MICRO FOCUS

Micro Focus VisiBroker 8.5 SP4

Release Notes

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

Revised 2017-01-25

Contents

Micro Focus VisiBroker 8.5.4 Release Notes

Installing VisiBroker

Before Installing SP4

This release updates VisiBroker 8.5. Before installing this Service Pack you must have VisiBroker 8.5 installed.

Installing SP4

To install this release:

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

Operating Systems Supported

- Microsoft Windows 10
- Microsoft Windows 8
- Microsoft Windows 7
- Microsoft Windows Vista
- Microsoft Windows XP
- Microsoft Windows Server 2008 (R2) (Standard & Enterprise editions)
- Microsoft Windows Server 2012 R2
- Embarcadero C++ Builder XE for Windows
- Solaris 10.x (SPARC)
- Solaris 10.x (x86 and x64)
- Solaris 11.x (SPARC)
- Solaris 11.x (x86 and x64)
- Red Hat Enterprise Linux 5.x (x86 and x64)
- Red Hat Enterprise Linux 6.x (x86 and x64)
- Red Hat Enterprise Linux 7.x (x86 and x64)
- SUSE Linux Enterprise Server 10.x (x86 and x64)
- SUSE Linux Enterprise Server 11.x (x86 and x64)
- SUSE Linux Enterprise Server 12.x (x86 and x64)
- HP UX 11i v3/11.31 on Itanium
- AIX 6.x (32 or 64 bit)
- AIX 7.1 (32 or 64 bit)
- Montavista Linux CGE V4 (x64)

For a full list of supported platforms, 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 v1.0.2j is now supported. See RPI 1106802 for details.

ECDH Cipher Suites

This Service Pack includes RPI 1101164, which introduced support for ECDH cipher suites.

ECDH using ECC certificates signed by ECDSA

This Service Pack adds support for ECDH_ECDSA_* cipher suites. The supported cipher suites are:-

TLS v1.0

- TLS_ECDH_ECDSA_WITH_3DES_EDE_CBC_SHA
- TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA
- TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA

TLS v1.2

- TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256
- TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256
- TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384

To enable these cipher suites a private key and certificate chain must be provided that conform to the following requirements:-

- The identity certificate must contain an ECC public key that is enabled for ECDH usage.
- The identity certificate must be signed by a capable ECDSA certificate (intermediate or root).
- The complete certificate chain must be provided.
- The ECC private key that corresponds to the server's identity certificate must be provided.

ECDH using ECC certificates signed by RSA

This Service Pack adds support for ECDH_RSA_* cipher suites. The supported cipher suites are:-

TLS v1.0

- TLS_ECDH_RSA_WITH_AES_128_CBC_SHA
- TLS_ECDH_RSA_WITH_AES_256_CBC_SHA
- TLS_ECDH_RSA_WITH_3DES_EDE_CBC_SHA

TLS v1.2:

- TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256
- TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384

To enable these cipher suites a private key and certificate chain must be provided that conform to the following requirements:

- The identity certificate must contain an ECDSA-capable key.
- The identity certificate must be signed by a RSA certificate (intermediate or root) that is authorised for signing.
- The complete certificate chain must be provided.
- The ECC private key that corresponds to the server's identity certificate must be provided.

ECDHE using ECC certificates signed by ECDSA

This Service Pack adds support for ECDHE_ECDSA_* cipher suites. The supported cipher suites are:-

TLS v1.0:

- TLS_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA

TLS v1.2

- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384

To enable these cipher suites a private key and certificate chain must be provided that conform to the following requirements:-

- The identity certificate must contain an ECDSA-capable public key.
- The identity certificate must be signed by an ECDSA certificate (intermediate or root).
- The complete certificate chain must be provided.
- The ECC private key that corresponds to the server's identity certificate must be provided.

ECDHE using RSA certificates

This Service Pack adds support for ECDHE_RSA_* cipher suites. The supported cipher suites are:-

TLS v1.0:

- TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA

TLS v1.2

- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384

To enable these cipher suites a private key and certificate chain must be provided that conform to the following requirements:-

- The identity certificate must contain an RSA public key that is authorised for signing.
- The identity certificate must be signed by an RSA certificate (intermediate or root).
- The complete certificate chain must be provided.
- The RSA private key that corresponds to the server's identity certificate must be provided.

All other rules pertaining to certificate management within VisiBroker apply. Refer to the *VisiBroker Security Guide* for more information.

