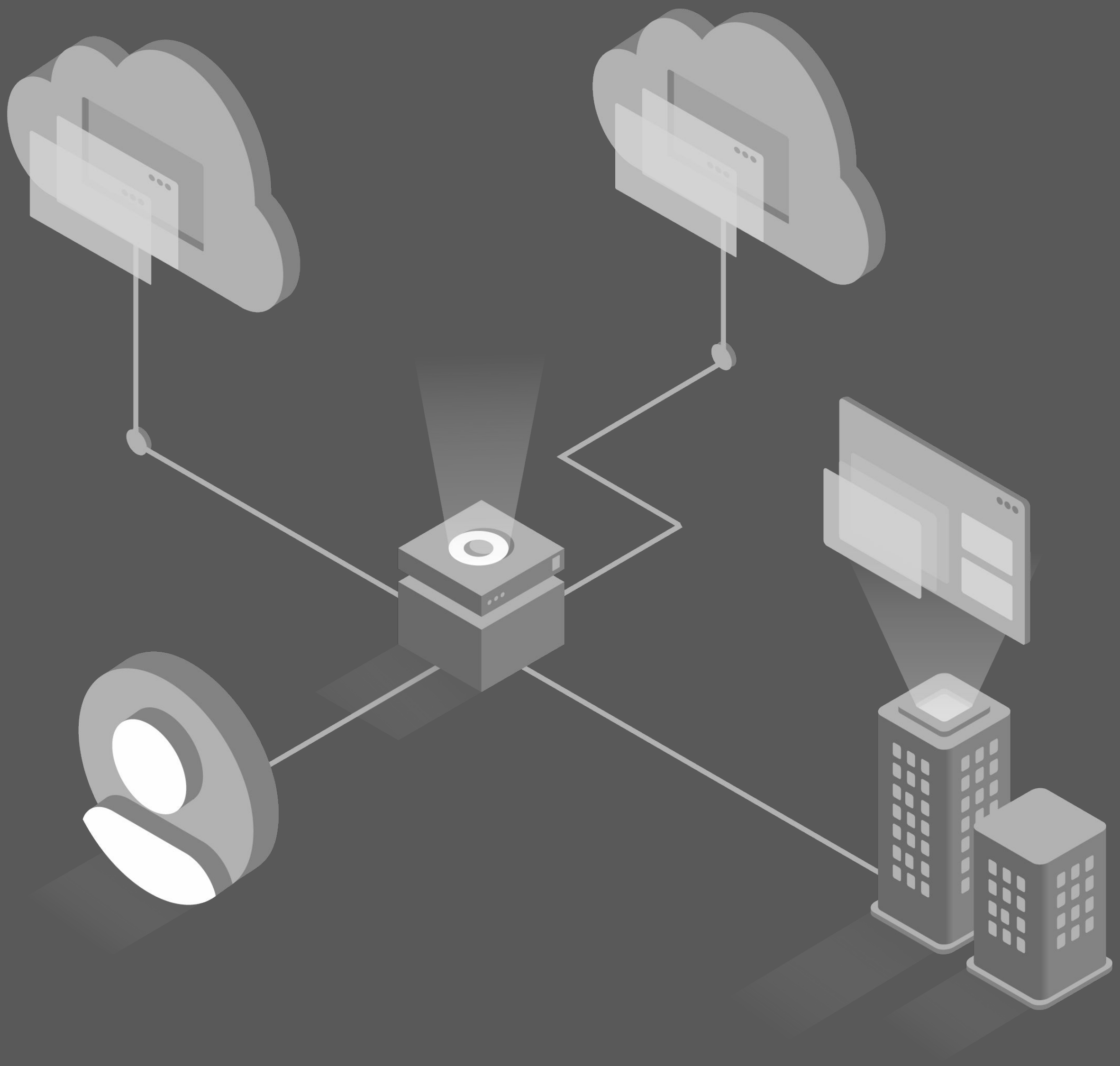
CA SiteMinder

Migration Guide



Executive Summary

Securely connecting users to applications is not a new problem. To address this challenge, many organizations adopted CA SiteMinder and supporting services like CA Advanced Authentication, and CA Directory to secure access to on-premise web applications.

With the emergence of new types of apps and use-cases, enabling access to all systems and users broke the CA model, which runs on-premises, is hard to operate and upgrade, requires manual tasks for adding MFA and integrating with SaaS apps, and cannot deliver security cost-effectively.

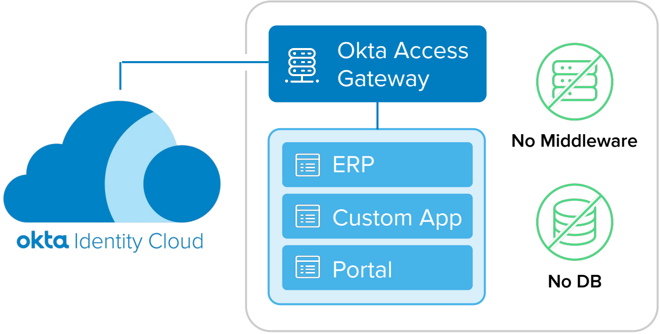
CA Siteminder Challenges

- Costly to Operate
- Hard to manage
- Hard to integrate security and SaaS
- Hard to improve security posture

Okta Access Gateway

Okta Access Gateway delivers Okta SSO and MFA from the cloud to your on-prem web apps, replacing on-prem SSO (CA), complementary servers like CA Advanced Authentication, and the underlying infrastructure, while supporting modern requirements.

The solution has a simple architecture that does not require additional middleware – admin and policy servers – and database – policy and user store – servers to operate.

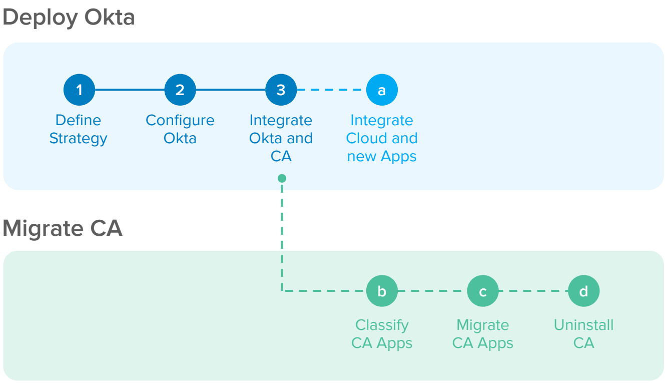


Access Gateway: Conceptual Architecture.
No additional databases and middleware required

CA SiteMinder Migration

The migration from CA SiteMinder to Okta is executed in two steps. In the first step, you deploy Okta and deprecate CA SiteMinder while providing universal SSO and MFA for all apps: from ground to cloud.

In the second step, you classify, migrate apps to Okta, and then uninstall CA SiteMinder and supporting servers:

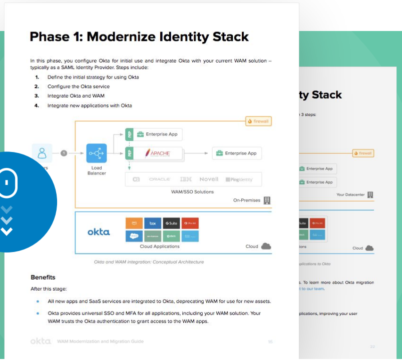


CA SiteMinder Migration Milestones

What's Next

For more details on how to migrating from CA SiteMinder to Okta, **continue reading this guide.**

For a tailored plan on how to migrate to Okta, reach us at www.okta.com/contact-sales



Phase 1: Modernize Identity Stack

In this phase, you configure Okta for initial use and integrate Okta with your current WAM solution – typically as a SAML Identity Provider. Steps include:

1. Define the initial strategy for using Okta
2. Configure the Okta service
3. Integrate Okta and WAM
4. Integrate new applications with Okta

Okta and WAM Integration: Conceptual Architecture

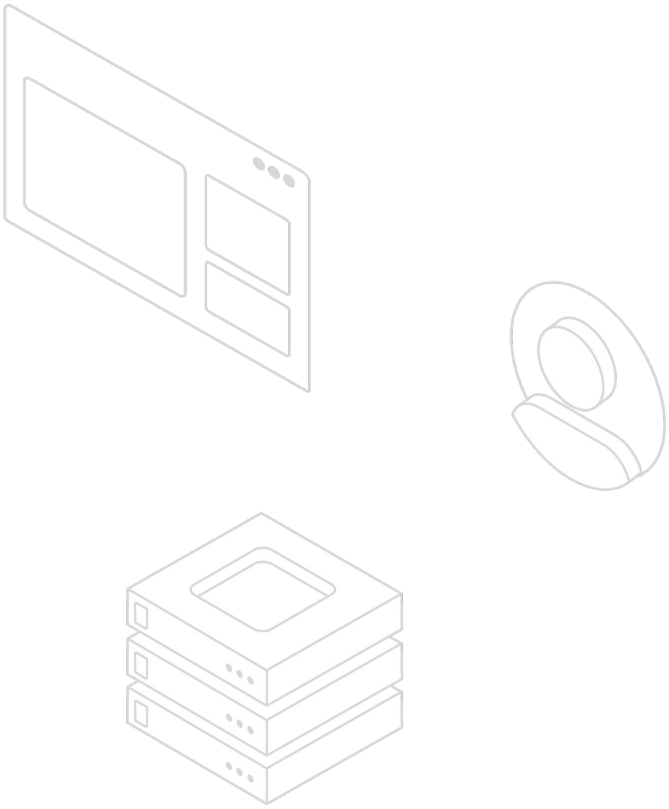
Benefits

After this stage:

- All new apps and SaaS services are integrated to Okta, deprecating WAM for use for new assets.
- Okta provides universal SSO and MFA for all applications, including your WAM solution. Your WAM trusts the Okta authentication to grant access to the WAM apps.

okta WAM Modernization and Migration Guide 16

Index



Introduction

Securely connecting users to applications is not a new problem. To address this challenge, many organizations adopted CA SiteMinder and supporting servers like CA Advanced Authentication and CA Directory to secure access to on-premise web applications.

