SECURING USER ACCESS TO ENTERPRISE SYSTEMS IN PRIVATE OR PUBLIC CLOUDS

How Appgate SDP reduces the attack surface with an individualized network segment of one

Introduction

Dispersed global teams conducting business from anywhere, anytime results in increased workforce mobility, distributed systems, and widely networked, on-demand computing environments. This adds layers of complexity to enterprise networks and puts critical data at risk. At the same time, enterprise IT and security teams are expected to deliver more applications, faster, and ensure availability from anywhere, 24/7 while maintaining a high security profile.

Not only do employees need to access critical business data and resources, but customers, partners, third-party contractors and vendors have continuous requirements for access as well. Combined with increasingly complex systems and lack of resources, it's difficult for IT teams to be agile and flexible when delivering services to the business. This creates a challenge when trying to provide comprehensive access control at a granular level and support a mobile workforce. One size doesn't fit all and enterprises need a flexible, scalable solution that meets the specific needs of their business and disperse workforce.

Traditional security and remote access tools like VPNs, next-generation firewalls and network access control (NAC) solutions provide all-or-nothing access control, typically offering carte-blanche access to all authenticated users. These tools don't address the potential for insider threats, stolen credentials, or malware attacks. Organizations often attempt to address these issues by applying multiple security tools that in the end result in a patchwork of silos, each one only solving a minor part of the broader challenge, imposing administrative overhead and requiring extensive manual activities. The result? Organizations are no more secure than before. Adding more tools and systems doesn't automatically improve security; in fact they add complexity which can reduce security.

Organizations need another way. IT teams need the ability to provide unified, granular access control to applications, services and infrastructure, whether on-premises or in the cloud. And they need to apply the same level of access control scrutiny to devices brought into the environment by third-parties, contractors, or even employees. Security professionals need the ability to make the network 'invisible', granting visibility and access to only the applications and services that users need to do their job.

Appgate SDP, an industry-leading Zero Trust Network Access solution, enables business agility and flexibility to adapt to the dynamic demands of the workforce, customers and the business to be competitive on a global scale. By providing a comprehensive, centralized approach to secure access control, enterprises can provide more flexibility in how employees, customers, partners, and third-parties access and interact with their most critical resources. Appgate SDP provides real-time access on a need-to-know basis, and enables a unified way to control access while maintaining a high security profile. Enterprises leveraging Appgate SDP can improve security, maintain productivity, reduce operational costs and be more flexible and efficient in delivering services and applications that keep the business running.

Moving to a Dynamic Network Architecture

Making security challenges even more complex, many organizations are in the process of strategically moving toward cloud-based infrastructures. Whether using an on-premises private cloud or a public cloud, these dynamic infrastructures bring with them many benefits, but also bring security and management challenges. There are three main cloud-specific security challenges:

1. Clouds are typically very dynamic, with an ever-changing set of servers being continually created and destroyed. This makes it much harder to manage from a security (and compliance) perspective.
2. Each cloud infrastructure takes a slightly different approach to user access policy and configuration. Most organizations use multiple clouds (an average of six), and security teams are challenged to make sense of the disparate security models used by each cloud.
3. Despite their advanced technological underpinnings, most cloud providers still follow a traditional all-or-nothing network access model limited to source and destination IP address controls—and aren't identity-centric.

In today's heightened threat landscape, these challenges translate into significant risk, and additional work for already-overburdened IT and security teams. Forward-thinking organizations recognize that they need a better approach; one that's purpose-built with simple, consistent, fine-grained, and dentity-centric security for both on-premises and cloud-based networks.

Today's Network Reality

Enterprises have done a reasonably good job of defining and managing "who has access to what" at an application level. However, in nearly every organization, this isn't supported consistently at the network level.

At best, the network may look something like Figure 1, with a relatively small number of VLANs, but in most organizations the network is even simpler.

