**Pasos técnicos para la actualización de los certificados digitales para la plataforma wso2 y de los servicios que utilizan esta plataforma.**

**Servidores involucrados.**

|  |  |  |  |
| --- | --- | --- | --- |
| SERVIDOR | TIPO | PIRVATE-KEY / CERTIFICATE | SERVICIO |
| srv-vhaproxyfe01 | pem | servicios.credicard.com.ve.pem | HAProxy |
|  | pem | vhaproxy.pem |  |
| srv-vhaproxyfe02 | pem | servicios.credicard.com.ve.pem | HAProxy |
|  | pem | vhaproxy.pem |  |
| srv-vhaproxybe01 | pem | slbbe.credicard.com.ve.pem | HAProxy |
|  | pem | vhaproxy.pem |  |
| srv-vhaproxybe02 | pem | slbbe.credicard.com.ve.pem | HAProxy |
|  | pem | vhaproxy.pem |  |
| srv-visprd01 | keystore / truestore | is.credicard.com.ve | wso2 |
| srv-visprd02 | keystore / truestore | is.credicard.com.ve | wso2 |
| srv-vesbadmprd01 | keystore / truestore | esb.credicard.com.ve | wso2 |
| srv-vesbprd01 | keystore / truestore | esb.credicard.com.ve | wso2 |
| srv-vesbprd02 | keystore / truestore | esb.credicard.com.ve | wso2 |
| srv-vesbprd03 | keystore / truestore | esb.credicard.com.ve | wso2 |
| srv-vdasprd01 | keystore / truestore | das.credicard.com.ve | wso2 |
| srv-vmpos01 | keystore | wsmpos.credicard.com.ve | JBOSS |
| srv-vmpos02 | keystore | wsmpos.credicard.com.ve | JBOSS |
| srv-vjbossprd01 | keystore | ws.credicard.com.ve | JBOSS |
| srv-vjbossprd02 | keystore | ws.credicard.com.ve | JBOSS |

Para identificar en cada servidor cual es el formato del certificado y si utiliza clave, se debe saber qué servicio es el que corre en el servidor (HAPRoxy, JBOSS, WSO2) y de esta forma buscar en sus archivos de configuración dicha información.

El personal de seguridad debe suministrar el CRT y el KEY de los siguiente certificados, al igual que el CRT público de la CA.

* slbbe.credicard.com.ve
* das.credicard.com.ve
* esb.credicard.com.ve
* is.credicard.com.ve
* ws.credicard.com.ve
* serviciosbancos.credicard.com.ve
* wsmpos.credicard.com.ve
* feclientservices.credicard.com.ve
* feservices.credicard.com.ve
* beservices.credicard.com.ve
* servicios.credicard.com.ve

Soporte de Aplicaciones en Línea debe suministrar el archivo de configuración “.cnf” desde donde con openssl crean los “CRS” de los certificados.

Los únicos certificados que deben utilizar la configuracion de “.cnf” son:

* slbbe.credicard.com.ve
* das.credicard.com.ve
* esb.credicard.com.ve
* is.credicard.com.ve
* ws.credicard.com.ve
* serviciosbancos.credicard.com.ve
* wsmpos.credicard.com.ve

**A continuación se detalla los pasos técnicos:**

En los servidores “**srv-vhaproxyfe01 y srv-vhaproxyfe02**” copiar el certificado ”servicios.credicard.com.ve.pem” en la ruta “/tmp”.

Realizar una copia del certificado actual.

|  |
| --- |
| cp -p /etc/ssl/servicios.credicard.com.ve.pem /etc/ssl/BackUp-Certs/ |

El formato PEM, debe tener el key del certificado, el crt del certificado y el crt público de la CA, para generarlo es asi.

|  |
| --- |
| [root@srv-vhaproxyfe02]:/>cat servicios.credicard.com.ve.key servicios.credicard.com.ve.crt ca.crt > servicios.credicard.com.ve.pem |

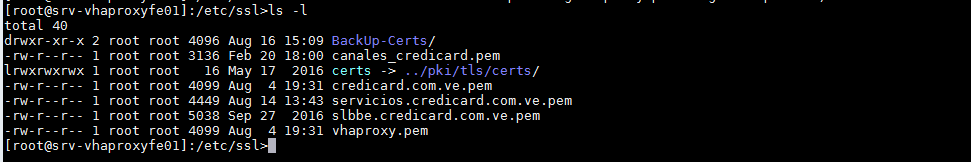
Antes de realizar la copia se debe verificar que el respaldo anterior fuera exitoso, Copiar el nuevo certificado.

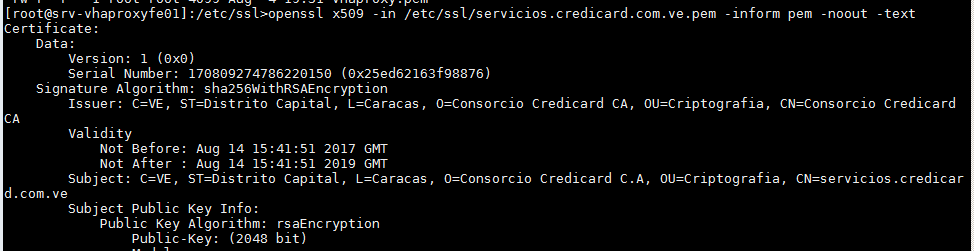
|  |
| --- |
| cp -p /tmp/servicios.credicard.com.ve.pem /etc/ssl/ |

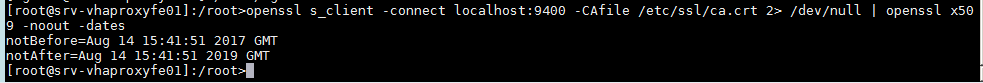
Reiniciar el servicio de HAProxy.

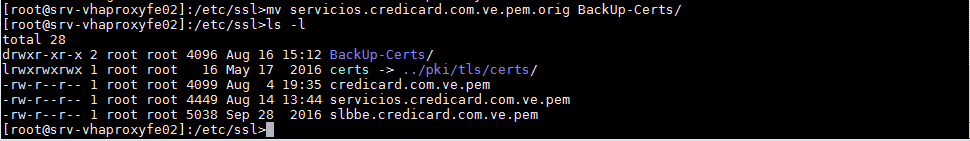
|  |
| --- |
| /etc/init.d/haproxy restart |

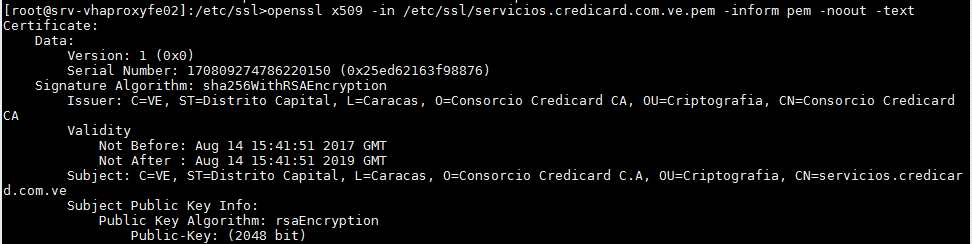
Evidencias que se ejecutó el cambio:

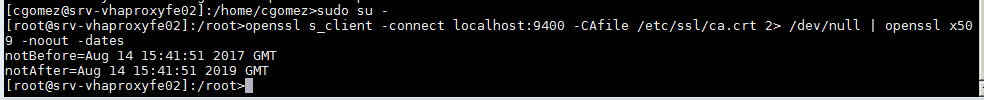












En los servidores “**srv-vhaproxybe01 y srv-vhaproxybe02**” copiar el certificado ”slbbe.credicard.com.ve.pem” en la ruta “/tmp”.

Realizar una copia del certificado actual.

|  |
| --- |
| cp -p /etc/ssl/ slbbe.credicard.com.ve.pem /etc/ssl/BackUp-Certs/ |

Antes de realizar la copia se debe verificar que el respaldo anterior fuera exitoso, Copiar el nuevo certificado.

El formato PEM, debe tener el key del certificado, el crt del certificado y el crt público de la CA, para generarlo es asi.

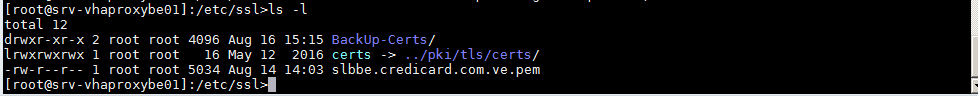
|  |
| --- |
| [root@srv-vhaproxyfe02]:/>cat slbbe.credicard.com.ve.key slbbe.credicard.com.ve.crt ca.crt > slbbe.credicard.com.ve.pem |

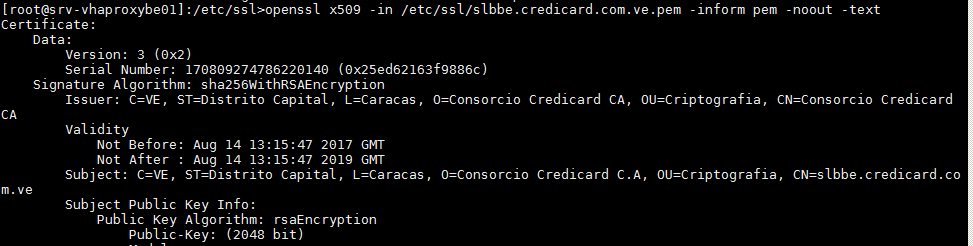
|  |
| --- |
| cp -p /tmp/ slbbe.credicard.com.ve.pem /etc/ssl/ |

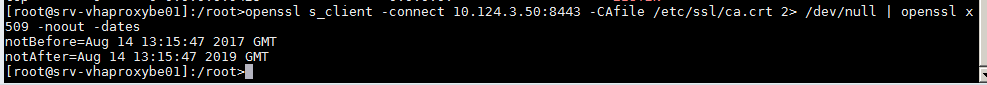
Reiniciar el servicio de HAProxy.

