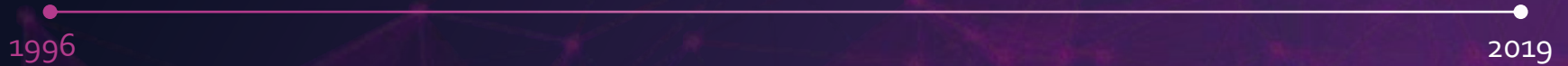
AppGate SDP Introduction

Basic architecture and key concepts

Cyxtera™

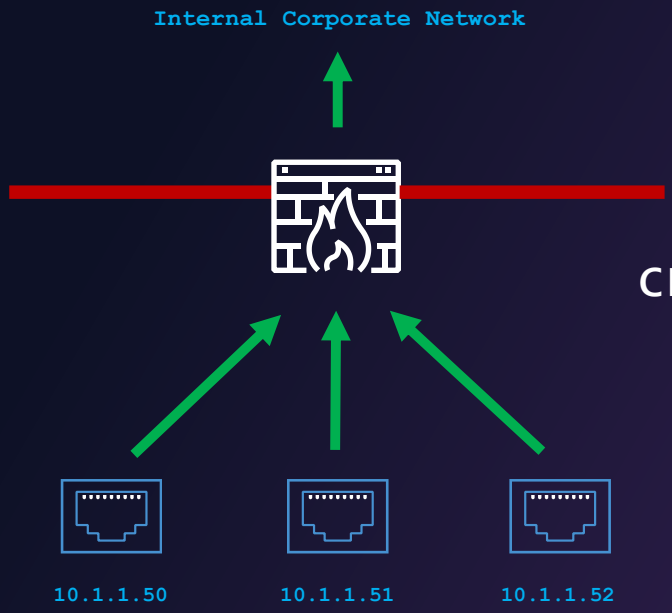
Yesterday's network security doesn't work for modern I.T.



Perimeter security has remained largely unchanged for the past 2 decades.

I.T. has moved on – distributed, dynamic and interconnected.

Hybrid I.T. has killed the perimeter.



CENTRALIZED VS HYBRID

Firewall rules are binary and static.

Should this IP block have access to this network (Y/N)?

But today's business isn't.

Is Jim's machine patched?

What's the current security posture?

Where is he?

What time is it?

What project is Jim working on?

What are his credentials?

Should Jim have access to the production SAP database server?

A better approach to network security:
Software-Defined Perimeter

Identity-centric

User- or device- based access control

Integrates with directory services and IAM

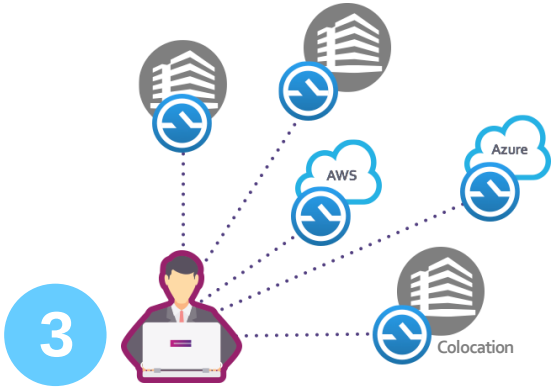
Context sensitive

Zero-trust model

Authentication before connection

Dynamically-provisioned 1:1 connectivity

Unauthorized resources completely dark



Built like cloud, for cloud

Distributed, stateless and highly scalable

Programmable and adaptive

Dynamic and on demand

AppGate creates a "Segment of One"

