STRATA

# The Ultimate Identity Modernization and Migration Checklist

Migration Planning > User Account Migration > App Migration > Coexistence > Retire Legacy

# Identity Modernization and Migration

Migrating apps and users from legacy identity systems, like SiteMinder, can be a daunting task. Complex analysis and planning, and the need to rewrite apps and perform extensive testing — all while running your business — are just the tip of the iceberg for migration challenges.

But, it doesn't have to be that way. Identity orchestration is transforming the migration process for good.

Strata's Maverics Migration software is a new approach for migration that all but eliminates traditional migration challenges. Maverics automates and simplifies the migration and modernization process with built-in best practices.

Successful identity migrations follow this predictable, five step pattern:

Migration Planning > User Account Migration > App Migration > Coexistence > Retire Legacy.

## Migration Planning and Discovery

Let's start with Migration Planning. In this step, you need to analyze your current identity infrastructure to understand and map out which apps you need to migrate.

Strata provides a specialized tool, Maverics Identity Discovery, that extracts and analyzes SiteMinder agents, configurations, and policies. Then, it logs and categorizes the level of effort it will take to migrate users and apps based on a complexity scale accounting for app utilization, access control policies, and any required customizations.

### Checklist for Maverics Identity Discovery for SiteMinder

#### Apps

- Which apps and resources does SiteMinder protect?
- How is SiteMinder protecting your apps?
- What session patterns are used, cookies, HTTP headers, Kerberos, SAML?
- Which apps are accessed most frequently?
- Which apps have the highest volume of users?

 

## Users

- Which directories are used for user authentication and authorization?
- What attributes do the response headers use?
- From where are those attributes sourced?
- Are attributes retrieved from multiple directories?
- Are user sessions being managed across realms and cookie domains?

## Policies

- Which of your apps and resources does SiteMinder protect?
- What SiteMinder Domains, realms, policies, rules, and responses are configured?
- What authentication schemes are being used: Basic, Forms, Client Certificate, Windows Authentication, SAML, or others?
- Does your SiteMinder deployment make use of global entities such as policies, responses, and rules?
- Which access policies are actively evaluated and enforced? With what frequency?
- Where are customizations utilized by SiteMinder? Expressions, authentication schemes, attributes in responses, login pages, or others?

## Infrastructure

- What SiteMinder agents and agent groups are deployed in the environment?
- Is SiteMinder using a proxy or have agents deployed behind proxy servers?
- Which SiteMinder Policy Servers are deployed in the environment?
- Is SiteMinder using Federation Partnerships with third-party IDPs?

 

## Complexity

- Do apps use fairly simple rules?
- Do apps require a large number of rules? Do rules make use of regex?
- Do apps require a large number of headers? Make use of LDAP groups?
- Do apps use custom logic or extensions in SiteMinder?

With this data in hand, you can plan your migration. We've seen organizations have great success by starting with simple, non-mission critical apps to prove the process first. For example, choosing apps with lower utilization, using simpler rules and policies, and getting relatively fewer or less complex attributes in headers. Understanding relative complexity will help you group your apps and users into cohorts that you can incrementally migrate.

# User Migration

Now that you have done the discovery and planning phase, it's time to choose your target identity system and migrate your users.

Start by creating an inventory of your users, attributes, groups and group memberships, and roles. Use your Discovery data to connect to your user repositories and plan how Maverics will map from your old schema(s) to your new identity system's schema.

Determine whether your target system supports standard protocols, such as SCIM, or whether proprietary user APIs must be used.

Identify from where additional attributes may be sourced, for example, databases or services exposed via APIs. Map out your existing login process and user experience.

## Checklist for Maverics user migration

### Key user migration considerations

- Migrate users in real time, with small cohorts following agile practices, such as blue-green testing.
- Migrate users transparently without changes to the login or application access experience.
- Avoid a 'Big Bang', hard cutover with complicated user notifications for password resets and procedures that spike help desk calls.
- Track which user accounts have been migrated.

### Identity provider sources

- Configure connectors to source identity systems.
- Collect user IDs and credentials during SiteMinder login.
- Read user attributes from LDAP/Active Directory.
- Read user attributes from SQL.
- Call REST services and APIs for user attributes.

 

## Attribute mapping

- Configure connectors to each target identity system.
- Configure Maverics with attribute mappings.
- Verify that users are created with all mandatory and any supplemental attributes.
- Verify that users can authenticate using the target identity system.

## Identity provider targets

- Create users in Azure AD B2B.
- Create users in Azure AD B2C.
- Create users in Okta Universal Directory.
- Create users in Ping Directory.

## Automatic MFA enrollment (optional)

- Enroll migrated users into OneKosmos automatically.
- Enroll migrated users into Yubikey automatically.
- Enroll migrated users into other MFA providers.

## Account fortification (optional)

- Screen accounts against a threat service to identify compromised accounts and automatically set them for remediation (password reset).
- Identify dormant accounts that haven't been active in a certain amount of time and automatically set them for remediation (password reset).

