User Provisioning in the Enterprise

By Ian Glazer, Lori Robinson, and Mat Hamlin

Table of Contents

Abstract

User provisioning is the means by which user accounts are created and maintained in a system (e.g., database, SaaS app, operating system, etc.). When we say that a user-provisioning system maintains a user account, we mean everything from changes to attributes in the user account, changes of entitlements or privileges associated with the user account, locking and unlocking the user account, and even deletion of the user account. User provisioning is primarily an admin-time affair: a user account is created (or changed) based on an administrative action as opposed to a user's action at the time of resource use. This article explores the uses and components of a user-provisioning system and focuses mainly on situations where user accounts are maintained in central repositories, typically enterprise and workforce settings.

Introduction

Creating and managing user accounts is the bedrock of any IAM system. The process is generally referred to as user provisioning and is used to establish the entitlements a user is given to access restricted resources (applications, documents or databases) maintained by the organization. User provisioning processes not only create user accounts and assign entitlements but also maintains those user account entitlement through the detection of meaningful lifecycle events such as changes to job responsibility and the application of policies to ensure. User provisioning is often used to ensure the right people have access to the right systems in a timely fashion and with entitlement appropriate for their responsibilities.

Terminology

- **Authoritative source(s):** The system of record (SOR) for identity data; an organization may have more than one authoritative source of data in their environment.
- **Entitlement catalog:** A database of entitlements and their related metadata. The catalog includes an index of entitlement data pulled from business systems, applications, and platforms, as well as technical and business descriptions of the entitlements or their use
- **Identity lifecycle management:** A process that detects changes in authoritative systems of record and updates identity records based on policies.
- **Identity repository:** The identity repository is a directory or a database that can be referenced by external systems and services (such as authentication or authorization services).
- **Reconciliation:** The process of identifying and processing changes to users and user access made directly on target systems.

- **User provisioning**: the means by which user accounts are created, maintained, and deactivated/deleted in a system according to defined policies.

What is User Provisioning?

User provisioning is setting up the entitlements for users to the resources to which they need access. User-provisioning technologies are deployed across multiple industries, including healthcare, education, financial services, government, retail, manufacturing, technology, etc.

Functionality supported by user-provisioning technologies include:

Identity lifecycle management: An identity and its associated attributes are the basis for authentication and authorization decisions made in an environment. It is, therefore, essential that the identity record is maintained. Provisioning systems detect changes in authoritative systems of record (such as a Human Resources database/repository) and update the identity record accordingly.

User account provisioning: As the name suggests, a user-provisioning system's primary function is user account provisioning (and de-provisioning). User-provisioning technologies automate the creation, maintenance, and deactivation/deletion of user accounts on target systems according to defined policies.

Self-service and delegated administration: User-provisioning systems provide interfaces that allow users to request access to systems, manage passwords and update their data. Delegated administrators can perform similar tasks such as onboarding and off-boarding users, password changes, profile updates, and entitlement assignments on behalf of others.

Workflow: Provisioning systems employ workflow tools that allow for the automation of provisioning processes and approval workflows. Using automated approval workflows, business stakeholders can validate and approve proposed changes before they are applied to target systems. While many decisions to grant access are automated through policy, others may require human intervention.

Audit and Reporting: Provisioning systems log all identity lifecycle management, access policies, and user provisioning transactions and provide reporting mechanisms to extract the logged data.

A note about governance: User-provisioning systems are often packaged with identity governance capabilities such as access review and certification, risk analysis, and identity analytics. The combined user provisioning and identity governance solutions may be

referred to as Identity Governance and Administration (IGA). This document focuses exclusively on user-provisioning functionality and does not include identity governance information. Similarly, password management, which may be packaged with provisioning solutions, is not covered in this document.

Business Drivers for Automated User Provisioning

Three primary business drivers justify the deployment of automated user-provisioning systems:

- **Operational efficiency**: The amount of administrative overhead associated with the manual creation and maintenance of user accounts is significant for medium- to large-size organizations. Without an automated process, it may be weeks before a user has access to the resources they need to perform their job duties or other tasks. User-provisioning systems automate the user account management process, reducing administrative overhead and improving time to productivity, resulting in operational efficiency.
- **Security:** Manual provisioning of user accounts may lead to security gaps such as overprivileged user accounts or orphaned accounts (active accounts assigned to inactive employees). Automated user account provisioning systems improve security by ensuring that user accounts and entitlements are provisioned according to policy and deprovisioned in a timely manner.
- **Compliance**: Various laws and regulations require organizations to demonstrate control over access to critical systems, resources, and data. User-provisioning systems enforce policy-based access controls and allow organizations to demonstrate the efficacy of these controls with reporting and attestation capabilities.

