

Identity Management Terminology



Identity management is an important technology for managing user objects, identity attributes, authentication factors and security entitlements. This is done by providing automated and self-service processes for on-boarding, termination and every change that impacts a user between these events.

Identity management encompasses a wide range of technologies and processes and consequently there may be ill defined or conflicting terminology relating to key concepts.

This document introduces key identity management terminology and offers clear, unambiguous definitions. The intent is to help the reader focus on solving real problems, rather than waste energy on the language of identity management.

Contents

- 1 Introduction** **1**

- 2 Participants in Identity Management** **1**

- 3 Business Processes** **2**

- 4 Business Processes** **3**

- 5 Identification** **4**

- 6 Authentication** **5**
 - 6.1 Passwords 6
 - 6.2 Lockouts and Expiration 6
 - 6.3 Challenge/Response 7
 - 6.4 Hardware and Software Tokens 7
 - 6.5 Biometrics 8
 - 6.6 PKI Certificates and Smart Cards 8
 - 6.7 Location-based Authentication 9

- 7 Authorization** **10**
 - 7.1 Access Control Lists 10
 - 7.2 Security Groups 10
 - 7.3 Virtual Groups 11
 - 7.4 Segregation of Duties 11
 - 7.5 Entitlement Management 12
 - 7.6 Role-based Access Control 13

8 Audit / Access Certification	15
8.1 Change Management	15
9 Directory	16
10 Single Signon	18
10.1 Token Passing Approaches	18
10.2 Enterprise Single Signon	19
10.3 Web Single Signon	19
10.4 Federation	20
11 Password Management	21
11.1 Password Policy	21
11.2 Password Synchronization	22
11.3 Self-Service Password Reset	23
11.4 Password Wallets	23
11.5 Password Recovery	23
12 User Provisioning	24
12.1 Automated Provisioning	25
12.2 Consolidated and Delegated Administration	25
12.3 Target Systems	26
13 Privileged Password Management	27
13.1 Sensitive Passwords	28
13.2 Password Locations	28
13.3 Password Disclosure	29
14 User Interfaces	29

1 Introduction

Identity management is an important technology for managing user objects, identity attributes, authentication factors and security entitlements. This is done by providing automated and self-service processes for on-boarding, termination and every change that impacts a user between these events.

Identity management encompasses a wide range of technologies and processes and consequently there may be ill defined or conflicting terminology relating to key concepts.

This document introduces key identity management terminology and offers clear, unambiguous definitions. The intent is to help the reader focus on solving real problems, rather than waste energy on the language of identity management.

2 Participants in Identity Management

- | | | |
|---|------------------------------------|--|
| 1 | User | Users are people whose access to systems and identity information must be managed. |
| 2 | Support Analyst | An IT support analyst is a user with special privileges, that allow him to assist other users, for example by resetting their forgotten passwords. |
| 3 | Target System Administrator | A system administrator is a user with absolute control over a target system (see 151 on Page 26). The system administrator may install any or all software on the managed system, can create or delete other users on that system, etc. |
| 4 | Security Administrator | A security administrator is a person responsible for maintaining a list of users, their identity attributes, their passwords or other authentication factors and their security privileges on one or more target systems (see 151 on Page 26). The security administrator may not have the responsibility or ability to reconfigure or otherwise manage the system itself – that is the job of a system administrator (see 3 on Page 1). |
| 5 | Application Owner | An application's owner is a person in a business organization who may have authorized purchase of the application and is in any case responsible for the use of that system. This is a business rather than technical role. |
| 6 | Data Owner | A data owner is a business role associated with responsibility for a given set of data. Normally this comes with responsibility to decide what users (see 1 on Page 1) in the organization may access the data in question and for the quality of the data. |
| 7 | Group Owner | Access to data, to applications and to features within applications is often controlled using security groups (see 74 on Page 10). Groups normally have owners – people in an organization responsible for managing membership in the group (see 76 on Page 11). |

- | | | |
|----|-----------------------------|---|
| 8 | Management Chain | In a business setting, users normally have managers, who in turn have their own managers. The sequence of managers, starting with a given user and ending with the highest individual in an organization is that user's management chain. Management chains are relevant to identity management as they are often used to authorize security changes. |
| 9 | Requester | Changes to user profiles or entitlements are often initiated by a requester – literally a person who makes a change request. In other cases they may be initiated by an automated process, which may or may not have a “virtual” (i.e., non-human) ID. |
| 10 | Recipient | Changes to user profiles or entitlements always have a recipient – that user profile which will be created, modified or deleted. |
| 11 | Authorizer | Changes to user profiles or entitlements may be subject to approval before they are acted on. In cases where approval is required, one or more authorizers are assigned that responsibility. |
| 12 | Delegated Authorizer | A given authorizer may not always be available. For example, authorizers may take holidays, be ill, be too busy to respond, etc. In these cases, an authorizer may wish to delegate his authority to another user – temporarily or permanent. The new authorizer is a delegated one. |
| 13 | Escalated Authorizer | A given authorizer may not always be available. In cases where an authorizer fails to respond to a request to approve or reject a requested change, and where the authorizer has not named a delegated authorizer (see 12 on Page 2), an automatic escalation process may select a replacement authorizer after a period of time. This replacement is the escalated authorizer. |

3 Business Processes

User profiles are created, changed and deleted in response to business processes. This section captures the most important processes that drive identity management.

- | | | |
|----|-----------------------|---|
| 14 | Onboarding | This is the process where users join an organization. It may refer to hiring new employees, bringing in contractors or signing up visitors to a web portal. |
| 15 | Access Support | Users may sometimes experience difficulty in relation to their security privileges (see 72 on Page 10). They will then typically contact a support analyst (see 2 on Page 1) for assistance, and that person will adjust their access rights. |

- | | | |
|----|------------------------------------|---|
| 16 | Authentication Support | Users may sometimes experience difficulty signing into a system or application. They may have forgotten their password or triggered an intruder lockout (see 43 on Page 6). In these cases, they may contact a support analyst (see 2 on Page 1) for assistance, such as a password reset (see 141 on Page 23). |
| 17 | Identity Change | User identity information may change for time to time. For example, people change their names after marriage or divorce, their phone number and address changes periodically, etc. Where systems and applications track this data, it must be changed whenever the real-world information changes. Such changes are called identity changes. |
| 18 | Security Entitlement Change | Users' needs to access sensitive resources may change for time to time. For example, an employee may join a new project, finish an old one or change roles. When this happens, new security entitlements are often needed and old ones should be removed. |
| 19 | Security Entitlement Audit | In many organizations, security entitlements have to be reviewed from time to time. This is done because business processes relating to changing needs (see 18 on Page 3) are often reliable with respect to granting new entitlements, but less reliable with respect to deactivating old, unneeded entitlements. A periodic audit can be used to find and remove such old, unneeded entitlements. |
| 20 | Termination | All users eventually leave an organization. Likewise, customers may terminate their relationship with vendors. Generically, these events are called termination. |

