appgate

Appgate SDP For AWS Reference Architecture

Revision History

Version	Who	Date
V 3.0	Chris Scheels	January 1, 2020
V 3.1	Felipe Conill	February 10, 2020
V 3.2	Felipe Conill	April 24, 2021
V 3.3	Andre Garrigo	March 31, 2022

Introduction

As businesses accelerate their adoption of AWS for high-value production workloads, the current threat and compliance landscape demands that security be considered up-front, and in fact act as an enabler, rather than an impediment to user productivity and business agility. The Software-Defined Perimeter (SDP) – a new, open security architecture – is ideally-suited to securing exactly the kind of distributed, dynamic, and API-driven environments in which AWS excels, and enables agility while improving user productivity.

Appgate SDP is Appgate's product built on the Software Defined Perimeter specification; implementing the core SDP principles as outlined by the Cloud Security Alliance's SDP Working Group, coupled with valuable extensions to support enterprise scale and operational management requirements. The most notable aspect of SDP is that while access policies are defined for identities, enforcement functions at the network layer. This provides direct connectivity from the user to multiple protected network resources while responding to environmental conditions and user attributes in real-time.

Appgate SDP has some very specific advantages over traditional legacy infrastructures:

- **User-Centric Network Security:** Appgate SDP provides application and service-specific authentication and authorization to uniquely grant network access from within and outside of the corporate perimeter. Appgate SDP dynamically creates a secure, encrypted network *"segment of one"* that is tailored to a user's specific attributes for each user session. Unlike traditional approaches, network access rules are not static, remaining unchanged for months or years, but are dynamically generated and enforced.

- **Cloud-Native:** Appgate SDP is designed to support IaaS environments, using a flexible, distributed deployment model, which suits many different cloud architectures. Appgate SDP automatically detects server instance creation and leverages user and server metadata to evaluate access. Driven by a common policy model, Appgate SDP orchestrates these elements -- dynamically controlling access by authenticated users to specified cloud resources.

- **Seamless Integrations:** Appgate SDP reduces costs by eliminating IP address configuration, ad-hoc third-party set-up, and managing user access across a hybrid cloud infrastructure security. Appgate combines authorization, encryption, and access control in one system while seamlessly integrating with existing identity management, multi factor authentication and SIEM solutions. Its API-first architecture enables businesses to utilize existing authentication, logging, and incident response processes to quickly support agile hybrid cloud security requirements into their operations and security processes.

- **Security On-Demand:** Appgate SDP is built on a distributed model to support a variety of use cases and provide an architecture which aligns with the security controls of the hosts and applications. This ensures that traffic is encrypted, and that any networks used to access the resources are in close proximity, eliminating the potential security risk presented by the intermediate network.

- **Compliance is Key:** Appgate SDP helps enterprises reduce regulatory compliance costs by reducing scope and audit complexity. Cloud providers offer some functionality for the myriad regulatory requirements, but Appgate SDP can greatly enhance these by providing a common logging and federated access framework. With this in place, Appgate SDP inherently reduces the number systems that fall within audit scope through its approach -- often eliminating the need for regulatory controls themselves. Robust 360-degree, user-centric logging also provides any evidence necessary to meet audit requirements.

- **Hybrid Cloud:** Even as organizations migrate to AWS, they often have on-premises resources. Appgate SDP's architecture and policy model supports access control for cloud and on-premise resources from a single, integrated platform.

The remainder of this document introduces the Software-Defined Perimeter architecture and an overview of several AWS reference architectures for Appgate SDP. For further information about Appgate SDP, please visit: https://www.appgate.com/secure-access/appgate-sdp

What is the Software Defined Perimeter?

The Software Defined Perimeter is a modern approach to the problem of securing today's networks. Its aim is to solve the challenge of stopping network attacks on application infrastructure, while ensuring user productivity and improving security operations efficiency. The Cloud Security Alliance's SDP Working Group has developed a clean-sheet approach that combines on-device authentication, identity-based access, and dynamically-provisioned connectivity.

Specifically, Appgate SDP is comprised of five distinct functions, to not only support the tenets of the SDP specification, but also enable high availability and fit within the operational framework of an enterprise.

Controller: Acts as the brains of the Appgate platform and is the acting control plane management function for orchestrating policies. These decisions may be granted simply based on IdP authentication or based on more complex decisions involving conditions, device posture, or 3rd party information.

Gateway: Connects the user's endpoint session to the resource. The Gateway is designed for carrier-grade high availability and throughput to meet the most demanding use cases.

