



Micro Focus VisiBroker 8.5 SP5

Release Notes

Micro Focus
The Lawn
22-30 Old Bath Road
Newbury, Berkshire RG14 1QN
UK
<http://www.microfocus.com>

Copyright © Micro Focus 2009-2018. All rights reserved.

MICRO FOCUS, the Micro Focus logo, and Micro Focus product names are trademarks or registered trademarks of Micro Focus Development Limited or its subsidiaries or affiliated companies in the United States, United Kingdom, and other countries. All other marks are the property of their respective owners.

Revised 2018-06-27

Contents

Micro Focus VisiBroker 8.5.5 Release Notes	2
Installing VisiBroker	2
Before Installing SP5	2
Installing SP5	2
Platforms and Compilers	2
New Features	3
OpenSSL	3
Configuration Properties.....	3
Deprecated Features	4
Certicom Security Provider	4
IPv4-only Smart Agent	4
Unsupported Features	4
VisiBroker for .NET.....	4
User Documentation	4
Resolved Issues	5
Issues resolved in this Service Pack.....	5
Issues resolved in previous HotFixes.....	6
Updates and SupportLine	9
Further Information and Product Support	9
Disclaimer	9

Micro Focus VisiBroker 8.5.5 Release Notes

Installing VisiBroker

Before Installing SP5

This release updates VisiBroker 8.5. Before installing this Service Pack you must have VisiBroker 8.5 installed. If you do not have VisiBroker 8.5, you should download it and follow the instructions in the *Installation Guide*.

Installing SP5

To install this release on top of an existing VisiBroker 8.5 installation:

1. Download the release archive to your VBROKERDIR folder.
2. Unpack the archive in the same folder.
3. Restart the application.

Platforms and Compilers

For a full list of platforms and compilers supported by VisiBroker 8.5 SP5, see <http://supportline.microfocus.com/prodavail.aspx>

JDK 8 on Solaris

Note that the Oracle JDK 8 for a Solaris (SPARC or x64) platform supports only a 64-bit JRE. In order to run a VisiBroker 32-bit product in a 64-bit environment, you will need to install a valid 32-bit Java JRE.

New Features

This release provides enhancements in the following areas.

OpenSSL

OpenSSL v1.0.2n is now supported.

Configuration Properties

The new properties `vbroker.se.iiop_tp.scm.iiop_tp.listener.useSelectorPool` and `vbroker.se.iiop_tp.scm.iiop_tp.listener.selectorMax` have been added by RPI 1111671. These were originally released with VisiBroker 8.5.4 HotFix 5, and have now been incorporated in to this service pack.

These properties modify the behavior of Java servers configured with the property `vbroker.se.iiop_tp.scm.iiop_tp.manager.type=Socket_nio`. See the section "Server-side thread pool IIOP_TP/IIOP_TP connection properties" in the ***VisiBroker for Java Developer's Guide*** for details.

With `vbroker.se.iiop_tp.scm.iiop_tp.listener.useSelectorPool` set, each IIOP (non-SSL) NIO Listener in the server will be constrained to consume a maximum of `vbroker.se.iiop_tp.scm.iiop_tp.listener.selectorMax` (default value: 20) instances of `java.nio.channels.Selector` at any given time. Note that setting this new property may cause threads to wait for a pooled Selector, so potential impact on performance should be carefully considered and profiled before setting the property.

If `vbroker.se.iiop_tp.scm.iiop_tp.listener.useSelectorPool` is not set, or is set to false, the number of Selectors in use will continue to be unbounded and to be some function of the number of concurrent connections for non-SSL NIO connections.

Note that there is no corresponding property `vbroker.se.iiop_tp.scm.ssl.listener.useSelectorPool`. SSL NIO connections continue to always pool Selectors, bounded by `vbroker.se.iiop_tp.scm.ssl.listener.selectorMax` (see the section "Security Properties for Java" in the ***VisiBroker Security Guide*** for this property).

Deprecated Features

Certicom Security Provider

The Certicom security provider was deprecated at VisiBroker 8.5 SP4. It is currently supported only for backwards compatibility, and Micro Focus recommends that users install the OpenSSL security provider.

