IBM Software Group

IBM WebSphere Application Server v6

WebSphere V6 Security

e business on demand.

IBM Confidential

Updated October 12, 2004

Agenda

- WebSphere Security model
- Java Authorization Contract for Containers (JACC) specification
- Tivoli Access Manager (TAM) Client integration in WebSphere

Section

Security Basics

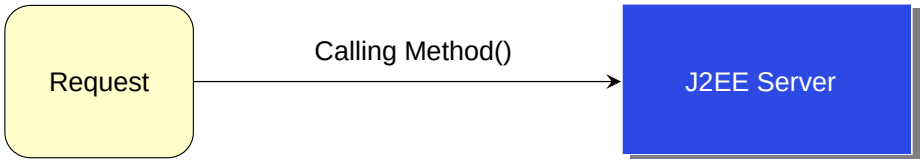
Authentication

- Authentication is the process of establishing whether a client is valid in a particular context
  - Client can be either an end user, a machine, or an application
- An *authentication mechanism* defines rules about security information and the format of how security information is stored in both credentials and tokens
  - whether a credential is forwardable to another process
- May use Authentication Registry to check the client identity
  - Registry stores userid, password and other user information
  - Certificate provides alternative way to establish identity

Authorization

- Authorization is the process that verifies a client has the appropriate privileges to perform an operation
  - Information can be stored many ways
    - Access-control list, capability lists
- J2EE uses role based authorization
  - During assembly, permissions to call methods are given to various roles
  - Roles define a set of permissions within an application
  - During deployment users and groups are assigned to these roles

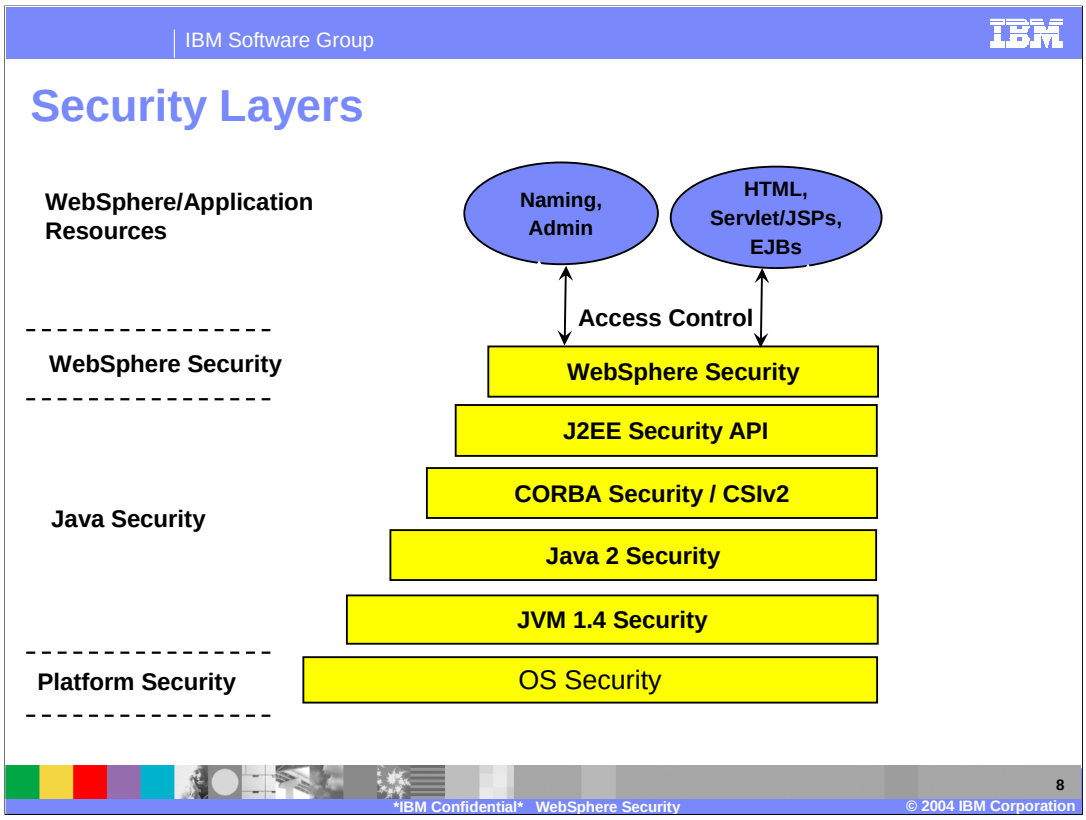
Access Decision Example



1. Challenge the requester to provide credentials (name/password)
2. Check the credentials. If successful, create a Subject with the user information including the groups that the user belongs to
3. Get the required roles for the method from the deployment descriptor
4. Get the assigned roles for the user from the binding file
5. If the required roles match any assigned roles, access is permitted
   - Otherwise denied

