CERTIFICAÇÃO DIGITAL COM JAVA.

Cléber da Silveira.

Certificado Digital x Assinatura Digital.

DBSERVER

- A certificação digital é uma tecnologia que provê um mecanismo seguro, capaz de fornecer autenticidade, confidencialidade e integridade a toda e qualquer informação eletrônica, sempre por meio de um certificado digital.

- Por outro lado, a assinatura digital faz uso dos mecanismos dispostos pela certificação digital, juntamente com algoritmos criptográficos baseados em chaves públicas e privadas (chaves assimétricas).

- Todos estes recursos são utilizados com vistas à segurança, para poder garantir legalmente a autenticidade e a validade da informação.

Certificado Digital x Assinatura Digital.

DBSERVER

- A certificação digital permite aplicar seus mecanismos de segurança a outros fins que não somente a assinatura digital.
- É possível também utilizá-la como mecanismo de criptografia simétrica que compartilha a mesma chave para cifrar e decifrar informações eletrônicas.
- Assim, é possível dizer que a assinatura digital faz uso dos dispositivos da certificação digital para realizar seus objetivos, deixando claras suas distinções.

Hierarquia da cadeia de autoridade certificadora.

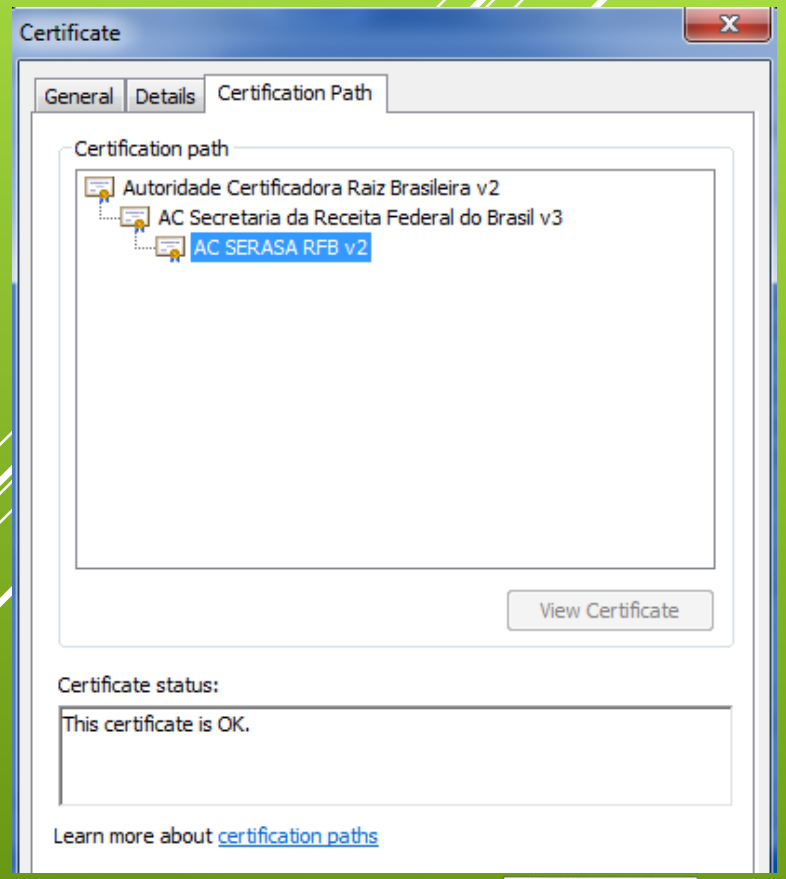
DBSERVER

AC – Raiz

No Brasil, o órgão responsável pelas políticas e normas técnicas e operacionais para a certificação digital é o ICP-Brasil (Infraestrutura de Chaves Públicas Brasileiras).

AC - Autoridade Certificadora

Uma Autoridade Certificadora (AC) é uma entidade, pública ou privada, subordinada à hierarquia da ICP-Brasil, responsável por emitir, distribuir, renovar, revogar e gerenciar certificados digitais.



Criptografia Assimétrica.

