ThreatFactor NSIA - Bug #1898

Definitions cannot be loaded

06/07/2017 06:51 PM - Luke Murphey

Status:	New	**Start date:**	06/07/2017
Priority:	Normal	**Due date:**	
Assignee:	Luke Murphey	**% Done:**	0%
Category:	Scan Engine	**Estimated time:**	0.00 hour
Target version:	1.0.7		

Description

java.lang.NoClassDefFoundError: sun/org/mozilla/javascript/internal/Context
net.lukemurphey.nsia.scan.ScriptDefinition.getScriptEngine(ScriptDefinition.java:162) at
net.lukemurphey.nsia.scan.ScriptDefinition.<init>(ScriptDefinition.java:127) at
net.lukemurphey.nsia.scan.ScriptDefinition.<init>(ScriptDefinition.java:110) at
net.lukemurphey.nsia.scan.ScriptDefinition.<init>(ScriptDefinition.java:102) at
net.lukemurphey.nsia.scan.ScriptDefinition.parse(ScriptDefinition.java:86) at
net.lukemurphey.nsia.scan.DefinitionSet.loadFromXml(DefinitionSet.java:457) at
net.lukemurphey.nsia.scan.DefinitionSet.loadFromString(DefinitionSet.java:738) at
net.lukemurphey.nsia.scan.DefinitionArchive.updateDefinitions(DefinitionArchive.java:1019) at
net.lukemurphey.nsia.web.views.DefinitionsImportView.process(DefinitionsImportView.java:94) at
net.lukemurphey.nsia.web.View.process(View.java:92) at net.lukemurphey.nsia.web.View.process(View.java:59) at
net.lukemurphey.nsia.web.WebConsoleServlet.doRequest(WebConsoleServlet.java:83) at
net.lukemurphey.nsia.web.WebConsoleServlet.doPost(WebConsoleServlet.java:130) at
javax.servlet.http.HttpServlet.service(HttpServlet.java:154) at javax.servlet.http.HttpServlet.service(HttpServlet.java:92) at
org.mortbay.jetty.servlet.ServletHolder.handle(ServletHolder.java:428) at
org.mortbay.jetty.servlet.WebApplicationHandler.dispatch(WebApplicationHandler.java:473) at
org.mortbay.jetty.servlet.ServletHandler.handle(ServletHandler.java:568) at
org.mortbay.http.HttpContext.handle(HttpContext.java:1530) at
org.mortbay.jetty.servlet.WebApplicationContext.handle(WebApplicationContext.java:633) at
org.mortbay.http.HttpContext.handle(HttpContext.java:1482) at org.mortbay.http.HttpServer.service(HttpServer.java:909) at
org.mortbay.http.HttpConnection.service(HttpConnection.java:820) at
org.mortbay.http.HttpConnection.handleNext(HttpConnection.java:986) at
org.mortbay.http.HttpConnection.handle(HttpConnection.java:837) at
org.mortbay.http.SocketListener.handleConnection(SocketListener.java:245) at
org.mortbay.util.ThreadedServer.handle(ThreadedServer.java:357) at
org.mortbay.util.ThreadPool$PoolThread.run(ThreadPool.java:534)