User Provisioning Logical Architecture

User-provisioning systems employ policies, workflows, and connectors to synchronize identity data from an authoritative system to an identity store and to provision user accounts to target applications.

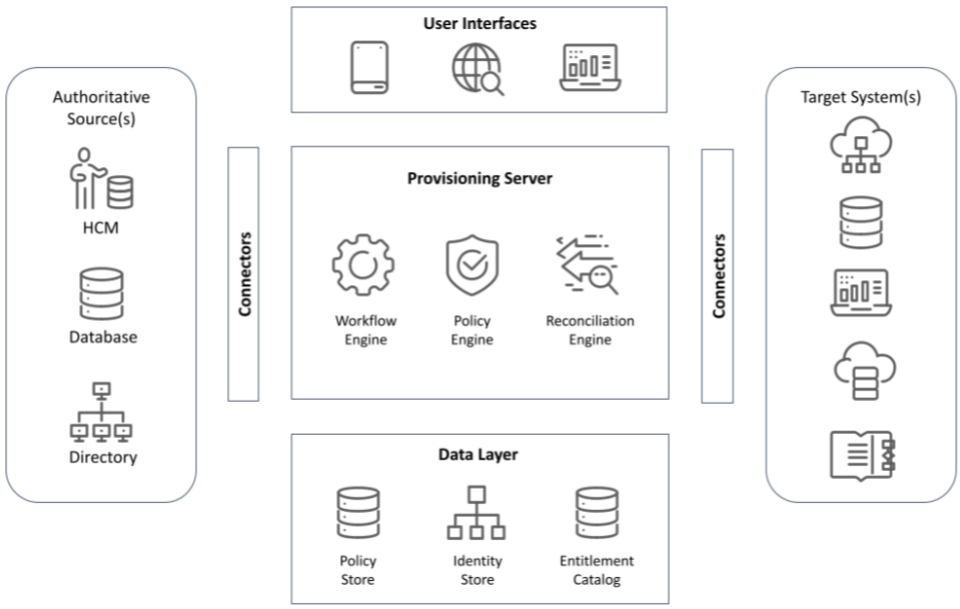


Figure 1: illustrates the standard architectural components of a user-provisioning ecosystem.

Authoritative source(s): The system of record (SOR) for identity data. The authoritative system publishes changes to the provisioning system. There may be more than one authoritative system in the environment. For example, in workforce use cases, the Human Capital Management (HCM) / Human Resources (HR) system may be the SOR for employee data, but contractor data may be stored in a procurement system.

Target system(s): Target systems subscribe to changes to identity records and are on the receiving end of the provisioning process. The provisioning system creates and manages user accounts and associated entitlements within the target system environment.

Connectors: The integration layer between the provisioning system and authoritative and target systems. There are various types of connectors: proprietary (application-specific connectors that communicate with app-specific APIs), generic (e.g., LDAP, JDBC, delimited text), or standards-based. See the standards section for more information on standards-based connectivity.

Provisioning server: The middleware layer responsible for data synchronization, mapping, and transformation; the application of business logic and access policies; and the orchestration of provisioning process flows. The provisioning server is comprised of the following functional components:

- **Account correlation rules:** correlates or matches disparate user accounts (in target or authoritative systems) with a single identity record and ensure that duplicate identities for a single person/entity are not created.
- **Data mapping rules:** maps and transforms data from the source context to the target context.
- **Account creation rules:** establishes standards for creating an identity record such as naming conventions, required attributes, password policy, location policy, etc.
- **Access policies**: determines access rights and entitlements that should be assigned to a user. See the Policies section for information on different types of access policies.
- **Workflow engine:** orchestrates the provisioning based on business processing logic and enables access request, approval, and review workflows.
- **Reconciliation engine:** discovers user accounts created directly in target systems (circumventing standard provisioning processes), ensures that the user account is in compliance with access policies, and correlates the user account with the individual's identity record.

