

Fedora Troubleshooting



Some information might not apply to the latest versions of Fedora. We should group this information by version.

This page is intended to document problems we encounter with Fedora, hopefully resulting in a better understanding of Fedora functionality and debugging methods. As we identify explanations for certain types of behavior, they will be merged into the "main" documentation or turned into a more traditional troubleshooting list.

General troubleshooting procedure

The best place to look for error messages are:

1. The shell from which you started fedora
2. nohup.out (usually in the directory from which you started fedora, but may be in server or server/config)
3. startup.log (in the log directory)
4. other fedora logs (though they're usually useless)

Checking for startup/shutdown

In the "primary" fedora logs (the ones whose names are timestamps):

- When server is shut down, log file is closed out, and ends with "Server shutdown requested."
- When server is started, log file begins with "Server home is...."

Hanging process

The solution to a process that hangs is almost always to increase the Java heap space. Find the batch/shell script that your process is running, and change the -Xmx setting. For most applications, at least 512mb is desired.

Non-updating stylesheet

2005-12-19: [Ryan Scherle](#) made a minor change to the FULL_VIEW_XSL for the Hohenberger collection (iudl:10) on rhyme. This change did not take effect.

- Tried restarting Fedora, making another change to the stylesheet, restarting Fedora again. No effect.
- Stopped my local Fedora to make sure it wasn't referencing the wrong stylesheet.
- Manually constructed the equivalent URL, based on contents of the bmech. **This worked.**
- Purged the collection disseminator and re-added it. This disseminator no longer shows up in the method list!
- Updated the stylesheet datastream in my local Fedora, to see if behavior was the same. **This worked.** (Which was different behavior than before.) The only difference was originally, I hit the Import button to update the stylesheet. The second time, I hit Edit and then Import.
- Purging the collection object and re-importing from my local Fedora fixed the behavior on rhyme.

Server not responding

2005-12-19 (ca. 10:40pm): Nagios reported the Fedora server on rhyme was critical. It appears that the server simply stopped responding, and as of 9am on the 20th, is still not responding.

- There are two nohup.out files, one in the home directory for the fedora user, and one in the server/config directory. These are not much help, as they don't include timestamps. The file in the home directory doesn't have any useful information, as the most recent entry is a known user-caused error from earlier in the day. The other file ends with some NullPointerExceptions, all in regular expression classes.
- The main fedora server log shows no errors. There is an entry at 10:33pm with the last successful Nagios query. The next entry is at 9:07am on the 20th (a manually run query to try diagnosing the problem), and shows 0 available connections in the database connection pool.
- There was nothing interesting in the tomcat logs.
- Randall reports that the Oracle instance was backing up at that time, which likely caused the problem.
- Restarting fedora fixed the problem.

Resource Index problems

2006-1-19: Noticed that (on rhyme) there were some Hohenberger objects that did not get listed in an RI search for all Hohenberger items.

1. Tried the Fedora rebuilder, but this only resulted in further corruption of the database. It turns out that the currently-released rebuilder doesn't work with Oracle.
2. Got access to the main Fedora CVS repository, and built a new copy of the rebuilder. Unfortunately, the new rebuilder is somewhat tied into the other code, so tried bringing all the code up to 2.1.
3. Even though the new RI rebuilder works fine with Oracle, the DB rebuilder doesn't, so the database corruption wasn't fully removed.
4. Reverted all code back to 2.0.
5. Fixed RI on rhyme by moving all content to another machine and then back to rhyme.

Can't import object

If an object cannot be imported from one repository to another, the known possibilities are:

1. Incorrect password.
2. An extremely long error message, complaining of `HttpServiceNotFoundException:[DefaultExternalContentManager]` and a "non-200 response code (500)". This means that the server-side `beSecurity.xml` file (or the XACML files it generates) doesn't have the correct IP address for the server.
3. Too many objects are being ingested at once. The ingest process has some memory leaks. It is better to break the ingest into batches of around 500 objects. (Fedora 2.2 is supposed to make this easier.)