Caused by:java.lang.ClassNotFoundException: sun.org.mozilla.javascript.internal.Context
java.net.URLClassLoader.findClass(URLClassLoader.java:381) at java.lang.ClassLoader.loadClass(ClassLoader.java:424) at
sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:331) at java.lang.ClassLoader.loadClass(ClassLoader.java:357) at
net.lukemurphey.nsia.scan.ScriptDefinition.getScriptEngine(ScriptDefinition.java:162) at
net.lukemurphey.nsia.scan.ScriptDefinition.<init>(ScriptDefinition.java:127) at
net.lukemurphey.nsia.scan.ScriptDefinition.<init>(ScriptDefinition.java:110) at
net.lukemurphey.nsia.scan.ScriptDefinition.<init>(ScriptDefinition.java:102) at
net.lukemurphey.nsia.scan.ScriptDefinition.parse(ScriptDefinition.java:86) at
net.lukemurphey.nsia.scan.DefinitionSet.loadFromXml(DefinitionSet.java:457) at
net.lukemurphey.nsia.scan.DefinitionSet.loadFromString(DefinitionSet.java:738) at
net.lukemurphey.nsia.scan.DefinitionArchive.updateDefinitions(DefinitionArchive.java:1019) at
net.lukemurphey.nsia.web.views.DefinitionsImportView.process(DefinitionsImportView.java:94) at
net.lukemurphey.nsia.web.View.process(View.java:92) at net.lukemurphey.nsia.web.View.process(View.java:59) at
net.lukemurphey.nsia.web.WebConsoleServlet.doRequest(WebConsoleServlet.java:83) at
net.lukemurphey.nsia.web.WebConsoleServlet.doPost(WebConsoleServlet.java:130) at
javax.servlet.http.HttpServlet.service(HttpServlet.java:154) at javax.servlet.http.HttpServlet.service(HttpServlet.java:92) at
org.mortbay.jetty.servlet.ServletHolder.handle(ServletHolder.java:428) at
org.mortbay.jetty.servlet.WebApplicationHandler.dispatch(WebApplicationHandler.java:473) at
org.mortbay.jetty.servlet.ServletHandler.handle(ServletHandler.java:568) at
org.mortbay.http.HttpContext.handle(HttpContext.java:1530) at
org.mortbay.jetty.servlet.WebApplicationContext.handle(WebApplicationContext.java:633) at
org.mortbay.http.HttpContext.handle(HttpContext.java:1482) at org.mortbay.http.HttpServer.service(HttpServer.java:909) at
org.mortbay.http.HttpConnection.service(HttpConnection.java:820) at
org.mortbay.http.HttpConnection.handleNext(HttpConnection.java:986) at
org.mortbay.http.HttpConnection.handle(HttpConnection.java:837) at

```
org.mortbay.http.SocketListener.handleConnection(SocketListener.java:245) at
org.mortbay.util.ThreadedServer.handle(ThreadedServer.java:357) at
org.mortbay.util.ThreadPool$PoolThread.run(ThreadPool.java:534)
```

History

#1 - 06/07/2017 06:59 PM - Luke Murphey

- Description updated

#2 - 06/07/2017 07:43 PM - Luke Murphey

Error is here, when attempting to control the context of the script:

```
        // 2 -- Set the context such that the script class loader will be used.
        Context context = Context.enter();
        context.setApplicationClassLoader(new ScriptClassLoader());
```

#3 - 06/07/2017 07:57 PM - Luke Murphey

- https://stackoverflow.com/questions/2822004/change-classloader
- https://stackoverflow.com/questions/30225398/java-8-scriptengine-across-classloaders

#4 - 06/07/2017 08:16 PM - Luke Murphey

It looks like I need to use a thread with a classloader. I might be able to set the class loader on the InvokeThread:. Below is the call structure:

- baseline
- evaluate
  - performAnalysis
    - InvokerThread

#5 - 06/07/2017 08:32 PM - Luke Murphey

Changes:

- Remove import of sun.org.mozilla.javascript.internal.Context
- Remove use of Context in getScriptEngine()
- getScriptEngine() eval likely needs to use a thread
- set the classloader for the InvokerThread class
- change baseline() to use a thread with the class loader

#6 - 06/07/2017 08:35 PM - Luke Murphey

Questions:

- Does a class use the classloader at runtime or does it inherit the classloader at initialization time?
  - "by default the thread context classloader at the time of nashorn engine creation. If the thread context class loader is null, then Nashorn's own loader - the extension loader - is used." (https://stackoverflow.com/questions/30225398/java-8-scriptengine-across-classloaders)

#7 - 06/07/2017 09:01 PM - Luke Murphey

Might be able to use https://docs.oracle.com/javase/8/docs/jdk/api/nashorn/jdk/nashorn/api/scripting/NashornScriptEngineFactory.html. See https://en.wikipedia.org/wiki/Nashorn_(JavaScript_engine).

#8 - 06/07/2017 10:24 PM - Luke Murphey

```
import jdk.nashorn.api.scripting.*;
import javax.script.*;

NashornScriptEngineFactory factory = new NashornScriptEngineFactory();

ClassLoader classLoader = this.getClass().getClassLoader();
ScriptEngine engine = factory.getScriptEngine(new String[] { "--global-per-engine" }, classLoader);
```

```
public class ScriptClassLoader extends ClassLoader {
    public Class<?> loadClass(String className) throws ClassNotFoundException {
        throw new ClassNotFoundException("Class " + className + " not found");
    }
}

ScriptClassLoader classLoader = new ScriptClassLoader();
```