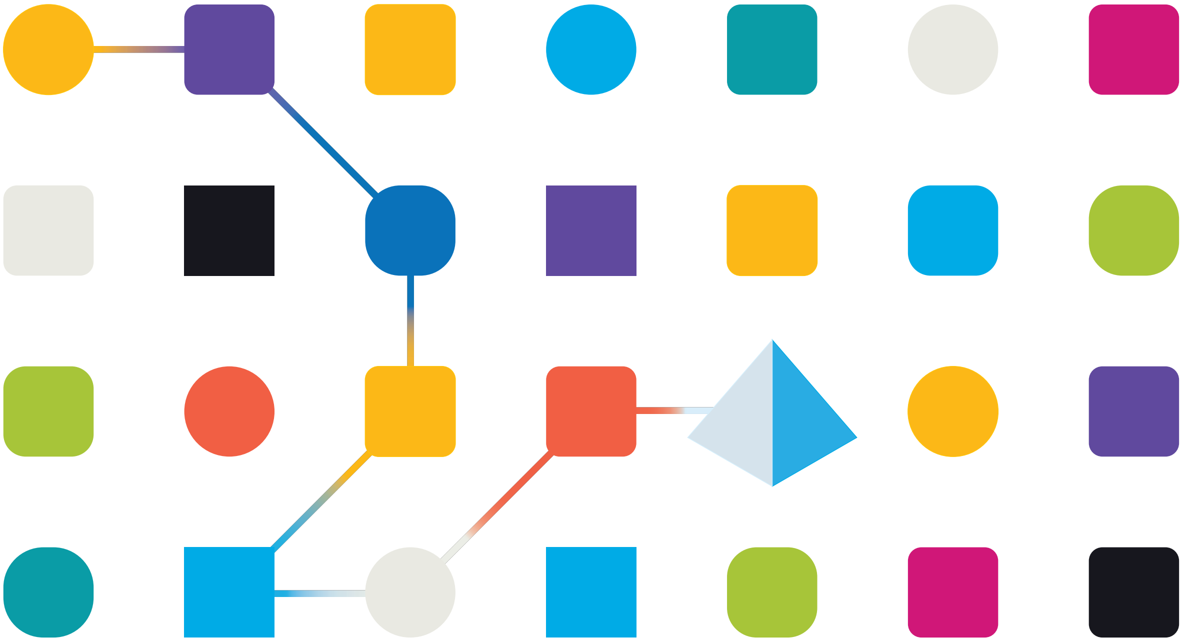
blueprism®

Blue Prism 6

AWS Reference Architecture Reference Guide

Document Revision: 1.0



Trademarks and Copyright

The information contained in this document is the proprietary and confidential information of Blue Prism Limited and should not be disclosed to a third-party without the written consent of an authorized Blue Prism representative. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying without the written permission of Blue Prism Limited.

All trademarks are hereby acknowledged and are used to the benefit of their respective owners.
Blue Prism is not responsible for the content of external websites referenced by this document.

Blue Prism Limited, 2 Cinnamon Park, Crab Lane, Warrington, WA2 0XP, United Kingdom.
Registered in England: Reg. No. 4260035. Tel: +44 370 879 3000. Web: www.blueprism.com

Contents

Introduction

The document provides an overview of the key considerations for an AWS based deployment of Blue Prism, along with reference architectures for the commonly requested patterns for a Cloud based deployment. The objective of this document is to explain the key considerations for deployment on AWS. A basic understanding of the AWS architecture is expected. The reference architectures contained within this document are based upon generalized assumptions and AWS design best practices and Reference Architectures. The architecture may need to be modified to suit a client deployment.

Intended Audience

This information is intended for use by system architects and designers who are seeking to gain an understanding of the options and considerations for deploying a Blue Prism environment in the Microsoft AWS Cloud.

AWS Services and Key Concepts

The following sections outline some of the relevant services and concepts that are key in designing an AWS based Blue Prism deployment.

AWS Services

The Blue Prism architecture uses several key AWS services and concepts. These are outlined below. Refer to the AWS documentation in the links for further information.