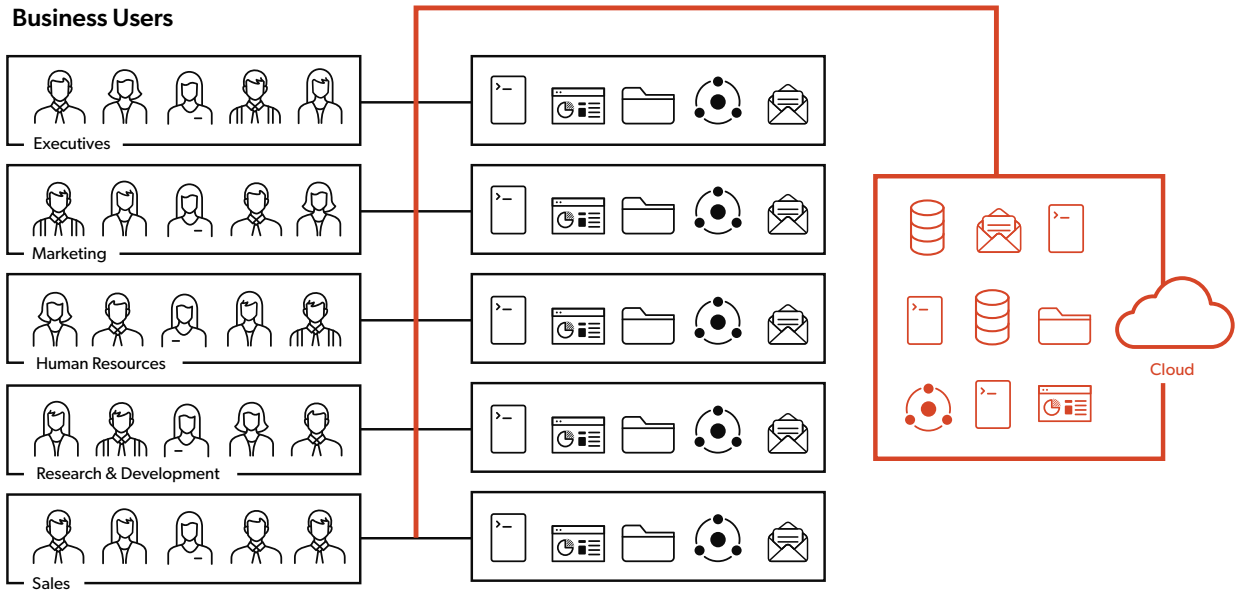


Figure 1: Organizational network access example

The biggest problem with this approach is that network access controls are defined in isolation from application access. Users end up with over-privileged network access, which introduces a significant security gap.

Adversaries are well aware of this gap and often leverage it once they have established a foothold. They don't see users with carefully controlled levels of application access. They instead see the entire network, as depicted in Figure 2, and move quickly to exploit weaknesses such as unpatched servers, known vulnerabilities, or default passwords.