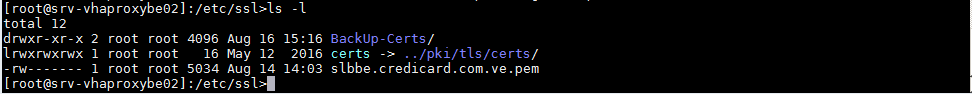
|  |
| --- |
| /etc/init.d/haproxy restart |

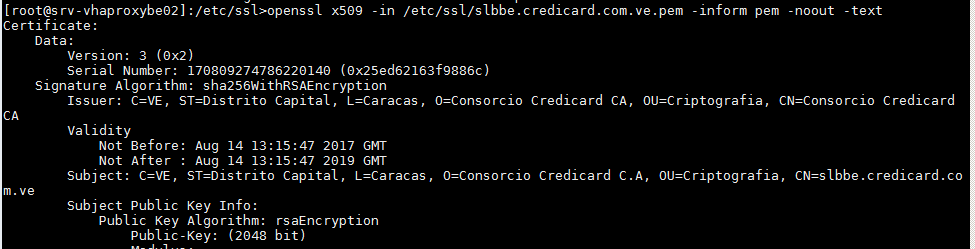
Evidencias que se ejecutó el cambio:

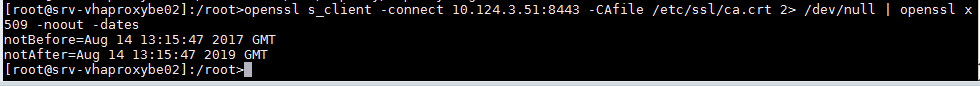












En los servidores “**srv-vjbossprd01 y srv-vjbossprd02**” copiar el certificado ”ws.credicard.com.ve.crt”, la llave primaria “ws.credicard.com.ve.key” y la CA de Credicard en la ruta “/tmp”, luego debemos crear un certificado en formato “P12” que tenga el certificado y la llave primara, esto es motivado que para integrar la llave primaria en el “KeyStore” debe estar en formato “P12”, la clave que utilizaremos es la misma que se colocara en el “KeyStore” y debe ser la misma clave que se viene utilizando en el archivo de configuración del JBOSS.

|  |
| --- |
| openssl pkcs12 -export -name ws.credicard.com.ve -in ws.credicard.com.ve.crt -inkey ws.credicard.com.ve.key -out ws.credicard.com.ve.p12  Enter Export Password:  Verifying - Enter Export Password: |

Creamos el “KeyStore” con el certificado creado en formato “P12” y utilizamos también la clave anteriormente explicada.

|  |
| --- |
| keytool -importkeystore -destkeystore jbosswso2.jks -srckeystore ws.credicard.com.ve.p12 -srcstoretype pkcs12 -alias ws.credicard.com.ve  Enter destination keystore password:  Re-enter new password:  Enter source keystore password: |

Agregamos dentro del “KeyStore” la CA de Credicard, utilizamos también la clave anteriormente explicada.

|  |
| --- |
| keytool -importcert-alias CA -file ca.crt -keystore jbosswso2.jks  Enter keystore password:  [ … ]  Trust this certificate? [no]: yes  Certificate was added to keystore |

Realizamos una consulta del “KeyStore” para certificar su estado.

|  |
| --- |
| keytool -v -list -keystore ./jbosswso2.jks |

Procedemos a realizar una copia del certificado actual y sustituirlo por el nuevo certificado creado en los pasos anteriores

Realizamos el respaldo.

|  |
| --- |
| /opt/jboss/jboss-eap-6.4/domain/configuration>cp -p jbosswso2.jks BackUp-Certs/ |

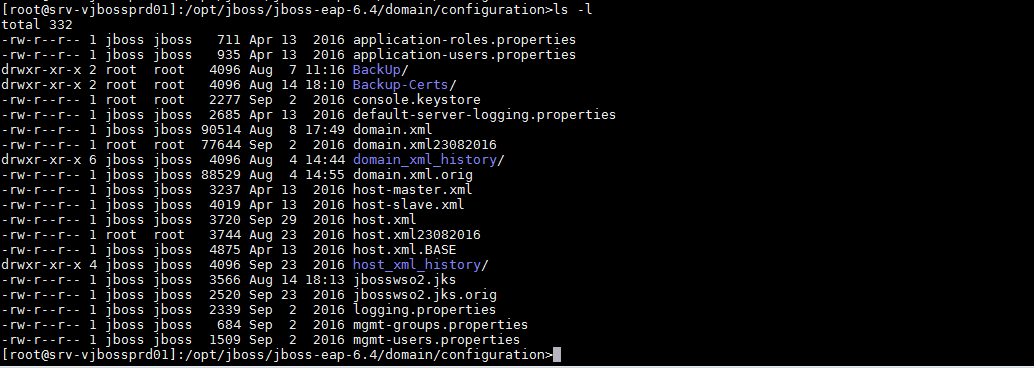
Antes de realizar la copia se debe verificar que el respaldo anterior fuera exitoso, realizamos la copia del nuevo certificado.

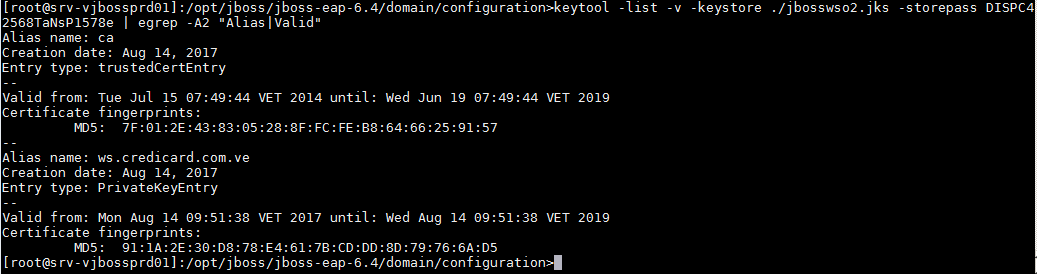
|  |
| --- |
| cp -p /tmp/jbosswso2.jks /opt/jboss/jboss-eap-6.4/domain/configuration |

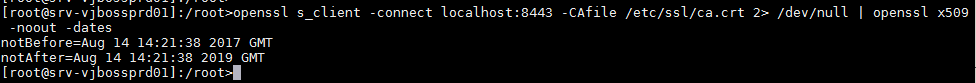
Reiniciamos el servicio de JBOSS.

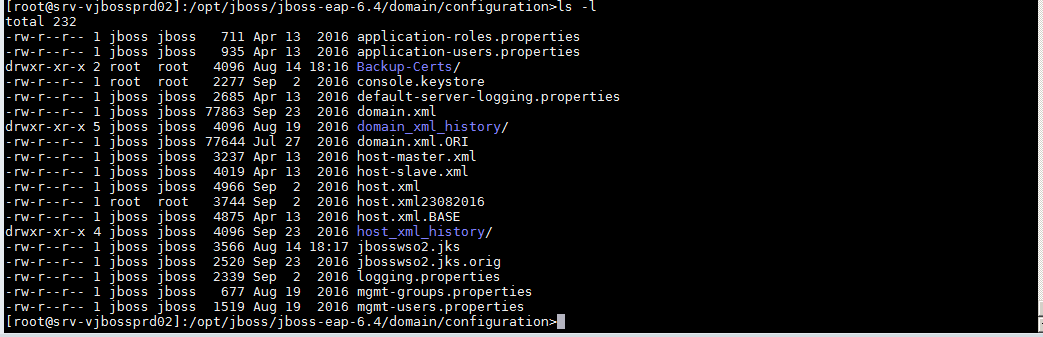
|  |
| --- |
| /etc/init.d/jboss-eap restart |

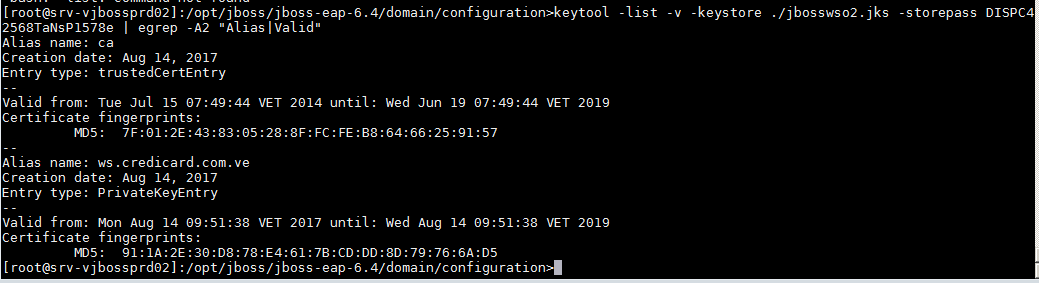
Evidencias que se ejecutó el cambio:

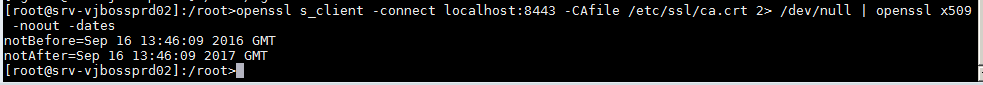












En los servidores “**srv-vmpos01 y srv-vmpos02**” copiar el certificado ”wsmpos.credicard.com.ve.crt”, la llave primaria “wsmpos.credicard.com.ve.key” y la CA de Credicard en la ruta “/tmp”, luego debemos crear un certificado en formato “P12” que tenga el certificado y la llave primara, esto es motivado que para integrar la llave primaria en el “KeyStore” debe estar en formato “P12”, la clave que utilizaremos es la misma que se colocara en el “KeyStore” y debe ser la misma clave que se viene utilizando en el archivo de configuración del JBOSS.

|  |
| --- |
| openssl pkcs12 -export -name wsmpos.credicard.com.ve -in wsmpos.credicard.com.ve.crt -inkey wsmpos.credicard.com.ve.key -out wsmpos.credicard.com.ve.p12  Enter Export Password:  Verifying - Enter Export Password: |

Creamos el “KeyStore” con el certificado creado en formato “P12” y utilizamos también la clave anteriormente explicada.

|  |
| --- |
| keytool -importkeystore -destkeystore wsmpos.jks -srckeystore wsmpos.credicard.com.ve.p12 -srcstoretype pkcs12 -alias wsmpos.credicard.com.ve  Enter destination keystore password:  Re-enter new password:  Enter source keystore password: |

