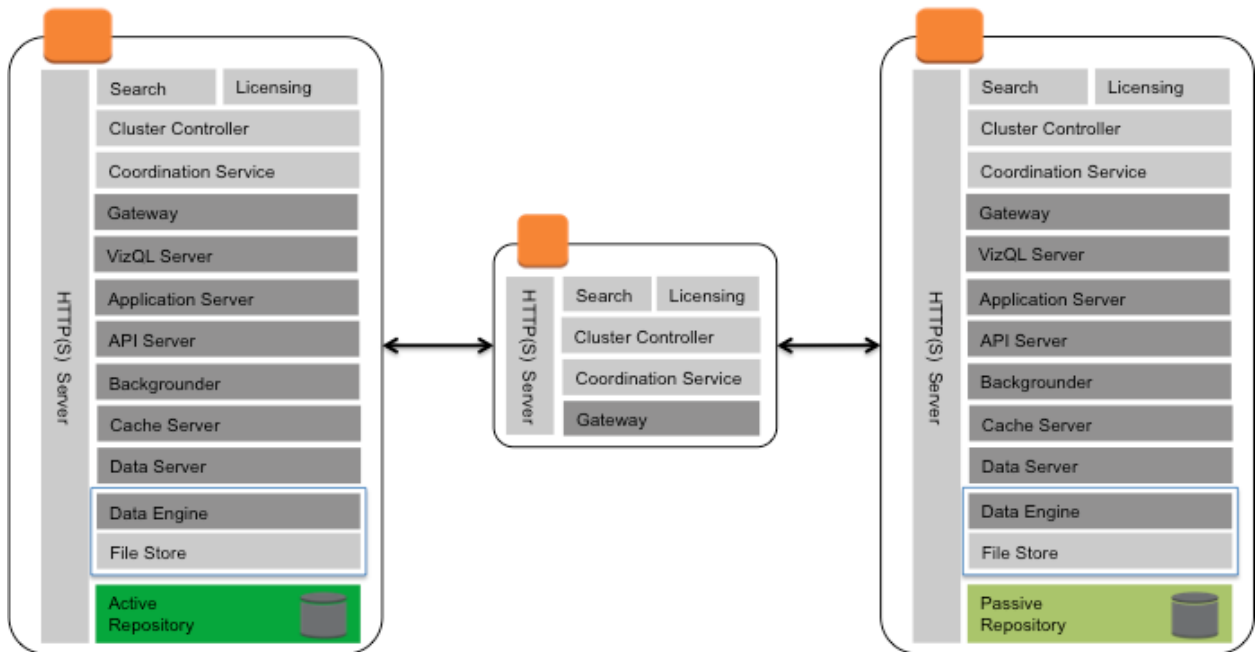


Figure 4: Tableau Server in Failover Mode

An ideally configured distributed Tableau Server environment status page should look similar to Figure 5.

Server Status

Process Status
The real-time status of processes running in Tableau Server.

Process	Primary 10.32.139.21	Worker 1 10.32.139.22	Worker 2 10.32.139.30
Cluster Controller	✓	✓	✓
Gateway	✓	✓	✓
Application Server		✓	✓ ✓
API Server		✓	✓
VizQL Server		✓	✓ ✓
Cache Server		✓	✓ ✓
Search & Browse	✓	✓	✓
Backgrounder		✓	✓
Data Server		✓	✓
Data Engine		✓	✓
File Store		✓	✓
Repository		✓	⊘

Refresh Status ✓ Active ⌛ Busy ⊘ Passive ⚠ Unlicensed ✖ Down ⊘ Status unavailable

Figure 5: Status Page of Tableau Server in Failover Mode

In this deployment, the data engine and repository processes have been moved from the primary to a worker, and the primary is running only the gateway process along with Search & Browse and Cluster Controller. Furthermore, only the repository component is in a passive state, and is kept up to date using replication; all other components are active-active. In the event of a failover, the passive repository becomes active, and your deployment continues to function.

Load Balancing

On AWS, **Elastic Load Balancing** automatically distributes incoming application traffic across multiple Amazon EC2 instances in the cloud. It enables you to achieve greater levels of fault tolerance in your applications, and seamlessly provides the required amount of load balancing capacity needed to distribute application traffic.

You can use Elastic Load Balancing to distribute requests across multiple gateways in a Tableau Server cluster. At this point, as illustrated in Figure 6, all three nodes have gateways, which are used to route requests to available server processes. Unlike the repository process, there aren't any passive or standby gateway processes—all gateways are active. When you add a load balancer to a

Tableau Server cluster, the URL that's accessed by Tableau Server users belongs to the load balancer, not to the primary Tableau Server.