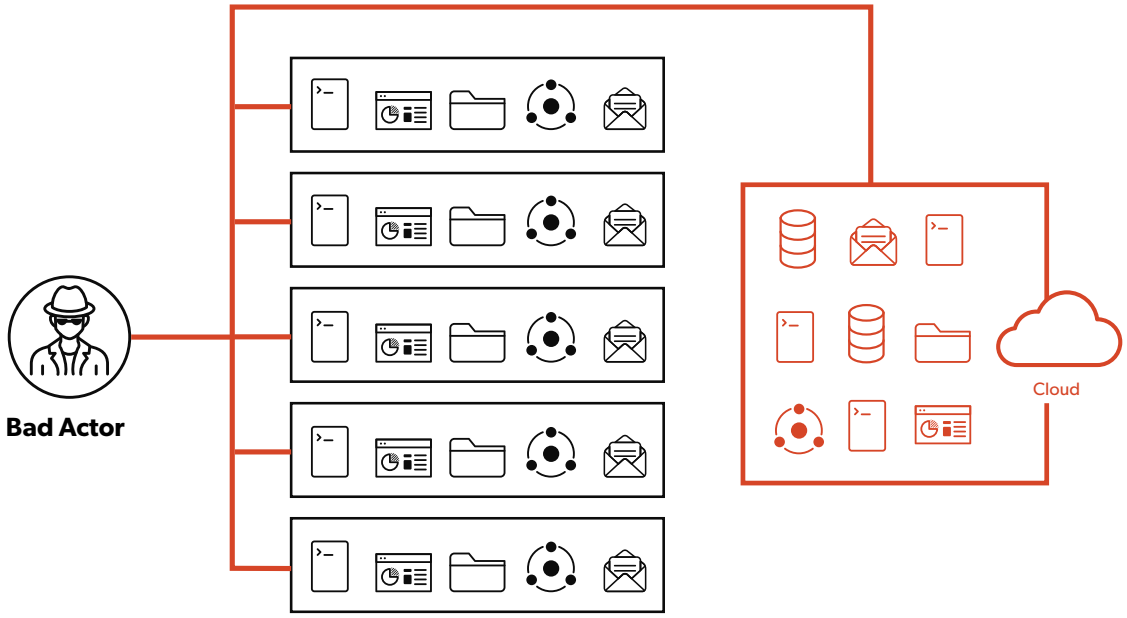


Figure 2: Bad actor's view of an organization's network—they see everything

Traditional network tools cannot help—it's simply too difficult and labor-intensive to attempt to solve this problem with tools such as firewalls, NAC, or VLANs. Each tool falls short of meeting security and management requirements even in relatively static enterprise environments, not to mention dynamic cloud environments.Instead of misapplying these traditional tools, Appgate SDP dynamically creates a segment of one between the users and the network resources they are entitled to access, as shown in Figure 3. Typically, network access should be proportional to the security context the user presents at the time. The more benign context they present, such as physical presence on a company network, one-time password, or certificates, the more network resources they can access. Ultimately, each user's network access entitlements are dynamically altered based on identity, device, network, and application sensitivity, driven by easily configured policies. By aligning network access with application access, users remain fully productive, while the attack surface area is dramatically reduced.

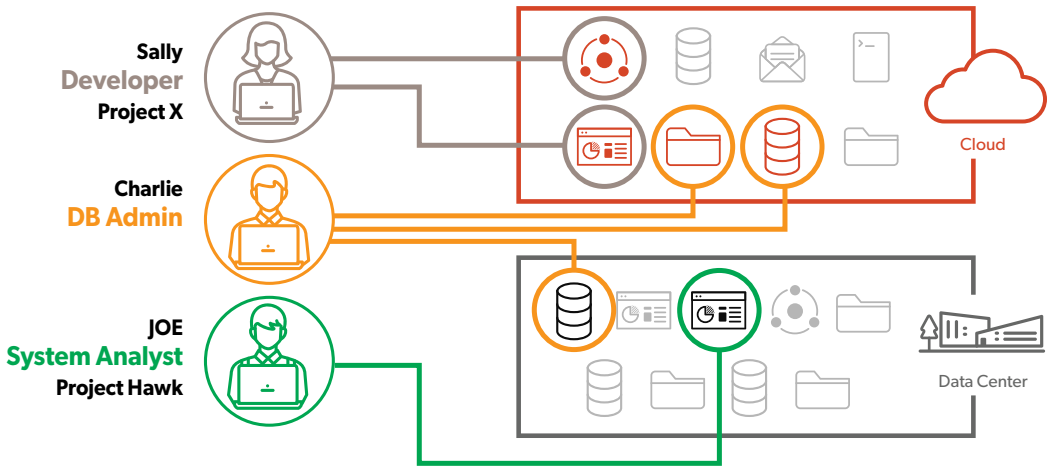


Figure 3: Appgate SDP dynamically creates a segment of one between the users and entitled to access.

Creating a Segment of One with Appgate SDP

Appgate SDP relies on three principles to achieve this new approach to security:

1. Rich user context
2. Fine-grained network access control
3. Real-time control

With traditional network security tools, the only context used to grant authorization is identity, often through weak, password-only authentication, and only validated at the time of authentication. But identity is only one contextual element used by Appgate SDP. It can leverage dozens of other aspects such as whether the user is logging in from a company-managed device on the corporate network, whether that user has passed a multi-factor authentication challenge, or if client-based anti-malware software is running.

