8 Replies Latest reply : Jul 2, 2014 8:37 AM by samuelfac

samuelfac Jun 26, 2014 3:43 PM

Autenticação com Certificado no Apache 2.2 + Jboss 4.2.3 (HTTPS)

This question has been **Answered.**

Olá pessoal. Esta é a primeira vez que posto algo aqui, pois sempre que tive alguma dúvida sempre achei a solução em algum tópico antigo. Mas desta vez ainda não encontrei algo que funcione. Então vamos a dúvida:

AMBIENTE: Apache 2.2 (HTTPS) => Jboss 4.2.3 (HTTPS)

Tenho uma aplicação rodando no Jboss, com autenticação via certificado digital, ao acessar diretamente a página JSP no Jboss funciona perfeitamente, o certificado abre, eu escolho o certificado, na pagina jsp utilizo `java.security.cert.X509Certificate certChain[] = (java.security.cert.X509Certificate[]) request.getAttribute("javax.servlet.request.X509Certificate");` para pegar o certificado e validar em minha aplicação. Isto funciona perfeitamente.

Agora quando eu coloco o APACHE antes do Jboss fazendo o redirecionamento de https para https do jboss, o request.getAttribute retorna sempre NULL. O que devo fazer??
Configurações que fiz:

```
<VirtualHost myserver:443>

    ServerName myserver.local

    SSLEngine On

    SSLProxyEngine On


    SSLCertificateFile ...myfile.crt

    SSLCertificateKeyFile ...myfile.key


    SSLCACertificateFile ...ca.crt

    SSLVerifyClient optional


    SSLVerifyDepth 2

    SSLOptions +ExportCertData + StdEnvVars
    <Proxy *>

        Order deny,allow

        Allow from all

    </Proxy>


    <Location />


    ProxyPass         https://myserver:8443/
    ProxyPassReverse https://myserver:8443/

    </Location>

</VirtualHost>
```

Também já tentei com: `request.getHeader("SSL_CLIENT_CERT")` mas também só retorna NULL

alguem tem alguma ideia para me ajudar??

Grato

Correct Answer
by Mauricio Magnani Jr on Jul 1, 2014 8:50 AM

Boa Noite Samuel,

Pelo que entendi o seguinte cenário está funcionando:

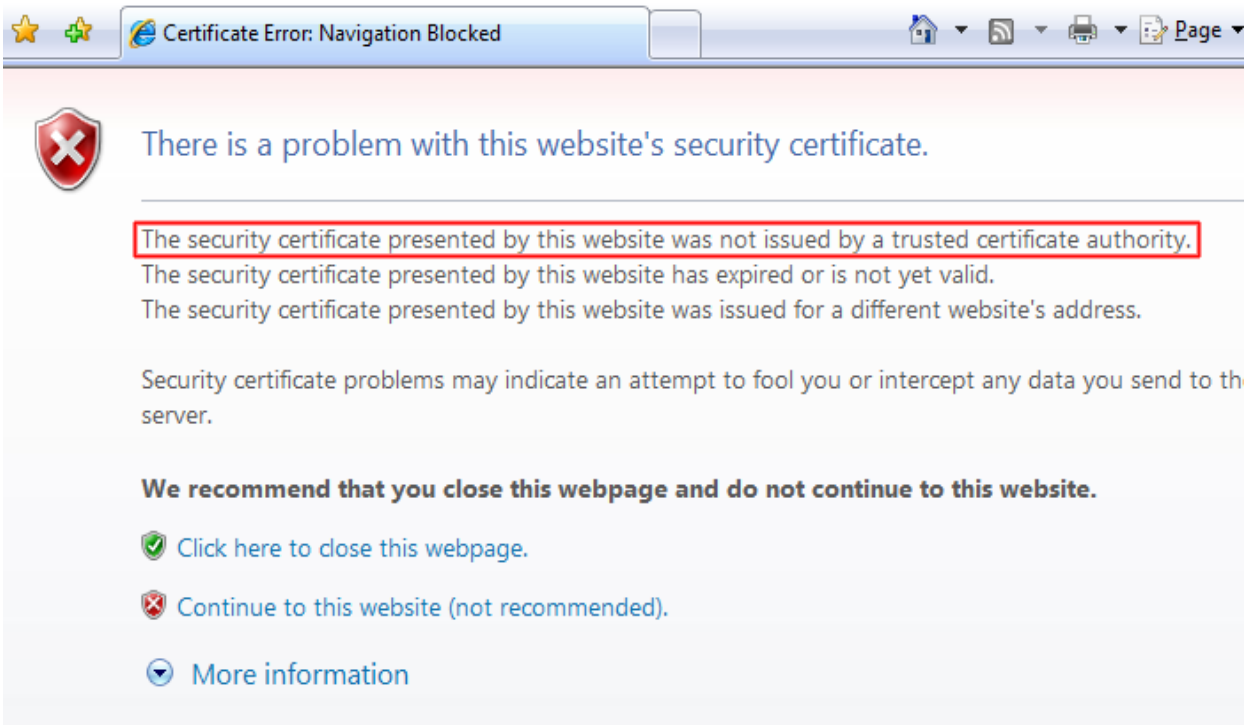
Client <--- HTTPS -----> JBoss 4.2.3

Você está tentando configurar o seguinte cenário:

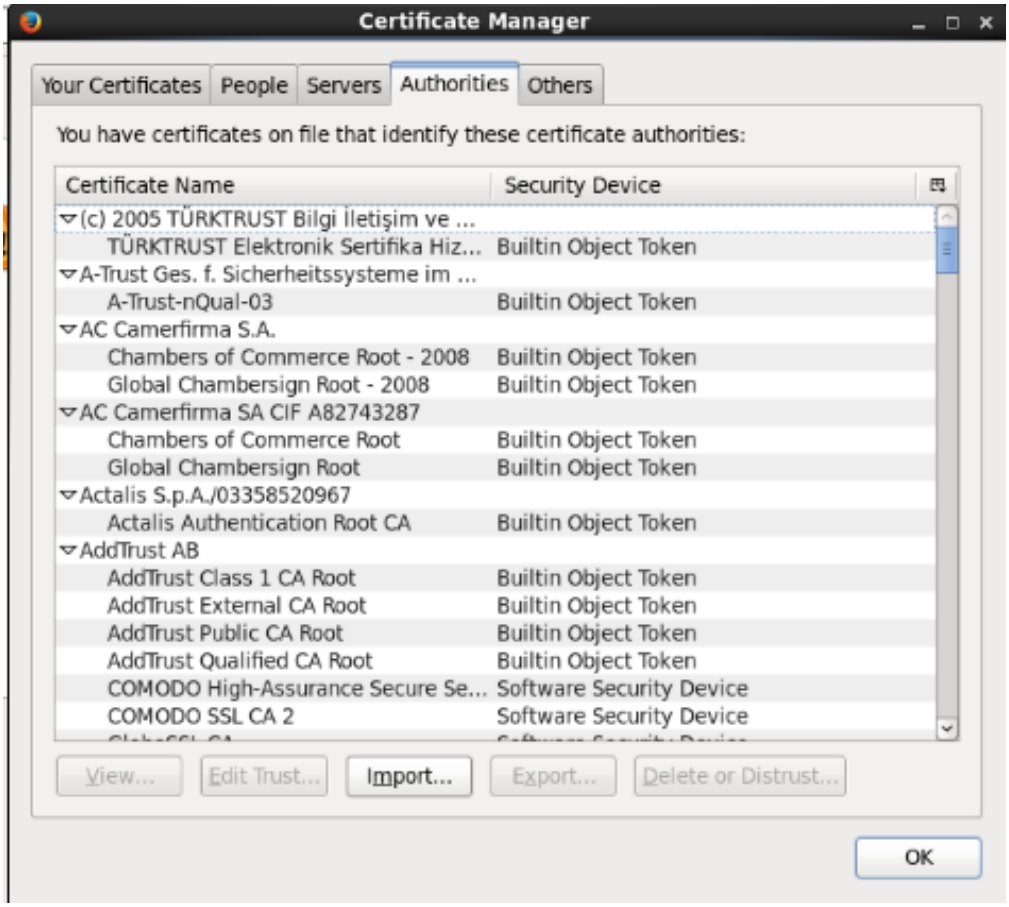
Client <--- HTTPS ---> Apache Web Server < ------ HTTPS -----> JBoss 4.2.3

O Cenário que você está tentanto configurar é conhecido como **Mutual Authentication** ou **Two-way Authentication** (http://en.wikipedia.org/wiki/Mutual_authentication). Para que essa estrutura possa funcionar de forma adequada, o certificado do client deve confiar no do servidor e vice-versa.

Primeiro vamos relembrar alguns conceitos... Quando um usuário utiliza o navegador para acessar uma aplicação (HTTPS), o Apache Web Server ou o JBoss irá fornecer um certificado. O navegador irá estabelecer uma conexão segura com o servidor. Isso é chamado de **One-way authentication**. Isso significa que o cliente vai verificar o identidade da aplicação, mas a aplicação não vai verificar a identidade do client. Se o certificado é auto-assinado, o navegador irá exibir um aviso a perguntar ao usuário se deve confiar ou não algo como a imagem abaixo:



No navegador existe uma lista de CA (autoridades certificadoras) que são confiáveis por padrão. Veja a lista no Firefox por exemplo:



Como vamos gerar os certificados auto assinados obviamente a nossa CA não estará presente nessa lista tornando o nosso certificado **"não confiável"**. O JDK também contém uma lista que geralmente está em **$JAVA_HOME/jre/lib/security/cacerts** e é nesse ponto que eu queria chegar

Acredito que você não esteja conseguindo pegar o certificado no Client quando utiliza o Apache Web Server como Proxy porque os certificados do JBoss (JVM) e do Apache não possuem uma relação de confiança ou seja, eles não são assinados pela mesma CA. Para que possamos estabelecer essa relação de confiança devemos criar uma autoridade certificadora e assinar os certificados do Apache e do JBoss.

