Tivoli® software

Redpaper

Axel Buecker
Dwijen Bhatt
Daniel Craun
Dr. Jayashree Ramanathan
Neil Readshaw
Govindaraj Sampathkumar

Integrated Identity and Access Management Architectural Patterns

Customers implement an integrated identity and access management (IAM) solution to address many business requirements. The overall driving requirement is to provide a combination of business processes and technologies, to manage and secure access to information and resources within the organization.

To achieve this goal, the IAM solution:

1. Provides a method of granting users access to applications and systems that are needed to perform their jobs.
2. Has the capability to authorize proper access levels to resources based on business policies.
3. Enables Web accessed resources, provides a way to authenticate people, and require a single sign-on (SSO) to access resources, once access is granted.
4. Has an audit trail to ensure proper operation of the IAM system.

In this Redpaper, we describe several common business use cases for an integrated IAM solution. We integrate IBM® Tivoli® Identity Manager, and IBM Tivoli Access Manager in a typical deployment to address these business use cases.

Introduction

A business organization must have efficient automated processes so that employees:

- Can quickly obtain access to the correct application
- Can terminate access when an employee leaves

IAM handles this critical security aspect of information technology (IT). In this Redpaper, we describe several common business use cases for IAM. We also describe typical architectural patterns for implementing the use cases, using Identity Manager and Access Manager.

In this section, we begin by defining certain terms used to describe an integrated IAM solution.

User identity Controls information used to describe a specific user in an *enterprise*. It is similar in format to human resource records, but does not possess confidential information (such as, social security number, pay grade or age) that is not pertinent, or appropriate when determining access to resources. The user identity, by itself, does not provide access to any resources.

Identity management Manages the user identity life cycle within an organization. It creates, or establishes the user identity, user identity operations, and finally the destruction of the user identity within the organization. Identity management controls these operations using an approval process that is required by the business.

User account Is the user information that provides access to a specific resource in the enterprise through a successful login operation. An authentication mechanism uses this user info as part of the login process. The user account information is stored in a user registry.

Provisioning Is the ability to manage all user account operations (add, modify, suspend, delete, password management) on individual user registries. Normally, a single user requires access to multiple resources in order to satisfy their job function, therefore requiring multiple user accounts. A users local access level, after successful login, is usually determined by a combination of user name, and user group membership, both are usually defined in the user account.

Access management Manages access control for various resources (systems and applications) within the enterprise. Access management also includes user account management.

Naturally, there is a complementing relationship between identity and access management (for example, an employee (user identity) has a job function that requires access to certain resources. This access is granted using a combination of linked accounts (user accounts) and access controls on those resources). An integrated IAM solution provides a combination of business processes and software technologies to manage and secure access to proprietary information within an enterprise. Not only is this good business practice, but several industry sectors are regulated through security compliance requirements that are directly aimed at their IAM solution. Security compliance requirements also play a big role in certain customers looking to implement an integrated IAM solution in their environment. For example, to meet compliance requirements, customers want to ensure that identities are removed for users who no longer have a business need for these identities. Customers also want to restrict access to resources given to a user based on business needs.

There are several common specific business requirements that organizations can address through an IAM solution:

- Provide a method of provisioning and de-provisioning of user accounts across the organization, using the approval processes established by the business. These requirements form the core of identity lifecycle management use cases.
- Provide the capability to authorize proper access levels to resources based on business policies. These capabilities form the core of access management use cases.
- For Web accessed resources, such as Web applications, the solution must provide a way to authenticate people, and only require a SSO to access granted resources. These requirements improve the user experience for people accessing various applications through the normal course of their daily activities within the company.
- Finally, there must be an audit trail to:
  - Ensure that proper identity management enforcement is in place
  - Ensure that proper access control enforcement is in place
  - Verify compliance with business policies

These requirements address the use cases around various compliance initiatives. Overall, an IAM deployment addresses the security requirements of a company through addressing the various human aspects of information security. This Redpaper describes several common business use cases, and then describes how Identity Manager and Access Manager integrate to address these business use cases.

This Redpaper is organized as follows:

- "Business context and requirements", on page 4
  - Describes the business context for an IAM solution
- "Common personas and use cases for business scenarios", on page 7
  - Describes several common IAM business use cases
- "Common architectural patterns", on page 13
  - Describes the architectural, and realization patterns of how Identity Manager and Access Manager integrate in a typical deployment to address these business use cases.
- In "Deployment considerations", on page 29
  - Considers the physical deployment architecture of Identity Manager and Access Manager, with specific focus on performance and high availability factors.
- "Guidelines and recommendations", on page 38
  - Provides guidelines and recommendations based on experiences collected by authors across a number of projects

Business context and requirements

In this section, we discuss business scenarios that have identity management and access management requirements. We also outline the key *personas* for each of these business scenarios. In this section, and throughout this publication:

IAM ecosystem Is a term used to include the different logical elements that constitute an IAM system.

Persona Is a role that a person plays in the context of a scenario. In a later section, we discuss in detail, use cases for each persona involved in each business scenarios.

Figure 1 on page 4 shows the company portal business scenario. In this scenario, the company has enterprise applications that internal employees must use. These enterprise applications use WebSphere® Portal[1] technology for their user interface, a lightweight directory access protocol (LDAP)[2,3] repository for storing users, and a relational database for storing their data.

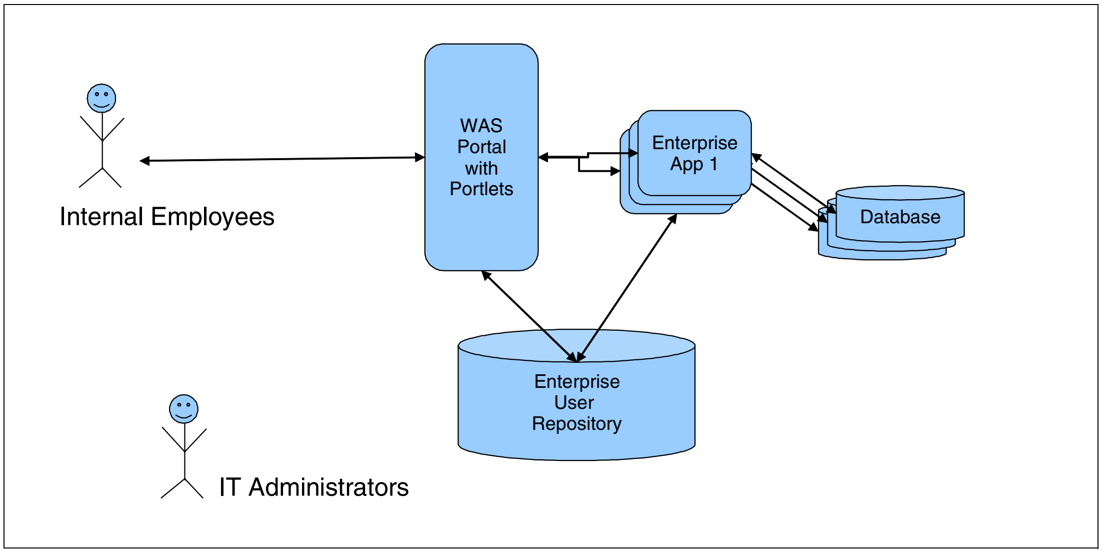


Figure 1 Business scenario - company portal

The applications are administered by IT administrators, and owned by application owners. The employee manager decides the job responsibility of the employee, and what applications they need access to perform their job. So, the key personas in this scenario are:

- Internal employee
- IT administrator
- Application owner
- Employee manager

Figure 2 on page 5 shows the external customer portal business scenario. In this scenario, the company has customer facing applications that internal employees must use. These customer facing applications use the WebSphere Portal technology for their user interface, an LDAP repository for storing users, and a relational database for storing their data.

[1] WebSphere Portal Resources: http://www.ibm.com/developerworks/websphere/zones/portal/proddoc.html
[2] Lightweight Directory Access Protocol: http://en.wikipedia.org/wiki/Lightweight_Directory_Access_Protocol
[3] IBM Tivoli Directory Server: http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?toc=/com.ibm.IBMDS.doc/toc.xml

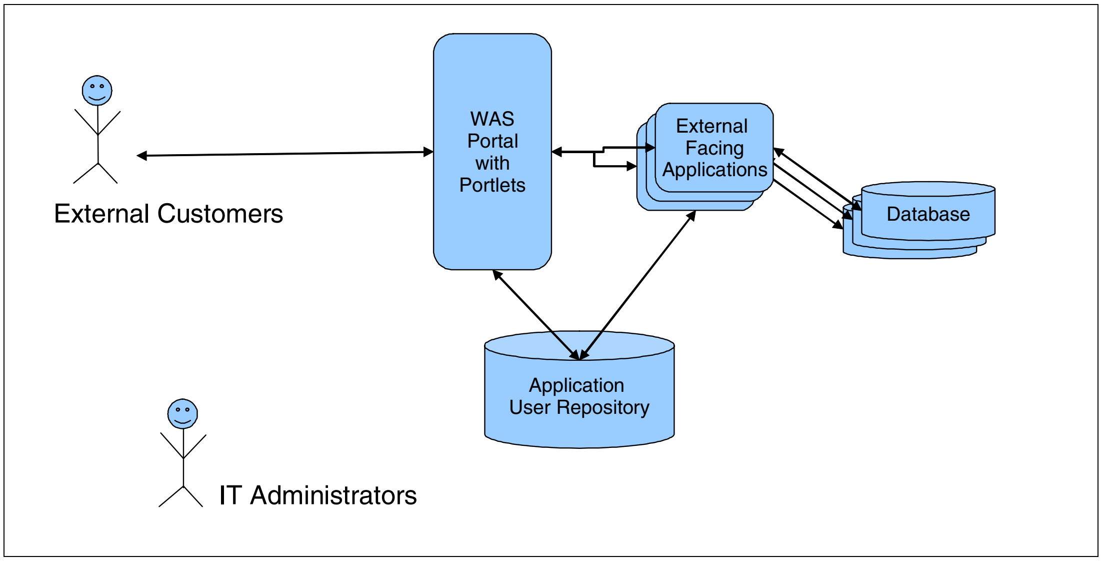


Figure 2 Business scenario - external customer portal

These applications are administered by IT administrators, and owned by application owners. The key personas in this scenario are:

- Internal employee
- IT administrator
- Application owner

Figure 3 on page 5 shows an IBM Service Management (ISM) scenario. It consists of multiple IBM Tivoli products used to automate management of IT environments.

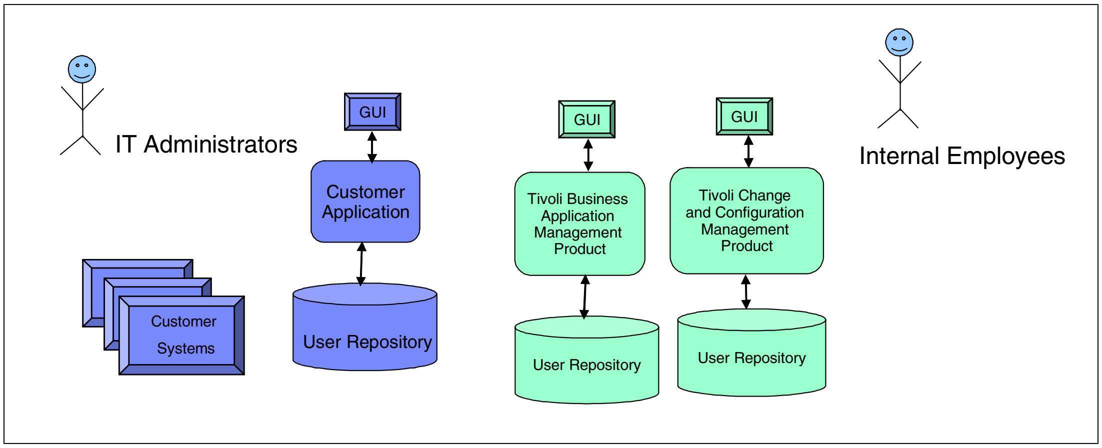


Figure 3 Business scenario - IBM Service Management architecture

For example, a company can use the Tivoli Change and Configuration Management product for managing changes (for example, software patches applied to their IT systems) in their IT environment, and the Tivoli Business Application Management products to ensure the availability and performance of their business critical applications. Each of these products can have their own user repository. In addition, customers have their own hardware systems, and custom applications that are owned by application owners. The internal employees need access to systems, and applications. IT administrators must have accounts on the Tivoli products.

In this scenario, the key personas are:

- Internal employee
- IT administrator
- Application owner
- Employee manager

In the above business scenarios, there is a need to have business processes to manage user accounts for the various personas, and to manage accesses to systems and resources needed by these personas. Organizations typically look to improve these business processes in several ways with the deployment of an IAM solution. The business objectives of deploying an IAM solution include cost reduction, improving employee productivity and efficiency, and reducing security risks:

1. Reduction of total cost of ownership (TCO) of their IT infrastructure by reducing administration, help desk, and technical support costs:

   a. Self-registration to get user accounts, and self-care of account related tasks, such as password resets to reduce help desk costs

   b. Sharing access and identity management infrastructure across multiple applications, to reduce administration, and technical support costs

2. Reduction of management overhead related to identity and access management by centralization:

   a. Lower overhead costs by automatically managing accounts, credentials, and access rights throughout the user life cycle

   b. Improve speed, accuracy, and cost structure for hire, and termination processes, by reducing the time it takes to give new employees access to required resources within the organization

3. Reduction of risk of incorrect information usage:

   a. Mitigation of concerns about misuse of personal and confidential information

   b. Mitigation of liability for not protecting personal information

4. Reduces the risk of an ex-employee retaining access to organizational resources:

5. Improve productivity by allowing users to perform self-service:

   a. Rapidly reset and manage their own passwords

   b. Decrease help desk calls by providing Web self-care interfaces to perform password, and personal information changes

6. Centralized control and local autonomy, ensures security, and consistent policy on the most sensitive systems:
    a. Provides support for legal and compliance initiatives for employee, and customer data
    b. Produce centralized reports on security policy, access rights, and audit events to quickly respond to internal audits and regulatory mandates
    c. Centralize policy based access control and audit trails across key information systems
    d. Obtain intelligence and recommended actions on compliance issues, eliminating the need to manually review policies

Common personas and use cases for business scenarios

Identity and Access Management systems touch many aspects of a business. People in many roles interact with them, directly and indirectly, during the various phases of deployment and operations. In this section, we list the various personas and their goals with respect to an IAM solution. We also list the common business use cases for an IAM solution.

Personas and goals

The following *personas* interact with an IAM solution.