Client: Installed on end-user devices, this component securely establishes an encrypted, tunneled network connection to the Appgate SDP Gateway(s), ensuring that all user traffic is secured. This component also performs device inspection and device posture checks, to enforce access policies.

Connector: Provides the ability to extend SDP to business premises, whether branch offices or other remote locations, to securely enforce policies based on device classification and observed behaviors.

Logging: Appgate SDP generates detailed and searchable user-centric access logs, which are valuable for security and compliance purposes. These logs can be stored within the Appgate SDP system or forwarded to an enterprise SIEM.

While many of the security components in SDP are well-proven, the integration of these components is quite novel. More importantly, the SDP security model has been shown to stop network attacks including DoS, Man-in-the-Middle, Server Query (OWASP10) as well as Advanced Persistent Threats (APT). SDP was not designed as another DMZ add-on to an existing set of security controls such Proxies or VPNs; rather, this new model provides an alternative to these outdated tools which were developed in a time before Cloud and hybrid Cloud became pervasive.

It is important to understand that Appgate SDP has filled in the gaps in the Cloud Security Alliance’s SDP model and extended the model with specific enhancements that are built on several key principles:

- The first principle is that users and devices operate outside the traditional perimeter. Today’s user populations are diverse and operate from many different physical locations. And they are accessing server resources that are increasingly located in the Cloud (or at least remotely from the users)

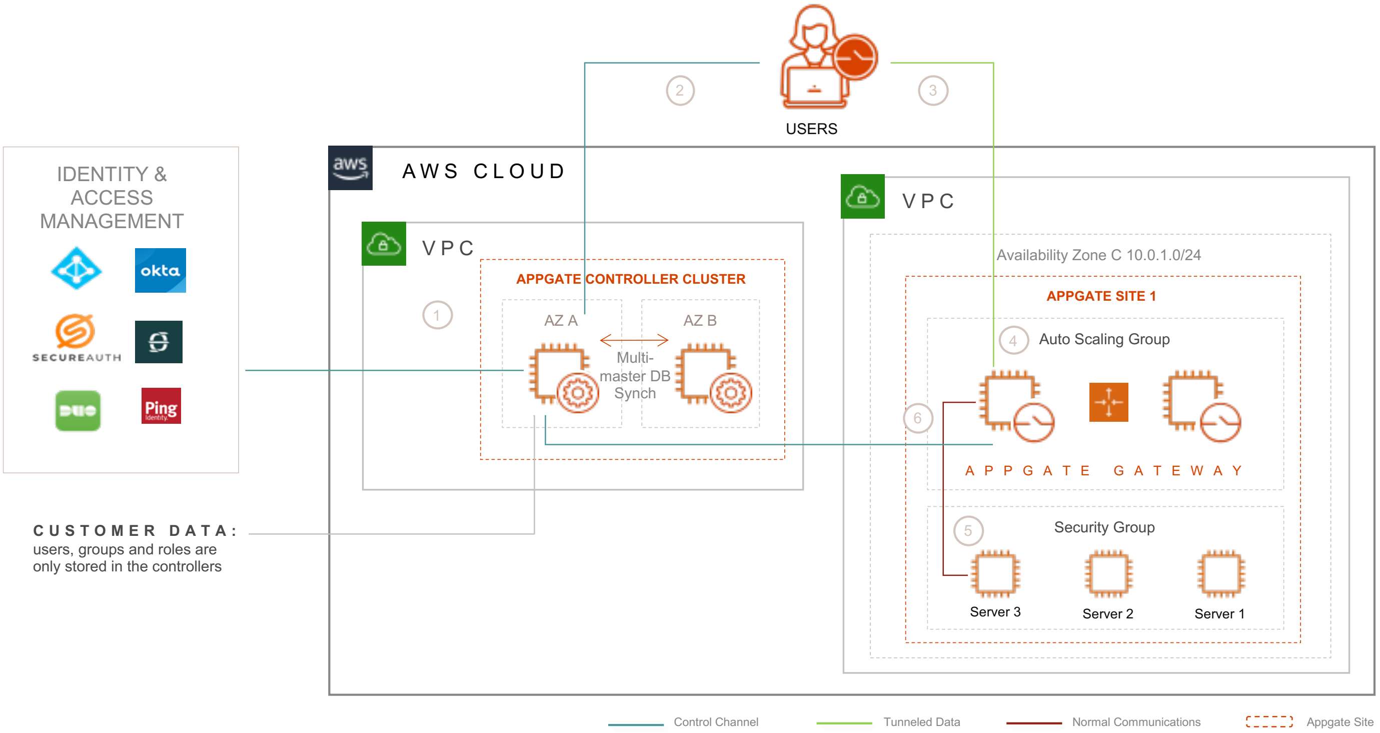
- The second principle of a Software-Defined Perimeter is the notion of “authenticate first, connect second.” Unlike a traditional network that connects users in various roles or groups to a network segment and then relies on application-level permissions for authorization, a Software-Defined Perimeter creates individualized permissions; as a user’s situation changes, the individualized permissions change. This ensures that only authorized users can connect to network resources. Resources are rendered invisible (“cloaked”) to dangerous reconnaissance activities, which greatly reduces the attack surface and significantly enhances an enterprise’s security posture.