1° Passo - Criar o Certificado da CA

Vamos criar um certificado auto assinado chamado **ca** e utilizar isso como a nossa autoridade certificadora. Como estou utilizando Linux criei um diretorio certs em /opt.

```
[root@mmagnani opt]# cd /opt/certs/

[root@mmagnani certs]# openssl req -new -newkey rsa -days 365 -x509 -keyout ca.key -out ca.crt

Generating a 2048 bit RSA private key
.....++++++
.....++++++
writing new private key to 'ca.key'
Enter PEM pass phrase:123456
Verifying - Enter PEM pass phrase:123456
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:BR
State or Province Name (full name) []:RIO DE JANEIRO
Locality Name (eg, city) [Default City]:RIO DE JANEIRO
Organization Name (eg, company) [Default Company Ltd]:JBUG BRASIL
Organizational Unit Name (eg, section) []:FORUM
Common Name (eg, your name or your server's hostname) []:mmagnani.csb
Email Address []:
```

Idetificar uma CA geralmente é uma tarefa simples. Basta encontrar o certificado em que o Owner e Issuer sejam os mesmos.

```
[root@mmagnani certs]# keytool -printcert -file ca.crt | head -n 2

Owner: CN=mmagnani.csb, OU=FORUM, O=JBUG BRASIL, L=RIO DE JANEIRO, ST=RIO DE JANEIRO, C=BR
Issuer: CN=mmagnani.csb, OU=FORUM, O=JBUG BRASIL, L=RIO DE JANEIRO, ST=RIO DE JANEIRO, C=BR
```

2° Passo - Criar o Certificado para o Apache Web Server

Vamos criar o certificado para o Apache Web Server e assina-lo com a nossa CA.

```
[root@mmagnani certs]# openssl genrsa -des3 -out httpd.key 1024

Generating RSA private key, 1024 bit long modulus
.....++++++
.....++++++
e is 65537 (0x10001)
Enter pass phrase for httpd.key:123456
Verifying - Enter pass phrase for httpd.key:123456
```

Agora vamos gerar o pedido para a assinatura:

```
[root@mmagnani certs]# openssl req -new -key httpd.key -out httpd.csr

Enter pass phrase for httpd.key:123456
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:BR
State or Province Name (full name) []:SAO PAULO
Locality Name (eg, city) [Default City]:SAO PAULO
Organization Name (eg, company) [Default Company Ltd]:MYCOMPANY
Organizational Unit Name (eg, section) []:TI
Common Name (eg, your name or your server's hostname) []:mmagnani.csb
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

O Common Name deve ser o nome (Nome / DNS) do servidor em que o Apache será utilizado.

3° Passo - Assinar o certificado com a nossa CA

Alguns arquivos devem ser criados:

```
# mkdir -p /opt/certs/demoCA/newcerts
# touch /opt/certs/demoCA/index.txt
# touch /opt/certs/demoCA/serial
# echo "01" > /opt/certs/demoCA/serial
```

Realize algumas configurações no arquivo /etc/pki/tls/openssl.cnf. Existem algumas coisas que devem ser opicionais para que os nossos testes sejam realizados com sucesso. O policy_match deve ficar como abaixo:

```
# For the CA policy
[ policy_match ]
countryName             = optional
stateOrProvinceName     = optional
organizationName        = optional
organizationalUnitName  = optional
commonName              = supplied
emailAddress            = optional
```

Precisamos também alterar o diretório padrão do OpenSSL.

```
# These are used by the TSA reply generation only.
#dir = /etc/pki/CA
dir       = ./opt/certs/demoCA            # TSA root directory
```

Finalmente assine o certificado utilizando o OpenSSL:

```
[root@mmagnani certs]# openssl ca -in httpd.csr -keyfile ca.key -cert ca.crt

Using configuration from /etc/pki/tls/openssl.cnf
Enter pass phrase for ca.key:123456
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 1 (0x1)
        Validity
            Not Before: Jun 27 03:34:22 2014 GMT
            Not After : Jun 27 03:34:22 2015 GMT
        Subject:
            countryName               = BR
            stateOrProvinceName       = SAO PAULO
            organizationName          = MYCOMPANY
            organizationalUnitName    = TI
            commonName                = mmagnani.csb
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                61:1A:60:C3:7A:89:B5:9D:B1:AE:57:92:46:A9:A7:A5:04:B4:B4:BD
            X509v3 Authority Key Identifier:
                keyid:0C:CA:47:DF:11:91:E3:D7:CC:98:C3:B5:E0:F1:4C:E8:F3:3C:74:62

Certificate is to be certified until Jun 27 03:34:22 2015 GMT (365 days)
Sign the certificate? [y/n]:y



1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=BR, ST=RIO DE JANEIRO, L=RIO DE JANEIRO, O=JBUG BRASIL,
OU=FORUM, CN=mmagnani.csb
        Validity
            Not Before: Jun 27 03:34:22 2014 GMT
            Not After : Jun 27 03:34:22 2015 GMT
        Subject: C=BR, ST=SAO PAULO, O=MYCOMPANY, OU=TI, CN=mmagnani.csb
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (1024 bit)
                Modulus:
                    00:cc:77:b4:f6:bb:87:bd:0c:e9:4f:8c:b4:d9:f0:
                    2c:04:80:6f:71:71:82:e1:aa:3e:e4:43:dc:7d:5f:
                    f4:72:a0:1a:cf:37:76:c2:43:77:d0:2a:d9:7e:29:
                    f8:24:94:c6:c4:0c:1e:58:9c:62:20:36:78:e1:13:
                    64:5f:a0:51:87:b3:44:f6:52:d9:f7:83:a3:fb:3e:
```

```
                c9:e9:45:3c:5e:d7:ee:36:43:c3:b7:96:66:57:f1:
                bc:52:ba:f3:10:d7:d8:bf:9a:50:8e:02:ca:d6:48:
                76:46:90:7f:c3:1b:50:a4:b2:48:b7:a2:8f:ef:e2:
                91:d4:41:12:32:3d:5f:6b:75
            Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                61:1A:60:C3:7A:89:B5:9D:B1:AE:57:92:46:A9:A7:A5:04:B4:B4:BD
            X509v3 Authority Key Identifier:
                keyid:0C:CA:47:DF:11:91:E3:D7:CC:98:C3:B5:E0:F1:4C:E8:F3:3C:74:62

    Signature Algorithm: sha1WithRSAEncryption
        3c:85:98:48:ed:d8:40:52:ec:e5:ab:82:7d:6b:c7:cd:ae:cf:
        1b:d0:d7:f9:fc:c8:1d:0d:1d:2d:45:bf:62:ec:cf:e4:f7:ba:
        64:f1:96:5c:58:fa:ad:4a:0c:91:c8:bd:f4:38:51:aa:50:92:
        8b:3a:6b:70:37:8d:f1:c6:21:a3:cb:7e:2a:ca:6a:7b:c0:76:
        77:69:4d:02:e6:3b:f7:0b:be:26:38:4a:25:ef:02:10:db:0a:
        74:9c:77:78:65:01:f4:2f:53:b8:70:87:3e:40:61:47:47:33:
        dd:f2:13:b2:5b:7c:97:3c:66:d2:1a:6b:15:ca:d2:f1:88:a8:
        ca:8a
-----BEGIN CERTIFICATE-----
MIICxjCCAi+gAwIBAgIBATANBgkqhkiG9w0BAQUFADB8MQswCQYDVQQGEwJCUjEX
MBUGA1UECAwOUklPIERFIEpBTkVJUk8xFzAVBgNVBAcMDlJJTyBERSBKQU5FSVJP
MRQwEgYDVQQKDAtKQlVHIEJSQVNJTDEOMAwGA1UECwwFRk9SVU0xFTATBgNVBAMM
DG1tYWduYW5pLmNzYjAeFw0xNDA2MjcwMzM0MjJaFw0xNTA2MjcwMzM0MjJaMFkx
CzAJBgNVBAYTAkJSMRIwEAYDVQQIDAlTQU8gUEFVTE8xEjAQBgNVBAoMCU1ZQ09N
UEFOWTELMAkGA1UECwwCVEkxFTATBgNVBAMMDG1tYWduYW5pLmNzYjCBnzANBgkq
hkiG9w0BAQEFAAOBjQAwgYkCgYEAzHe09ruHvQzpT4y02fAsBIBvcXGC4ao+5EPc
fV/0cqAazzd2wkN30CrZfin4JJTGxAweWJxiIDZ44RNkX6BRh7NE9lLZ94Oj+z7J
6UU8XtfuNkPDt5ZmV/G8UrrzENfYv5pQjgLK1kh2RpB/wxtQpLJIt6KP7+KR1EES
Mj1fa3UCAwEAAaN7MHkwCQYDVR0TBAIwADAsBglghkgBhvhCAQ0EHxYdT3BlblNT
TCBHZW5lcmF0ZWQgQ2VydGlmaWNhdGUwHQYDVR0OBBYEFGEaYMN6ibWdsa5Xkkap
p6UEtLS9MB8GA1UdIwQYMBaAFAzKR98RkePXzJjDteDxTOjzPHRiMA0GCSqGSIb3
DQEBBQUAA4GBADyFmEjt2EBS7OWrgn1rx82uzxvQ1/n8yB0NHS1Fv2Lsz+T3umTx
IlxY+q1KDJHIvfQ4UapQkos6a3A3jfHGIaPLfirKanvAdndpTQLmO/cLviY4SiXv
AhDbCnScd3hlAfQvU7hwhz5AYUdHM93yE7JbfJc8ZtIaaxXK0vGIqMqK
-----END CERTIFICATE-----
Data Base Updated
```

Observe o certificado assinado /opt/certs/demoCA/newcerts/01.pem

