MICRO FOCUS

CORBA® An OMG Standard

Discover the Future of CORBA

Orbix 6.3.12

Release Notes

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

2021-05-07

Contents

Orbix 6.3.12 Release Notes

These release notes contain information about the Orbix 6.3.12 release from Micro Focus. They contain information that might not appear elsewhere in the documentation. Read them in their entirety before you install the product.

New Features and Enhancements

Orbix 6.3.12 includes the following new features and enhancements:

Linux C++11 ABI installer

Orbix now supports a new Linux C++11 ABI platform, which has been built against the C++11 ABI with GCC 8.

This new installer is supported on the following platforms:

- SUSE Linux Enterprise Server 15
- RedHat Enterprise Linux 8

IDL boolean type now maps to C++ bool type

Orbix 6.3.12 for Linux C++11 ABI maps the IDL boolean type to a typedef of the C++ inbuilt bool type in generated code and across all CORBA APIs. Previously, this type (CORBA::Boolean) was mapped to a typedef of an unsigned char, an implementation choice that dated from the period when the inbuilt bool type was poorly supported across compilers.

This change aligns Orbix 6.3.12 with both the OMG IDL-to-C++ Mapping v1.3 (https://www.omg.org/spec/CPP/About-CPP/) and also the IDL-to-C++11 Mapping v1.5 (https://www.omg.org/spec/CPP11/About-CPP11/) in this area.

IDL Basic Data Types are now mapped per the IDL-to-C++11 Language Mapping

Each IDL Basic Data Type in the table below is now mapped to the listed fixed size integral type as defined by the C99 header `<cstdint>`.

OMG IDL	CORBA C++ type	Previous Orbix for Linux versions (x86 / x64)	Orbix 6.3.12 for Linux C++11 ABI
`short`	`CORBA::Short`	`short`	`std::int16_t`
`long`	`CORBA::Long`	`long / int`	`std::int32_t`
`long long`	`CORBA::LongLong`	`long long`	`std::int64_t`
`unsigned short`	`CORBA::UShort`	`unsigned short`	`std::uint16_t`
`unsigned long`	`CORBA::ULong`	`unsigned long / unsigned int`	`std::uint32_t`
`unsigned long long`	`CORBA::ULongLong`	`unsigned long long`	`std::uint64_t`
`octet`	`CORBA::Octet`	`unsigned char`	`std::uint8_t`

TLS 1.3 support

Orbix now supports TLS 1.3. This is courtesy of an upgrade to `openssl` 1.1.1 for the C++ side, and of taking advantage of the features of the latest JDKs in Java8 and Java11.

TLS 1.3 is enabled by default for new deployments in both C++ and Java. If you are upgrading an existing Orbix 6 deployment, see "Upgrading and TLS 1.3".

Enhanced cipher suites

Orbix has added the following new cipher suites which are necessary for TLS 1.3 communication:

- TLS_AES_256_GCM_SHA384 (C++ and Java)
- TLS_AES_128_GCM_SHA256 (C++ and Java)
- TLS_CHACHA20_POLY1305_SHA256 (C++ only)
- TLS_AES_128_CCM_8_SHA256 (C++ only)
- TLS_AES_128_CCM_SHA256 (C++ only)

Note

Cipher suites supported on C++ only may be supported on Java as well, in future JDKs.

Updated default protocol list

The default protocol list now supports:

- TLS 1.3
- TLS 1.2
- TLS 1.1
- TLS 1.0

This list can be updated using configuration variables. For more information see the *Orbix 6.3.12 Security Guide*.

Updated default cipher suites

The default cipher list now includes the following ciphers:

- TLS_AES_256_GCM_SHA384
- RSA_WITH_AES_256_CBC_SHA256
- RSA_WITH_AES_256_CBC_SHA
- RSA_WITH_AES_128_CBC_SHA
- RSA_WITH_AES_128_CBC_SHA256

This list can be updated using configuration variables. For more information see the *Orbix 6.3.12 Security Guide*.

