Ssl Connector Guide

THE JETTY SSL CONNECTORS

There are 2 ssl connectors in jetty-- SslSocketConnector and the SslSelectChannelConnector. The SslSocketConnector is built on top of the Jetty SocketConnector which is Jetty's implementation of a blocking connector. It makes use of java's SslSocket to add the security layer. On the other hand, SslSelectChannelConnector is an extension of Jetty's SelectChannelConnector which makes use of non-blocking IO. For its security layer, it uses java nio SslEngine. Both Connectors can be configured in the same way. Only difference is in the implementation.

CONFIGURATION

The following is an example of an SslSocketConnector configuration. An SslSelectChannelConnector may be configured the same way-- just change the value of class to "org.mortbay.jetty.security.SslSelectChannelConnector".

```
<Call name="addConnector">
    <Arg>
      <New
class="org.mortbay.jetty.security.SslSocketC
onnector">
        <Set name="Port">8443</Set>
        <Set name="maxIdleTime">30000</Set>
        <Set name="keystore"><SystemProperty
name="jetty.home" default="."
/>/etc/keystore</Set>
        <Set
name="password">OBF:1vny1zlo1x8e1vnw1vn61x8g
1zlu1vn4</Set>
        <Set
name="keyPassword">OBF:1u2u1wml1z7s1z7a1wnl1
u2g</Set>
        <Set
name="truststore">><SystemProperty
name="jetty.home" default="."
/>/etc/keystore</Set>
        <Set
name="trustPassword">OBF:1vny1zlo1x8e1vnw1vn
61x8g1zlu1vn4</Set>
      </New>
    </Arg>
  </Call>
```

If there is no value for the "truststore", it will use the "keystore" value. Passwords can be obfuscated by running org.mortbay.util.Password as a main class.

Other properties which can be set for SslSocketConnector/SslSelectChannelConnector are:

- keystoreType - default value: "JKS"
- trustStoreType - default value: "JKS"
- sslKeyManagerFactoryAlgorithm - set to the value of the "ssl.KeyManagerFactory.algorithm" system property. If there is no such property, this defaults to "SunX509"
- sslTrustManagerFactoryAlgorithm - set to the value of the "ssl.TrustManagerFactory.algorithm" system property. If there is no such property, this defaults to "SunX509"
- secureRandomAlgorithm - default value is null
- provider - defaults to the SunJSSE provider
- protocol - default value is "TLS"
- excludeCipherSuites - see SSL Cipher Suites

reference http://java.sun.com/j2se/1.5.0/docs/guide/security/jsse/JSSERefGuide.html#SunJSSE