- Virtual Private Cloud – Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of the Amazon Web Services (AWS) cloud where you can launch AWS resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.
- Regions and Availability Zones - Amazon EC2 is hosted in multiple locations world-wide. These locations are composed of regions and Availability Zones. Each region is a separate geographic area. Each region has multiple, isolated locations known as Availability Zones. Amazon EC2 provides you the ability to place resources, such as instances, and data in multiple locations. Resources aren't replicated across regions unless you do so specifically.
- AWS Directory Service - AWS Directory Service provides multiple ways to set up and run Amazon Cloud Directory and Microsoft AD with other AWS services. AWS Directory Service for Microsoft Active Directory (Enterprise Edition), also known as Microsoft AD, enables your directory-aware workloads and AWS resources to use a managed Active Directory and enables Blue Prism to be integrated into a domain.
- Security Group - A security group acts as a virtual firewall that controls the traffic for one or more instances. When you launch an instance, you associate one or more security groups with the instance. You add rules to each security group that allow traffic to or from its associated instances. You can modify the rules for a security group at any time; the new rules are automatically applied to all instances that are associated with the security group.

AWS SQL Server RDS vs SQL Server on EC2

The selection of an SQL Platform as a Service (PAAS) based architecture is not just dependent on the sizing. The limitations (and advantages) of using a PAAS platform should also be understood. These are detailed fully here, however the main considerations in context of a Blue Prism deployment will be:

- **Size** – Primarily limited by the current maximum size of 4TB, as Blue Prism generates a high volume of database logging.
- **Speed of deployment** – Using PAAS will enable an environment to be initiated more quickly
- **Management and Control** – Managed platform vs full control over configuration and security
- **Cost** – Using a PAAS solution may be cheaper overall (if correctly sized)
- **Features** – Access to some features is limited in AWS RDS vs a full Infrastructure as a Service (IAAS) installation.

AWS Resource Configuration and Costing

Most of a Blue Prism deployment on AWS is made up of EC2 instances running as Blue Prism robots. For a successful, basic deployment the following AWS resources should be considered – this list is neither exhaustive nor definitive for a deployment:

- VPC – located in an appropriate region.
- EC2 instance(s) with SQL Server installed or an AWS RDS implementation.
- EC2 instance(s) with Blue Prism installed and configured acting as Blue Prism Application Servers.
- EC2 instance(s) with Blue Prism installed and configured acting as Blue Prism Clients.
- EC2 instances with Blue Prism installed and configured acting as Blue Prism Robots.

The resources listed above should be sized appropriately according to the deployment, customer requirements and Blue Prism best practices.

To ensure high availability of a Blue Prism deployment the resources listed should be spread across availability zones within the AWS region. Further information on this is contained within this guide.

Total Cost of Ownership (TCO) for Blue Prism infrastructure deployed on AWS can be calculated using the AWS pricing calculator. The cost displayed in the calculator is for the infrastructure only, for a true TCO value Blue Prism license costs should be included.

Blue Prism AWS Reference Architectures

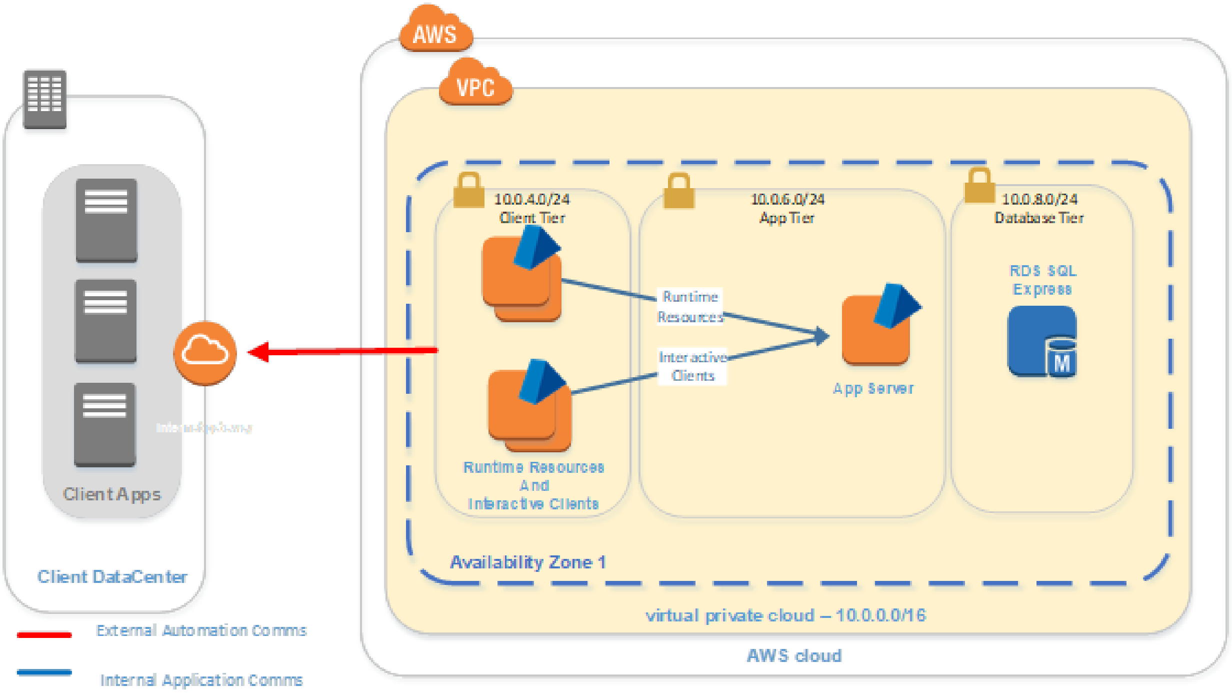
The following sections outline the core reference architectures for the expected deployment models for Blue Prism on the AWS cloud. These are supplemented by peripheral design considerations and scenarios, such as integration with Single Sign-On (SSO) in Supporting Architecture Patterns.