As technology evolved, organizations adopted new solutions that require modern access control. From SaaS, to mobile access, to Infrastructure as a Service, and Single Page Apps (SPAs), securing access to these systems at scale challenges the CA SiteMinder security model, which requires a heavy infrastructure on-premises and private networking, is expensive to update, and cannot deliver cost-effective security:

> *"By 2022, IDaaS will be the chosen delivery model for more than 80% of access management deployments globally"*
>
> Gartner
>
> *Gartner's Magic Quadrant for Access Management, Worldwide 2018*

> *"The biggest benefit of using IDaaS compared to on-premises IAM solutions is a 30% to 35% lower ongoing maintenance rate"*
>
> FORRESTER
>
> *Forrester Wave: Identity-As-A-Service, 2017*

Okta Identity Cloud delivers modern identity as a service solution that addresses the requirements of today and tomorrow. Okta securely connects users to any web application. This includes on-prem web apps traditionally protected by CA SiteMinder as well as SaaS, mobile, modern, and IaaS apps, servers, and many more IT resources that cannot be easily protected by CA SiteMinder.

This whitepaper describes the milestones and best practices for migrating from CA SiteMinder to Okta. For guidance and a tailored plan on how to migrate from CA SiteMinder to Okta, reach us at

www.okta.com/contact-sales

CA SiteMinder: Architectural Limitations

CA SiteMinder focused on just a limited part of today's IAM challenges. It is designed as an on-premise Web Access Management (WAM) software primarily built to provide password-based Single Sign-On (SSO) to on-premise web apps. It does not include native support to MFA, user directories, user self service, user registration, user provisioning, and just-in-time (JIT) account creation – these capabilities require additional CA products with their own architectures, installation, and upgrade cycles. These additional components also require manual integrations into CA SiteMinder using scripts, custom code, and shims.

Some of the most common CA SiteMinder challenges include, but are not limited to:

- **Upgrade complexity:** Upgrades – including major releases, service packs and hotfixes – can take months or even years to deploy into production. Extensive testing is required when performing an upgrade because these upgrades are known to often break other features and functions. CA SiteMinder has more than 60 third-party libraries built-in – each of which can have their own security vulnerabilities, but the only way to fix those is to upgrade the entire solution.
- **High-Availability (HA) complexity:** The HA configuration requires multiple connections to be opened, configured, and monitored. Scripting must be done to determine a policy server status, and policy server will try to start serving requests as soon as they startup without allowing an administrator to first test the recovered server. In HA environments, the Siteminder registry file must be manually copied from one policy server to another to ensure they are synchronized. The HA configuration often means that multiple log files have to be merged together from multiple servers to determine what a user session accessed.
- **Administration complexity:** The system is hard to manage, with a cumbersome administrative UI to setup and connect to SiteMinder. Also, there's a risk of Policy Store corruption when multiple IT Administrators are using the UI in parallel. While it is easy to see which users have access to a single application, the only way to get a list of apps assigned to a user is to iterate through each application in the system. The only way to determine which apps are still running is through detailed log analysis or the Agent discovery feature which requires additional components to be installed – session server, and is known to cause policy store corruption when enabled.
- **Lack of pre-built integrations:** SiteMinder lacks a catalog with pre-built integrations for SaaS, PaaS, IaaS, and Mobile apps. Administrators should instead use "run books" for approx 60 SaaS apps. Most run books were written for SiteMinder 12.52 (now end of life) and have not been updated for newer releases of SiteMinder and SaaS updates, or for new SaaS apps.

- **Proprietary SDKs:** SiteMinder offers Software Development Kits (SDKs) for creating and embedding custom agents in apps, however the SDKs are limited – to Java and C – and use a proprietary protocol for communicating with Policy Servers. SiteMinder lacks SDKs based on open-standards – such as SAML or OpenID Connect, does not provide SDKs for the most popular programming languages and frameworks – like Go, Android, iOS, SpringBoot, AngularJS, and ReactJS – used in microservices and emerging app architectures.

- **Security Gaps:** SiteMinder uses a single session for all apps, which can be stolen and replayed unless the Session Assurance infrastructure is deployed and configured. Timeouts are set according to the app the user first logs into and can only be set to the application the user is currently visiting with complex changes to the policies. SiteMinder cookie providers often look like a Cross-Site Request Forgery (CSRF/XSRF) to the organization. There are no modern security controls such as WebAuthn, Geo-Fencing, and detection of Tor exit nodes (Darknet)

- **Fragile Configuration:** Despite being positioned as an enterprise class WAM solution for the past 20 years, SiteMinder can be fragile. Updating one setting can oftentimes have unintended consequences for another feature and many features are incompatible with each other. Due to the synchronous nature of the connections between Agents, Policy Servers, and the User Store, a small degradation on network or user directory performance can significantly reduce the SiteMinder service throughput, resulting in queuing and replaying of requests - sometimes referred to as the “Spiral of Death”. There is no isolation of custom code – i.e., custom Auth Schemes, Active Responses, Active Policies, Active Rules, Assertion Generator plugin-ins, and Message consumer plugins – inside the Policy Server. The run of non-optimized code can cause memory leaks and crashes in the Policy Server, affecting the overall service. With these limitations, providing SSO with High Availability require multiple servers and databases for processing and storing policies, users, auditing, private keys, and MFA configuration.

Addressing these requirements with CA SiteMinder – and its underlying agent architecture – is both **cost-prohibitive** – requiring licensing, deploying, and management of multiple server components such as Admin Servers, Policy Servers, Policy Databases, CA Directory, CA Advanced Authentication, and SMS gateways – **or not possible** – CA SiteMinder does not provide key cost-saving capabilities such as an app catalog with thousands of integrations, modern security controls, and automatic updates.

Okta Identity Platform: Overview

The Okta Identity Platform is an Identity as a Service (IDaaS) solution that provides Single Sign-On, Adaptive MFA, User Directory, Account Provisioning, Server Management, and API Authorization from the cloud. The platform is:

Globally available, 100% multitenant, stateless, and redundant

Regularly updated with security enhancements and new features

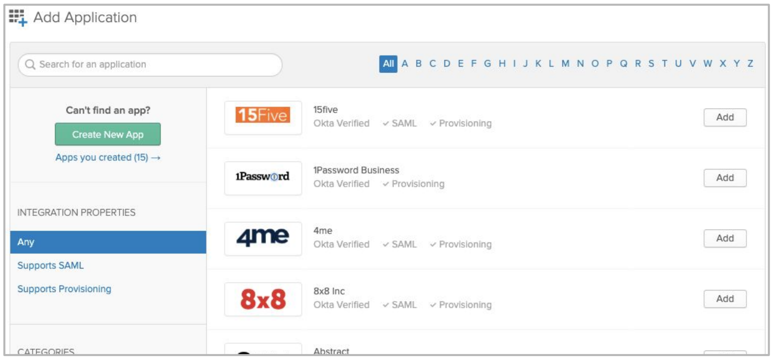
Built with a zero planned downtime architecture: the service is updated live, without scheduled downtime

Built to support apps regardless of where they are hosted: from ground to cloud

The Okta Identity Platform is capable of supporting both modern and enterprise applications via:

Okta Integration Network

The Okta Integration Network is a catalog of 6,000+ out-of-the-box integrations plus step-by-step instructions to connect your users to any technology while avoiding vendor lock-in.



Add Application

Search for an application

All A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Can't find an app?

Create New App

Apps you created (15) →

INTEGRATION PROPERTIES

Any

Supports SAML

Supports Provisioning

CATEGORIES

15five
Okta Verified ✓ SAML ✓ Provisioning
Add

1Password Business
Okta Verified ✓ Provisioning
Add

4me
Okta Verified ✓ SAML ✓ Provisioning
Add

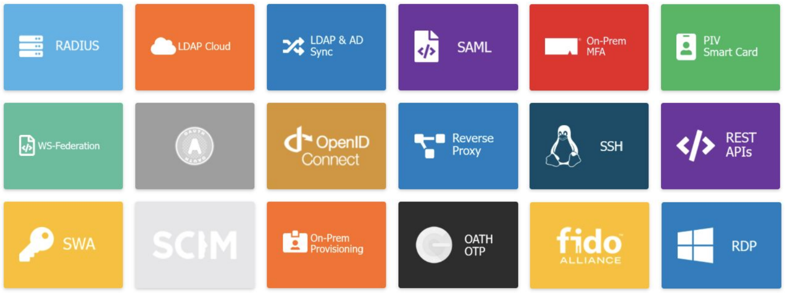
8x8 Inc
Okta Verified ✓ SAML ✓ Provisioning
Add