4 Business Processes

- | | | |
|----|------------------------------|---|
| 21 | User Creation | When users join an organization, they are normally granted access to systems and applications. This is called user creation. |
| 22 | Access Deactivation | When termination happens, user access rights relating to an organization's systems and applications must be removed. This removal is called access deactivation. |
| 23 | Application Migration | Vendors release new versions of their software all the time. When this happens, customers often choose to upgrade. Upgrades may require data from the old system, including data about users, to be migrated to the new system. An identity management system can be used to aid in this migration process. |

5 Identification

- 24 **Login Accounts** | Systems and applications where users have the ability to login and access features and data generally assign a login account to each user. Login accounts usually include a unique identifier for the user, some means of authentication (see [Section 6](#) on [Page 5](#)), security entitlements and other, personally identifying information such as the user's name, location, etc.
- 25 **Login ID** | The unique identifier that a user types to sign into a system or application is that user's login ID on that system.
- 26 **Identity Attributes** | Each piece of identifying information about a user can be thought of as an attribute of that user. Users have identity attributes, each of which may be stored on one or more target systems.
- 27 **User Profile** | The set of login accounts (see [24](#) on [Page 4](#)), identity attributes (see [26](#) on [Page 4](#)) and security entitlements (see [72](#) on [Page 10](#)) associated with a single (human) user.
- 28 **Profile ID** | A profile ID is a globally unique identifier for a human user.
- 29 **Alias** | An alias is a local ID that a user has on a given system (see [151](#) on [Page 26](#)) which is different from the user's global ID (see [28](#) on [Page 4](#)).
- 30 **ID Name Space** | A name space for unique identifiers is a system or domain within which no two users may have the same ID. Every system (see [151](#) on [Page 26](#)) has its own namespace. Another example is mail domains (i.e., the part of an SMTP e-mail address following the @ sign), where the part of each user ID preceding the @ sign must be unique within its domain.
- 31 **Global ID** | A global ID is a unique identifier that spans two or more systems. A truly global ID – one that is guaranteed to be unique among every system in the world, is a user's fully qualified SMTP e-mail address. Another truly global ID might be a user's country code followed by that country's local equivalent of a social security number, social insurance number or resident number. Global IDs may be global only over a few systems, rather than every system on Earth.
- 32 **Local ID** | A local ID is a user's unique identifier within the context of a single system (see [151](#) on [Page 26](#)). It may be the same as that user's profile ID (see [28](#) on [Page 4](#)), or it may be an alias (see [29](#) on [Page 4](#)).

- 33 **ID Reconciliation** | ID reconciliation is a process by which an organization maps local IDs (see 32 on Page 4) in different name spaces (see 30 on Page 4) to one-another, and to the global profile IDs (see 28 on Page 4) of the users that own them. For example, ID reconciliation may be required to map IDs such as “smithj” on a mainframe system to IDs such as “john.w.smith” on an Active Directory domain.

6 Authentication

- 34 **Authentication** | Authentication is a process by which a user proves his identity to a system – normally when logging in.
- 35 **Authentication Factor** | An authentication factor is something a user presents to a system in order to prove his identity. It may be something he (and hopefully only he) knows, or proof of possession of a physical object, or a measurement of some physical characteristic (biometric) of the living human user. In other words, something the user knows, or something he has, or something he is.
- 36 **Multi-Factor Authentication** | Multi-factor authentication means authentication using multiple factors (see 35 on Page 5). For example, a user might sign into a system with a combination of two things he knows, or a combination of something he knows and something he has, or perhaps something he knows, something he has and something he is.

The premise is that adding authentication factors makes it more difficult for a would-be attacker to simulate a legitimate authentication and consequently impersonate a legitimate user.
- 37 **Strong Authentication** | Strong authentication refers to an authentication process (see 34 on Page 5) which is difficult to simulate. It may be based on use of multiple authentication factors (see 36 on Page 5) or use of a single but hard-to-spoof authentication factor (see 35 on Page 5).
- 38 **Security Equivalence** | Two authentication processes are considered to be equivalent if (a) they are about equally difficult to defeat or (b) by defeating one of them, an intruder can subsequently defeat the other.

An example of the latter is a PIN-based enrollment of challenge/response data. In this scenario, users are e-mailed a PIN, which they use to authenticate and complete a personal challenge/response profile. This profile may later be used in the context of self-service password reset. In this scenario, the PINs are equivalent to the challenge/response data, and that is equivalent to user login passwords – so ultimately enrollment PINs are security-equivalent to login passwords (a bad thing!).

- 39 **Security Credentials** | Credentials are the data used to both identify and authenticate a user. The most common credentials are login IDs and passwords. Other credentials refer to other types of authentication factors, including biometric samples of the user, public key certificates, etc.

6.1 Passwords

- 40 **Password Authentication** | The most common authentication factor (see 35 on Page 5) is a password. It is a string of characters that is known to the user and to the system into which the user signs in, but (hopefully) kept secret from other users and systems.
- 41 **Personal Identification Number (PIN)** | A PIN is a short, numeric password (see 40 on Page 6). PINs are commonly used with bank debit cards and as a secondary authentication factor (see 36 on Page 5) accompanying technologies such as biometrics or hardware tokens.
- 42 **Pass Phrase** | A pass phrase is a longer password, where users are encouraged to type multiple words, rather than just one, in order to make it more difficult for a would-be attacker to guess the password value..

6.2 Lockouts and Expiration

- 43 **Intruder Lockout** | An intruder lockout is a flag set on a login account (see 24 on Page 4) when too many consecutive, failed login attempts have been made in too short a time period. Intruder lockouts are intended to prevent attackers from carrying out brute force password guessing attacks.
- On some systems, intruder lockouts are cleared automatically, after a period of time has elapsed. On others, administrative intervention is required to clear a lockout.
- Note that on some systems and applications, intruder lockouts and administrator lockouts are entangled (they use the same flag). This is a poor but common design.
- 44 **Administrator Lockout** | An administrator lockout is a flag set by an administrator to disable logins on an account (see 24 on Page 4).
- Administrator lockouts normally precede permanent deletion of the account, and provide an opportunity to retrieve data from the account before it is removed.
- Note that on some systems and applications, intruder lockouts and administrator lockouts are entangled (they use the same flag). This is a poor but common design.