Section

WebSphere Security Model



In WebSphere v4 there was security at many different levels. For WebSphere v5 and v6, the security options and capabilities at many of these levels has been enhanced. With WebSphere V5 on the zSeries platform, the WebSphere security layers are intended to work like those on any other platform. What's unique to each platform is the OS security.

•Operating System Security - The security infrastructure of the underlying operating system provides certain security services to the WebSphere Security Application. This includes the file system security support to secure sensitive files in WebSphere product installation. The WebSphere system administrator can configure the product to obtain authentication information directly from the operating system user registry, for example the NT Security Access Manager - SAM.

- •On zSeries, we refer to that as SAF, which stands for System Authorization Facility. SAF is really just another name for the RACF or CA-Top Secret or CA-ACF2 security software and security database that will be present on any OS/390 or z/OS operating environment. These security products contain information about users, groups of users, resources, and user's or group's access to those resources. The purpose of these products its to provide authentication and access control for the OS/390 and z/OS environment.

•JVM 1.4 - The JVM security model provides a layer of security above the operating system layer.

•CORBA Security - Any calls made among secure ORBs are invoked over a Secure Association Service (SAS) or CSIv2 layer that sets up the security context and the necessary quality of protection. After the session is established, the call is passed up to the enterprise bean layer. This layer is for DISTRIBUTED platform only

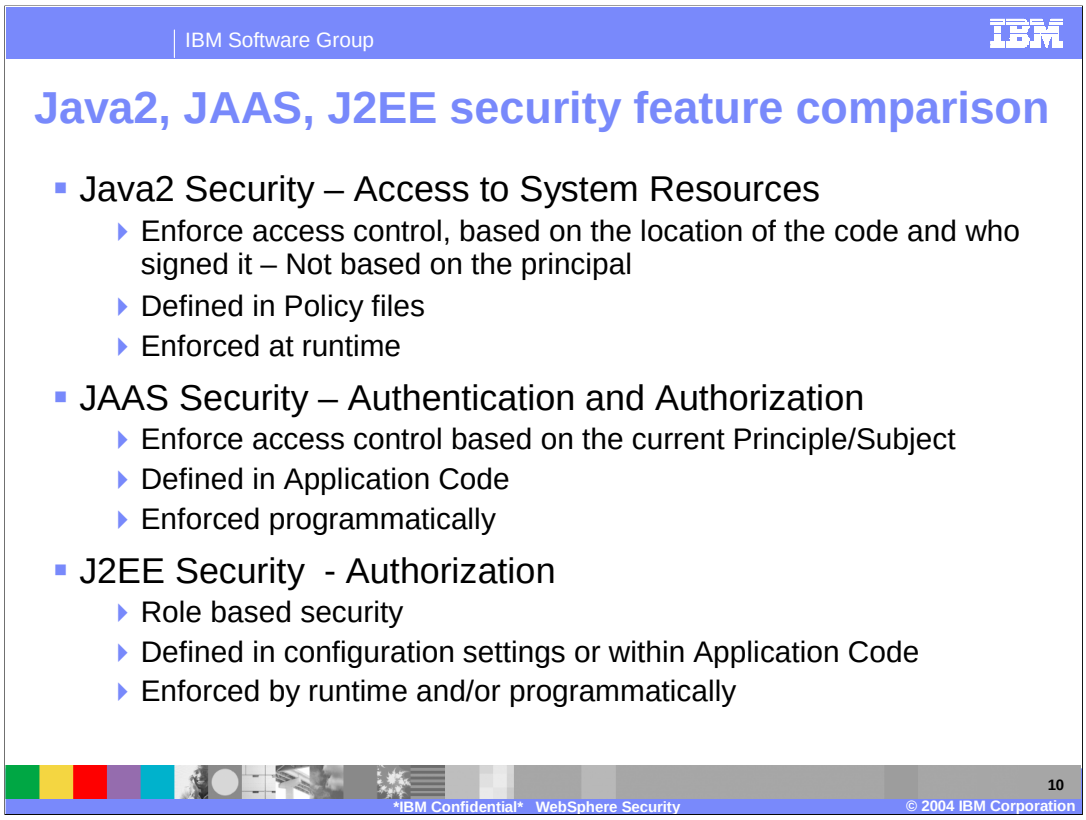
•J2EE Security - The security collaborator enforces J2EE based security policies and support J2EE security APIs.