Abstract

Okta Integration Network Catalog

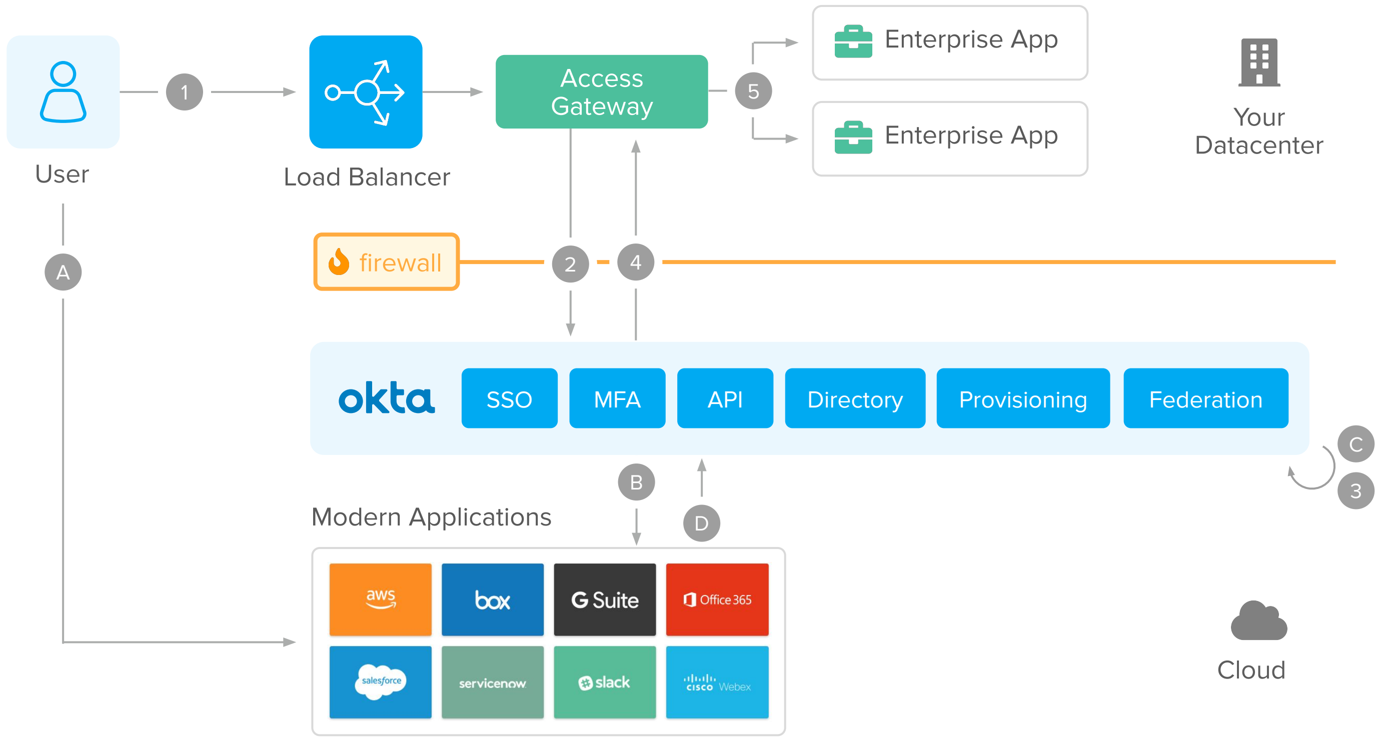
Integration Patterns

Okta provides Native support for the 18 top open standards and patterns that allows you to protect the most complex Hybrid IT environments without using multiple identity solutions.



Integration Patterns supported by Okta

The Okta Identity Platform architecture works as follows:



Okta: Conceptual Infrastructure for Modern and Enterprise Applications

Modern (Cloud, SPAs, and Mobile) apps flow

A. User accesses a cloud or mobile application.

B. The app validates the session. If there's no session, it redirects the user for a federated authentication in Okta.

C. Okta authenticates the user using the native identity functions: Single Sign-On, User Directory Store, Adaptive MFA, and Federation.

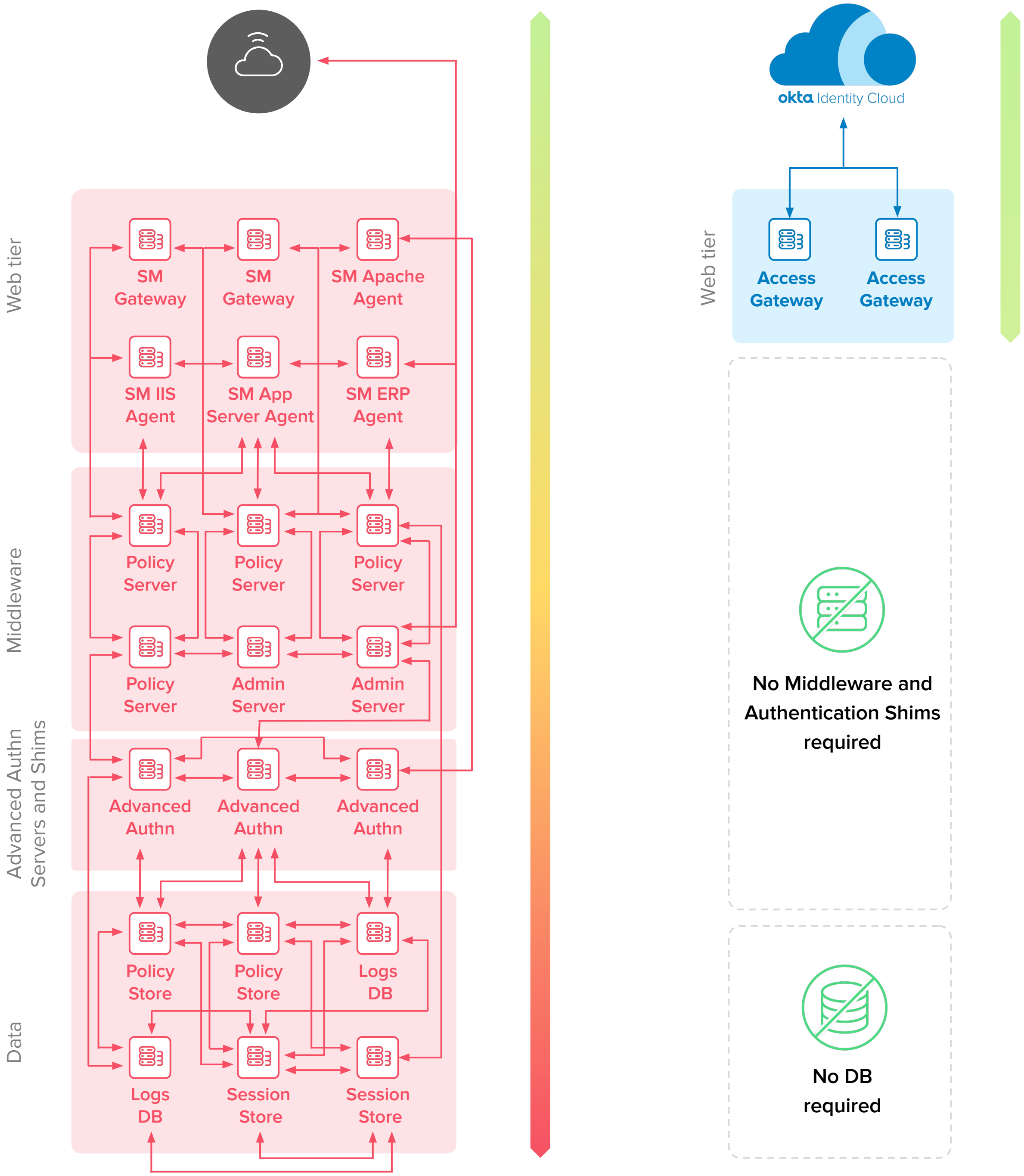
D. Upon access approval, the user is redirected back to the Cloud of Mobile App. The app completes the federation process, establishes the user session, and responds to the user request.

Enterprise/On-Premise apps flow

1. User accesses web application
2. Access Gateway intercepts the request – similarly to CA SiteMinder Gateway, App Server, or Web Agents. If the session does not exist, it redirects the user for a federated authentication on Okta.
3. Okta authenticates the user using the cloud identity functions – Single Sign-On, User Directory Store, Adaptive MFA, and Federation.
4. The user is redirected back to Access Gateway to establish the on-prem session, authorize access per URL – aka realms –, and sends information to the enterprise app via on-prem patterns like Header-Based authentication – aka responses, or IWA/Kerberos.
5. The enterprise app captures the user info, process, and responds to the request.

How Okta replaces CA SiteMinder

Okta replaces CA SiteMinder and delivers cloud SSO and Adaptive MFA to on-prem web apps via Access Gateway. Access Gateway supports the integration patterns natively used by on-prem web apps – such as Kerberos, IWA, responses (Header-Based authentication), and realms (URL authorization), replacing CA SiteMinder without requiring changes in code. The solution has a simple deployment model that does not require additional middleware and database servers:



Identity Management Infrastructure: Before and After Okta

How Okta delivers value beyond CA SiteMinder capabilities

Okta delivers native features beyond typical Identity and Access Management deployments. You can take advantage of these features to support new use-cases, improve your security posture, and return of investment.

This section list use-cases you can address natively with Okta beyond the CA SiteMinder capabilities and without requiring additional components – such as CA Identity Suite, CA Advanced Authentication, and CA Directory – or third party solutions – such as RSA or Symantec VIP:

Automate user onboarding & offboarding to applications and AD/LDAP directories.

Enable User Self-Service and avoid help desk costs with app requests, passwords reset, and account recovery provided out-of-the-box.

Automate management and provisioning of Office 365 accounts while avoiding PowerShell scripts, tickets, and manual intervention.

Consolidate AD domains and reduce AD footprint without implementing intermediate directories or data integration products.

Meet compliance requirements and secure access to VPNs, Virtual Desktops, and Network Applications with native Adaptive MFA capabilities

Secure access to DevOps Servers and Workloads with Advanced Server Access.