45	Disabled Account	A disabled account is one where the administrator lockout flag has been set.
46	Expired Password	An account (see 24 on Page 4) is said to have an expired password if the user will be forced to change passwords (see 140 on Page 23) after the next successful login.
47	Account Termination Date	An account has a termination date if logins will not be possible after a given time/date.
48	Password Expiry Date	An password has an expiry date if the user will be forced to change it on the first successful login after a given time/date.

6.3 Challenge/Response

49	Challenge/Response Authentication	Often used as a backup for passwords, challenge/response authentication is where users are asked to answer a series of personal questions where no-one else is likely to know the answer. While individual personal questions may be poor forms of authentication, correct answers to a whole series of such questions may be sufficiently robust to be used as an authentication factor.
----	--	---

6.4 Hardware and Software Tokens

50	One-Time Password	A one-time password (OTP) is an algorithm used to produce a different password every time a user needs to authenticate. OTP passwords may be time-based (i.e., the password for any given minute/hour/date is different and may be computed both by the user and the system into which the user wishes to authenticate). OTP passwords may also be series based (the password value depends on the number of times the user has signed on before), or may be computed by the user in response to a challenge presented by the server.
51	Hardware Token	A hardware token is a small device, typically either the size of a credit card or suitable for attaching to a user's key chain, which computes a one time password (see 50 on Page 7). Users use a hardware token to prove possession of a device (i.e., something they have) as an authentication factor (see 35 on Page 5).
52	Software Token	A software token is the same as a hardware token (see 51 on Page 7) except that it is installed as a piece of software on a device that the user already has – such as a cell phone, PDA or the user's personal computer.

6.5 Biometrics

- | | | |
|----|---------------------------------|--|
| 53 | Biometric Authentication | Biometric authentication requires that some measurement of the user's body, metabolism or behaviour is compared to a similar measurement enrolled earlier. A successful match is used as a successful authentication. |
| 54 | Finger Print | A fingerprint is a form of biometric authentication where the characteristic being measured is the pattern of ridges on one or more of a user's fingers. |
| 55 | Finger Vein | Finger vein authentication is a measurement of the pattern of living veins inside one or more of a user's fingers. |
| 56 | Palm Print | A palm print is a form of biometric authentication where the characteristic being measured is the pattern of ridges on the skin of a user's whole hand. |
| 57 | Palm Vein | Palm vein authentication is a measurement of the pattern of living veins inside one or more of a user's whole hands. |
| 58 | Voice Print | A voice print is a form of biometric authentication where the characteristic being measured is the timbre, tone, speed, volume, etc. of the user's voice, typically speaking the same phrases at both enrollment and authentication times. |
| 59 | Iris Scan | An iris scan is an image of a user's iris pattern in one or both eyes. |
| 60 | Retina Scan | A retina scan is an image of the blood vessel pattern in one or both of a user's retinas. |
| 61 | Typing Cadence | The time interval between keystrokes when typing a particular phrase can be used to differentiate between different people typing the same phrase. |

6.6 PKI Certificates and Smart Cards

- | | | |
|----|------------------------------|---|
| 62 | Asymmetric Encryption | Asymmetric encryption is encryption where matching pairs of keys are used. What is encrypted with one key in a matched pair can only be decrypted by the other key – it cannot be decrypted with the original key, or with any other key. |
| 63 | Public Key | A public key is one of a two matched keys, which a user or system distributes widely and publicly. This key is well known to as many users and systems as possible as the user's public key. |

- | | |
|---------------------------------|---|
| 64 Private Key | A private key is one of a two matched keys, which a user or system keeps secret and makes an effort to protect. No-one but the user who generated a public/private key pair should have access to the user's private key. |
| 65 Certificate Authority | A certificate authority is an organization whose public key is very well known, whose private key is very well protected, and whose business function is to encrypt the public keys belonging to users and systems with its own private key and to publish the resulting encrypted public keys ((see 66 on Page 9)). |
| 66 Certificate | <p>A certificate is a public key that has been encrypted by a certificate authority (CA). Since the CA's public key is well known, anyone can decrypt the certificate to find the original public key.</p> <p>Since the CA's business is to verify that a given public key was generated by the user it purportedly comes from, public keys signed by the CA can be trusted to really belong to their stated owner.</p> <p>Certificates are useful for signature verification (a document is encrypted by the user's private key, and this is verified using the user's certificate) and authentication (a user is asked to encrypt something, and if the user's certificate can decrypt it, then the user must have possessed the matching private key).</p> |
| 67 Smart Card | <p>A smart card is a credit-card-sized device that houses an integrated circuit, with some processing and storage capabilities. Smart cards are often used to carry a user's private encryption key and one or more certificates (the user's signed public key or other keys).</p> <p>Smart cards are useful for authentication since they constitute an authentication factor (something the user has) and they often require a second factor (e.g., user typing in a password) to be activated, which is a second factor (something the user knows).</p> |

6.7 Location-based Authentication

- | | |
|----------------------------------|---|
| 68 Network Endpoint | A network endpoint is a device with which a user accessed network services. Examples include corporate or home PCs, smart phones, PDAs, Internet and Intranet kiosks, etc. |
| 69 Network Access Control | Network access control is a technology that validates the security settings, location, ownership, anti-malware software installation or other characteristics of a network endpoint before allowing that device to access network services. |

- 70 **Dial-Back** | Dial back validates a user's physical location using the telephone system. In its original form, when users connected their PCs to the network with telephone modems, a user would connect to a corporate network, identify himself, hang-up and wait for a corporate server to call him back at home.
- With more modern technology, a user may sign into a corporate network, identify himself and wait for a single-use random PIN to be phoned or text messaged to his home or cellular telephone. This PIN is subsequently used to authenticate to a network service.
- 71 **E-mail Based Authentication** | Applications may defer identification and authentication of a user to an e-mail system, essentially eliminating any need to manage or support the authentication process directly. This is typically as follows:
1. The user identifies himself to an application by typing his e-mail address.
 2. An e-mail containing a randomized URL is sent to that address.
 3. If the user can click on the e-mail, he has demonstrated that he has access to the e-mail account, and is therefore authenticated.
- This is a weak form of authentication, since it is impossible to say how secure the user's e-mail service is, but it is adequate for many applications.

7 Authorization

7.1 Access Control Lists

- 72 **Security Entitlements** | A security entitlement is a right granted to a user's account (see 24 on Page 4) on a given system (see 151 on Page 26) to access some data or function.
- 73 **Access Control List** | An access control list connects a user or group of users to one or more security entitlements. For example, users in group "accounting" are granted the entitlement "read-only" to the data "budget file."