Agregamos dentro del “KeyStore” la CA de Credicard, utilizamos también la clave anteriormente explicada.

|  |
| --- |
| keytool -importcert-alias CA -file ca.crt -keystore wsmpos.jks  Enter keystore password:  [ … ]  Trust this certificate? [no]: yes  Certificate was added to keystore |

Realizamos una consulta del “KeyStore” para certificar su estado.

|  |
| --- |
| keytool -v -list -keystore ./wsmpos.jks |

Procedemos a realizar una copia del certificado actual y sustituirlo por el nuevo certificado creado en los pasos anteriores

Realizamos el respaldo.

|  |
| --- |
| /opt/jboss/jboss-eap-6.2/domain/configuration>cp -p wsmpos.jks BackUp-Certs/ |

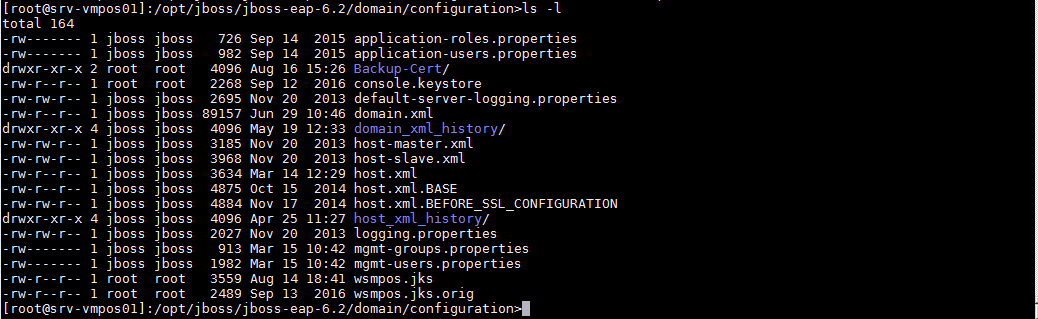
Antes de realizar la copia se debe verificar que el respaldo anterior fuera exitoso, realizamos la copia del nuevo certificado.

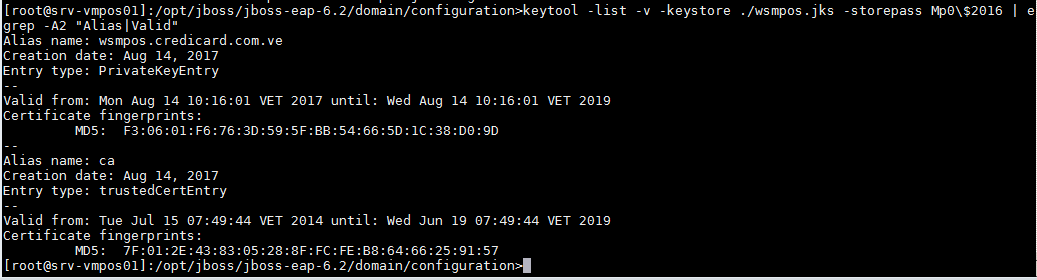
|  |
| --- |
| cp -p /tmp/wsmpos.jks /opt/jboss/jboss-eap-6.2/domain/configuration |

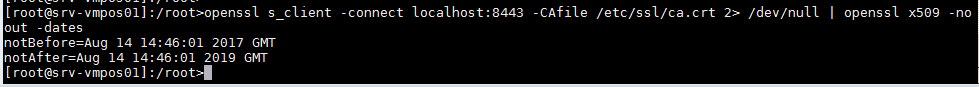
Reiniciamos el servicio de JBOSS.

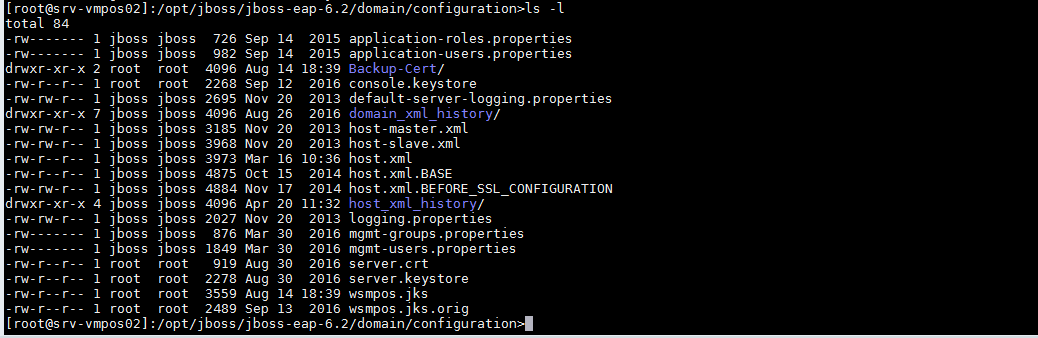
|  |
| --- |
| /etc/init.d/jboss-eap restart |

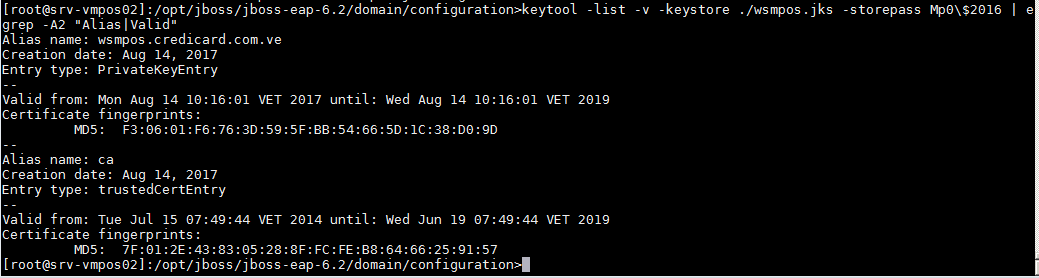
Evidencias que se ejecutó el cambio:

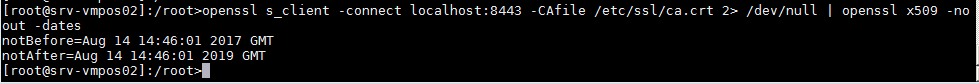












En los servidores “**srv-visprd01 y srv-visprd02**” copiar los certificados ”servicios.credicard.com.ve.crt, slbbe.credicard.com.ve.crt, serviciosbancos.credicard.com.ve.crt, ws.credicard.com.ve.crt, is.credicard.com.ve.crt, esb.credicard.com.ve.crt, das.credicard.com.ve.crt, wsmpos.credicard.com.ve.crt, feclientservices.credicard.com.ve.crt”, la llave primaria “is.credicard.com.ve.key” y la CA de Credicard en la ruta “/tmp”, luego debemos crear un certificado en formato “P12” que tenga el certificado y la llave primara “is.credicard.com.ve.key, is.credicard.com.ve.crt”, esto es motivado que para integrar la llave primaria en el “KeyStore” debe estar en formato “P12”, la clave que utilizaremos es la misma que se colocara en el “KeyStore” y debe ser la misma clave que se viene utilizando en el archivo de configuración del wso2.

|  |
| --- |
| openssl pkcs12 -export -name is.credicard.com.ve -in is.credicard.com.ve.crt -inkey is.credicard.com.ve.key -out is.credicard.com.ve.p12  Enter Export Password:  Verifying - Enter Export Password: |

Creamos el “KeyStore” con el certificado creado en formato “P12” y utilizamos también la clave anteriormente explicada. Muy importante escribir bien el alias.

|  |
| --- |
| keytool -importkeystore -destkeystore wso2carbon.jks -srckeystore is.credicard.com.ve.p12 -srcstoretype pkcs12 -alias is.credicard.com.ve  Enter destination keystore password:  Re-enter new password:  Enter source keystore password: |

Agregamos dentro del “KeyStore” la CA de Credicard, utilizamos también la clave anteriormente explicada.

|  |
| --- |
| keytool -importcert-alias CA -file ca.crt -keystore wso2carbon.jks  Enter keystore password:  [ … ]  Trust this certificate? [no]: yes  Certificate was added to keystore |

Realizamos el mismo paso utilizado para agregar la CA dentro del “Keystore” con cada uno de los certificados “servicios.credicard.com.ve.crt, slbbe.credicard.com.ve.crt, serviciosbancos.credicard.com.ve.crt, ws.credicard.com.ve.crt, esb.credicard.com.ve.crt, das.credicard.com.ve.crt, wsmpos.credicard.com.ve.crt, feclientservices.credicard.com.ve.crt”.

Realizamos una consulta del “KeyStore” para certificar su estado.

|  |
| --- |
| keytool -v -list -keystore ./wso2carbon.jks -storepass DIP9863LONDR3S3cu4d0 |

Al tener armado completamente el “KeyStore” con los certificados, la llave primaria y la CA de Credicard, procedemos a realizar una copia del certificado actual y sustituirlo por el nuevo certificado creado en los pasos anteriores

Realizamos el respaldo.

|  |
| --- |
| cp -p /opt/wso2/wso2is-5.1.0/repository/resources/security/wso2carbon.jks /opt/wso2/wso2is-5.1.0/repository/resources/security/BackUp-Certs/ |

Antes de realizar la copia se debe verificar que el respaldo anterior fuera exitoso, copiamos el nuevo certificado.

|  |
| --- |
| cp -p /tmp/ wso2carbon.jks /opt/wso2/wso2is-5.1.0/repository/resources/security |

Continuamos en los mismos servidores porque ahora debemos crear el “TrueStore”, debemos crear un certificado en formato “P12” que tenga el certificado y la llave primara “feclientservices.credicard.com.ve.key, feclientservices.credicard.com.ve.crt”, esto es motivado que para integrar la llave primaria en el “KeyStore” debe estar en formato “P12”, la clave que utilizaremos es la misma que se colocara en el “TrueStore” y debe ser la misma clave que se viene utilizando en el archivo de configuración del wso2.

|  |
| --- |
| >openssl pkcs12 -export -name feclientservices -in feclientservices.credicard.com.ve.crt -inkey feclientservices.credicard.com.ve.key -out feclientservices.credicard.com.ve.p12  Enter Export Password:  Verifying - Enter Export Password: |

Creamos el “TrueStore” con el certificado creado en formato “P12” y utilizamos también la clave anteriormente explicada.

|  |
| --- |
| keytool -importkeystore -destkeystore client-truststore.jks -srckeystore feclientservices.credicard.com.ve.p12 -srcstoretype pkcs12 -alias feclientservices  Enter destination keystore password:  Re-enter new password:  Enter source keystore password: |

Agregamos dentro del “TrueStore” la CA de Credicard, utilizamos también la clave anteriormente explicada.

|  |
| --- |
| keytool -importcert-alias CA -file ca.crt -keystore client-truststore.jks  Enter keystore password:  [ … ]  Trust this certificate? [no]: yes  Certificate was added to keystore |

Realizamos el mismo paso utilizado para agregar la CA dentro del “TrueStore” con cada uno de los certificados “slbbe.credicard.com.ve.crt, das.credicard.com.ve.crt, ws.credicard.com.ve.crt, is.credicard.com.ve.crt, esb.credicard.com.ve.crt, serviciosbancos.credicard.com.ve.crt, servicios.credicard.com.ve.crt, wsmpos.credicard.com.ve.crt”.

