BEA WebLogic SIP Server®

Release Notes

Version 2.1
Revised: December 2, 2005

Contents

WebLogic SIP Server 2.1 Features and Changes

Welcome to BEA WebLogic SIP Server 2.1! WebLogic SIP Server™ integrates SIP Servlet technologies with J2EE 1.3 and other leading Internet standards to provide a reliable framework for developing highly available, scalable, and secure telecommunications applications. WebLogic SIP Server's seamless integration of disparate, heterogeneous platforms and applications enables your network to leverage existing software investments and share the enterprise-class services and data that are crucial to building next-generation telephony applications.

The following sections describe the new features and changes made in the WebLogic Server 2.1 general release and in intermediate releases:

What's New in WebLogic SIP Server 2.1

This section details major differences between WebLogic SIP Server 2.1 and earlier versions. This section includes information about the following:

Architectural Changes

The architecture of WebLogic SIP Server 2.1 has been dramatically improved to provide increased performance, higher availability, and flexibility in configuring available resources. The most visible change in architecture is in WebLogic SIP Server's use of two separate clusters, referred to as the engine tier and data tier, that can be sized independently of one another to increase the call throughput or availability of an installation. (Development installations and small production installations can also use a single server instance as needed.) See Overview of the WebLogic SIP Server Architecture for more information about these changes. See also Capacity Planning for WebLogic SIP Server Deployments for information about sizing each cluster to match the performance and reliability goals of your applications.

Warning: When you configure a domain with multiple engine and data tier servers, you must accurately synchronize all server system clocks to a common time source (to within one or two milliseconds) in order for the SIP protocol stack to function properly. See Configuring NTP for Accurate SIP Timers for more information.

Application Porting Guidelines

SIP Servlets developed for previous versions of WebLogic SIP Server must observe new coding practices and requirements in order to operate in the version 2.1 distributed environment:

- Applications should not create threads.
- All Servlets should be non-blocking.
- All Session data must be serializable.
- Servlets must use the `distributable` tag, in order to deploy them to a cluster of engine tier WebLogic SIP Server instances. The `distributable` tag is not required and is ignored if you deploy to a single combined-tier (non-replicated) WebLogic SIP Server instance.

These and other porting requirements are described in Requirements and Best Practices for WebLogic SIP Server Applications in *Developing SIP Servlets with WebLogic SIP Server*.

New Security Features

WebLogic SIP Server now supports the `P-Asserted-Identity` SIP header as described in RFC3325. See Trusted Host Forwarding with P-Asserted-Identity in *Configuring and Managing WebLogic SIP Server*.

WebLogic SIP Server now supports Client-Cert authentication as well as BASIC and Digest authentication. See Configuring Client-Cert-Based Authentication for SIP Applications in *Configuring and Managing WebLogic SIP Server*. Client-Cert authentication is disabled by default; the switch to enable is defined in `ClusterMBean` and `ServerMBean`.

WebLogic SIP Server 2.1 now includes an RDBMS (JDBC) Digest Identity Assertion provider as well as the LDAP provider included in previous versions. In addition, both the RDBMS and LDAP providers support reverse-encrypted passwords as well as clear-text and hashed password values. See Configuring Digest Authentication for more information.

Configuration Changes

The following sections describe changes in the way you configure and manage WebLogic SIP Server 2.1.

datatier.xml Changes (Formerly statetier.xml)

The configuration file used to define partitions and replicas in the data tier is now named `datatier.xml`. In addition, the main XML element defined in the file has changed to `data-tier` (formerly `state-tier`). The location of the data tier configuration file has also changed; both `datatier.xml` and `statetier.xml` are located in the *`DOMAIN_DIR`*`/sipserver/config` subdirectory, where *`DOMAIN_DIR`* is the root directory of the WebLogic SIP Server domain.

Load Balancer Configuration Changes

Load balancer addresses are no longer defined in the `sipserver.xml` configuration file. Instead, the load balancer address and port number must be defined in the **External Listen Address** and **External Listen Port** fields of a SIP channel on each engine tier server. See Configuring Load Balancer Addresses in *Configuring and Managing WebLogic SIP Server*.