DBSERVER

- A criptográfica assimétrica é constituída pela chave pública e pela chave privada.
- A chave utilizada para criptografar é diferente da chave para de criptografar.
- A chave pública pode ser conhecida por todos e é utilizada para cifrar o documento claro. Por sua vez, a chave privada deve permanecer secreta e é utilizada para decifrar o documento cifrado.

Protocolo e Layer's.

DBSERVER

- **Por padrão para comunicação e autenticação com o certificado digital o protocolo de comunicação é o HTTPS e os layers de comunicação são:**

1. **SSL 3.0 – Nativo em JDK7u75 e inferiores e opcional em JDK8.**
2. **TLS 1.2 – Nativo em JDK8u1 e superiores e opcional em JDK7.**

* Para habilitar o SSL 3 em no JDK8u1 em JRE_HOME/lib/security/java.security informar jdk.tls.disabledAlgorithms=SSLv3

Como nesse momento vamos abordar JEE7 iremos trabalhar com o SSL3.

Java e Certificados Digitais.

DBSERVER

- **KeyStore** - É o repositório do Java onde você armazena seus certificados digitais, ou seja, suas assinaturas, as credencias que confirmam sua identidade.
- **TrustStore** - É o repositório de certificados de terceiros, que confiamos.

- **Cacert** - É um certificado de terceiros localizado na TrustStore.

Exemplo de importação de um arquivo de certificado digital.

```
C:\Program Files\Java\jdk1.7.0_79\jre\bin>keytool -importkeystore -srckeystore "C:\Users\clebers\Desktop\Certificado Digital\certificadopfx.pfx" -srcstoretype pkcs12
Enter destination keystore password:
Re-enter new password:
Enter source keystore password:
Entry for alias lojas sheldon farroupilha ltda me:23381596000112 successfully imported.
Import command completed:  1 entries successfully imported, 0 entries failed or cancelled
```

Mais em:

docs.oracle.com/javase/7/docs/technotes/tools/windows/keytool.html

Hands on com SSL.

DBSERVER

- Agora iremos gerar um Cacert através de um Handshake entre nosso cliente Java e um servidor HTTPS/SSL para recebermos suas chaves públicas.
- Configurando o pacote javax.net.ssl para estabelecer a conexão SSL.
- **Certificado tipo A1 -** Pode ser armazenado como um arquivo local, dispensando o uso de cartões inteligentes ou tokens.
- ***Criptografia Hash SHA1-*** 20 bytes.
- **Algoritimo de assinatura – RSA**

Hands on Assinando Documentos XML's.

DBSERVER

- **Transform -** javax.xml.crypto.dsig.Transform.ENVELOPED - http://www.w3.org/2000/09/xmldsig#enveloped-signature (XML Signature Syntax and Processing Version 2.0)
- **Canonical XML -** http://www.w3.org/TR/2001/REC-xml-c14n-20010315
- **PKCS#12** - Chave pública e chave privada.
- **Certificado tipo A1 -** Pode ser armazenado como um arquivo local, dispensando o uso de cartões inteligentes ou tokens.
- **Arquivo a ser assinado -** org.w3c.dom.Document
- ***Criptografia Hash SHA1-*** 20 bytes.
- **Algoritimo de assinatura – RSA -** http://www.w3.org/2000/09/xmldsig#rsa-sha1
- **Wrapper – javax.xml. crypto.dsig.XMLSignature**

Hands on Assinando Documentos PDF's.

DBSERVER

- **PKCS#12** - Chave pública e chave privada.
- **Certificado tipo A1 -** Pode ser armazenado como um arquivo local, dispensando o uso de cartões inteligentes ou tokens.
- **Arquivo a ser assinado -** Portable Document Format.
- ***Criptografia Hash SHA1-*** 20 bytes.
- **Algoritimo de assinatura – RSA -** http://www.w3.org/2000/09/xmldsig#rsa-sha1
- **Wrapper – IText (itextpdf.com)**

Hands on - Configurando JBoss com HTTPS, SSL e JAX-WS com policy.

DBSERVER

- **PKCS#12** - Chave pública e chave privada.
- **SSLv**3.
- **Security Real Undertow.**
- **RSA**
- **HTTPS**
- **WebServices JAX-WS com** TransportToken obrigatório.
- **JBoss.**

JBoss™

OBRIGADO.

Cléber da Silveira.