Secure Access to custom APIs, with API Access Management and Okta's OAuth and OpenID Connect APIs and SDKs for the top-10 programming languages.

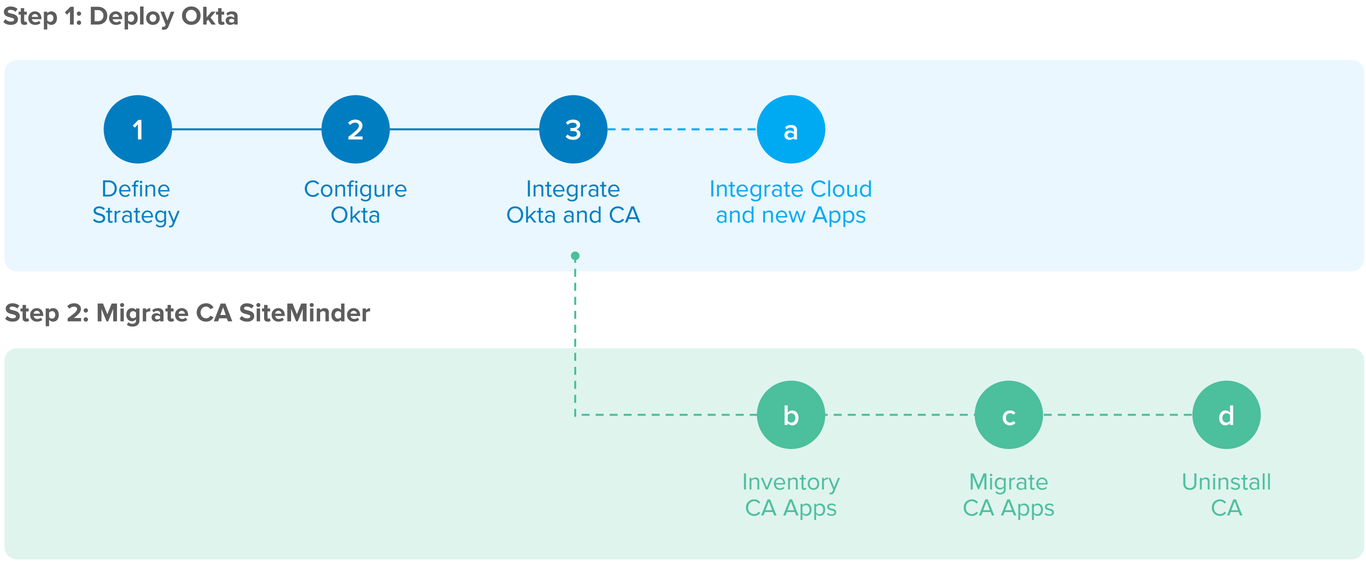
Provide access and SSO to all users – contractors, partners, and customers – from a single identity solution.

To learn more about the initiatives and projects you can accomplish with Okta, reach out to our team.

CA SiteMinder Migration Steps

As a modern identity solution, Okta can operate in co-existence with CA SiteMinder or as a single solution for all apps.

Okta is implemented through the following steps:



Okta Implementation Phases across different scenarios.

The implementation steps are:

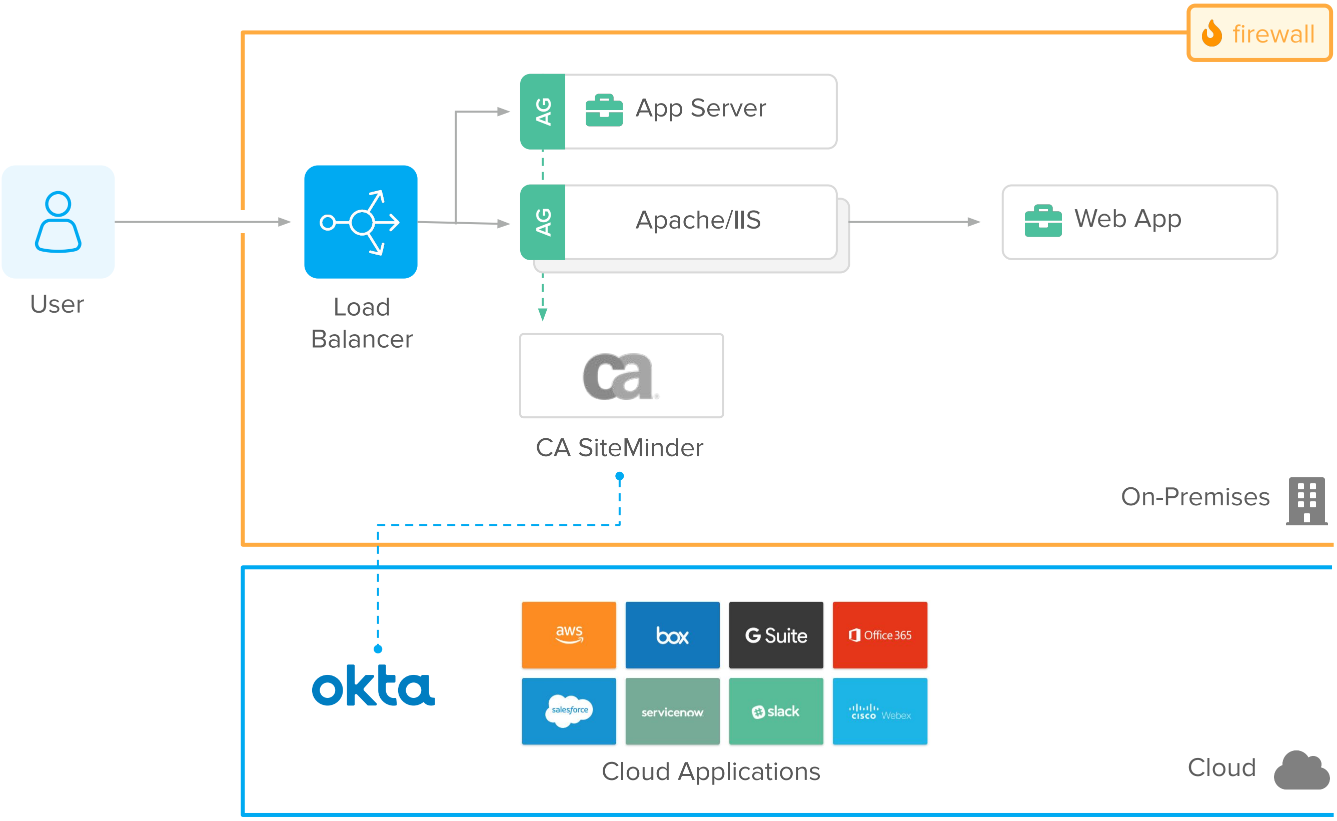
- **Deploy Okta**
  Define how you use Okta, configure the service, and integrate Okta with CA SiteMinder as an Identity Provider. After this step, all new applications and cloud (SaaS, PaaS, and IaaS) apps are integrated into Okta, deprecating CA SiteMinder for new integrations. Also, Okta provides universal SSO and MFA for all applications, including the ones protected by CA SiteMinder. Due to Okta's availability as a SaaS service and integration wizards, this phase is executed at a fast pace.
- **Migrate CA SiteMinder**
  In this step, you migrate from CA SiteMinder to Okta in 3 tasks: 1) identify and classify CA SiteMinder apps, 2) migrate these apps to Okta, and 3) uninstall CA SiteMinder.

The Okta migration is incremental. You can start with deploying Okta and migrating your CA SiteMinder at your own pace. Within each scenario, you improve your security posture and user experience while reducing your footprint and improving your Return of Investment.

Step 1: Deploy Okta

In this step, you configure Okta for initial use and integrate Okta and CA SiteMinder – with Okta as the SAML Identity Provider. This step includes:

1. Define strategy for using Okta
2. Configure the Okta service
3. Integrate Okta and CA SiteMinder (**optional**)
4. Integrate new applications with Okta



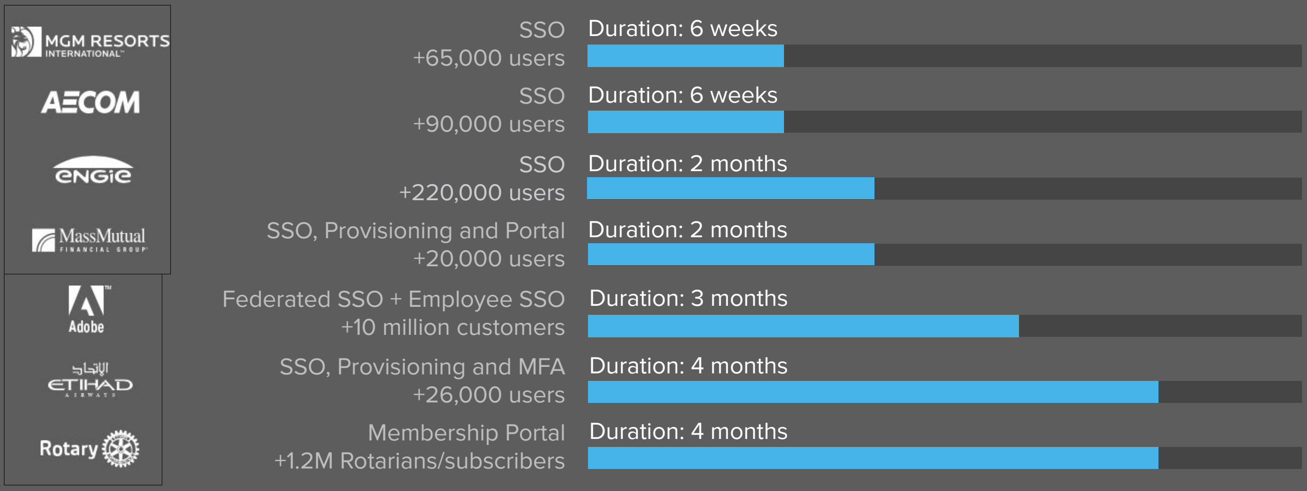
Okta and CA SiteMinder integration: Conceptual Architecture

Benefits

After this step:

- All new apps and SaaS services are integrated to Okta, deprecating CA SiteMinder.
- Okta provides universal SSO and MFA for all applications, including CA SiteMinder. CA SiteMinder trusts the Okta authentication to grant access to its apps.