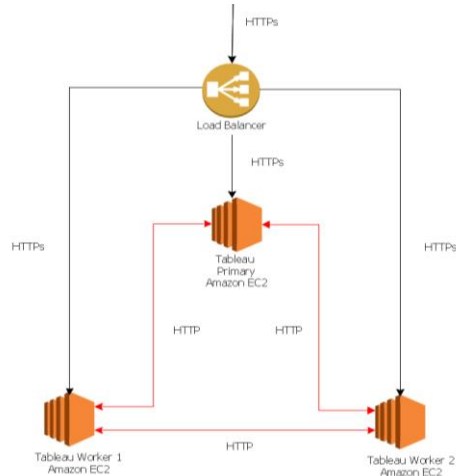


Figure 6: Tableau Server with Elastic Load Balancing

High Availability

Now that you have provided redundancy for the data engine, repository, and gateway by adding multiple workers, you can additionally build redundancy for the primary Tableau Server. You can do this by creating a backup of the primary Tableau Server. Although the backup primary needs to be licensed during installation, it does not count as one of the three environments allowable under the Tableau EULA.

To configure for high availability, you need to run a failover cluster plus an additional computer as the backup primary for your primary Tableau Server. If you configure for high availability, the primary Tableau Server and the backup primary may be running few or no Tableau Server processes.

Tableau Server checks for licenses at 8-hour intervals. If the primary fails, you have up to 8 hours (depending on the time of the last licensing check) to bring the backup primary online. For example, if the licensing check occurred 7 hours and 50 minutes ago, you have 10 minutes to bring the backup primary online. If the licensing check occurred 1 minute ago, you have 7 hours and 59 minutes.

Deploying Tableau Server on AWS

In this section, we’ll walk through the steps of deploying a Tableau Server cluster on AWS. We’ll also show how to use some of the services and Tableau Server features mentioned in previous sections for deploying Tableau Server in a highly available and scalable configuration on AWS.

Before following the deployment steps, make sure that you have access to an AWS account, Tableau Server 9.0 installation files, and a valid product key based on named users or a minimum of 16 cores for core-based licenses.

Step 1: Create a VPC Configured for High Availability

The following steps assume that you’ve created an Amazon VPC in the AWS cloud with at least four subnets (two public and two private) in different Availability Zones, as illustrated in Figure 7. For information about how to create a VPC with public and private subnets, see [Scenario 2: VPC with Public and Private Subnets](#) in the *Amazon VPC User Guide*.⁷

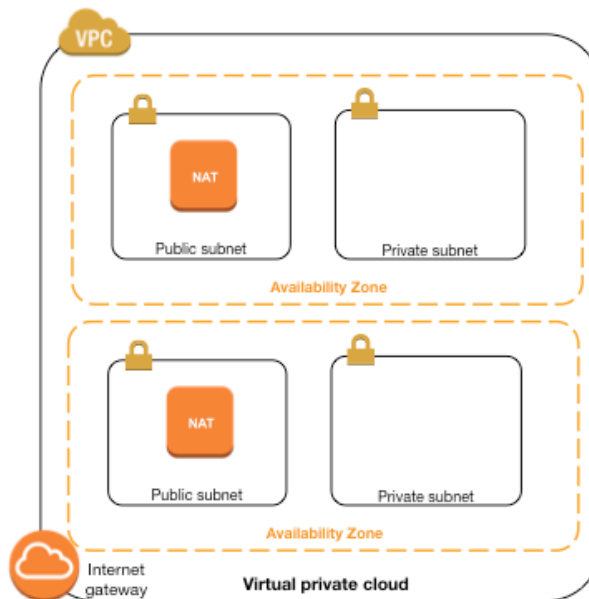


Figure 7: Amazon VPC Setup with Subnets for High Availability Deployment

After you've created the VPC, launch two Windows instances in both public subnets for use as Remote Desktop gateways. For further information on how to launch and connect to a Windows instance, see [Getting Started with Amazon EC2 Windows Instances](#) in the *Amazon EC2 User Guide*.⁸

Step 2: Create an AWS Directory Service for the VPC

Follow the steps in the [Creating a Directory with Simple AD section](#) of the AWS documentation to create a fully managed, Samba-based directory in the AWS cloud.⁹ When you create a directory with Simple AD, AWS Directory Service creates two directory servers and DNS servers on your behalf. The directory servers are created in different subnets in an Amazon VPC. This is done for redundancy so that your directory remains accessible even if a failure occurs.