Identity repository: Identity records are stored in an identity repository. The identity repository is a directory or a database that can be referenced by external systems and services (such as authentication or authorization services). The identity record includes attributes associated with the identity and a record of all user accounts associated with the identity.

Entitlement catalog: A database of entitlements and their related metadata. The catalog includes an index of entitlement data pulled from business systems, applications, and platforms. The entitlement data can be enriched with metadata such as risk scores and business-friendly descriptions of entitlements that can be displayed to users during access requests, access reviews, and certifications.

System configuration and audit store: A dedicated repository to hold information such as system configuration, identity mapping, policy, role definition, and workflow data. This repository may also serve as the store for audit logs.

User interfaces: User-provisioning systems include administrative, end-user, and delegated administration interfaces. Administrative interfaces are used for the set-up and configuration of the system. End-user and delegated administration interfaces are used for access requests, approval workflows, reporting, profile updates, etc. Provisioning systems typically include web-based interfaces that can be accessed from a pc or mobile device. While not standard, some provisioning vendors offer a mobile app for self-service and approval workflows.

Given their highly connected and interactive nature, user-provisioning systems must be open and extensible. The provisioning provider should provide open APIs, no or low code workflows, and generic connectors that allow for flexibility in the system.

User Provisioning Process Flow

User-provisioning technologies allow organizations to efficiently manage thousands of identities by capturing lifecycle events and ensuring that user accounts and their associated privileges are kept up-to-date and accurate. These processes reduce administrative overhead and improve security. That said, automated user account provisioning is a complex, multifaceted process that includes three distinct phases:

- **Event trigger**: A business event or a change to an identity that triggers a provisioning action
- **Policy administration**: Application of access policies that bind the identity to specific user accounts and entitlements
- **User account provisioning**: Creation, maintenance, deactivation, or deletion of user accounts in target applications

Event Trigger

The act of provisioning begins with an event. Such an event could be:

- The creation of a new employee in an HR system.
- A modification to an entry in Active Directory moving a person from one business unit to another.
- A ticket being created in an IT Service Management (ITSM) or Help Desk ticketing system.
- A person directly interacting with the provisioning system to request a change to a user account.

There are three primary types of events:

- Join
- Move
- Leave

Joiners, Movers, and Leavers (JML) are grist for the user provisioning mill. Managing JML processes becomes the work of an identity system. This work includes connecting the user-provisioning system to trigger sources and then constructing policies to be evaluated for each event type for each target system.

Join

The easiest way to think about the Join event is when a new employee joins a company. She needs her benefits and payroll set up along with user accounts in IT systems. In its

purest sense, Join events are meant to create a net new identity and net new user accounts in IT systems.[i]

Move

When a person changes roles with an enterprise, she likely needs access to new business systems and to have access to her older ones removed. This is the purpose of the Move event. You can think of Move as a change in the relationship between the organization and person. They might change which business unit they report to, get promoted, or change their last name.[ii]

Leave

A person retiring is the simplest example of a Leave event. In such a case, the person's user accounts need to be deleted or at least locked to prevent further use in all target systems. Another example is when a contractor's project concludes and the contract ends. The story is the same; a Leave event triggers the user-provisioning system to remove their access.

User-provisioning technologies provide various mechanisms to capture JML events, including:

- **Automated provisioning:** The user-provisioning system "listens" for events from systems of record such as Human Resources, ITSM, or a directory.
- **Batch processing:** The provisioning system executes a regularly scheduled process that polls an authoritative source for changes and generates an output file.
- **Self-service request:** Today's user provisioning solutions include an end-user access request portal where end-users or managers can request access to specific systems and rights needed to perform their business responsibilities. The user or a delegated administrator updates the user profile or makes an access change request via the self-service interface.
- **Manual/Ticket:** In some instances, an organization may use a ticketing system or other manual process to notify the identity team of a change needed on the identity record. In this case, the identity administrator would update the identity record directly to trigger downstream policy and provisioning activities.
- **Reconciliation event:** Reconciliation is the process of identifying and processing changes to users and user access made directly on target systems. When an organization configures a user provisioning solution for centralized management of user access, that does not prevent changes from occurring directly on a target system. So, to ensure the consistency of user access and user attributes across the organization, the user-provisioning system will periodically ***reconcile*** what it knows about users and their access to a specific target system. This reconciliation is accomplished by gathering and comparing all user data on the target system (full reconciliation) or by processing known changes to user access based on a changelog or other time-based query. When changes or variances are identified in a reconciliation process, events are triggered and processed based on defined

policies. The result of reconciliation could be to synchronize changes from the target system to other systems, or it could be to roll back any locally applied changes that occurred outside of the user provisioning solution.