Changes in Queue Length-Based Overload Protection

When using queue- length-based overload protection controls, WebLogic SIP Server now monitors the *sum* of the lengths of the `sip.transport.Default` and `sip.timer.Default` execute queues, rather than only the length of `sip.transport.Default`. Also, the default overload configuration initiates overload protection when the combined queue size reaches 200 simultaneous requests, and releases overload control when the combined size falls to 150 simultaneous requests. See overload in *Configuring and Managing WebLogic SIP Server*.

sipserver.xml Changes

The schema for `sipserver.xml` has changed in WebLogic SIP Server. See the Engine Tier Configuration Reference (sipserver.xml) in *Configuring and Managing WebLogic SIP Server*. Notable changes include:

- Proxy entries now use single-elements that include a full URI (rather than multiple elements for each portion of a URI).
- The terminology and XML elements for regulating incoming SIP calls has changed. See overload.

In addition to schema changes, the location of `sipserver.xml` has changed in this release. The `sipserver.xml` is included as part of the `sipserver` enterprise application that implements the SIP container features of WebLogic SIP Server. See Overview of WebLogic SIP Server Configuration and Management.

Network Configuration Using Channels

WebLogic SIP Server 2.1 no longer uses the (previously-deprecated) `connector` element in `sipserver.xml` for configuring network connections. Instead, all connections are defined using:

- The listen address and listen port for the WebLogic SIP Server instance, and
- All network channel resources configured for the WebLogic SIP Server instance.

Multiple network channels can be defined to support multiple transport protocols or to configure multiple network interfaces on multi-homed server hardware. See the Managing WebLogic SIP Server Network Resources in *Configuring and Managing WebLogic SIP Server*.

Access Logging Configuration Changes

Access logging is no longer configured by defining a filter in the `sipserver.xml` configuration file. See Enabling Access Logging in *Developing SIP Servlets with WebLogic SIP Server* for information about the new XML configuration elements.

Container Changes for send() Calls

In previous WebLogic SIP Server releases, if an application called the `send()` method within a SIP request method such as `doInvite()`, `doAck()`, `doNotify()`, and so forth, the SIP Servlet container immediately transmitted the `send()` call to the network. In WebLogic SIP Server 2.1, `send()` calls are instead buffered in the order in which they are called, and transmitted in order only after the control has returned to the SIP Servlet container.

Warning: Applications must not wait or sleep after a call to `send()`, because the request or response is not transmitted until control returns to the SIP Servlet container.

What’s New in WebLogic SIP Server 2.0 SP2

WebLogic SIP Server 2.0.2 introduces the following features:

- Digest Authentication is now supported using a new LDAP Digest Authentication Identity Asserter and a separate authorization provider. See Configuring Digest Authentication for WebLogic SIP Server.
- Deployment Descriptors in `weblogic.xml` now support mapping principal and role names to roles defined in `sip.xml`. See Securing SIP Servlet Resources.
- SNMP support is now enabled by default; no patch is necessary.
- Operator documentation for WebLogic SIP Server SNMP Traps is now available. See Understanding and Responding to SNMP Traps.
- Administrator documentation is available for help in configuring Administration Servers in a WebLogic SIP Server domain. See Administration Server Best Practices in *Managing and Monitoring WebLogic SIP Server*.
- Developer documentation is available for using the Eclipse IDE to develop SIP Servlets with WebLogic SIP Server. See Developing SIP Servlets Using Eclipse.

Deprecated Features in WebLogic SIP Server 2.0 SP1

WebLogic SIP Server 2.0.1 is a restricted release of a BEA product and is subject to change in future releases. The following features are specifically called out as being deprecated in WebLogic SIP Server 2.0.1:

- Load balancer URIs will be obtained from Network Channel attributes, instead of `sipserver.xml`, in a future release of WebLogic SIP Server.
- Future releases of WebLogic SIP Server may not use the built-in WLSSecurityManagerFilter and custom security manager to implement digest authentication.
- `sipserver.xml` may not be used to configure SIP Servlet container features in future releases.
- The session backup implementation is deprecated and subject to change in future releases.