Realizamos una consulta del “TrueStore” para certificar su estado.

|  |
| --- |
| keytool -v -list -keystore ./ client-truststore.jks -storepass J4P4nFYT1125LOJAeFE2 |

Al tener armado completamente el “TrueStore” con los certificados, la llave primaria y la CA de Credicard, procedemos a realizar una copia del certificado actual y sustituirlo por el nuevo certificado creado en los pasos anteriores

Realizamos el respaldo.

|  |
| --- |
| cp -p /opt/wso2/wso2is-5.1.0/repository/resources/security/client-truststore.jks /opt/wso2/wso2is-5.1.0/repository/resources/security/BackUp-Certs/ |

Antes de realizar la copia se debe verificar que el respaldo anterior fuera exitoso, copiamos el nuevo certificado.

|  |
| --- |
| cp -p /tmp/client-truststore.jks /opt/wso2/wso2is-5.1.0/repository/resources/security |

Reiniciamos el servicio de wso2 IS server.

|  |
| --- |
| /etc/init.d/wso2is\_server restart |

Después de cambiar el KeyStore, el TrueStore y de reiniciar, Debemos actualizar (encriptar con el nuevo jks) en el user stores de la console admin para IS.

MAIN > User Stores > List > Edit

Usuario configurado en el USER STORE usrwso2 u$rP0rt4l (LDAP)

Pero antes debemos estar seguros que sea la clave correcta, para validar eso nos conectamos a uno de los servidores LDAP “srv-vccs-ldap-01 - 10.124.0.220” y ejecutamos el siguiente comando y debe retornar todos los usuarios del árbol del LDAP

|  |
| --- |
| ldapsearch -x -LLL -H ldap://10.124.0.220 -D "uid=usrwso2,ou=portal unificado,ou=aplicaciones,ou=activos,o=credicard,dc=credicard,dc=com.ve" -w u\$rP0rt4l -b dc=credicard,dc=com.ve |

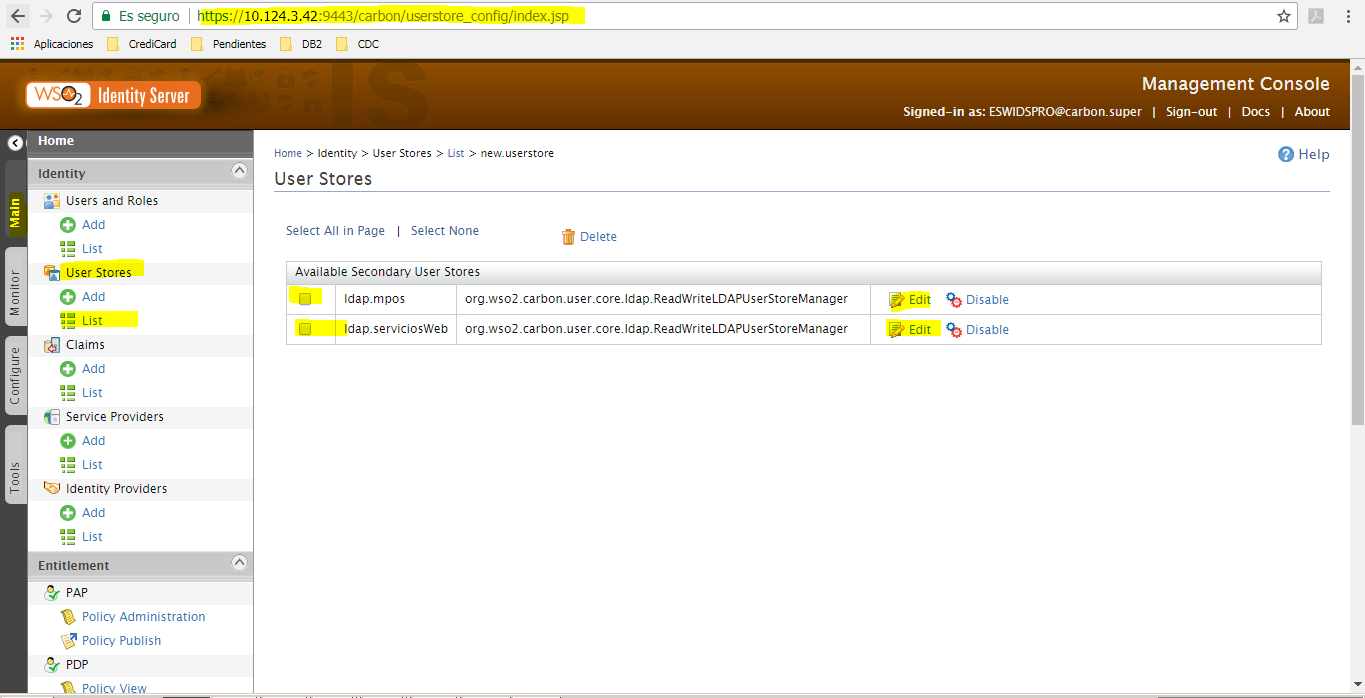
También es bueno que se realice desde los servidores IS una consulta al balanceador ACE “10.124.2.81” por el puerto “636” para estar seguros que está retornando el certificado de los LDAP.