## Complement Active Directory Sync tools from Microsoft and Okta

- Add incremental migration capability to Active Directory Sync.
- Microsoft's Active Directory Sync only works with Active Directory.
- Maverics enables additional user repositories like LDAP, database, and APIs to complement Active Directory data.

## Seamless user experience

- Identity provider changes are transparent to users.
- Maverics can present any login experience to the user.

# App Migration

With your users where they need to be, it's time to focus on migrating your apps from your legacy systems to cloud identity. Start with the inventory of apps from your Discovery research.

## No app rewrites

The critical element with App Identity migration is to avoid rewriting apps. This is because often you don't have access to the source code; it's a commercial off-the-shelf app or you don't have technical bandwidth to manually refactor the login code of your apps. Use identity orchestration as an abstraction layer to insulate you from having to touch apps to change identity.

### Checklist for Maverics app identity migration

#### Required identity session types

- HTTP headers
- Cookies
- Kerberos
- OIDC
- JWT tokens
- SAML

# Transforming tokens

Apps support varying ways to manage user sessions including cookies, HTTP headers, OIDC tokens, SAML and Kerberos. One key to interoperability is the ability to transform one type of token into another.

### Token transformation

- Use claims from an OIDC flow to populate HTTP headers with required attributes.
- Use SAML attributes to populate HTTP headers with required attributes.
- Combine attributes from OIDC claims or SAML with attributes from any other identity source to enrich user profiles.
- Turn any token into any other token format.

# Migration factory

If you have hundreds — or thousands — of apps that need to migrate you need to be thinking about scale. To scale, you need to automate and templatize everything. Using Maverics as an abstraction layer and by operationalizing the migration, you can migrate apps in hours rather than months as it would without an abstraction layer.

### Required app migration tools

- Tools to plan migration cohorts.
- Migration checklist management.
- App Owner support tools/emails to streamline app readiness.

 

# Hybrid coexistence

During an incremental migration process, your legacy and cloud identity systems will have to operate in a coexistence mode for some period of time. During this period, Maverics will determine which identity provider to use for a given user or app.

Many organizations must keep some apps and data on-premises, for good reasons. Securely extending access to on-premises apps using cloud identity while taking advantage of Zero Trust architecture is essential to support modern distributed workforces.

## Checklist for Maverics multi-cloud identity coexistence

### Secure hybrid access requirements

- Keep sensitive apps and data on-premises while using Azure AD as your primary authentication and identity provider.
- Keep sensitive apps and data on-premises while using Okta, or other cloud identity systems, as a primary authentication and identity provider.
- Extend MFA to on-premises apps without rewriting apps or requiring provider-specific SDKs.
- Enrich user attributes at runtime from multiple sources.

# Coexistence use cases

During the migration transition you need to run both old and new systems in parallel. Here are the most important coexistence use cases:

- Run SiteMinder, OAM, or other legacy identity in parallel with your cloud identity system.
- Incrementally move apps from SiteMinder to a cloud identity system.
- Create and manage user sessions for apps protected by both SiteMinder and your cloud identity system.

 

# Retire legacy IAM and enjoy cloud benefits

Now that your apps and users are able to use modern identity, it's time to shut down the legacy environment and move on to enjoying the benefits of multi-cloud powered by identity orchestration.

## Checklist for Maverics multi-cloud identity decommissioning legacy and multi-Cloud identity

### Savings from retiring legacy identity

- Save time and money by not rewriting apps.
- Recover budget from spend on legacy maintenance and support contracts.
- Recover budget from retiring related infrastructure and related expenses including authorization servers, policy servers, retired databases, and directories.
- Focus your team on innovation and away from the toil of managing ancient infrastructures without a future.

### Benefits from multi-cloud identity

- Take advantage of the best technologies available to power your business.
- Have greater flexibility and more choices for pricing models and customer experience.
- Open up your business to new opportunities available through identity orchestration.
- Get out of custom coding integration between apps and identity allowing you to do more progressive projects and initiatives that move you forward.

# Conclusion

Successful identity migrations follow this predictable pattern: Migration Planning > User Account Migration > App Migration > Coexistence > Retiring Legacy. Using Maverics as an abstraction layer empowers you to avoid rewriting apps when moving to the Cloud and when switching identity systems.

Using identity orchestration software, like Strata's Maverics Platform, to migrate your users and apps to the Cloud enables you to focus your attention on higher-level issues and gets you out of the custom identity integration code business once and for all.

# About Strata

Strata is pioneering the concept of identity orchestration for distributed, multi-cloud identity. The Maverics Platform enables enterprises to seamlessly unify on-premises and cloud-based authentication and access systems for consistent identity management in multi-cloud environments. Strata's distributed approach to identity enables organizations to break decades-old vendor lock-in that has prevented a broader transition of enterprise workloads to public cloud infrastructures. The company's founders co-authored the SAML open standard for identity interoperability, created the first cloud identity services, delivered the first open-source identity products, and are now building the first distributed identity platform.

Visit us at Strata.io

 