General Assumptions and Limitations:

- A connection is in place between the Blue Prism network and any external networks where the automated applications reside. This may be via a VPN, internal Vnet Peering (for cloud hosted applications) or AWS Direct Connect
- If the target applications are expected to be on the other end of the VPN / AWS Direct connect link, careful consideration must be given to routing and bandwidth across this connection
- Blue Prism clients and runtimes will need to have all necessary software installed to facilitate the automation of remote applications
- Beyond high level recommendations regarding segregation of components, network design is outside the scope of this document.
- The Authentication components are not shown within these sections. See Supporting Architecture Patterns for considerations.
- Access to an AWS account has been configured and suitable Windows licensing for constituent resources have been sourced
- Management of deployment keys, secrets and certificates are not included in this document as AWS account management is the responsibility of the deploying party.
- All resource specifications should be defined based on the recommendations in the Infrastructure Reference Guide, which is available from the Blue Prism Portal.

Small non-critical or POC Environment using SQL PAAS

This pattern makes use of the AWS RDS SQL Server Database Platform as a Service (PAAS) offering and is only suitable for small non-critical environments.

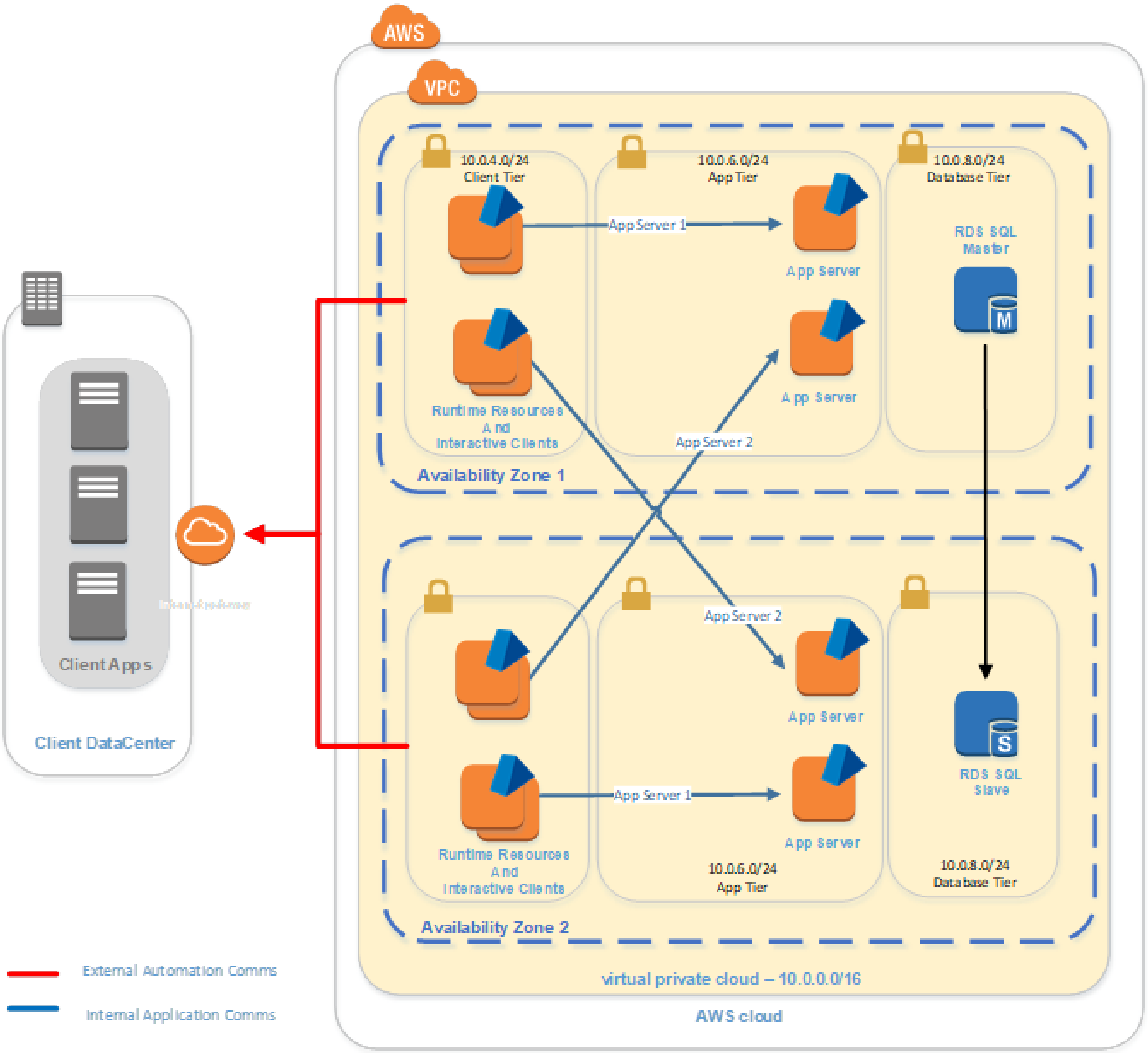


Key Design Considerations

- SQL Server Express may be used for POC environments to minimize cost. Other editions should be selected for anything but the most basic of environments.
- A single Application server is depicted here, however this may be increased to 2 or more as required

Small to Medium Scale Deployment using SQL PAAS with HADR