•WebSphere Security - WebSphere security enforces security policies and services in a unified manner on access to Web resources and enterprise beans. It consists of WebSphere security technologies and features to support the needs of a secure enterprise environment.

Note: SAS (Security Attribute Service) in CSIv2 is different from SAS (Security Association Service) which is IBM proprietary protocol used in WebSphere v4. Also WAS v4 SAS on distributed platform is different than z/SAS for WAS z/OS.

WebSphere v6 Security

- The security configuration and setting is cell wide in Network Deployment cell
  - DMgr, all Node Agents and all Servers have the same security configuration applied
    - Authentication mechanism, registry, etc.
  - Some security settings can be overridden on individual Application Servers
    - Turning off Application security
- Global Security must be enabled



The main difference between Java2, JAAS, and J2EE Security is how the security is enforced.

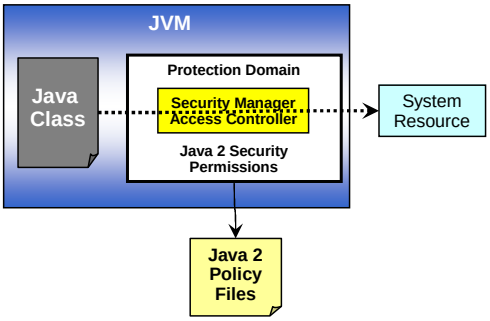
Java2 Security is enforced by the JVM and by Server with the permissions specified in policy files.

JAAS Security is enforced programmatically from within an application.

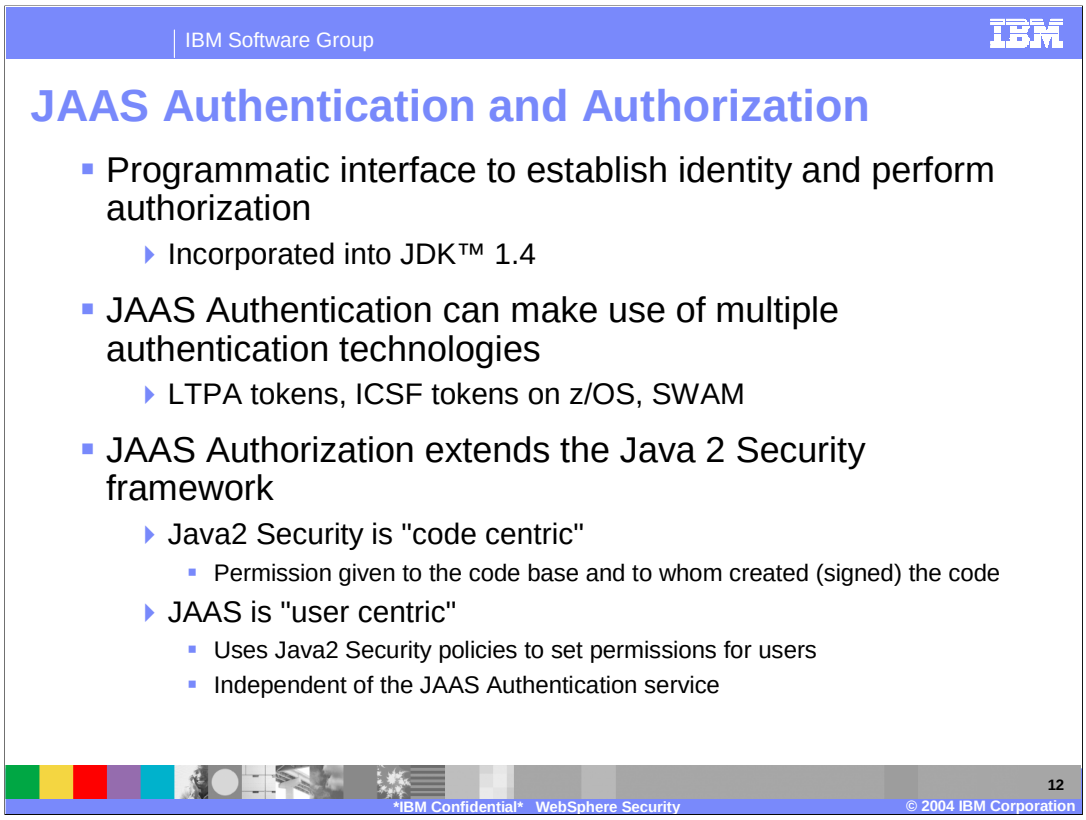
J2EE Security is enforceable by the J2EE server or programmatically from within an application.