```
[root@mmagnani certs]# cat /opt/certs/demoCA/newcerts/01.pem

Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=BR, ST=RIO DE JANEIRO, L=RIO DE JANEIRO, O=JBUG BRASIL,
OU=FORUM, CN=mmagnani.csb
        Validity
            Not Before: Jun 27 03:34:22 2014 GMT
            Not After : Jun 27 03:34:22 2015 GMT
        Subject: C=BR, ST=SAO PAULO, O=MYCOMPANY, OU=TI, CN=mmagnani.csb
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (1024 bit)
                Modulus:
                    00:cc:77:b4:f6:bb:87:bd:0c:e9:4f:8c:b4:d9:f0:
                    2c:04:80:6f:71:71:82:e1:aa:3e:e4:43:dc:7d:5f:
                    f4:72:a0:1a:cf:37:76:c2:43:77:d0:2a:d9:7e:29:
                    f8:24:94:c6:c4:0c:1e:58:9c:62:20:36:78:e1:13:
                    64:5f:a0:51:87:b3:44:f6:52:d9:f7:83:a3:fb:3e:
                    c9:e9:45:3c:5e:d7:ee:36:43:c3:b7:96:66:57:f1:
                    bc:52:ba:f3:10:d7:d8:bf:9a:50:8e:02:ca:d6:48:
                    76:46:90:7f:c3:1b:50:a4:b2:48:b7:a2:8f:ef:e2:
                    91:d4:41:12:32:3d:5f:6b:75
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                61:1A:60:C3:7A:89:B5:9D:B1:AE:57:92:46:A9:A7:A5:04:B4:B4:BD
            X509v3 Authority Key Identifier:
                keyid:0C:CA:47:DF:11:91:E3:D7:CC:98:C3:B5:E0:F1:4C:E8:F3:3C:74:62


    Signature Algorithm: sha1WithRSAEncryption
        3c:85:98:48:ed:d8:40:52:ec:e5:ab:82:7d:6b:c7:cd:ae:cf:
        1b:d0:d7:f9:fc:c8:1d:0d:1d:2d:45:bf:62:ec:cf:e4:f7:ba:
        64:f1:96:5c:58:fa:ad:4a:0c:91:c8:bd:f4:38:51:aa:50:92:
        8b:3a:6b:70:37:8d:f1:c6:21:a3:cb:7e:2a:ca:6a:7b:c0:76:
        77:69:4d:02:e6:3b:f7:0b:be:26:38:4a:25:ef:02:10:db:0a:
        74:9c:77:78:65:01:f4:2f:53:b8:70:87:3e:40:61:47:47:33:
        dd:f2:13:b2:5b:7c:97:3c:66:d2:1a:6b:15:ca:d2:f1:88:a8:
        ca:8a
-----BEGIN CERTIFICATE-----
MIICxjCCAi+gAwIBAgIBATANBgkqhkiG9w0BAQUFADB8MQswCQYDVQQGEwJCUjEX
MBUGA1UECAwOUklPIERFIEpBTkVJUk8xFzAVBgNVBAcMDlJJTyBERSBKQU5FSVJP
MRQwEgYDVQQKDAtKQlVHIEJSQVNJTDEOMAwGA1UECwwFRk9SVU0xFTATBgNVBAMM
DG1tYWduYW5pLmNzYjAeFw0xNDA2MjcwMzM0MjJaFw0xNTA2MjcwMzM0MjJaMFkx
CzAJBgNVBAYTAkJSMRIwEAYDVQQIDAlTQU8gUEFVTE8xEjAQBgNVBAoMCU1ZQ09N
UEFOWTELMAkGA1UECwwCVEkxFTATBgNVBAMMDG1tYWduYW5pLmNzYjCBnzANBgkq
```

```
hkiG9w0BAQEFAAOBjQAwgYkCgYEAzHe09ruHvQzpT4y02fAsBIBvcXGC4ao+5EPc
fV/0cqAazzd2wkN30CrZfin4JJTGxAweWJxiIDZ44RNkX6BRh7NE9lLZ94Oj+z7J
6UU8XtfuNkPDt5ZmV/G8UrrzENfYv5pQjgLK1kh2RpB/wxtQpLJlt6KP7+KR1EES
Mj1fa3UCAwEAAaN7MHkwCQYDVR0TBAIwADAsBglghkgBhvhCAQ0EHxYdT3BlblNT
TCBHZW5lcmF0ZWQgQ2VydGlmaWNhdGUwHQYDVR0OBBYEFGEaYMN6ibWdsa5Xkkap
p6UEtLS9MB8GA1UdIwQYMBaAFAzKR98RkePXzJjDteDxTOjzPHRiMA0GCSqGSIb3
DQEBBQUAA4GBADyFmEjt2EBS7OWrgn1rx82uzxvQ1/n8yB0NHS1Fv2Lsz+T3umTx
llxY+q1KDJHIvfQ4UapQkos6a3A3jfHGIaPLfirKanvAdndpTQLmO/cLviY4SiXv
AhDbCnScd3hlAfQvU7hwhz5AYUdHM93yE7JbfJc8ZtIaaxXK0vGIqMqK
-----END CERTIFICATE-----
```

Para finalizar copie e renomei o certificado como abaixo:

```
[root@mmagnani certs]# cp /opt/certs/demoCA/newcerts/01.pem /opt/certs/httpd.crt
```

4° Passo - Criar os Certificados do JBoss

Agora vamos gerar keystore utilizando o keytool que é a ferramente padrão do JDK para criação e gerenciamento de certificados.

```
[root@mmagnani certs]# /usr/java/jdk1.7.0_60/bin/keytool -genkey -keystore jboss.jks -
storepass 123456 -keypass 123456 -keyalg RSA -validity 365 -alias jboss
What is your first and last name?
  [Unknown]:  MAURICIO MAGNANI
What is the name of your organizational unit?
  [Unknown]:  TI
What is the name of your organization?
  [Unknown]:  MY COMPANY
What is the name of your City or Locality?
  [Unknown]:  SAO PAULO
What is the name of your State or Province?
  [Unknown]:  SAO PAULO
What is the two-letter country code for this unit?
  [Unknown]:  BR
Is CN=MAURICIO MAGNANI, OU=TI, O=MY COMPANY, L=SAO PAULO, ST=SAO PAULO,
C=BR correct?
  [no]:  YES
```

```
[root@mmagnani certs]# /usr/java/jdk1.7.0_60/bin/keytool -list -v -keystore jboss.jks --alias
jboss
Enter keystore password:
Alias name: jboss
Creation date: Jun 27, 2014
Entry type: PrivateKeyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=MAURICIO MAGNANI, OU=TI, O=MY COMPANY, L=SAO PAULO, ST=SAO
PAULO, C=BR
Issuer: CN=MAURICIO MAGNANI, OU=TI, O=MY COMPANY, L=SAO PAULO, ST=SAO
PAULO, C=BR
Serial number: 6d31886d
Valid from: Fri Jun 27 22:52:59 BRT 2014 until: Sat Jun 27 22:52:59 BRT 2015
Certificate fingerprints:
  MD5:  7D:7E:F8:1F:48:91:7D:99:46:83:42:5C:07:CB:27:E1
  SHA1: 56:08:23:17:AB:A4:63:58:8E:F1:05:28:C9:76:6A:4E:13:3A:AB:92
  SHA256:
C2:0E:0D:34:E9:2B:22:F1:B8:09:52:6C:DB:B1:7E:4D:92:78:16:14:3C:66:AD:97:06:36:04:81:A
  Signature algorithm name: SHA256withRSA
  Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 47 A8 00 3A 96 5E 14 43   4F 1F 99 10 6A A8 41 A2  G..:.^.CO...j.A.
0010: AF E9 3F 70                                        ..?p
]
]
```

```
[root@mmagnani certs]# /usr/java/jdk1.7.0_60/bin/keytool -certreq -keyalg RSA -alias jboss -
file jbosscert.csr -keystore jboss.jks
Enter keystore password:
```

```
[root@mmagnani certs]# openssl ca -in jbosscert.csr -keyfile ca.key -cert ca.crt
Using configuration from /etc/pki/tls/openssl.cnf
Enter pass phrase for ca.key:
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 2 (0x2)
        Validity
            Not Before: Jun 28 01:58:09 2014 GMT
            Not After : Jun 28 01:58:09 2015 GMT
        Subject:
            countryName               = BR
            stateOrProvinceName       = SAO PAULO
            organizationName          = MY COMPANY
            organizationalUnitName    = TI
            commonName                = MAURICIO MAGNANI
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                47:A8:00:3A:96:5E:14:43:4F:1F:99:10:6A:A8:41:A2:AF:E9:3F:70
            X509v3 Authority Key Identifier:
                keyid:0C:CA:47:DF:11:91:E3:D7:CC:98:C3:B5:E0:F1:4C:E8:F3:3C:74:62

Certificate is to be certified until Jun 28 01:58:09 2015 GMT (365 days)
Sign the certificate? [y/n]:y



1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 2 (0x2)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=BR, ST=RIO DE JANEIRO, L=RIO DE JANEIRO, O=JBUG BRASIL,
OU=FORUM, CN=mmagnani.csb
        Validity
            Not Before: Jun 28 01:58:09 2014 GMT
            Not After : Jun 28 01:58:09 2015 GMT
        Subject: C=BR, ST=SAO PAULO, O=MY COMPANY, OU=TI, CN=MAURICIO
MAGNANI
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:de:eb:db:eb:cf:23:21:ed:4e:2a:e5:6c:57:13:
                    0d:45:0f:db:79:7f:d7:21:fb:80:f4:fc:43:37:5f:
                    42:12:14:d7:6c:0f:d5:4b:4f:88:a2:b9:ae:4b:b5:
                    c5:7d:2e:8f:30:77:f0:91:55:be:b1:e8:c1:6f:96:
```

```
                a0:16:41:5d:ea:d4:12:c7:d8:cd:ec:db:35:22:62:
                47:02:5a:0b:2a:ac:a6:50:69:c0:19:71:ab:6b:c5:
                df:72:a0:2c:25:06:77:b3:37:24:c0:af:2d:13:6d:
                c5:93:0c:9c:00:58:10:a4:3e:92:84:25:a6:10:10:
                d7:88:9d:24:30:0e:26:19:6b:80:40:39:31:2b:f5:
                a7:1f:ae:a9:27:14:bd:66:e7:c3:76:35:e0:ac:68:
                88:fc:fd:89:0c:69:3b:06:f5:08:45:14:4a:64:da:
                09:74:7d:05:41:77:cb:93:4c:bd:11:54:ea:fd:ce:
                b6:69:34:0d:7e:29:10:a8:c7:bf:cb:8c:0a:c6:4a:
                8d:76:27:99:0a:60:f6:29:f7:f1:1e:10:cb:08:d6:
                8c:26:81:dc:cd:d4:6f:00:5c:37:4a:15:f0:d2:57:
                c4:42:9b:4f:f7:64:67:2d:fe:cd:86:eb:c3:1b:d5:
                b1:c8:14:55:d8:92:a4:7d:c2:5d:d1:7e:9c:5c:60:
                a5:d5
            Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                47:A8:00:3A:96:5E:14:43:4F:1F:99:10:6A:A8:41:A2:AF:E9:3F:70
            X509v3 Authority Key Identifier:
                keyid:0C:CA:47:DF:11:91:E3:D7:CC:98:C3:B5:E0:F1:4C:E8:F3:3C:74:62