Because Okta is provided as a cloud service, the initial configuration is executed at a fast pace:



Examples of time to deploy and rollout Okta SSO, MFA, and Provisioning across different companies and industries

1. Define Strategy

In this task, you list the requirements for your identity solution. Examples of requirements include:

- What users – i.e. employees, partners, customers, contractors – will use Okta?
- What system – i.e. AD, LDAP – will store the user data?
- What applications will be integrated with Okta Single Sign-On?
- What are the security policies?

Since Okta provides out-of-the-box features beyond SiteMinder, your requirements can also extend to:

- Do you want to use HR as the source of truth for accounts?
- Do you want to provision accounts to downstream apps?
- Do you want to retire LDAP and use Okta as the user store?
- What MFA factors will be used? Do you want to use Push Notification or FIDO2?
- Do you want to turn on out-of-the-box MFA enrollment and self-service MFA management?

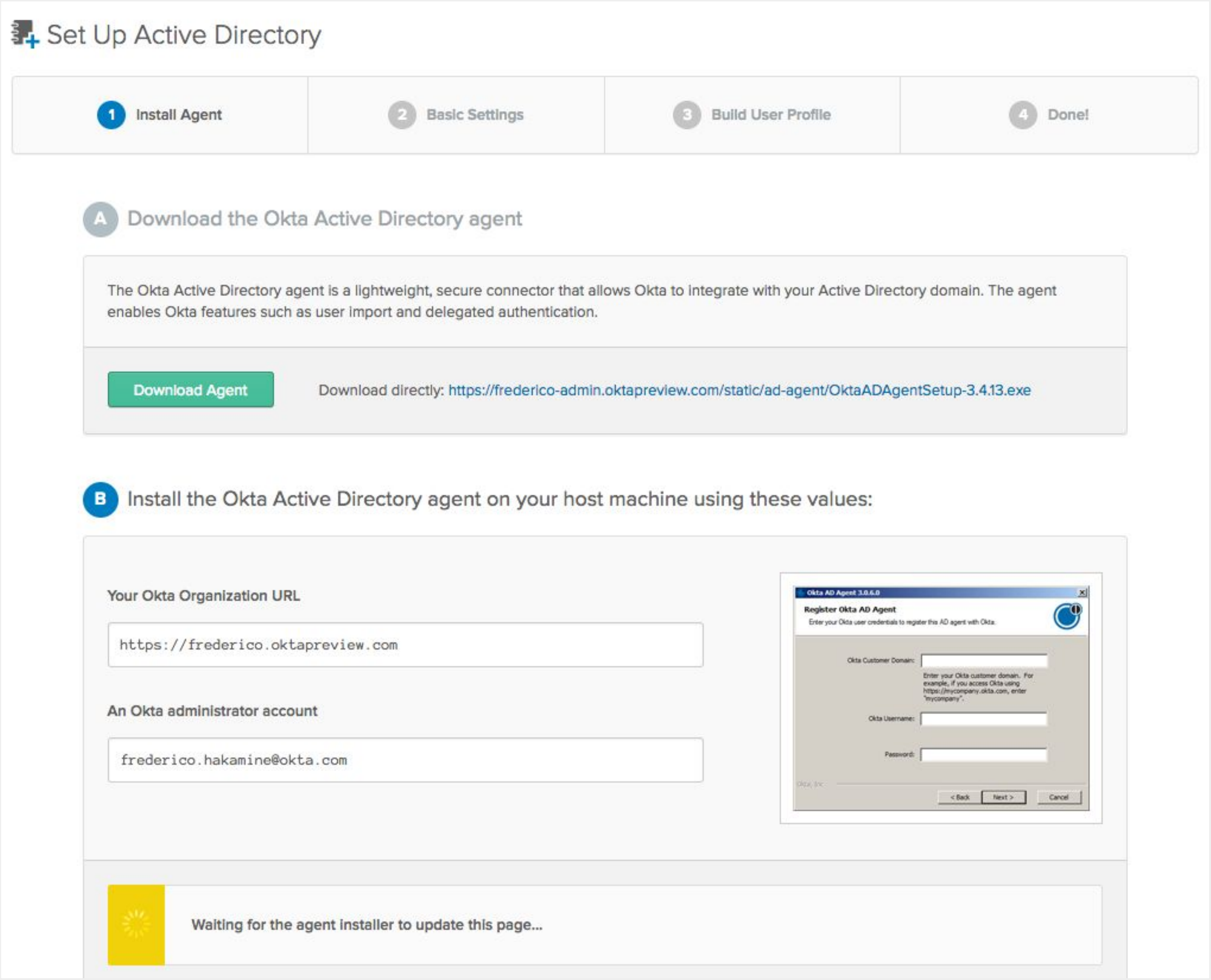
The requirements will define which services and settings will be turned on in your Okta tenant. Since Okta is a subscription-based platform, you can change requirements as you go. Turning on additional features is easy and does not require additional setup, saving you money while reducing complexity.

2. Configure the Okta service

In this task, you configure initial policies and settings in Okta. Deployments with a pre-existing CA SiteMinder install usually include:

- Install an LDAP or AD Agent to sync users and groups with Okta.
- Configure an initial authentication and password reset policy.
- Configure an initial policy for MFA enrollment and enforcement.

The initial Okta configuration is facilitated with default configurations and integration wizards:



Set Up Active Directory

A Download the Okta Active Directory agent

The Okta Active Directory agent is a lightweight, secure connector that allows Okta to integrate with your Active Directory domain. The agent enables Okta features such as user import and delegated authentication.

Download directly: https://frederico-admin.oktapreview.com/static/ad-agent/OktaADAgentSetup-3.4.13.exe

B Install the Okta Active Directory agent on your host machine using these values:

Your Okta Organization URL

https://frederico.oktapreview.com

An Okta administrator account

frederico.hakamine@okta.com

Okta AD Agent 3.0.6.0

Register Okta AD Agent

Enter your Okta user credentials to register this AD agent with Okta.

Okta Customer Domain:

Enter your Okta customer domain. For example, if you access Okta using https://mycompany.okta.com, enter "mycompany".

Okta Username:

Password:

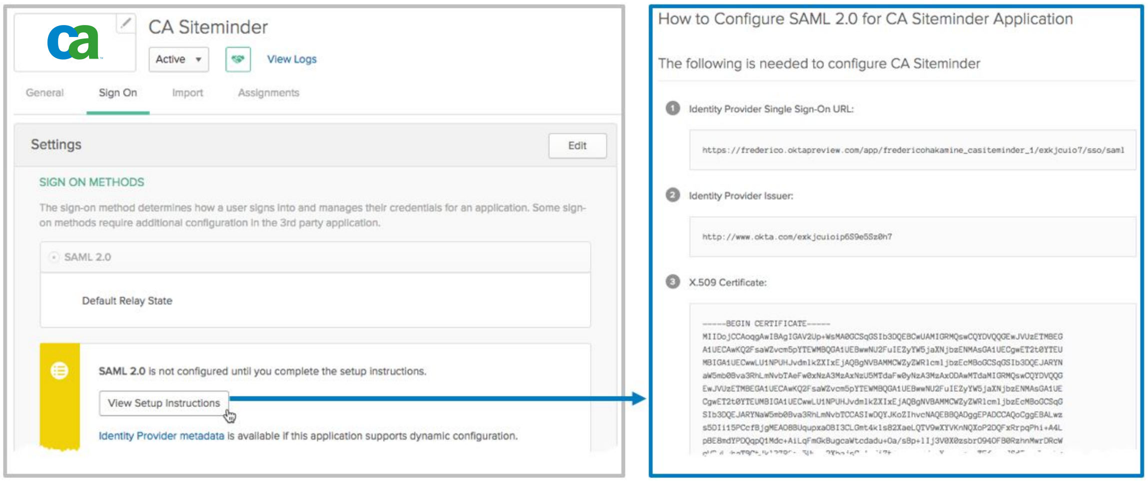
Waiting for the agent installer to update this page...

Active Directory Integration in four steps

By the end of this task, you should have users ready to access Okta with the same credentials they use to access on-premise systems, plus Adaptive Multi-Factor Authentication.

3. Integrate Okta and CA SiteMinder (optional)

While the migration from SiteMinder to Okta can go quickly, you can opt-in to use the Okta service to access SiteMinder app before the migration. With this integration, your users can take advantage of Okta SSO, Multi-Factor Authentication, Self-Service UIs, and Dashboard to access apps from both Okta and SiteMinder, anticipating the security and user experience benefits even before the migration is completed. This integration, uses Okta as a SAML Identity Provider for SiteMinder:



The sign-on method determines how a user signs into and manages their credentials for an application. Some sign-on methods require additional configuration in the 3rd party application.

SAML 2.0 is not configured until you complete the setup instructions.

Identity Provider metadata is available if this application supports dynamic configuration.