Fedora won't start

ModuleInitializationException / Connection refused

```
fedora.server.errors.ModuleInitializationException: Error while attempting
to check for and create non-existing table(s): java.sql.SQLException:
Server connection failure during transaction. Due to underlying exception:
'java.net.SocketException:java.net.ConnectException: Connection refused: connect'.

** BEGIN NESTED EXCEPTION **

java.net.SocketException
MESSAGE: java.net.ConnectException: Connection refused: connect

STACKTRACE:

java.net.SocketException: java.net.ConnectException: Connection refused:
connect at com.mysql.jdbc.StandardSocketFactory.connect(StandardSocketFactory.java:156)
```

This usually means you are trying to start Fedora with the wrong database profile (and it's trying to connect to the wrong database). Make sure you run "fedora-start *dbprofile*" where *dbprofile* is usually `oracle` for the large Fedora instances, or `mckoi` for smaller test Fedora instances.

ModuleInitializationException / DefaultAuthorization

```
fedora.server.errors.ModuleInitializationException at
fedora.server.security.DefaultAuthorization.postInitModule(DefaultAuthorization.java:390)
```

This usually means there is something wrong with the database or resource index:

1. Make sure the database is running.
2. Use "fedora-rebuild *dbprofile*" to rebuild the database.
3. Try starting Fedora again. If you get the same error, use "fedora-rebuild *dbprofile*" to rebuild the resource index.

Programmatic Issues:

Concurrency Exceptions:

The following exceptions:

- `fedora.server.errors.StorageDeviceException`
 - `fedora.server.errors.GeneralException`
- have been seen with the message referring to a db concurrency error.

Furthermore, the concurrency issues seem to be interfering with the purging of old items. A purge will seem to complete normally, only to have the item remaining in the repository, causing the following exceptions:

- `fedora.server.errors.ObjectExistsException`
- `fedora.server.errors.ObjectAlreadyInLowLevelStorageException`

That last exception is when the item is gone, but the datastreams for that object still exist. This indicates that when the purge fails (presumably from a concurrency error), the transaction is not properly rolled back.

Details at 11:00.

Unicode issues

If there are problems with Unicode, it is best to ensure both the Fedora server and the client are using UTF-8. This can be accomplished by adding the `URIEncoding="UTF-8"` attribute to the Connector element(s?) in `server_fedoraTemplate.xml` in Fedora's Tomcat/conf directory, as well as to `server.xml` (where?). For clients, add `request.setCharacterEncoding("UTF-8")` where needed.

Fedora Disseminator/Datastream Binding key name

When writing code to create a disseminator, there is a structure used to bind a datastream to a disseminator, which looks something like this:

```
dsb = new DatastreamBinding[]
{
    new DatastreamBinding(
        "NULLBIND",
        "Binding to DC.",
        "DC",
        "0")
};

dsbm = new DatastreamBindingMap(
    "bdef:iudlPagedImage",
    "bmech:iudlPagedImage",
    "PagedImage specific behaviors",
    StateType.ACTIVE.toString(),
    dsb);

diss = new Disseminator("bdef:iudlPagedImage",
    "bmech:iudlPagedImage",
    FedoraItem.GetFedoraDateTime(),
    "DISS_PAGED",
    "PagedImage specific behaviors",
    "DISS_PAGED.0",
    dsbm,
    StateType.ACTIVE.toString());

foxml.addDisseminator(diss);
```

Notice the "NULLBIND", above. This is not typical; typically, the "NULLBIND" will be the name of the datastream, which is "DC" in this case. This is an example of where the binding is only because Fedora requires at least 1 DS binding for a disseminator, even if it is not really needed.