Traditional network access controls operate at a segment level, granting user access to an entire network segment in an all-or-nothing fashion, comprising hundreds of hosts. Due to these limitations, organizations are forced to rely on application-level access control—that is, relying solely on authentication to protect their systems. In today's world of heightened threats, where users are one click away from being compromised, this is a big problem. The new approach requires combining network and application access with the ability to hide everything on the network—segments, hosts, services and applications—unless a user is specifically authorized to see them.

But to take full advantage of this segment of one approach, context and access controls also need to take into consideration what the user is trying to do, at the time they're trying to do it.

To respond to what the user is doing, the system has to have real-time controls in place. This makes perfect sense when you pose the question "Why should I trust an identity presented 12 hours ago?" Just like many online services, the system should require you to present a password or one-time password (OTP) before it grants access to a system. Real-time controls also make the user's life easier because they only need to present a minimum set of credentials initially and will only have to present more if they try to access more sensitive systems.

It's All About the User

In the fight to defend our networks, servers and data from adversaries, it's important to have a frame of reference to work from. Traditional security views defenses from the perimeter, and in fact many defenses are still located at the network boundary

today. Before the network perimeter dissolved, it was the perfect place. But, this is no longer the case. In today's world, it only makes sense to have the user as the reference point. Appgate SDP delivers this identity-centric access to resources across traditional networks, in the cloud or in virtualized environments.

Appgate SDP System Components

Appgate SDP secures the network with a software-defined perimeter—a network security model that dynamically creates one-to-one network connections between the user and the resources they access. Appgate SDP is deployed in-line in the network, however unlike other such solutions, Appgate SDP is unique in two ways.

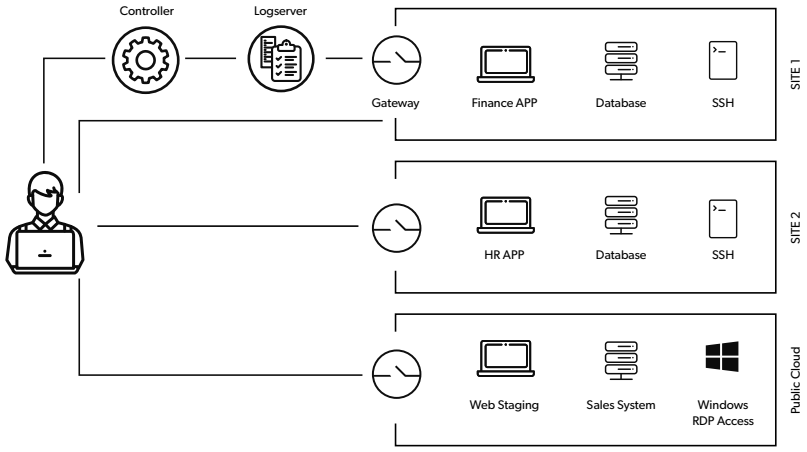
1. It's made up of a distributed set of components.
2. It leverages a virtual network driver on the user device to interact directly with the user when needed.

Appgate SDP Key Components

Appgate SDP has three core components: the Controller, the Client and the Gateway.

Client devices authenticate to the Controller, which evaluates credentials, and applies access policies (based on the person, environment and infrastructure). The Controller returns a cryptographically signed token back to the client, which contains the authorized set of network resources.

When the user attempts to access a resource—for example by opening a web page on a protected server—the network driver forwards the token to the appropriate cloaked Gateway, which then applies additional policies in real time—for example, to control access based on network location, device attributes, or time of day. The Gateway may permit access, deny access, or require an additional action from the user, such as prompting for a one-time password.



Additional Notes on the Architecture

The key Appgate SDP elements are deployed as one or more appliances, available in virtual or physical formats. As discussed below, multiple Controllers, Gateways and LogServers can be deployed to achieve high availability. Communications between appliances are secured using Transport Layer Security (TLS) to ensure integrity and confidentiality. With Single-Packet Authorization, enterprises hide all Appgate SDP services on the network from unauthorized users. This approach enables secure deployment of SDP on public-facing networks. The appliances themselves are based on a customized, hardened Linux distribution, which provides confidentiality, integrity, and availability.