CORBA Compliance

Orbix 6.3 complies with the following specifications:

- CORBA 2.6
- GIOP 1.2 (default), 1.1, and 1.0
- C++ Language Mapping (formal/99-07-41)
- IDL-to-Java Language Mapping (formal/99-07-53)
- Object transaction service (OTS) 1.1 and 1.2

Platforms and Compilers

For a full list of platforms and compilers supported by Orbix 6.3.12, see the ***Orbix 6.3.12 Platform Support Notes*** (https://www.microfocus.com/documentation/orbix/orbix6312/orbix6.3.12_platform_support_notes.pdf).

Deprecated Features

The following features are deprecated. Deprecated features are supported for only backwards compatibility, and may be removed in a future release:

- Firewall Proxy Service

Firewall Proxy Service

The Firewall Proxy Service is deprecated as of Orbix 6.3.12.

Unsupported Features

The following features were no longer supported as of Orbix 6.3.11:

- Legacy cipher suites

Legacy cipher suites

Advances in cryptanalysis mean that many older encryption methods can no longer be considered secure, and cipher suites using such methods cannot be recommended and may even not be supported by recent JDK versions. Therefore, Orbix 6.3 no longer supports the following:

- Any null-encryption cipher suites
- Any export-strength cipher suites

See the ***Orbix 6.3.12 Security Guide*** for more information on cipher suites.

Migration from Previous Versions

To upgrade to Orbix 6.3.12 from existing Orbix 6.3.x installations:

- Back up existing installations before you upgrade.
- Go to the Orbix 6.3.x directory and run the Orbix 6.3.12 installer. The Orbix installer overwrites the existing version.
- If you experience any difficulty with running existing applications, see the following sections:
  - Upgrading and TLS 1.3
  - MIOP issues on Linux platforms
  - Java TLS 1.3 half-close and synchronization issue

For details on installing Orbix 6.3.x service packs, see the ***Orbix Installation Guide***. For details on migrating from earlier Orbix versions, see the migration and upgrade documentation at https://supportline.microfocus.com/productdoc.aspx.

Upgrading and TLS 1.3

Updating Orbix 6.3 to the new TLS 1.3 support settings by default may cause an issue when installing on top of an existing deployment, if no TLS 1.3 ciphers are enabled. As these TLS 1.3 ciphers are new and completely different from TLS 1.2 ciphers they did not exist previously and therefore were not present in Orbix 6.3.11 (or earlier versions) by default.

If a mismatch occurs, you may see a message similar to the following when re-starting your servers or services:

```
IDL:omg.org/CORBA/NO_PERMISSION:1.0: minor = 0x49540A41
(IT_IIOP_TLS:SPECIFIED_POLICIES_INCOMPATIBLE), completion status = NO
```

This means that you need to change either of the following configuration variables:

- If you wish to continue using your existing cipher suites and use TLS 1.2, then change your protocol variable setting as follows:

```
policies:mechanism_policy:protocol_version =
["TLS_V1_2"];
```

- If you wish to upgrade your cipher suites to use TLS 1.3, update your cipher suites variable to the new default value:

```
policies:mechanism_policy:ciphersuites =
["TLS_AES_256_GCM_SHA384",
"RSA_WITH_AES_256_CBC_SHA256",
"RSA_WITH_AES_256_CBC_SHA",
"RSA_WITH_AES_128_CBC_SHA",
"RSA_WITH_AES_128_CBC_SHA256"];
```

For more information on these variables and their possible settings, see the Orbix 6.3.12 *Security Guide*.

Known Issues

Orbix 6.3.12 may be affected by the following known issues:

Installer cannot load a valid Java virtual machine

When running the installer on Windows, you may encounter an error indicating that no valid Java virtual machine could be loaded. In this case, check that Java is properly installed and that the Java `bin` directory is on the system path. (Having the Java `bin` directory on a current user path is not sufficient, as the installer runs in elevated mode and might not inherit the user environment.)