Fine-grained controls reduce attack surface

1:1 ENCRYPTED NETWORK SEGMENT

Dynamic, 1:1 network segment

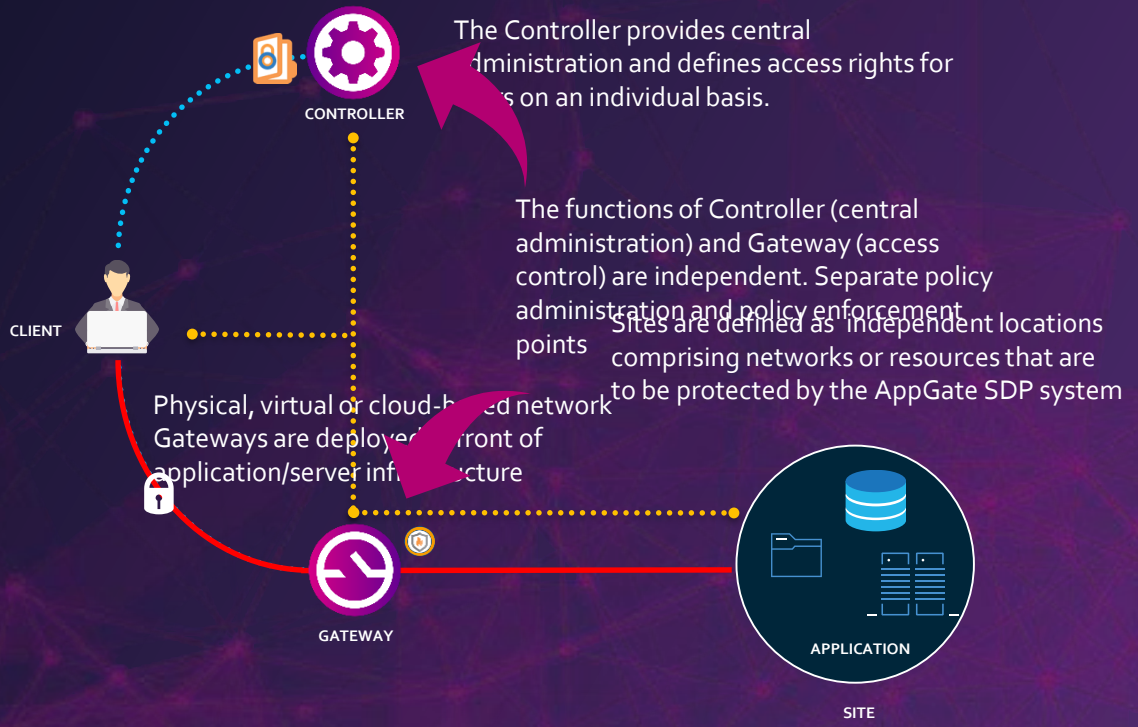
- Encrypted for greater security
- User sees only their authorized resources
- All other resources completely dark, inaccessible
- Entitlements adjusted in real time as necessary

AppGate SDP Components

- **Client**
  - Runs silently, runs everywhere: win/mac/linux/android/ios
- **Controller**
  - Authentication, Authorization, Policy Management, Identity Provider Integrations etc.
- **Gateway**
  - VPN Server, Firewall, Real-time enforcer
- **LogServer (optional)**
  - Store, parse, query, visualize audit logs

How AppGate works

1. Client makes access request to the Controller (the Policy decision point). Typically linked with Identity Providers. Contains the Policies that define Entitlements (access rules)
2. Controller checks context, authenticates and authorizes user verifying any number of claims within each session and passes the Entitlements token to the Client
3. The Client passes the token on to the Gateways which provision a micro-firewall instance started on a separate thread just for that user. The Gateway then translates the Entitlements into a set of individualized micro-firewall rules
4. Dynamic **Segment of One** network is built for this session. For each packet received from the Client, the correct rules allow, conditionally allow or block access.
5. AppGate continuously monitors for any context changes, adapts user access accordingly