Use the two private subnets available within your Amazon VPC to create the Simple AD so that you can run Tableau Server across Availability Zones.

Step 3: Deploy Three EC2 Instances in the VPC

Now deploy three Amazon EC2 nodes (m3.2xlarge instances with the latest Windows 2012R2 AMI) across two Availability Zones, as illustrated in Figure 8. We'll use one node as the primary server and the other two nodes as workers. The EC2 instances can be auto-joined to the Simple AD domain created in step 2 either via the console or by following the steps in [Joining a Windows Instance to an AWS Directory Service Domain](#) in the AWS documentation.¹⁰

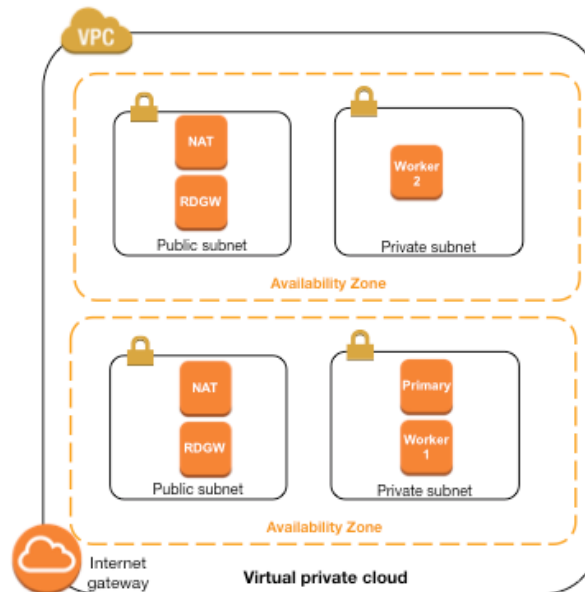


Figure 8: Amazon EC2 Instance for Deploying Tableau Server

After you have launched the instances, connect to them from one of the Remote Desktop Gateway (RDGW) instances by using the credentials provided for the AWS Directory Service, and then create a domain account for running Tableau Server by following the steps in the [Creating Users and Groups](#) section of the *AWS Directory Service Administration Guide*.¹¹

Step 4: Install and Configure Tableau Server

Now we'll install the Tableau Server on the EC2 instances we launched in step 3 and configure the instances as primary and worker servers.

Step 4a: Install the Tableau Server primary

On the Amazon EC2 instance tagged as *Primary*, follow these steps:

1. Download the Tableau Server installation file from the Tableau Customer Account Center. You can also follow the steps outlined in the [Tableau Knowledge Base article](#) on how to access installation files and product keys.¹²

Make sure to install the application into drive D instead of using the default path C:\Program Files\Tableau\Tableau Server.

2. Run the Tableau Server installation file, and follow the on-screen instructions to complete setup and install the application.
3. After the installation is complete, choose **Next** to open the **Product Key Manager** window and enter the product keys you obtained from the Tableau Customer Portal.
4. The **Tableau Server Configuration Utility** will be displayed. You can set configuration options now, before the server starts at the end of the installation process.
5. Configure options on the **General** tab:
 - a. Under the Network Service account, use the domain account created in step 3. The user name should include the domain name.

Figure 10: Choosing a Domain Account for Tableau Server

- b. Choose **Use Active Directory** to authenticate users on the server. You can also optionally select the **Enable automatic logon** check box.

Figure 11: Configuring Tableau Server to use Windows Authentication

6. If you want to set up the cluster to use SSL, follow these instructions in the *Tableau Server Administrator Guide*:
 - External SSL for clients
 - Internal SSL for server components
7. Stop Tableau Server on the primary node (see [Tableau Server Monitor](#) in the *Tableau Server Administrator Guide* to learn how¹³).

Step 4b: Install Tableau Server workers

On the Amazon EC2 instances tagged as *Worker1* and *Worker2*, follow these steps:

1. Download the Tableau Server Worker software from the Tableau Customer Account Center.
2. Run Tableau Server Worker Setup on all additional computers that you want to add to the Tableau Server cluster.
3. During installation, when asked to provide the primary server details, specify **tableau-primary.domain.com** as the name.

Step 4c: Configure Tableau Server in distributed mode

Once the worker software is installed on worker computers, return to the primary server and open the Configuration Utility by choosing **Tableau Server 9.0, Configure Tableau Server** on the Start menu.