This pattern makes use of the AWS RDS SQL server Database Platform as a service (PAAS) offering and is designed to support HADR.



Key Design Considerations

- AWS RDS SQL Server is based upon standard editions and versions of the Microsoft SQL server platform. As such, consider which edition and version is appropriate, based on the overall requirements and latest release notes and recommendations for the Blue Prism platform.
- Standard or Enterprise Edition must be selected for a multi-AZ deployment.
- The database sizing recommendations for Blue Prism within the Infrastructure Reference Guide should be used as a baseline for selecting the appropriate specification.
- AWS SQL Server instances are sized based on “DB Instance Classes”. The selection of the appropriate instance class is very dependant on expected volumes and scale, however it is likely that (for production grade instances) a minimum of db.m4.2xlarge will be necessary. DB Instance Classes may be changed, however an outage would be incurred during any change in the class.
- Multiple availability zones and application servers are included in this design, to provide DR and high availability in event of a failure. A single AZ and / or application server may be deployed for small scale proof of concept or non business critical environments
- Amazon RDS only supports increasing storage on a SQL Server DB instance for General Purpose SSD or Provisioned IOPs SSD drives. If the RDS instance is created with Magentic disks, it is important to choose an appropriately sized tier, otherwise a migration to a new instance would be necessary if this size is not sufficient. Further documentation on Amazon RDS Storage can be found here and here.
- For critical processes, consider distributing workload between devices in both Availability zones. It is good practice to have Runtimes in each zone split between app servers in AZ1 and AZ2, to enable continuation of service in the event of planned maintenance in one AZ.