- Creating general-purpose SIP filters by extending a base filter class is deprecated in this release.

WebLogic SIP Server 2.1 Known Issues

The following table summarizes known issues and problems in WebLogic SIP Server 2.1.

Note: This section describes only those issues associated with the SIP Servlet container and data replication features of WebLogic SIP Server 2.1. See also the WebLogic Server 8.1 Known Issues for information about known problems with WebLogic Server 8.1, which provides the underlying OA&M and J2EE capabilities of WebLogic SIP Server 2.1.

Change Request Number	Description
n/a	The default timer queue configured for new WebLogic SIP Server instances, `sip.timer.Default`, contains only 3 threads. This number of threads yields poor performance for most application deployments. BEA recommends increasing the thread count of `sip.timer.Default` to 15 threads, and then profiling your applications to determine if additional or fewer threads are necessary. To change the thread count: 1) Access the Administration Console for your system. 2) Expand the Servers tab. 3) Right-click the name of the server to configure, and select View Execute Queues. 4) Select the `sip.timer.Default` queue from the table. 5) Increase the thread count to 15, and click Apply.
n/a	WebLogic SIP Server MIB objects are read-only. You cannot modify a WebLogic SIP Server configuration using SNMP.

Change Request Number	Description
n/a	This version of Weblogic SIP Server exhibits two behaviors that do not conform to the JSR 116 specification: • MIME content is returned as a String object, rather than as a `javax.mail.Multipart` object as encouraged by the specification. • `isPersistent`, used for re-instantiating `ServletTimer` after server restarts, is not implemented.
CR236491	If you attempt to install WebLogic SIP Server 2.1 on Fedora Core 3 or 4 with `selinux` running, the installer throws a `java.lang.UnsatisfiedLinkError` exception. You cannot install WebLogic SIP Server while `selinux` is active.
CR236688	If you configure two or more data tier replicas using the default WebLogic Server Listen Address configuration (which specifies no listen address), multiple data tier instances on the same machine cannot connect to one another. This problem occurs because, using the default Listen Address configuration, JNDI objects in the first booted server bind to all local IP addresses. To avoid this problem, always enter a valid IP address for each configured data tier server instance.
CR237223	In a WebLogic SIP Server installation with two engine tier nodes and two data tier nodes in a partition (two replicas), if the connection to the data tier becomes "split" such that each engine tier server can only reach a different data tier node, one of the replicas is forced offline. To recover from this situation, always configure the Node Manager utility to restart data tier replicas automatically when a replica fails. This enables the replica to rejoin its associated partition and update its copy of the call state data without having to manually restart the server.
CR239032	The **Tcp Connect Timeout Millis** attribute is applied only to SIP protocol channels. The timeout setting is ignored for channels configured for the SIPS protocol. See Managing WebLogic SIP Server Network Resources.

Change Request Number	Description

CR243700	WebLogic SIP Server does not persist session attributes after a Servlet makes a call to `setAttribute()`. For example, in the following code sample the call to `modifyState()` does not persist call state data in the data tier:
	`Foo foo = new Foo(..);`
	`appSession.setAttribute("name", foo); // This persists the call state.`
	`foo.modifyState(); // This change is not persisted.`
	As a workaround, ensure that Servlet code always modifies the call state *before* calling `setAttribute()`, as in:
	`Foo foo = new Foo(..);`
	`foo.modifyState();`
	`appSession.setAttribute("name", foo);`

CR244201	The Administration Console lists the `sipserver` implementation application as a standard J2EE application, and allows a Console user to redeploy or even remove the application from a running WebLogic SIP Server installation. The `sipserver` application must never be undeployed or redeployed except indirectly via the `ConfigManagerRuntimeMBean`. Redeploying the application yields several nested exceptions starting with `InstanceAlreadyExistsException`, and forces running data tier server instances to shut down. To avoid these problems, never redeploy or undeploy the `sipserver` application using the Administration Console or `weblogic.Deployer` utility. Perform all engine tier configuration changes using the SIP Servers node in the Console or the WLST command-line utility, as described in Configuring Engine Tier Container Properties.
CR244502	The Administration Console allowed you to uncheck the **Outbound Enabled** attribute for a SIP or SIPS network channel, even though SIP and SIPS network channels can always originate outbound connections. In addition, the Console allowed you to select the **HTTP Enabled for This Protocol** attribute for SIP and SIPS channels even though HTTP and SIP/SIPS are not supported on the same port number. The Console code was modified to make these attributes read-only for SIP and SIPS network channels.