Alternatively, you can use the installer's LAX_VM parameter to select a Java virtual machine which is present on the system. After the LAX_VM parameter, you need to specify the absolute path of the java executable. For example:

```
microfocus_orbix_6.3.12_win_64_vs2015.exe LAX_VM
"%JAVA_HOME%/bin/java.exe"
```

MIOP issues on Linux platforms

On newer Linux platforms, Orbix applications using the Multicast Inter-ORB Protocol (also called MIOP or EGMIOP) may not work correctly when initially installed because of the default restrictive network rules of the system firewall service.

Micro Focus recommend running the following command to configure the firewall service to permit MIOP messages:

```
firewall-cmd --permanent --zone=public
--add-protocol=udp
```

Java TLS 1.3 half-close and synchronization issue

Orbix 6.3.12 may be affected by a known JDK issue in which applications do not work well with the TLS 1.3 half-close policy. If, after migrating from TLS 1.2 to TLS 1.3, some of your applications hang awaiting further operations, then you are probably seeing this issue.

See https://bugs.openjdk.java.net/browse/JDK-8208526 for more details.

Java 11 and security

Java 11 also includes a number of security updates, and for older applications the current deployed security certificates may not work out of the box (for example, the key size, or signature algorithm strength may not satisfy the default security configuration of the JDK). Where feasible Micro Focus recommends upgrading any security certificates and cipher suites to work with the default security configuration in Java 11.

Using TLS with older JDKs

Older JDKs (JDK7 and versions of JDK8 before update 162) do not install the Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files by default.

These files are needed to support the SHA-256 ciphers which are recommended for greater security and as such are now enabled in Orbix by default, so Micro Focus recommends that you download and install these unlimited strength `.jar` files.

If for some reason you do not wish to install these files on top of older JDKs then you will need to make the following change:

1. Remove the `*256*` ciphers from their cipher suite lists, this would involve updating the `tls.xml` file in `$OrbixInstallDir/asp/6.3/etc/conf/tls.xml` for future deployments.
2. If you wish to deploy the secure configuration repository (CFR) and the security service, then in addition to the above change you must also add a cipher list which omits the stronger ciphers. Add this under the "`is2_inFile`" section in the same `tls.xml`.

For example:

```
<configData>
    <dataId>policies:mechanism_policy:ciphersuites</dataId>
    <dataType>list</dataType>
    <dataValue>RSA_WITH_RC4_128_SHA</dataValue>
    <dataValue>RSA_WITH_RC4_128_MD5</dataValue>
    <dataValue>RSA_WITH_DES_CBC_SHA</dataValue>
    <dataValue>RSA_WITH_3DES_EDE_CBC_SHA</dataValue>
    <dataValue>RSA_WITH_AES_128_CBC_SHA</dataValue>
</configData>
```

Microsoft Visual Studio 2015 compiler

Orbix 6.3.12 supports Microsoft Visual Studio 2015 version 14.0.25431.01 Update 3 or later versions. Support is not provided for earlier versions than Microsoft Visual Studio 2015 version 14.0.25431.01 Update 3, because of compiler issues discovered during testing.

Micro Focus advises customers to install Visual Studio 2015 using the Visual Studio 2015 Web installer (not the ISO installer) so as to avail themselves of the latest fixes available from Microsoft.

Older security algorithms with recent JDKs

Recent JDKs may by default disable the use of what are considered legacy algorithms. This includes older protocols, ciphers, digests and also may include insisting that key sizes used are above a certain size. This is due to progress in cryptanalysis which has rendered some of these older algorithms no longer strong enough.

Micro Focus highly recommends that any certificates used in secure Orbix applications that are signed using older functions, such as with an MD5 digest signature, are regenerated to use at least a SHA-2 digest signature.

Benign warning when launching the Windows installer

When installing Orbix 6.3.12 on Windows, the installer may issue a warning about a missing java.dll, or a registry key specifying the wrong version of Java. This is a benign warning and can be safely ignored.

This warning is issued because the installer does a thorough search across the system for a usable version of Java with which to launch the installer. When an incomplete installation is found, the warning may be issued.