    Signature Algorithm: sha1WithRSAEncryption
        e4:7e:cb:98:58:70:9b:9c:5e:e3:26:b0:9f:74:16:7b:ce:be:
        1b:67:dd:56:36:ac:f0:41:68:5b:97:9a:2a:7a:9f:91:33:1d:
        54:07:a6:dd:9d:2a:d5:10:df:f1:bd:d6:76:c9:bd:3e:aa:57:
        80:af:d1:fd:db:26:89:3f:d7:e9:26:06:ff:ea:b9:05:d9:5d:
        15:98:16:40:30:06:d9:c6:ba:1e:55:54:86:a8:3c:3b:77:a3:
        82:2f:80:b4:54:90:ec:73:80:a5:bf:d0:58:79:20:9a:d6:dc:
        ff:ea:28:07:9a:fb:66:c7:e2:12:3d:36:7f:76:cd:d4:fd:96:
        98:56
-----BEGIN CERTIFICATE-----
MIIDTzCCArigAwIBAgIBAjANBgkqhkiG9w0BAQUFADB8MQswCQYDVQQGEwJCUjEX
MBUGA1UECAwOUklPIERFIEpBTkVJUk8xFzAVBgNVBAcMDlJJTyBERSBKQU5FSVJP
MRQwEgYDVQQKDAtKQlVHIEJSQVNJTDEOMAwGA1UECwwFRk9SVU0xFTATBgNVBAMM
DG1tYWduYW5pLmNzYjAeFw0xNDA2MjgwMTU4MDlaFw0xNTA2MjgwMTU4MDlaMF4x
CzAJBgNVBAYTAkJSMRIwEAYDVQQIEwlTQU8gUEFVTE8xEzARBgNVBAoTCk1ZIENP
TVBBTlkxCzAJBgNVBAsTAlRJMRkwFwYDVQQDExBNQVVSSUNJTyBNQUdOQU5JMIIB
IjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA3uvb688jIe1OKuVsVxMNRQ/b
eX/XIfuA9PxDN19CEhTXbA/VS0+IormuS7XFfS6PMHfwkVW+sejBb5agFkFd6tQS
x9jN7Ns1ImJHAIoLKqymUGnAGXGra8XfcqAsJQZ3szckwK8tE23FkwycAFgQpD6S
hCWmEBDXiJ0kMA4mGWuAQDkxK/WnH66pJxS9ZufDdjXgrGiI/P2JDGk7BvUIRRRK
ZNoJdH0FQXfLk0y9EVTq/c62aTQNfikQqMe/y4wKxkqNdieZCmD2KffxHhDLCNaM
JoHczdRvAFw3ShXw0lfEQptP92RnLf7NhuvDG9WxyBRV2JKkfcJd0X6cXGCl1QID
AQABo3sweTAJBgNVHRMEAjAAMCwGCWCGSAGG+EIBDQQfFh1PcGVuU1NMIEdlbmVy
YXRlZCBDZXJ0aWZpY2F0ZTAdBgNVHQ4EFgQUR6gAOpZeFENPH5kQaqhBoq/pP3Aw
HwYDVR0jBBgwFoAUDMpH3xGR49fMmMO14PFM6PM8dGIwDQYJKoZIhvcNAQEFBQAD
gYEA5H7LmFhwm5xe4yawn3QWe86+G2fdVjas8EFoW5eaKnqfkTMdVAem3Z0q1RDf
8b3Wdsm9PqpXgK/R/dsmiT/X6SYG/+q5BdldFZgWQDAG2ca6HlVUhqg8O3ejgi+A
tFSQ7HOApb/QWHkgmtbc/+ooB5r7ZsfiEj02f3bN1P2WmFY=
```

```
-----END CERTIFICATE-----
Data Base Updated
```

```
[root@mmagnani certs]# cp demoCA/newcerts/02.pem jbossraw.crt
grep -A 50 "BEGIN CERTIFICATE" jbossraw.crt > jbosscert.crt
keytool -printcert -file jbosscert.crt | head -n 2
keytool -import -v -trustcacerts -alias ca -file ca.crt -keystore jboss.jks
```

```
[root@mmagnani certs]# /usr/java/jdk1.7.0_60/bin/keytool -printcert -file jbosscert.crt | head -
n 2
Owner: CN=MAURICIO MAGNANI, OU=TI, O=MY COMPANY, ST=SAO PAULO, C=BR
Issuer: CN=mmagnani.csb, OU=FORUM, O=JBUG BRASIL, L=RIO DE JANEIRO, ST=RIO
DE JANEIRO, C=BR
[root@mmagnani certs]# /usr/java/jdk1.7.0_60/bin/keytool -import -v -trustcacerts -alias ca -
file ca.crt -keystore jboss.jks
Enter keystore password:
Owner: CN=mmagnani.csb, OU=FORUM, O=JBUG BRASIL, L=RIO DE JANEIRO, ST=RIO
DE JANEIRO, C=BR
Issuer: CN=mmagnani.csb, OU=FORUM, O=JBUG BRASIL, L=RIO DE JANEIRO, ST=RIO
DE JANEIRO, C=BR
Serial number: 966b6e2d25f81ec7
Valid from: Fri Jun 27 00:01:13 BRT 2014 until: Sat Jun 27 00:01:13 BRT 2015
Certificate fingerprints:
  MD5:  A7:D1:5E:BB:8D:C1:4D:BB:43:81:D7:0B:F9:71:18:40
  SHA1: B8:87:20:29:D9:BC:98:FD:AB:C8:21:56:0E:1E:4A:1D:81:95:3D:08
  SHA256:
17:ED:CB:5E:12:68:B8:E7:3E:2B:0B:0E:59:28:02:54:3C:4F:D8:51:B4:05:42:1B:4F:2A:34:63:8(
  Signature algorithm name: SHA1withRSA
  Version: 3
Extensions:
#1: ObjectId: 2.5.29.35 Criticality=false
AuthorityKeyIdentifier [
KeyIdentifier [
0000: 0C CA 47 DF 11 91 E3 D7   CC 98 C3 B5 E0 F1 4C E8  ..G.....L.
0010: F3 3C 74 62                                        .<tb
]
]

#2: ObjectId: 2.5.29.19 Criticality=false
BasicConstraints:[
  CA:true
  PathLen:2147483647
]

#3: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 0C CA 47 DF 11 91 E3 D7   CC 98 C3 B5 E0 F1 4C E8  ..G.....L.
0010: F3 3C 74 62                                        .<tb
]
]

Trust this certificate? [no]:  yes
Certificate was added to keystore
[Storing jboss.jks]
```

[root@mmagnani certs]# /usr/java/jdk1.7.0_60/bin/keytool -import -v -alias jbosscert -file jbosscert.crt -keystore jboss.jks
Enter keystore password:
Certificate was added to keystore
[Storing jboss.jks]

[root@mmagnani certs]# /usr/java/jdk1.7.0_60/bin/keytool -list -keystore jboss.jks
Enter keystore password:

Keystore type: JKS
Keystore provider: SUN

Your keystore contains 3 entries

ca, Jun 27, 2014, trustedCertEntry,
Certificate fingerprint (SHA1):
B8:87:20:29:D9:BC:98:FD:AB:C8:21:56:0E:1E:4A:1D:81:95:3D:08
jboss, Jun 27, 2014, PrivateKeyEntry,
Certificate fingerprint (SHA1):
56:08:23:17:AB:A4:63:58:8E:F1:05:28:C9:76:6A:4E:13:3A:AB:92
jbosscert, Jun 27, 2014, trustedCertEntry,
Certificate fingerprint (SHA1):
8B:CC:61:FD:23:70:8A:E6:50:61:56:0B:0F:08:83:ED:91:5F:41:2A

Tenho que organizar essa resposta e colocar as configurações.

Servlet Utilizada nos Testes