7.2 Security Groups

- 74 **Security Group** | A security group is a named collection of users, which has been defined in order to simplify the assignment of entitlements (see 72 on Page 10). The idea is to assign multiple entitlements to the group, rather than assigning entitlements, again and again, to every user that belongs to the group.

- 75 **Nested Groups** | Nested groups are groups (see 74 on Page 10) that contain, among their members, other groups. This is a powerful construct but it can be complicated for applications to support and may cause performance problems if not implemented well. Active Directory is one system that effectively supports nested groups.
- 76 **Group Membership** | A group membership is the assignment of a given user (see 24 on Page 4) to a given security group (see 74 on Page 10).

7.3 Virtual Groups

- 77 **Virtual Group** | On some systems, management of membership (see 76 on Page 11) in large groups (see 74 on Page 10) does not scale. This may be due to technical problems with the underlying implementation. For example, on Sun or IBM LDAP directories, groups should not have more than a few thousand members, or else performance will suffer.
- In these cases, it may be preferable to create a “virtual” group, whose membership is not explicitly defined. Instead, membership in a virtual group is calculated at runtime, by evaluating a logical expression based on identity attributes (see 26 on Page 4). For example, users may be said to belong to a group “Dallas-Managers” if their location attribute is equal to “DFW” and their position attribute is set to “Manager.”
- In other words, virtual groups are named expressions that evaluate to boolean true for users that are considered to be members of a group.

7.4 Segregation of Duties

- 78 **Segregation of Duties Policy** | A segregation of duties (SoD) policy is a rule regarding user entitlements (see 72 on Page 10) intended to prevent fraud. It stipulates that one user may not concurrently be assigned two or more key functions in a sensitive business process.
- 79 **Static SoD Policy** | A static segregation of duties (see 78 on Page 11) policy is one that prevents one login account (see 24 on Page 4) or user profile (see 27 on Page 4) from having two or more conflicting entitlements (see 72 on Page 10). These entitlements may be thought of as a toxic combination. For example, the same user may not both authorize an expense and print the cheque to pay for it.

80 Dynamic SoD Policy

A dynamic segregation of duties (see 78 on Page 11) policy is one that prevents one login account (see 24 on Page 4) or user profile (see 27 on Page 4) from performing two or more conflicting actions relating to the same business transaction. For example, while it may be appropriate for the same user to have both the vendor-management and payment-management entitlements, it is not acceptable for the same user to both create a vendor and authorize a payment to that vendor.

7.5 Entitlement Management**81 Security Entitlement**

The Burton Group defines an entitlement as:

An entitlement is the object in a system's security model that can be granted or associated to a user account to enable that account to perform (or in some cases prevent the performance of) some set of actions in that system. It was commonly accepted that this definition of entitlement referred to the highest-order grantable object in a system's security model, such as an Active Directory group membership or SAP role, and not lower-order objects such as single-file permission setting.

Definition by Ian Glazer, in *Access Certification and Entitlement Management v1*, September 9, 2009.

<http://www.burtongroup.com/Client/Research/Document.aspx?cid=1732> (login required)

82 Entitlement Management	<p>Entitlement management refers to a set of technologies and processes used to coherently manage security rights across an organization. The objectives are to reduce the cost of administration, to improve service and to ensure that users get exactly the security rights they need.</p> <p>These objectives are attained by creating a set of robust, consistent processes to grant and revoke entitlements across multiple systems and applications:</p> <ol style="list-style-type: none"> 1. Create and regularly update a consolidated database of entitlements. 2. Define roles, so that entitlements can be assigned to users in sets that are easier for business users to understand. 3. Enable self-service requests and approvals, so that decisions about entitlements can be made by business users with contextual knowledge, rather than by IT staff. 4. Synchronize entitlements between systems, where appropriate. 5. Periodically invite business stake-holders to review entitlements and roles assigned to users and identify no-longer-appropriate ones for further examination and removal.
----------------------------------	--

7.6 Role-based Access Control

83 Simple Role	<p>A simple role is a collection of entitlements (see 72 on Page 10) defined within the context of a single system (see 151 on Page 26). Roles are used to simplify security administration on systems and applications, by encapsulating popular sets of entitlements and assigning them as packages, rather than individually, to users.</p>
84 Enterprise Role	<p>An enterprise role is a collection of entitlements (see 72 on Page 10) spanning multiple systems or applications (see 151 on Page 26). Like simple roles, enterprise roles are used to simplify security administration on systems and applications, by encapsulating popular sets of entitlements and assigning them as packages, rather than individually, to users.</p>
85 Role Change	<p>A role change is a business process where a user's job function changes and consequently the set of roles and entitlements that the user is assigned should also change. Some old entitlements should be removed (immediately or after a period of time), some old entitlements should be retained, and some new entitlements should be added.</p>
86 Explicit Role Assignment	<p>A role may be explicitly assigned to a user – i.e., some database will include a record of the form “user X should have role Y.”</p>

<p>87 Implicit Role Assignment</p>	<p>A role may be implicitly assigned to a user – i.e., some database will include a rule of the form “users matching requirements X should be automatically assigned role Y.”</p>
<p>88 Role Model</p>	<p>A role model is a set of role definitions and a set of implicit or explicit role assignments.</p>
<p>89 Entitlement Model</p>	<p>Entitlement (or privilege) model is a synonym for role model (see 88 on Page 14).</p>
<p>90 Role Management</p>	<p>Roles and role assignment are unlikely to remain static for any length of time. Because of this, they must be managed – the entitlements associated with a role must be reviewed and updated and the users assigned the role, implicitly or explicitly, must be reviewed and changed. The business processes used to effect these reviews and changes are collectively referred to as role management (sometimes enterprise role management).</p>
<p>91 Role Mining</p>	<p>Where enterprise roles (see 84 on Page 13) are used to manage entitlements, they must first be defined and assigned to users. These definitions normally take place in the context of an organization where users already have entitlements – some of them required for their jobs, and others inappropriate or stale. Role mining refers to an analysis of existing entitlements in an effort to extract a workable role model.</p>
<p>92 Role Policy Enforcement</p>	<p>Where entitlements on multiple systems are modeled with enterprise roles (see 88 on Page 14), an enforcement process can periodically compare actual entitlements with those predicted by the model and respond to variances – by automatically making corrections, asking for deviations to be approved, etc. This periodic checking process is called role policy enforcement.</p>
<p>93 Role Violation</p>	<p>A role violation is a situation where a user is assigned an entitlement that contradicts a user’s role assignment (see 84 on Page 13). The entitlement may be excessive – i.e., not predicted by the role, or it may be inadequate – i.e., the role assignment predicts that the user should have an entitlement, but the user does not.</p>
<p>94 Approved Exception</p>	<p>An approved exception is a role violation (see 93 on Page 14) which has been flagged as acceptable, and which consequently may be removed from violation reports and/or not corrected.</p>