Policy Administration

In the past, organizations have managed users' access to target systems in an ad hoc manner; given the complexities of the enterprise environment, this is no longer viable. They need documented rules to determine who should have access to which target systems; furthermore, they need to control what kind of entitlements and privileges people have in those target systems. This is the role of policies in a user-provisioning system. Instead of leaving the details to an administrator to determine which groups a user account ought to be a member of, a policy can describe which groups are required, optional, and even forbidden for people to be a member of.

A policy can be thought of as a way to bind groups of people to groups of target systems with groups of related access (entitlements, privileges, etc.) In this way, there are always two components of user provisioning: Who and What. The Who portion of the policy describes the inclusion criteria for which people the policy will apply. For example, all full-time employees, contractors, and finance people are all examples of the Who portion of a policy. The What portion of the policy describes the user accounts and associated entitlements and privileges a person gets. The What can be very coarse-grained, for example, the creation of a user account in all target systems, to very fine-grained, as when this specific entitlement and these two specific privileges are created in the target system.

Different kinds of policies use different combinations of Who and What to help identity practitioners govern access. While the overall topic of policies can be extremely broad, for the purposes of this article, let's focus on four kinds of policies:

- Birthright
- Role-based
- Segregation of duties
- Workflow approvals

It is important to note that a user provisioning process won't have just a single policy for a target system or event. Policies can be combined and applied to multiple target systems and triggering events.

Birthright

There are specific systems and entitlements that often broad swaths of the organization need; this kind of access is considered a birthright. Examples of such policies include:

- All full-time employees need email, calendar, collaboration, and file sharing.
- Everyone in the Finance department needs at least minimal access to the financial reporting system.

- Interns need access to the 'Intern Team Excellence' collaboration channel.

Birthright policies can be thought of as defining access that is fundamental to certain kinds of people who have a relationship with the organization. Such access does not need additional scrutiny, review, or approval; simply by being a person who matches the criteria of the policy (such as being a member of the Finance department) that person is allowed to have and will get user accounts in certain systems with specified levels of access (as managed by their user account's associated entitlements and privileges.) More often than not, birthright policies grant coarse-grained access to target systems; that is to say, they might only give someone a user account in an email system but not necessarily access to specific distribution groups. Birthright policies are most commonly applied as part of a Join event and typically occur by assigning one or more business roles. Birthright events can also happen as a part of a Move event, specifically when a person moves from one business function to another. For example, when a person is hired into the Accounting division within an organization, they'd receive birthright access to things like email and the productivity suite and basic access to critical accounting systems. When that person then transfers to Corporate Strategy, they lose access to the account systems, gain access to budget forecasting systems, and continue to retain access to email and productivity tools.

Role-based

Because an organization can have many business functions and thus lots of different business responsibilities as well as tens of thousands of individual entitlements in their systems, there is no way to manage who gets access at an individual level. Trying to do so would quickly lead to the management of tens of millions of combinations of people and privileges. User-provisioning systems attempt to bring order to this chaos by using roles to aggregate people and entitlements into more manageable policy components.

Much is made of roles in identity management.[iii] Roles can come in a variety of flavors; this article focuses on business roles and technical roles. A business role is a way to aggregate people who share the same business responsibilities. For example, a retail banking organization might have a business role called "Teller" and use it to describe the access appropriate for people who work as tellers. The second kind of role we'll need to understand for this article is a "Technical" role. A technical role is a way to aggregate the entitlements and privileges required within one or more target systems to perform a task. For example, the same retail bank could have a technical role called "Check and Update Balances," which gives user accounts in their systems the ability to check and update savings account balances.