```
01. @WebServlet(name = "X509Test", urlPatterns = {"/X509Test"})
02. public class X509Test extends HttpServlet {
03.
04.
05.     @Override
06.     public void doGet(HttpServletRequest req, HttpServletResponse res) thrc
07.       IOException {
08.
09.     res.setContentType("text/plain");
10.     PrintWriter out = res.getWriter();
11.
12.
13.     X509Certificate[] certs = (X509Certificate[]) req.getAttribute("java.se
14.
15.     if (certs != null) {
16.       for (int i = 0; i < certs.length; i++) {
17.         out.println("Client Certificate [" + i + "] = " + certs[i].toString
18.       }
19.     } else {
20.       if ("https".equals(req.getScheme())) {
21.         out.println("This was an HTTPS request, " + "but no client certific
22.       } else {
23.         out.println("This was not an HTTPS request, " + "so no client certi
24.       }
25.     }
26.   }
27.
28.
29. }
```

Continua...

1008 Views Tags: jboss , proxy , https

Average User Rating

(0 ratings)

Adriano Schmidt Jun 26, 2014 9:29 PM (in response to samuelfac)

1. Re: Autenticação com Certificado no Apache 2.2 + Jboss 4.2.3 (HTTPS)

Fala Samuel!

Tenta fazer um teste trocando o:
SLVerifyClient optional
por:
SSLVerifyClient require

Acho que vai dar um erro.. aí teremos que configurar mais algumas coisas.. faz um teste e depois avisa a gente aqui..

Abraço!
Adriano Schmidt
www.localhost8080.com.br

Actions Like (0)

Adriano Schmidt Jun 26, 2014 9:34 PM (in response to samuelfac)

2. Re: Autenticação com Certificado no Apache 2.2 + Jboss 4.2.3 (HTTPS)

Talvez tenhamos que configurar o SSLCACertificateFile também..

Actions Like (0)

Adriano Schmidt Jun 26, 2014 9:52 PM (in response to samuelfac)

3. Re: Autenticação com Certificado no Apache 2.2 + Jboss 4.2.3 (HTTPS)

Como você gerou seus certificados? com openssl?

Eu já fiz isso em um ambiente de teste em outras verções do apache e JBoss e fiz dessa forma:

yum install mod_ssl
yum install openssl ca-certificates
mkdir /etc/ssl/server-certs
cd /etc/ssl/server-certs
openssl genrsa -des3 -out my-server.key 2048
openssl rsa -in my-server.key -out my-server.key.public
openssl req -new -key my-server.key -out my-server.csr
openssl x509 -req -days 365 -in my-server.csr -signkey my-server.key -out my-server.cer

Actions Like (0)

Mauricio Magnani Jr Jul 1, 2014 8:50 AM (in response to samuelfac)

Correct Answer 4. Re: Autenticação com Certificado no Apache 2.2 + Jboss 4.2.3 (HTTPS)

Boa Noite Samuel,

Pelo que entendi o seguinte cenário está funcionando:

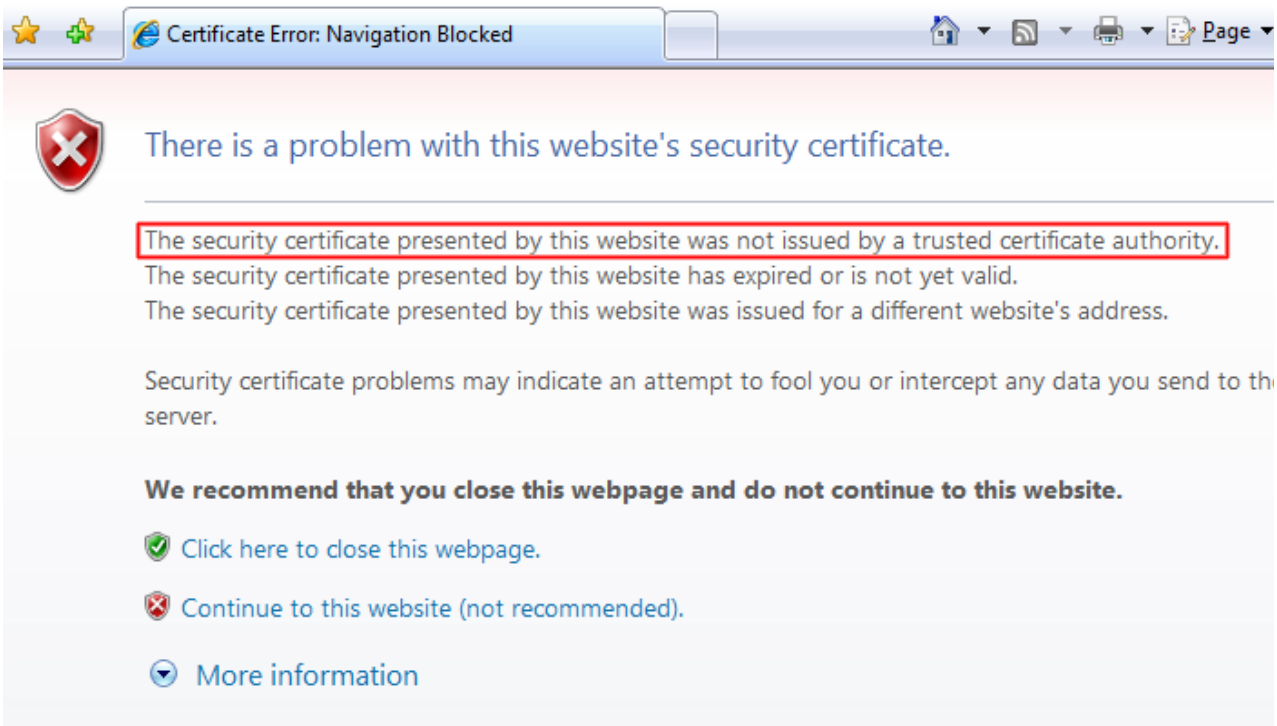
Client <--- HTTPS -----> JBoss 4.2.3

Você está tentando configurar o seguinte cenário:

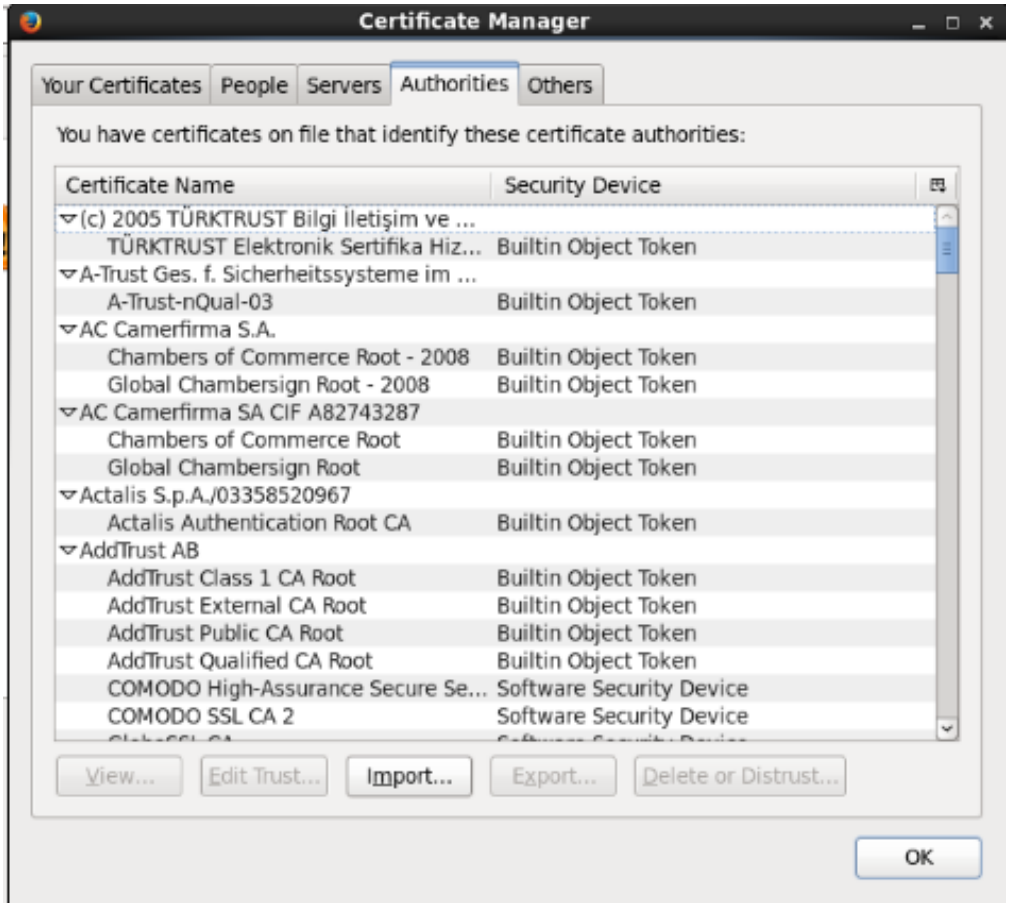
Client <--- HTTPS ---> Apache Web Server < ------ HTTPS -----> JBoss 4.2.3

O Cenário que você está tentanto configurar é conhecido como **Mutual Authentication** ou **Two-way Authentication** (http://en.wikipedia.org/wiki/Mutual_authentication). Para que essa estrutura possa funcionar de forma adequada, o certificado do client deve confiar no do servidor e vice-versa.

Primeiro vamos relembrar alguns conceitos... Quando um usuário utiliza o navegador para acessar uma aplicação (HTTPS), o Apache Web Server ou o JBoss irá fornecer um certificado. O navegador irá estabelecer uma conexão segura com o servidor. Isso é chamado de **One-way authentication**. Isso significa que o cliente vai verificar o identidade da aplicação, mas a aplicação não vai verificar a identidade do client. Se o certificado é auto-assinado, o navegador irá exibir um aviso a perguntar ao usuário se deve confiar ou não algo como a imagem abaixo:



No navegador existe uma lista de CA (autoridades certificadoras) que são confiáveis por padrão. Veja a lista no Firefox por exemplo:



Como vamos gerar os certificados auto assinados obviamente a nossa CA não estará presente nessa lista tornando o nosso certificado **"não confiável"**. O JDK também contém uma lista que geralmente está em **$JAVA_HOME/jre/lib/security/cacerts** e é nesse ponto que eu queria chegar

Acredito que você não esteja conseguindo pegar o certificado no Client quando utiliza o Apache Web Server como Proxy porque os certificados do JBoss (JVM) e do Apache não possuem uma relação de confiança ou seja, eles não são assinados pela mesma CA. Para que possamos estabelecer essa relação de confiança devemos criar uma autoridade certificadora e assinar os certificados do Apache e do JBoss.

1° Passo - Criar o Certificado da CA

Vamos criar um certificado auto assinado chamado **ca** e utilizar isso como a nossa autoridade certificadora. Como estou utilizando Linux criei um diretorio certs em /opt.

```
[root@mmagnani opt]# cd /opt/certs/

[root@mmagnani certs]# openssl req -new -newkey rsa -days 365 -x509 -keyout ca.key -out ca.crt

Generating a 2048 bit RSA private key
.....++++++
.....++++++
writing new private key to 'ca.key'
Enter PEM pass phrase:123456
Verifying - Enter PEM pass phrase:123456
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:BR
State or Province Name (full name) []:RIO DE JANEIRO
Locality Name (eg, city) [Default City]:RIO DE JANEIRO
Organization Name (eg, company) [Default Company Ltd]:JBUG BRASIL
Organizational Unit Name (eg, section) []:FORUM
Common Name (eg, your name or your server's hostname) []:mmagnani.csb
Email Address []:
```

Idetificar uma CA geralmente é uma tarefa simples. Basta encontrar o certificado em que o Owner e Issuer sejam os mesmos.