8 Audit / Access Certification

- | | |
|------------------------------------|--|
| 95 Access Certification | Over time, users may accumulate entitlements (see 72 on Page 10) which are no longer needed or appropriate for their job function. Access certification is a process by which appropriate business stake-holders, such as users' managers or application owners, can periodically review entitlements and identify those that should be removed. |
| 96 Attestation | Attestation is synonymous with access certification (see 95 on Page 15). This term highlights the aspect of certification where stake-holders attest to the appropriateness of entitlements, rather than flagging those that should be removed. Both signing off on appropriate entitlements and flagging inappropriate ones should be done in tandem. |
| 97 Organizational Hierarchy | An organizational hierarchy is an organization of user profiles that identifies zero or one managers for each user. This hierarchy may be useful in the context of access certification (see 95 on Page 15), change authorization (see 99 on Page 15) or automated escalation (see 13 on Page 2). |

8.1 Change Management

- | | |
|-------------------------------|--|
| 98 Change Request | A change request consists of one or more proposed changes to user profiles (see 27 on Page 4), such as creating new profiles, adding new accounts (see 24 on Page 4) to existing profiles, changing identity attributes (see 26 on Page 4), Requests may be subject to authorization (see 99 on Page 15) before being implemented. |
| 99 Approval Workflow | An approval workflow is a business process where human actors may enter, review, approve, reject and/or implement a change request (see 98 on Page 15). |
| 100 Parallel Approvals | <p>A parallel authorization process is one where multiple authorizers are invited to comment concurrently – i.e., the identity management system does not wait for one authorizer to respond before inviting the next.</p> <p>Parallel authorization has the advantage of completing more quickly, as the time required to finish an authorization process is the single longest response time, rather than the sum of all response times.</p> |

- | | |
|---|---|
| 101 Consensus Authorization | <p>Approval by consensus is a form of parallel authorization where not all authorizers must respond before a change request is implemented. For example, any two of three authorizers may be sufficient to approve a request.</p> <p>Consensus authorization is implemented in order to expedite the approvals process and make sure that it is completed even in cases where some authorizers are unavailable to respond.</p> |
| 102 Veto Power | <p>Veto power is a right assigned to authorizers in an approvals process whereby rejection of a change request by the authorizer who has veto power cancels the request, regardless of any approvals previously received from other authorizers.</p> |
| 103 Sequential Approvals | <p>A sequential authorization process is one where multiple authorizers are invited to comment, one after another.</p> <p>Sequential (or serial) authorization has the advantage of minimizing the nuisance to authorizers in the event that an early authorizer rejects a change request (see 98 on Page 15).</p> |
| 104 Authorization Reminders | <p>Authorizers in an approvals process (see 99 on Page 15) may not respond to invitations to review a change request (see 98 on Page 15) in a timely manner. When this happens, automatic reminders may be sent to them, asking them again to review change requests.</p> |
| 105 Delegation of Approval Authority | <p>Authorizers may wish to schedule periods of time during which they will be unavailable (example: vacations), and during which their authority to approve change requests should be transferred to others. The process by which an authorizer transfers authority – temporarily or permanently – is delegation.</p> |
| 106 Automatic Escalation | <p>In the event that an authorizer has been invited to review a change request, has not responded, has been sent reminders (see 104 on Page 16), has nonetheless not responded, and has not delegated his authority (see 105 on Page 16), an identity management system may automatically select an alternate authorizer, rather than allow the approvals process to stall. Automatically rerouting requests to alternate authorizers is called escalation.</p> |

9 Directory

- | | |
|----------------------|--|
| 107 Directory | <p>A directory is a network service which lists participants in the network – users, computers, printers, groups, etc. It is intended to be a convenient and robust mechanism for publishing and consuming information about these participants.</p> |
|----------------------|--|

108 Directory Object	A directory object is an item in a directory. Example objects include users, user groups, computers and more. Objects may be organized into a hierarchy (see 109 on Page 17) and contain identifying attributes (see 26 on Page 4).
109 Directory Hierarchy	A directory can be organized into a hierarchy, in order to make it easier to browse or manage. Directory hierarchies normally represent something in the physical world, such as organizational hierarchies or physical locations. For example, the top level of a directory may represent a company, the next level down divisions, the next level down departments, etc. Alternately, the top level may represent the world, the next level down countries, next states or provinces, next cities, etc.
110 Lightweight Directory Access Protocol (LDAP)	LDAP is a simple and standardized network protocol used by applications to connect to a directory, search for objects and add, edit or remove objects.
111 LDAP over SSL	LDAPS is the short name for LDAP connections made over secure socket layers (SSL). Where LDAP is a plaintext protocol, LDAPS is encrypted and so more secure.
112 X.500 Protocols	X.500 is a family of standardized protocols for accessing, browsing and maintaining a directory. It is functionally similar to LDAP (see 110 on Page 17) but is generally considered to be more complex and has consequently not been widely adopted.
113 Virtual Directory	A virtual directory is an application that exposes a consolidated view of multiple physical directories over an LDAP interface. Consumers of the directory information connect to the virtual directory's LDAP service, and "behind the scenes" requests for information and updates to the directory are sent to one or more physical directories, where the actual information resides. Virtual directories enable organizations to create a consolidated view of information that - for legal or technical reasons - cannot be consolidated into a single physical copy.
114 Meta Directory	A meta directory is an application that collects information from two or more physical directories, to create a master copy with all relevant data about every object of interest. Conflicts, errors and omissions in the data may be corrected during this merge process, and the resulting data, which should be clean and correct, can then be sent back to the original directories. Meta directories are used to implement auto-provisioning (see 146 on Page 25), auto-termination (see 147 on Page 25) and identity synchronization (see 148 on Page 25).

10 Single Signon

- 115 Single Signon** | Single sign-on (SSO) is any technology that replaces multiple, independent system or application login prompts with a consolidated authentication (see 34 on Page 5) process, so that users don't have to repeatedly sign in.
- 116 Reduced Signon** | A synonym for single sign-on (see 115 on Page 18) which recognizes that authentication is normally reduced but often not to just one step.