Change Request Number	Description
CR245393	The WebLogic Server Administration Console has several problems that can affect the configuration of WebLogic SIP Server: • CR241785: The Console does not prevent a user from assigning null to attributes that require actual values. For example, when configuring the Digest authentication provider, the Console will persist null to the mandatory `DigestRealmName` attribute if none is specified, even though the server will fail to start with this configuration. If you experience this problem, you must manually edit the `config.xml` file to enter the mandatory value for `DigestRealmName`. • CR241822: The Console does not prevent a user from configuring multiple identity asserter providers that have the same active token type. Again, such a configuration prevents the server from starting, and you must manually edit `config.xml` to either remove a provider or assign unique token types to all providers. • CR241825: When you view the configuration page for the digest identity asserter provider, the Console always shows `PLAINTEXT` as the configured value in the drop-down menu for the Password Encryption Type attribute. `PLAINTEXT` is displayed even if you have previously configured the provider with an alternate encryption type, such as `REVERSIBLEENCRYPTED` or `PRECALCULATEDHASH`. Note that the Console does persist the actual value that you selected for the Password Encryption Type attribute when you click apply; ignore the displayed value when you return to the configuration page.

Change Request Number	Description
CR249459	When using a replicated WebLogic SIP Server installation, the default execute queue configuration for data tier servers poses a risk that garbage collection pauses in engine tier servers will cause delays in servicing other engine tier servers. To minimize this risk, on each data tier server set the thread count for the weblogic.kernel.Default queue to twice the number of engine tier servers in your deployment. You can set the thread count in the Administration Console by following these instructions: 1. Expand the Servers tab in the left pane. 2. Right-click the name of a data tier server and select View Execute Queues. 3. Click the weblogic.kernel.Default queue in the right pane. 4. Change the Thread Count attribute to equal twice the number of engine tier servers in y our system. 5. Click Apply. 6. Repeat the above instructions for each data tier server. Increasing the thread count in this manner minimizes the risk that garbage collection pauses in an engine tier server will delay service to other engine tier servers in the data tier.
CR252501	In the Administration Console, the Monitoring->General tab displays "Undefined" for the Active Application Session Count and Active SIP Session Count attributes when monitoring a replicated WebLogic SIP Server deployment. There is currently no workaround for this problem.
CR253622	If you define a `message-debug` element with the `level` set to "full" and you also specify the `-Dwlss.SipEngine` debugging option at startup, the server fails to start and displays a NullPointerException. To avoid this problem, either omit the debugging startup option or change the `message-debug level` element to "basic" and specify a valid `format` as described in Enabling Access Logging.

Resolved Problems in the WebLogic SIP Server 2.1

The following table summarizes the issues that were resolved in WebLogic SIP Server 2.1.

Change Request Number	Description
CR222494	The SIP Servlet container did not support `ejb-link` and `resource-ref` entries defined in the `sip.xml` deployment descriptor file. Instead the values had to be defined in `weblogic.xml` as a workaround. The code was modified to support these entries directly in `sip.xml`.
CR235377	Call overload controls were not enabled by default. This problem was address with a code fix.
CR236024	WebLogic SIP Server sometimes threw a NullPointerException when running a User Agent Client (UAC) against a proxy servlet that proxied back to the same engine tier server instance. The problem caused the exception: `<Client timer task failed with fatal status java.lang.NullPointerException at com.bea.wcp.sip.engine.server.SipServletMessageImpl.getDialogId(SipServletMessageImpl.java:274)` The problem was solved with a code fix.