Security architect

The security architect helps establish and maintain long term security strategy for the enterprise. The security architect is the person responsible for ensuring that business requirements are met by the IAM solution:

- Use the technology to implement the business goals. For example, enhance the employee application computing environment (business to employee or b2e), facilitate integration with trade partners (business to business or b2b), and extend business directly to consumers using internet (business to consumers or b2c).
- Map business policies to associated security requirements, and define how those requirements are realized using security related technologies
- Integrate security with the current environment, and interoperate with key vendor products
- Make the security environment consistent, centralized, and consolidated through a well planned deployment strategy

Internal employee

An IAM system enables the internal employees of the business to have a consistent user experience from the SSO facilities provided by the solution. It also enables them to perform self-service within the bounds of the approval and compliance policies of the business. Internal employees have the following goals from the IAM system:

- Access to applications and accounts on systems needed to perform their job.
- Be able to access self-service password-related activities, such as expiration of password, change password, forgotten password, and so on

External customer and partner

Similar to the internal employees, external customers of a business benefit in much the same ways from an IAM system. Their goals from an IAM system are to:

- Access to the companys external facing applications
- Be able to access self-service password-related activities, such as expired password renewal, password change, forgotten password, and so on

IT administrator

IT administrators work with the security architects to implement centralized identity and access management that meet the access and identity requirements of business policies. Their goals from an IAM system are to:

- Specify access control to ensure only authorized users get access to applications
- Specify account management policies to ensure only authorized users get access to accounts on systems
- Ensure internal employees, and external customers can use self-service password-related activities
- Handle changes needed when a new employee joins, employee changes their last name, employee changes departments, and protecting new applications

Application owner

The application owner is responsible for approving requests for access to applications that they own. The application owners goals from an IAM system are to:

- Be compliant with all security audit reviews
- Promptly approve valid requests for accesses to applications that they own

Employee manager

The employee manager is responsible for approving requests pertaining to the Identity life cycle management events of the internal employee. The employee manager's goals from an IAM system are to:

- Keep his, or her department compliant with all security audit reviews
- Approve valid requests for access to applications, and systems from their employees

Business use cases

In this section we describe, at a high level, the common business use cases addressed by an IAM solution. These use cases arise from an organizations need to automate business processes around identity and access management. This section does not go into the details of how these use cases are realized in practice; see “Realization of business use cases”, on page 16 for how that is done using IBM Tivoli products.

Enrollment of a new employee

The provisioning process is automated such that when a new employee is entered onto the Human Resources management system, the employee's data triggers an automated process in which the required employee accounts and accesses are automatically created. The objective of this use case is to automate the processes that provide the employee access to systems, and applications they need to perform their job.

Personas involved

The following *personas* are involved in this use case:

Security architect Defines provisioning, and access policies that underlie enrollment automation.

IT administrator Implements the provisioning, and access policies created by the security architect.

Employee manager Approves the provisioning requests for the new employee.

Application owner Approves the application requests for the new employee.

Employee job role change

The solution enables user identity, and access provisioning activities to automatically initiate when a users job role changes. In cases where employees move to a different department and assume a new job role, access is automatically changed with the employee job reclassification.

Personas involved

The following personas are involved in this use case:

Employee manager Approves the employee job role change, and the resulting provisioning requests.

Application owner Approves the application requests (addition, as well as deletion) for the employee.

Employee name change

The solution enables provision activities to automatically initiate when a users name changes. For example, last name or family surname changes are common when employees in a company get married, or divorced. Further, the company typically has an identity policy or naming standard that requires user IDs to contain parts of the employee's name. For example, first letter of the first name concatenated with the last name. This use case requires the user ids to be changed on the endpoints in a way that preserves existing authorization privileges for the user on these endpoints.

Personas involved

The following personas are involved in this use case:

Employee manager Approves the employee name change and the resulting provisioning requests.

Internal employee Submits the name change request, and starts using the new user accounts provisioned.

Termination of an employee

The de-provisioning process is automated. As soon as an employee is removed from the human resources management system, all of the employees accounts are automatically deleted.

Personas involved

The following personas are involved in this use case:

Employee manager Approves the employee termination, and any resulting de-provisioning requests.

Application owner Approves the application requests (deletion) for the employee.

Consistent password management

Password life cycle events such as expiration, modify, and reset are consistently managed across the IAM ecosystem. Further, consistent password strength policies are enforced across all the systems that are part of the IAM ecosystem, regardless of whether employees use a common self-service interface, or the native system interfaces to perform password change operations. Password strength policies vary based on the resource being accessed, organization and role of the user. The password that is reset, or changed after appropriate validation is synchronized across systems (both forward and reverse).

Personas involved

The following *personas* are involved in this use case:

Security architect Defines the password expiration policies that are enforced.

IT administrator Configures the password expiration policies.

Internal employee Performs password change operations as required.

Role Based Access Control

True Role Based Access Control (RBAC) is defined by the deployment of an automated system that takes a new hire entry in the HR system, and automatically provision correct access controls to all appropriate resources across the enterprise based on the user identity information, with no administrator action involved. While this is certainly the design goal of any IAM solution, the reality is that the real world granularity of access management prevents the affordable and timely deployment of true RBAC. However, auto-provisioning (RBAC for selected target applications/systems) is achievable in a reasonable time frame, and continued fine tuning of the IAM solution over time gets the solution closer and closer to true RBAC (the ultimate goal). The mechanics of accomplishing RBAC is through the definition and configuration of *organizational roles* in the IAM solution, such that certain user identity information is logically grouped into like access control levels (called entitlements) that result in provisioning user accounts on appropriate platforms, that together grant the correct access control required to do the persons job function.

Personas involved

The following *personas* are involved in this use case:

Security architect Defines the RBAC model that is adopted across the IAM ecosystem. This consists of defining the criteria for each role which determines the entitlements or privileges that members of that role receive.

IT administrator Configures the roles and the criteria defined by the security architect.

On-boarding and integrating a new application

As, and when new applications are rolled out within the enterprise, the IAM solution must make it easy to bring them into the IAM ecosystem. When the applications are part of the IAM ecosystem, provisioning, and enforcing access to the applications is governed using the same consistent IAM processes.

Personas involved

The following *personas* are involved in this use case:

Security architect Defines the architecture for how the application is on-boarded.

IT administrator Deploys the new application, and makes the required configuration changes to integrate the application into the IAM ecosystem.

Compliance audit and reporting

The IAM solution must audit creation, deletion, profile updates of user identities, and accounts. Further, the solution must audit both successful, and failed attempts to access resources for all types of users including administrators. Auditors require tracking of all access to customers non-public personal information. Not only is a record made of each access to a record, but also any data transfer, change, and deletion. Further, auditing and reporting addresses the following business use cases:

- External controls: showing compliance for various standards and legal requirements, such as Sarbanes-Oxley Act, Basel II, and HIPAA.
- Internal Controls: showing compliance to an organization's security policies.
- Checking enforcement and effectiveness of IT controls, for accountability, and vulnerability and risk analysis.
- Forensic investigations of security incidents (for example, who accessed my application yesterday at 2 AM?).

Personas involved

The following personas are involved in this use case:

Security architect In consultation with the enterprise compliance officer, defines the policies for capturing events that are relevant to audits.

IT administrator Configures the system to capture and manage audit data, and generate periodic reports for audit data.

Auditor Reviews the audit reports that are required for performing audits related to compliance to regulations to ensure audit data is being captured.

Self-management of user profile and access to resources

Through a single self-service interface, this solution provides for self-management of user profile information, and automatic replication of accurate profile data to key enterprise systems. The self-service interface provides the user with the ability to request, delete, approve, and modify access to different applications, and also manage passwords from a single console.

Personas involved

The following *personas* are involved in this use case:

IT administrator Deploys the self-service interface application.

Internal employee Uses the self-service interface to perform self-service operations, such as password change and reset.

Employee manager Approves valid requests submitted using the self-service interface.

Application owner Approves valid application requests (addition, as well as deletion) submitted using the self-service interface.

Forgotten password recovery and reset

Our solution enables users to recover from a forgotten password in a secure manner through various means, such as challenge and response questions.

Personas involved

The following *personas* are involved in this use case:

IT administrator Configures the challenge response administration.

Internal employee Sets up the challenge response questions, and uses the challenge response to recover forgotten passwords.

Single sign-on to Web applications

This solution provides SSO to all Web applications, and synchronization of user names, and passwords across the system. SSO is a capability of an IAM system that reduces the:

- Number of different credentials that a user possesses
- Number of times that a user must enter those credentials

The ultimate realization of SSO is defined as users only having a single credential, and only having to enter it once on any given day, any solution that brings an organization closer to this utopia delivers business value.

Benefits of SSO to the user include:

- Improved productivity, due to less time spent recalling and supplying authentication credentials
- Less reliance on the help desk when credentials are forgotten, because there are now fewer credentials to remember
- Greater chance that the user is able to protect their credentials, for example, less reliance on insecure storage of passwords
- A less frustrating, and distracting user experience, because you do not need to login multiple times

Benefits of SSO to the organization implementing the IAM solution include:

- Reduced calls to the help desk for password resets
- A common, consistent infrastructure for authentication that reduces the cost of ownership
- The ability to rapidly develop and deploy applications

Personas involved

The following *personas* are involved in this use case:

Security architect Defines the SSO architecture, including the protocols used for SSO, and the systems that participate in SSO.

IT administrator Configures the SSO environment.

Internal employee Benefits from the SSO experience in accessing various applications.

Common architectural patterns

In this section, we discuss high-level architecture for using IBM Tivoli Identity Manager (ITIM) and IBM Tivoli Access Manager (ITAM) products to address the identity, and access management for the various business scenarios.

Figure 4 on page 14 shows the logical architecture of an integrated identity, and access management system that uses the Identity Manager, Access Manager, and WebSphere Portal products. The user (is one of the *personas* discussed earlier) interacts with this system from their personal computer using a Web browser, and an e-mail client. They access various Web applications protected by the WebSEAL component of the Access Manager for e-business product. The user logs into these applications using their Access Manager user ID, and WebSEAL provides SSO to these applications. The details are described later in this section. Access Manager uses LDAP as the user repository. The LDAP server is made available by setting it up in a master, or replica configuration. Some of the applications protected by WebSEAL execute within the WebSphere Portal server.

The Access Manager Policy Server manages the access control policy, and replicates them to policy enforcement points, such as WebSEAL, and Access Manager for Operating System (ITAMOS) products. Access Manager for Operating Systems uses this policy to enforce authorization checks for access to files and directories on UNIX® systems.

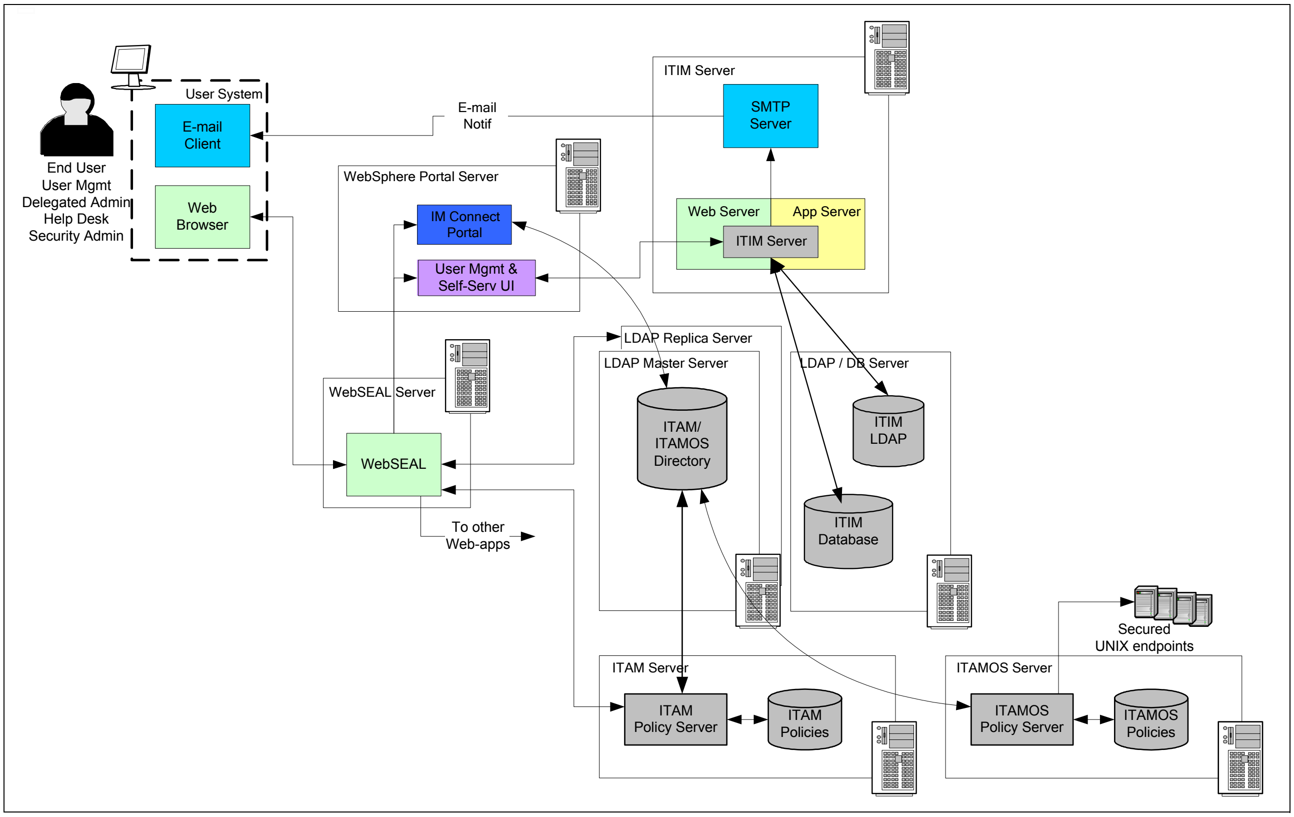


Figure 4 Identity Manager, Access Manager, and WebSphere Portal integration - logical architecture overview

The Identity Manager is used to manage the Access Manager accounts. Although not shown in Figure 4 Identity Manager is also used to manage accounts on other systems. Identity Manager generates e-mail notifications to the user using the simple mail transfer protocol (SMTP) server. Identity Manager has its own LDAP registry, and its own relational database.

Figure 5 on page 15 shows the physical architecture of an integrated identity and access management system that uses the Identity Manager, Access Manager, and WebSphere Portal products. Here, Web requests from external users come through a load balancer and gets routed to Access Manager WebSEAL, which is in the DMZ between two firewalls. The Access Manager, Identity Manager, WebSphere Portal, and LDAP systems are all in the intranet behind the second firewall.

The Access Manager Policy Server is configured in a master and standby configuration to provide high availability. The user logs in using their Access Manager user ID using WebSEAL, and the Trust Association Interceptor (TAI)[4] is used to create the WebSphere credential for the user, which is then used by the WebSphere Portal based application (more details on how this is done is discussed later in this section). Access Manager is also shown here to function as the authorization provider for WebSphere, using the Access Manager Authorization Server that serves as an authorization enforcement point.