10.1 Token Passing Approaches

- 117 Kerberos** | A technology, originally developed at MIT but over time also adopted by Microsoft and made available on Windows, Unix, database and mainframe platforms, which separates authentication from applications. A user signs into the Kerberos system and is issued a cryptographic ticket – containing assertions about the user's identity and security group memberships. The Kerberos software on the user's computer forwards this ticket to other applications which the user wishes to access, instead of requiring the user to sign into each application separately.
- 118 Security Assertion Markup Language (SAML)** | An XML-based protocol whereby one web service (the identity provider) may make assertions about the identity or rights of a user (the principal) to another web service (the service provider).
SAML allows for single sign-on between domains, in cases where cookies, for example, cannot be used (web browsers in general only allow cookies to be submitted to the same domain that issued them).
In practical terms, users authenticate to the identity provider. When users attempt to access content or applications on the service provider, their web browser is directed to request SAML assertions from the identity provider and pass those back to the service provider. In this way, the service provider no longer has to authenticate the user directly, and instead relies on statements about the user made by the identity provider, which does authenticate the user.

10.2 Enterprise Single Signon

119 Enterprise Single Signon

A technology which reduces the number of times that a user must sign into systems and applications by automatically populating login ID and password fields when applications ask for user authentication. This is done by monitoring what is displayed on a user's desktop and - when appropriate - typing keystrokes on behalf of the user. In short – “screen scraping” the user's desktop.

In short, applications are unmodified and continue to perform user authentication. Reduced sign-on is achieved by auto-populating rather than removing login prompts.

120 Credential Database

Most enterprise SSO (see 119 on Page 19) systems work by storing the various login IDs and passwords for a user in a database of some form and retrieving this information when the time comes to auto-populate a login prompt. This database should be protected, as it contains sensitive information. It may be physically local to the user's workstation, or stored in a directory (see 107 on Page 16), or in an enterprise relational database (ERDB). The credential database should definitely be encrypted.

10.3 Web Single Signon

121 Web Access Management

A web access management (also web single sign-on or WebSSO) system authenticates users as they access one or more web applications and may limit what URLs, application features or data users may access. This is normally accomplished by diverting user web browsers from “native” application login pages to the WebSSO authentication page and then diverting users back to application pages, using a cookie installed in the web browser to track user identity, authentication state and assigned entitlements.

One of the advantages of WebSSO is that multiple, separate login pages are replaced by a single, shared authentication process, so the frequency of user logins is reduced.

122 Web Proxy

A web proxy acts on behalf of one or more web browsers, fetching web pages for users and possibly adding capabilities such as caching (to reduce an organization's bandwidth usage), filtering (to block unwanted content) and monitoring (to record user activity).

Web proxies act on behalf of one or more users.

123 Reverse Web Proxy

A reverse web proxy intercepts user attempts to access one or more web applications, may modify the HTTP or HTTPS requests (for instance, inserting credentials), and requests web pages on behalf of the user.

Reverse web proxies act on behalf of one or more web servers.

WebSSO systems (see 121 on Page 19) may be implemented using a reverse web proxy architecture, which insert user application credentials into each HTTP stream.

The reverse web proxy architecture has the advantage of not requiring software to be installed on each web application – attractive when a WebSSO system is integrated with a large number of web applications.

124 Web Server Agent

An agent installed on a web server may be used to implement a WebSSO system (see 121 on Page 19) by injecting user identification, authentication and authorization data into the requests sent from a user's browser to the web server. more web applications, may modify the HTTP or HTTPS requests (for instance, inserting credentials), and requests web pages on behalf of the user.

The server agent architecture has the advantage of not requiring new hardware to be deployed when implementing a WebSSO system.

125 WebSSO Authentication Server

In a WebSSO system, one or more servers are dedicated to the function of authenticating users and determining what operations they will be permitted to perform. These are called authentication servers.

10.4 Federation**126 Federation**

Federation is both a technology and a business relationship. The business relationship is one where one organization (A) trusts a partner (B) to authenticate and authorize users who will subsequently be allowed to access A's resources (typically web applications) without having user records on A's network.

This technology depends on a business relationship with implicit trust of B by A.

127 Trust Relationship

A trust relationship is a codified arrangement between two domains where users and/or services exist. One domain (A) trusts the other (B) to identify, authenticate and authorize B's users to access A's resources.

Simple trust relationships are two-way, while complex ones may have groups of multi-way trust (i.e., any organization in a group trusts any other to make assertions about its own users).

11 Password Management

11.1 Password Policy

- | | |
|--|--|
| 128 Password Policy | <p>A password policy is a set of rules regarding what sequence of characters constitutes an acceptable password. Acceptable passwords are generally those that would be too difficult for another user or an automated program to guess (thereby defeating the password mechanism).</p> <p>Password policies may require a minimum length, a mixture of different types of characters (lowercase, uppercase, digits, punctuation marks, etc.), avoidance of dictionary words or passwords based on the user's name, etc.</p> <p>Password policies may also require that users not reuse old passwords (see 131 on Page 21) and that users change their passwords regularly (see 132 on Page 21).</p> |
| 129 Complexity Rules | <p>Password complexity rules are those parts of a password policy designed to ensure that users choose hard-to-guess passwords. Examples are requirements to use long passwords, to use mixed case or to avoid dictionary words.</p> |
| 130 Password Representation Constraints | <p>Most systems have limits regarding what can be stored in the password field. Limits generally break down into two types – which characters may be incorporated into a password, and how long a password can be.</p> |
| 131 Password History | <p>A password history is some representation of one or more previously used passwords for a given user. These passwords are stored in order that they may be compared to new passwords chosen by the user, to prevent the user from reusing old passwords.</p> <p>The reason for password history is the notion that, given enough time, an attacker could guess a given password. To avoid this, passwords should be changed periodically and not reused.</p> |
| 132 Password Expiry | <p>Password expiry is a process whereby users are forced to periodically change their passwords. An expiration policy may be represented as the longest number of days for which a user may use the same password value.</p> <p>The reason for password expiry is the notion that, given enough time, an attacker could guess a given password. To avoid this, passwords should be changed periodically and not reused.</p> |
| 133 Password Age | <p>Password age is the number of days since a password was last changed.</p> |