- The third principle is that the access controls should be placed as close to the protected hosts as possible. When the user attempts to access a resource – for example, by opening a web page on a protected server -- the Client redirects the request to the closest Gateway via a secure tunnel. This in turn applies additional policies in real-time; for example, to control access based on the user’s network location. This premise allows clients to make multiple connections to multiple gateways simultaneously across clouds to address the user’s specific connectivity needs.

Having multiple Gateways (access points) makes the SDP highly suitable for hybrid environments – allowing consistent access policies to be applied to legacy network, data center and cloud environments. New Sites are independent of one another and easily deployed with short lead-times; they simply require Internet access.

Core Appgate SDP Architecture

Appgate SDP draws on user context to dynamically create a secure, encrypted network “segment of one” that is tailored for each user session.



IT WORKS AS FOLLOWS:

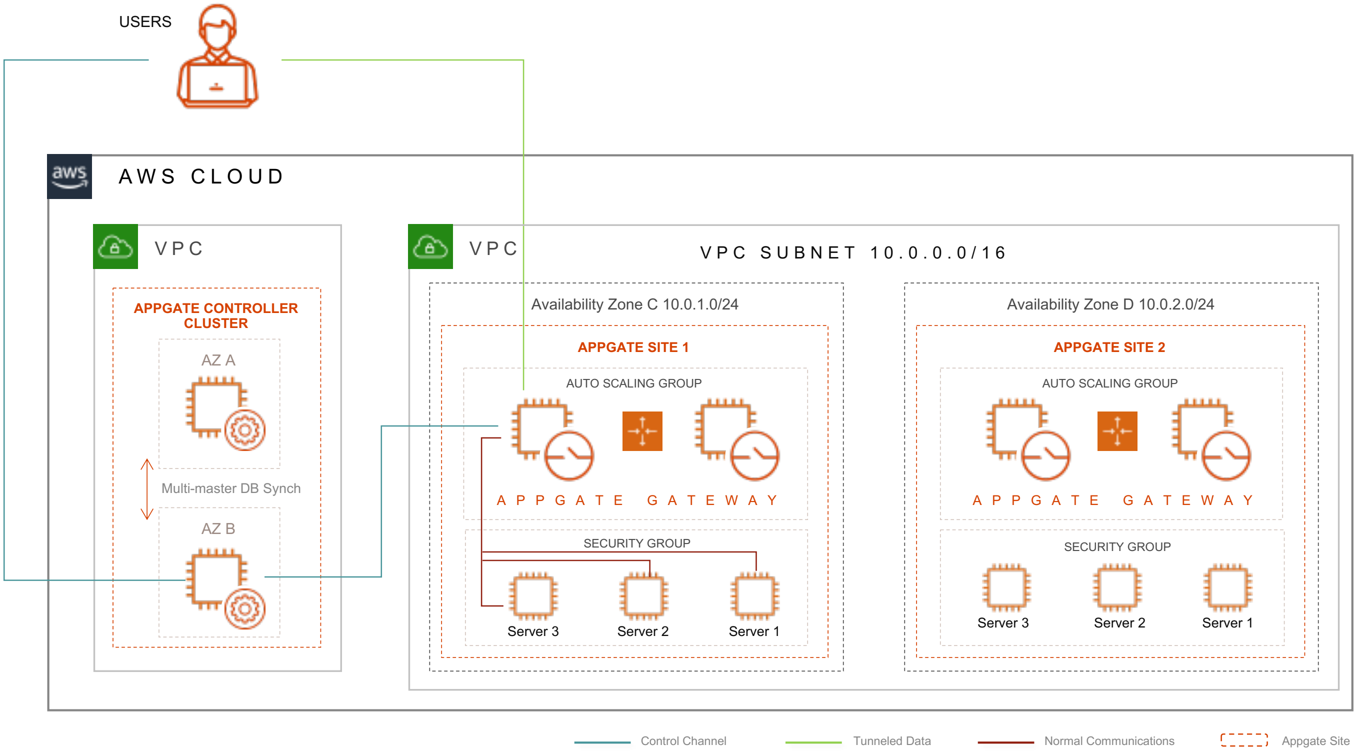
1. User authenticates to the Controller
2. Controller applies policies based on user attributes, roles, and context; and issues a signed token listing the resources to which the user has access
3. User attempts to access a protected resource behind a Gateway
4. Gateway evaluates policies in real-time, ensuring that all conditions are met – for example, network location, time of day, device health, and service metadata, such as security groups. Users may be prompted for additional information, such as a one-time password
5. If permitted, the Gateway opens a connection to the target resource for the user
6. Gateway automatically detects newly launched services, and based on metadata and policies, adjusts user access