```
[root@mmagnani certs]# keytool -printcert -file ca.crt | head -n 2

Owner: CN=mmagnani.csb, OU=FORUM, O=JBUG BRASIL, L=RIO DE JANEIRO, ST=RIO DE JANEIRO, C=BR
Issuer: CN=mmagnani.csb, OU=FORUM, O=JBUG BRASIL, L=RIO DE JANEIRO, ST=RIO DE JANEIRO, C=BR
```

2° Passo - Criar o Certificado para o Apache Web Server

Vamos criar o certificado para o Apache Web Server e assina-lo com a nossa CA.

```
[root@mmagnani certs]# openssl genrsa -des3 -out httpd.key 1024

Generating RSA private key, 1024 bit long modulus
.....++++++
.....++++++
e is 65537 (0x10001)
Enter pass phrase for httpd.key:123456
Verifying - Enter pass phrase for httpd.key:123456
```

Agora vamos gerar o pedido para a assinatura:

```
[root@mmagnani certs]# openssl req -new -key httpd.key -out httpd.csr

Enter pass phrase for httpd.key:123456
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:BR
State or Province Name (full name) []:SAO PAULO
Locality Name (eg, city) [Default City]:SAO PAULO
Organization Name (eg, company) [Default Company Ltd]:MYCOMPANY
Organizational Unit Name (eg, section) []:TI
Common Name (eg, your name or your server's hostname) []:mmagnani.csb
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

O Common Name deve ser o nome (Nome / DNS) do servidor em que o Apache será utilizado.

3° Passo - Assinar o certificado com a nossa CA

Alguns arquivos devem ser criados:

```
# mkdir -p /opt/certs/demoCA/newcerts
# touch /opt/certs/demoCA/index.txt
# touch /opt/certs/demoCA/serial
# echo "01" > /opt/certs/demoCA/serial
```

Realize algumas configurações no arquivo /etc/pki/tls/openssl.cnf. Existem algumas coisas que devem ser opicionais para que os nossos testes sejam realizados com sucesso. O policy_match deve ficar como abaixo:

```
# For the CA policy
[ policy_match ]
countryName             = optional
stateOrProvinceName     = optional
organizationName        = optional
organizationalUnitName  = optional
commonName              = supplied
emailAddress            = optional
```

Precisamos também alterar o diretório padrão do OpenSSL.

```
# These are used by the TSA reply generation only.
#dir = /etc/pki/CA
dir       = ./opt/certs/demoCA            # TSA root directory
```

Finalmente assine o certificado utilizando o OpenSSL:

```
[root@mmagnani certs]# openssl ca -in httpd.csr -keyfile ca.key -cert ca.crt

Using configuration from /etc/pki/tls/openssl.cnf
Enter pass phrase for ca.key:123456
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 1 (0x1)
        Validity
            Not Before: Jun 27 03:34:22 2014 GMT
            Not After : Jun 27 03:34:22 2015 GMT
        Subject:
            countryName               = BR
            stateOrProvinceName       = SAO PAULO
            organizationName          = MYCOMPANY
            organizationalUnitName    = TI
            commonName                = mmagnani.csb
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                61:1A:60:C3:7A:89:B5:9D:B1:AE:57:92:46:A9:A7:A5:04:B4:B4:BD
            X509v3 Authority Key Identifier:
                keyid:0C:CA:47:DF:11:91:E3:D7:CC:98:C3:B5:E0:F1:4C:E8:F3:3C:74:62

Certificate is to be certified until Jun 27 03:34:22 2015 GMT (365 days)
Sign the certificate? [y/n]:y



1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=BR, ST=RIO DE JANEIRO, L=RIO DE JANEIRO, O=JBUG BRASIL,
OU=FORUM, CN=mmagnani.csb
        Validity
            Not Before: Jun 27 03:34:22 2014 GMT
            Not After : Jun 27 03:34:22 2015 GMT
        Subject: C=BR, ST=SAO PAULO, O=MYCOMPANY, OU=TI, CN=mmagnani.csb
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (1024 bit)
                Modulus:
                    00:cc:77:b4:f6:bb:87:bd:0c:e9:4f:8c:b4:d9:f0:
                    2c:04:80:6f:71:71:82:e1:aa:3e:e4:43:dc:7d:5f:
                    f4:72:a0:1a:cf:37:76:c2:43:77:d0:2a:d9:7e:29:
                    f8:24:94:c6:c4:0c:1e:58:9c:62:20:36:78:e1:13:
                    64:5f:a0:51:87:b3:44:f6:52:d9:f7:83:a3:fb:3e:
```

```
                c9:e9:45:3c:5e:d7:ee:36:43:c3:b7:96:66:57:f1:
                bc:52:ba:f3:10:d7:d8:bf:9a:50:8e:02:ca:d6:48:
                76:46:90:7f:c3:1b:50:a4:b2:48:b7:a2:8f:ef:e2:
                91:d4:41:12:32:3d:5f:6b:75
            Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                61:1A:60:C3:7A:89:B5:9D:B1:AE:57:92:46:A9:A7:A5:04:B4:B4:BD
            X509v3 Authority Key Identifier:
                keyid:0C:CA:47:DF:11:91:E3:D7:CC:98:C3:B5:E0:F1:4C:E8:F3:3C:74:62

    Signature Algorithm: sha1WithRSAEncryption
        3c:85:98:48:ed:d8:40:52:ec:e5:ab:82:7d:6b:c7:cd:ae:cf:
        1b:d0:d7:f9:fc:c8:1d:0d:1d:2d:45:bf:62:ec:cf:e4:f7:ba:
        64:f1:96:5c:58:fa:ad:4a:0c:91:c8:bd:f4:38:51:aa:50:92:
        8b:3a:6b:70:37:8d:f1:c6:21:a3:cb:7e:2a:ca:6a:7b:c0:76:
        77:69:4d:02:e6:3b:f7:0b:be:26:38:4a:25:ef:02:10:db:0a:
        74:9c:77:78:65:01:f4:2f:53:b8:70:87:3e:40:61:47:47:33:
        dd:f2:13:b2:5b:7c:97:3c:66:d2:1a:6b:15:ca:d2:f1:88:a8:
        ca:8a
-----BEGIN CERTIFICATE-----
MIICxjCCAi+gAwIBAgIBATANBgkqhkiG9w0BAQUFADB8MQswCQYDVQQGEwJCUjEX
MBUGA1UECAwOUklPIERFIEpBTkVJUk8xFzAVBgNVBAcMDlJJTyBERSBKQU5FSVJP
MRQwEgYDVQQKDAtKQlVHIEJSQVNJTDEOMAwGA1UECwwFRk9SVU0xFTATBgNVBAMM
DG1tYWduYW5pLmNzYjAeFw0xNDA2MjcwMzM0MjJaFw0xNTA2MjcwMzM0MjJaMFkx
CzAJBgNVBAYTAkJSMRIwEAYDVQQIDAlTQU8gUEFVTE8xEjAQBgNVBAoMCU1ZQ09N
UEFOWTELMAkGA1UECwwCVEkxFTATBgNVBAMMDG1tYWduYW5pLmNzYjCBnzANBgkq
hkiG9w0BAQEFAAOBjQAwgYkCgYEAzHe09ruHvQzpT4y02fAsBIBvcXGC4ao+5EPc
fV/0cqAazzd2wkN30CrZfin4JJTGxAweWJxiIDZ44RNkX6BRh7NE9lLZ94Oj+z7J
6UU8XtfuNkPDt5ZmV/G8UrrzENfYv5pQjgLK1kh2RpB/wxtQpLJIt6KP7+KR1EES
Mj1fa3UCAwEAAaN7MHkwCQYDVR0TBAIwADAsBglghkgBhvhCAQ0EHxYdT3BlblNT
TCBHZW5lcmF0ZWQgQ2VydGlmaWNhdGUwHQYDVR0OBBYEFGEaYMN6ibWdsa5Xkkap
p6UEtLS9MB8GA1UdIwQYMBaAFAzKR98RkePXzJjDteDxTOjzPHRiMA0GCSqGSIb3
DQEBBQUAA4GBADyFmEjt2EBS7OWrgn1rx82uzxvQ1/n8yB0NHS1Fv2Lsz+T3umTx
IlxY+q1KDJHIvfQ4UapQkos6a3A3jfHGIaPLfirKanvAdndpTQLmO/cLviY4SiXv
AhDbCnScd3hlAfQvU7hwhz5AYUdHM93yE7JbfJc8ZtIaaxXK0vGIqMqK
-----END CERTIFICATE-----
Data Base Updated
```