A role-based policy in a user-provisioning system uses business and technical roles to govern access for more specific sets of people, entitlements, and privileges in target systems than birthright policies do. For example, a user-provisioning system might have a birthright policy that gives all full-time employees access to email. An additional role-based policy might grant Tellers access to a specific mailing list and shared drive.

Segregation of Duties

Stemming from the fallout of the WorldCom and Enron accounting scandals, the Sarbanes-Oxley Act had a profound impact on business practices.[iv] These impacts made their way to user provisioning. As part of compliance activities, organizations looked to their user provisioning policies to not only grant access to people but also prevent "toxic combinations" of access. A toxic combination of access is one in which a person has privileges that could enable some form of fraud, such as the ability to create a new vendor and issue a payment to that vendor. This combination of access would allow a bad actor to create a fictitious company in the financial system and then divert monies to that company. Another application of a toxic combination policy is to prevent anyone who isn't a system administrator from having system admin or highly privileged entitlements.

If roles are a way of describing what someone should have, then segregation of duty (SoD) policies are a way of describing what they must not have. Such policies are typically evaluated when a provisioning event is triggered so new toxic combinations are not introduced into target systems and existing ones are detected and remediated.

Workflow Approvals

Workflow approvals are an essential component of the policy management toolbox. Organizations with a mature provisioning deployment may only auto-provision 70-80% of access using birthright rules, roles, or SoD. So how does the remaining 20-30% of access get provisioned? The answer is self-service access request and approval workflows.

Workflow approvals are used when a human needs to make a policy decision. If a rule or role is not available, the provisioning system invokes a workflow process that routes an access request to a designee for approval. For example, an employee may make a self-service access request that is routed to a line manager for approval. The workflow approval process applies a layer of control and documents the access policy decision.

User Account Provisioning

Once a provisioning event has been triggered and policy evaluated to determine what user account attributes, entitlements, and privileges need to be set or changed, then that information needs to make its way into the target system to affect the change to the user account stored locally there. How the necessary changes are made in the target system is the act of provisioning. Provisioning can be accomplished in two primary ways:

- Automated
- Manual

Automated

Automated user account provisioning is the process of creating and maintaining a user account in the target system using automated processing. To automate the user account

provisioning process, the target system must provide a user management API or other means for the user-provisioning solution to systematically create, manage, and deactivate/delete user accounts.

Automated user account provisioning in target systems is the ultimate goal of user-provisioning technologies, but it is not without challenges. Each target system is an island. The user-provisioning system must maintain connections to the various target systems, which can be a heavy lift.

Manual

Manual provisioning requires human intervention to affect the change to the user account in the target system. This intervention often takes the form of the details of a user-provisioning event being sent to a team or a person who takes that information and manually keys it into the target system, using the target system's unique user management interfaces. The information required could be sent via email or work ticket.

Manual provisioning introduces humans into a critical step of user provisioning, creating two specific risks. First, the person who manually works with the target system to create and change user accounts, by definition of the work she does, is a highly privileged user. It is a good practice to minimize the distribution of such privileges, but sometimes it is necessary. The second risk, manual provisioning, introduces is the possibility of human error. The person might misread or mistype an attribute, entitlement, or privilege, thus incorrectly setting the user account in the target system. While that might result in a minor annoyance, such as misspelling a user's name, it might also lead to the assignment of incorrect privileges or even a toxic combination of entitlements.

It is fair to ask why manual provisioning is needed or wanted, given such risks. Manual provisioning is needed because not all target systems have APIs to which automatic provisioning connectors can connect. That homegrown general ledger system running on an extremely old operating system is an example of such. Another example is situations in which the target system is actually managed by a managed service provider and the identity team does not have direct access to that service provider. In that case, the change to a user account needs to be sent to the managed service provider via an email or ticket to trigger them to make the necessary change.

Manual provisioning is wanted because automation isn't worth the effort. Consider an application with very few users, entitlements, or changes required, or all three. An identity team may decide that it is not worth deploying (or possibly building) an automatic provisioning connector but instead choose to accept the human cost and risk of manual provisioning. It is a best practice to apply automated provisioning to high volume (lots of users), high velocity (frequent changes to user accounts), and high value (mission-critical, financial material, etc.) systems. Conversely, it is not a best practice to automate every

single system in the enterprise because, eventually, the costs to maintain connectors are simply not worth it.