Policies

The point of reference used in the design of Appgate SDP is the user. This should be a familiar concept for most enterprises who have been using Active Directory (AD) and group memberships for years. The big difference with Appgate SDP is that network access controls and a dynamic network topology is now centered on the user (and his or her device). This gives enterprises one consistent model which consistently secures all levels of the infrastructure. This is a great improvement over traditional security in which application security is based on directory group membership, while network security is based on location and access method.

How does a dynamic, identity-centric system work?

In traditional network security solutions, hundreds, if not thousands of static firewall rules are pushed to the firewall by administrators and made to go live. Like many static systems, these live rules often run forever. In fact, in many enterprises there is such a complex morass of firewall rules that no one truly understands "who has access to what" in the network. All it takes is one mistake and the adversary has a way in.

In contrast, in a world where everything is by default hidden on the network, there are no rules to push because the Zero Trust rule applies. In a dynamic identity-centric model the administrator creates the policies based around the user. Policies will generate rules that are dynamically enforced in the Gateways. Let's look at an example of what this means via three simple policies:

- Policy 1—Anyone on the sales team is permitted to access the CRM system.
- Policy 2—All customer support employees get call center access on weekdays during business hours.
- Policy 3—After a One-Time Password (OTP) has been used, sales managers can get access to the invoicing application.

When a salesperson connects and authenticates, his or her entitlement list for each gateway is used to establish a secure connection to the Gateway(s), and the client sets routes from an entitlement token or from a fully-rendered firewall ruleset. This is a very powerful and efficient model compared with the traditional firewall ruleset approach.

Policy Distribution

How do we get this entitlement package to the Gateway where the rules will be enforced? The process, not surprisingly, follows the Appgate SDP identity-centric model. When a user's device first connects to the network, the Appgate SDP driver transparently connects to the Controller, which issues a set of three cryptographically secured items to the Client. These items contain the information needed for establishing secure connections between the Client and each Gateway, and for enforcing the rules in real-time at each Gateway. The three items are:

1. The Claims Token—which stores user attributes (referred to as "claims").

The claims token is provided by the Controller to the Client on successful user authentication. The claims token is a cryptographically-signed file containing validated trusted claims (think of these as user attributes) related to the identity and context of the Client. Because this token is signed by the Controller, it allows each Gateway to trust its contents, even though it's indirectly delivered to the Gateway via the Client.

When the token is about to expire, the Client will automatically ask the Controller for a new token. The token will be automatically reissued by the Controller, and will contain the latest set of Client and user attributes.

2. The Entitlements Tokens—which stores the set of network resources that a user has permission to access, subject to real-time policies.

The entitlement tokens contain a list of all the user's entitlements (access rights) for each site. A site is a group of servers/network resources accessible through a particular Gateway (or Gateway cluster in a High Availability deployment). Entitlement tokens are created by the Controller as part of the Client authentication process. The Controller filters all policies to find all services that the user is entitled to, and replies to the Client with signed entitlement tokens - one per site. The Client sends the relevant entitlement token to the appropriate Gateway once it is connected.

This dynamic identity-centric approach means that user-specific policies are delivered to the policy enforcement point by the user's device. This may seem a bit counter-intuitive, but it's a key part of the Appgate SDP identity-centric model, since policies are distributed and enforced in a way which is consistent with the topology. In fact, as we'll see later, this architecture enables a very innovative approach to scalability and high availability. With the user in the center, our topology allows Clients to make connections to multiple Gateways at the same time. And because each Client holds all the tokens it needs, new connections can be made easily when required without needing to involve the Controller.

3. The Client Certificate– which is used to identify the user and the Client to Gateways.

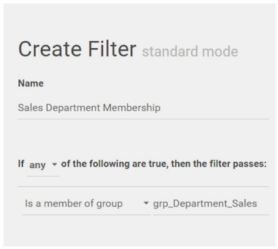
The Client certificate is generated after installation of the virtual network driver, to identify the user and the device. The virtual network driver uses this certificate to create mutual TLS connections to both Controller and Gateways. This certificate is only used to establish a secure tunnel, not for giving access to any resource. Remember, Gateways will not permit access to any resources unless the Entitlement token permits it.