In any case, the "NULLBIND" name is required by the bmech, and if the name is wrong, a non-obvious error will occur. In this example, I inadvertently used "DC" instead of "NULLBIND" in the code, and received the following error from the Ingest Tool output summary:

```

[2006-05-02T15:38:34.475] [ERROR] Errors during ingestion:
AxisFault
faultCode: {http://schemas.xmlsoap.org/soap/envelope/}Server.userException
faultSubcode:
faultString: fedora.server.errors.GeneralException: Unable to add or modify object (commit canceled)
faultActor:
faultNode:
faultDetail:
    {http://xml.apache.org/axis/}hostname:bl-ldlp-mallow

fedora.server.errors.GeneralException: Unable to add or modify object (commit canceled)
    at org.apache.axis.message.SOAPFaultBuilder.createFault(SOAPFaultBuilder.java:221)
    at org.apache.axis.message.SOAPFaultBuilder.endElement(SOAPFaultBuilder.java:128)
    at org.apache.axis.encoding.DeserializationContext.endElement(DeserializationContext.java:1087)
    at org.apache.xerces.parsers.AbstractSAXParser.endElement(Unknown Source)
    at org.apache.xerces.impl.XMLNSDocumentScannerImpl.scanEndElement(Unknown Source)
    at org.apache.xerces.impl.XMLDocumentFragmentScannerImpl$FragmentContentDispatcher.dispatch(Unknown
Source)
    at org.apache.xerces.impl.XMLDocumentFragmentScannerImpl.scanDocument(Unknown Source)
    at org.apache.xerces.parsers.XML11Configuration.parse(Unknown Source)
    at org.apache.xerces.parsers.XML11Configuration.parse(Unknown Source)
    at org.apache.xerces.parsers.XMLParser.parse(Unknown Source)
    at org.apache.xerces.parsers.AbstractSAXParser.parse(Unknown Source)
    at javax.xml.parsers.SAXParser.parse(SAXParser.java:375)
    at org.apache.axis.encoding.DeserializationContext.parse(DeserializationContext.java:227)
    at org.apache.axis.SOAPPart.getAsSOAPEnvelope(SOAPPart.java:696)
    at org.apache.axis.Message.getSOAPEnvelope(Message.java:424)
    at org.apache.axis.handlers.soap.MustUnderstandChecker.invoke(MustUnderstandChecker.java:62)
    at org.apache.axis.client.AxisClient.invoke(AxisClient.java:206)
    at org.apache.axis.client.Call.invokeEngine(Call.java:2765)
    at org.apache.axis.client.Call.invoke(Call.java:2748)
    at org.apache.axis.client.Call.invoke(Call.java:2424)
    at org.apache.axis.client.Call.invoke(Call.java:2347)
    at org.apache.axis.client.Call.invoke(Call.java:1804)
    at edu.indiana.dlib.fedora.apim.FedoraAPIMBindingSOAPHTTPStub.ingest(FedoraAPIMBindingSOAPHTTPStub.java:
654)
    at edu.indiana.dlib.fedora.FedoraUtil.ingestObject(FedoraUtil.java:562)
    at edu.indiana.dlib.fedora.server.ingestTool.IngestToolWorker.ingest(IngestToolWorker.java:831)
    at edu.indiana.dlib.fedora.server.ingestTool.IngestToolPagedProcessor.ingestPagedData
(IngestToolPagedProcessor.java:396)
    at edu.indiana.dlib.fedora.server.ingestTool.IngestToolWorker.run(IngestToolWorker.java:539)
    at edu.indiana.dlib.fedora.server.IngestTool.doIngestData(IngestTool.java:411)
    at edu.indiana.dlib.fedora.server.IngestTool.doGet(IngestTool.java:134)
    at javax.servlet.http.HttpServlet.service(HttpServlet.java:689)
    at javax.servlet.http.HttpServlet.service(HttpServlet.java:802)
    at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:252)
    at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:173)
    at org.apache.catalina.core.StandardWrapperValve.invoke(StandardWrapperValve.java:213)
    at org.apache.catalina.core.StandardContextValve.invoke(StandardContextValve.java:178)
    at org.apache.catalina.core.StandardHostValve.invoke(StandardHostValve.java:126)
    at org.apache.catalina.valves.ErrorReportValve.invoke(ErrorReportValve.java:105)
    at org.apache.catalina.core.StandardEngineValve.invoke(StandardEngineValve.java:107)
    at org.apache.catalina.connector.CoyoteAdapter.service(CoyoteAdapter.java:148)
    at org.apache.coyote.http11.Http11AprProcessor.process(Http11AprProcessor.java:831)
    at org.apache.coyote.http11.Http11AprProtocol$Http11ConnectionHandler.process(Http11AprProtocol.java:
639)
    at org.apache.tomcat.util.net.AprEndpoint$Worker.run(AprEndpoint.java:1203)
    at java.lang.Thread.run(Thread.java:595)

```

Examination of the FoXML and the Mets MD will show that both are well-formed and valid.