AWS Reference Deployment Scenarios

Appgate SDP has been deployed by AWS customers in many ways to support numerous different enterprises, satisfying multiple regulatory, scaling, and cost control requirements. The following scenarios describe commonly used architectures to begin achieving these goals within AWS.

Scenario 1: Appgate SDP in AWS with Multiple Sites

In this scenario, the entire Appgate SDP system is deployed within AWS. There are two distinct "Sites," each protected by separate Gateway clusters.



Architecture Explained

The Appgate SDP Controller is deployed as a two-node cluster within a single VPC, with each controller deployed in a separate availability zone for redundancy. The Controllers utilize multi-master database synchronization for high availability. Clients utilize DNS load balancing to connect to one of the clustered controllers.

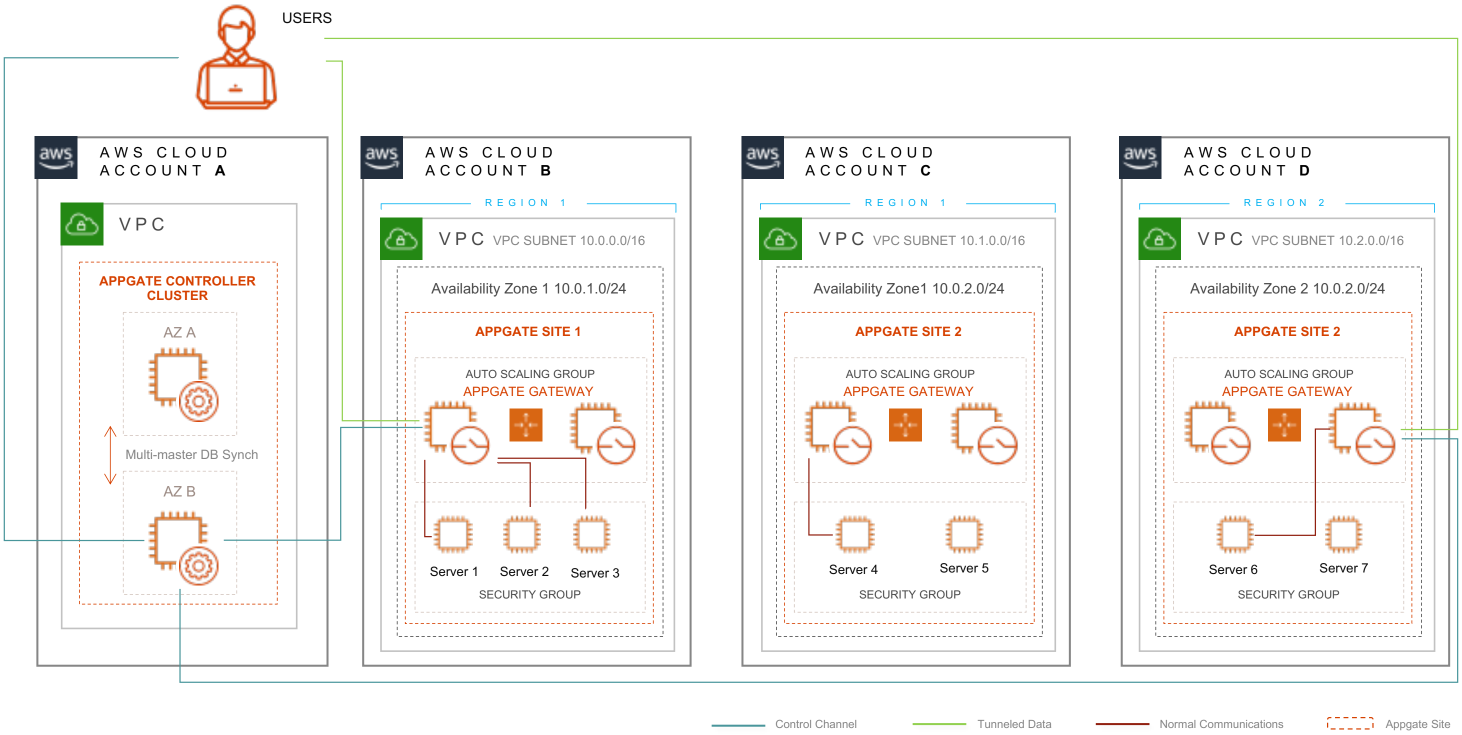
Gateway clusters sit inline on the network, in front of a "Site," which is a logically related group of servers. Clients obtain the list of Gateways in each cluster from the Controller (after successful authentication) and connect to one of the Gateways in the cluster. Client state is synchronized back to each client, so if a Gateway fails, the client will reconnect to another Gateway in the cluster and send its state to the new Gateway to reestablish communications with the target servers. In this example, each Gateway cluster protects servers within a single availability zone.