IPv4-only Smart Agent

The current default IPv4-only Smart Agent (OSAgent) is deprecated. It will be replaced in a future release by the dual IPv6 and IPv4 communications Smart Agent. This dual Smart Agent is available for VisiBroker 8.5.5 as a separate update. Note that the current IPv4-only Smart Agent is not interoperable with the new dual IPv6/IPv4 Smart Agent. However the current IPv4-only Smart Agent will still be available as a patch in the future.

Unsupported Features

VisiBroker for .NET

VisiBroker for .NET has been discontinued, and references to it have been removed from the product and its documentation.

User Documentation

New documentation released with this Service Pack is available online, from <https://www.microfocus.com/support-and-services/documentation/>.

Service Pack Archives do not contain the updated documentation, so the documentation accessed from within the product for these versions is the legacy documentation from the VisiBroker 8.5 GA version. Any platforms that have a new installation since 8.5 (such as Windows 10, introduced at 8.5 SP3) will contain the documentation that was current at the time of introduction.

Resolved Issues

The resolved issues that customers have reported are listed in this section. The numbers that follow each issue are the Reported Problem Incident number followed by the Customer Incident Numbers (in parentheses). RPIs that have numbers only (and no text) are included to confirm that the RPIs have been fixed, since no further information is required.

Issues resolved in this Service Pack

This section includes issues that are resolved for the first time in this Service Pack.

- The VisiBroker Server program would coredump during shutdown. The core dump was caused by a thread race condition at IPC runtime. It has been fixed by a proper thread synchronization approach.

584680

- 584824
- 598766
- 630134
- The potential risk of running 32-bit product on Linux with a 64-bit inode file system such as XFS has been addressed.

630327

- 630899
- 633418
- 1084768 (2542508)
- Applications compiled with `-std=c++11` / `-std=c++0x` or higher could terminate abruptly under some circumstances when the application server or interceptor code raised exceptions. This no longer occurs.

Note that:

- You must recompile the user C++ application;
- The g++ default is `-std=c++14` from GCC 6 onwards.

1110025 (3112105)

- The user documentation has been updated to clarify that the value of `vbroker.orb.bufferCacheTimeout` is specified in seconds.

1113392 (3146268)

Issues resolved in previous HotFixes

This section includes issues that were fixed in HotFixes to VisiBroker 8.5 SP4, and are now incorporated into SP5.

- A faulty conditional statement existed within `CORBA_MarshalInBuffer::underflow()` where if the requested size (which itself is retrieved from the data stream) is sufficiently large, an integer wraparound could be caused, resulting in a reversal in the intended logic.

This has been fixed by refactoring that conditional statement to prevent such a possibility.

628364

- When reading a string value from a `VISistream`, it was possible for an integer overflow to occur relating to the length of the string before allocating the buffer for receiving it. String values are immediately preceded by a string length field within the stream. Once the string length has been read, `CORBA::string_alloc()` is called to prepare a heap buffer suitable for copying the received string value into. However, if the value of the string length field is read as `0xFFFFFFFF`, then `CORBA::string_alloc()`, which increments the requested size to allow for a NULL terminator, allocates a zero-length buffer. The subsequent copy of the received string data results in heap corruption.

This has been protected against by checking the string length field before the call to `CORBA::string_alloc()`. If its value is `0xFFFFFFFF (-1)`, a `CORBA::MARSHAL` exception is thrown.

628365

- Manipulated length field values within a serialised `VISServiceId` object could cause an Out of Bounds Read error during reconstruction of that object.

This has been protected against by range checking each length field value prior to it being used (during construction of `VISServiceId` from a `CORBA_MarshalInBuffer` stream).

628366

- VisiBroker Java servers configured with the server connection manager type set to `Socket_nio` might not release all possible file descriptors immediately that a client disconnected. These file descriptors could linger until some further IO event occurred, for example a new incoming client connection. This issue was caused by problems in the connection closure code. These problems have been fixed and the issue no longer occurs.