To generate the Client certificate, the Client generates a public/private key pair and requests a signed certificate from the Controller based on its claims token and public key. The Client certificate will uniquely identify the user/client/device combination, and is used to prove its identity when talking with Controllers and Gateways.

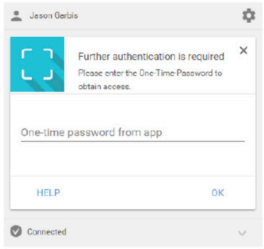
Now that we've explained the key items issued by the Controller, let's explore some other aspects of the system in detail.

Policy Enforcement Types

Policies are at the heart of Appgate SDP's network access control, since they're the vehicle by which security teams determine which users get access to which resources. Unlike traditional network access controls, which use static rules that control access on an IP-address basis, Appgate SDP relies on two rich and dynamic layers.First, when users authenticate to the Controller, the system collects a set of verified user attributes—the claims. These attributes are a collection of information about the user (attributes or group memberships obtained from directory or identity systems), the device (obtained by the Appgate SDP driver running on the device), and about the overall context (for example: the time of day, user's authentication method or other attributes). Combined, these claims make up a rich user context on which administrators build policies. These policies are enforced at two points in time:

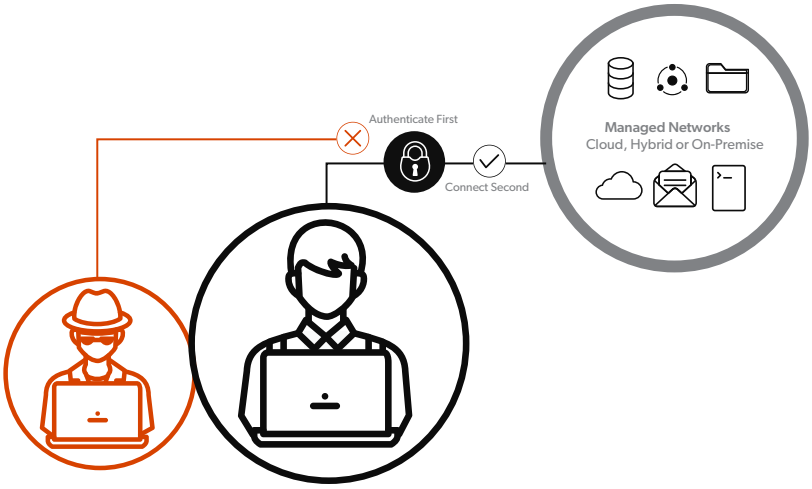


Authentication



Access

It takes an "authenticate first, connect second" approach, ensuring that only authorized users can connect over an encrypted connection to network resources. This reduces the attack surface and significantly improves security.



Authentication Filters

Filters are evaluated at authentication by the Controller, and determine the maximum set of entitlements that a user may have access to, subject to the access-time conditions described below. Filters should be used to evaluate user and environmental attributes that are relatively static, and aren't expected to change on a daily (or hourly) basis. For example, directory group memberships, directory attributes, and user tags (assigned by association with an Appgate SDP identity provider).

These filters are the primary way that users are given entitlements to access network resources (discussed in detail below, in the Entitlements section).

Access Conditions

The second level of enforcement occurs at the Gateway, and are real-time conditions that are evaluated at the time the user is attempting access. Conditions must be true for the user to obtain network access to the target resource, with the specific entitlement (e.g. network protocol, port, and host).

Conditions are suitable for attributes that are expected to be dynamic, and change regularly. Examples of conditions include: time of day, network location, user device login information, and authentication strength.

What is perhaps most interesting about conditions is that they support the concept of Remedy Actions. If a condition fails, the policy can be configured to prompt users to remedy the condition, at the time of access.