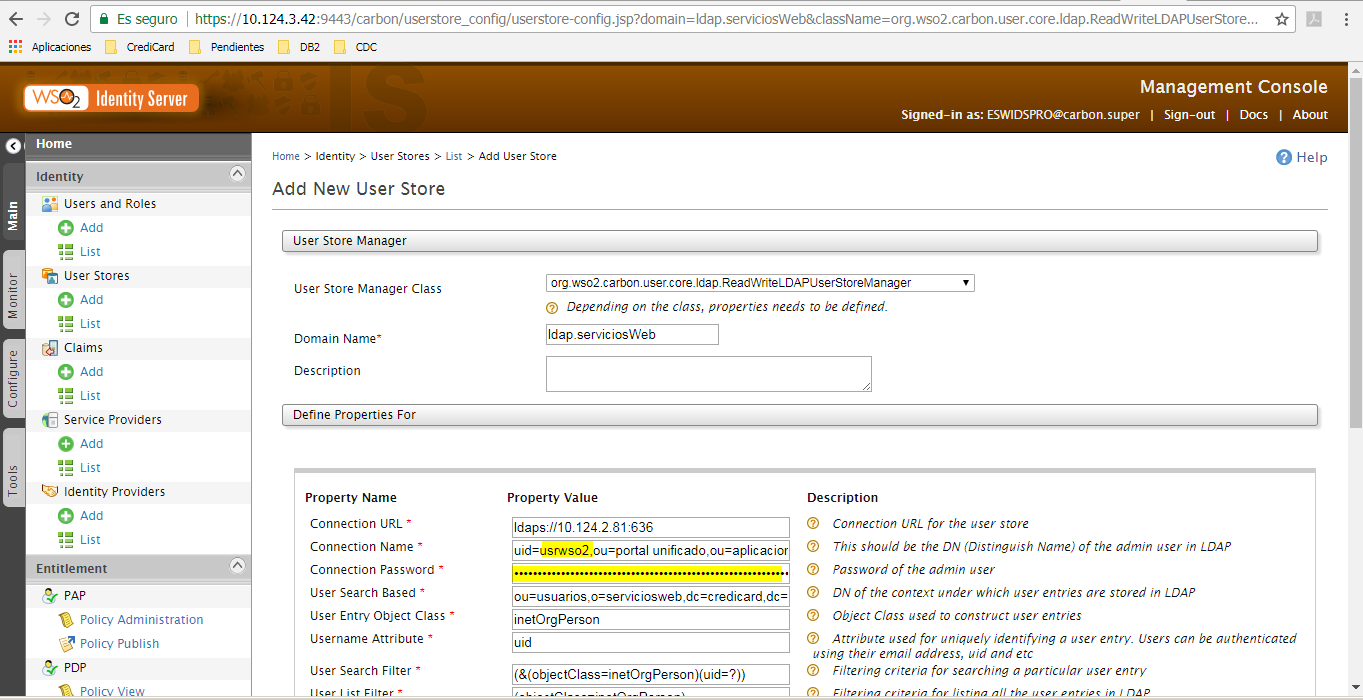
|  |
| --- |
| [root@srv-visprd01]: >openssl s\_client -connect 10.124.2.81:636 -CAfile /home/wso2/ca.crt  CONNECTED(00000003)  depth=1 C = VE, ST = Distrito Capital, L = Caracas, O = Consorcio Credicard CA, OU = Criptografia, CN = Consorcio Credicard CA  verify return:1depth=0 C = VE, ST = Distrito Capital, L = Caracas, O = Consorcio Credicard C.A, OU = Criptografia, CN = ldapclusprd.credicard.com.ve, emailAddress = criptografia@credicard.com.ve  verify return:1  ---  Certificate chain  0 s:/C=VE/ST=Distrito Capital/L=Caracas/O=Consorcio Credicard C.A/OU=Criptografia/CN=ldapclusprd.credicard.com.ve/emailAddress=criptografia@credicard.com.ve  i:/C=VE/ST=Distrito Capital/L=Caracas/O=Consorcio Credicard CA/OU=Criptografia/CN=Consorcio Credicard CA  ---  Server certificate  -----BEGIN CERTIFICATE-----  MIID2DCCAsACCAI8GRdbXGhNMA0GCSqGSIb3DQEBCwUAMIGTMQswCQYDVQQGEwJW  RTEZMBcGA1UECAwQRGlzdHJpdG8gQ2FwaXRhbDEQMA4GA1UEBwwHQ2FyYWNhczEf  MB0GA1UECgwWQ29uc29yY2lvIENyZWRpY2FyZCBDQTEVMBMGA1UECwwMQ3JpcHRv  Z3JhZmlhMR8wHQYDVQQDDBZDb25zb3JjaW8gQ3JlZGljYXJkIENBMB4XDTE2MTAz  MTE0NTQwMloXDTE4MTAzMTE0NTQwMlowgcgxCzAJBgNVBAYTAlZFMRkwFwYDVQQI  DBBEaXN0cml0byBDYXBpdGFsMRAwDgYDVQQHDAdDYXJhY2FzMSAwHgYDVQQKDBdD  b25zb3JjaW8gQ3JlZGljYXJkIEMuQTEVMBMGA1UECwwMQ3JpcHRvZ3JhZmlhMSUw  IwYDVQQDDBxsZGFwY2x1c3ByZC5jcmVkaWNhcmQuY29tLnZlMSwwKgYJKoZIhvcN  AQkBFh1jcmlwdG9ncmFmaWFAY3JlZGljYXJkLmNvbS52ZTCCASIwDQYJKoZIhvcN  AQEBBQADggEPADCCAQoCggEBAMM13yniIDAxFPKw1JMPQmvZMNmSEkR2FAxA+tss  RKvFWlYiD0sQghfPiTZxiA/jnI0CVoRFXffrO868k+FBHzxd0KonN9BIpjYchlWH  q8C90D98Z3fYzydlTt1u/rB7mCCqW606d35z+Lhx/w0aCrIf3enaZeO/qQQ5jleZ  5LmYk+qVGoAWGNJEUZRPSvGFxNJR1OOQ9XJhxpjdZsOAksmnXLixsoGhOMRjVscb  w1dztrv2Jksa4MDezl2azfNJvS/2ljXCickoLenXtOY9H8XHQqu4JpIEKY8SLyat  JfgGHskJBva7H9qywU1Op1f925hKBFlsl0RuAB4zptq0QKsCAwEAATANBgkqhkiG  9w0BAQsFAAOCAQEAGXogCeH4O2VH5KQOYGPZozdUmqPwYoJ2OjHmupD42ThS4Egx  zGM5nR50krVEVy1C650O+/SoKc10BR8PhYxOpxE3A58qT11hGdcYRUyYtg3qZmGt  M5wkMVKOpiPPOROqD1+GYEmq4dotLWBULgtRbIuWFnj7DO+g22eUiXlmyyDeCyso  j0emIPM1mB02TUKteDCAjahTTsUnxj9rLs04fealjn7XRhTKuka3qvZj3tdxpWbE  N9V8wHSdbML70hhKNygDowm35ymQkSt+QxmZXnz0QmrjGqFbLqZYgsv9w5nyuvr2  cbH61s3VHkhVpusEqqcYkf6QeXH63QP/r2WyRg==  -----END CERTIFICATE-----  subject=/C=VE/ST=Distrito Capital/L=Caracas/O=Consorcio Credicard C.A/OU=Criptografia/CN=ldapclusprd.credicard.com.ve/emailAddress=criptografia@credicard.com.ve  issuer=/C=VE/ST=Distrito Capital/L=Caracas/O=Consorcio Credicard CA/OU=Criptografia/CN=Consorcio Credicard CA  ---  No client certificate CA names sent  Server Temp Key: ECDH, prime256v1, 256 bits  ---  SSL handshake has read 1502 bytes and written 397 bytes  ---  New, TLSv1/SSLv3, Cipher is ECDHE-RSA-AES256-SHA  Server public key is 2048 bit  Secure Renegotiation IS supported  Compression: NONE  Expansion: NONE  SSL-Session:  Protocol : TLSv1.2  Cipher : ECDHE-RSA-AES256-SHA  Session-ID: 093FA5D584F58CEE9C2B439C3103F5E6400453CBEB118FECBA7C0E009BE3D1FD  Session-ID-ctx:  Master-Key: 5CC128841F0A7ECC768874521E51D56BC722EE738734B6B9D8FA8254319C809BCF9D6DB28C943F70532691EC5675B1B0  Key-Arg : None  Krb5 Principal: None  PSK identity: None  PSK identity hint: None  Start Time: 1502908312  Timeout : 300 (sec)  Verify return code: 0 (ok)  --- |

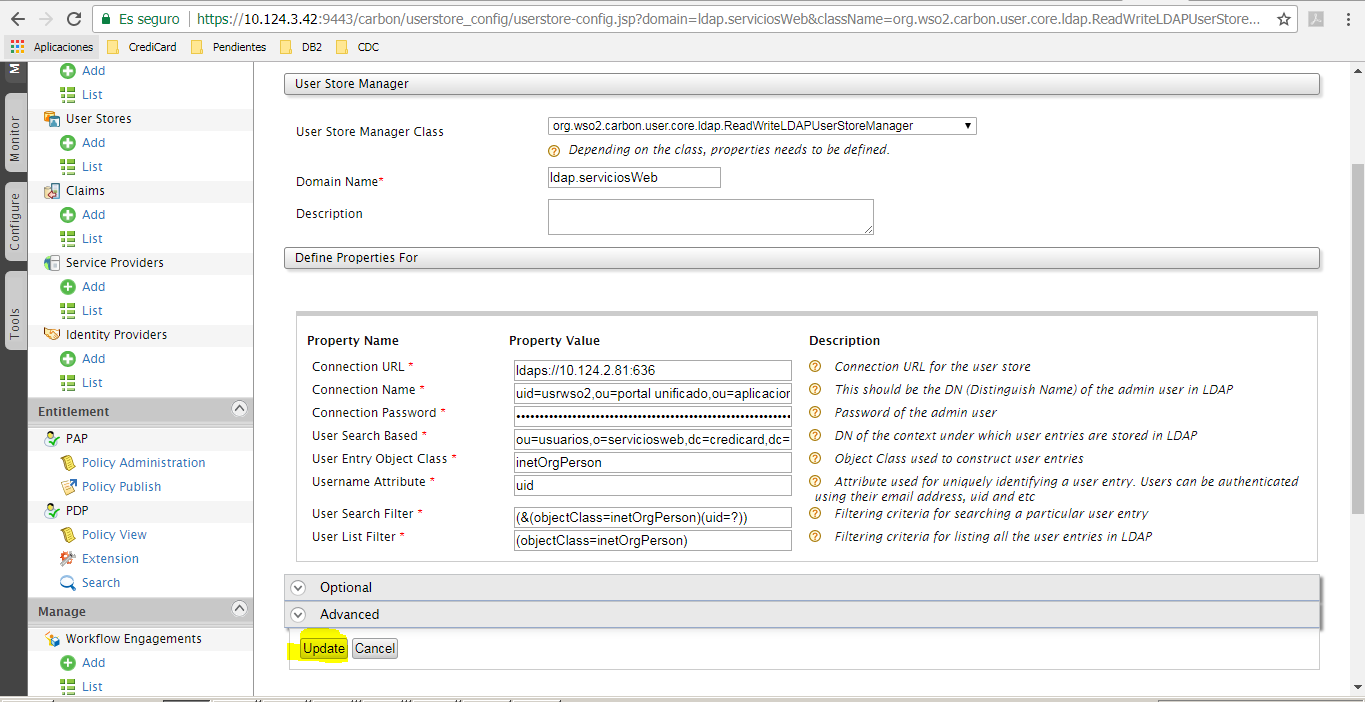
El Update se debe hacer en ambos servidores

https://10.124.3.42:9443/carbon/admin/login.jsp

https://10.124.3.43:9443/carbon/admin/login.jsp

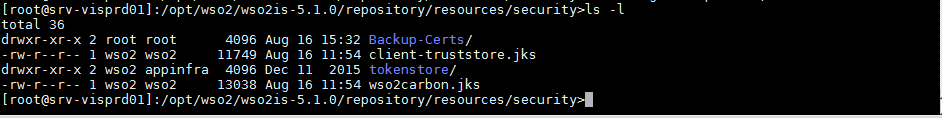


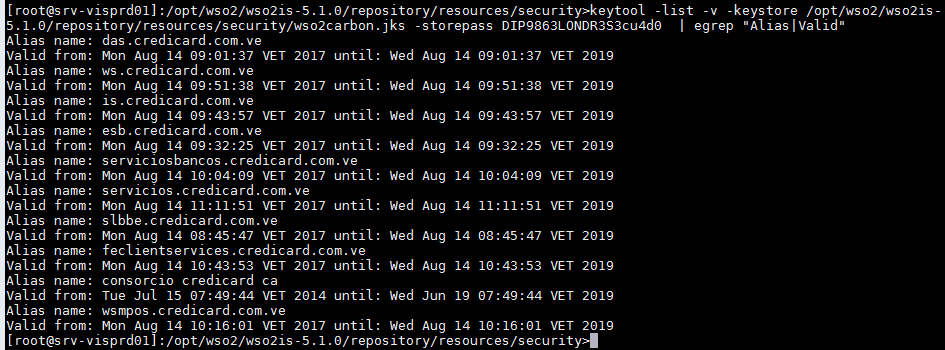


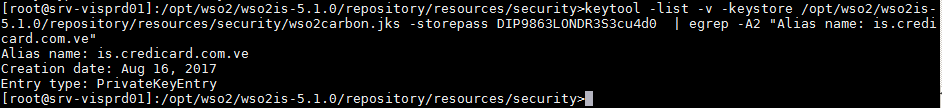


Evidencias que se ejecutó el cambio:

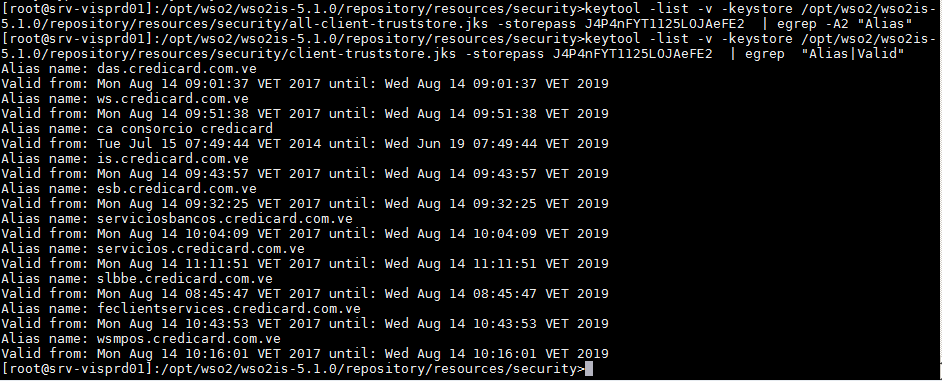
srv-visprd01 – wso2carbon.jks

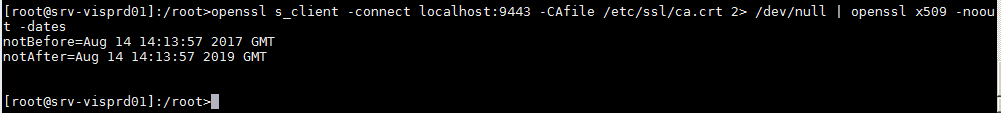




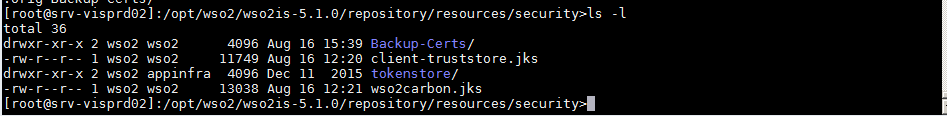


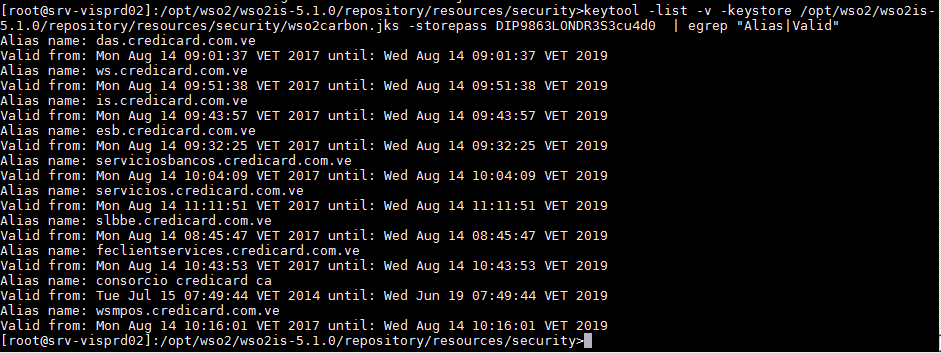
srv-visprd01 – client-truststore.jks

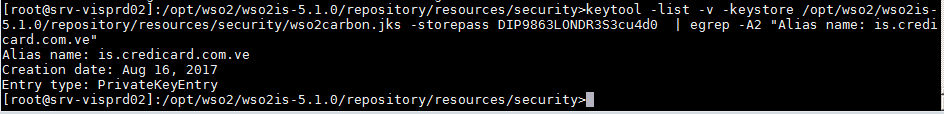




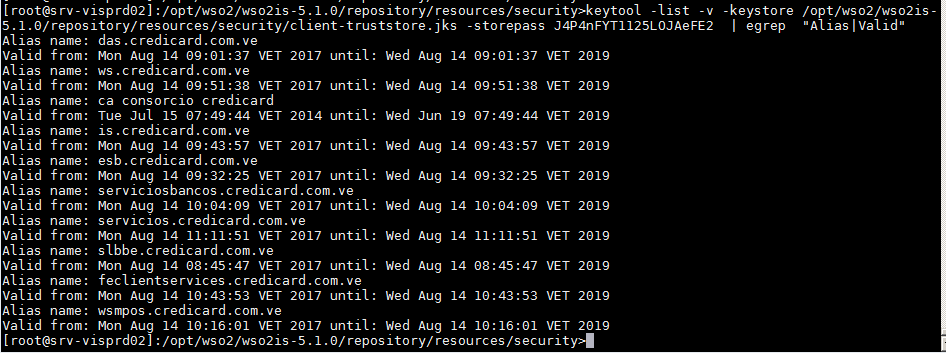
srv-visprd02 –wso2carbon.jks

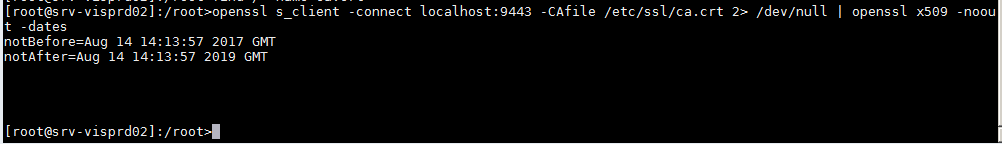






srv-visprd02 – client-truststore.jks





En los servidores “**srv-vesbadmprd01, srv-vesbprd01, srv-vesbprd02 y srv-vesbprd03**” copiar los certificados ”servicios.credicard.com.ve.crt, slbbe.credicard.com.ve.crt, serviciosbancos.credicard.com.ve.crt, ws.credicard.com.ve.crt, is.credicard.com.ve.crt, esb.credicard.com.ve.crt, das.credicard.com.ve.crt, wsmpos.credicard.com.ve.crt, feclientservices.credicard.com.ve.crt”, la llave primaria “esb.credicard.com.ve.key” y la CA de Credicard en la ruta “/tmp”, luego debemos crear un certificado en formato “P12” que tenga el certificado y la llave primara “esb.credicard.com.ve.key, esb.credicard.com.ve.crt”, esto es motivado que para integrar la llave primaria en el “KeyStore” debe estar en formato “P12”, la clave que utilizaremos es la misma que se colocara en el “KeyStore” y debe ser la misma clave que se viene utilizando en el archivo de configuración del wso2.

|  |
| --- |
| openssl pkcs12 -export -name esb.credicard.com.ve -in esb.credicard.com.ve.crt -inkey esb.credicard.com.ve.key -out esb.credicard.com.ve.p12  Enter Export Password:  Verifying - Enter Export Password: |

Creamos el “KeyStore” con el certificado creado en formato “P12” y utilizamos también la clave anteriormente explicada. Muy importante escribir bien el alias

|  |
| --- |
| keytool -importkeystore -destkeystore wso2carbon.jks -srckeystore esb.credicard.com.ve.p12 -srcstoretype pkcs12 -alias esb.credicard.com.ve  Enter destination keystore password:  Re-enter new password:  Enter source keystore password: |

Agregamos dentro del “KeyStore” la CA de Credicard, utilizamos también la clave anteriormente explicada.

|  |
| --- |
| keytool -importcert-alias CA -file ca.crt -keystore wso2carbon.jks  Enter keystore password:  [ … ]  Trust this certificate? [no]: yes  Certificate was added to keystore |

Realizamos el mismo paso utilizado para agregar la CA dentro del “Keystore” con cada uno de los certificados “servicios.credicard.com.ve.crt, slbbe.credicard.com.ve.crt, serviciosbancos.credicard.com.ve.crt, ws.credicard.com.ve.crt, is.credicard.com.ve.crt, das.credicard.com.ve.crt, wsmpos.credicard.com.ve.crt, feclientservices.credicard.com.ve.crt”.

Realizamos una consulta del “KeyStore” para certificar su estado.

|  |
| --- |
| keytool -v -list -keystore ./wso2carbon.jks -storepass DIP9863LONDR3S3cu4d0 |

Al tener armado completamente el “KeyStore” con los certificados, la llave primaria y la CA de Credicard, procedemos a realizar una copia del certificado actual y sustituirlo por el nuevo certificado creado en los pasos anteriores

Realizamos el respaldo.

|  |
| --- |
| cp -p /opt/wso2/wso2esb-4.9.0/repository/resources/security/wso2carbon.jks /opt/wso2/wso2esb-4.9.0/repository/resources/security/BackUp-Certs/ |

Antes de realizar la copia se debe verificar que el respaldo anterior fuera exitoso, copiamos el nuevo certificado.

|  |
| --- |
| cp -p /tmp/wso2carbon.jks /opt/wso2/wso2esb-4.9.0/repository/resources/security |

Continuamos en los mismos servidores porque ahora debemos crear el “TrueStore”, debemos crear un certificado en formato “P12” que tenga el certificado y la llave primara “feclientservices.credicard.com.ve.key, feclientservices.credicard.com.ve.crt”, esto es motivado que para integrar la llave primaria en el “KeyStore” debe estar en formato “P12”, la clave que utilizaremos es la misma que se colocara en el “TrueStore” y debe ser la misma clave que se viene utilizando en el archivo de configuración del wso2.

|  |
| --- |
| >openssl pkcs12 -export -name feclientservices -in feclientservices.credicard.com.ve.crt -inkey feclientservices.credicard.com.ve.key -out feclientservices.credicard.com.ve.p12  Enter Export Password:  Verifying - Enter Export Password: |

Creamos el “TrueStore” con el certificado creado en formato “P12” y utilizamos también la clave anteriormente explicada.

|  |
| --- |
| keytool -importkeystore -destkeystore client-truststore.jks -srckeystore feclientservices.credicard.com.ve.p12 -srcstoretype pkcs12 -alias feclientservices  Enter destination keystore password:  Re-enter new password:  Enter source keystore password: |

Agregamos dentro del “TrueStore” la CA de Credicard, utilizamos también la clave anteriormente explicada.

|  |
| --- |
| keytool -importcert-alias CA -file ca.crt -keystore client-truststore.jks  Enter keystore password:  [ … ]  Trust this certificate? [no]: yes  Certificate was added to keystore |

Realizamos el mismo paso utilizado para agregar la CA dentro del “TrueStore” con cada uno de los certificados “slbbe.credicard.com.ve.crt, das.credicard.com.ve.crt, ws.credicard.com.ve.crt, is.credicard.com.ve.crt, is.credicard.com.ve.crt, serviciosbancos.credicard.com.ve.crt, servicios.credicard.com.ve.crt, wsmpos.credicard.com.ve.crt”.