How to Configure SAML 2.0 for CA Siteminder Application

The following is needed to configure CA Siteminder

1. Identity Provider Single Sign-On URL:

```
https://frederico.oktapreview.com/app/fredericohakamine_casiteminder_1/exkjcuio7/sso/saml
```

2. Identity Provider Issuer:

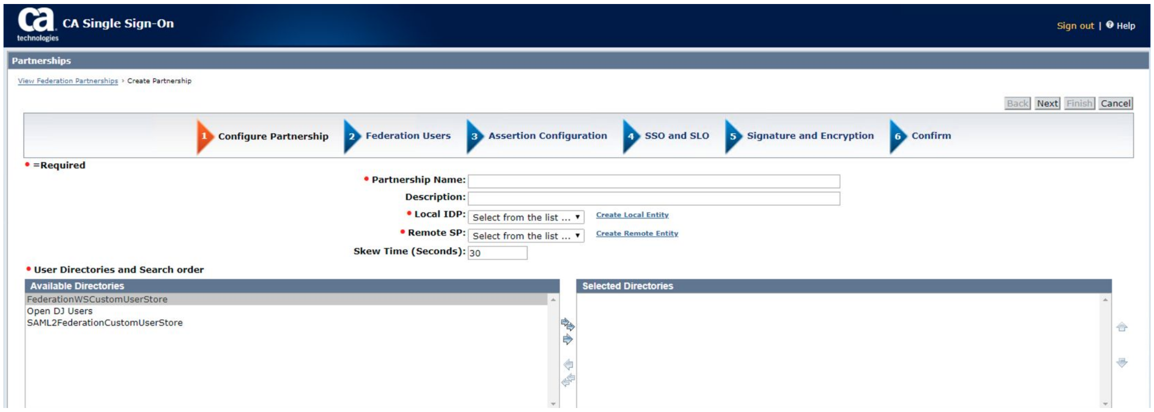
```
http://www.okta.com/exkjcuioip6S9e5Sz0h7
```

3. X.509 Certificate:

```
-----BEGIN CERTIFICATE-----
MIIDojCCAoqgAwIBAgIGAV2Up+WsMA0GCSqGSIb3DQEBCwUAMIGRMQswCQYDVQQGEwJVUzETMBEG
A1UECAwKQ2FsaWZvcm5pYTEWMBQGA1UEBwwNU2FuIEZyYW5jaXNjbzENMAsGA1UECgwET2t0YTEU
MBIGA1UECwwLU1NPUHJvdmlkZXIxEjAQBgNVBAMMCWZyZWRlcmljbzEcMBoGCSqGSIb3DQEJARYN
aW5mb0Bva3RhLmNvbTAeFw0xNzA3MzAxNzU5MTdaFw0yNzA3MzAxODAwMTdaMIGRMQswCQYDVQQG
EwJVUzETMBEGA1UECAwKQ2FsaWZvcm5pYTEWMBQGA1UEBwwNU2FuIEZyYW5jaXNjbzENMAsGA1UE
CgwET2t0YTEUMBIGA1UECwwLU1NPUHJvdmlkZXIxEjAQBgNVBAMMCWZyZWRlcmljbzEcMBoGCSqG
SIb3DQEJARYNaW5mb0Bva3RhLmNvbTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBALwz
s5DIi15PCcfBjgMEAOBBUqupxaOBI3CLGmt4k1s82XaeLQTV9wXYVKnNQXoP2DQFxRrpqPhi+A4L
pBE8mdYPDQqpQ1Mdc+AiLqFmGkBugcaWtcdadu+Oa/sBp+lIj3V0X0zsbrO94OFB0RzhnMwrDRcW
```

Okta Application Integration Wizard – Integrating with CA SiteMinder as Service Provider

On the CA SiteMinder side, configure Okta as a SAML or OpenID Connect Identity Provider. This configuration is simplified by importing the `metadata.xml` file provided automatically by Okta:



• =Required

• Partnership Name:

Description:

• Local IDP: Select from the list ...

• Remote SP: Select from the list ...

Skew Time (Seconds): 30

• User Directories and Search order

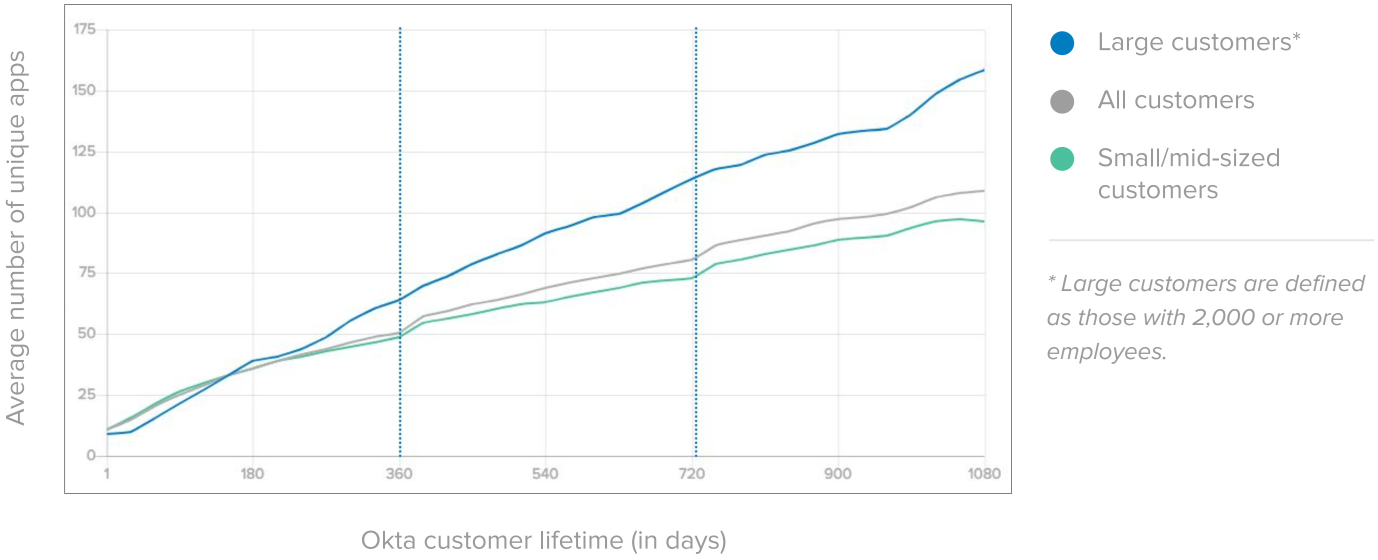
Available Directories	Selected Directories
FederationWSCustomUserStore Open DJ Users SAML2FederationCustomUserStore	

On CA SiteMinder, configure Okta as the Identity Provider

The **CA SiteMinder** official documentation has more details on how to integrate CA SiteMinder with an external Identity Provider like Okta: https://docops.ca.com/ca-single-sign-on/12-8/en/configuring/legacy-federation/configure-a-saml-2-0-identity-provider/

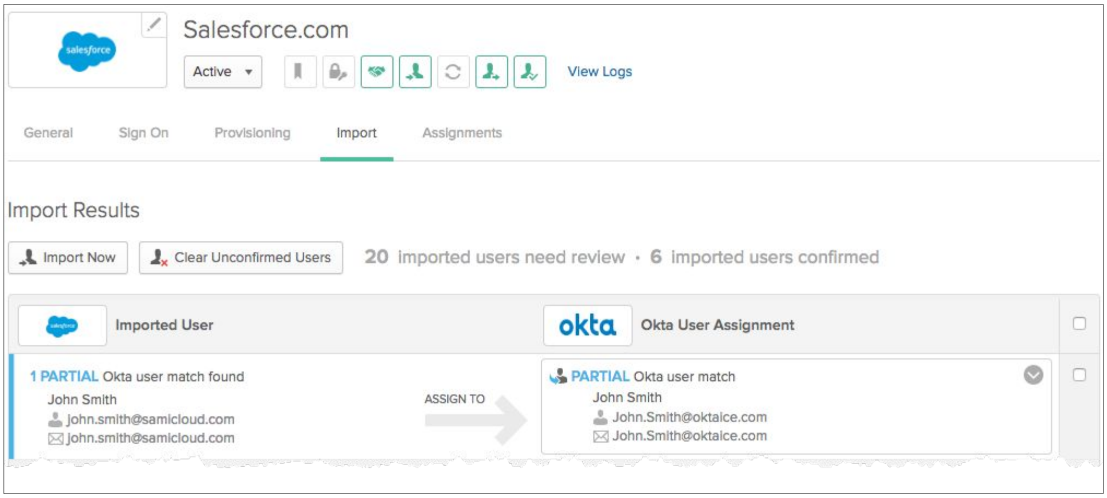
4. Integrate new applications with Okta

In this task, you integrate Okta with existing cloud applications using the Okta Integration Network (OIN). You can immediately take advantage of the Okta platform to protect any new SaaS or on-prem applications. This allows you to leverage the out-of-the-box integrations and open-standard SDKs making integrations easier, faster, and more reliable than with SiteMinder:



** Large customers are defined as those with 2,000 or more employees.*

Okta provides prescriptive integration guides for 6,000+ applications that support both net new and existing subscriptions. On existing apps, Okta is capable of importing and matching user records:



20 imported users need review • 6 imported users confirmed

1 PARTIAL Okta user match found
John Smith
john.smith@samicloud.com
john.smith@samicloud.com

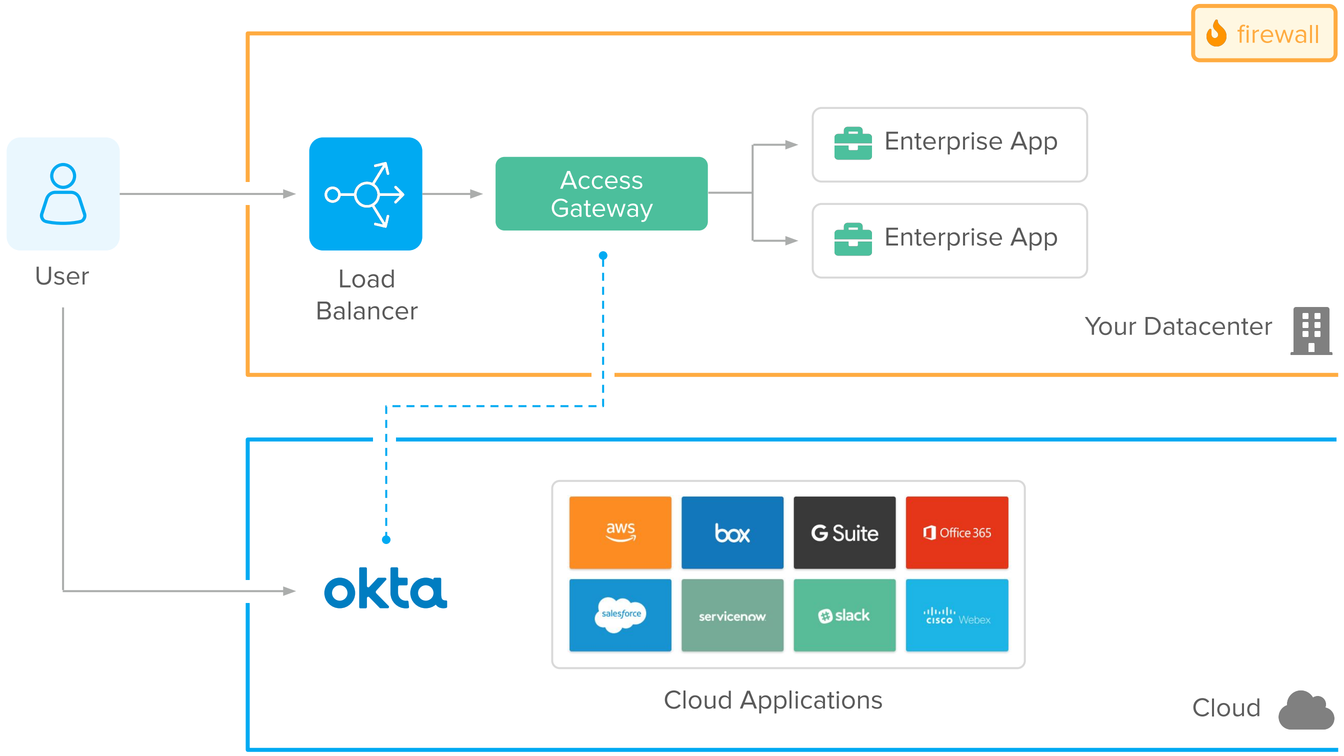
PARTIAL Okta user match
John Smith
John.Smith@oktaice.com
John.Smith@oktaice.com

Okta Integration Network: Onboarding systems with pre-existing users

Step 2: Migrate CA SiteMinder

In this step, you migrate your identity stack from CA SiteMinder to Okta in 3 tasks:

1. Inventory – list and classify – your CA SiteMinder applications
2. Migrate CA SiteMinder applications to Okta
3. Uninstall CA SiteMinder



Conceptual Architecture after migrating CA SiteMinder applications to Okta

Okta provides tailored migration options for CA SiteMinder deployments. To learn more about Okta migration options and to get a migration tailored to your company, reach out to our team.

Benefits

After this step, Okta acts as the single identity provider for all apps, improving your user experience, Return of Investment, and retiring the CA SiteMinder deployment.

1. Inventory your CA SiteMinder applications

In this task, you take an inventory of apps currently protected by your CA SiteMinder solution and then classify the applications based on the integration used. The integrations typically used on enterprise CA SiteMinder deployments ranked by popularity are:

Architecture	Integration
1. Gateway-based authentication	1. Headers/Responses
2. Agent-based authentication	2. Java Application Servers
	3. SAML to COTS and SaaS apps
	4. ERP Based (eBusiness Suite, Peoplesoft) apps

2. Migrate CA SiteMinder Applications to Okta

In this step, you migrate applications from CA SiteMinder to Okta. The migration tasks include:

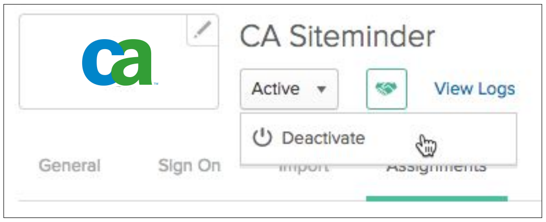
- Select in which order applications are migrated from CA SiteMinder to Okta.
- Start by migrating SAML apps.
- Install Okta Access Gateway and Gradually migrate other apps – i.e. Agents/Gateways apps – copying the SiteMinder Responses and Authorization Policies.
- Uninstall unused CA SiteMinder agents.

Ideally, your migration should start from low-risk apps with the same type of integration. Like most customers, your migration confidence grows up to a point where you migrate apps of the same kind in bulk. The bulk migration expedites the process time while reducing costs.

3. Uninstall the legacy CA SiteMinder service

After all apps are migrated from CA SiteMinder to Okta, your CA environment does not receive any requests. Now is time to turn off the SiteMinder service. Turning off the SiteMinder service includes:

- Deactivate the integration with Okta (configured on 3: Integrate Okta and your existing CA SiteMinder).
- Monitor access to CA SiteMinder, Web Agents, Gateways, and HTTP Servers for few days to confirm the service is not being used in rogue apps – apps not identified in the classification process.



- Take final backups and uninstall CA SiteMinder.

To learn more about the initiatives and projects you can accomplish with Okta, reach out to our team.

Appendix A:

Modernization/Migration FAQ

This chapter lists common questions around CA SiteMinder modernization and migration to Okta.

What's the user experience during migration to Okta?

The migration from CA SiteMinder to Okta does not impact the end-user experience significantly. During the Okta configuration, users are imported to Okta automatically – via the Okta LDAP and AD agents – and use the same credentials from CA SiteMinder – and its user directory – to access Okta. Depending on your deployment scenario and stage, users will see Okta as the new login page. The users' reaction to the new login page tends to be positive, mainly due to the page speed on the browser and the UI responsiveness on mobile access. Okta provides an End User Adoption toolkit that you can use for a successful launch to end-users.

Does the Access Gateway support Realms (URL-based Authorization)?

Access Gateway supports the following authorization scenarios:

Authorization Complexity	**Example**
Public Assets (No authz)	intranet.org.com/public
App-level Authz	intranet.org.com/app1 and intranet.org.com/app2
App basic URI Authz	intranet.org.com/app1/admin and intranet.org.com/app1/home
App Deep Authz	intranet.org.com/app1/admin/x/a and intranet.org.com/app1/admin/a/t
Dynamic URI Authz	intranet.org.com/app1/{userid}/status

Can the Access Gateway replace proprietary SDK integrations such as the CA SiteMinder C and Java SDKs?

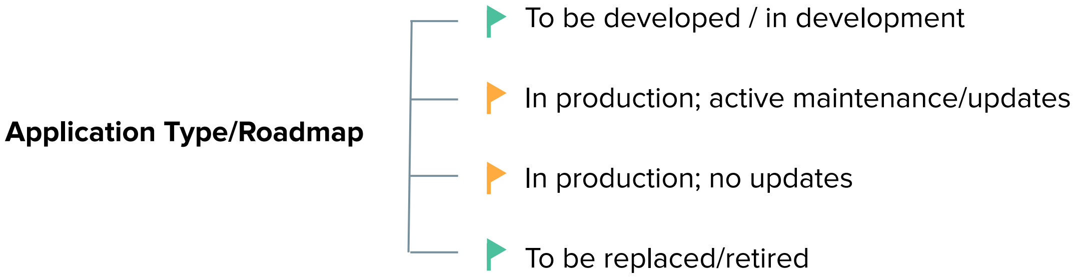
No. Okta integrates with applications via Access Gateway – HTTP request level – and via open standards – OpenID Connect, OAuth, LDAP. The use of proprietary CA SiteMinder SDKs – e.g. CA SiteMinder C and Java SDK – creates vendor lock-in with CA and the proprietary SiteMinder protocol. Migrating these apps require changes in their source code regardless of which new identity solution you adopt. The recommended action for apps with proprietary CA SiteMinder SDKs is to replace the SDK with open-standard integrations. The use of open standards allows you to adopt solutions while avoiding vendor lock-in.

Does Okta support integrations to ERP and COTS systems?

Access Gateway and Okta supports multiple ERPs/COTS including SAP NetWeaver, Oracle eBusiness Suite, Peoplesoft, JD Edwards, Agile PLM, Outlook Web Access, and Sharepoint On-Prem.

How to approach and application updates and future developments in environments with Okta and CA SiteMinder?

Use this image as a guide:


Application Type/Roadmap
To be developed / in development
In production; active maintenance/updates
In production; no updates
To be replaced/retired

- **Applications "To be developed / in development" should incorporate modern authentication using Okta**. This future proof the application and improve its support for API authorization and multi-cloud environments.
- **For applications "to be replaced/retired" before you uninstall CA SiteMinder**, consider waiting for the application retirement. For low-risk applications to be retired after the CA SiteMinder uninstall, you can implement Okta's Secure Web Authentication (Okta's Form Fill technology).
- **For applications that are in production on the foreseeable future**, check its maintenance/update cadence. Applications with proper maintenance and constant updates usually offer better support for federated authentication.