630692

- The VisiSecure OpenSSL implementation allowed a Client without a certificate to connect to a Server that is configured to NOT require authentication, but is additionally configured to require peer certificates. This has been fixed.

633230

- VisiBroker for Java applications could crash because of Out of Memory exceptions during CDR unmarshaling of arguments.

A new Java server property `vbroker.orb.cdrMaxInputBufferSize` has been created to specify the maximum allowed CDR input buffer size.

By specifying a value greater than zero, you can limit the upper size of the CDR input buffer on the Server. If the CDR input buffer size is greater than the specified property value, the server will throw a `CORBA.MARSHAL` exception and discard the CDR message.

If an invalid value is specified (less than zero), the Server will throw a `CORBA.BAD_PARAM` and exit.

If no value is specified, or if you specify a value of zero for the property, the previous behavior is preserved.

Be cautious when choosing a value for this property. Setting the value too small can result in the ORB rejecting important messages.

1104804 (2860816)

- The VisiBroker NameService (VisiNaming) could terminate unexpectedly when configured to use JDBC persistence with the latest Oracle database versions because it was using a previously deprecated Oracle API. See the section "*Desupport of oracle.jdbc.driver*" at <http://www.oracle.com/technetwork/testcontent/111070-readme-083278.html> for details of this API change. This has been corrected and the problem no longer occurs.

1108579 (2887414)

- The code dealing with idle connections in the Java ORB could occasionally under-perform by not reaping all potential expired connections because of an error in the code. This has been fixed.

1109211 (3101359)

- A `SocketTimeoutMonitor` thread was left running indefinitely, even after a call to `ORB.destroy()` on the Java ORB. This no longer happens. The thread will now terminate no later than the value of `vbroker.orb.socketTimeoutMonitorPeriod` after the successful completion of `ORB.destroy()`.

1109236 (3102222)

- Corrupt incoming CDR encoded messages containing strings with invalid very large string-length values could cause a `java.lang.OutOfMemoryError` in VisiBroker for Java. Incoming message string lengths are now checked for validity before any memory is allocated, and the problem no longer occurs.

Note: The IBM J9 Java 7 ORB has been identified to be a potential source of such corrupt messages. If, after deploying VisiBroker 8.5 SP5, you experience `MARSHAL_EXCEPTION` errors in processes using the IBM JDK ORB that are attempting to interoperate with VisiBroker, please contact Micro Focus technical support for advice, quoting this issue reference. Note that to date this problem has not been seen in the same circumstances if the IBM J9 Java 8 ORB is used instead.

1111359 (3126415)

- A new property has been added:

```
vbroker.se.iiop_tp.scm.iiop_tp.listener.useSelectorPool
```

See [Configuration Properties](#) for details.

1111671 (3128606)

Updates and SupportLine

Our Web site gives up-to-date details of contact numbers and addresses.

Further Information and Product Support

Additional technical information or advice is available from several sources.

The product support pages contain a considerable amount of additional information, such as:

- The WebSync service, where you can download fixes and documentation updates.
- The Knowledge Base, a large collection of product tips and workarounds.
- Examples and Utilities, including demos and additional product documentation.

To connect, enter <https://www.microfocus.com> in your browser to go to the Micro Focus home page.

Note: Some information may be available only to customers who have maintenance agreements.

If you obtained this product directly from Micro Focus, contact us as described on the Micro Focus Web site, <https://www.microfocus.com>. If you obtained the product from another source, such as an authorized distributor, contact them for help first. If they are unable to help, contact us.

Disclaimer

This software is provided "as is" without warranty of any kind. Micro Focus disclaims all warranties, either express or implied, including the warranties of merchantability and fitness for a particular purpose. In no event shall Micro Focus or its suppliers be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages, even if Micro Focus or its suppliers have been advised of the possibility of such damages. Some states do not allow the exclusion or limitation of liability for consequential or incidental damages so the foregoing limitation may not apply.

Micro Focus is a registered trademark.
Copyright © Micro Focus 2018. All rights reserved.