SHIVA

Procedimiento para Conectar al CALIPSO Seguro en Desarrollo

 

# Hitorial de Cambios

| Fecha | Versión | Documento/s asociado/s | Modificado Por | Descripción  (Nota: agregue una breve descripción de la modificación efectuada) |
| --- | --- | --- | --- | --- |
| 02/01/2015 | 1.0 |  | Leandro Garayoa | Versión Inicial |
|  |  |  |  |  |

Tabla de Contenidos

[Hitorial de Cambios 1](#_Toc408498457)

[1 Introducción 3](#_Toc408498458)

[2 Procedimiento 3](#_Toc408498459)

[2.1 Modificar el archivo de propiedad 3](#_Toc408498460)

[2.2 Obtención de Certificados 3](#_Toc408498461)

[2.3 Importación de Certificados 5](#_Toc408498462)

[2.4 Link al Keystore para su utilización en Spring (opcional) 5](#_Toc408498463)

# Introducción

El presente documento tiene como objetivo principal, presentar el procedimiento para conectar correctamente al Calipso de Telecom situado en dwin0369

# Procedimiento

## Modificar el archivo de propiedad

Sobre el archivo shiva\_desa\_interno.properties, se deberá agregar la letra **s** al ‘http’ en la propiedad

Antes

ws.url.calipsoCobranzas=http://dwin0369/Web\_Ev2\_int/CobranzasWS.php

Ahora

ws.url.calipsoCobranzas=http**s**://dwin0369/Web\_Ev2\_int/CobranzasWS.php

## Obtención de Certificados

Se debe obtener los certificados del Servidor dwin0369. Para eso, se debe ejecutar el siguiente comando dentro de un equipo Linux o cualquier equipo que tenga conexión a ese servidor y la aplicación openssl:

openssl s\_client -showcerts –connect dwin0369:443 </dev/null

Esto nos traerá los certificados del servidor. Un extracto de la salida se puede apreciar en el siguiente ejemplo:

[z306015@dlnx0161 ~]$ openssl s\_client -showcerts -connect dwin0369:443  </dev/null

CONNECTED(00000003)

depth=1 DC = ar, DC = com, DC = telecom, DC = arg, DC = telecom, CN = IssuingCATelecom

verify error:num=20:unable to get local issuer certificate

verify return:0

---

Certificate chain

0 s:/C=AR/ST=Buenos Aires/L=Buenos Aires/O=Telecom Argentina SA/OU=Datacenter/CN=dwin0369

   i:/DC=ar/DC=com/DC=telecom/DC=arg/DC=telecom/CN=IssuingCATelecom

-----BEGIN CERTIFICATE-----

MIIGIjCCBQqgAwIBAgIKU+8MqgAAAABSRzANBgkqhkiG9w0BAQUFADCBizESMBAG

CgmSJomT8ixkARkWAmFyMRMwEQYKCZImiZPyLGQBGRYDY29tMRcwFQYKCZImiZPy

...

Cada certificado está entre un -----BEGIN CERTIFICATE----- y -----END CERTIFICATE-----. Entonces, hay que exportar cada una de ellas a un archivo \*.cer. Por ejemplo: el siguiente es un certificado de dwin0369.