AWS SQL High Availability and Failover

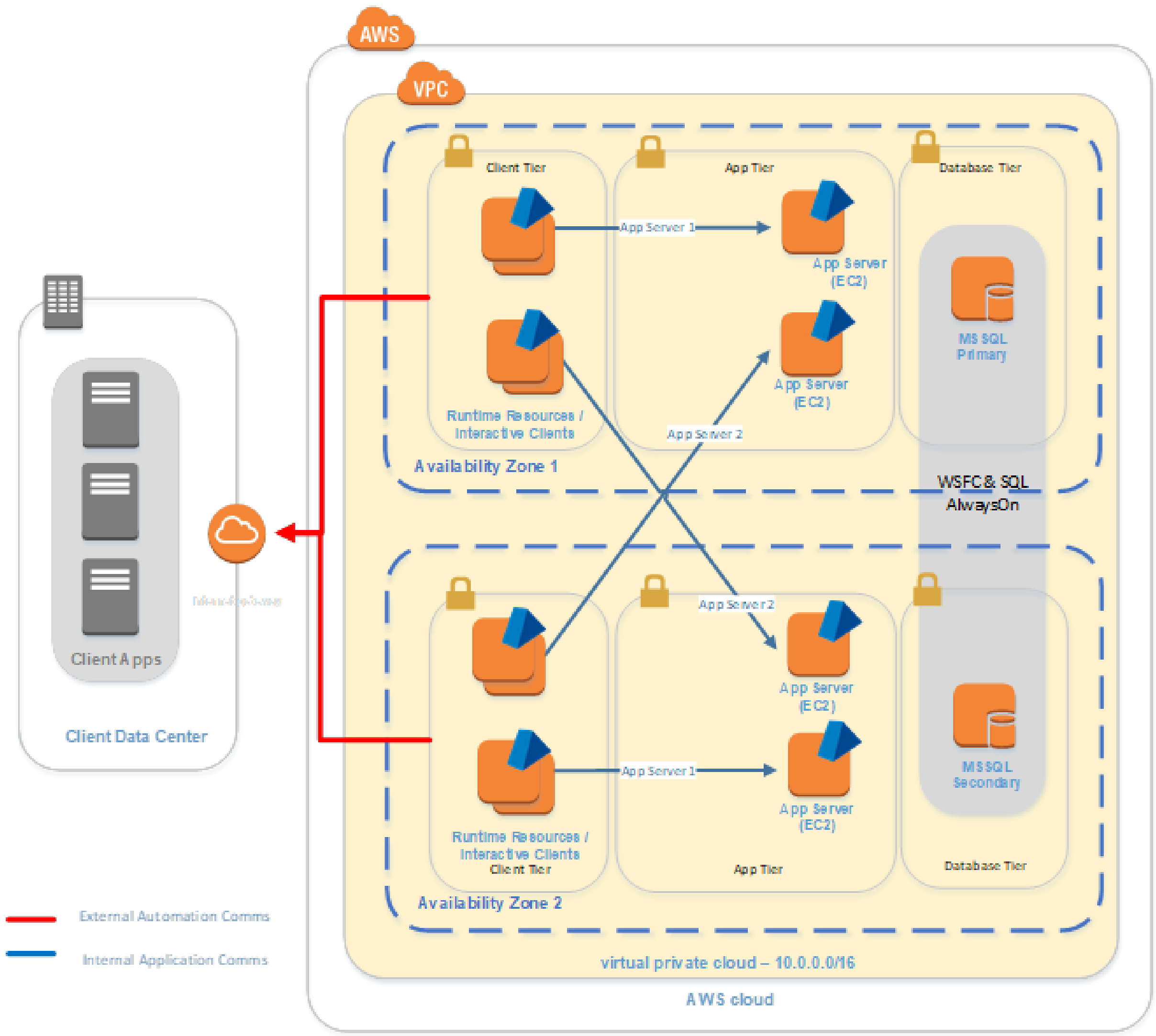
Amazon RDS offers Multi-AZ support for Amazon RDS for SQL Server. This high availability option uses SQL Server Mirroring technology with additional improvements to meet the requirements of enterprise-grade production workloads running on SQL Server. The Multi-AZ deployment option provides enhanced availability and data durability by automatically replicating database updates between two AWS Availability Zones. The replication, failure detection and failover mechanisms of AWS SQL Database are fully automated and operate without human intervention. This architecture is designed to ensure that committed data is never lost and that data durability takes precedence over all else.