ECC Curves

OpenSSL v1.0.2j (as included in this Service Pack) supports many pre-defined ECC curves (also known as 'named curves' or 'elliptic curves"). A complete list of the available named curves can be obtained by executing the following command, using the v1.0.2j **openssl** utility (in this example, for Red Hat):

```
./openssl ecparam -list_curves
```

However, as part of an overall review of the cryptographic defaults with OpenSSL in response to the LogJam and FREAK attacks, default support for elliptic curves weaker than 256-bit was removed at version 1.0.2b. This Service Pack utilises this default behavior.

VisiBroker supports the well-known TLS elliptic curves as defined in the following IANA RFC:

- RFC4492: http://www.iana.org/go/rfc4492
- RFC7027: http://www.iana.org/go/rfc7027

You are free to select any of the curves from this list when generating your ECC keys/certificates. OpenSSL will treat keys generated using them transparently.

While you may use any of the supported curves, the following curves were selected for the purposes of verifying this functionality:

- brainpoolP512r1
- prime256v1
- secp256k1

ECDHE ephemeral keys

When an elliptic curve key is specified as part of the server's certificate and key configuration, ECDHE ephemeral keys are created on the same curve that was chosen for the server's primary key.

At VisiBroker 8.5 SP4, the list of supported curves that may be used with the `vbroker.security.server.socket.ecdheCurve` property is as follows:

- The following older curves, now insecure, are included only for backward compatibility:
  - `sect163k1`
  - `sect163r1`
  - `sect163r2`
  - `sect193r1`
  - `sect193r2`
  - `sect233k1`
  - `sect233r1`
  - `sect239k1`
  - `sect283k1`
  - `sect283r1`
  - `sect409k1`
  - `sect409r1`
  - `sect571k1`
  - `sect571r1`
  - `secp160k1`
  - `secp160r1`
  - `secp160r2`
  - `secp192k1`
  - `secp192r1`
- Secure curves:
  - `prime192v1`
  - `secp224k1`
  - `secp224r1`
  - `secp256k1`
  - `secp256r1`
  - `prime256v1`
  - `secp384r1`
  - `secp521r1`
  - `brainpoolP256r1`
  - `brainpoolP384r1`
  - `brainpoolP512r1`

Further curves may be added as the TLS specification changes in the future.

If no elliptic curve key is detected, the ECDHE_RSA cipher suites require that an elliptic curve is chosen to create the ECDHE temporary keys that will be used in the session. The ECDHE_RSA curve will be selected automatically. However you can also use the new server-side property `vbroker.security.server.socket.ecdheCurve` to set the curve that will be used for ECDHE cipher suites. The value of this property is a comma-separated list of curves, each of which must match one of the well-known elliptic curves as defined by IANA (the Internet Assigned Numbers Authority) for use with TLS.

- If one curve name is listed, that curve is used for all ECDHE_RSA keys.
- If more than one curve is listed, those named replace the default curved collection (listed under the next bullet point) that is used to produce a random choice of curve. Specifying multiple curves enables you to remove any that become deprecated in future.

Each curve in the list is checked at the time of the first connection and any invalid strings result in an EINVAL exception, with a logged message indicating the invalid string if the security module's logging is switched on.

- If no curve is set, the property defaults to using a random curve from the following list of the most secure curves:
  - `secp192r1` (also listed as `prime192v1`)
  - `secp224k1`
  - `secp224r1`
  - `secp256k1`
  - `prime256v1` (also listed as `secp256r1`)
  - `secp384r1`
  - `secp521r1`
  - `brainpoolP256r1`
  - `brainpoolP384r1`
  - `brainpoolP512r1`

Note: If `secp192r1` is specified, `prime192v1` will appear in the logs; these names indicate the same curve. Similarly, if `prime256v1` is specified, `secp256r1` will appear.

A previous ANSI X9.62 standard, *Public Key Cryptography For The Financial Services Industry*, defined some of the same curves as the IANA list, but with different names. Where this occurs either name can be used in the VisiBroker configuration.

For example:

- The IANA `secp192r1` is the same as the ANSI `prime192v1`
- The IANA `secp256r1` is the same as the ANSI `prime256v1`

Logjam Mitigation

Logjam (CVE-2015-4000) is a vulnerability in the TLS protocol which allows a man-in-the-middle attacker to downgrade vulnerable TLS connections using ephemeral Diffie-Hellman key exchange to 512-bit export-grade cryptography. In response to this threat, OpenSSL added mitigation for TLS clients by rejecting handshakes with non-ECC DH parameters shorter than 768 bits at version 1.0.2b. This restriction was increased to 1024 bits at version 1.0.2f, to offer stronger cryptographic assurance for all TLS connections using ephemeral Diffie-Hellman key exchange.

As a consequence of the above, server certificates containing non-ECC keys of less than 1024 bits will be rejected by a client when using DHE_* cipher suites.