One common use of this is to enforce step-up authentication when a user attempts to access a higher-risk resource. Let's say the user has logged in with just a username/password combination, but is attempting to access a funds transfer application. Security policy may require the entry of an OTP.

Appgate SDP will automatically prompt the user before network access is granted.Appgate SDP also supports remedy actions that prompt for a text explanation—effectively recording the reason that the user needs access for audit purposes. Other types of Remedy Actions include prompting for a password, or notifying the user about why access is denied (which can be helpful in eliminating potential user frustration).

Entitlements

At authentication, the Controller issues a cryptographically-signed set of Entitlement tokens to the Client, which contains the set of resources (entitlements) to which the Client is permitted access (subject to policy enforcement as described above).

In Appgate SDP, entitlements define the network access to a target resource that a user may be permitted to have, plus any restricting conditions.

Network access includes: TCP access from the Client, reverse TCP access (connections initiated from the target to the Client), Internet Control Message Protocol (ICMP) access to different target hosts, or User Datagram Protocol (UDP) access. Take a look at some example Entitlements below, and then we'll explain some aspects that are unique to Appgate SDP:

- Client is entitled to TCP access to port 80 on server 10.0.0.1, if the time is between 09.00 and 17.00
- Client is entitled to TCP access to port 443 on all servers on subnet 10.1.0.0/24
- Client is entitled to send ICMP Type 0 (Ping) to server named loadtester.example.com
- Client is entitled to TCP access to port 22 on all AWS server instances with AWS Security Group AcmeProduction
- Client is entitled to (UDP) access to all ports on server 10.1.0.4

These entitlements don't say anything about the user. Recall that users are determined by the authentication-time policies (Filters) described above. This keeps the entitlements simple. Notice that while some policies identify specific hosts by name or IP address, there are some that use dynamic resolution—specifically for servers running within Amazon Web Services EC2 (AWS). In this case, the Gateway will automatically detect new (or terminated) server instances, and adjust user access based on AWS Security Group or Tag. This eliminates the need to manually adjust access for each server instantiation, and avoids the typical coping mechanism of simply granting users overly broad network access.

Identity Providers

Appgate SDP's identity-centric and context-aware model requires support for multiple identity providers. Identity providers are used as a source of user authentication, as well as user attributes (claims). Appgate SDP supports multiple concurrent identity providers, which is useful when an organization must support multiple disparate user populations (such as employees and third-parties), and supports LDAP, RADIUS, and locally-defined users. Because users are automatically tagged by their source identity provider, it's simple to define policies that precisely control access for different user populations. For example, users from a third-party hardware maintenance vendor could have their own identity provider, and be limited to accessing those specific machines for which they must perform remote maintenance—all through a simple policy.

Tunneling

Clients obtain their individualized and secure network segment of one through a secure encrypted tunnel. This tunnel is created from the Client's virtual network adapter (we often refer to this as a tunneling driver) to each Gateway. (Clients also have a secure connection to the Controller, but that connection is not used for communication to any target system). The connection from each Client to each Gateway is secured by either TLS (for TCP connections), or Data Transport Layer Security (DTLS) (for UDP connections).

Scalability

Appgate SDP is architected with scalability in mind, across several dimensions. First, each Gateway is designed for multiprocessing and is highly multi-threaded. The more cores available in a Gateway, the more traffic it can handle. The Gateways create a separate rules engine for each client/device connection, which is load-balanced across the available network daemons—one for every CPU core. This scales much better than having one monolithic daemon process spread across multiple cores, and allows an optimized utilization of the hardware acceleration like AES-NI and HMAC which is available on every core. By adding more cores, the Gateway has more network daemons it can use, which offers linear scalability.

This dynamic identity-centric approach continues inside the network daemon itself. As opposed to having one enormous, consolidated ruleset, Appgate SDP has a huge performance advantage by using a per-client rules engine. In a traditional firewall, each connection request is parsed against the full ruleset and any that apply are utilized. Because Appgate SDP can very easily link each connection (user) to that user's rules, we only need to parse a tiny ruleset for each user. It is entirely fitting to run each Client's rules engine in its own thread. This thread performs all the encryption functions and firewalling necessary for that user's connection. This model is immensely scalable and offers the ability to deploy dynamic access controls on a scale not possible with traditional architectures.

