Hochschule RheinMain
DCSM - Informatik
Prof. Dr. Reinhold Kröger

Verteilte Systeme
WS 2015/16

Übungsblatt 9
Praktische Übungen

Ziel dieser Übung ist es das bekannte Filesystem aus den bisherigen Übungen als Webservice bereitzustellen und mit einem Client darauf zuzugreifen. Hierzu wird die SOAP Webservice-Implementierung von Apache CXF verwendet, da diese über eine Implementierung des WS-Security Standards verfügt. SOAP Webservices stellen einen RPC-Mechanismus bereit, ähnlich dem von Thrift. Neben der Umsetzung des Webservice soll als Sicherungsmaßnahme die Kommunikation verschlüsselt werden. Hierzu werden SSL und X.509-Zertifikate verwendet.

Literatur:
[1] http://cxf.apache.org

[2] http://cxf.apache.org/docs/a-simple-jax-ws-service.html

[3] http://cxf.apache.org/docs/defining-contract-first-webservices-with-wsdl-generation-from-java.html

[4] http://cxf.apache.org/docs/ws-security.html

[5] https://www.sslshopper.com/article-most-common-java-keytool-keystore-commands.html

[6] http://blog.montezuma.it/2013/02/apache-cxf-272-and-ws-security-how-to-encrypt-with-x509.html

[7] http://ws.apache.org/commons/tcpmon/

In Ihrem SVN finden Sie die folgenden Eclipse-Projekte deployed:

1. „VFileSystemServerAPI“ enthält wieder das bekannte Java-Interface für das Filesystem. In diesem Projekt sollen Sie keine Änderungen vornehmen.
2. „VFileSystemServerCLI“ enthält die bereits bekannte Benutzerschnittstelle, die auch in dieser Aufgabe wieder verwendet wird.
3. „VFileSystemLibs“ enthält die notwendigen Bibliotheken für die Aufgabe. Sie müssen **nicht** CXF herunter laden und installieren!
4. „VFileSystemEmulator“ enthält den bereits bekannten Emulator. In dieser Aufgabe müssen hier keine Änderungen vorgenommen werden!
5. „VFileSystemWSAPI“ enthält ein Java-Interface, welches Sie ergänzen und mit den CXF-Annotationen versehen müssen, damit darauf ein Webservice generiert werden kann.

6. „VFileSystemServerWS“ enthält die Klassen um einen Webservice aufsetzen und betreiben zu können. Dieser verwendet das Interface im VFileSystemWSAPI-Projekt. In diesem Projekt sind auch die Keystores des Servers für die Verschlüsselung und Authentifizierung zu speichern.
7. „VFileSystemClientWS“ enthält die Klassen für einen Webservice-Client. Die Keystores für den Clients sind hier zu speichern.

Außerdem finden Sie das Programm TCPmon (für Linux AMD64) in Ihrem SVN.

Hinweise:

- Denken Sie daran, dass Sie neue Dateien (beispielsweise die Keystores) mit `svn add <filename>` dem SVN-Repository hinzufügen müssen.
- Verwenden Sie unbedingt die Passwörter die angegeben sind!
- Verwenden Sie für die Lösung der Aufgabe Eclipse mit Java in der Version 1.7 (voreingestellt).

Aufgabe 9.1 (Webservices):

Informieren Sie sich darüber, wie mit CXF Webservices umgesetzt werden. Finden Sie heraus, welche Annotationen CXF unterstützt und wie damit die eine Java-Schnittstelle zu einer Webservice-Definition erweitert werden kann.

Aufgabe 9.2 (Schreiben des Filesystem-Webservices):

(a) Im Projekt VFileSystemWSAPI ist ein Interface gegeben, welches eine FileSystem-Schnittstelle beschreibt. Diese ist an die bisherige Schnittstelle angelehnt. Um die Anzahl der Methoden zu reduzieren, gibt es die get_file_info() und get_folder_info()-Methode, die die Informationen über Dateien und Ordner in Form eines Objekts zusammengefasst zurückgeben. Machen Sie sich mit dem Interface vertraut.

(b) Annotieren Sie das das Interface im VFileSystemWSAPI, um dieses mittels CXF als Webservice bereitstellen zu können. Stellen Sie sicher, dass alle Parameter in der Webservice-Beschreibung einen sprechenden Namen haben. Des Weiteren sind die beiden Klassen FolderInfo und FileInfo zu annotieren, damit diese automatisch als über JAXB übertragen werden können. Annotieren Sie auch die Exception, damit diese über den Webservice übertragen werden kann.

(c) Legen Sie im `VFileSystemServerWS` eine Klasse `FileSystemWSImpl` an, die das `FileSystemWS`-Interface implementiert. Diese Klasse sollte im Konstruktor eine Referenz auf das Filesystem-Interface übergeben bekommen, die verwendet wird, den Service umzusetzen. Legen Sie in der Main-Klasse einen Emulator an und setzen Sie damit den Webservice-Server auf. Der Webservice soll unter der Adresse „`http://<hostname>:<port>/filesystem`“ veröffentlicht werden. Überprüfen Sie, ob ihr Webservice läuft, in dem Sie in einem Web-Browser die URL „`http://<hostname>:<port>/filesystem?wsdl`“ aufrufen und sich das WSDL-File anschauen.