[4] Tivoli Access Manager Trust Association Interceptor (TAI++), David Winters and Kerry Gunn, IBM DeveloperWorks: http://www.ibm.com/developerworks/tivoli/library/t-tamtai/

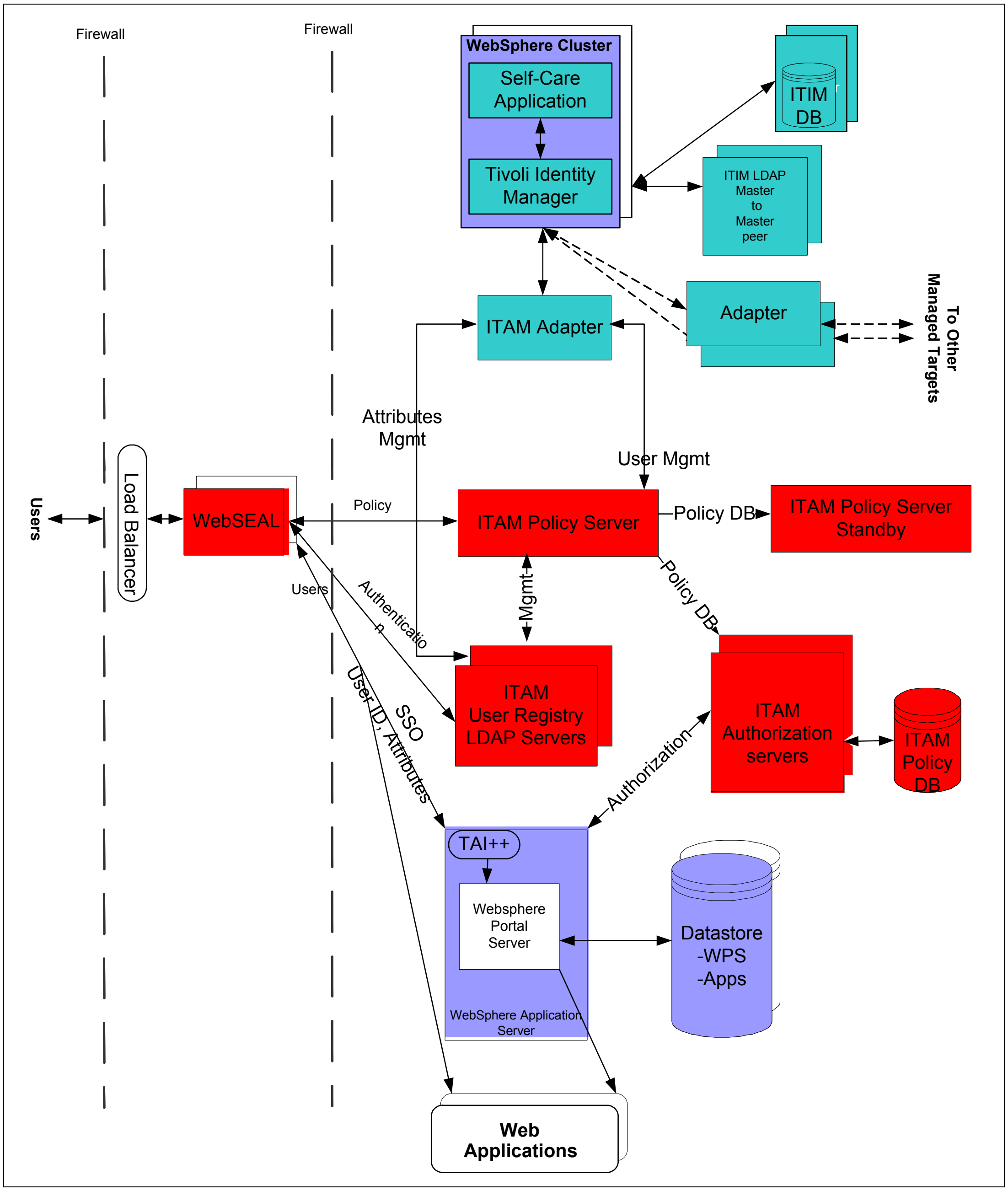


Figure 5   Identity Manager, Access Manager, and portal integration - physical architecture overview

Identity Manager is used to manage Access Manager accounts, as well as other targets using the Identity Manager adapters. Identity Manager is deployed in a WebSphere cluster for high availability. Identity Manager has its own LDAP, that is deployed in master to master peer configuration for high availability, and load balancing. The Identity Manager self-care application enables users to service, or manage their own accounts without requiring administrator assistance.

Realization of business use cases

This section details how IBM Tivoli products, Identity Manager and Access Manager integrate to implement each of the business use cases outlined in “Business use cases”, on page 8.

Enrollment of a new employee

In Figure 6 on page 16 we show a realization of the business use case pertaining to the enrollment of a new employee. The figure shows how the new employee's *user identity* is created within the IAM ecosystem as part of the employee's hiring process.

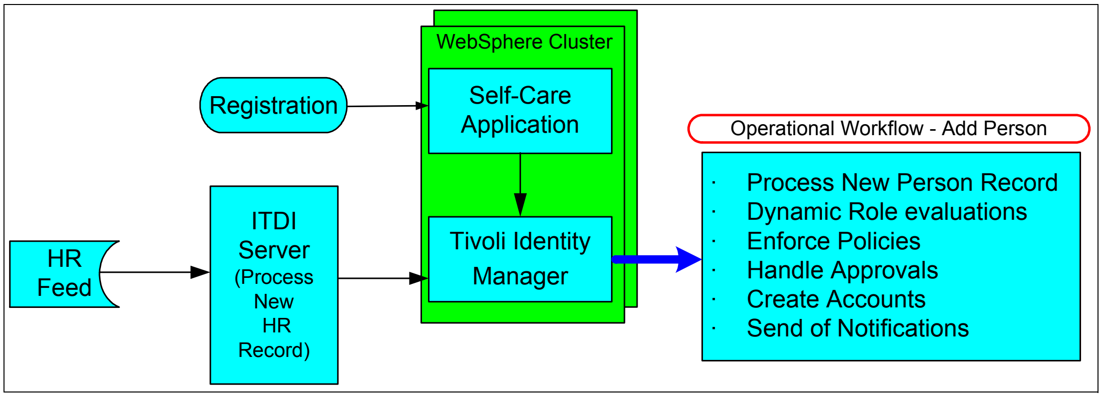


Figure 6 Enrollment of a new employee

HR Feed represents the data feed from human resource (HR) systems. This typically consists of HR related data, such as new employees, terminated employees, organization changes, and so on. This data is entered into the system in various forms, including in the form of flat files that are imported, or by programmatic connections to human resource management systems, such as through the IBM Tivoli Directory Integrator (ITDI Server)[5] as shown in Figure 6 on page 16. The new HR records are processed within the Identity Manager, and then taken through several operational workflows within Identity Manager, each of which is customized to suit the specific business requirements of each organization:

Process new person record Within the Identity Manager, each employees *user identity* is represented as a *person record*. The first step in the identity life cycle management is to create a *person record* to represent an user identity.

Dynamic role evaluations The newly created *person record* is dynamically evaluated for assignment to specific organizational roles based on configured criteria.

Enforce policies The newly created record is then evaluated against the various configured policies pertaining to the employees job role, organization, and so on. These policies include provisioning policies, identity policies, and so on.

Handle approvals The evaluation of the provisioning policies results in requests for provisioning user accounts on various applications for the new employee. These provisioning requests are routed through the appropriate approval steps, typically involving the new employees manager, and application owners for granting of approvals.

[5] IBM Tivoli Directory Integrator: http://www.ibm.com/software/tivoli/products/directory-integrator/

Create accounts When the approvals are granted, user accounts are created for the new employee on the various applications in conformance to the applicable identity policies. The completion of this process enables the employee to access various applications and systems needed to perform his or her job. One such account created is the account in Access Manager.

Send of notifications When the appropriate accounts are created, notifications are sent to the employee.

An alternative provisioning approach, is the employee self registration process, where an employee can access the self-care application directly to enroll into the system, instead of the employee record getting created through the HR feed process.

Employee job role change

The realization of this use case is shown in Figure 7 on page 17.

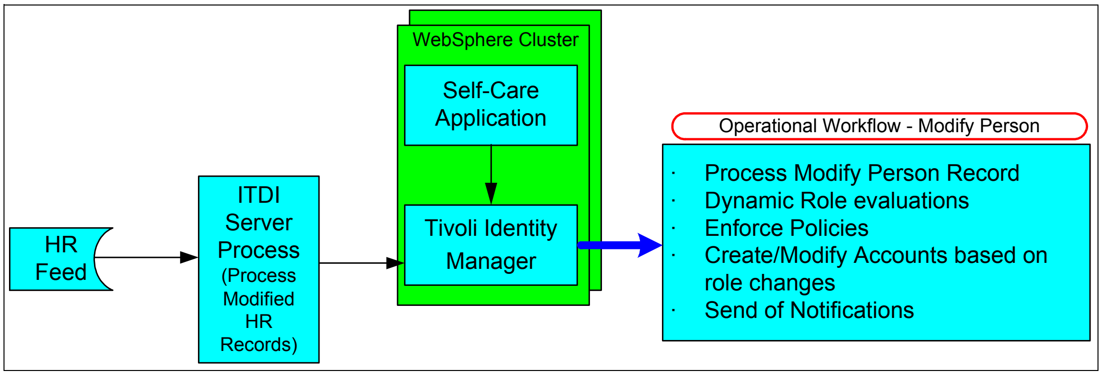

WebSphere Cluster
Self-Care Application
Tivoli Identity Manager
HR Feed
ITDI Server Process (Process Modified HR Records)
Operational Workflow - Modify Person
Process Modify Person Record
Dynamic Role evaluations
Enforce Policies
Create/Modify Accounts based on role changes
Send of Notifications

Figure 7 Employee job role change

The IAM system is notified of the employees job role change through the HR feed into the Tivoli Directory Integrator server, just as in the previous use case realization. The modified HR record is then taken through several customized operational workflows within the Tivoli Identity Manager:

Dynamic role evaluations The modified record is dynamically evaluated for assignment to specific organizational roles based on configured criteria.

Enforce policies The modified record is then evaluated against the various configured policies pertaining to the employees new job role, organization, and so on.

Handle approvals The evaluation of the provisioning policies can result in requests for provisioning new user accounts, modification of user accounts, or deletion of user accounts for the employee on various applications. These provisioning requests are routed through the appropriate approval steps, typically involving the employee's manager and application owners for granting of approvals.

Create or modify accounts

Based on the evaluations performed against policies and role changes, accounts are created or modified for the employee. This enables the employee to productively access various applications and systems needed to perform his or her new job.

Send of notifications

Once appropriate accounts are created, notifications are sent to the employee.

Employee name change

The user identities assigned to an employee must meet the company naming standards. Employee name changes result in the identity naming standards violations, and initiate processes to bring them into conformance. Figure 8 on page 18 shows this process.

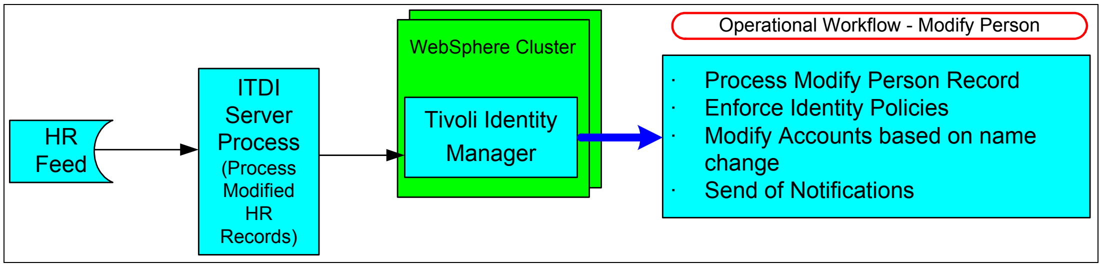

HR Feed
ITDI Server Process (Process Modified HR Records)
WebSphere Cluster
Tivoli Identity Manager
Operational Workflow - Modify Person
· Process Modify Person Record
· Enforce Identity Policies
· Modify Accounts based on name change
· Send of Notifications

Figure 8 Employee name change

The IAM system is notified of the employees name change through the HR feed into the Tivoli Directory Integrator server. The modified HR record is then taken through several customized operational workflows within the Tivoli Identity Manager:

Enforce identity policies

The modified record is evaluated against the configured identity policy based on the employees new name.

Modify accounts

Based on the evaluations performed against applicable identity policy, the employees user accounts are modified to conform to the identity policy. The modification is performed in a way that preserves the users existing access levels on each application.

Send of notifications

Once the user accounts are modified, the employee is notified of the resulting change in the user name for the accounts.

Termination of an employee

The realization for this use case is shown in Figure 9 on page 19. The intent of the realization is to reverse many of the processes of identity creation so that the terminated employee can no longer access applications that they were previously authorized to access.

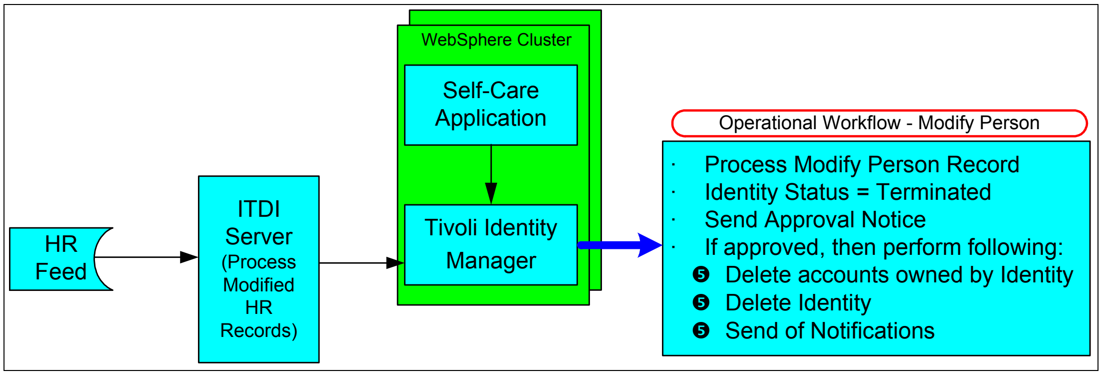

WebSphere Cluster
Self-Care Application
Tivoli Identity Manager
HR Feed
ITDI Server (Process Modified HR Records)
Operational Workflow - Modify Person
Process Modify Person Record
Identity Status = Terminated
Send Approval Notice
If approved, then perform following:
Delete accounts owned by Identity
Delete Identity
Send of Notifications

Figure 9 Termination of an employee

The IAM system is notified of the employees termination through the HR feed into the Tivoli Directory Integrator server. The modified HR record is then taken through several customized operational workflows within the Identity Manager:

Identity status The record is modified to set the status to *terminated*.

Send approval notice The required approval notices are sent to relevant parties for deleting accounts, and modifying the privileges. This is typically the employees manager, and owners of applications to which the employee had access.

Delete/Suspend If the approvals are granted, the accounts belonging to the terminated employee are either deleted, or suspended. This ensures that the terminated employee can no longer access the accounts he or she was entitled to as an employee of the company.

Consistent password management

The business motivation for this use case is both ease of user experience, as well as enforcement of consistent password strength policies to ensure better security. The details of the realization of this use case are shown in Figure 10 on page 19.

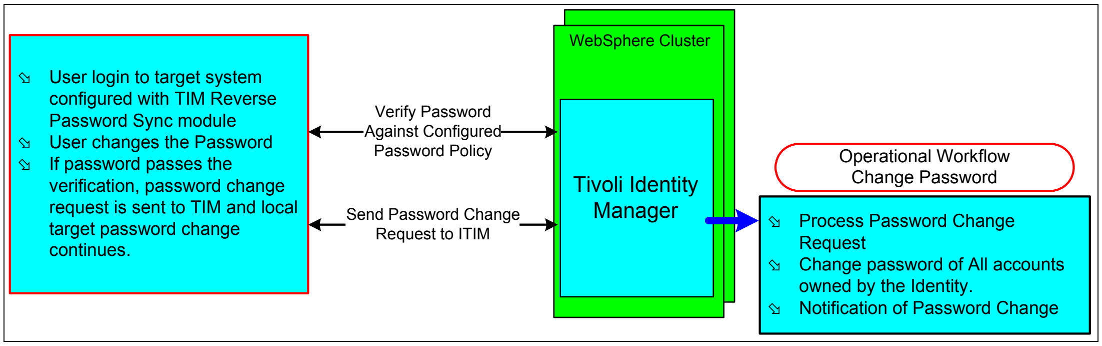

User login to target system configured with TIM Reverse Password Sync module
User changes the Password
If password passes the verification, password change request is sent to TIM and local target password change continues.
Verify Password Against Configured Password Policy
Send Password Change Request to ITIM
WebSphere Cluster
Tivoli Identity Manager
Operational Workflow Change Password
Process Password Change Request
Change password of All accounts owned by the Identity.
Notification of Password Change

Figure 10 Consistent password management

There are two ways that Identity Manager is used in the realization of this use case to centralize management:

1. Users log into Identity Manager and change their password. The changed password is validated against a configured password strength policy, and if successful, is synchronized with all endpoints, such as Access Manager, Active Directory®, and so on.
2. Users log into the target system that is configured with the Identity Manager reverse password synchronization module[6], and change their password there. Typical instances of such target systems are Access Manager (for access of Web applications), and Active Directory (for network logons such as local area network logon). The Identity Manager reverse password synchronization module is an installable component that captures password change events on a target system, and then communicate with Identity Manager to:
   - Verify that the password meets the password strength policy configured on Identity Manager (for example, is of a minimum length, contains numerals, and so on)
   - If the verification succeeds, store the new password centrally in the Identity Manager repository

The IAM system centrally enforces consistent password management across multiple target systems.

