How do I trace the Service Manager Web Tier from Browser via Web Application Server to Service Manager Server?

Solution

Where a complete end to end trace of the Service Manager Web Tier is needed use the following steps:

**Stop the server**

**Data To Collect:**

- Time and date of issue occurence.  
- Exact and unambiguous steps to reproduce.  
- The login name of the user executing the test.  
- Screenshot of the issue.  
- Expected behaviour.  
- Actual behaviour including exact errors, screenshots and any JavaScript errors in the browser.  
- All sm logs (so serverside\_webclient.log for debug sm servlet container, sm\_webtier\_debug.log will be the web tier trace as defined in log4j.properties).  
- All web application server and web server logs.  
- The Fiddler 2 trace collected.   
- sm.ini  
- sm.cfg  
- web.xml  
- Web server and web application server confirguration files  
  
**Steps to configure the web tier trace:**

1) Configure the Web Tier:  
a) Locate the web.xml file in the Web Application Server installed applications directory.  
    For Example: webapps\sc\WEB-INF\web.xml  
b) Open the file with a text editor  
c) Search for the parameter name honorUrlPort  
d) Ensure that the value for this parameter is set to true (The value should be directly below the parameter name)  
e) Copy the debug version of the log4j.properties file into the \web-inf folder.  
f) Save the changes and Exit  
g) Move all the tomcat logs  
h) Clear the web application server cache (on Tomcat the work directory should be deleted)   
i) Recycle the Web Application Server and Web Server for the changes to take effect (as necessary)

2) Start a tracing port on the Service Manager Server  
a) Launch a dos prompt or Unix terminal equivalent  
b) Navigate to the Service Manager Server RUN directory  
c) Enter the following command: (The reference to 12345 can be changed to an available port if 12345 is already being used or unavailable)

sm -httpPort:12345 -RTM:3 -debugdbquery:999 -debughttp:1 -log:serverside\_webclient.log

3) Launch the Web Client and connect to the correct port  
a) Clear IE cache  
b) Download and install Fiddler 2 - <http://www.telerik.com/download/fiddler/fiddler2>  
c) (If using SSL) In IE, export your personal certificate - format as base64 named “ClientCertificate.cer”  
d) (If using SSL) Place the certificate into C:\users\{domain user}\documents\Fiddler2\  
e) (If using SSL) Fiddler 2 logs should appear as such  
f) (If using SSL) In Fiddler 2, complete the following actions  
i) Open tools > fiddler options > https  
ii) Tick the decrypt https traffic box  
iii) Trust & install the fiddler root certificate  
iv) Click and drag the “Any Process” target icon onto the IE window. This will commence logging on IE.  
g) In IE, utilize the following URL and try to reproduce the issue  
h) (If using SSL) https://<hostnameMatchingSSLCertificate>:8080/webtier-9.3x/index.do?serverPort=12345&tracesoap=true  
i) (If not using SSL) http://<hostname>:8080/webtier-9.3x/index.do?serverPort=12345&tracesoap=true

4) Reproduce the issue collecting exact steps taken by the users and then collect data defined above.

**log4j.properties:**

log4j.rootLogger=debug,R  
#uncomment next line to output to console.  
#log4j.appender.stdout=org.apache.log4j.ConsoleAppender  
log4j.appender.stdout.layout=org.apache.log4j.PatternLayout  
# Pattern to output the caller's file name and line number.  
log4j.appender.stdout.layout.ConversionPattern=%5p [%t] (%F:%L) - %m%n  
log4j.appender.R=org.apache.log4j.RollingFileAppender  
log4j.appender.R.File=sm\_webtier\_debug.log  
log4j.appender.R.MaxFileSize=10000KB  
#Keep one backup file  
log4j.appender.R.MaxBackupIndex=10  
log4j.appender.R.layout=org.apache.log4j.PatternLayout  
log4j.appender.R.layout.ConversionPattern=%d %p %t %c - %m%n  
log4j.logger.com.hp.ov.sm.client.eclipse.web=DEBUG  
log4j.logger.com.hp.ov.cwc=DEBUG