AppGate SDP 8 key concepts

Cyxtera™

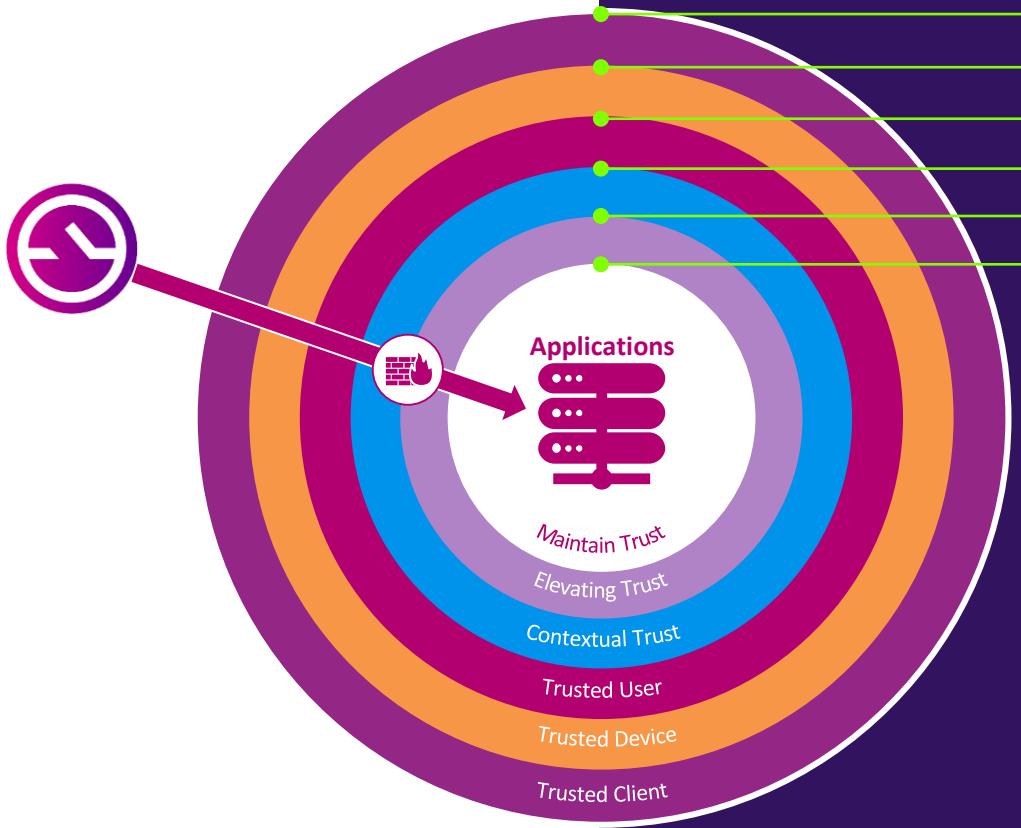
AppGate SDP – The 8 key concepts

- Zero-trust defense
- Hyper-scaling
- Robust, easy to deploy architecture
- Highly available – always on connectivity
- Designed around the user
- Dynamic resource resolving
- Adaptive authorization
- Role based API driven administration

Zero-trust defense

6 LAYER TRUST MODEL

1. Single packet authorization
2. Only on-boarded devices
3. Validate user's identity
4. Context defines access rights
5. Elevate trust if required
6. Monitor user's behavior



Robust, easy to deploy architecture

Tokens are used to pass information between the Controller, Client and Gateway. They contain all the information needed for authentication, authorization and real-time access control.