11.2 Password Synchronization

- 134 **Password Synchronization** | Password synchronization technology helps users to maintain the same password on two or more systems. This, in turn, makes it easier for users to remember their passwords, reducing the need to write down passwords or to call an IT help desk to request a new password, to replace a forgotten one.
- 135 **Global Password Policy** | A global password policy is a policy (see [128 on Page 21](#)) designed to combine the policies of multiple target systems. It is the product of combining the strongest of each type of complexity rule (see [129 on Page 21](#)) and the most limited representation capabilities (see [130 on Page 21](#)) of the systems where passwords will be synchronized.
- 136 **Web-based Password Synchronization** | Web-based password synchronization works by having a user sign into a consolidated web page to change multiple passwords, rather than waiting for each system or application to prompt the user to change just one password.
- Users typically sign into the password synchronization web page using a primary login ID and password and can then specify a new password, which will be applied to multiple systems and applications.
- A password synchronization web application typically must enforce a password policy (see [128 on Page 21](#)), which should be at least as strong as the policies in each of the target applications (see [151 on Page 26](#)).
- 137 **Transparent Password Synchronization** | Transparent password synchronization works by intercepting native password changes on an existing system or application and automatically forwarding the user's chosen new password to other systems. It is called transparent since the user is not presented with any new user interface.
- Transparent password synchronization typically must enforce a multi-system password policy (see [128 on Page 21](#)) in addition to the native policy of the system where synchronization is initiated. This policy should be at least as strong as the policies in each of the target applications (see [151 on Page 26](#)).
- 138 **Automatic Password Synchronization** | Automatic password synchronization is a synonym for transparent password synchronization (see [137 on Page 22](#)).
- 139 **Password Synchronization Trigger** | A password synchronization trigger is the component of a transparent password synchronization system (see [137 on Page 22](#)) which detects the initial password change event and starts the synchronization process.

11.3 Self-Service Password Reset

- 140 **Password Change** | A routine password change is a process where a user authenticates to a system using his login ID and password, and chooses a new password – either voluntarily or because the old password has expired (see 132 on Page 21).
The only credentials involved in a routine password change are the user's identifier, old password and new password.
- 141 **Password Reset** | A password reset is a process where a user who has either forgotten his own password or triggered an intruder lockout (see 43 on Page 6) on his own account can authenticate with something other than his password and have a new password administratively set on his account.
Password resets may be performed by a support analyst (see 2 on Page 1) or by the user himself (self-service).
- 142 **Self-Service Password Reset** | Self-service password reset (SSPR) is a self-service password reset process (see 141 on Page 23). Users normally authenticate using challenge/response (see 49 on Page 7), a hardware token (see 51 on Page 7) or a biometric (see 53 on Page 8).
SSPR is normally deployed to reduce IT support cost, by diverting the resolution of password problems away from the (expensive, human) help desk.

11.4 Password Wallets

- 143 **Password Wallet** | A password wallet is an application used by a single user to store that user's various passwords, typically in encrypted form.

11.5 Password Recovery

- 144 **Password Recovery** | Many applications offer weak encryption of data, such as office documents or spreadsheets. Such encryption is susceptible to brute force to key recovery, and such key recovery is offered by password recovery applications, most often offered to users who forgot the passwords they used to protect their own documents.

12 User Provisioning

145 User Provisioning

A user provisioning system is shared IT infrastructure which is used to externalize the management of users, identity attributes and entitlements from individual systems and applications.

User provisioning is intended to make the creation, management and deactivation of login accounts and other user objects, which are spread across multiple systems, faster, cheaper and more reliable. This is done by automating and codifying business processes such as onboarding and termination and connecting these processes to multiple systems.

User provisioning systems work by automating one or more processes:

- **Identity synchronization:**
Detect changes to personal data, such as phone numbers or department codes, on one system and automatically make matching changes on other systems for the same user.
- **Auto-provisioning:**
Detect new user records on a system of record (such as HR) and automatically provision those users with appropriate access on other systems and applications.
- **Auto-deactivation:**
Detect deleted or deactivated users on an authoritative system and automatically deactivate those users on all other systems and applications.
- **Self-service requests:**
Enable users to update their own profiles (e.g., new home phone number) and to request new entitlements (e.g., access to an application or share).
- **Delegated administration:**
Enable managers, application owners and other stake-holders to modify users and entitlements within their scope of authority.
- **Authorization workflow:**
Validate all proposed changes, regardless of their origin and invite business stake-holders to approve them before they are applied to integrated systems and applications.
- **Consolidated reporting:**
Provide data about what users have what entitlements, what accounts are dormant or orphaned, change history, etc. across multiple systems and applications.

As well, a user provisioning system must be able to connect these processes to systems and applications, using connectors that can:

- List existing accounts and groups.
- Create new and delete existing accounts.
- Read and write identity attributes associated with a user object.
- Read and set flags, such as “account enabled/disabled,” “account locked,” and “intruder lockout.”
- Change the login ID of an existing account (rename user).
- Read a user’s group memberships.

12.1 Automated Provisioning

146 Automated Provisioning

Automated provisioning systems typically operate on a data feed from a system of record, such as a human relations (HR) system and automatically create login IDs and related logical access rights for newly hired employees or contractors.

It should be noted that automated provisioning normally operates without a user interface – i.e., data flows in from one system and out to one or more other systems, without any further user input in between.

Auto-provisioning reduces IT support costs and can shorten the time required to provision new users with requisite access rights.

147 Automated Termination

Automated termination systems typically operate on a data feed from a system of record, such as a human relations (HR) system and automatically disable access rights for existing users when they have left an organization.

It should be noted that automated termination normally operates without a user interface – i.e., data flows in from one system and out to one or more other systems, without any further user input in between.

Auto-termination reduces IT support costs and can make access deactivation both faster and more reliable than manual processes.

148 Identity Synchronization

Identity synchronization systems map identity attributes (see 26 on Page 4) between different systems and automatically propagate changes from one system to another.

It should be noted that identity synchronization normally operates without a user interface – i.e., data flows in from one system and out to one or more other systems, without any further user input in between.

For example, an e-mail system may be authoritative for each user's SMTP e-mail address, an HR system for the same users' employee number and department code, a white pages application for each user's phone number and so on. An identity synchronization system makes sure that all of these systems have correct and up-to-date information in each of these fields.

12.2 Consolidated and Delegated Administration

149 Consolidated Administration

A consolidated administration system allows a security administrator to create, modify or delete user records on multiple systems at once. It acts as a more efficient replacement for the native user management tools in each of the systems with which it has been integrated.

- 150 Delegated Administration** | A delegated administration system allows a some users to manage the accounts (see 24 on Page 4) of other users on some systems (see 151 on Page 26). Delegated administration is intended to move user management out of a central IT function, decentralizing it so that it is performed by IT or business users who are more closely familiar with the users whose profiles (see 27 on Page 4) are being managed.
- Delegated user administration may be thought of as consolidated user administration plus filters that limit what one user can see of and do to another.