Java 2 Security

- Provides an access control mechanism to manage the Application's access to System level resources
  - File I/O, Network Connections (Sockets), Property files, etc…
  - Policy based
- Policies define a set of permissions available from various signers and/or code locations
  - Stored in Policy files
- All Java code runs under a security policy
  - Grants access to certain resources



- Java code needs access to certain System Resources
- Java code will need to get the permission from Java 2 Access Control
- Access Control looks at the Java 2 Policy file(s) to determine if the requesting Java code has the appropriate permission



JAAS is an alternative to the built in security support in WebSphere allowing programmatic authentication and authorization to be used within an application.

PAM (Pluggable Authentication Module) frameworks allow for authentication methods to be implemented without changing any of the login services, reducing the effort in using new authentication technologies. With PAM, applications remain independent from underlying authentication services.

New authentication modules for things such as smart cards and other devices can be easily added based on the PAM framework. PAM also enables the Solaris Operating Environment to work in environments where multiple security mechanisms are in place. http://java.sun.com/security/jaas/doc/pam.html

With Java 2 Security, you only have the ability to distinguish running code based on where it came from and who signed it. JAAS extends this support by adding the ability to provide access based on the identity of the user actually running the code.

Supported Authentication Mechanism

Authentication Mechanism	Intended Use and Supported Package
Simple WebSphere Authentication Mechanism (SWAM) Not available and not needed in WebSphere Application Server v6 Network Deployment and higher packages	▪For simple, non-distributed, single application server environments ▪Does not support *forwardable* credentials or Single Sign ON (SSO) ▪Caller identity is not forwarded from client on one server to EJB on another server - What gets forwarded in unauthenticated credential which may fail on the receiving server
Local Third Party Authentication (LTPA) mechanism Available on all platforms and packages	▪For distributed, multiple application server environments ▪Support *forwardable* credentials or Single Sign ON (SSO) through cryptography ▪Requires all the servers authentication registry to be a centrally shared registry like LDAP
Integrated Cryptographic Service Facility (ICSF) Only on z/OS platforms	▪For distributed, multiple application server environments ▪Support *forwardable* credentials or Single Sign ON (SSO) ▪Supports all WebSphere supported Authentication Registry

Local Third Party Authentication (LTPA)

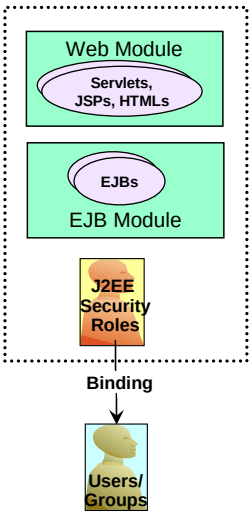
- Intended for distributed, multiple application server and machine environments
- Supports forwardable credentials and SSO
- LTPA protocol uses cryptographic keys (LTPA keys) to encrypt and decrypt user data that passes between the servers
  - If Servers are in different cell, the LTPA keys need to be shared
    - Generate, Export and Import LTPA keys in the admin console
  - All servers in the domain must be synchronized

Single Sign-on (SSO)