1. In the Configuration Utility, enter your password on the **General** tab.
2. On the **Servers** tab, choose **Add** to add a worker server. Enter its IPv4 address or computer name. Enter **1** for each process. Select **Repository** and **Search & Browse**, but leave the **Gateway** check box cleared. Choose **OK**.

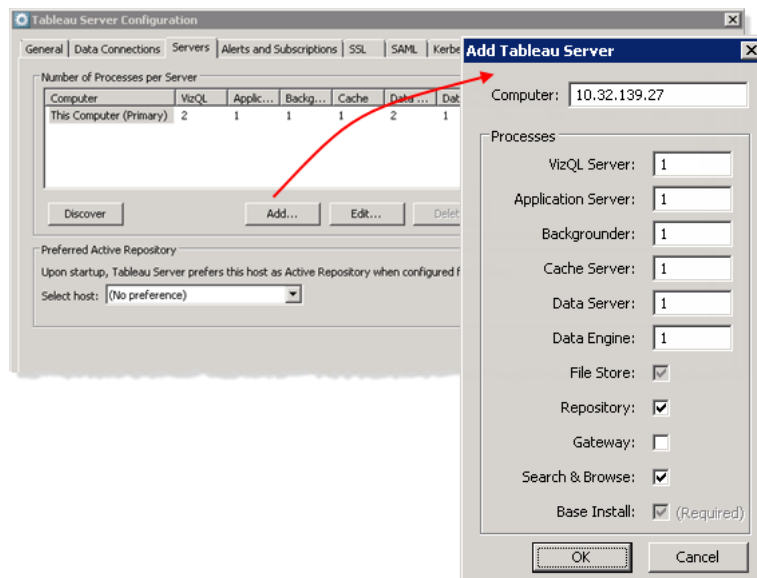


Figure 10: Setting up a Worker for High Availability Deployment

3. Choose **Add** to add a second worker server. Enter its IPv4 address or computer name. Enter **1** for every process except for **Data Engine** (set that to **0**). Leave the **Repository** check box cleared, but select the **Gateway** check box. Choose **OK**.
4. Still on the **Servers** tab, select **This Computer (Primary)** and choose **Edit**. Set every process to **0**, clear the **Repository** check box, but keep **Gateway** selected. Choose **OK**.
5. Choose **OK** again to close the Configuration Utility, and then start the primary server so your changes can take effect.
6. Stop the primary server and open the Configuration Utility. On the **Servers** tab, select the first worker and choose **Edit**. Select the **Gateway** check box. Choose **OK**.
7. Select the second worker and choose **Edit**. Set **Data Engine** to **1** and select the **Repository** check box. Choose **OK**, and then **OK** again to close the Configuration Utility. Start Tableau Server.

Step 5: Create a Load Balancer as a Front-End for the Tableau Server Cluster

Follow the steps outlined in [Getting Started with Elastic Load Balancing](#) in the AWS documentation to launch a load balancer within your VPC.¹⁴

1. In step 1 of the instructions, if you want the load balancer to be publicly accessible, select the two public subnets. Otherwise, select the **Create an internal load balancer** check box and choose the two private subnets. Figure 11 shows two public subnets being selected for the load balancer.

If you choose to expose the load balancer with a public endpoint, make sure that you configure Elastic Load Balancing with SSL, as explained in [Create an HTTPS Load Balancer](#) in the AWS documentation.¹⁵

This wizard will walk you through setting up a new load balancer. Begin by giving your new load balancer a unique name so that you can identify it from other load balancers you might create. You will also need to configure ports. Traffic from your clients can be routed from any load balancer port to any port on your EC2 instances. By default, we've configured your load balancer with a standard web server on port 80.

Load Balancer name:

Create LB Inside:

Create an internal load balancer: (what's this?)

Enable advanced VPC configuration:

Listener Configuration:

Load Balancer Protocol	Load Balancer Port	Instance Protocol	Instance Port
HTTP	80	HTTP	80

Select Subnets

You will need to select a Subnet for each Availability Zone where you wish traffic to be routed by your load balancer. If you have instances in only one Availability Zone, please select at least two Subnets in different Availability Zones for your load balancer.