Benefits and When to Use

This scenario is reflective of a relatively basic cloud environment. This could be utilized in conjunction with a hybrid cloud environment in scenarios whereby the identity provider is cloud-based or in scenarios in which enterprises are migrating to the cloud.

This simple model offers high availability and Zero Trust enforcement to protected resources. Additionally, with this Gateway model, users are restricted beyond VPC granularity to specific resources within a subnet, as defined by the Appgate SDP policy.

Scenario 2: Appgate SDP in AWS across Multiple Accounts



Architecture Explained

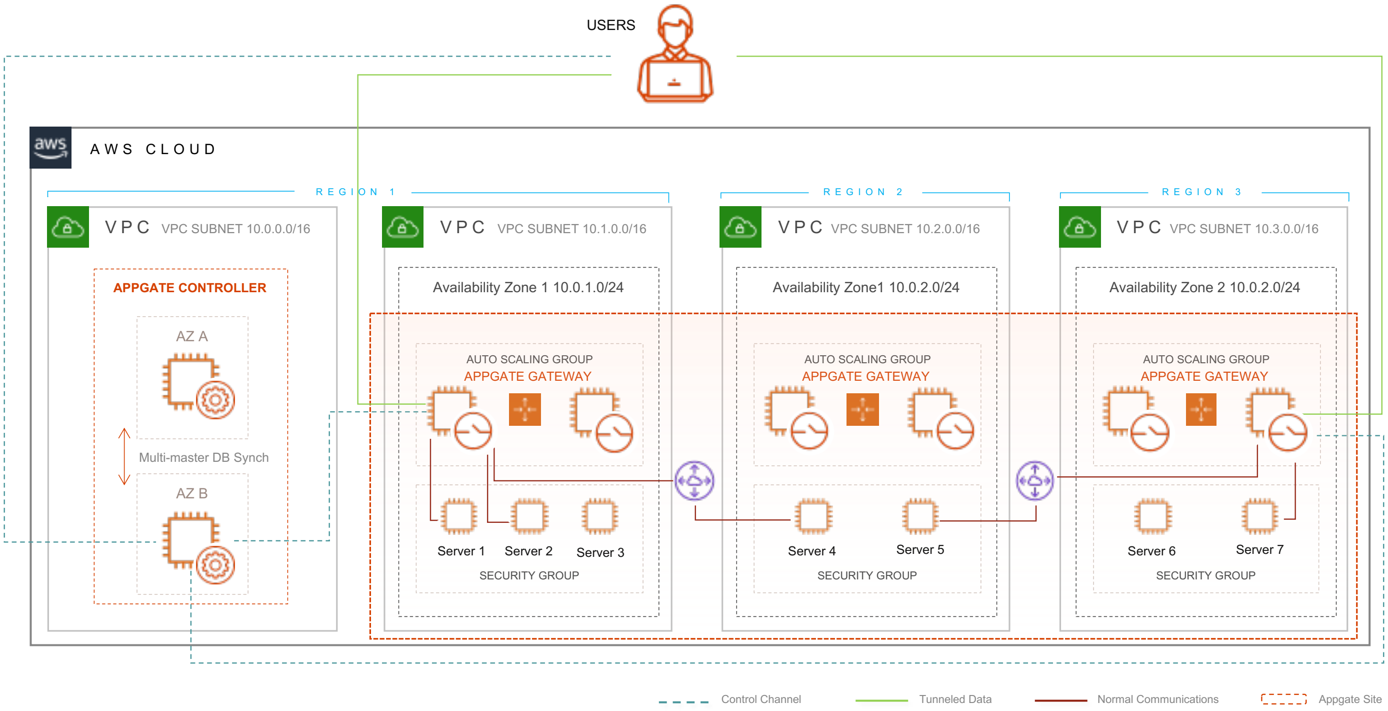
This scenario is a variant of Scenario 1, but worth pointing out as it addresses a common need whereby the resources are restricted by account. As new resources are created or destroyed (or their instance metadata is changed), access is simultaneously propagated to the users so they can instantly access their permitted resources. Users have transparent and concurrent access to permitted resources across these accounts.