12.3 Target Systems

- 151 Target System** | Systems and applications where information about users resides and which are integrated into an identity management infrastructure are called target systems. They may include directories, operating systems, databases, application programs, mainframes, e-mail systems, etc.
- 152 Target Connector** | A connector is a piece of software used to integrate an identity management system with a given type (see 154 on Page 26) of target system (see 151 on Page 26).
- 153 Agent** | An agent is another term for a target connector.
- 154 Target Platform** | A target platform is a type of target system (see 151 on Page 26). For example, it might be an operating system (e.g., Unix, Windows), a type of database (e.g., Oracle, Microsoft SQL) or a type of application (e.g., SAP R/3, PeopleSoft). An identity management system typically needs a different connector for each type of integrated target platform.
- 155 Local Agent** | A local agent is an agent installed on the target system itself.
- Installation of local agents requires change control on the target system itself – something which may be difficult and/or undesirable on a production system or application.
- Local agents are well positioned to detect changes to user objects on a target system in real time, forwarding these changes to an identity management system which may act on them.
- Communication between an identity management system and a local agent can always be protected, even if the native communication protocols of the target system are insecure.

- 156 Remote Agent** | A remote agent is an agent installed on an identity management server, rather than on the target system (see 151 on Page 26).
- Installation of remote agents requires no change control on the target system itself, making them easier to deploy and possibly more scalable, when hundreds or thousands of target systems are involved.
- Local agents normally cannot detect changes to user objects on a target system in real time, so must poll target systems for changes periodically.
- Communication between an identity management system and a local agent may not be secure, since it relies on the native communication protocols of the target system, which in some cases may be vulnerable to eavesdropping or data injection.
- 157 Coarse-Grained User Provisioning** | Coarse grained user provisioning is a process where new accounts are created for new users, with basic entitlements rather than all of the required entitlements.
- This may be easier to automate and faster to deploy, but requires further, manual intervention before a new user can be fully productive.
- 158 Fine-Grained User Provisioning** | Fine-grained user provisioning is a process where new accounts are created for new users, with all of the entitlements that a new user will require – identity attributes (see 26 on Page 4), group memberships (see 76 on Page 11) and other objects, such as home directories and mail folders, already created.
- This may be more complex to automate and longer to deploy, but eliminates further, manual intervention before a new user can be fully productive.

13 Privileged Password Management

- 159 Privileged Account** | A privileged account is a login ID on a system or application which has more privileges than a normal user. Privileged accounts are normally used by system administrators to manage the system, or to run services on that system, or by one application to connect to another.
- 160 Shared Account** | A shared account is a login ID on a system or application that is used by more than one human or machine user. Privileged accounts (see 159 on Page 27) are often shared: for example, `root`, `sa` or `Administrator` by system administrators.

13.1 Sensitive Passwords

- | | |
|--|---|
| 161 Local Administrator Password | A local administrator password is the password to an account used by system administrators to install, configure and manage a system or application. Examples are <code>Administrator</code> on Windows, <code>root</code> on Unix/Linux and <code>sa</code> on Microsoft SQL Server. |
| 162 Service Account Password | A service account password is used on Windows systems to start a service program which runs in a context other than that of the SYSTEM user. The service control manager uses a login ID and password (of the service account) to start the service program. |
| 163 Embedded Application Password | An embedded application password is a password stored in one application and used to connect to another. A common example is a database (ID and password stored on a web application and used to connect to the database, to fetch and update database records. |

13.2 Password Locations

- | | |
|----------------------------------|---|
| 164 Security Database | A security database is the native storage used by a system or application to house records about users, passwords and privileges. Examples are the SAM database in Windows, <code>passwd</code> file on Unix/Linux and RACF on mainframes. |
| 165 Server Passwords | Server passwords are passwords stored in the security database on a network server. Servers typically have fixed addresses, are (almost) always turned on and respond to requests they receive on the network. |
| 166 Workstation Passwords | Workstation passwords are passwords stored in the security database on a user's workstation (PC or laptop). Workstations typically have dynamic addresses, are sometimes turned off and do not respond to requests they receive from the network. |
| 167 Mobile Passwords | Mobile passwords are passwords stored in the security database on a portable device, such as a PDA or smart phone. Mobile devices typically have dynamic addresses, are sometimes turned off and may not respond to requests they receive from the network, other than special cases such as phone calls and text messages. |

13.3 Password Disclosure

- | | |
|---------------------------------------|--|
| 168 Password Disclosure | <p>Password disclosure is a process where a stored copy of a password, matching a password in a target system's security database, is revealed to a human or machine user. For example, it might be the process of revealing a stored copy of an administrator password to a system administrator.</p> |
| 169 Human Password Disclosure | <p>This is password disclosure to a human being - for example using on a web page.</p> |
| 170 Checkin / Checkout | <p>Password disclosure may be limited, in the sense that a password is regularly changed, and only a limited number of users are allowed to have access to the current password value at any given time.</p> <p>For example, only a single person might be granted administrative privileges (via disclosure of an administrator password) to a given system at once.</p> <p>A checkin/checkout process is one where a user "checks out" a password, much like a library book, and "checks it back in" when finished. The password may be changed at checkin time.</p> |
| 171 Multi-Key Password Release | <p>Password disclosure may require authorization. For example, system administrator A may need a password, but might not be allowed to see the password until other people – say B and C, approve the disclosure. Multi-key password disclosure refers to any process where the actions of more than one person are required to disclose a password.</p> |

14 User Interfaces

- | | |
|---|--|
| 172 Graphical Identification and Authentication (GINA) | <p>The Graphical Identification and Authentication (GINA) is a subsystem on Windows 2000 and XP computers which handles user authentication at the login screen and screen saver and which intercepts the Ctrl-Alt-Delete key sequence.</p> <p>When users forget their Windows password, some mechanism is required to present them a user interface despite the fact that they cannot get by the GINA and access their desktop.</p> |
| 173 Vista Credential Provider | <p>On Vista workstations, a credential provider infrastructure replaces the GINA infrastructure from previous versions of Windows. A credential provider may be installed to provide the same functionality as a GINA extension (see 175 on Page 30).</p> |

174 **Secure Kiosk Account (SKA)**

A secure kiosk account is a special Windows login ID and password, which is well known to users (for example, it may be advertised on the wallpaper image of the login screen). Special security policies are applied to this account, so that when it signs into a Windows workstation, a locked down (kiosk-mode) web browser is launched instead of the normal Windows desktop.

A SKA is a mechanism that allows users to access a self-service password reset web application (see 142 on Page 23) despite being locked out of the initial workstation login screen.

175 **GINA Extension**

A GINA extension is software installed on a Windows computer that adds a user interface element to the normal GINA screen. This user interface activates a self-service password reset (see 142 on Page 23) screen, enabling users who are locked out of the Windows login screen to resolve their own problem.