What technical recommendations are applicable for when implementing the Access Gateway?

When implementing the Access Gateway, consider the following best practices:

- To avoid network conflicts, consider placing Access Gateway close to your current Agents and HTTP servers.
- To meet resiliency requirements, implement the same high-availability as your Gateway/HTTP Server and performance test your configuration.
- To avoid URL rewriting or re-bookmarks from users, try to keep the same application domains.
- Use your Load Balancer to gradually migrate traffic from CA to the Access Gateway
    - Use the Load Balancer rules to direct/balance traffic between CA SiteMinder and Access Gateway.
    - Balancing strategies include network origin or round-robin with % distribution.
    - To adopt gradual migration via Load Balancer rules, make sure you have session stickiness/persistence. The persistence makes sure users that established a session in CA is not routed to Access Gateway and vice-versa.
    - Document a sanity check script for testing each path (some Load Balancers allow you to determine your path through request headers) to help you confidently ramp-up the migration.
    - You can also use the Load Balancer policies to fallback traffic to CA in case you need to troubleshoot the Access Gateway deployment.

What technical recommendations are applicable for when uninstalling CA SiteMinder?

Before uninstalling CA SiteMinder, consider following the best practices:

- Consider monitoring CA SiteMinder for a period before uninstalling (so you can detect integration gaps).
- Disable CA agents.
- Take a backup of your entire environment before uninstalling the system.

To learn more about the initiatives and projects you can accomplish with Okta, reach out to our team.

Appendix B:

Example of App migration from CA SiteMinder to Okta

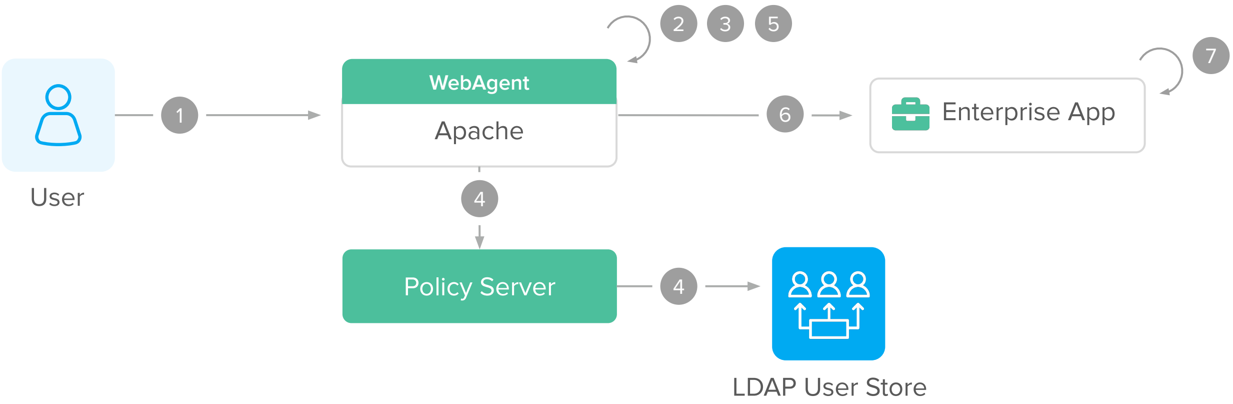
This chapter provides an example of how an application is migrated from CA SiteMinder to Okta.

Example: Using SiteMinder Responses to provide HTTP headers to applications

The most common architecture to integrate CA SiteMinder with applications is by using SiteMinder Responses to provide HTTP headers containing the users identity to the applications. The application is first secured either with a SiteMinder webagent - a plugin directly on the web server, or the SiteMinder Access Gateway (a reverse proxy with a web agent built-in). Regardless if used as a web server plugin or on a reverse proxy, the agent is responsible for determining if the requested resource is protected by SiteMinder, prompting the users for authentication, performing session validation, and performing url level authorization. Once URL authorization is complete, SiteMinder will add HTTP headers to the request based on the SiteMinder responses configured in the Admin UI. These responses most commonly contain attributes from the SiteMinder user store, but may also invoke custom C or Java code (active responses). Care must be taken when developing these active responses since a coding or memory leak in the active response code and cause the SiteMinder policy server to crash.

Before migrating to Okta

Before Okta, the application used CA SiteMinder with CA's HTTP Agent on Apache as reverse-proxy for authentication:



Conceptual Architecture: Response/Header-based Authentication with traditional CA SiteMinder solutions (SP initiated flow)

IdP-Initiated Flow

SiteMinder does not have a concept of a dashboard – no IdP initiated flow. As a result, users would access these applications directly, using exclusively the SP-Initiated flow.

SP-Initiated Flow

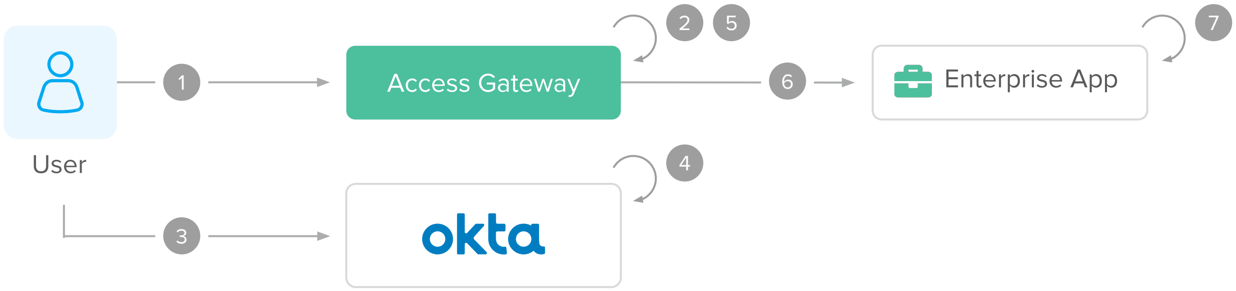
The SP initiated flow is triggered when users try to access the enterprise applications served by the HTTP server integrated to CA SiteMinder via HTTP Agent:

1. User tries to access an enterprise application URL via an Agent or Reverse Proxy.
2. HTTP Agent intercepts the requests and detects no SSO session.

   The user is redirected to the login page for authentication.
3. User submits his/her credentials (usually username and password).
4. The CA SiteMinder agent and Policy Server authenticates user credentials against an LDAP or AD server.
5. After login, SiteMinder establishes an SSO session and authorize the user access.
6. After the realm/URL authorization, HTTP Agent adds HTTP header variables – containing information about the logged user – to the request and allows the request to reach the enterprise application.
7. Enterprise app receives the request and reads the HTTP header variables to establish an app session.
8. Enterprise app processes the request and returns a page to the end-user.
9. The SSO session is reused on subsequent requests for authentication and realm/authorization.

After migrating to Okta

In the migration to Okta, the reverse proxy serving enterprise apps changes from CA to Access Gateway. The gateway acts the same way as a CA HTTP Agent or Gateway and can deliver the responses/header variables expected by the enterprise app. Due to this, the enterprise application can operate with Okta without changes in source code.



Conceptual Architecture:
Response/Header-based Authentication with Okta (SP initiated flow)

IdP-Initiated Flow

The IdP initiated flow is triggered when the user, either from your intranet portal or from the Okta Dashboard, clicks on a shortcut to the enterprise application.

1. User clicks access an enterprise application from the Okta dashboard or your intranet portal.
2. Okta redirects user to the enterprise application URL (protected by the Access Gateway).
3. Access Gateway receives the request and performs an initial SAML federation with Okta. This step is transparent to users already logged into Okta.
4. Access Gateway establishes a session cookie and authorizes the request URL.
5. After realm/authorization, the Access Gateway adds responses (HTTP header variables) – containing information about the logged user –to the request and allows the request to reach the enterprise application. (The gateway uses the same integration as CA SiteMinder to avoid code updates in the enterprise app).
6. Enterprise app receives the request and reads the responses (HTTP header variables) to establish an app session.
7. Enterprise app processes the request and returns a page to the end-user.
8. The Access Gateway session cookie is reused for subsequent requests. The Access Gateway validates each request for SSO and realm/authorization.

SP-Initiated Flow

The SP initiated flow is triggered when users try to access the enterprise applications served by the Reverse Proxy integrated to the CA SiteMinder/SSO Server:

1. User tries to access an enterprise application URL via Access Gateway.
2. Access Gateway intercepts the requests and detects no session cookie.

   Access Gateway performs a SAML assertion with Okta.
3. If the user is not logged into Okta, a login page is displayed

   User submits his/her credentials (and optionally MFA) to Okta.
4. Okta authenticates the user credentials internally or via delegated authentication to LDAP/AD servers.
5. After the successful login, a SAML assertion is returned to the Access Gateway.

   Access Gateway establishes a session cookie and authorizes the request URL.
6. After realm/authorization, the Access Gateway adds responses (HTTP header variables) – containing information about the logged user –to the request and allows the request to reach the enterprise application. (The gateway uses the same integration as CA SiteMinder to avoid code updates in the enterprise app).
7. Enterprise app receives the request and reads the responses on the HTTP header to establish an app session.
8. Enterprise app processes the request and returns a page to the end-user.
9. The Access Gateway session cookie is reused for subsequent requests. The Access Gateway validates each request for SSO and realm/authorization.

To learn more about the initiatives and projects you can accomplish with Okta, reach out to our team.