The Role of Standards

The identity industry recognized that the proliferation of proprietary user management APIs would lead to a lack of automated provisioning and make it difficult for organizations to mitigate the risks inherent in manual user provisioning. Starting with Directory Service Markup Language in 1999, followed by Service Provisioning Markup Language[v] (SPML) in 2003, and finally followed by System for Cross-domain Identity Management[vi] (SCIM) in 2011, the industry has produced standards. The latest version of SCIM, version 2, has had significant uptake and, as of the second half of 2021, signs that the standards community is interested in making further enhancements. The fact that there have been at least three different standards with multiple versions is a testament to both the challenge of building a viable standard and the changes in the application development world.

For a user provisioning standard to be considered successful, it requires adoption from both user-provisioning system providers and application vendors. This "it takes two" challenge had thwarted mass adoption, especially in the era of on-premise software. The era of cloud computing and SaaS has seen a marked increase in the number of service providers willing to use SCIM v2 as well as user-provisioning technology providers. If the reader's IT organization is building custom applications, it is worth investigating the implementation of SCIM v2 in those apps to facilitate automated user provisioning, especially in high-volume, high-velocity, and high-value applications.

Why is User Provisioning Challenging?

User-provisioning systems have been on the market for 20+ years. In that time, they have garnered a reputation for being difficult and expensive to deploy, and many organizations have found it challenging to realize a return on investment. Why?

User provisioning is similar to data integration technologies in many ways. Like data integration technologies, user-provisioning systems aggregate and synchronize data to many different systems and services in the environment. Each new connection adds a new level of complexity. The process of onboarding a single application to a provisioning environment requires an understanding of the application's user management APIs and authorization construct, deployment of a connector, configuration of access policies, and implementation of policies and procedures for managing users (e.g., rules against creating or managing users directly within the application). Adding one application can be complex; consider the complexity when you have hundreds or even thousands of applications in your environment.

Another aspect that can be difficult is data quality issues in the SOR. Ideally, there is one authoritative SOR, but this is not always the case. Data collisions may happen when information is coming from multiple authoritative sources. Also, the administrators and users of the SOR may not understand the downstream effects of insufficient and poor-quality data. For example, they may not populate certain fields, enter inaccurate data, or delay event triggers. This all has implications for the provisioning system's ability to accurately update identity records and access entitlements. Managing the data quality process can be taxing on identity practitioners.

Another common challenge is policy definition. Identity practitioners are responsible for configuring access policies (rules/roles), but they don't own access decisions. The line of business in partnership with audit, legal, governance, risk management and compliance (GRC), etc., own access decisions. The effort to collect this data for provisioning policy and role definition is a significant undertaking.

Last but not least, maintaining the entitlement catalog can be a difficult task. A single organization may have hundreds of thousands, if not millions, of entitlements. The effort to collect entitlements and metadata should not be underestimated.

While the advantages of automated user account provisioning are well understood, deployments can be challenging. Identity practitioners should obtain executive support and set proper expectations, build a structured process for onboarding applications (e.g., dedicated resources, intake forms/surveys, automation, etc.), and set clear key performance indicators that show continued progress.

The Next Generation, Hybrid-Approach to Provisioning

While user-provisioning technologies have been around for quite some time, user-provisioning technologies were developed at a time when the IT environment was much more contained. Target systems and systems of record were located on-premises, and users were primarily employees accessing resources onsite.

Cloud, mobile, work from home, and various other initiatives have changed the dynamics of user provisioning. Now authoritative and target systems are hosted in the cloud (and on-premises), and external users are accessing internal resources.

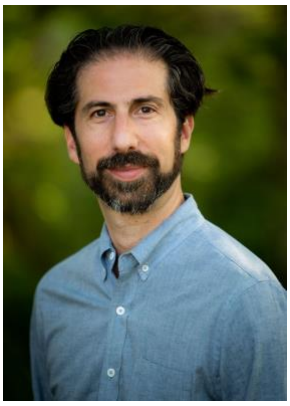
In traditional security models, provisioning is an admin-time function (users are pre-provisioned into systems by an administrator). Contrast this with authentication and authorization technologies that are run-time functions that occur at the point the user is logging into the system. Security technologies like SIEM, DLP, and threat detection kick in once the user is in session.