Also, you will probably see something like this in the Fedora logs:

```

<fine time="1146660751629" thread="15">fedora.server.management.UploadServlet: Successful upload,
id=uploaded://447</fine>
<fine time="1146660751644" thread="15">fedora.server.management.UploadServlet: Successful upload,
id=uploaded://448</fine>
<info time="1146660751957" thread="17">fedora.server.storage.DefaultDOManager: COMMIT:
  Retrieving ManagedContent datastream from internal uploaded location: uploaded://448</info>
<info time="1146660752082" thread="17">fedora.server.storage.DefaultDOManager: COMMIT:
  Replacing ManagedContent datastream with internal id: iudl:11978+OVERVIEW+OVERVIEW.0</info>
<info time="1146660752082" thread="17">fedora.server.storage.DefaultDOManager: COMMIT:
  Retrieving ManagedContent datastream from internal uploaded location: uploaded://447</info>
<info time="1146660752300" thread="17">fedora.server.storage.DefaultDOManager: COMMIT:
  Replacing ManagedContent datastream with internal id: iudl:11978+METADATA+METADATA.0</info>
<info time="1146660752316" thread="17">fedora.server.storage.DefaultDOManager: COMMIT:
  Deleting ManagedContent datastream. id: iudl:11978+OVERVIEW+OVERVIEW.0</info>
<info time="1146660752410" thread="17">fedora.server.storage.DefaultDOManager: COMMIT:
  Deleting ManagedContent datastream. id: iudl:11978+METADATA+METADATA.0</info>
<warning time="1146660752582" thread="17">fedora.server.storage.DefaultDOManager: COMMIT:
  Object wasn't found in permanent low level store, but that might be ok...continuing with purge.</warning>
<info time="1146660752941" thread="17">fedora.server.storage.DefaultDOManager: COMMIT:
  Deleting from FieldSearch indexes...</info>
<info time="1146660752941" thread="17">fedora.server.storage.DefaultDOManager: COMMIT:
  Deleting from ResourceIndex...</info>
<info time="1146660755097" thread="17">fedora.server.storage.DefaultDOManager: COMMIT:
  Finished deleting from ResourceIndex...</info>
<warning time="1146660755097" thread="17">Error carried up to API-M level:
fedora.server.errors.GeneralException: Unable to add or modify object (commit canceled)
  at fedora.server.storage.DefaultDOManager.doCommit(DefaultDOManager.java:1113)
  at fedora.server.storage.SimpleDOWriter.commit(SimpleDOWriter.java:342)
  at fedora.server.management.DefaultManagement.ingestObject(DefaultManagement.java:172)
  at fedora.server.management.FedoraAPIMBindingSOAPHTTPImpl.ingest(FedoraAPIMBindingSOAPHTTPImpl.java:136)
Caused by: java.lang.NullPointerException: (no detail provided)
</warning>

```

Examination of that log might seem to indicate that the failure (the first warning) is referring to the uploaded METADATA DS. However, it is not – Fedora has finished with the upload processing, and is now referring to the improper reference in the datastream binding. This failed reference in the DS binding causes the later GeneralException which proves fatal to the ingest.

It would be nice to see better errors from Fedora for this particular issue.

401 when calling management API (Version 2.1.1 and below)

Ref: http://www.fedora.info/bugzilla/show_bug.cgi?id=144

Removed the JAAS-realm from the server_config template and restarted. Seems to be working. This issue is said to be fixed in Fedora 2.2.

Fedora Admin Client: Unconnected sockets not Implemented Exception (Version 2.2.1)

Symptom: The Fedora Admin Client refuses to add a new "Internal XML" or "Managed" datastream with error message "Unconnected sockets not implemented" or similar.

Configuration: Fedora APIM and the upload servlet is over SSL encrypted connection. A self signed certificate is used, this shouldn't happen if the security certificate is signed by a known authority (such as Verisign). Fedora Version 2.2.1, JDK Version 5.

Description: Normal APIM operations (over SOAP) do not require the client to specify the correct password to the certificate file. However, the upload operation requires the correct password to be specified (for example, in fedora-admin.bat). I'm not sure why we don't run into the same issue in the ingest tool because we don't specify a password there. It might be the case that if a password is specified, it should match the certificate password.

Fix: Modify the default password to the truststore file in the fedora-admin.bat.