The implications of a failover event for RDS are as follows:

- The average failover time for RDS is 60s. During this time, the application servers will continually retry the connection to the database until they reconnect.
- It is possible (depending on the activity being performed at the time, logging settings, etc) that any running processes will fail with exceptions during the time taken for the RDS failover. These will need to be captured and managed after the failover event.
- Active Interactive Client sessions may see interruptions during the failover event.

Medium to Large Scale Deployment Using SQL Clustering

This pattern involves the deployment of a SQL cluster with Always on Availability Groups (AAG) for HADR. Further specific guidance on this topic is provided here.



Key design considerations

- The sizing of the SQL environment (compute and database) should be based on the recommendations within the Infrastructure Reference Guide.
- The number of application servers may be scaled up, according to the size of environment. Refer to the Blue Prism Infrastructure Reference Guide for additional information on sizing.
- It is recommended to use EC2 instances for the runtimes and application servers – see the additional guidance on using AWS Workspaces in Supporting Architecture Patterns. The size of instance should be based on the recommendations within the Infrastructure Reference Guide.
- The use of Server images may have an implication on licensing or support for the client software, though this is rare.

Supporting Architecture Patterns

The following sections outline some of the supporting architecture patterns for peripheral services, such as Authentication and remote connectivity. These are likely to be highly variable, depending on the client's existing AWS usage and strategy.

Authentication

There are multiple options for integration of the Blue Prism environment into a Directory within AWS:

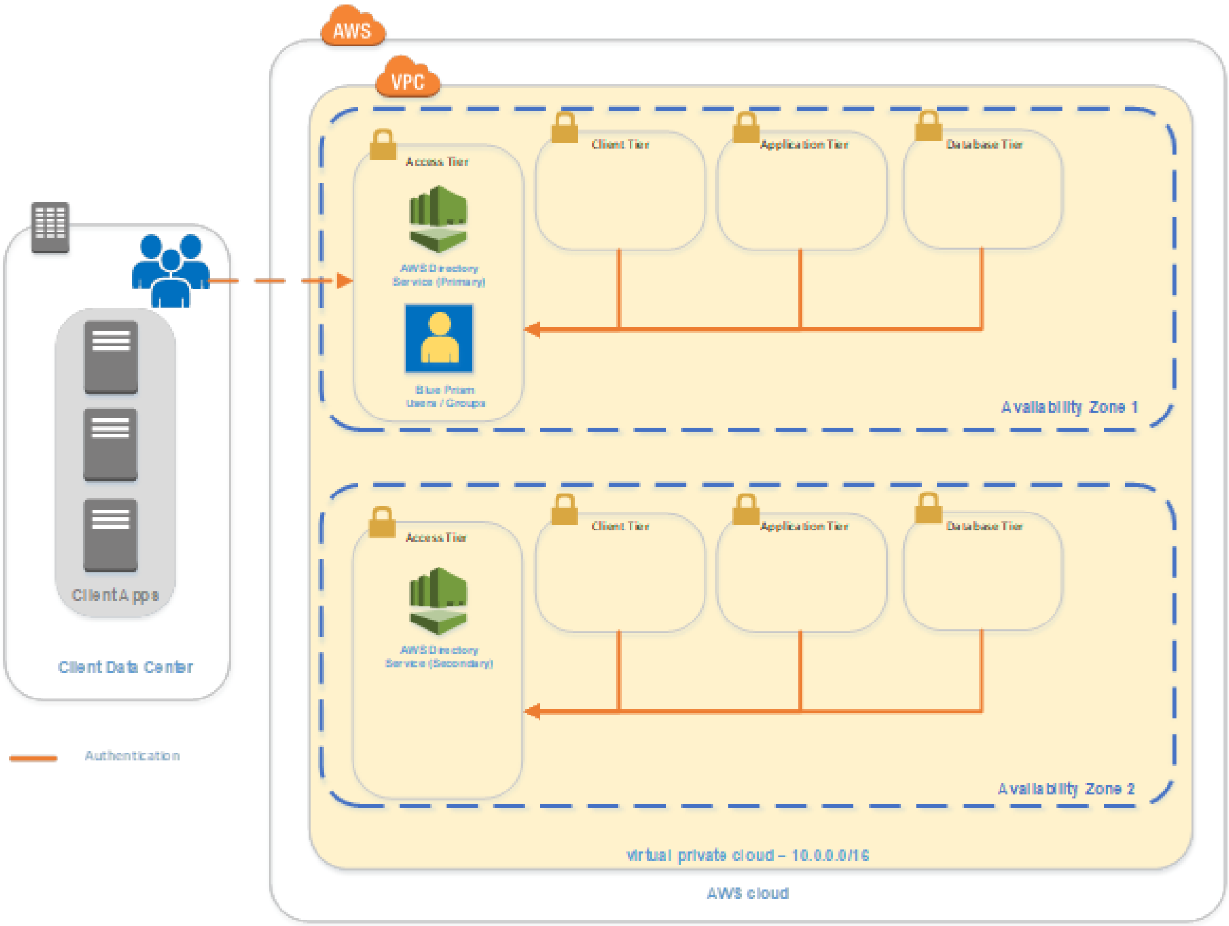
- Active Directory using managed AWS Directory Service (Microsoft AD Enterprise Edition) on the AWS Cloud. Integration into other LDAP offerings within the AWS Directory Services is not supported.
- Active Directory using self-managed Active Directory on the AWS Cloud
- Hybrid scenario – Extending on premises AD to the AWS Cloud