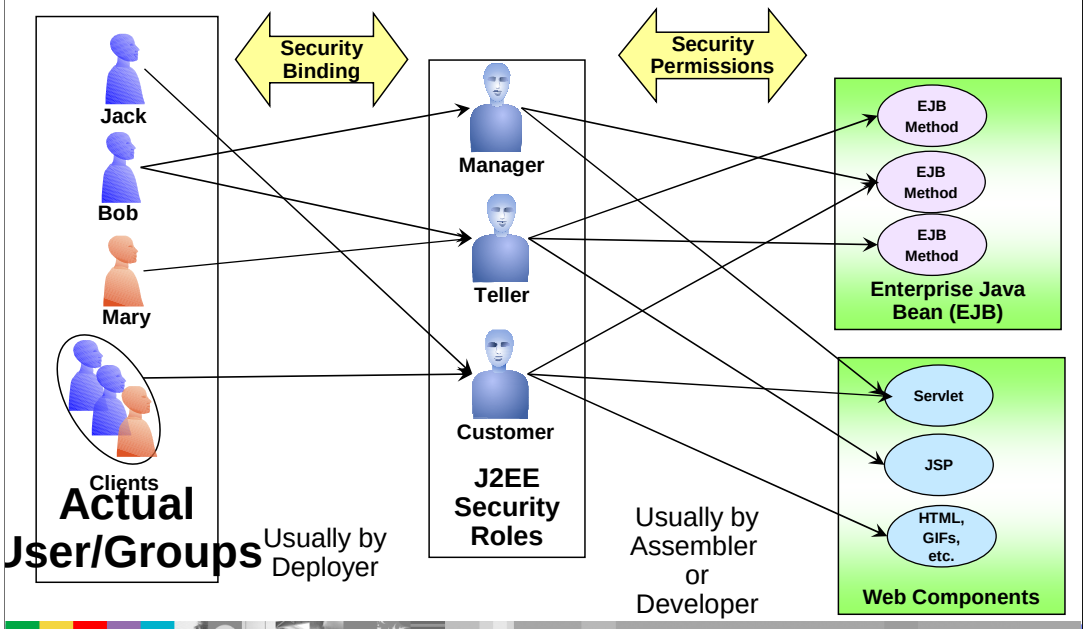
- User authenticates only once in a DNS domain and can access resources in other WebSphere Application Server cells without getting prompted again
  - Requires LTPA across the cells within the domain participating in SSO
  - Same realm names on each system in the SSO domain
    - For local OS
      - On the Windows platform, the realm name is the domain name, if a domain is in use, or the machine name
      - On the UNIX platform, the realm name is the same as the host name.
    - For LDAP, the realm name is the host:port of the LDAP server

J2EE Security Roles: Application Authorization

- Authorization is performed using the J2EE Security Roles
  - Specify security at an abstract level w/o knowledge of actual users and groups
- Security roles are then applied to the Web and EJB application components
  - EJB methods or Web URIs
- Binding of the users and groups to the J2EE security roles are usually done at the application install time
  - Binding information can be saved in the IBM binding file (default) or can use a JACC provider (like Tivoli Access Manager)



Securing J2EE Application artifacts



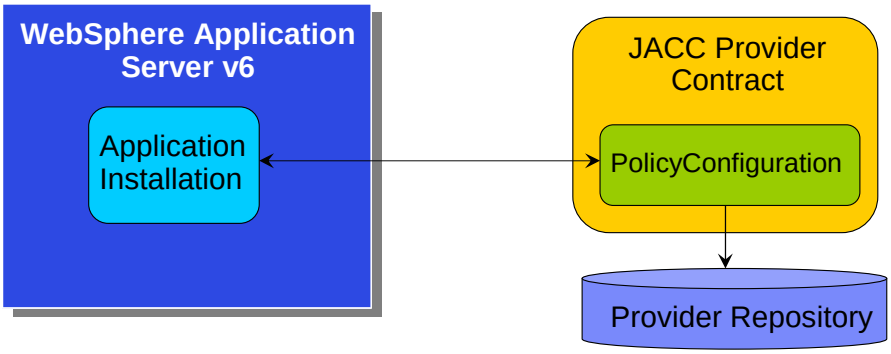
Section

JACC Specification

JACC Introduction

- JACC allows applications servers to interact with third party authorization providers via standard interfaces to make authorization decisions
  - JACC defines permission classes for both the EJB and Web container
  - Handle both J2SE and J2EE permissions
- Does not specify how to assign principals to roles