Gateway clusters are linearly scalable. Adding new Gateways incurs no performance overhead due to the absence of inter-Gateway communication or state synchronization. This permits Appgate SDP deployments to scale to meet even the largest environments' performance requirements.

High Availability

Traditional network security appliance clusters are typically limited to two or three nodes to keep the state synchronization within limits. Appgate SDP has an innovative approach to High Availability (HA), which nicely dovetails with the scalability approach, to deliver reliable failover without loss of client connection state.

Appgate SDP accomplishes this by leveraging our unique, identity-centric, context-aware model. Instead of incurring the overhead of synchronizing state between Gateway instances for a site, Client state is maintained at each Client. Each Gateway synchronizes the Client state with the Appgate SDP Client network driver. This adds very little network traffic, since each Client's connection state is minimal—on the order of 100 bytes of data.

In the event that one Gateway becomes unavailable, the Client simply connects to an alternative Gateway, and sends its tokens and state. There's no need to reconnect with the Controller or to reauthenticate. The tokens contain everything to set up a trusted connection, with current access rules. In this regard—because state is fully synchronized with the Client—the Gateways are scalable and recoverable as if they were stateless. The Controller is also stateless, so if a Controller goes down, existing users are unaffected.

Built for the Cloud: Protecting IaaS

As challenging as network security is in traditional enterprise IT environments, applying network access controls for the cloud is even harder. 60% of security professionals acknowledged that their teams can't keep up with the pace of cloud automation, self-service and DevOps changes. And, many organizations have multiple cloud platforms in operation—such as an on-premises virtualized environment along with a public cloud-based IaaS platform.

Appgate SDP is designed to solve this problem consistently, and to easily handle the dynamic nature of cloud environments. In particular, Appgate SDP supports the concept of automated name resolution that leverages Cloud Security Groups, and Tags.

Amazon EC2

Within EC2, customers can either use Tags or Security Groups to categorize server instances. The information within the EC2 Tags is purely declarative (that is, it has no semantic meaning or impact on access controls). EC2 Security Groups do control network access, but at a very coarse-grained level (e.g. by source subnet, or source IP address, which might map to dozens or hundreds of users in the case of a NAT environment).

Appgate SDP supports fine-grained, adaptive access control to Amazon EC2 resources through the use of a name resolver. By automatically detecting new server instances, and intelligently combining EC2 Tags and Security Groups with user context, Appgate SDP will automatically adjust user access to these new instances.

For example, within an Entitlement definition, you can specify the AWS Security Group or Tag instead of a specific hostname:

- aws://security-group:FinanceProduction
- aws://tag:Department=QA

For both of these examples, Appgate SDP will, via the AWS API, resolve all server instances that match, and issue entitlements for users (subject to filters and conditions as usual, of course). More importantly, the Gateway will dynamically and automatically detect new server instances within AWS, and adjust user access appropriately. Because the policy enforced within the Gateway uses this dynamic resolution, Clients will automatically be granted access to newly instantiated servers, without requiring any changes or communication from the Controller.

Conclusion

Appgate SDP is a distributed, dynamic and scalable Zero Trust Network Access solution for fine-grained network access control. It draws on user context to dynamically create a network segment of one that's tailored for each user session and hides all network resources—servers, services, and applications—except those that the user is authorized to see. By making the rest of the network invisible, enterprises can simplify their security infrastructure, while granting access with confidence. Appgate SDP provides real-time, context-aware access, enforces the principle of least privilege, and easily controls access while maintaining a strong security stance.

About Appgate

Appgate is the secure access company. We empower how people work and connect by providing solutions purpose-built on Zero Trust security principles. This people-defined security approach enables fast, simple and secure connections from any device and location to workloads across any IT infrastructure in cloud, on-premises and hybrid environments. Appgate helps organizations and government agencies worldwide start where they are, accelerate their Zero Trust journey and plan for their future. Learn more at appgate.com.

appgate