WebLogic Server 7.0 Single Sign-On: An Overview

Today, a growing number of applications are being made available over the Web. These applications are typically comprised of different components, each of which may have its own authentication scheme or user registry. As are result, development and operations staffs are faced with the increasingly difficult problem of how to require their user community to authenticate or "sign-on" once, yet have access to each of the components of the application. The ability to require a user to sign-on to an application only once is called **Single Sign-On**.

BEA's WebLogic Server 7.0 attempts to address a number of the issues concerning Single Sign-On support through a series of approaches, each focused at a different tier in which a portion of an application is housed, and addressing the problems that appear in that tier. The new security architecture found in WebLogic Server 7.0 provides essential integration points, called **security providers**, that allow best-in-breed solutions to be "plugged in" by commercial security vendors or customers themselves. In addition, WebLogic Server 7.0 provides implementations for most of these providers.

Web Tier Single Sign-On

With a growing focus on the Web as the user interface to applications, the Web tier is fast becoming the most common place where the requirement for Single Sign-On appears. In the Web tier, the user of a browser is prompted to authenticate their identity to the application. This identity is then propagated to the application server and utilized in the authentication of the user. The result of a successful sign-on is a cookie that is scoped to the DNS domain in which the application server is resident. This cookie is then returned to the browser where it is sent with each future request to the application server.

As a J2EE 1.3 conformant application server, WebLogic Server 7.0 supports all of the required mechanisms for authentication and Single Sign-On. These include: (1) basic authentication, where the user's credentials are only protected by a simple base64 encoding; (2) form-based, where the credentials are not protected and sent in clear text; and certificate-based, where X.509 certificates held by the client are used with the SSL or TLS secure protocol to establish the identity of the user. The first two forms of authentication typically require the use of a secure transport, such as SSL or TLS, in order to protect the rather weak security utilized to protect the user's credentials.

In addition to the J2EE prescribed mechanisms, WebLogic Server 7.0 can also be configured to support the use of additional authentication technology. This is possible through the use of the extensible WebLogic Security Framework. The use of the WebLogic Security Framework allows for the consistent enforcement of authentication policies regardless of whether the client is a browser or an application. This also removes the requirement for each application having to provide its own servlet filter with which to perform authentication. A further benefit is that this allows the type of authentication mechanisms, such as the use of a stronger authentication mechanism, to be changed without requiring changes in the application itself.

Single Sign-On Extensions

WebLogic Server 7.0 goes beyond the J2EE 1.3 specification with its support for Single Sign-On by supporting fail over and load balancing to other members of a cluster without the need to re-authenticate to each cluster member. Furthermore, if the session has been configured for file or JDBC persistence, then the Single Sign-On solution can be extended even further to other non-clustered servers within the same DNS domain. Finally, it is possible to both disable Single Sign-On, as well as to create a group of different applications or Web components that participate in a Single Sign-On group by specifying a name, which is used to control the scope of a cookie within the `weblogic.xml` deployment descriptor file. The use of Single Sign-On groups provides the ability for a single DNS domain to be home to multiple applications, each of which can control which other applications or Web components are allowed to participate as part of its particular Single Sign-On environment.

Cross-Domain Single Sign-On

Although the J2EE 1.3 specification does not address the concept of a Single Sign-On environment that spans multiple DNS domains, WebLogic Server 7.0 is uniquely designed to support this type of environment through partnerships with other security vendors. **Cross-Domain Single Sign-On** allows users to authenticate once but access multiple applications, even if these applications reside in different DNS domains. This provides the ability to construct a network of affiliates or partners that can participate in a Single Sign-On domain.

Through the use of the Security Framework contained in WebLogic Server 7.0, it is possible for standards-based solutions to be utilized to provide Single Sign-On integration at both the affiliate, as well as the global level. In particular, emerging standards and de-facto standards such as the Security Assertions Markup Language (SAML) from Oasis and Internet-wide solutions such as Microsoft Passport™ and the result of the Liberty Alliance can be integrated with WebLogic Server 7.0 to create an even broader Single Sign-On solution.

Beyond the Web Tier

In today’s enterprise applications, it is rare that all the components that make up the application are all contained in the Web tier or are hosted on the same application server. As a result, it is critical that a Single Sign-On solution support integration of those components as part of the overall offering.

WebLogic Server 7.0 provides a Single Sign-On solution that extends beyond the Web tier in order to incorporate integration with legacy systems and other J2EE application servers as part of its standard product features. The mechanisms used to support this functionality can be extended by customers, security vendors, or ISVs to provide enhanced capabilities.

Single Sign-On with Legacy Systems

As part of the Single Sign-On solution provided by WebLogic Server 7.0, application adapters defined by the J2EE Connector Architecture are able to acquire the credentials necessary to authenticate with the other components that make up the business solution. The credential acquisition capabilities are provided as part of the new security framework contained in the WebLogic Server 7.0 release. Through this framework, the Connector container is able to retrieve the appropriate set of credential information for the target, based on the information in the deployment descriptor, and the mapping contained in the one of the Credential Mapping providers. The credential information is then passed to the adapter, where it can be utilized to authenticate to the target.