Active Directory Domain Services on the AWS Cloud: Quick Start Reference Deployment

As the underlying solution used with AWS Directory Services is Microsoft Active Directory, the considerations and integration approach and best practices for configuration are identical to the standard guidance for configuring Blue Prism to work with AD. The design principles for establishing these solutions are outside of the scope of this document.

Active Directory using AWS Directory Services (Microsoft AD Enterprise Edition)

This scenario takes advantage of the AWS Directory Services (Microsoft AD Enterprise Edition) option to provision and manage AD DS on the AWS cloud. Instead of fully managing AD DS yourself, you rely on AWS Directory Service for tasks such as building a highly available directory topology, monitoring domain controllers, and configuring backups and snapshots. As this is effectively a full deployment of Microsoft AD, the considerations in terms of how it is connected and structured for a Blue Prism environment are identical to using a manually deployed set of domain controllers.



Key considerations

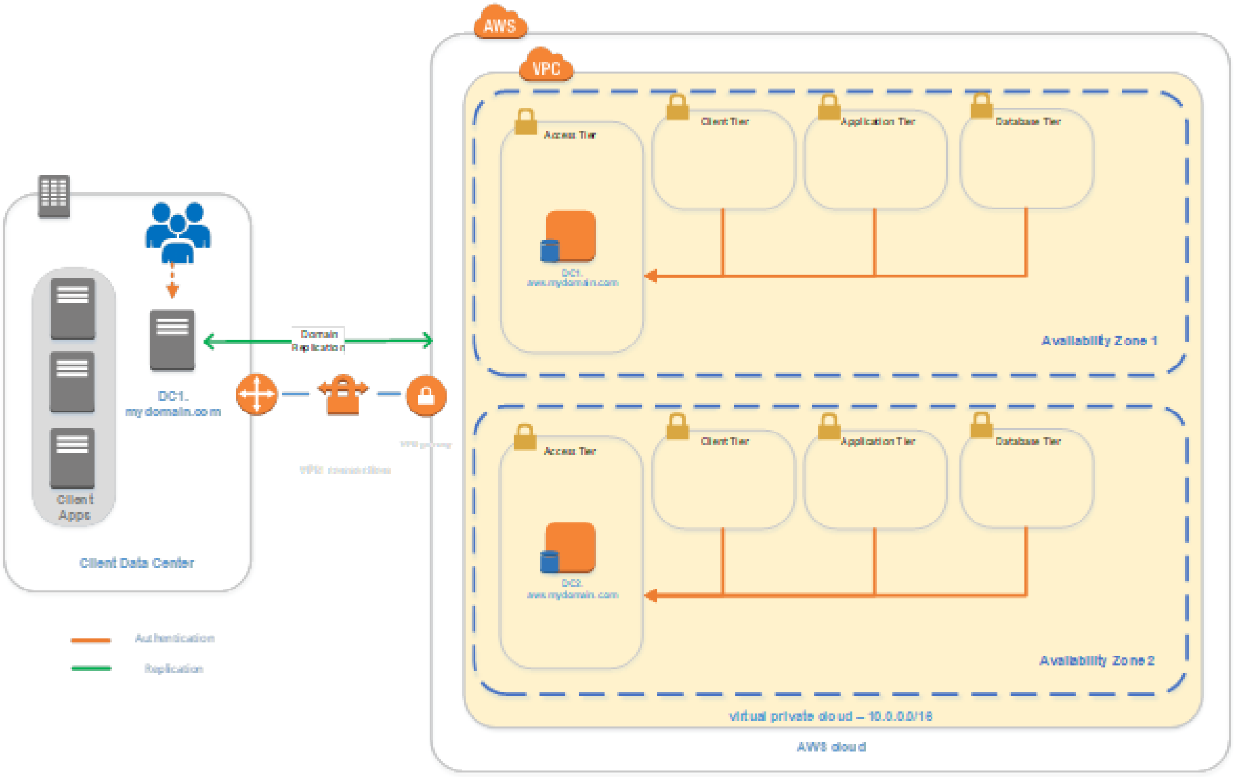
- AWS deploys fully functional Microsoft AD (Enterprise Edition) domain controllers, thus the configuration and considerations for Blue Prism are identical. Refer to the Blue Prism Documentation for integrating Blue Prism with Active Directory for further details.
- Consider the authentication mechanism for automated Applications. If the target applications are hosted on premise, but there is no trust between the on premise domain and AWS, then Blue Prism will need to be authorized to use accounts from the On Premise domain.
- The above design assumes the use of multiple AZs for HADR, hence the deployment of a second AWS DS Domain Controller within the second AZ. If deploying within a single AZ, a second domain controller can be deployed within the same AZ.