(d) Im gegebenen Eclipse-Projekt VFileSystemClientWS implementieren Sie den Client für den Webservice. Hierzu sollen Sie wieder die bekannte CLI-GUI verwenden. Da sich die Java-Schnittstelle von der des Webservice unterscheidet, müssen Sie auch hier eine Abbildung realisieren. Implementieren Sie hierzu das Java-Interface aus VFileSystemServerAPI in einer neuen Klasse und setzen Sie dort die Aufrufe auf den Webservice um. Verbinden Sie dann den Client mit dem Webservice.

(e) Machen Sie sich mit TCPMon [6] vertraut und nutzen Sie den Monitor anschließend, um den Datenverkehr zwischen Client und Server zu überwachen. Dokumentieren Sie Ihre Ergebnisse im Ordner „doku“ im „VFileSystemServerWS“-Projekt. Vergessen Sie nicht die neuen Dateien dem SVN hinzuzufügen!

Aufgabe 9.3 (Generierung und Austausch von Zertifikaten):

(a) Erzeugen Sie sich unter Nutzung von [4] und [5] drei Java-Keystores mit X.509 Zertifikaten nach dem RSA-Verfahren und mit 2048-bit Schlüssellänge:
- client1.jks
- client2.jks
- server.jks

Verwenden Sie für alle Keystores das Passwort: „geheim“.
Verwenden Sie für den Server-Keystore: „server“ als alias und „geheimServer“ als Passwort.
Verwenden Sie für die Client-Keystores: „client1“, bzw. „client2“ als alias und „geheimClient“ als gemeinsames Passwort.

(b) Exportieren Sie jeweils den öffentlichen Schlüssel des Zertifikats und importieren Sie in beide Client-Keystores den öffentlichen Schlüssel des Server (serverCert.rsa), sowie in den Server-Keystore die beiden öffentlichen Schlüssel der Clients (client1Cert.rsa, client2Cert.rsa). Dadurch können die Clients verschlüsselte Nachrichten mit dem Server austauschen, ohne dass ein Client die Nachrichten des anderen Clients entschlüsseln kann.

(c) Speichern Sie den Server-Keystore und die exportierten Client-Zertifikate in den Ordner „keystore“ im „VFileSystemServerWS“-Projekt. Speichern Sie die beiden Client-Keystores und das Server-Zertifikat in den Ordner „keystore“ im „VFileSystemClientWS“-Projekt. Vergessen Sie nicht, die neuen Dateien dem SVN hinzuzufügen.

Aufgabe 9.4 (Signieren der Nachrichten)

(a) Passen Sie unter Nutzung der Interceptoren von [3] Ihren Client so an, dass er beim Senden einer neuen Anfrage diese mit dem Schlüssel des Clients signiert. Des Weiteren soll er die Antworten des Servers mit dem bekannten öffentlichen Zertifikat des Servers überprüfen.

(b) Passen Sie unter Nutzung der Interceptoren von [3] Ihren Server so an, dass er zum einem Antworten an den Client signiert und zum anderen die Anfragen der Clients mit den bekannten öffentlichen Zertifikaten der Clients überprüft.

(c) Nutzen Sie TCPMon, um erneut den Datenverkehr zwischen Client und Server zu betrachten. Welche Elemente sind hinzugekommen? Dokumentieren Sie Ihre Ergebnisse im Ordner „doku“ des „VFileSystemServerWS“-Projekts.

Aufgabe 9.5 (Verschlüsseln der Nachrichten) Optional mit 4 Extrapunkten

(a) Erweitern Sie ihren Client dahingehend, dass er ausgehende Nachrichten verschlüsselt und Ihren Server, dass er Nachrichten der Clients entschlüsselt. Verwenden Sie eine AES-128-Verschlüsselung.

(b) Erweitern Sie ihren Server dahingehend, dass er ausgehende Nachrichten passend mit dem öffentlichen Schlüssel des Clients, der die Anfrage gestellt hat, verschlüsselt (hierzu das ausgehende Property `WSHandlerConstants.ENCRYPTION_USER` auf `"useReqSigCert"` setzen) und ihren Client so, dass er die Nachrichten entschlüsselt. Verwenden Sie eine AES-128-Verschlüsselung (größere Schlüssellängen sind auf Grund von „Exportbeschränkungen“ nicht einfach zu konfigurieren).

(c) Nutzen Sie TCPMon, um erneut den Datenverkehr zwischen Client und Server zu betrachtetn. Welche Elemente sind hinzugekommen? Dokumentieren Sie Ihre Ergebnisse im Ordner „doku“ des „VFileSystemServerWS“-Projekts.