Role-Based Access Control

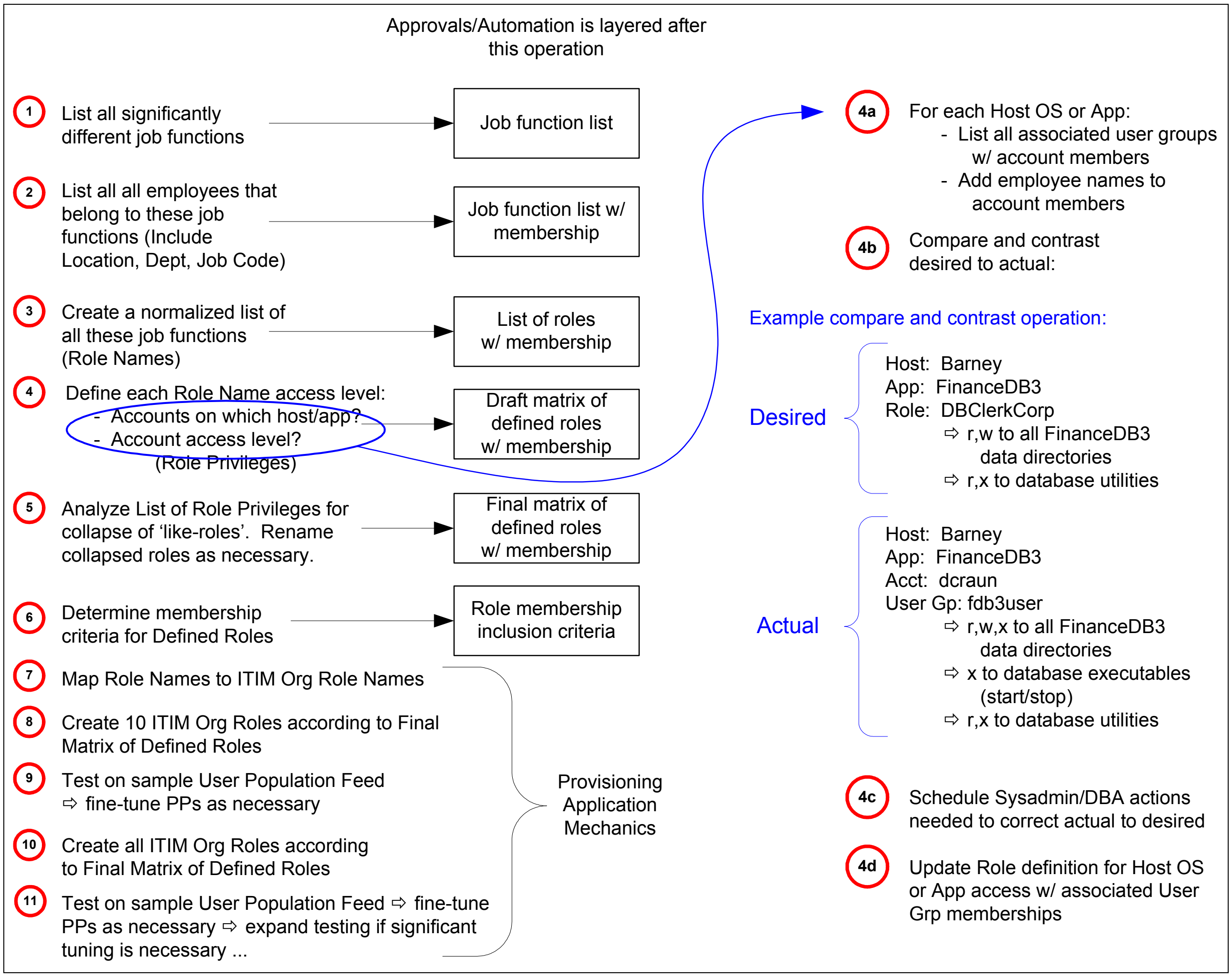
As previously stated, configuring Role Based Access Control (RBAC) in an IAM solution is usually the ultimate goal, but it is achieved through a succession of phased RBAC stepping stones. These phases are:

- Password management with Identity Manager
- Manual provisioning - basic account management (BAM) with Identity Manager
- Automated provisioning (combination of manual and automatic provisioning)
- Automatic provisioning (automated provisioning for all managed end points, with approval mechanisms configured - some small actions necessary by admin)
- True RBAC (automatic provisioning of all user accounts after entry into HRMS - no admin involvement required)

Automated provisioning (automatic provisioning for selected target applications, or systems) is achievable in a reasonable time frame, and continued fine tuning of the IAM solution over time gets the solution closer and closer to true RBAC (the ultimate goal). The Identity Manager mechanics of processing a new user using RBAC is the same as shown in “Enrollment of a new employee”, on page 9. The difference lies in the amount of provisioning automation that has to be achieved to enable true RBAC.

Figure 11 on page 21 depicts the necessary procedure to attempt a true RBAC configuration in Identity Manager. The level of effort needed to complete this procedure is usually prohibitive, but is used to achieve the phased RBAC milestones shown above.

[6] For more information see the IBM Redpaper *Reverse Password Synchronization with IBM Tivoli Identity Manager*, REDP-4299 at http://www.redbooks.ibm.com/abstracts/redp4299.html



Approvals/Automation is layered after this operation

1 List all significantly different job functions → Job function list

2 List all all employees that belong to these job functions (Include Location, Dept, Job Code) → Job function list w/ membership

3 Create a normalized list of all these job functions (Role Names) → List of roles w/ membership

4 Define each Role Name access level:
- Accounts on which host/app?
- Account access level?
(Role Privileges) → Draft matrix of defined roles w/ membership

5 Analyze List of Role Privileges for collapse of 'like-roles'. Rename collapsed roles as necessary. → Final matrix of defined roles w/ membership

6 Determine membership criteria for Defined Roles → Role membership inclusion criteria

7 Map Role Names to ITIM Org Role Names

8 Create 10 ITIM Org Roles according to Final Matrix of Defined Roles

9 Test on sample User Population Feed ⇨ fine-tune PPs as necessary

10 Create all ITIM Org Roles according to Final Matrix of Defined Roles

11 Test on sample User Population Feed ⇨ fine-tune PPs as necessary ⇨ expand testing if significant tuning is necessary ...

Provisioning Application Mechanics

4a For each Host OS or App:
- List all associated user groups w/ account members
- Add employee names to account members

4b Compare and contrast desired to actual:

Example compare and contrast operation:

Desired
Host: Barney
App: FinanceDB3
Role: DBClerkCorp
⇨ r,w to all FinanceDB3 data directories
⇨ r,x to database utilities

Actual
Host: Barney
App: FinanceDB3
Acct: dcraun
User Gp: fdb3user
⇨ r,w,x to all FinanceDB3 data directories
⇨ x to database executables (start/stop)
⇨ r,x to database utilities

4c Schedule Sysadmin/DBA actions needed to correct actual to desired

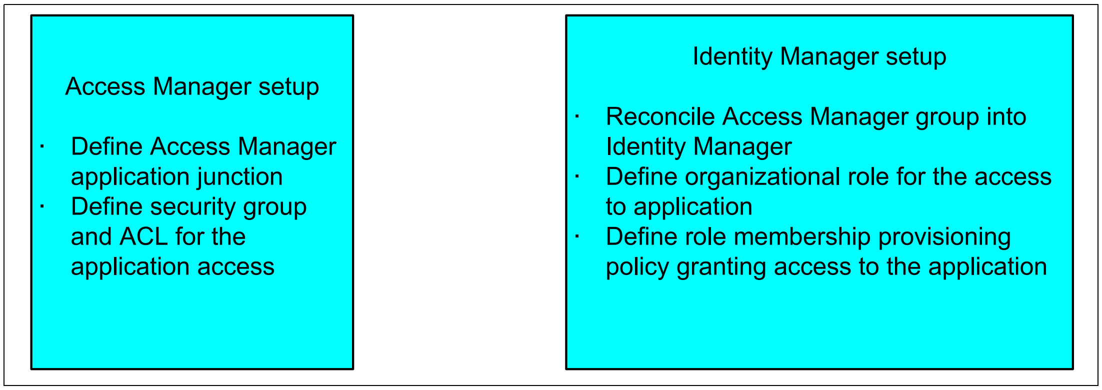
4d Update Role definition for Host OS or App access w/ associated User Grp memberships

Figure 11 RBAC implementation procedures

On-boarding or integrating an application

In Figure 12 on page 21, we show the process of integrating a Web based application into the IAM ecosystem at a high level.

> **Note:** This does not depict application specific items that are needed for this integration.



Access Manager setup

- Define Access Manager application junction
- Define security group and ACL for the application access

Identity Manager setup

- Reconcile Access Manager group into Identity Manager
- Define organizational role for the access to application
- Define role membership provisioning policy granting access to the application

Figure 12 Web-based application integration process

The following tasks must be accomplished:

1. Within Access Manager, configuration must ensure that the application is adequately access controlled.
   a. Defining a *junction* for the application URL within Access Manager[7]. This ensures that when users attempt to access the application at that URL, the Access Manager WebSEAL reverse proxy forces them to go through an authentication and authorization.
   b. Defining required security groups, and access control lists (ACL) for application access within Access Manager. This ensures that only users belonging to the security group configured in the ACL are authorized to access the application.
2. Within Identity Manager, steps that enable provisioning of users to the new security group created in Access Manager would follow. This consists of:
   a. Reconciling the Access Manager security groups into Identity Manager so that they are used in provisioning policies configured on Identity Manager
   b. Defining any organization roles required to authorize provisioning of the new application to users
   c. And finally, defining the provisioning policy to enable users to be provisioned access to the new application

Compliance audit and reporting

Both Identity Manager, and Access Manager are configured to collect audit events of various types. For example, Access Manager is configured to:

- Collect audit events for:
  - Authentication
  - Access control checks
  - User management operations
    - Group creation
    - Adding users to groups
    - Create and update ACLs
  - Provide *out of the box* audit reports for audit events

Identity Manager is configured to:

- Generate audit events for:
  - Workflow processing
  - Provisioning
  - Account recertification
- Provide *out of the box* audit reports for audit events

The Identity Manager, and Access Manager *out of the box* reports are used for incident investigation, and operational needs. In addition, the audit events from Identity Manager, and Access Manager are processed using Tivoli Compliance Insight Manager product[8] to check compliance to security policies, and assess compliance posture of the IAM solution.

[7] For more details on how to define junctions refer to the *IBM Tivoli Access Manager Administration Guide Version 6.0*, SC32-1686, at :
http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/topic/com.ibm.itame.doc/am60_admin.htm

[8] For more information consult the IBM Redbooks® publication *Compliance Management Design Guide with IBM Tivoli Compliance Insight Manager*, SG24-7530 at http://www.redbooks.ibm.com/abstracts/sg247530.html

Self-management of user profile and access to resources

The general patterns for user self-service are shown in Figure 13 on page 23. Figure 13 shows the flow for both the initial self registration, and subsequent self management subscription .

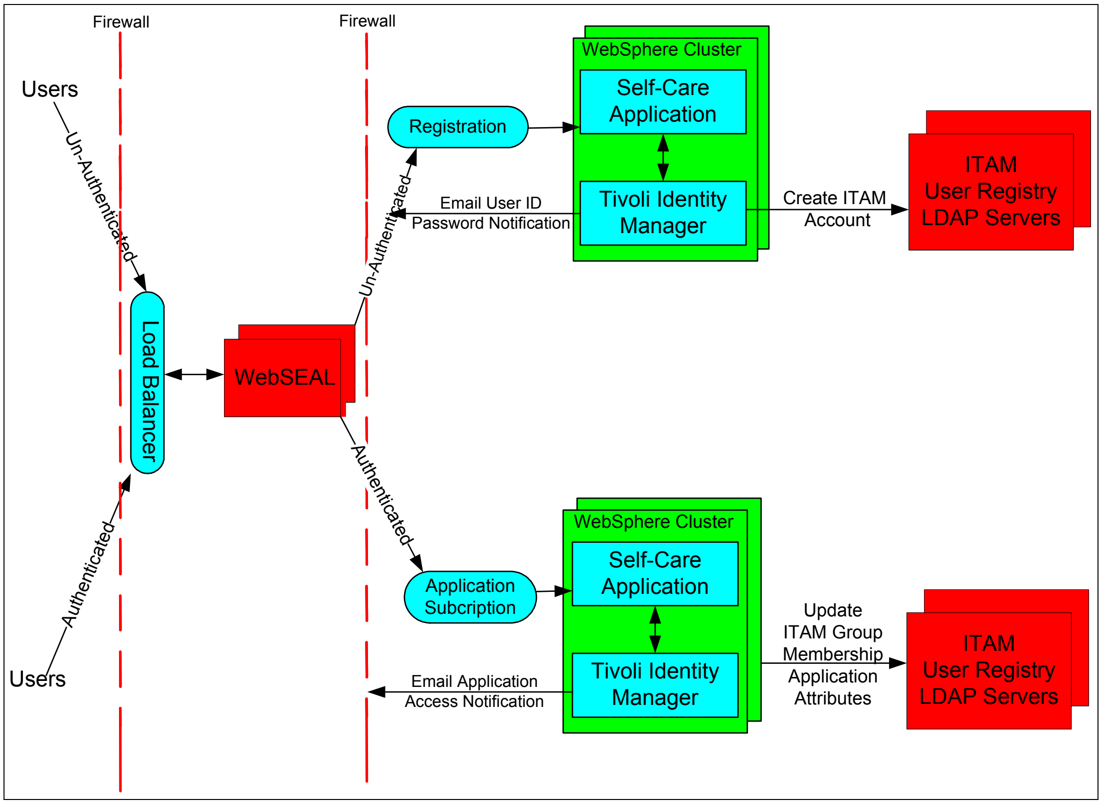


Figure 13 IAM use cases – registration and application subscriptions

Users performing initial self registration to the system are allowed to access the *self-care application* as u*nauthenticated* users by the security Web proxy (for example, WebSEAL). The self-care application front-ends the self registration onto the identity repository that is managed by the Identity Manager. Upon completion of the self-registration process, the user is emailed a User ID, and credentials to access applications. As shown in Figure 13 on page 23, Identity Manager handles the creation of various accounts needed by the user to access various applications, including the creation of the Access Manager account for the user.