JACC Example

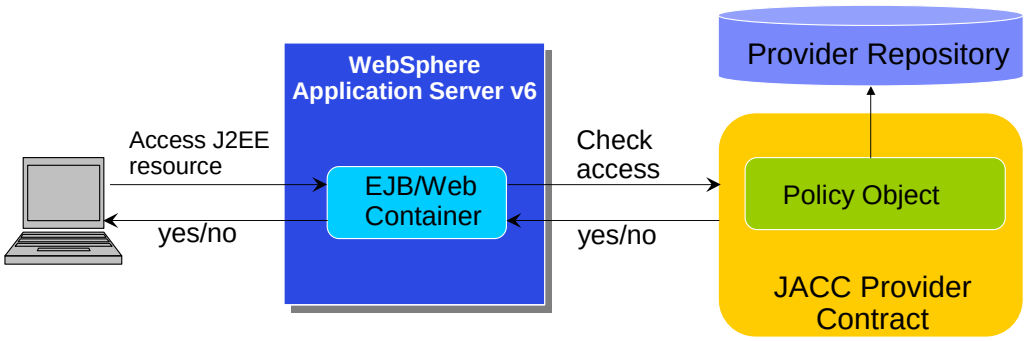


- Create contextID unique to the module being installed
- Get PolicyConfiguration Object for the contextID
- Propagate security policy information for the module using the PolicyConfiguration Object

Deployment an Application Using JACC

- During application installation, translate the security policy in the deployment descriptor to the appropriate permission objects
- Associate the permission objects with the appropriate roles
- Create a unique identity (contextID) for the module being deployed
- Propagate the information to the provider using the PolicyConfiguration object implemented by the provider
- Link all the modules in an application and commit

Application Server Container Requirements



- Create contextID for the module being accessed
- Create the appropriate Permission object for the resource
- Register information required by the specification
- Delegate the access decision to the Policy object

Application Server Container Requirements

- Authenticate the user by checking the user's credentials
- Create a permission object for the resource being accessed
- Register required information by using the PolicyContextHandler objects
- Create the unique identity for the module being accessed
- Call the java.security.Policy object implemented by the provider to make the access decision

Section

Tivoli Access Manager (TAM) Integration

TAM Introduction

- Provides unified authentication and authorization services for heterogeneous environments
- Enforces security by using a single security policy server across multiple file types, application providers, devices and protocol
- Access decisions are based on information held external to the application

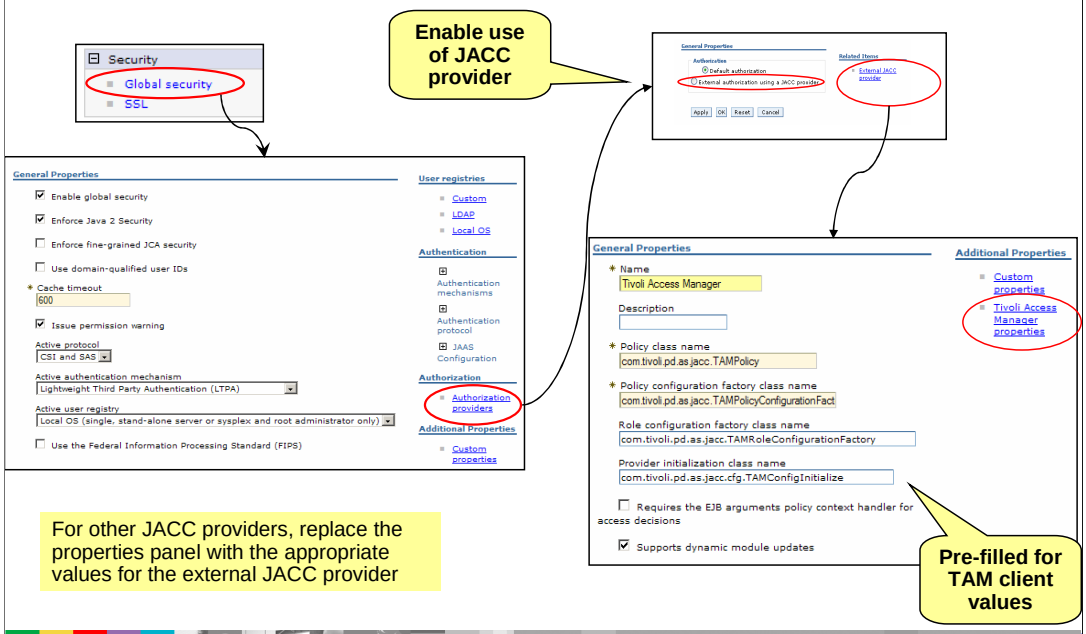
TAM Integration

- The TAM client pieces are embedded in the WebSphere Application Server
- The TAM server is bundled in the WebSphere Application Server v6 Network Deployment
- TAM is the default JACC provider for WebSphere