Configuration Properties

- A new property, `vbroker.security.CSS.throw_ssl_exceptions`, has been added to deal with the issue described in RPI 623947. See the *Security Guide* for a full description.
- A new property, `vbroker.security.server.socket.ecdheCurve`, has been added to specify the elliptic curve to be used with ECDHE cipher suites, where no curve is specified by the certificate. See the sections *Security Properties for Java* and *Security Properties for C++* in the *Security Guide* for a full description.
- A new Smart Agent property, `vbroker.agent.verifyMaxClients`, has been added, as described under RPI 1086621. It specifies the maximum number of clients that the Smart Agent will try to verify on each occasion that it is woken up by `keepAliveTimer`. The default if not set is `50`. For example:

```
osagent -Dvbroker.agent.verifyMaxClients 23
```

See VisiBroker properties in both the ***VisiBroker for Java Developer's Guide*** and the ***VisiBroker for C++ Developer's Guide*** for a full description.

Deprecated Features

Certicom Security Provider

The Certicom security provider is 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.

User Documentation

New documentation released with this Service Pack is available online, from https://supportline.microfocus.com/productdoc.aspx.

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.

- 595546.
- 603011.
- 603536.
- 604502.
- The certificates distributed with the VisiSecure examples have been updated.

  609248
- 610722.
- A memory leak occurred in the C++ ORB when calling `ORB::string_to_object()` with a `file://`*URI* form of IOR. This leak could also be seen in related circumstances when using API functions that use `::string_to_object()` internally, such as `ORB::resolve_initial_references()`. This no longer occurs.

  622065
- At VisiBroker 8.5 SP3 HF05, VisiBroker introduced support for ECDHE_RSA ciphers (see RPI 1101164). However the ECC curve ('elliptic curve' or 'named curve') used on all ECDHE connections with no EC key present was fixed to always be the `Prime256` curve.

  At VisiBroker 8.5 SP4, when no EC curve is specified a curve is randomly chosen from a selection of the most secure curves. This curve may instead be defined by the new property `vbroker.security.server.socket.ecdheCurve`. See the section ECC Curves in New Features for more information.

  622998
- In VisiBroker 8.5 SP3, the permissions of the VisiBroker audit log files (`./var/mf_license.dat` and `./var/mf_licensej.dat`) could potentially be changed to levels that would make them inaccessible, causing subsequent servers to fail to start. This required users to reset their file permissions. With SP4 this user intervention is no longer required.

  1080353 (2528185)
- Added the new Smart Agent configuration property `vbroker.agent.verifyMaxClients`. This specifies the maximum number of clients that the Smart Agent will try to verify on each occasion that it is woken up by `keepAliveTimer`. The default if not set is `50`. See VisiBroker properties in both the

VisiBroker for Java Developer's Guide and the ***VisiBroker for C++ Developer's Guide*** for a full description.

1086621 (2594693)

- IPv6 support has been enabled on Solaris 10.x and 11.x.

  1093155 (2686288), 1097024 (2801563)

- The option "`vbroker.security.wallet.type`" now works correctly when passed a PKCS12 file.

  1097469 (2804749)

- Fixed an issue whereby the SSL connection between a Visi.Net 8.5 CSharpClient and VBC 8.5 SP2 Server using the OpenSSL provider would be abruptly closed by the server when it received a second request on the existing SSL connection.

  1098168 (2810033)

- Documentation is amended to specify that Oracle JDK 8 on Solaris (SPARC or x64) supports only a 64-bit JRE. See JDK 8 on Solaris for details.

  1103224 (2851588)

- Support for `ECDH_*` & ECDHE_* ciphers using TLS 1.2 has been extended to SUSE Linux versions 10, 11 and 12.

  1106067 (2834566)

- An error `org.omg.CORBA.BAD_PARAM: CSIV2 Protocol error: TSS responded with CompleteEstablishContext when client sent MessageInContext` could be seen on middle-tier secure Java VisiBroker servers. The error occurred in usecases that featured a very heavy concurrent load of multi-threaded TLS connection establishment, teardown, and re-establishment. This is now fixed.

  1106657 (2871530), 1107544 (2881809)

- VisiBroker 8.5.3 has been upgraded to use the OpenSSL security toolkit version 1.0.2j. See https://www.openssl.org/news/openssl-1.0.2-notes.html for more details.

  1106802 (2876333)

- A bug in the IDL compiler resulted in a `CORBA::UNKNOWN` exception being thrown from the generated Java code. This occurred when marshalling IDL-defined unions which have no explicit default case label, and a set of case labels that does not completely cover the possible values of the discriminant. This no longer occurs.