Appliance based – physical, virtual or cloud instances

Controllers - stateless

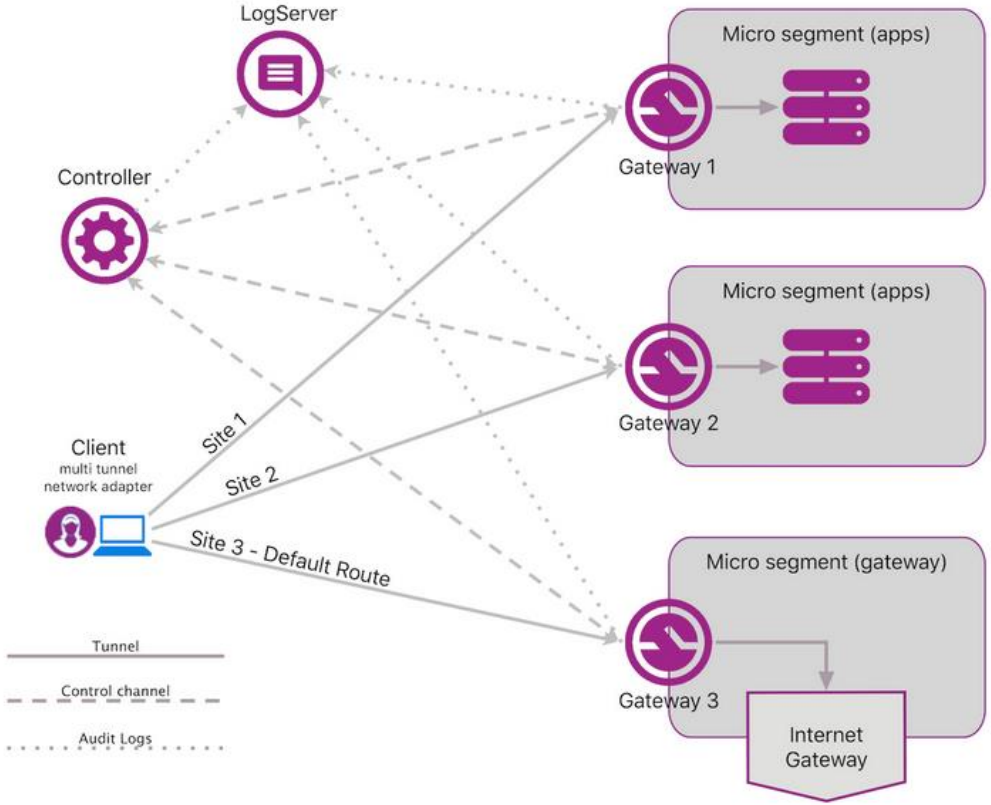
- Secure token issuing service
- Client passes-on renewable tokens
- Rebooting Controller won't affect existing tokens/users

Gateways - no pre-set firewall rules

- No real-time Controller <> Gateway communication
- No Gateway <> Gateway communication
- Runs a separate firewall service for each user

Designed around the user

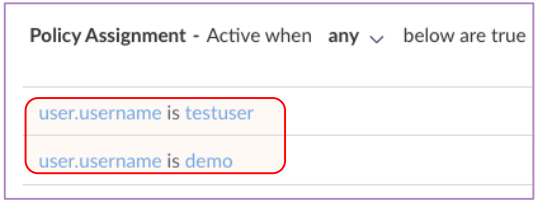
- AppGate SDP works by creating direct one-to-one TLS/DTLS connections between users and the Sites they need to access – multi tunnel network adapter
- Functions of Controller (central administration) and Gateway (access control) are independent
  LogServer can also be added
- Concept of Sites allows Gateways to be deployed to protect any group of target hosts, defined subnets

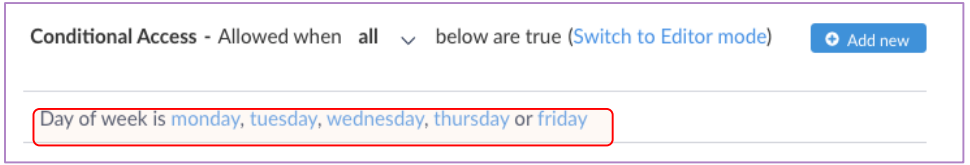


Adaptive authorization

Claims are key-value pairs that relate to the identity and context of the user and device. 3 types of claims:

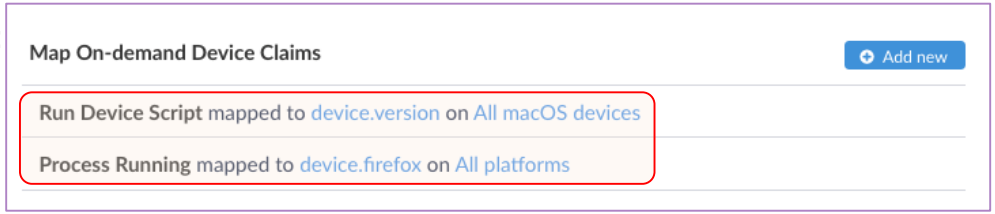
- User: often static claims such as username from the IdP
- Device: often dynamic claims such as IP address from the connecting device
- System: often dynamic claims such as country code from the Gateway





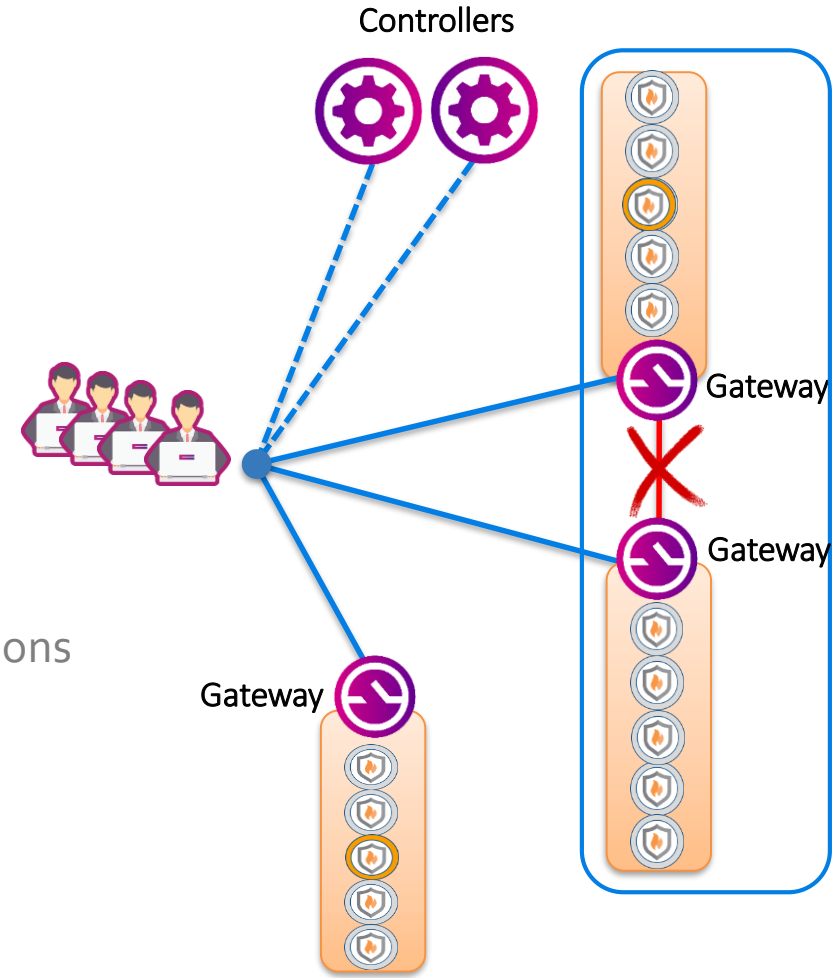
Two classes of availability within the system:

- Fixed: set by the system - will always be gathered.
- On-demand: configured by the admin



Hyper-scaling

- Multi-Controller group (up to 6 currently)
- Multi-Master Controller
  - Background replication
- Support for multiple sites
- Multiple Gateways per site
- HA Gateways
- Multiple per-user firewall services
- No wasteful Gateway <> Gateway communications
- Multi-tunnel network adapter
- Autonomous clients



Highly available – always on connectivity

Client auto-connects in the background

Gateway syncs states once connected:

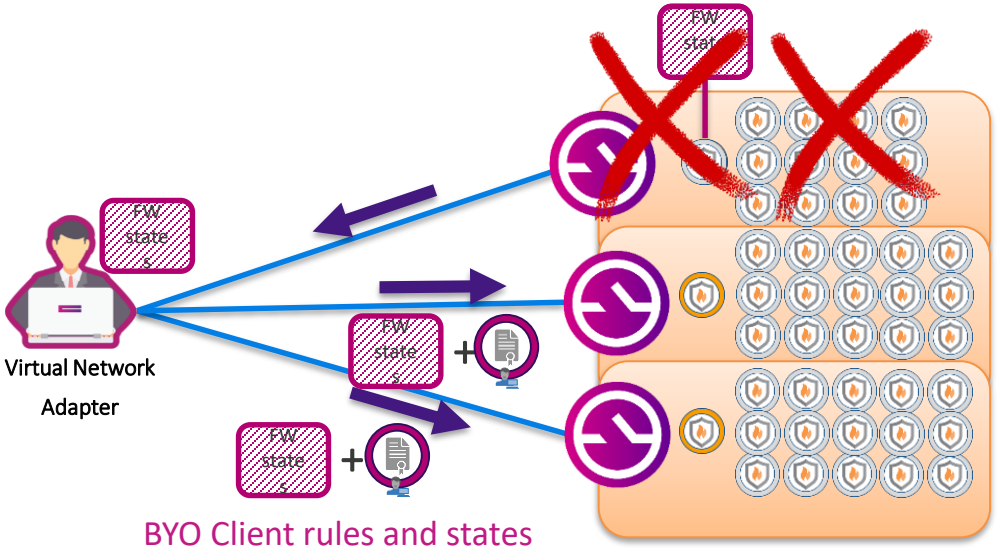
- Tiny states – can be synced to client
- States signed - tamper proof
- States sent - to device memory

Gateway fails:

- Tokens & states transferred to any other member of Gateway pool

Same model used for roaming:

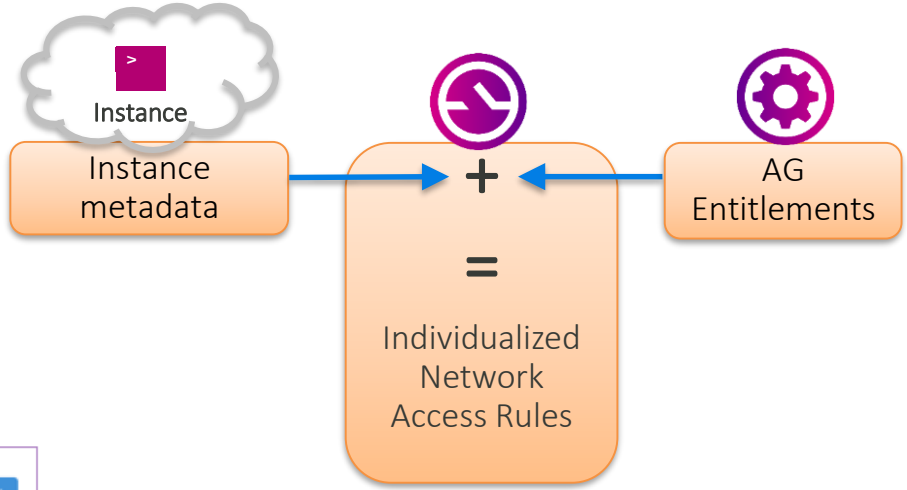
- Moving from 4G to Wi-Fi states are sent to any Gateway (even the original) when connectivity is re-established (by OS)

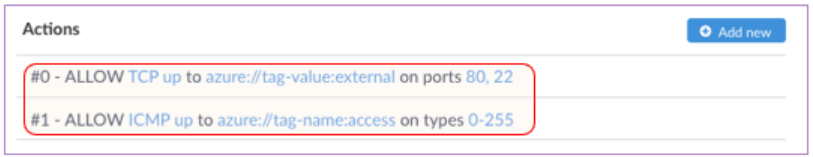


Dynamic resource resolving

Managing static firewall rules in today's fast moving environments is next to impossible!