VPC vpc-3b24b45e (10.0.0.0/16) | VPC

Available Subnets

Actions	Availability Zone	Subnet ID	Subnet CIDR	Name
<input type="checkbox"/>	us-west-2a	subnet-68a5cd0d	10.0.1.0/24	Private-2a
<input type="checkbox"/>	us-west-2b	subnet-65088612	10.0.3.0/24	Private-2b

Selected Subnets

Actions	Availability Zone	Subnet ID	Subnet CIDR	Name
<input checked="" type="checkbox"/>	us-west-2a	subnet-68a5cd0e	10.0.0.0/24	Public-2a
<input checked="" type="checkbox"/>	us-west-2b	subnet-7e088609	10.0.2.0/24	Public-2b

Figure 11: Load Balancer Setup with Public Subnets

- In step 2 of the instructions in [Getting Started with Elastic Load Balancing](#), ensure that your security group is configured to allow access on port 80 or 443 only, with source limited to hosts or ranges of hosts that will access Tableau Server.
- In step 4, you can specify the ping path as “/”.
- In step 5, select the Tableau Server instances and ensure that **Enable Cross-Zone Load Balancing** is selected so that the load balancer can load-balance the traffic across the instances in multiple Availability Zones.

When Tableau Server is deployed with Elastic Load Balancing without configuring the load balancer information, some URLs may refer to the instance itself, and not to the Elastic Load Balancer.

Alternatively, you can configure Tableau Server to work with a load balancer by performing the following steps.

- Create a subnet for your load balancer that has a CIDR block with a /27 bitmask, for 32 IP addresses.

The IP addresses provided by Elastic Load Balancing are dynamic, and Tableau Server needs a list of static IP addresses for this configuration. To make this work, we're creating a subnet with the smallest possible CIDR range so that the IP addresses the load balancer can have are limited to a finite set.

For the next steps we'll use the **tabadmin** command. This command enables you to perform administrative tasks from the command line on Tableau Server and installs with Tableau Server by default. For a general overview, see [How to Use tabadmin](#) in the Tableau Server documentation.¹⁶

2. In the Tableau Server bin directory, enter the following command, where *name* is the URL that will be used to reach Tableau Server through the load balancer:

```
tabadmin set gateway.public.host "name"
```

3. Enter the following command, where *server1*, *server2*, and so on are the IP addresses for the given CIDR range of subnets for Elastic Load Balancing:

```
tabadmin set gateway.trusted "server1,server2,...,server30"
```

4. Run the **config** command:

```
tabadmin config
```

5. Start the server so the changes can take effect.

Conclusion

Deployment of Tableau Server on AWS differs very little from deployment on traditional infrastructure. In this whitepaper, we presented a number of AWS platform considerations relating to security, storage, compute configurations, management, and monitoring that need to be considered to get the best out of your Tableau Server deployment on AWS. Following the best practices and guidelines provided in this paper will help you achieve optimal performance, availability, and reliability.

Contributors

The following individuals and organizations contributed to this document:

- Rahul Bhartia, Solutions Architect, Amazon Web Services
- Russell Christopher, Pre/Post Sales, Tableau Software
- Ken Chestnut, Global Ecosystem Lead, Amazon Web Services
- Ashley Jaschke, Manager of Product Management, Tableau Software
- Dustin Smith, Sr. Product Marketing Manager, Tableau Software

Notes

¹ https://aws.amazon.com/marketplace/seller-profile/ref=dtl_pcp_sold_by?id=oef849c5-c2fa-4699-9f67-c47555e148a5

² <http://www.tableau.com/products/techspecs#server>

³ <http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EBSVolumeTypes.html#IOcredit>

⁴ http://onlinehelp.tableau.com/current/server/en-us/help.htm#perf_extracts_view.htm

⁵ <http://onlinehelp.tableau.com/current/server/en-us/admin.htm>

⁶ http://onlinehelp.tableau.com/current/server/en-us/help.htm#security_auth.htm

- 7 http://docs.aws.amazon.com/AmazonVPC/latest/UserGuide/VPC_Scenario2.html
- 8 http://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/EC2Win_GetStarted.html
- 9 http://docs.aws.amazon.com/directoryservice/latest/adminguide/create_directory.html
- 10 <http://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/ec2-join-aws-domain.html>
- 11 http://docs.aws.amazon.com/directoryservice/latest/adminguide/creating_ad_users_and_groups.html
- 12 <http://kb.tableau.com/articles/knowledgebase/downloading-tableau-products>
- 13 <http://onlinehelp.tableau.com/current/server/en-us/help.htm#monitor.htm>
- 14 <http://docs.aws.amazon.com/ElasticLoadBalancing/latest/DeveloperGuide/elb-getting-started.html>
- 15 <http://docs.aws.amazon.com/ElasticLoadBalancing/latest/DeveloperGuide/elb-create-https-ssl-load-balancer.html>
- 16 http://onlinehelp.tableau.com/current/server/en-us/help.htm#tabadmin_overview.htm