Change Request Number	Description
CR236379	Configuration files used inconsistent naming conventions for the data tier and replicas within the data tier. The configuration file schema has changed to consistently use the term "data tier" to refer to the cluster of WebLogic SIP Server instances that manage call state data, "partition" to refer to a managed portion of the call state, and "replica" to refer to an individual WebLogic SIP Server instance within a partition. See Configuring Data Tier Partitions and Replicas and Data Tier Configuration Reference (datatier.xml).
CR236479	The SNMP MIB for WebLogic SIP Server was previously available only from Managed Servers running in a domain. The code was modified to make WebLogic SIP Server MIB entries available from the Administration Server as well as Managed Servers. See Configuring SNMP.
CR237487	WebLogic SIP Server did not listen for UDP messages on a non-default network channel that specified IP_ANY/0.0.0.0 as the listen address. The code was modified so that the server listens for incoming UDP messages on any IP interface when you define a network channel with 0.0.0.0 as the listen address. See Configuring Servers to Listen on Any IP Interface (0.0.0.0) in *Configuring and Managing WebLogic SIP Server*.

Change Request Number	Description

CR238527	When using the Hostpot 1.4.2_05 VM and running under heavy loads, the UDP NIO socket would sometimes fail with:
	`java.io.IOException: Interrupted system call`
	`at sun.nio.ch.PollArrayWrapper.poll0(Native Method)`
	`at sun.nio.ch.PollArrayWrapper.poll(PollArrayWrapper.java:100)`
	`at sun.nio.ch.PollSelectorImpl.doSelect(PollSelectorImpl.java:64)`
	`at sun.nio.ch.SelectorImpl.lockAndDoSelect(SelectorImpl.java:59)`
	`at sun.nio.ch.SelectorImpl.select(SelectorImpl.java:70)`
	`at com.bea.wcp.sip.engine.connector.transport.UdpTransportModule.run(UdpTransportModule.java:413)`
	The recovery from this failure caused the failure to lose 10 seconds of network traffic. This problem was resolved with a code fix.

CR239030	The previously deprecated XML configuration elements for defining trusted hosts have been replaced with new configuration elements. See sip-security in *Configuring and Managing WebLogic SIP Server*.

<table>
<tr><th>Change Request Number</th><th>Description</th></tr>
<tr><td>CR239250</td><td>In a replicated environment, or in a single server environment with debugging turned on, adding sleep time at the end of a <code>doMessage()</code> call could result in the error:

<code><Error> <WLSS.Session> <BEA-331410> <Invalid CSeq header. request=NOTIFY</code>

<code>sip:8005551212@172.17.24.251;appsessionid=app-nnw8zlr1voya:840a17c2649c84d0a47f06f1d7062cd2%40172.17.24.251;pxxx=12341234,</code>

<code>CSeq header=1 NOTIFY, CSeq number in this dialog=1></code>

This problem was resolved with a code fix.</td></tr>
<tr><td>CR240087</td><td>When waiting for over 60 minutes between an INVITE and a BYE message, a load testing proxy application would sent a 481 response even though the call should not be stateful. For example:

<code>2005-08-22 14:12:51: Aborting call on unexpected message for Call-ID</code>

<code>'1-8415@10.32.4.213': while expecting '200' response, received 'SIP/2.0 481</code>

<code>Call/Transaction Does Not Exist</code>

<code>To: testuser <sip:proxy@10.32.4.213:5060>;tag=1</code>

<code>Content-Length: 0</code>

<code>CSeq: 2 BYE</code>

<code>Call-ID: 1-8415@10.32.4.213</code>

<code>Via: SIP/2.0/UDP 10.32.4.213:5061;branch=z9hG4bK-1-6</code>

<code>From: sipp <sip:sipp@10.32.4.213:5061>;tag=1</code>

<code>Server: BEA WebLogic SIP Server 2.1.0.0</code>

This problem was resolved by adding a new container configuration parameter, <code>default-behavior</code>, which defines wether WebLogic SIP Server should act as a proxy or a user agent (UA) in the absence of an available, matching application. See default-behavior in Configuring and Managing WebLogic SIP Server.</td></tr>
</table>

Change Request Number	Description
CR240670	Prior to version 2.1, a WebLogic SIP Server 2.1 engine tier server would start up even if no SIP network channels were targeted to the server (for example, if a new engine tier server was configured manually and no channels were created). The code was changed so that engine tier servers now throw an exception and fail to start if no SIP channels have been configured for the server. The new error message is: `<Error> <WLSS.Engine> <BEA-330075> <There are no sip channels targeted to server "`*`servername`*`">`
CR241600	The previous version of the `findme` example application did not work in a domain having multiple engine tier servers in a cluster. The example code and documentation were modified to support a clustered environment. See Build the Example.