  1107717 (2882896)

Issues resolved in previous HotFixes

This section includes issues that were fixed in HotFixes to VisiBroker 8.5 SP3, and are now incorporated into SP4.

- Support for all cipher suites that include usage of the RC4 cipher has been removed in VisiBroker for C++.

  620867

- SSL transport connection errors could sometimes be incorrectly reported as:

  ```
  "org.omg.CORBA.BAD_PARAM: CSIV2 Protocol error: TSS did not respond
  with a SAS context vmcid: 0x0  minor code: 0  completed: No"
  ```

  This error can occur in usecases that featured a very heavy concurrent load of multi-threaded TLS connection establishment, teardown, and re-establishment. It is caused when a client tries to make a fresh request over a SSL connection that has been very recently closed by the connection idle and closure mechanism. The exception detail message has been changed to the form:

  ```
  "org.omg.CORBA.BAD_PARAM: CSIV2 Protocol error: TSS did not respond
  with an SAS context in case of NO_PERMISSION:
  org.omg.CORBA.NO_PERMISSION: SSLException:
  javax.net.ssl.SSLException: Connection has been shutdown vmcid: 0x0
  minor code: 0  completed: No"
  ```

  This description enables the root cause to be identified. A new configuration property has been added:

  `vbroker.security.CSS.throw_ssl_exceptions=true/false` (default false)

  The default is `false`. If this property is set to `true` then the exception thrown will instead be a `NO_PERMISSION`, as is the usual case for SSL exceptions in VisiBroker. The exception will then be of the form:

  ```
  "org.omg.CORBA.NO_PERMISSION: SSLException:
  javax.net.ssl.SSLException: Connection has been shutdown:
  javax.net.ssl.SSLException: java.net.SocketException: Broken pipe
  vmcid: 0x0  minor code: 0  completed: No"
  ```

  623947

- Certificates and private keys can now be supplied to the application in DER encoded format.

  1096539 (2695802), 608872

- IPv6 support has now been enabled for VisiBroker 8.5.3.

  1097024 (2801563); 1093155 (2686288)

- VisiBroker 8.5 now supports ECDH_* and ECDHE_* ciphers, using both TLS v.1.0 and TLS v.1.2 protocols. See ECDH cipher suites for more information.

  1101164 (2834566)

- The SSL Connection handshake timeout was not working. It now correctly takes the value of `vbroker.security.server.ssl.handshakeTimeout` and the problem no longer occurs.

  1103242 (2851199)

- `ORB.init` would crash if `java.net.InetAddress.getLocalHost()` threw an exception. This has been fixed by enabling the VisiBroker Java ORB to handle a `java.net.UnknownHostException` and to initialize using the IPv4 loopback address.

  To enable this change in behavior, specify the VisiBroker Java property `vbroker.orb.enableUnknownHost=true` in the set of properties being passed to the Java application.

  You must also specify a valid IP address using either of the following VisiBroker properties:

  - `vbroker.se.<srvr_eng_name>.host`
  - `vbroker.se.<srvr_eng_name>.proxyHost`

  If you are using the VisiBroker C++ Java launchers to start a Java application (on a system that cannot resolve a valid IP Address) then you must enable the `vbroker.orb.enableUnknownHost` property in the `vbj.config` file.

  1103550 (2852964)

- A race condition existed in the Server Request Interceptor whereby it was possible for a `vbsec::Subject` object that was held by `ServerConnectionContext::_connectionSubject` as an `auto_ptr`, and was pointed to by raw C++ pointers in either of:

  - the thread local storage area of the SharedCurrent object, or
  - a local variable within `CSIV2ServerReqInt::receive_request_service_contexts()`,

  to be deleted when `_connectionSubject` was updated in another thread. Any existing pointers to the original object were left dangling, and any attempt to dereference such a pointer resulted in a segmentation fault.

  This issue has been fixed by extending the lifetime of those objects that pass through `ServerConnectionContext::_connectionSubject` until the parent `ServerConnectionContext` object is destroyed.

  1103962 (2855521) and 1106543 (2854749)

- Java NIO SSL Thread Pool worker threads would raise repeated `java.nio.channels.ClosedByInterruptException` exceptions when the VisiBroker property `threadMaxIdle` was set to a small value. This has been fixed by correcting the logic that stops Idle threads which have been scheduled for removal.

  1104601 (2858108)

- The SSL Connection handshake timeout was not working. It now correctly takes the value of `vbroker.security.server.ssl.handshakeTimeout` and the problem no longer occurs on Windows systems.

  1104978 (2861275)

- Modified the behaviour of VisiNotify to handle the situation where an invalid CosNotify Filter criterion (only one operand) is added to VisiNotify by a consumer.

  1106725 (2875192)

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.