Active Directory using Self Deployed and Managed AD on AWS

This pattern is effectively identical to the above, except that you will deploy and manage the domain controllers yourself, as opposed to using the AWS Directory Services capability.

Extending On-Premises AD DS installation to the AWS Cloud (Single AD Forest)

This scenario is likely to be used when the client requires Authentication and control to be available within their on-premise directory, or extension of existing authorizations into the cloud domain.



Key considerations

- The AWS domain controllers must be part of the same Active Directory forest and all devices must be members of the domain in order to support Single Sign-On for Blue Prism and it’s components.

Other AWS Capabilities

This section outlines some of the other technologies and capabilities that are available from AWS and how they affect or impact a Blue Prism environment.

Blue Prism periodically tests various cloud platform capabilities and assess their suitability for use in deployments.

AWS Workspaces

The use of AWS Workspaces for Blue Prism interactive client machines is supported for control and management of the environment only. Installation of Blue Prism and any other requisite software can be built into a custom workspace bundle to ensure a reliable deployment.

AWS Workspaces are a 'skinned' version of the underlying Server Operating System, in that they visually appear to be a client operating system but are missing some key libraries for automation. Therefore, it is recommended that checks are carried out to ensure that all software to be used on an AWS Workspace can run on that operating system.

The use of AWS Workspaces for Blue Prism runtime resources is not supported as full functionality of all Blue Prism automation capabilities and techniques cannot be assured. For this reason, Blue Prism process design on interactive clients is also not supported.

It is recommended that any Workspace environment is integrated into Microsoft Active Directory (AD) Infrastructure by using either an AWS Managed AD or the AD Connector provided by AWS.

AWS Elastic Load Balancers

Elastic Load Balancers (ELBs) is a catch-all term for the suite of platform load balancers provided by AWS, consisting of application, Network and Classic Load Balancers, respectively.

As detailed in version 6 of the Load Balancing Guide, when using load balancers in a Blue Prism deployment, the session affinity pattern must not rely on anything which is inserted into requests or responses. This is primarily achieved by AWS Elastic Load Balancers inserting session cookies into requests. Unfortunately, AWS Application and Classic Load Balancers only allow cookie-based session affinity and so are not supported.

AWS Network Load Balancers (NLBs) do provide a source IP affinity pattern, but have proven to be unsuitable for use with Blue Prism. An AWS NLB may, or may not, purge its persistence cache when a target group changes, and can also indiscriminately strip security headers from WCF messages. This functionality causes issues with Blue Prism Runtime Resources, therefore AWS NLBs are not currently supported. High Availability can still be achieved by installing software on an EC2 instance. Solutions like HAProxy, F5 or other load balancing software provide full features and capabilities for load balancing.