The second flow that is depicted has to do with subsequent user subscription management, where users request access to new applications. To accomplish this, users are authenticated using their registered user ID, and password. The self-care application presents a list of applications that the user reviews to request user access. After the user submits the access request to Identity Manager, the request is processed by the Identity Manager operational workflow. The system then notifies the user, typically by e-mail. The system also handles the updates of necessary group memberships, and so on, in Access Manager to authorize the user to access applications.

Forgotten password recovery and reset

In Figure 14 on page 24 we show a realization of the business use case *Forgotten password Recovery and Reset* using IBM Tivoli products.

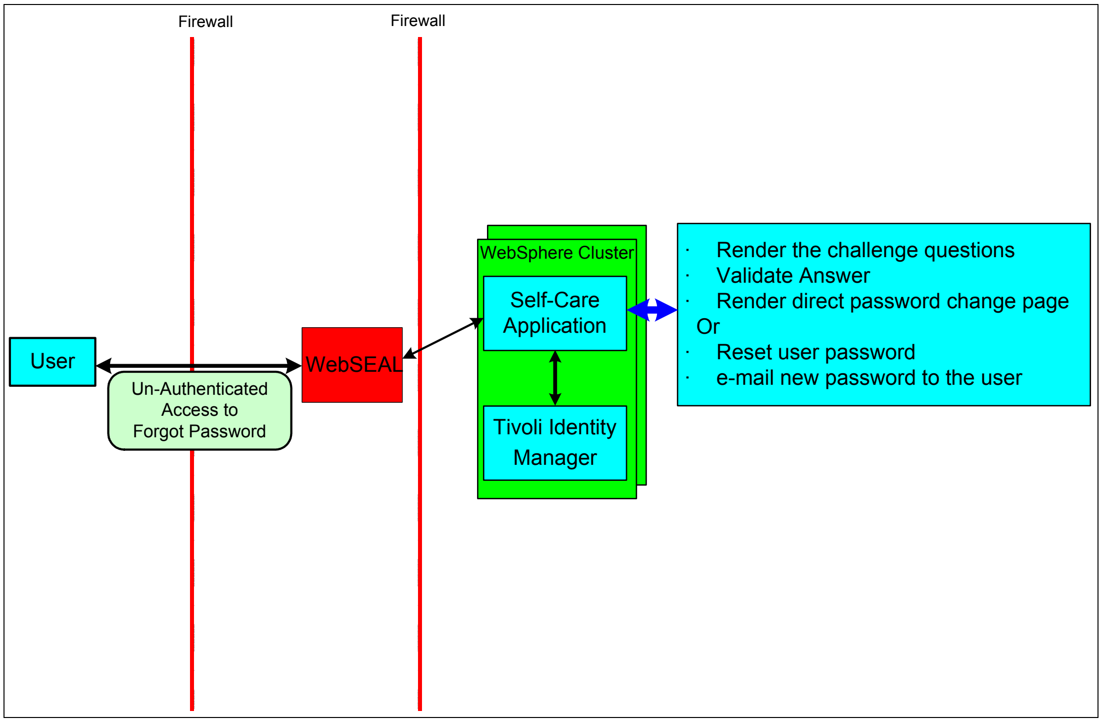

Firewall
Firewall
User
Un-Authenticated Access to Forgot Password
WebSEAL
WebSphere Cluster
Self-Care Application
Tivoli Identity Manager
Render the challenge questions
Validate Answer
Render direct password change page
Or
Reset user password
e-mail new password to the user

Figure 14 Forgotten password recovery and reset

In this case, the user has forgotten the password used to authenticate WebSEAL. In order to facilitate the recovery of the forgotten password by the user, WebSEAL allows unauthenticated access to the self-care application pages that address the *forgotten password* scenario. The user types in the user name, and selects the f*orgot password* link, which takes the user to the self-care application using an unauthenticated connection. The user is shown a set of previously selected challenge questions, such as *Mother's maiden name*, *Best friend in High School*, and so on, the user is expected to provide answers, and submit them to the system. These answers are validated by the system before letting the user either set a new password, or reset the password. If the user selects to reset the password, the new generated password is typically emailed to the user. This new password is usually a temporary, one time use password, and the user is forced to change to a new password upon re-authentication.

SSO to Web applications

In this section we describe a set of common Web-application SSO integrations provided by Access Manager. Additional SSO integrations between Access Manager and third-party products are found on the IBM support site at: http://www.ibm.com/Search/?q=TIVOIAM00.

SSO architecture

The general pattern for SSO is shown in Figure 15 on page 25.

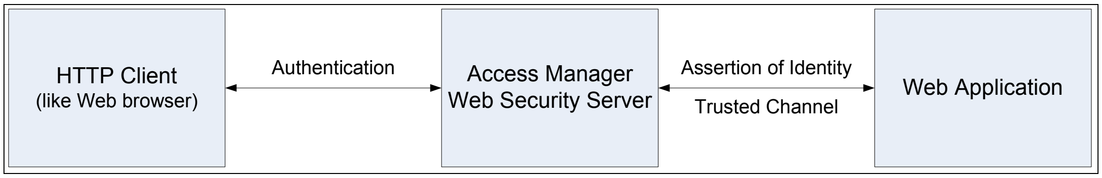


Figure 15 Architecture for Web SSO

In the previous diagram, the Access Manager Web security server is a logical component that can represent one of the following:

- WebSEAL, Access Manager's Web reverse proxy
- A plug-in for an existing Web server, such as IBM HTTP server, Apache Web server, Sun™ Java™ Web system Web server, or Microsoft® Internet Information Server

The SSO solution we describe below is implemented with either the Web reverse proxy (WebSEAL), or a plug-in for another Web server, except in cases where one, or the other is mentioned explicitly.

After authenticating a user, the Access Manager Web security server achieves SSO to applications by asserting the identity of the user to the Web application, using a mechanism that also establishes *trust*:

- Provides independence between the authentication mechanisms used by the Access Manager Web security server to authenticate the user/HTTP client, and the method for achieving SSO with Web applications
- For achieving SSO with the Web application server in which an application runs, as well as directly with an application
- Allows SSO to be achieved with multiple Web applications behind a single Access Manager Web security server
- Allows the SSO methods to be different for each Web application that is secured

In the realizations in the following sections, the different mechanisms for identity assertion and trust establishment are discussed for some common Web applications and application servers.

SSO to WebSphere Application Server

WebSphere Application Server (WAS) provides a couple of mechanisms by which Access Manager is integrated for SSO. Because the integration point is the application server itself, any application which runs a WebSphere Application Server, and relies on the application servers security from this integration. Examples in the IBM software portfolio are WebSphere Portal, and Tivoli Change and Configuration Database (CCMDB).

The first SSO mechanism exploits the *trust association interceptor* interface (TAI++) in the WebSphere Application Server. TAI++ is an interface that allows information from the incoming HTTP request to assert an identity to the WebSphere Application Server. The configuration in the Access Manager Web security server is configured to include the serialized Access Manager credential in the iv-creds HTTP header. This credential is a collection of attributes about a user, not just a simple user identity. The WebSphere Application Server provides a module that is able to receive the Access Manager credential from the Access Manager Web security server, and extract the user identity from the credential. The identity data in the Access Manager credential is trusted by the TAI++ module in a number of ways:

1. In the case of WebSEAL, the junction from WebSEAL is a mutually authenticated SSL to protect the information in transit (this is considered a heavyweight protection mechanism). A lightweight trust mechanism can alternatively be used to establish trust through collusion. A configuration property in the Access Manager Web security server (for example, *basicauth-dummy-passwd* in the WebSEAL configuration file) becomes the *bind password* included in the HTTP basic authorization header of the requests to WebSphere Application Server. The TAI++ module has a configuration property called a *bind username*. If the TAI++ module can successfully use the combination of the *bind username,* and the *bind password* to authentication, and to the Access Manager, or WebSphere Application Server directory, then trust is considered to be established. Use of SSL junctions is optional if this method of trust establishment is used.
2. Validation of the contents of the *Via* HTTP header. The *Via* header is inserted, or updated by WebSEAL to indicate that the request passed through WebSEAL before arriving at the WebSphere Application Server.
3. Presence of Access Manager-specific HTTP headers, such as *iv-user*, *iv-groups* or *iv-creds.*

Beyond SSO, the Access Manager credential is then used within the WebSphere Application Server for container level authorization, behind the Java authorization contract for containers (JACC) interface as shown in Figure 16 on page 26.

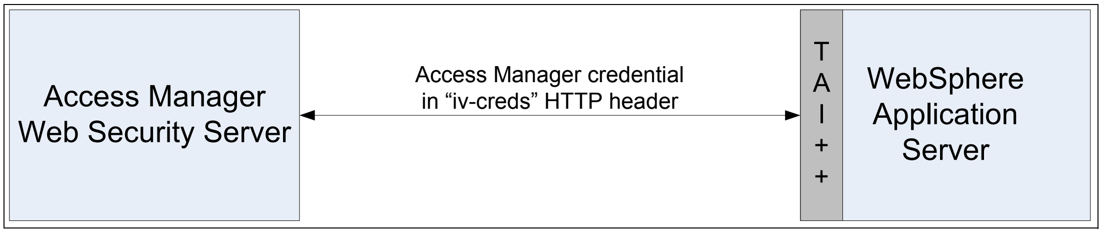

Access Manager Web Security Server
Access Manager credential in “iv-creds” HTTP header
TAI++
WebSphere Application Server

Figure 16 SSO to WebSphere Application Server using TAI++

The second SSO mechanism uses the IBM proprietary *lightweight third party authentication* (LTPA) token, depicted in Figure 17 on page 27. LTPA tokens contain information about a user, typically just the user identity. The LTPA token is protected with symmetric key encryption, so that there is a configuration step to ensure that the LTPA key from WebSphere Application Server is shared with the Access Manager Web security server. Access Manager Web security servers can generate a lightweight third party authentication (LTPA) token and include it in a cookie, including HTTP requests to WebSphere Application Server. WebSphere Application Server extracts the LTPA token from the request and attempts to validate the data in the token to extract the user identity. The assertion of the identity using LTPA token is trusted because of the cryptography used to protect the token contents. There is not necessarily needed for an SSL junction.

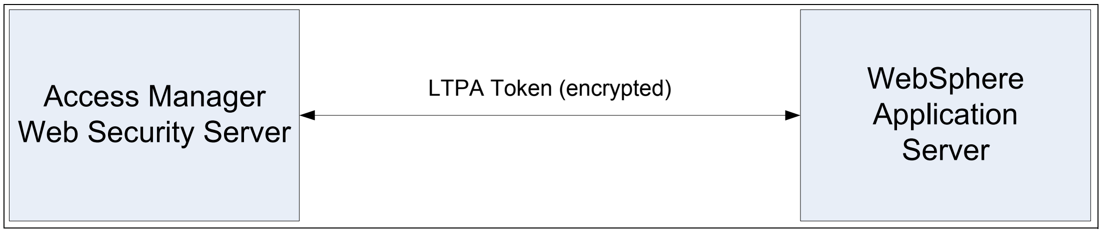

Access Manager Web Security Server
LTPA Token (encrypted)
WebSphere Application Server

Figure 17 SSO to WebSphere Application Server using LTPASSO to Domino Server

Lotus® Domino® servers also support LTPA as an authentication mechanism, the approach described in the previous section is used with domino as well, as depicted in Figure 18 on page 27. In fact, the same LTPA key is shared between WebSphere Application Server, Domino, and Access Manager Web security server in an environment. Again, the trust of an identity assertion in this case is intrinsic to the LTPA token, and an SSL junction is not necessarily required.

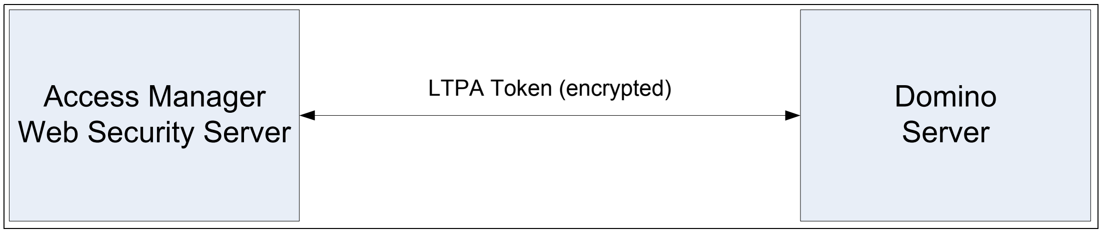


Figure 18 SSO to Domino using LTPA

SSO to Microsoft ASP.NET application server

Microsoft ASP.NET provides a flexible authentication mechanism that allows for custom authentication modules to be implemented. The Access Manager, or Microsoft .NET integration package (AMNET) available from the IBM support site: (`http://www.ibm.com/Search/?q=TIVOIAM00`) provides one of these modules.

SSO is achieved by Access Manager Web security server asserting either the user identity (Figure 19 on page 27), or the Access Manager credential (Figure 20 on page 27).

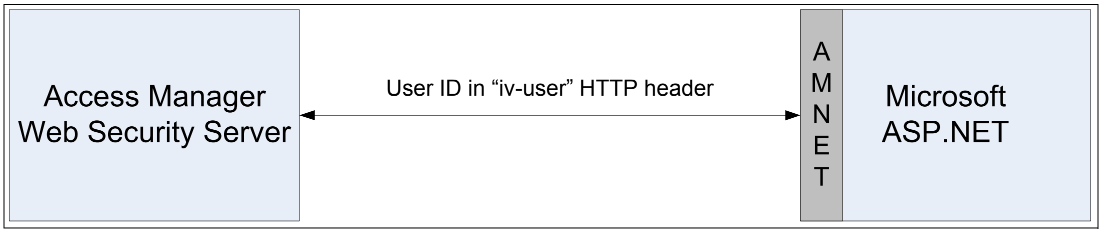

Access Manager Web Security Server
User ID in “iv-user” HTTP header
AMNET
Microsoft ASP.NET

Figure 19 SSO to ASP.NET by asserting User ID

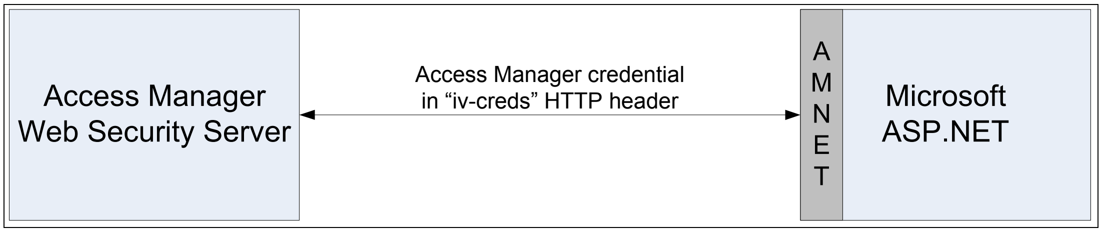

Access Manager Web Security Server
Access Manager credential in “iv-creds” HTTP header
AMNET
Microsoft ASP.NET

Figure 20 SSO to ASP.NET by asserting Access Manager credential

Trust is established in a number of ways that are similar to the trust mechanisms used in achieving SSO to WebSphere Application Server:

- Use of SSL junctions (WebSEAL only) - server, or mutual authentication using X.509 certificates
- Presence of Access Manager-specific HTTP headers
- Contents of the HTTP Via header
- Directory bind

This allows system administrators to choose a model that suits the risk profile of the application being protected.