Benefits and When to Use

Appgate provides seamless agility for multi-account environments that utilize cost structures or manage multiple client accounts. This provides users (such as DevOps teams) secure and compliant access without downtime due to waiting for administrators to update IAM roles or security groups. Additionally, users can simultaneously access multiple resources without needing to disconnect.

Scenario 3: Appgate SDP in AWS with a Single Site Across Regions

Like in scenario 1, the entire Appgate SDP system is deployed within AWS. However, in this scenario the Site is protected by a Gateway cluster. This Site spans three AWS Regions. The Controller cluster is highly available, deployed to multiple Availability Zones.



Architecture Explained

In this scenario, the Controllers are deployed across multiple AWS Regions. Clients obtain an IP address for a Controller through DNS load balancing as in Scenario 1 (this is unaffected by their distribution across AWS Regions).

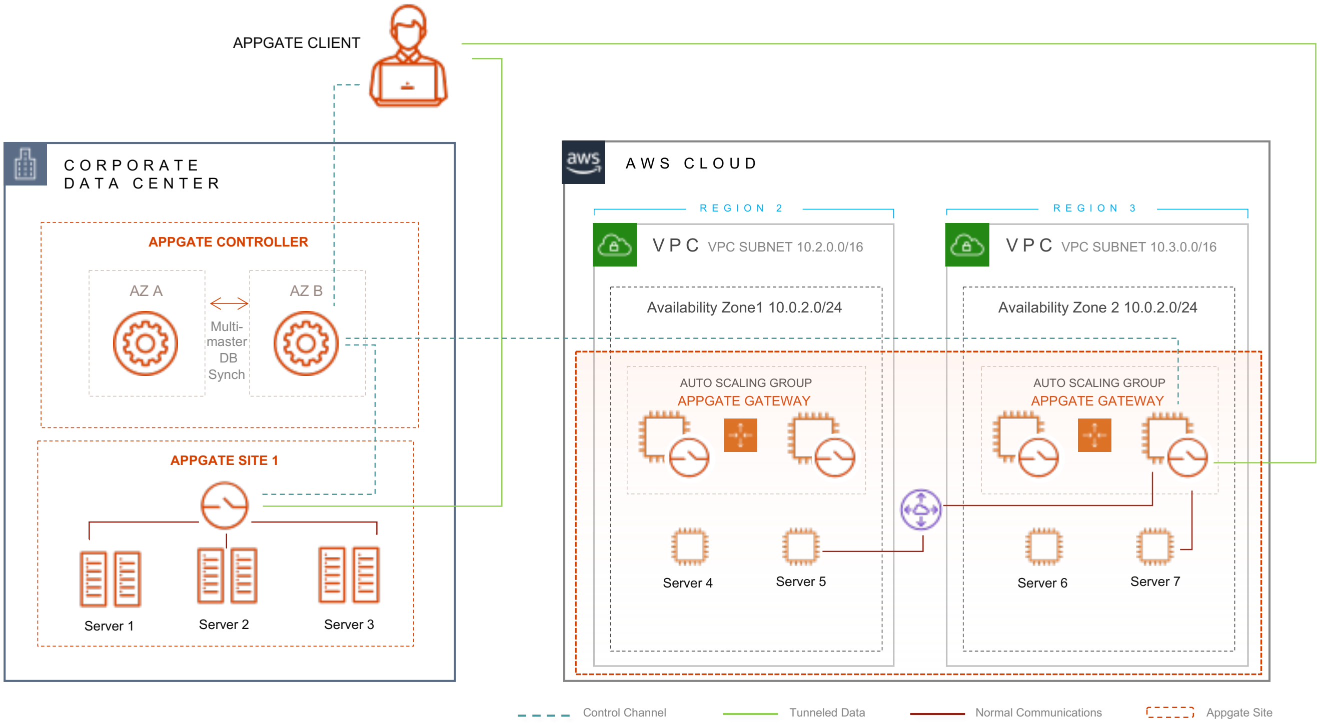
The Gateways are deployed differently: there is a single Site, with a Gateway cluster gating access to all resources in the Site. This Site spans three AWS Regions. In the diagram, the client connects to a Gateway running in Availability Zone D, and its tunneled traffic is directed by the Gateway to servers running in Availability Zone 1 in Region 1 and to a server running in Region 2. The client is further connected to a Gateway in Region 3 and traffic is tunneled to servers in Region 3 and also in Region 2. Note that this requires network routes from the Gateway in Availability Zone 1 to the protected servers running in Availability Zone 2, as well as to servers in Region 3.