Realizamos una consulta del “TrueStore” para certificar su estado.

|  |
| --- |
| keytool -v -list -keystore ./client-truststore.jks -storepass J4P4nFYT1125LOJAeFE2 |

Al tener armado completamente el “TrueStore” con los certificados, la llave primaria y la CA de Credicard, procedemos a realizar una copia del certificado actual y sustituirlo por el nuevo certificado creado en los pasos anteriores

Realizamos el respaldo.

|  |
| --- |
| cp -p /opt/wso2/wso2esb-4.9.0/repository/resources/security/client-truststore.jks /opt/wso2/wso2esb-4.9.0/repository/resources/security/BackUp-Certs/ |

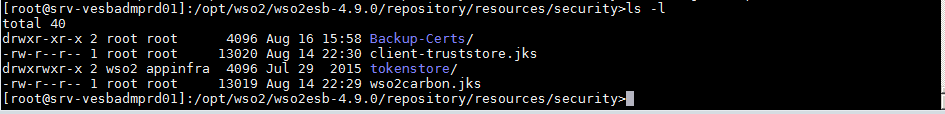
Antes de realizar la copia se debe verificar que el respaldo anterior fuera exitoso, copiamos el nuevo certificado.

|  |
| --- |
| cp -p /tmp/client-truststore.jks /opt/wso2/wso2esb-4.9.0/repository/resources/security |

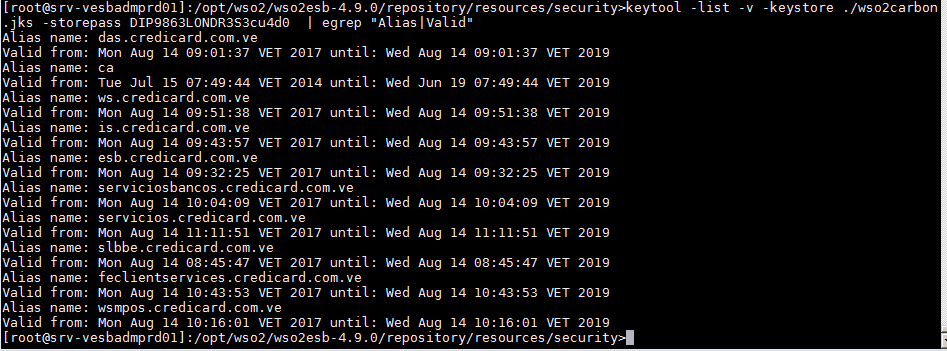
Reiniciamos el servicio de wso2 ESB en los servidores.

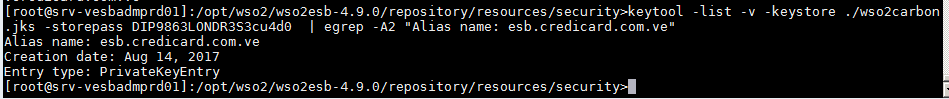
|  |
| --- |
| /etc/init.d/wso2esb\_server restart |

Evidencias que se ejecutó el cambio:

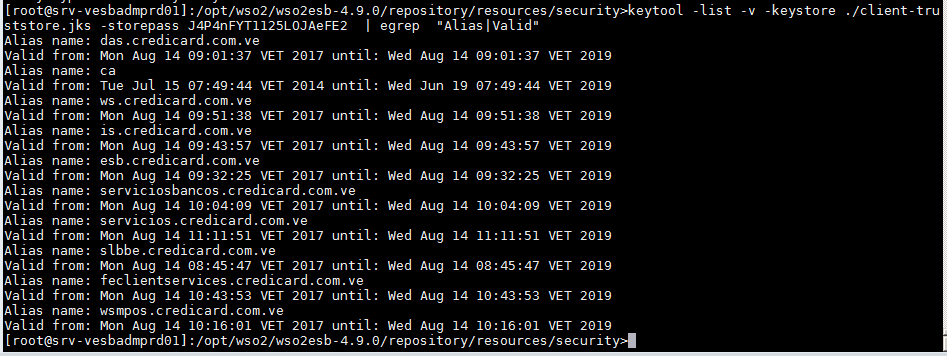


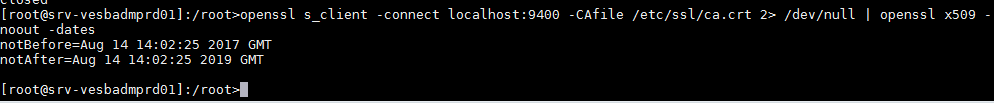
srv-vesbadmprd01 – wso2carbon.jks

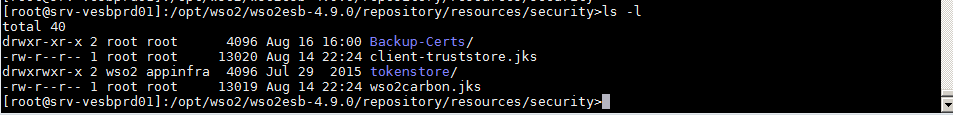




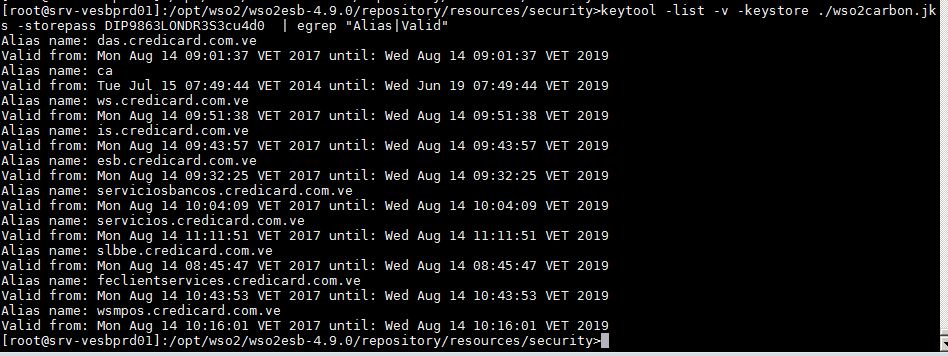
srv-vesbadmprd01 – client-truststore.jks