- Multi site deployments
- Failover sites
- IP addresses changes
- Dev-Ops
- Auto-scaling
- Life cycle management





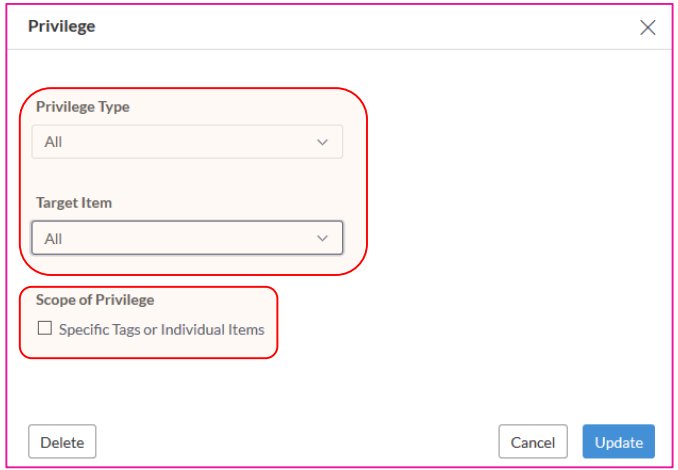
Role based API driven administration

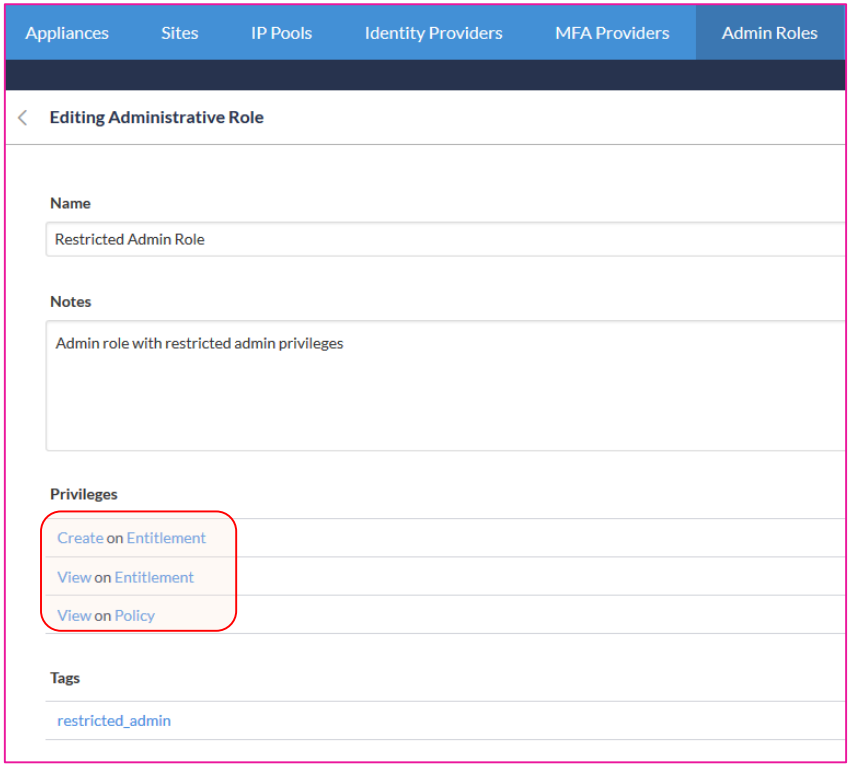
Delegated – multi role admin

Multi tenant / Multi Site

API based for dynamic configuration

Type and Target define the exact privilege of an Admin Role. Tags limit the scope of the privilege.





AppGate SDP Introduction

The end…