Benefits and When to Use

This type of deployment is what may be found in a DevOps environment where access is needed across VPCs, but data privacy roles restrict personnel from access. Previously, this type of management and compliance was operationally prohibitive, but with Appgate SDP, enterprises can utilize AWS metadata to dynamically control access at a logical layer versus a physical layer. This allows an administrator to simply change tags in an open environment where Appgate can dynamically resolve and redefine permissions.

Scenario 4: Appgate SDP Hybrid Deployment

In this scenario, Appgate SDP is deployed in a hybrid model, with the Controller and some Gateways running on-premise, and with two Gateway-protected sites running in AWS. (The Gateway deployment is identical to scenario 1 and will not be discussed here).



Architecture Explained

In this deployment model, client interaction with the Controller is identical to that described in scenario 1. This is a good illustration of an Appgate SDP distributed architecture, in which the Controller location is physically distanced from that of the Gateways and the resources they protect.

Benefits and When to Use

One of the benefits of being “software defined” is that, as software, it is inherently more agile and flexible compared to a physical or even virtualized version of traditional, perimeter-centric security solutions. This illustration shows how the controller can reside on a physical premise (perhaps as part of an existing Appgate SDP deployment) and by easily deploying incremental Gateways, can extend the same security posture to the cloud. Additionally, this model could utilize Appgate SDP’s Resolvers to dynamically grant on-premise access model to AWS. For example, a user in New York could have access to their local development environment, but only their resources in that environment, their staging environment, and their resources plus the resources they need in the production environment -- all based on using a common metadata taxonomy.

Instead of users having access to an entire network segment (CIDR block) for staging or production VPC, they now only have access to the specific resources that they need, and access is optimized based on performance, costing and scaling goals (e.g., Dev environment is kept on-prem to minimize AWS restart costs and source upload time.) Appgate SDP provides this out of the box: completely and seamlessly.

Scenario 5: AWS Transit Gateway or Transit VPC

In this scenario, Appgate SDP is deployed wholly within AWS. This scenario provides a High Availability pair of Appgate Controllers for Authorization and Authentication. The Appgate Gateway clusters terminate the mutually encrypted mTLS tunnels and send the traffic to the resources based on the Routing Table associated with the Transit Gateway. Resources within AWS connected by the Transit Gateway are protected by the Appgate Gateways. More information regarding AWS Transit Gateways can be found on the AWS site.