Observe o certificado assinado /opt/certs/demoCA/newcerts/01.pem

```
[root@mmagnani certs]# cat /opt/certs/demoCA/newcerts/01.pem

Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=BR, ST=RIO DE JANEIRO, L=RIO DE JANEIRO, O=JBUG BRASIL,
OU=FORUM, CN=mmagnani.csb
        Validity
            Not Before: Jun 27 03:34:22 2014 GMT
            Not After : Jun 27 03:34:22 2015 GMT
        Subject: C=BR, ST=SAO PAULO, O=MYCOMPANY, OU=TI, CN=mmagnani.csb
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (1024 bit)
                Modulus:
                    00:cc:77:b4:f6:bb:87:bd:0c:e9:4f:8c:b4:d9:f0:
                    2c:04:80:6f:71:71:82:e1:aa:3e:e4:43:dc:7d:5f:
                    f4:72:a0:1a:cf:37:76:c2:43:77:d0:2a:d9:7e:29:
                    f8:24:94:c6:c4:0c:1e:58:9c:62:20:36:78:e1:13:
                    64:5f:a0:51:87:b3:44:f6:52:d9:f7:83:a3:fb:3e:
                    c9:e9:45:3c:5e:d7:ee:36:43:c3:b7:96:66:57:f1:
                    bc:52:ba:f3:10:d7:d8:bf:9a:50:8e:02:ca:d6:48:
                    76:46:90:7f:c3:1b:50:a4:b2:48:b7:a2:8f:ef:e2:
                    91:d4:41:12:32:3d:5f:6b:75
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                61:1A:60:C3:7A:89:B5:9D:B1:AE:57:92:46:A9:A7:A5:04:B4:B4:BD
            X509v3 Authority Key Identifier:
                keyid:0C:CA:47:DF:11:91:E3:D7:CC:98:C3:B5:E0:F1:4C:E8:F3:3C:74:62

    Signature Algorithm: sha1WithRSAEncryption
        3c:85:98:48:ed:d8:40:52:ec:e5:ab:82:7d:6b:c7:cd:ae:cf:
        1b:d0:d7:f9:fc:c8:1d:0d:1d:2d:45:bf:62:ec:cf:e4:f7:ba:
        64:f1:96:5c:58:fa:ad:4a:0c:91:c8:bd:f4:38:51:aa:50:92:
        8b:3a:6b:70:37:8d:f1:c6:21:a3:cb:7e:2a:ca:6a:7b:c0:76:
        77:69:4d:02:e6:3b:f7:0b:be:26:38:4a:25:ef:02:10:db:0a:
        74:9c:77:78:65:01:f4:2f:53:b8:70:87:3e:40:61:47:47:33:
        dd:f2:13:b2:5b:7c:97:3c:66:d2:1a:6b:15:ca:d2:f1:88:a8:
        ca:8a
-----BEGIN CERTIFICATE-----
MIICxjCCAi+gAwIBAgIBATANBgkqhkiG9w0BAQUFADB8MQswCQYDVQQGEwJCUjEX
MBUGA1UECAwOUklPIERFIEpBTkVJUk8xFzAVBgNVBAcMDlJJTyBERSBKQU5FSVJP
MRQwEgYDVQQKDAtKQlVHIEJSQVNJTDEOMAwGA1UECwwFRk9SVU0xFTATBgNVBAMM
DG1tYWduYW5pLmNzYjAeFw0xNDA2MjcwMzM0MjJaFw0xNTA2MjcwMzM0MjJaMFkx
CzAJBgNVBAYTAkJSMRIwEAYDVQQIDAlTQU8gUEFVTE8xEjAQBgNVBAoMCU1ZQ09N
UEFOWTELMAkGA1UECwwCVEkxFTATBgNVBAMMDG1tYWduYW5pLmNzYjCBnzANBgkq
```

```
hkiG9w0BAQEFAAOBjQAwgYkCgYEAzHe09ruHvQzpT4y02fAsBIBvcXGC4ao+5EPc
fV/0cqAazzd2wkN30CrZfin4JJTGxAweWJxiIDZ44RNkX6BRh7NE9lLZ94Oj+z7J
6UU8XtfuNkPDt5ZmV/G8UrrzENfYv5pQjgLK1kh2RpB/wxtQpLJlt6KP7+KR1EES
Mj1fa3UCAwEAAaN7MHkwCQYDVR0TBAIwADAsBglghkgBhvhCAQ0EHxYdT3BlbINT
TCBHZW5lcmF0ZWQgQ2VydGlmaWNhdGUwHQYDVR0OBBYEFGEaYMN6ibWdsa5Xkkap
p6UEtLS9MB8GA1UdIwQYMBaAFAzKR98RkePXzJjDteDxTOjzPHRiMA0GCSqGSIb3
DQEBBQUAA4GBADyFmEjt2EBS7OWrgn1rx82uzxvQ1/n8yB0NHS1Fv2Lsz+T3umTx
llxY+q1KDJHIvfQ4UapQkos6a3A3jfHGIaPLfirKanvAdndpTQLmO/cLviY4SiXv
AhDbCnScd3hlAfQvU7hwhz5AYUdHM93yE7JbfJc8ZtIaaxXK0vGIqMqK
-----END CERTIFICATE-----
```

Para finalizar copie e renomei o certificado como abaixo:

```
[root@mmagnani certs]# cp /opt/certs/demoCA/newcerts/01.pem /opt/certs/httpd.crt
```