Resolved Problems for Service Pack 2

The following table summarizes the issues that were resolved in WebLogic SIP Server 2.0.2.

Change Request Number	Description
CR211125	The product license file, sip-license.xml, was moved to the WebLogic SIP Server product directory (*BEA_HOME*/wlss202).
CR217316	The code was modified so that SIP message Via, Contact, and Record-Route headers can be populated with the correct IP address on multihomed machines. See Setting Up Connectors.
CR218114	WebLogic SIP Server generated an exception if an INVITE request contained a Route header having the server's IP address. The code was modified to remove the Route header and forward the request.
CR218136	If a SIP request did not contain a Max-Forwards header, WebLogic SIP Server would throw a javax.servlet.sip.TooManyHopsException and respond with code 483. This behavior did not match the specified behavior, which is to decrement the Max-Forwards value by one of the header is present. The code was changed to match the specified behavior.
CR218285	WebLogic SIP Server threw an exception if a From address contained a "<" or ">" symbol. The code was modified to address this problem.
CR218359	The code was modified to support configurable SIP timers. See Configuring SIP Timers.

Change Request Number	Description
CR219912, CR221880	WebLogic SIP Server did not observe the Record-Route header for BYE or ACK messages. Instead, BYE and ACK messages were sent using the Contact header, ignoring the record-route hierarchy. This problem was solved with a change to the code.
CR221952	WebLogic SIP Server did not generate an error if it could not bind to a configured UDP port (for example, if the port was already in use). The code was modified to generate an appropriate message when the server cannot bind to a configured port.
CR224690	To improve proxy performance, the code was modified to first use UDP with a fixed buffer size, and then switch to TCP only if the message size exceeds the MTU size. This code fix also improves compliance with the SIP specification, because responses that exceed the configured MTU size are no longer rejected, but are instead sent over UDP if the original Request was via UDP.
CR231206	When forwarding messages having Route header values and a proxyTo() destination, WebLogic SIP Server forwarded to the proxyTo() destination rather than the first Route value. The code was changed to comply with the SIP specification; Route headers are now handled based on the transaction user in question and on whether a dialog is established.
CR231208	WebLogic SIP Server could potentially proxy a CANCEL request if a SIP application failed to implement a doCancel() method but the doRequest() method proxied the CANCEL. The code was modified to ensure that CANCEL requests are not proxied even if the SIP application attempts to proxy them in doRequest().
CR231821	WebLogic SIP Server returned an ArrayIndexOutOfBounds exception when 0 bytes were read from a stream. This issue was resolved with a code fix.

Change Request Number	Description
CR231849	WebLogic SIP Server 2.0.2 includes a “no-op” authentication provider, called the Identity Assertion Authenticator, that performs neither group population nor user existence checking. You can configure this provider and use it with the Digest Identity Asserter provider when neither group population nor user existence checking is required, in order to save an additional round-trip connection to the LDAP server. See Configuring Digest Authentication for WebLogic SIP Server for more information.
CR231852	Trusted hosts can now be configured in order to bypass authentication for listed host addresses. See Configuring Trusted Hosts.
CR231857	A deadlock situation could occur when two messages were received by a B2BUA servlet at the same time and each leg tried to forward the response to the other leg. This issue was resolved with a code fix.
CR231887	Weblogic SIP Server did not consider DNS names in Route headers when proxying requests; if a Route header contained a DNS name that resolved to the IP address of WebLogic SIP Server itself, the Route header was not removed when proxying the request to its destination. The code was changed to resolve the DNS name in the Route header and remove the header if the name resolved to the IP address of the server. WebLogic SIP Server caches the IP address to avoid repeated DNS lookups.