TAM Integration: continued

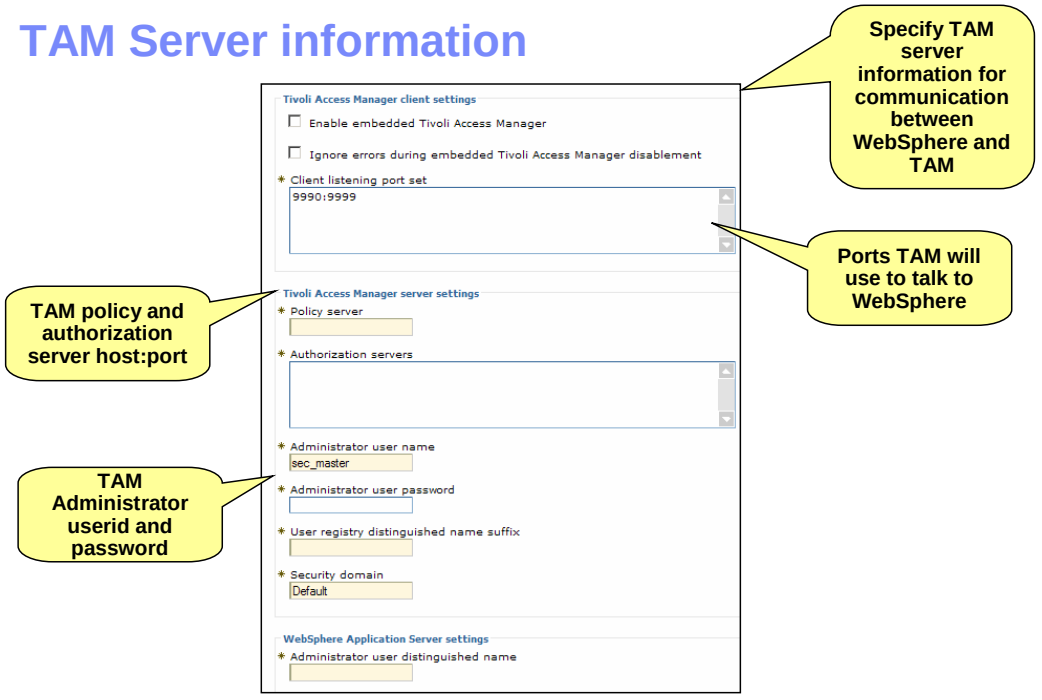
- When TAM is used as the JACC provider, the GUI panels and the wsadmin scripting used to associate the Principals (users/groups) to roles directly communicate with the TAM server
- The TAM client can be configured using the scripting and the GUI management facilities of WebSphere
- Authentication can also be performed by the TAM server

TAM integration in Admin Console



For other JACC providers, replace the properties panel with the appropriate values for the external JACC provider

TAM Server information



Section

Summary and References

Summary

- Overview of role-based authorization
- Details of the JACC specification
- TAM integration in WebSphere

References

- JSR 115 specifications
  - www.jcp.org/en/**jsr**/detail?id=**115**
- J2EE 1.4 Tutorial
  - http://java.sun.com/j2ee/1.4/docs/tutorial/doc/index.html

Template Revision: 10/01/2004 4:46 PM

Trademarks and Disclaimers

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

IBM	iSeries	OS/400	Informix	WebSphere
IBM(logo)	pSeries	AIX	Cloudscape	MQSeries
e(logo)business	xSeries	CICS	DB2 Universal Database	DB2
Tivoli	zSeries	OS/390	IMS	Lotus

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both. Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both. Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are
trademarks of Intel 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. Linux is a registered trademark of Linus Torvalds. Other company, product and service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or program(s) described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. 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 in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.

This Page Intentionally Left Blank