Use of this module is configured on a per-application basis, rather than in the ASP.NET container itself, which provides flexibility when a single application server is hosting multiple applications with differing authentication, and SSO requirements.

Other identity assertion examples

Access Manager has SSO integrations with a number of other packaged applications, including SAP®, Siebel®, and PeopleSoft®. In these cases, the pattern is similar, Access Manager Web security server asserts the user identity by inserting the identity of the authenticated user in the *iv-user* HTTP header, as shown in Figure 21 on page 28. The applications are aware of SSO solutions, and have their own configuration items to specify where in the HTTP request stream to look for the user identity. Consult the published integration documents for the specifics of these integrations: (http://www.ibm.com/Search/?q=TIVOIAM00).

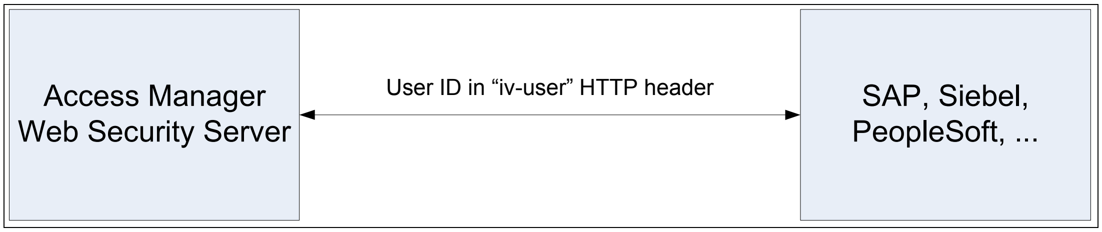

Access Manager Web Security Server
User ID in "iv-user" HTTP header
SAP, Siebel, PeopleSoft, ...

Figure 21 Other identity assertion SSO solutions

Explicit authentication to Web application

In some cases, SSO integrations are required with applications that are not SSO aware, and need, or have to retain their existing authentication mechanism. These applications are configured to solicit the username and password credentials from the user in a HTTP basic authorization header (Figure 22 on page 28), or using an HTML form (Figure 23 on page 29).

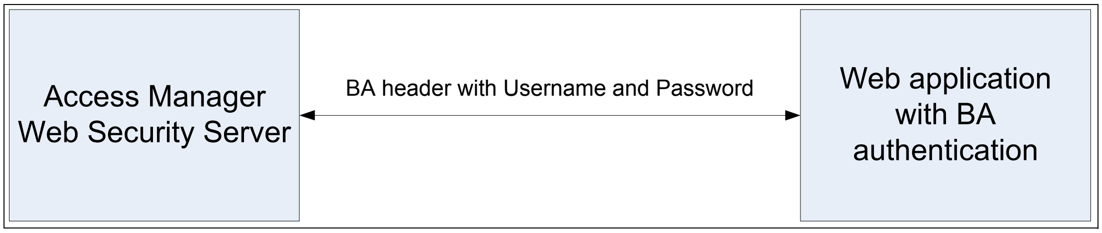

Access Manager Web Security Server
BA header with Username and Password
Web application with BA authentication

Figure 22 Explicit authentication using BA header

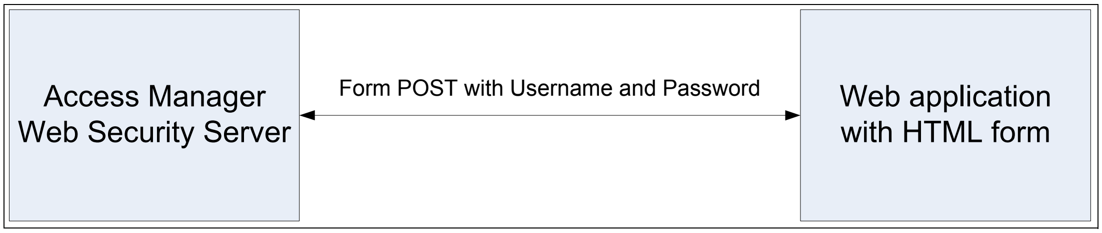

Access Manager Web Security Server
Form POST with Username and Password
Web application with HTML form

Figure 23 Explicit authentication using HTML form

For these situations, Access Manager provides a mechanism for SSO where it can imitate the user and their Web browser from the perspective of the Web application, and send user credentials in the BA header, or in a form POST. Access Manager is able to retrieve the credential data from various places, most importantly is the use of Access Manager's global sign-on (GSO).

GSO is a subsystem in Access Manager that is able to securely store credentials for other applications on behalf of a Access Manager user. For example, for Access Manager user *jsmith*, credentials (*John.Smith, mypassword*) for another application is stored in the Access Manager directory. The credentials are retrieved as part of an SSO solution as shown above. From a management perspective, the Identity Manager adapter for Access Manager includes the ability to manage a user's GSO data.

Note: Building an SSO solution in this way adds administrative complexity in the form of password synchronization, and is a recommended approach only as an interim measure, or when making the application *SSO aware* is investigated, and is not technically feasible.

Related Tivoli SSO offerings

Tivoli Federated Identity Manager (TFIM) provides a solution for single sign-on between multiple administrative domains - cross enterprise, cross department, or cross data center. Federated Identity Manager achieves single sign-on through the use of open industry standards such as Security Assertion Markup Language (SAML), WS-Federation and the Liberty Identity Federation Framework. Federated Identity Manager also offers an enterprise-scale solution that integrates tightly with Tivoli Access Manager for e-Business, as well as solutions for smaller, partner organizations requiring a simple solution architecture with minimal impact on existing IT infrastructure.

For more details, refer to
http://www.ibm.com/software/tivoli/products/federated-identity-mgr

Deployment considerations

In this section, we consider physical deployment architectures of Identity Manager and Access Manager with specific focus on performance and high availability factors.

One of the first things needed when discussing the overall solution performance, or high availability aspects is to define some terms, because this area is commonly misunderstood and often misrepresented. There is a difference between high availability and fault tolerance, and choosing either of these features for a solution design results in very different infrastructures.

High Availability (HA) views availability not as a series of replicated physical components, but as a set of system-wide, shared resources that cooperate to guarantee essential services.

High availability systems do not automatically cutover, but require some combination of scripting, or manual intervention to transfer operations to standby components. HA systems are operational again in usually less than 30 minutes.

Each of these HA support components work together to form an HA complex. In order to achieve a higher level of high availability, component monitoring is needed to form an auto-alert system, which would lower potential Identity Manager and Access Manager HA response time. Identity Manager and Access Manager monitoring most likely involves the IBM Tivoli Enterprise Console® (IBM TEC) paired with alerts generated from a deployed IBM Tivoli Monitoring (ITM) solution.

> **Note:** More information about IBM Tivoli Access Manager, IBM Tivoli Identity Manager and IBM Tivoli Directory Server Monitoring Solutions is found at:
> `http://www.ibm.com/software/tivoli/opal?NavCode=1TW10TM3Z`,
> `http://www.ibm.com/software/tivoli/opal?NavCode=1TW10TM40`, and
> `http://catalog.lotus.com/wps/portal/topal/details?catalog.label=1TW10TM41`

Terms associated with high availability to describe individual components within an HA solution:

Hot standby A method of redundancy in which the primary and secondary (or backup) systems run simultaneously. The data is mirrored to the secondary server in real time so that both systems contain identical information.

Warm standby A method of redundancy in which the secondary (or backup) system runs in the background of the primary system. Data is mirrored to the secondary server at regular intervals, which means that there are times when both servers do not contain the exact same data.

Cold standby A method of redundancy in which the secondary (or backup) system is only called upon when the primary system fails. The system on cold standby receives scheduled data backups, but less frequently than a warm standby. Cold standby systems are used for non-critical applications, or in cases where data is changed infrequently.

Fault tolerance Relies on specialized hardware to detect a hardware fault and instantaneously switch to a redundant hardware component.

Fault tolerant systems can automatically cutover to secondary components in case of a single component fault with no user action. Although apparently seamless and offering non-stop service, a high premium is paid in both hardware cost and performance, because the redundant components do no parallel processing. More importantly, the fault tolerant model does not address multiple software failures, by far the most common reason for downtime.

> **Note:** IBM offers a high availability solution for the AIX® platform called HACMP™ (High Availability Cluster Multi-processing).

Identity Manager performance factors

There are several different components in an Identity Manager infrastructure, both software and hardware. In order to optimize the performance of Identity Manager, an insight into the actual inner workings of Identity Manager and the asynchronous operations conducted are needed, along with actual performance testing. The good news is that this insight and testing revealed some rather straight forward resource allocation tips.

Figure 24 on page 31 shows the major components of an Identity Manager infrastructure that is optimized for maximum throughput of all transactions (all types of possible transactions). This setup is only about Identity Manager performance, with no regard for configuring an HA, or Fault Tolerant solution. As seen in the diagram, an HA setup of the Identity Manager application servers (a single WebSphere Application Server cell with nodes on different physical hosts), and a separated Identity Manager, RDBMS and LDAP server can achieve the highest transaction throughput, provided the resources on the physical hosts are optimized as shown. The reasons for this resource mix are:

- The Identity Manager application server (and underlying WebSphere Application Server) are very CPU and memory intensive, but are not usually I/O bound (usually waiting for results of operations sent to backend databases, or to provisioning Adapters) and do not require a large software footprint. Thus, accomplishing load balancing using the WebSphere Application Server HA setup, and maximizing the speed/number of CPUs and RAM are the biggest contributors to throughput optimization.
- The RDBMS and LDAP servers are both very memory intensive applications. In high traffic situations, they are often I/O bound. Additionally, all the data for Identity Manager resides in these databases (users and all Identity Manager configurations are in LDAP, in-flight transactions and Identity Manager logs are in RDBMS). Servers with large, high-speed HDs, high I/O cards, and high RAM are the biggest contributors to throughput optimization.

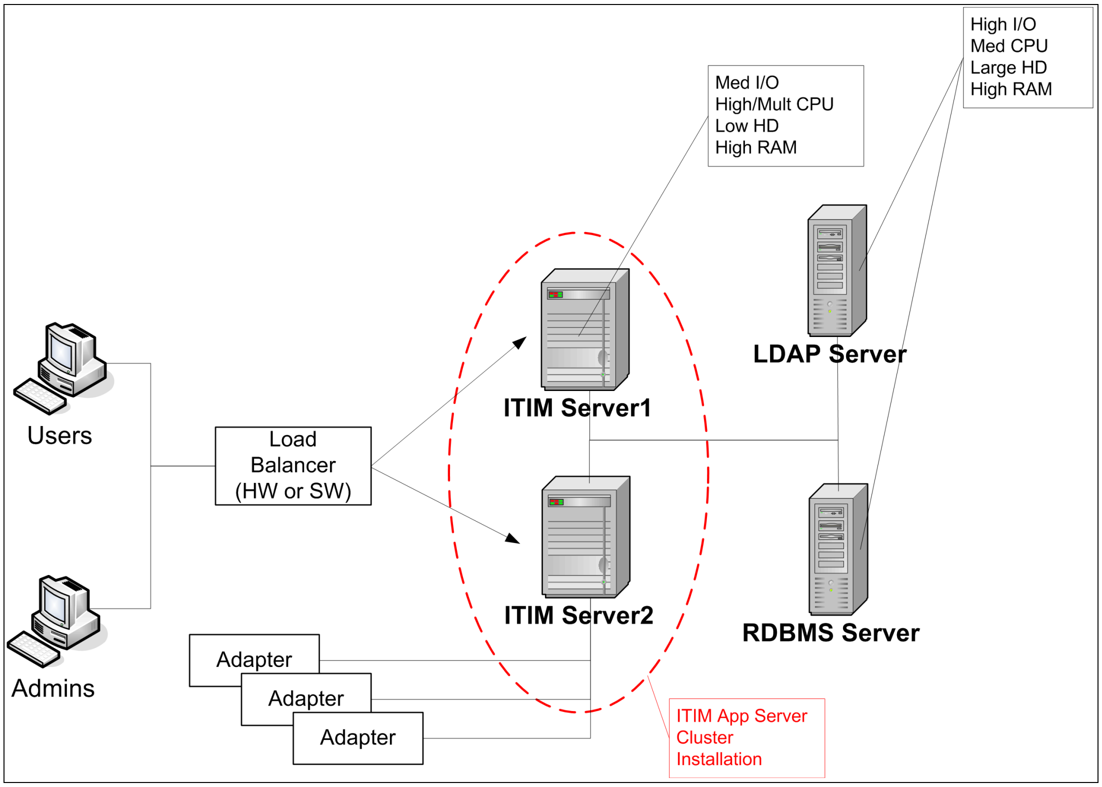


Figure 24 Identity Manager infrastructure performance factors

Identity Manager high availability

The Identity Manager application is a partial high availability solution with a default installation, but can achieve high availability status with additional configurations. Specifically, certain components of the application infrastructure are highly available by default (the Identity Manager application servers), but other components are not installed as highly available by default (the Identity Manager LDAP and Identity Manager RDBMS Servers). However, by implementing warm standby servers (see definitions above) with a data synchronization mechanism outside of the default Identity Manager application installation, you can achieve a high availability system solution.

The Identity Manager application is a partial fault tolerant solution with a default installation, but can achieve fault tolerant solution status with additional configurations. Specifically, certain components of the application infrastructure are fault tolerant by default (the Identity Manager application servers), but other components are not installed as fault tolerant by default (the Identity Manager LDAP and Identity Manager RDBMS Servers). However, by implementing standby servers (see definitions above) with a data synchronization mechanism outside of the default Identity Manager application installation, you can achieve a fault tolerant system solution.

Identity Manager HA design

As stated previously, the Identity Manager application is a partial high availability solution with a default installation. However, by creating warm standby servers with synchronization mechanisms outside of Identity Manager, an overall Identity Manager HA solution is achieved (see Figure 25 on page 33):

- The Identity Manager application server (WebSphere Application Server application) is installed as a single-server or as a multiple node cluster.
  - In cluster mode, together with an IP sprayer, the Identity Manager application servers enable both load balancing and high availability.
- The Identity Manager user registry is not, by default, a high availability component.
  - Achieving high availability involves having standby (idle) components, and configuring a synchronization mechanism to ensure data is not lost during a single failure casualty. The standby components is quickly promoted to master status, thus achieving high availability status for this component.