4° Passo - Criar os Certificados do JBoss

Agora vamos gerar keystore utilizando o keytool que é a ferramente padrão do JDK para criação e gerenciamento de certificados.

```
[root@mmagnani certs]# /usr/java/jdk1.7.0_60/bin/keytool -genkey -keystore jboss.jks -
storepass 123456 -keypass 123456 -keyalg RSA -validity 365 -alias jboss
What is your first and last name?
  [Unknown]:  MAURICIO MAGNANI
What is the name of your organizational unit?
  [Unknown]:  TI
What is the name of your organization?
  [Unknown]:  MY COMPANY
What is the name of your City or Locality?
  [Unknown]:  SAO PAULO
What is the name of your State or Province?
  [Unknown]:  SAO PAULO
What is the two-letter country code for this unit?
  [Unknown]:  BR
Is CN=MAURICIO MAGNANI, OU=TI, O=MY COMPANY, L=SAO PAULO, ST=SAO PAULO,
C=BR correct?
  [no]:  YES
```

```
[root@mmagnani certs]# /usr/java/jdk1.7.0_60/bin/keytool -list -v -keystore jboss.jks --alias
jboss
Enter keystore password:
Alias name: jboss
Creation date: Jun 27, 2014
Entry type: PrivateKeyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=MAURICIO MAGNANI, OU=TI, O=MY COMPANY, L=SAO PAULO, ST=SAO
PAULO, C=BR
Issuer: CN=MAURICIO MAGNANI, OU=TI, O=MY COMPANY, L=SAO PAULO, ST=SAO
PAULO, C=BR
Serial number: 6d31886d
Valid from: Fri Jun 27 22:52:59 BRT 2014 until: Sat Jun 27 22:52:59 BRT 2015
Certificate fingerprints:
  MD5:  7D:7E:F8:1F:48:91:7D:99:46:83:42:5C:07:CB:27:E1
  SHA1: 56:08:23:17:AB:A4:63:58:8E:F1:05:28:C9:76:6A:4E:13:3A:AB:92
  SHA256:
C2:0E:0D:34:E9:2B:22:F1:B8:09:52:6C:DB:B1:7E:4D:92:78:16:14:3C:66:AD:97:06:36:04:81:A7:1
  Signature algorithm name: SHA256withRSA
  Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 47 A8 00 3A 96 5E 14 43   4F 1F 99 10 6A A8 41 A2  G..:.^.CO...j.A.
0010: AF E9 3F 70                                        ..?p
]
]
```

```
[root@mmagnani certs]# /usr/java/jdk1.7.0_60/bin/keytool -certreq -keyalg RSA -alias jboss -
file jbosscert.csr -keystore jboss.jks
Enter keystore password:
```

```
[root@mmagnani certs]# openssl ca -in jbosscert.csr -keyfile ca.key -cert ca.crt
Using configuration from /etc/pki/tls/openssl.cnf
Enter pass phrase for ca.key:
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 2 (0x2)
        Validity
            Not Before: Jun 28 01:58:09 2014 GMT
            Not After : Jun 28 01:58:09 2015 GMT
        Subject:
            countryName               = BR
            stateOrProvinceName       = SAO PAULO
            organizationName          = MY COMPANY
            organizationalUnitName    = TI
            commonName                = MAURICIO MAGNANI
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                47:A8:00:3A:96:5E:14:43:4F:1F:99:10:6A:A8:41:A2:AF:E9:3F:70
            X509v3 Authority Key Identifier:
                keyid:0C:CA:47:DF:11:91:E3:D7:CC:98:C3:B5:E0:F1:4C:E8:F3:3C:74:62

Certificate is to be certified until Jun 28 01:58:09 2015 GMT (365 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 2 (0x2)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=BR, ST=RIO DE JANEIRO, L=RIO DE JANEIRO, O=JBUG BRASIL,
OU=FORUM, CN=mmagnani.csb
        Validity
            Not Before: Jun 28 01:58:09 2014 GMT
            Not After : Jun 28 01:58:09 2015 GMT
        Subject: C=BR, ST=SAO PAULO, O=MY COMPANY, OU=TI, CN=MAURICIO MAGNANI
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:de:eb:db:eb:cf:23:21:ed:4e:2a:e5:6c:57:13:
                    0d:45:0f:db:79:7f:d7:21:fb:80:f4:fc:43:37:5f:
                    42:12:14:d7:6c:0f:d5:4b:4f:88:a2:b9:ae:4b:b5:
                    c5:7d:2e:8f:30:77:f0:91:55:be:b1:e8:c1:6f:96:
                    a0:16:41:5d:ea:d4:12:c7:d8:cd:ec:db:35:22:62:
```

```
                47:02:5a:0b:2a:ac:a6:50:69:c0:19:71:ab:6b:c5:
                df:72:a0:2c:25:06:77:b3:37:24:c0:af:2d:13:6d:
                c5:93:0c:9c:00:58:10:a4:3e:92:84:25:a6:10:10:
                d7:88:9d:24:30:0e:26:19:6b:80:40:39:31:2b:f5:
                a7:1f:ae:a9:27:14:bd:66:e7:c3:76:35:e0:ac:68:
                88:fc:fd:89:0c:69:3b:06:f5:08:45:14:4a:64:da:
                09:74:7d:05:41:77:cb:93:4c:bd:11:54:ea:fd:ce:
                b6:69:34:0d:7e:29:10:a8:c7:bf:cb:8c:0a:c6:4a:
                8d:76:27:99:0a:60:f6:29:f7:f1:1e:10:cb:08:d6:
                8c:26:81:dc:cd:d4:6f:00:5c:37:4a:15:f0:d2:57:
                c4:42:9b:4f:f7:64:67:2d:fe:cd:86:eb:c3:1b:d5:
                b1:c8:14:55:d8:92:a4:7d:c2:5d:d1:7e:9c:5c:60:
                a5:d5
            Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                47:A8:00:3A:96:5E:14:43:4F:1F:99:10:6A:A8:41:A2:AF:E9:3F:70
            X509v3 Authority Key Identifier:
                keyid:0C:CA:47:DF:11:91:E3:D7:CC:98:C3:B5:E0:F1:4C:E8:F3:3C:74:62

    Signature Algorithm: sha1WithRSAEncryption
        e4:7e:cb:98:58:70:9b:9c:5e:e3:26:b0:9f:74:16:7b:ce:be:
        1b:67:dd:56:36:ac:f0:41:68:5b:97:9a:2a:7a:9f:91:33:1d:
        54:07:a6:dd:9d:2a:d5:10:df:f1:bd:d6:76:c9:bd:3e:aa:57:
        80:af:d1:fd:db:26:89:3f:d7:e9:26:06:ff:ea:b9:05:d9:5d:
        15:98:16:40:30:06:d9:c6:ba:1e:55:54:86:a8:3c:3b:77:a3:
        82:2f:80:b4:54:90:ec:73:80:a5:bf:d0:58:79:20:9a:d6:dc:
        ff:ea:28:07:9a:fb:66:c7:e2:12:3d:36:7f:76:cd:d4:fd:96:
        98:56
-----BEGIN CERTIFICATE-----
MIIDTzCCArigAwIBAgIBAjANBgkqhkiG9w0BAQUFADB8MQswCQYDVQQGEwJCUjEX
MBUGA1UECAwOUklPIERFIEpBTkVJUk8xFzAVBgNVBAcMDlJJTyBERSBKQU5FSVJP
MRQwEgYDVQQKDAtKQlVHIEJSQVNJTDEOMAwGA1UECwwFRk9SVU0xFTATBgNVBAMM
DG1tYWduYW5pLmNzYjAeFw0xNDA2MjgwMTU4MDlaFw0xNTA2MjgwMTU4MDlaMF4x
CzAJBgNVBAYTAkJSMRIwEAYDVQQIEwlTQU8gUEFVTE8xEzARBgNVBAoTCk1ZIENP
TVBBTlkxCzAJBgNVBAsTAlRJMRkwFwYDVQQDExBNQVVSSUNJTyBNQUdOQU5JMIIB
IjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA3uvb688jIe1OKuVsVxMNRQ/b
eX/XIfuA9PxDN19CEhTXbA/VS0+IormuS7XFfS6PMHfwkVW+sejBb5agFkFd6tQS
x9jN7Ns1ImJHAloLKqymUGnAGXGra8XfcqAsJQZ3szckwK8tE23FkwycAFgQpD6S
hCWmEBDXiJ0kMA4mGWuAQDkxK/WnH66pJxS9ZufDdjXgrGiI/P2JDGk7BvUIRRRK
ZNoJdH0FQXfLk0y9EVTq/c62aTQNfikQqMe/y4wKxkqNdieZCmD2KffxHhDLCNaM
JoHczdRvAFw3ShXw0lfEQptP92RnLf7NhuvDG9WxyBRV2JKkfcJd0X6cXGCl1QID
AQABo3sweTAJBgNVHRMEAjAAMCwGCWCGSAGG+EIBDQQfFh1PcGVuU1NMIEdlbmVy
YXRlZCBDZXJ0aWZpY2F0ZTAdBgNVHQ4EFgQUR6gAOpZeFENPH5kQaqhBoq/pP3Aw
HwYDVR0jBBgwFoAUDMpH3xGR49fMmMO14PFM6PM8dGIwDQYJKoZIhvcNAQEFBQAD
gYEA5H7LmFhwm5xe4yawn3QWe86+G2fdVjas8EFoW5eaKnqfkTMdVAem3Z0q1RDf
8b3Wdsm9PqpXgK/R/dsmiT/X6SYG/+q5BdldFZgWQDAG2ca6HlVUhqg8O3ejgi+A
tFSQ7HOApb/QWHkgmtbc/+ooB5r7ZsfiEj02f3bN1P2WmFY=
-----END CERTIFICATE-----
```

```
Data Base Updated
```

```
[root@mmagnani certs]# cp demoCA/newcerts/02.pem jbossraw.crt
grep -A 50 "BEGIN CERTIFICATE" jbossraw.crt > jbosscert.crt
keytool -printcert -file jbosscert.crt | head -n 2
keytool -import -v -trustcacerts -alias ca -file ca.crt -keystore jboss.jks
```

```
[root@mmagnani certs]# /usr/java/jdk1.7.0_60/bin/keytool -printcert -file jbosscert.crt | head -n
2
Owner: CN=MAURICIO MAGNANI, OU=TI, O=MY COMPANY, ST=SAO PAULO, C=BR
Issuer: CN=mmagnani.csb, OU=FORUM, O=JBUG BRASIL, L=RIO DE JANEIRO, ST=RIO
DE JANEIRO, C=BR
[root@mmagnani certs]# /usr/java/jdk1.7.0_60/bin/keytool -import -v -trustcacerts -alias ca -file
ca.crt -keystore jboss.jks
Enter keystore password:
Owner: CN=mmagnani.csb, OU=FORUM, O=JBUG BRASIL, L=RIO DE JANEIRO, ST=RIO
DE JANEIRO, C=BR
Issuer: CN=mmagnani.csb, OU=FORUM, O=JBUG BRASIL, L=RIO DE JANEIRO, ST=RIO
DE JANEIRO, C=BR
Serial number: 966b6e2d25f81ec7
Valid from: Fri Jun 27 00:01:13 BRT 2014 until: Sat Jun 27 00:01:13 BRT 2015
Certificate fingerprints:
  MD5:  A7:D1:5E:BB:8D:C1:4D:BB:43:81:D7:0B:F9:71:18:40
  SHA1: B8:87:20:29:D9:BC:98:FD:AB:C8:21:56:0E:1E:4A:1D:81:95:3D:08
  SHA256:
17:ED:CB:5E:12:68:B8:E7:3E:2B:0B:0E:59:28:02:54:3C:4F:D8:51:B4:05:42:1B:4F:2A:34:63:8C:A
  Signature algorithm name: SHA1withRSA
  Version: 3
Extensions:
#1: ObjectId: 2.5.29.35 Criticality=false
AuthorityKeyIdentifier [
KeyIdentifier [
0000: 0C CA 47 DF 11 91 E3 D7   CC 98 C3 B5 E0 F1 4C E8  ..G.....L.
0010: F3 3C 74 62                                        .<tb
]
]

#2: ObjectId: 2.5.29.19 Criticality=false
BasicConstraints:[
  CA:true
  PathLen:2147483647
]

#3: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 0C CA 47 DF 11 91 E3 D7   CC 98 C3 B5 E0 F1 4C E8  ..G.....L.
0010: F3 3C 74 62                                        .<tb
]
]

Trust this certificate? [no]:  yes
Certificate was added to keystore
[Storing jboss.jks]
```

[root@mmagnani certs]# /usr/java/jdk1.7.0_60/bin/keytool -import -v -alias jbosscert -file jbosscert.crt -keystore jboss.jks
Enter keystore password:
Certificate was added to keystore
[Storing jboss.jks]

[root@mmagnani certs]# /usr/java/jdk1.7.0_60/bin/keytool -list -keystore jboss.jks
Enter keystore password:

Keystore type: JKS
Keystore provider: SUN

Your keystore contains 3 entries

ca, Jun 27, 2014, trustedCertEntry,
Certificate fingerprint (SHA1):
B8:87:20:29:D9:BC:98:FD:AB:C8:21:56:0E:1E:4A:1D:81:95:3D:08
jboss, Jun 27, 2014, PrivateKeyEntry,
Certificate fingerprint (SHA1):
56:08:23:17:AB:A4:63:58:8E:F1:05:28:C9:76:6A:4E:13:3A:AB:92
jbosscert, Jun 27, 2014, trustedCertEntry,
Certificate fingerprint (SHA1):
8B:CC:61:FD:23:70:8A:E6:50:61:56:0B:0F:08:83:ED:91:5F:41:2A

Tenho que organizar essa resposta e colocar as configurações.

Servlet Utilizada nos Testes

```
01. @WebServlet(name = "X509Test", urlPatterns = {"/X509Test"})
02. public class X509Test extends HttpServlet {
03.
04.
05.     @Override
06.     public void doGet(HttpServletRequest req, HttpServletResponse res) throws
07.       IOException {
08.
09.     res.setContentType("text/plain");
10.     PrintWriter out = res.getWriter();
11.
12.
13.     X509Certificate[] certs = (X509Certificate[]) req.getAttribute("java.serv
14.
15.     if (certs != null) {
16.       for (int i = 0; i < certs.length; i++) {
17.         out.println("Client Certificate [" + i + "] = " + certs[i].toString()
18.       }
19.     } else {
20.       if ("https".equals(req.getScheme())) {
21.         out.println("This was an HTTPS request, " + "but no client certificat
22.       } else {
23.         out.println("This was not an HTTPS request, " + "so no client certifi
24.       }
25.     }
26.   }
27.
28.
29. }
```

Continua...

Actions Like (2)

Adriano Schmidt Jun 30, 2014 1:53 AM (in response to samuelfac)

5. Re: Autenticação com Certificado no Apache 2.2 + Jboss 4.2.3 (HTTPS)

Fala Samuel.. conseguiu avançar nesse assunto? abraço!

Actions Like (1)

Mauricio Magnani Jr Jun 30, 2014 9:00 AM (in response to samuelfac)

6. Re: Autenticação com Certificado no Apache 2.2 + Jboss 4.2.3 (HTTPS)

Bom dia Samuel,

Hoje termino essa resposta

Abs

Actions Like (1)

samuelfac Jul 1, 2014 9:09 AM (in response to Adriano Schmidt)

7. Re: Autenticação com Certificado no Apache 2.2 + Jboss 4.2.3 (HTTPS)

Bom dia Pessoal, desculpa a demora de responder, só hj voltei a mexer com isso.

Adriano Schmidt :
Alterado de:
SLVerifyClient optional
por:
SSLVerifyClient require

Erro:
ssl_error_handshake_failure_alert

Sim foi gerado com o openssl

Mauricio Magnani Jr :
vc entendeu direitinho meu ambiente... aguardo o restante da configuração.

Obrigado.

Actions Like (0)

samuelfac Jul 2, 2014 8:37 AM (in response to Mauricio Magnani Jr)

8. Re: Autenticação com Certificado no Apache 2.2 + Jboss 4.2.3 (HTTPS)

Bom dia.

Teria como vc dar continuidade?

Grato.

Actions Like (0)