-----BEGIN CERTIFICATE-----

MIIGIjCCBQqgAwIBAgIKU+8MqgAAAABSRzANBgkqhkiG9w0BAQUFADCBizESMBAG

CgmSJomT8ixkARkWAmFyMRMwEQYKCZImiZPyLGQBGRYDY29tMRcwFQYKCZImiZPy

LGQBGRYHdGVsZWNvbTETMBEGCgmSJomT8ixkARkWA2FyZzEXMBUGCgmSJomT8ixk

ARkWB3RlbGVjb20xGTAXBgNVBAMTEElzc3VpbmdDQVRlbGVjb20wHhcNMTQxMjA1

MTIzNjE3WhcNMTYxMjA0MTIzNjE3WjCBgjELMAkGA1UEBhMCQVIxFTATBgNVBAgT

DEJ1ZW5vcyBBaXJlczEVMBMGA1UEBxMMQnVlbm9zIEFpcmVzMR0wGwYDVQQKExRU

ZWxlY29tIEFyZ2VudGluYSBTQTETMBEGA1UECxMKRGF0YWNlbnRlcjERMA8GA1UE

AxMIZHdpbjAzNjkwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCoh9NS

Y/CYGjZgSxYo4UlkoA53tLNPEBM3YO/uu2SLnfxGAmzo1leh5TeUS9ZI9yG0jWjx

FzWX7XinTzQI4gHMOnQZ7Mu1k4PeiXi8//7ZS0cvG8MwUmAnwrGMEkDB5nSZq4Ku

9+mIkOvPZ4aNCLhA/uorZ3FZeD7uSfmJTYY8r5Z12om7p9ANdq+1S3S/WN0+0W56

AdbBqcNKfGvRwMl2jSc9UInGf3MnTBKMNdGK+zuX5Rih4eU4WVBSNhcWlkbklfi7

Ip2XV/nyycXeMiFV+IYY6h9DXp273QXPNwa/OaBaFCEd7qd1G0/qsptWPJ1fcCwx

Efu8xylir6znXdixAgMBAAGjggKNMIICiTALBgNVHQ8EBAMCBaAwHQYDVR0lBBYw

FAYIKwYBBQUHAwIGCCsGAQUFBwMBMHgGCSqGSIb3DQEJDwRrMGkwDgYIKoZIhvcN

AwICAgCAMA4GCCqGSIb3DQMEAgIAgDALBglghkgBZQMEASowCwYJYIZIAWUDBAEt

MAsGCWCGSAFlAwQBAjALBglghkgBZQMEAQUwBwYFKw4DAgcwCgYIKoZIhvcNAwcw

HQYDVR0OBBYEFPf/DrzbHYPO3KTBHRGEQ+Ot0Q81MB8GA1UdIwQYMBaAFFsUBp6W

gg/y9SoQDnBWaOqF6CxIMHgGA1UdHwRxMG8wbaBroGmGMmh0dHA6Ly9wa2kudGVs

ZWNvbS5jb20uYXIvQ1JML0lzc3VpbmdDQVRlbGVjb20uQ1JMhjNodHRwOi8vcGtp

Mi50ZWxlY29tLmNvbS5hci9DUkwvSXNzdWluZ0NBVGVsZWNvbS5DUkwwgb0GCCsG

AQUFBwEBBIGwMIGtMD4GCCsGAQUFBzAChjJodHRwOi8vcGtpLnRlbGVjb20uY29t

LmFyL0FJQS9Jc3N1aW5nQ0FUZWxlY29tLmNydDA/BggrBgEFBQcwAoYzaHR0cDov

L3BraTIudGVsZWNvbS5jb20uYXIvQUlBL0lzc3VpbmdDQVRlbGVjb20uY3J0MCoG

CCsGAQUFBzABhh5odHRwOi8vcGtpLnRlbGVjb20uY29tLmFyL09DU1AwPgYJKwYB

BAGCNxUHBDEwLwYnKwYBBAGCNxUIgsmoAIOUwCqC6YE5hrnBGobA0weBaIHt7yaG

y/9GAgFkAgEHMCcGCSsGAQQBgjcVCgQaMBgwCgYIKwYBBQUHAwIwCgYIKwYBBQUH

AwEwDQYJKoZIhvcNAQEFBQADggEBAI3S1W5z60jQ6crDXfv4EqtbvWkotDx6e8QS

M4tuvkZDXZkVaFPvfcVsgKZKSwF4xePq+yUtpHgWO/QwD465oW/PHK+ObMYKhrh5

Ed7p8SlG4Vdvrv4ed/o+swITRCA2Rm23OTJXo9BcEq9ry0hmguueplr3IGREZo/M

JC1HaoV6PKGRGGgzfrdRls53dEki2b+r+89K0GakUhqU5QULjKQmR8kDCYorU+9T

MvyLDgsZtj80MarR3uDW3Cb9kcbDE9p3sOJ7R5FrHWANrfVLQXTqWhpzD+0G+iRq

7m8GCQyyl4yi7USVkcZKzM+11JPhm/2RxYqErAWhioPQKNQ0QlA=

-----END CERTIFICATE-----

Copiarlo todo a un archivo calipso.cer. Continuar con los demás certificados como el Calipso\_root.

## Importación de Certificados

Estos certificados bajados del servidor dwin0369, hay que importarlos a un keystore. La aplicación tiene un keystore generado por Interconexiones en la carpeta /home/z306015/certificates/Shiva.keystore. Por las dudas , para no modificarlo, se puede crear una copia en otra carpeta o hacer un backup de este archivo

Para la importación de los certificados se debe utilizar el siguiente comando:

keytool -importcert -file <archivo.cer> -keystore Shiva.keystore

Hay que reemplzar <archive.cer> por los certificados que hemos creado en el punto anterior. La importación requiere de ingresar la clave del keystore. Ejemplo: Tendremos que importar 4 certificados al keystore.

keytool -importcert -alias dev\_calipso -file calipso.cer -keystore Shiva.keystore

Password: pass (shiva2014)

Certificate was added to keystore

Una vez importados los certificados, se puede revisarlos con el siguiente comando:

keytool -list -keystore Shiva.keystore

## Link al Keystore para su utilización en Spring (opcional)

Primero, hay que entender que el Keystore de JBoss es independiente del Keystore que utiliza Spring. Sin embargo, JBoss y Spring puede utilizar el mismo Keystore sin problemas. Eso es porque Spring no tiene acceso al Keystore de Jboss. Por ende, Spring busca el Keystore por defecto del JDK. El certificado por defecto que utilizan todas las aplicaciones si no se especifica el Keystore es el JAVA\_HOME/jre/lib/security/cacerts.

El cacerts incluye varios certificados por defecto. Para no modificarlo, se lo puede renombrar a cacerts.bak, y después crear un enlance simbólico al Keystore que queremos utilizar.

El comando para hacer el link es el siguiente:

ln -s /home/z306015/certificates/Shiva.Keystore cacerts

Hay que crear este link dentro de la ruta JAVA\_HOME/jre/lib/security

Entonces Spring y Java levantará el cacerts que apunta a Shiva.Keystore con los certificados importados.