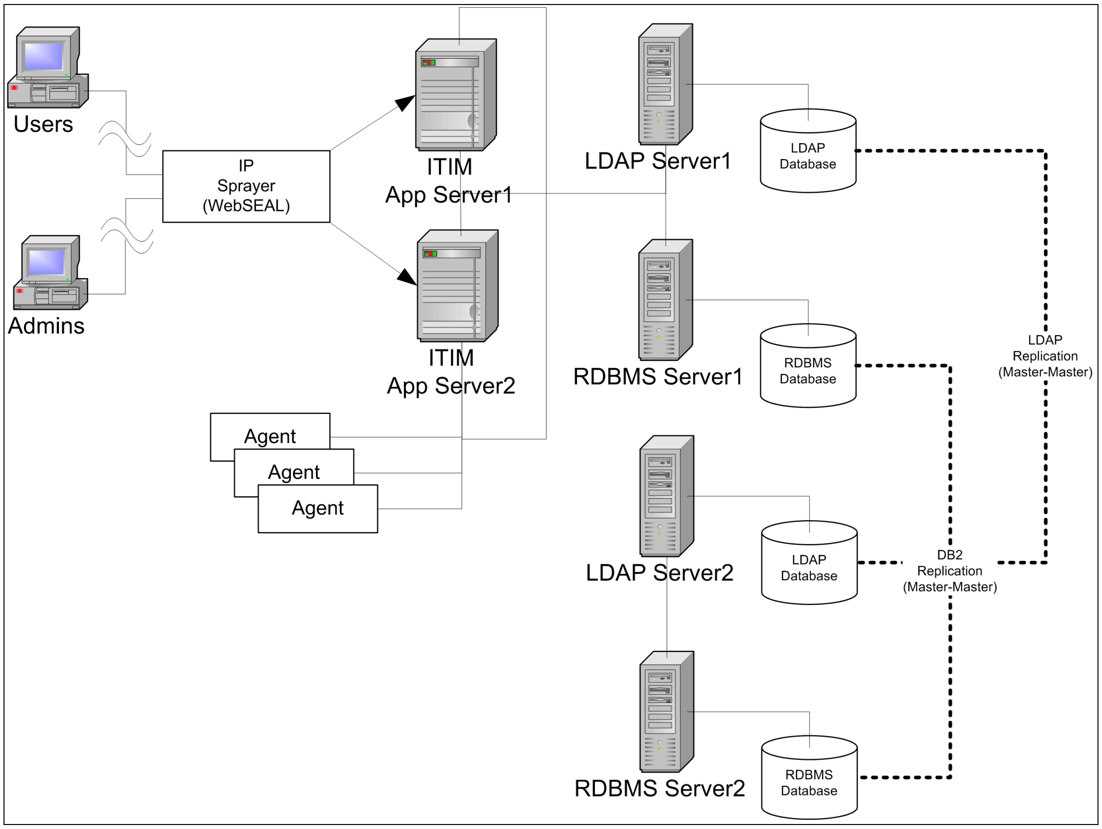


Figure 25 Identity Manager high availability schematic

Identity Manager fault tolerant design

As stated previously, the Identity Manager application is a partially fault tolerant solution with a default installation. However, by creating standby servers with synchronization mechanisms outside of Identity Manager, an overall Identity Manager fault tolerant solution is achieved (see Figure 26 on page 34):

- The Identity Manager application server (WebSphere Application Server application) is installed as a single-server, or as a multiple node cluster.
  - In cluster mode, together with a load balancer, the Identity Manager application servers enable both load balancing and fault tolerance.
- The Identity Manager user registry is not, by default, a fault tolerant component (1 Identity Manager: 1 LDAP: 1 RDBMS)
  - Achieving fault tolerant status involves having standby (idle) components and configuring a synchronization mechanism to ensure data is not lost during a single failure casualty. The failover mechanism is achieved through the use of a load balancer (configured for failover only) for switching off of a non-functioning LDAP server, and configuring a DB2® high availability disaster recovery (HADR) cluster with two nodes on separate physical hosts. All instances of a single component failure stopping the Identity Manager application are eliminated, and fault tolerance is achieved.
  - An alternate option is shown (see Figure 27 on page 34) because of the high capital cost involved in associated hardware needed to implement this solution. Both options achieve the same fault tolerance, but option two reduces the required database servers.

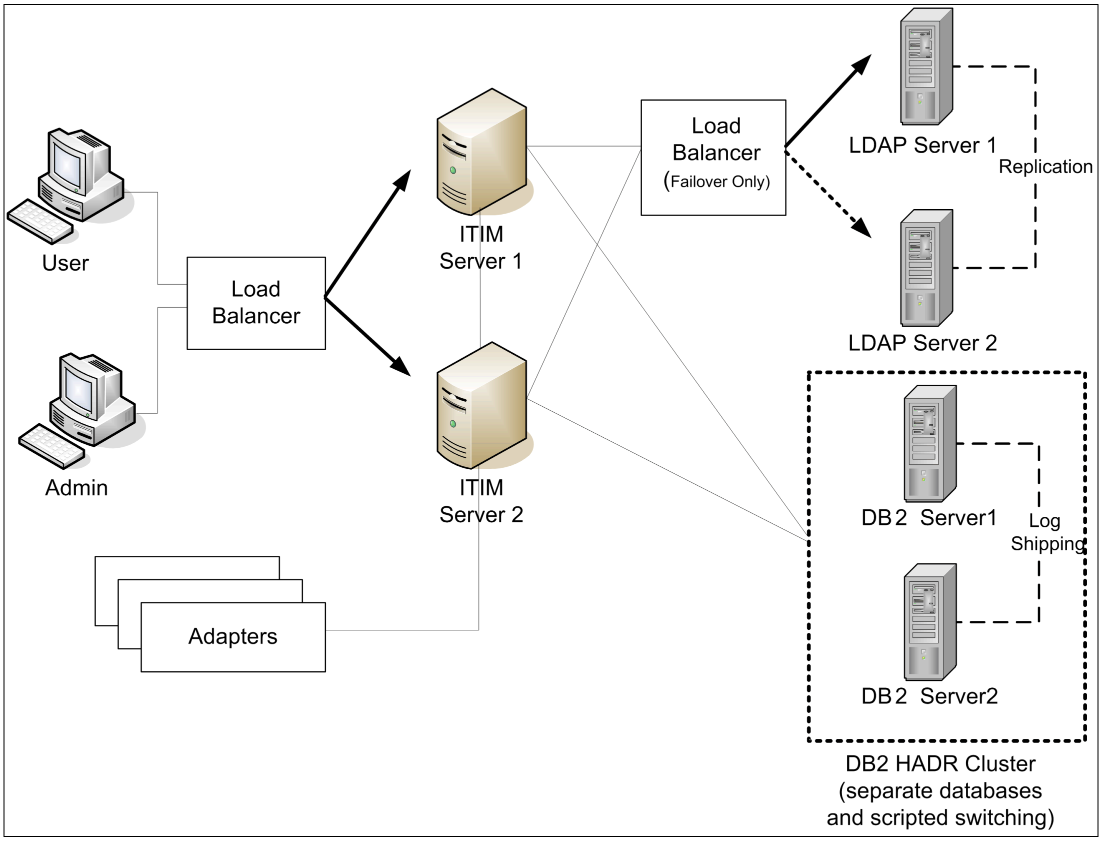


Figure 26 Identity Manager fault tolerant option 1

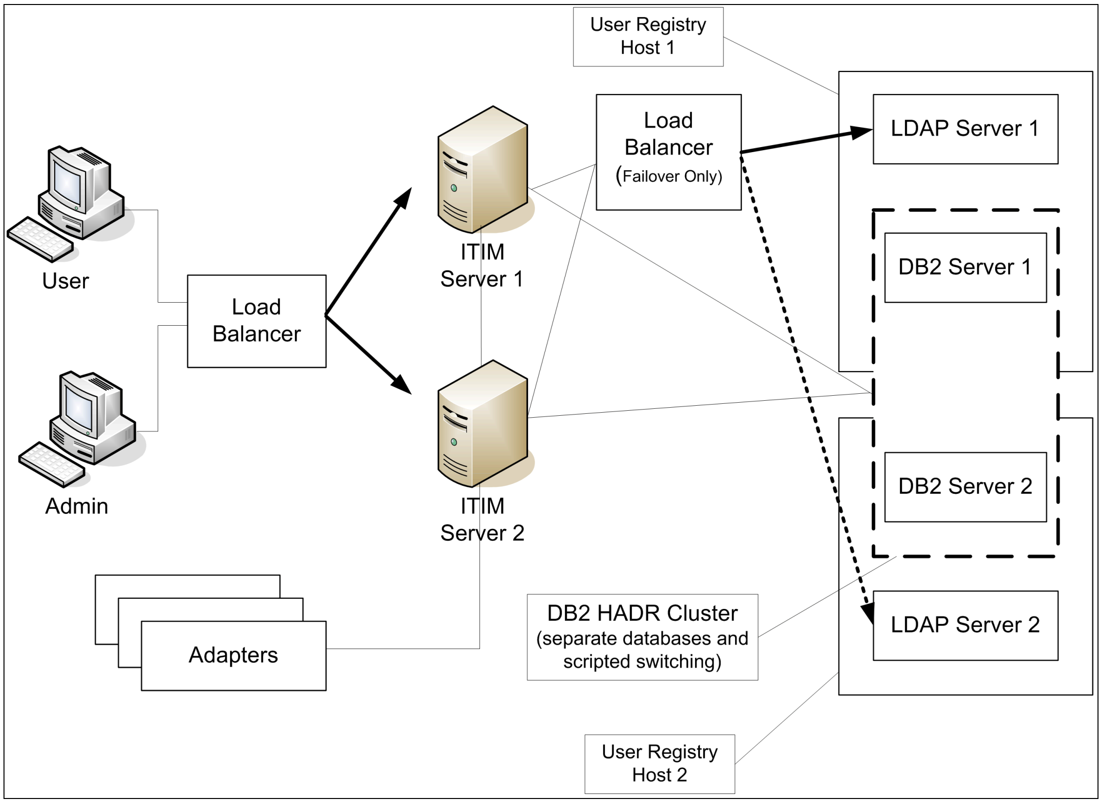


Figure 27 Identity Manager fault tolerant option 2 (HW resources minimized)

Access Manager high availability design

The Access Manager infrastructure consists of a management server, a user registry, and a policy database. There is only one IBM supported high availability solution for Access Manager. This is shown in Figure 28 with the Access Manager server residing on a high availability cluster multi-processing (HACMP) complex. This solution achieves not only high availability, but partial fault tolerance. Disaster recovery (DR) is a clone of the infrastructure we show in Figure 28:

- The Access Manager user registry component is designed for high reliability and high availability.
- The Access Manager gateway service (WebSEAL) is designed for high availability (if configured with IP sprayer and at least two WebSEAL Servers).
- The management process does not support more than one policy database (Master), which is achieved through the use of the HACMP complex.

Note: High availability solutions always take considerable initial capital investment and continuing maintenance issues to ensure adequate operation. It is important to note here that even if the Access Manager server goes down, normal access control and connections are maintained through the Access Manager server endpoints (WebSEAL or secured endpoints). All that is lost during this period is the inability to change security policy.

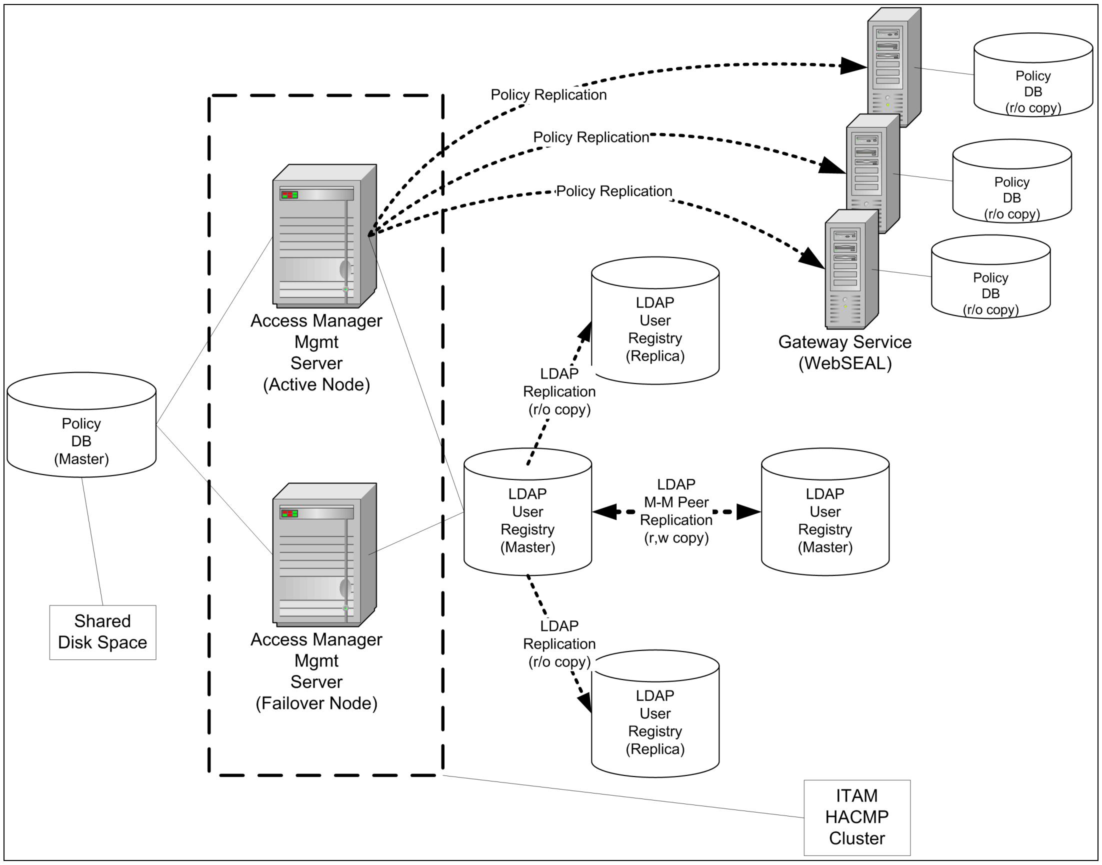

Policy Replication
Policy Replication
Policy Replication
Policy DB (r/o copy)
Policy DB (r/o copy)
Policy DB (r/o copy)
Gateway Service (WebSEAL)
Access Manager Mgmt Server (Active Node)
LDAP User Registry (Replica)
LDAP Replication (r/o copy)
Policy DB (Master)
LDAP User Registry (Master)
LDAP M-M Peer Replication (r,w copy)
LDAP User Registry (Master)
Shared Disk Space
Access Manager Mgmt Server (Failover Node)
LDAP Replication (r/o copy)
LDAP User Registry (Replica)
ITAM HACMP Cluster

Figure 28 Access Manager high availability schematic

Access Manager alternate high availability design

Due to the high costs involved, most customers do not choose the approved HA solution as long as a viable alternative is available. The good news is that there is an alternate solution shown in Figure 29 on page 37. Although this is not a supported IBM design, it is straight forward, and a low-to-moderate risk solution. This design achieves a *warm standby* status for the Access Manager operating systems infrastructure. Additionally, the secondary Access Manager server is located somewhere else, and achieve a disaster recovery (DR) capability to the solution, with no requisite additional HW resources.

- The Access Manager user registry component is designed for high reliability and high availability.
- The Access Manager gateway service (WebSEAL) is designed for high availability (if configured with IP sprayer, and at least two WebSEAL Servers).
- The management process does not support more than one policy database (Master) online at a time. However, you can configure an off-line standby server with a shared policy database. If the master Access Manager server fails, the secondary Access Manager server is quickly promoted.
- (Optional) To minimize the promotion operations necessary for the secondary Access Manager server, a virtual IP address is configured; to the Access Manager servers from the gateway services (WebSEALs), through the use of a network-based load-balancing hardware (such as the Firepass Server from F5 Networks, Inc., as shown in Figure 29 on page 37). Policy replication can continue transparently after a secondary Access Manager server promotion, due to the virtual IP address mapping.