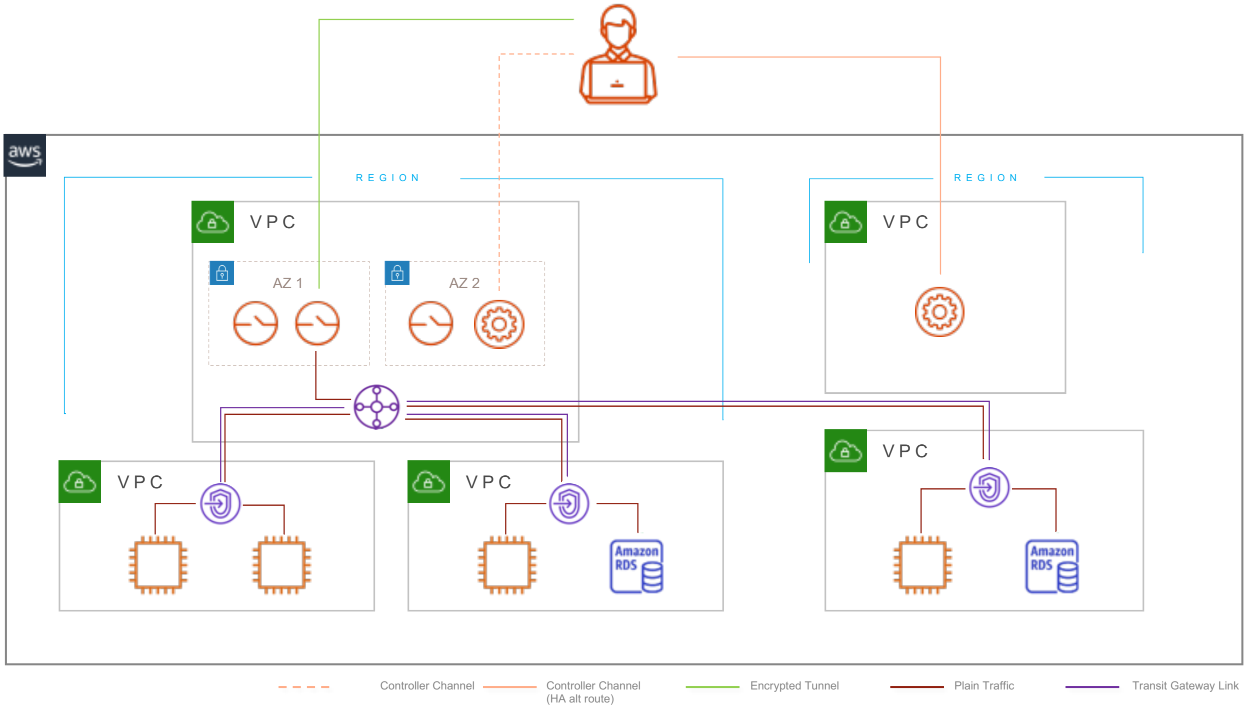
Appgate Gateways can be scaled vertically, through instance sizing, or horizontally using Autoscaling Groups native to AWS. This allows the customer to size and scale the Gateway cluster, tailoring it to meet the demand of their end-users.

Benefits and When to Use

AWS Transit Gateway is a service that enables customers to connect their Amazon Virtual Private Clouds (VPCs) to a single gateway. This is beneficial when customers need to connect multiple VPCs without using VPC peering which only provides point-to-point connectivity between a pair of VPCs.

With AWS Transit Gateway, you only create and manage a single connection from the central gateway into each Amazon VPC, on-premises data center, or remote office across your network. Transit Gateway acts as a hub that controls how traffic is routed among all the connected networks, which act as the spokes.

Large enterprises that manage many AWS accounts can benefit from deploying an AWS Transit Gateway with their Appgate deployment. Individual AWS accounts are typically configured as individual sites. This can add to the overall complexity of the solution. Leveraging the AWS Transit Gateway service allows for a single point of connectivity into the AWS environment making for a simpler and more efficient design. Customers with multiple branch locations that are connected into AWS can leverage the Transit Gateway as well by terminating those connections to the Transit Gateway.


aws
REGION
REGION
VPC
AZ 1
AZ 2
VPC
VPC
VPC
Amazon RDS
VPC
Amazon RDS
Controller Channel
Controller Channel (HA alt route)
Encrypted Tunnel
Plain Traffic
Transit Gateway Link

Backup and Recovery in AWS

An Appgate SDP collective can be restored using our backup and restore scripts as specified in the Appgate SDP Admin Guide. Furthermore, in AWS you can leverage EBS Snapshots of the Controllers to restore Controllers in the Region. As a best practice, one should:

- Schedule an automatic backup of the EBS volume for your Controllers
- Restore the EBS into a new AMI
- Build a new EC2 instance

Please reference our Backup and Recovery guide for step-by-step instructions on how to backup Appgate.

Evolve with Appgate SDP for AWS

AWS is constantly evolving and Appgate SDP’s ability to connect and secure resources at the network layer, utilizing software defined technology, proves to be a winning combination. Services such as AWS Transit Gateway, as well as application-level technology, can be easily bridged and controlled within a hybrid cloud environment. For example, customers utilizing AWS Transit Gateway can capitalize on internal AWS efficiencies while adding policy-enabled, encrypted security to simplify east-west communications while reducing threat exposure.

Similarly, AWS’s rich application service portfolio can be connected using Appgate SDP to provide permission-based access and restrictions to ensure compliance and eliminate unauthorized and unaccounted access.