A Credential Mapping provider is responsible for providing a mapping between the combination of the requestor’s identity and the desired resource to the credential set appropriate for authenticating with the desired resource for that requestor. In order to allow for easier administration, the identity of the requestor used in the mapping can be either the username under which the requestor authenticated, or the name of a group in which the requestor must be a member.

Each Credential Mapping provider can be associated with one or more credential formats, including Kerberos tickets, SAML name assertions, or others. The format of the credentials supported by a given Credential Mapping provider is registered with WebLogic Server at the time the provider is instantiated. BEA provides a WebLogic Credential Mapping provider that can yield credentials in the form of username and password as part of the WebLogic Server 7.0 release.

Single Sign-On with Other J2EE Application Servers

The conformance with Level 0 of the Common Secure Interoperability (CSI) v2 specification allows WebLogic Server 7.0 to participate in a secure, Single Sign-On environment with Enterprise JavaBeans (EJBs) hosted in other conformant J2EE application servers over the IIOP protocol.

WebLogic Server 7.0 supports the required GSSUP username-password authentication mechanism as a means to authenticate to another conformant J2EE application server, as well as the ability act as the target of such authentication. It can also support the use of Identity Tokens, as described in the CSI v2 specification, when configured in a trust relationship between the two containers.

Message-Driven Beans

Another aspect of Single Sign-On amongst J2EE application servers that can occur is when an application utilizes Message-Driven Beans with a foreign JMS provider, such as IBM MQueue Series. In this scenario, the WebLogic Server EJB container must authenticate itself to the foreign JMS implementation in order to retrieve the queued messages that are to be dispatched to the application code.

As with J2EE Connection adapters, WebLogic Server 7.0 utilizes the Credential Mapping providers as a means to obtain the necessary credentials to authenticate with the foreign JMS provider.

The use of the Credential Mapping provider allows the mapping between the foreign JMS provider and the requestor's identity to the appropriate credential set to be defined using the WebLogic Server Administration Console. In addition, because Credential Mapping providers are able to be written by customers and external vendors, custom implementations that contain enhanced mappings are possible.

Support for Perimeter Authentication

With the emergence of new integration paradigms such as Web services and the growing requirements for extending Single Sign-On throughout the enterprise, the requirement for support of authentication that occurs outside of the application server itself is becoming more prevalent. **Perimeter authentication** is the term used to describe the scenario where the process of authenticating the identity of a remote user is performed at the perimeter of an application or enterprise. This is typically accomplished by the remote user specifying an asserted identity and some form of corresponding proof material, normally in the form of a pass phase, which is used to perform the verification. The **authentication agent**, the entity that actually vouches for the identity, can take many forms such as a Virtual Private Network (VPN), firewall, an enterprise authentication service, or some form of global identity service.

Each of these forms of authentication agents has a common characteristic: they all perform an authentication process that results in an artifact or **token** that is must be presented to determine information about the authenticated user at a later time. The format of the token varies from vendor to vendor, but there are efforts in Oasis to define a standard token format using XML, in addition to a current standard for Attribute Certificates (which is based on the X.509 standard for digital certificates).

But even after all of this, if the applications and the infrastructure on which they are built are not designed to support this concept, enterprises are still forced to require that their remote users re-authenticate to the applications within the network.

WebLogic Server 7.0 is designed to extend the Single Sign-On concept all the way to the perimeter through support for identity assertion. Provided as a critical piece of the WebLogic Security Framework, the concept of identity assertion allows WebLogic Server 7.0 to utilize the authentication mechanism provided by perimeter authentication schemes such as Checkpoint's OPSEC, the emerging Security Assertion Markup Language (SAML), or enhancements to protocols such as Common Secure Interoperability (CSI) v2 to achieve this functionality.

Support for perimeter authentication requires the use of an Identity Assertion provider that is designed to support one or more token formats. Multiple and different Identity Assertion providers can be registered for use. The tokens are transmitted as part of any normal business request, using the mechanism provided by each of the various protocols supported by WebLogic Server 7.0. Once a request is received with WebLogic Server, the entity that handles the processing of the protocol message recognizes the existence of the token in the message. This information is used in a call to the WebLogic Security Framework that results in the appropriate Identity Assertion provider being called to handle the verification of the token. It is the responsibility of the Identity Assertion provider implementation to perform whatever actions are necessary to establish validity and trust in the token, and the ability to provide the identity of the user with a reasonable degree of assurance, without the need for the user to re-authenticate to the application.

Summary

WebLogic Server 7.0 provides a rich Single Sign-On environment on which enterprise applications can be built. The Single Sign-On capabilities reach far beyond those of any other J2EE application server to address the issues found in today's enterprise environments. The open WebLogic Security Framework allows the Single Sign-On capabilities to be integrated with the best-of-breed security vendors and allows for customization to meet an enterprise's ever-changing needs.