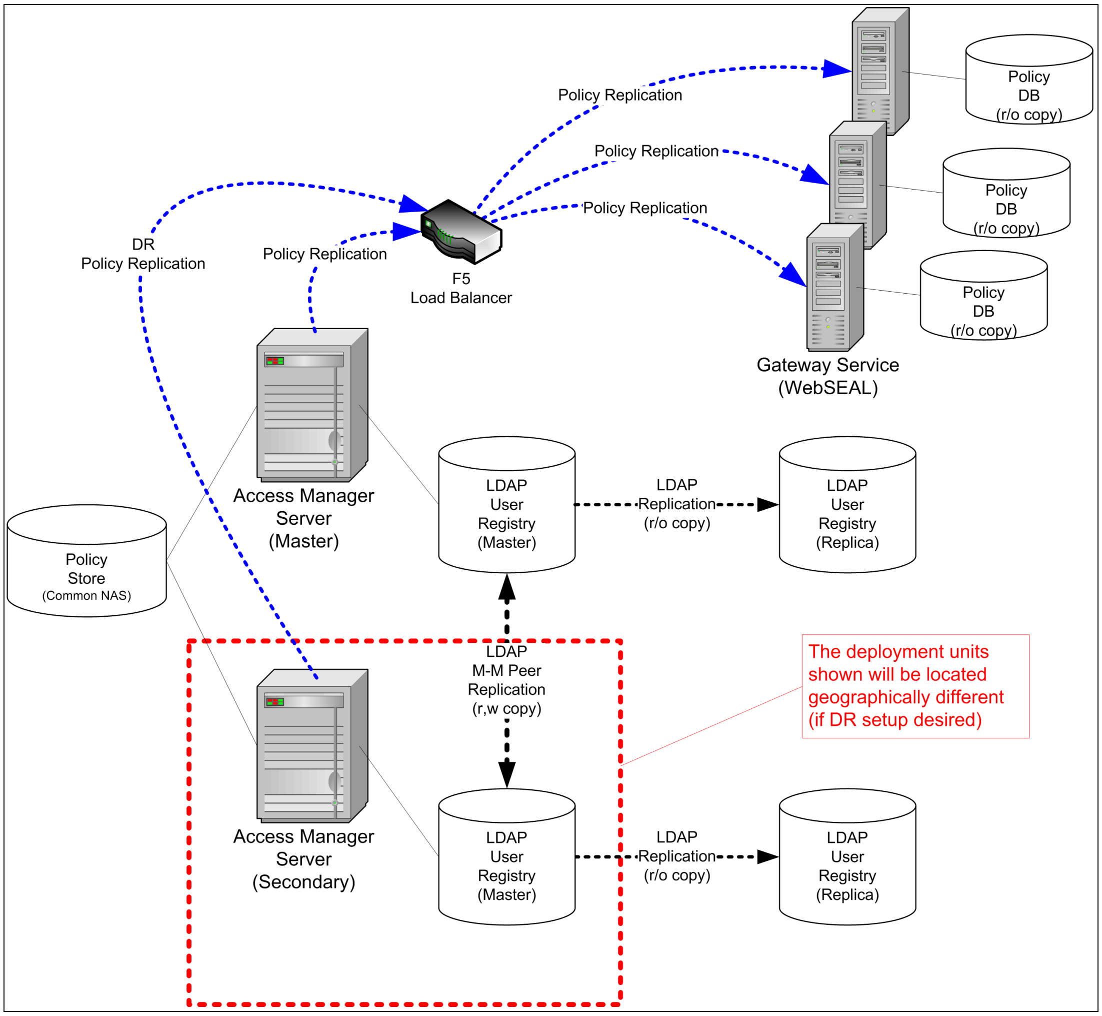


Figure 29 Access Manager alternate high availability schematic

Access Manager fault tolerant design

Access Manager is configured to achieve partial fault tolerant solution status. As described in the previous section, and shown in Figure 29 on page 37, a fault tolerant solution is achieved with the following exceptions:

- The shared policy store
- The master LDAP user registry

Access Manager combo adapter

The Access Manager combo adapter for Identity Manager is constructed using Tivoli Directory Integrator. This provides a flexible development platform, and a high degree of customizing by a customer, or system integrator. When making design decisions around the use of this adapter, keep the following things in mind:

- The adapter can manage additional, non-Access Manager LDAP attributes, whether they are in the standard *inetorgperson object class*, or in a *custom object class*. If standard *inetorgperson* attributes are used, the only customizing required is in the Identity Manager user interface, not in the adapter configuration.

- When managing non-Access Manager attributes from the *inetorgperson object class*, it is recommended that the same name used for the attribute in the adapter schema, be used in the directory. This keeps the attribute mapping and assembly lines simpler, and reduces the amount of customizing needed in the adapter.
- If the adapter is being deployed on multiple Tivoli Directory Integrator servers for high availability, remember that the Access Manager, runtime for Java, must be installed and configured on each Directory Integrator server.
- The design of the reconciliation function in the adapter is optimized for reconciling all entries in a single operation. Reconciliation filters provide minimal benefit in this case.
- The adapter needs to communicate with the Access Manager Policy Server and Directory Server used by Access Manager. This is the case even if the adapter is only being used to manage Access Manager attributes in the directory. Ensure that firewalls rules are designed to allow this communication.

Guidelines and recommendations

In this section we describe a range of recommendations for designing and implementing integrated identity management solutions with Tivoli software. Some general project planning and design guidelines are outlined, followed by some product specific recommendations based on experiences collected by the authors across a number of projects.

Planning and design

The planning and nature of an identity management program can have a large impact on the success of any implementation work. Common themes for project success are described in this section.

- Use an agile deployment approach to deliver capability and business value in a sequence of shorter phases, rather than a fewer number of long running projects. This kind of approach can also complement the efforts. For example:
  - Implement Identity Manager adapters and commence reconciliation early in the project cycle. Even before accounts are managed by Identity Manager, the centralization of account data can deliver some value in terms of reporting and visibility to auditors.
  - Start with simple Identity Manager workflows and increase the level of customizing in cycles. This can avoid over-engineering Identity Manager customizing simply because *it can be done*.
- Define organizational roles in Identity Manager to implement provisioning policies. However, representing every role found in the environments HR system in Identity Manager is rarely required. Roles that have a large number of members are more valuable when you are simplifying provisioning policy design.
- Understand, and review current business processes before deciding how to implement an Identity Manager workflow. Automating inefficient business processes can deliver values, possibly with a higher customizing cost. Reviewing and attempting to optimize existing business processes in an early phase of an identity management project helps to deliver greater overall value.
- Plan an education program for users who interact with the identity management system, for example, managers who are approvers. A low rate of adoption from these stakeholders erodes the value of the technical solution being provided.
- Analyze data that becomes identity feed for the identity management solution. Identify issues with identity feed data as early as possible, to minimize their impact on the project.

- Represent the minimum amount of organizational hierarchy in Identity Manager to meet provisioning policy objectives. Representing the complete organizational hierarchy in Identity Manager is rarely necessary, and its dynamic nature in most organizations create unnecessary burdens.
- Ensure that the test environment for systems that are managed by Identity Manager match their production environments as closely as possible. An identity management project often highlights inconsistencies, and often lead to unforeseen issues in production environments.

Identity Manager deployment best practices

The Identity Manager reference manuals and release notes go into detail about the options and choices available during an Identity Manager installation (in terms of platforms supported), clustering, middleware components supported, and component placement. However, experience shows us that there are some choices that are recommended no matter what the deployment details are, due to the added flexibility and ease of maintenance issues.

These recommendations are listed below:

- The Identity Manager application server is always installed as a WebSphere Application Server cluster (WebSphere Application Server - Network Deployment), even if the cluster is a single node. Setup of the Identity Manager application server as a cluster allows high availability) (HA), and scalable load balancing to be achieved anytime after initial installation with minimal configuration interference (no teardown or minor reconfiguring).
- The Identity Manager user registry is located on a separate platform (host). From both a performance and scalability viewpoint, the user registry (both Identity Manager LDAP and RDBMS) has different resource constraints than the Identity Manager application server. Therefore, the physical resources (CPU, HD, RAM), as well as the tuning characteristics, are different from the Identity Manager App Server, and the Identity Manager user registry. This relationship cannot be optimized unless the components are on separate platforms.
- The application server and database middleware used for Identity Manager must be customized to meet Identity Manager requirements. We recommend using IBM middleware, such as WebSphere Application Server, and DB2 with Identity Manager.
- Keep Identity Manager LDAP suffix identical across all environments - development, test, QA, and production.
  - This helps when using the import / export functions
  - This allows easier policies and workflow migrations from Development/Test → QA → Production
- Define organization hierarchy based on access needs.
- Do not define your hierarchy based on corporate structures, they tend to change frequently.
- Manage targets in phases:
  - Password management
  - Manual provisioning
  - Role base manual provisioning
  - Role base auto provisioning
- Keep Identity Manager DB data and its transaction logs on a separate partition.
- Keep Identity Manager LDAP data and its transaction logs on a separate partition

- If master to master Identity Manager LDAP replication is configured, do *NOT* configure replication agreements until the initial identity load is complete. Once the initial load is complete, copy the records from primary LDAP to secondary LDAP, and then create replication agreements.
- Check the LDAP logs for attributes that are searched, but are not indexed. Create indexes for attributes not indexed.
- Run **`reorgs`** and **`runstats`** after:
  - Large reconciliation
  - Identity load
  - Periodic basis (weekly)

Access Manager 6.0

In this section we provide a non-exhaustive list of recommendations when designing Access Manager 6.0 solutions.

- If intending to implement SSO and authorization for Web applications, structure the project so that SSO is implemented first to begin to deliver business value, and plan to deliver authorization integration for the same Web application in a subsequent project phase.
- SSO techniques requiring password synchronization (WebSEAL GSO junctions and Forms-based SSO) are used sparingly, and only when other SSO techniques are considered and excluded. This is because of the additional administrative complexity in keeping GSO data synchronized with the database-of-record in the target system.
- The Access Manager authorization subsystem is designed so that inheritance of security policy (access control lists (ACL), protected object policies, and authorization rules) is used to reduce the number of explicit policy definitions and attachments. Think carefully about a security policy design that does not adhere to this principle, it can result in a solution that is complex to administer.
- When defining ACL in Access Manager, make use of group-based entries in the ACL to simplify the administration of policy.
- Develop a naming scheme for policy objects: ACLs, protected object policies, authorization rules, protected objects, groups, etc that are used across the entire Access Manager environment. The consistency that this kind of design provides simplifies troubleshooting and auditing of the Access Manager environment, and the applications it protects.
- In cases where an application protected by WebSEAL has multiple instances for high availability, a load-balancer is typically not required between WebSEAL and those instances. WebSEAL has the ability to junction multiple instances at the same junction point, balance loads across those instances, and can deal with individual application instances being unavailable.
- Load-balancers placed in front of a cluster of WebSEAL servers must be configured for session affinity, (also known as "stickiness") so that requests within a single user session uses the same WebSEAL server instance. This avoids the more expensive failover authentication processing that is otherwise the case.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics discussed in this IBM Redbooks publication.

- *Identity Management Design Guide with IBM Tivoli Identity Manager*, SG24-6996
- *Identity Management Advanced Design for IBM Tivoli Identity Manager*, SG24-7242
- *Enterprise Security Architecture Using IBM Tivoli Security Solutions*, SG24-6014
- *Compliance Management Design Guide with IBM Tivoli Compliance Insight Manager*, SG24-7530
- *High Availability and Scalability Guide for DB2 on Linux, UNIX, and Windows*, SG24-7363

The team that wrote this IBM Redpaper

This paper was produced by a team of specialists from around the world working at the International Technical Support Organization, Austin Center.

Axel Buecker is a Certified Consulting Software IT Specialist at the International Technical Support Organization, Austin Center. He writes extensively and teaches IBM classes worldwide on areas of Software Security Architecture and Network Computing Technologies. He holds a degree in computer science from the University of Bremen, Germany. He has 21 years of experience in a variety of areas related to Workstation and Systems Management, Network Computing, and e-business Solutions. Before joining the ITSO in March 2000, Axel worked for IBM in Germany as a Senior IT Specialist in Software Security Architecture.

Dwijen Bhatt is a Senior Managing Consultant for the Central Team of IBM Software group for Tivoli Services for past 10 years. He has architected and deployed many security solutions using Tivoli products over last 9 years at various clients and has over 18 years of experience in the systems management field. He received a Bachelor's degree in Computer Engineering from the Case Western Reserve University, Cleveland, OH in 1986, and a Master's degree in Mathematics from Memphis State University, Memphis, TN in 1989.

Daniel Craun is a Certified IT Specialist in the IBM Software Services for Tivoli (ISST) Security Practice. He resides in San Diego, and is a senior level architect in the Security Practice. He has been with Tivoli Services for over 11 years, and has over 10 years of direct experience with security product architectures and security team lead activities. He has over 20 years experience in system administration duties, and is a University of California San Diego (UCSD) Extension Adjunct Professor teaching “TCP/IP Fundamentals and the Internet” as well as “Advanced TCP/IP and the Internet” courses. Daniel has been one of the leading Security Deployment (multiple product) subject matter experts for 10 years. He developed the first security deployment course, and is one of the principal quality reviewers for all subsequent upgrades of security-related courses.

Dr. Jayashree Ramanathan is an IBM Senior Technical Staff Member and security architect in the Tivoli division of IBM at Austin, Texas. She has 9 years of experience in the security area and her expertise includes security software for access control, auditing, security events, compliance, authentication, and SSO. She received an M. Tech. degree in computer science from the Indian Institute of Technology in Mumbai, India in 1985, and a Ph.D. degree in computer science from Michigan State University in 1992. She joined IBM in 1992. She has authored several technical papers in the security and distributed memory management areas.

Neil Readshaw is a Senior Certified IT Specialist in Tivoli's Worldwide Customer Solutions (SWAT) team. He is based in the Gold Coast, Australia. He has 15 years of experience in software development, network management, information security, and systems integration. He holds degrees in Computer Systems Engineering and Computer Science from the University of Queensland, as well as the Certified Information Systems Security Professional (CISSP) and IT Infrastructure Library® (ITIL®) certifications. He has written extensively for the Tivoli Developer Domain on the IBM developerWorks® site.

Govindaraj Sampathkumar is a Senior Software Engineer in the Tivoli division of IBM at Research Triangle Park, North Carolina. He has 13 years of professional experience as a software developer and architect in various distributed and networking systems and technologies. His areas of interest and expertise include security integration and management, high availability and clustering, parallel and distributed computing, and network protocols. He received an M. Tech. degree in Computer Science & Engineering from the Indian Institute of Technology in Kanpur, India, and has pursued some graduate level studies in Computer Science at the University of North Carolina at Chapel Hill.

Thanks to the following people for their contributions to this project:

Renee' Johnson
International Technical Support Organization, Austin Center

Tim Hahn, David Kuehr-McLaren, Nataraj Nagaratnam, Sridhar Muppidi, Sadanand Bajekal, John Robertson, and Zoran Radenkovic
IBM US

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

This document REDP-4423-00 was created or updated on May 29, 2008.

Send us your comments in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:
  ibm.com/redbooks
- Send your comments in an e-mail to:
  redbooks@us.ibm.com
- Mail your comments to:
  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400 U.S.A.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at:
http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

Redbooks (logo) ®
AIX®
DB2®
developerWorks®
Domino®
HACMP™
IBM®
Lotus®
Redbooks®
Tivoli Enterprise Console®
Tivoli®
WebSphere®

The following terms are trademarks of other companies:

SAP, and SAP logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries.

Oracle, JD Edwards, PeopleSoft, Siebel, and TopLink are registered trademarks of Oracle Corporation and/or its affiliates.

IT Infrastructure Library, IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

Java, Sun, Sun Java, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Active Directory, Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.