Such an incomplete Java installation is typically a leftover install of Java JRE installed via the "Java Update" mechanism. In order to ensure that the installation is wiped properly, run the "Java Update" installer, and at the end of the installation, agree to uninstall previous older JRE installations.

Oracle Solaris Studio 12.4 and 12.5 compiler

Oracle Solaris Studio 12.4 and 12.5 compiler is not certified with Orbix 6.3.12. A compiler issue was uncovered while certifying Orbix 6.3.8 with Studio 12.4. The compiler issue relates to an inconsistent behavior in mangling symbol names between Studio 12.4 and earlier compiler versions.

The compiler vendor fixed this defect in Studio 12.6 and Orbix 6.3.12 is certified to work with Studio 12.6.

Micro Focus advises customers to refrain from using Oracle Solaris Studio 12.4 and 12.5 with Orbix 6.3.12.

Deployment problem on Windows 7 or Windows 2008 R2 VM on VMWare

You might encounter an intermittent failure to deploy services on virtual machines with only 1 CPU. This problem does not occur when the virtual machine has 2 or more CPUs.

Supported platforms for Actional

Integration with the Aurea Actional® Application Performance Monitoring system is supported by Orbix for all the supported Orbix platforms, and also for the following Windows platforms:

- Microsoft Windows Visual Studio 2008 (32 and 64-bit)
- Microsoft Windows Visual Studio 2010 (32 and 64-bit)

Spaces in install path and itant

If your Orbix installation path contains spaces, and you use the `itant` tool to build the Java demos, the following message might appear in the console output:

```
C:\Program%20Files\Micro%20Focus\Orbix\asp\6.3\demos\corba\demo
   .xml could not be found
```

This is a benign message and can be ignored. The Java demos build successfully.

Secure CFR domain with replicated services

In a secure configuration repository (CFR)-based domain with replicated Orbix services, CFR replica sets can not be automatically shrunk. This issue does not occur in an insecure CFR-based domain. If you have to remove CFR replicas in a secure CFR-based domain, please contact Orbix technical support.

Compiling 64-bit C++ applications

When compiling 64-bit applications with the C++ Sun Studio 12 Update 2 compiler on a Solaris x86 platform, there may be issues relating to compiling certain demos delivered with Orbix. The issue

relates to a known compiler bug in the C++ compiler. Oracle is aware of this issue, and as a workaround suggests compiling the code with the `-O1` flag instead of using the debug `-g` flag.

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.

- When deploying a domain descriptor for a CFR domain, properties specified in the deployment descriptor are now applied for all services.

  1119789 (3218151)

- The IDL compiler now rejects illegal fixed type definitions.

  Any fixed type definition where the scale is greater than the number of digits (such as typedef fixed<3, 5> idlFixed1;) is now flagged as an error by the IDL compiler.

  If you require the previous behavior where such a typedef is not flagged by the IDL compiler, use the IDL compiler -3 switch.

  644136

- If a fixed typedef was defined in one file, and a fixed type using that fixed typedef was defined in a second file, then when the C++ IDL compiler was run against that second file, the generated stub code could not be compiled.

  The IDL compiler has been fixed to address this issue and now generates code that can be compiled correctly.

  644134

Issues resolved in previous HotFixes

This section includes issues that were fixed in HotFixes to Orbix 6.3 SP11, and are now incorporated into SP12.

- The Orbix 6 IDL compiler has been enhanced such that it can ignore annotations that may appear in an IDL file. This change ensures that the IDL compiler can continue to compile IDL files as if they did not contain any IDL-RS annotations.

  641246

Other Resources

The following additional resources are available:

- The most up-to-date versions of Orbix technical documentation are available at: https://supportline.microfocus.com/productdoc.aspx

- The Orbix Knowledge Base is a database of articles that contain practical advice on specific development issues, contributed by developers, support specialists, and customers. This is available at: https://community.microfocus.com/microfocus/corba/orbix/w/knowledge_base/

- Contact Micro Focus technical support at: https://www.microfocus.com