Modern security models that include remote access, cloud computing, and zero-trust security principles require a new approach to user provisioning: a just-in-time (JIT), least privilege approach. This modern approach to provisioning means that rather than pre-provisioning users at admin-time, users are provisioned at run-time and they are given the minimal privileges necessary to complete a task. Consequently, organizations are deploying a hybrid approach to provisioning that includes a mix of traditional API-based, SAML/OIDC-based, and SCIM-based connectors.

A hybrid provisioning architecture allows organizations to pre-provision to a set of applications (typically on-premises) but use OIDC, SAML, and SCIM-based connectors to enable JIT provisioning (primarily used in cloud and SaaS product use cases).

Author Bios

Ian Glazer

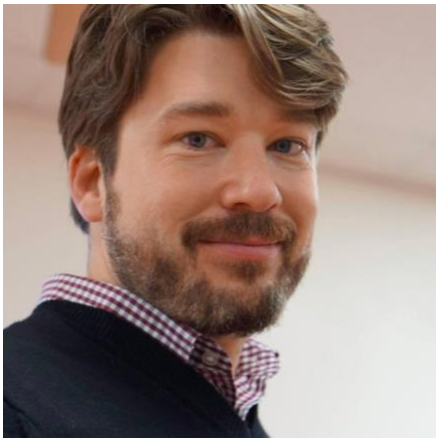


Ian Glazer is the founder and president of Weave Identity – an advisory services firm. Prior to founding Weave, Ian was the Senior Vice President for Identity Product Management at Salesforce. His responsibilities include leading the product management team, product strategy, and identity standards work. Earlier in his career, Ian was a research vice president and agenda manager on the Identity and Privacy Strategies team at Gartner, where he oversaw the entire team's research. He is a Board Emeritus and the co-founder of IDPro and works to deliver more services and value to the IDPro membership, raise funds for the organization, and help identity management professionals learn from one another. During his career in the identity industry, he has co-authored a patent on federated user provisioning, co-authored and contributed to user provisioning specifications, and is a noted blogger, speaker, and photographer of his socks.

Lori Robinson

Lori Robinson is the Vice President of Enterprise Identity Product Management at Salesforce where leads a team responsible for Salesforce's enterprise identity management program. Before joining Salesforce she was the VP Product and Market Strategy at SailPoint. She also served as the Managing Vice President and Analyst at Gartner where she covered the identity governance and administration, privileged access management, and consumer IAM markets. Lori is a recognized industry thought leader, speaker, and publisher. She is passionate about advancing opportunities for women in IT and has led various user groups, round tables, and events for women in identity.

Mat Hamlin



Mat Hamlin is the Vice President of Product Management for Platform Identity at Salesforce. His responsibilities include leading the product management team, product strategy, and innovation for the Identity Services on the Salesforce core platform. Prior to Salesforce, he served in multiple product management roles at SailPoint, Oracle, and Sun Microsystems, focusing on User Provisioning, Access Governance, and Enterprise Role Management.

Change Log

Date	Change
2022-06-03	V1 published

[i] More on Joiner, Mover, and Leaver is available in Cameron, A. & Grewe, O. (2022) "An Overview of the Digital Identity Lifecycle (v2)", *IDPro Body of Knowledge* 1(7). doi: https://doi.org/10.55621/idpro.31

[ii] Changes to name or place of residence can be just as impactful as a change in job role or reporting structure.

[iii] More on roles is available in McKee, M. K., (2021) "Policy-Based Access Controls", *IDPro Body of Knowledge* 1(4). doi: https://doi.org/10.55621/idpro.61 and Koot, A., (2020) "Introduction to Access Control (v3)", *IDPro Body of Knowledge* 1(6). doi: https://doi.org/10.55621/idpro.42.

[iv] Corporate Finance Institute, "Top Accounting Scandals: A recap of the top scandals in the past," n.d., https://corporatefinanceinstitute.com/resources/knowledge/other/top-accounting-scandals/ (accessed 17 May 2022).

[v] For the sake of transparency, one of the authors of this article contributed to the SPML v2 standard and apologizes for the mistakes he didn't know he was making at the time.

[vi] For the sake of transparency, one of the authors of this article contributed to the SCIM v2 standard and is proud of that work.