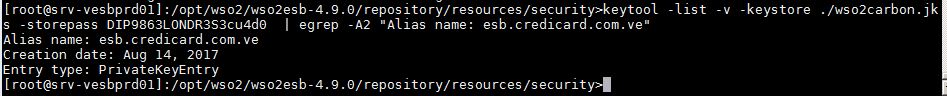




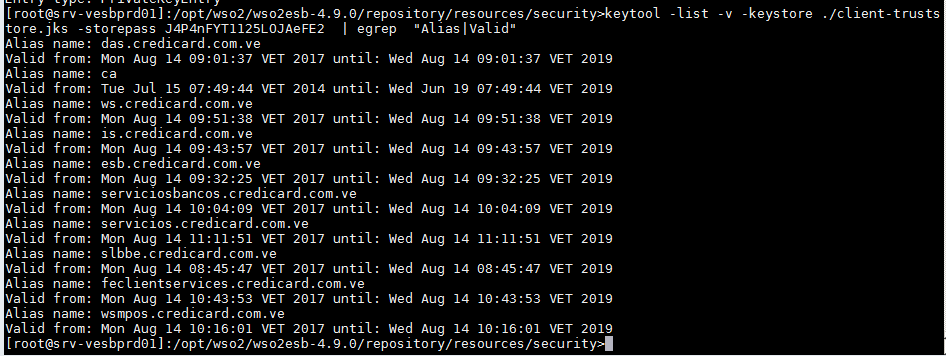


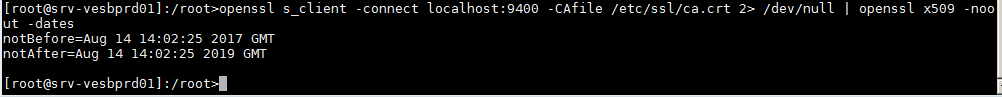
srv-vesbprd01 – wso2carbon.jks

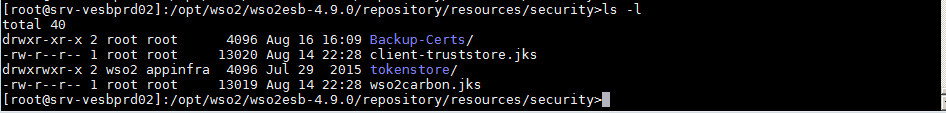




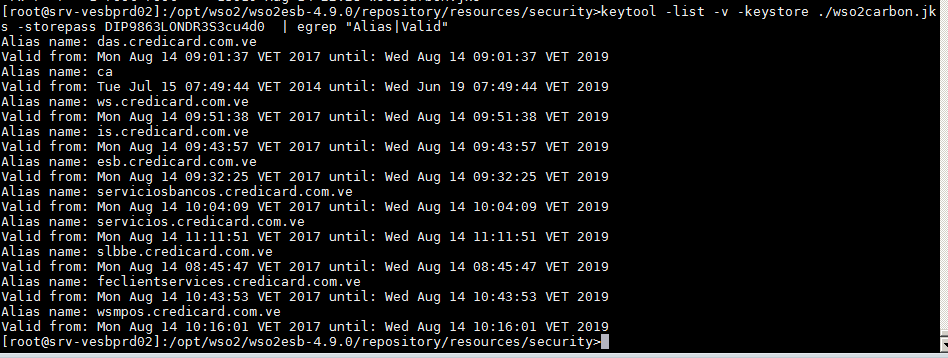
srv-vesbprd01 – client-truststore.jks

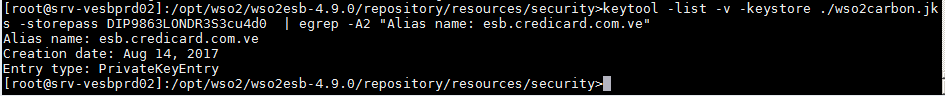




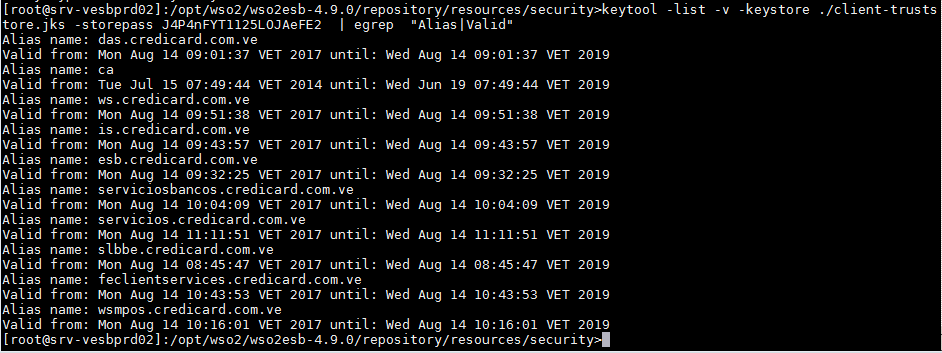


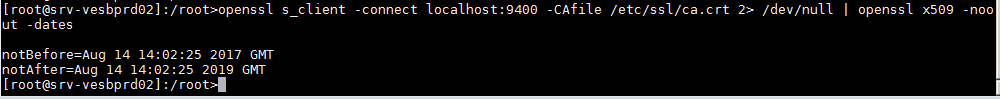
srv-vesbprd02 – wso2carbon.jks

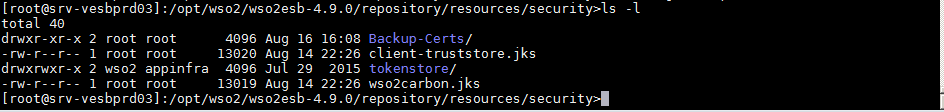




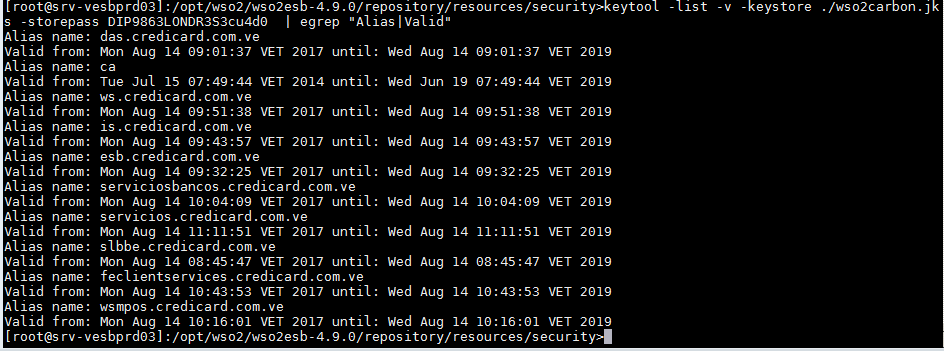
srv-vesbprd02 – client-truststore.jks

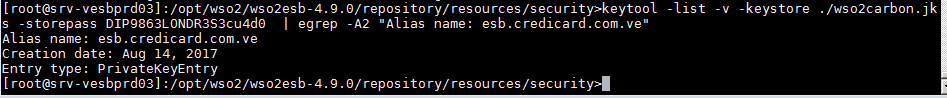




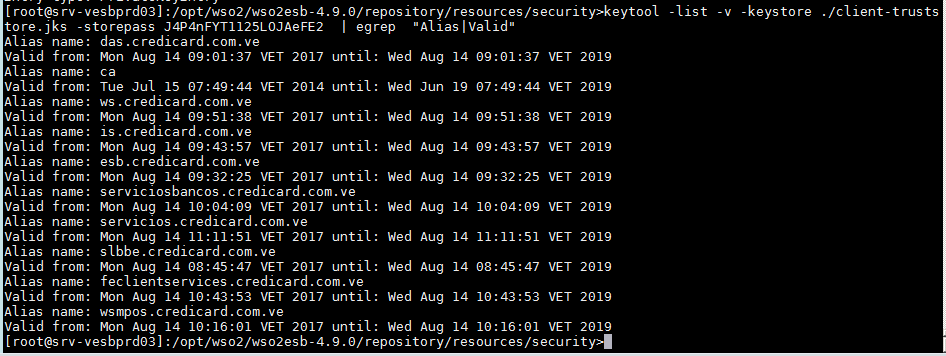


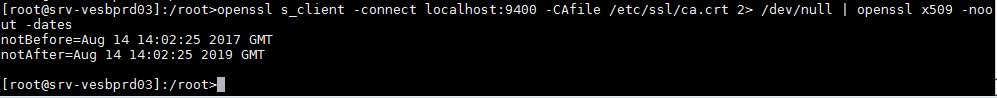
srv-vesbprd03 – wso2carbon.jks





srv-vesbprd03 – client-truststore.jks





En el servidor “**srv-vdasprd01**” copiar los certificados ”servicios.credicard.com.ve.crt, slbbe.credicard.com.ve.crt, serviciosbancos.credicard.com.ve.crt, ws.credicard.com.ve.crt, is.credicard.com.ve.crt, esb.credicard.com.ve.crt, wsmpos.credicard.com.ve.crt, feclientservices.credicard.com.ve.crt, das.credicard.com.ve.crt”, la llave primaria “das.credicard.com.ve.key” y la CA de Credicard en la ruta “/tmp”, luego debemos crear un certificado en formato “P12” que tenga el certificado y la llave primara “das.credicard.com.ve.key, das.credicard.com.ve.crt”, esto es motivado que para integrar la llave primaria en el “KeyStore” debe estar en formato “P12”, la clave que utilizaremos es la misma que se colocara en el “KeyStore” y debe ser la misma clave que se viene utilizando en el archivo de configuración del wso2.

|  |
| --- |
| openssl pkcs12 -export -name das.credicard.com.ve -in das.credicard.com.ve.crt -inkey das.credicard.com.ve.key -out das.credicard.com.ve.p12  Enter Export Password:  Verifying - Enter Export Password: |

Creamos el “KeyStore” con el certificado creado en formato “P12” y utilizamos también la clave anteriormente explicada. Muy importante escribir bien el alias

|  |
| --- |
| keytool -importkeystore -destkeystore wso2carbon.jks -srckeystore das.credicard.com.ve.p12 -srcstoretype pkcs12 -alias das.credicard.com.ve  Enter destination keystore password:  Re-enter new password:  Enter source keystore password: |

Agregamos dentro del “KeyStore” la CA de Credicard, utilizamos también la clave anteriormente explicada.

|  |
| --- |
| keytool -importcert-alias CA -file ca.crt -keystore wso2carbon.jks  Enter keystore password:  [ … ]  Trust this certificate? [no]: yes  Certificate was added to keystore |

Realizamos el mismo paso utilizado para agregar la CA dentro del “Keystore” con cada uno de los certificados “servicios.credicard.com.ve.crt, slbbe.credicard.com.ve.crt, serviciosbancos.credicard.com.ve.crt, ws.credicard.com.ve.crt, is.credicard.com.ve.crt, esb.credicard.com.ve.crt, wsmpos.credicard.com.ve.crt, feclientservices.credicard.com.ve.crt”.

Realizamos una consulta del “KeyStore” para certificar su estado.

|  |
| --- |
| keytool -v -list -keystore ./wso2carbon.jks -storepass DIP9863LONDR3S3cu4d0 |

Al tener armado completamente el “KeyStore” con los certificados, la llave primaria y la CA de Credicard, procedemos a realizar una copia del certificado actual y sustituirlo por el nuevo certificado creado en los pasos anteriores

|  |
| --- |
| cp -p /opt/wso2/wso2das-3.0.1/repository/resources/security/wso2carbon.jks /opt/wso2/wso2das-3.0.1/repository/resources/security/BackUp-Certs/ |

Antes de realizar la copia se debe verificar que el respaldo anterior fuera exitoso, copiamos el nuevo certificado.

|  |
| --- |
| cp -p /tmp/wso2carbon.jks /opt/wso2/wso2das-3.0.1/repository/resources/security |

Continuamos en los mismos servidores porque ahora debemos crear el “TrueStore”, debemos crear un certificado en formato “P12” que tenga el certificado y la llave primara “feclientservices.credicard.com.ve.key, feclientservices.credicard.com.ve.crt”, esto es motivado que para integrar la llave primaria en el “KeyStore” debe estar en formato “P12”, la clave que utilizaremos es la misma que se colocara en el “TrueStore” y debe ser la misma clave que se viene utilizando en el archivo de configuración del wso2.

|  |
| --- |
| openssl pkcs12 -export -name feclientservices -in feclientservices.credicard.com.ve.crt -inkey feclientservices.credicard.com.ve.key -out feclientservices.credicard.com.ve.p12  Enter Export Password:  Verifying - Enter Export Password: |

Creamos el “TrueStore” con el certificado creado en formato “P12” y utilizamos también la clave anteriormente explicada.

|  |
| --- |
| keytool -importkeystore -destkeystore client-truststore.jks -srckeystore feclientservices.credicard.com.ve.p12 -srcstoretype pkcs12 -alias feclientservices  Enter destination keystore password:  Re-enter new password:  Enter source keystore password: |

Agregamos dentro del “TrueStore” la CA de Credicard, utilizamos también la clave anteriormente explicada.

|  |
| --- |
| keytool -importcert-alias CA -file ca.crt -keystore client-truststore.jks  Enter keystore password:  [ … ]  Trust this certificate? [no]: yes  Certificate was added to keystore |

Realizamos el mismo paso utilizado para agregar la CA dentro del “TrueStore” con cada uno de los certificados “slbbe.credicard.com.ve.crt, das.credicard.com.ve.crt, ws.credicard.com.ve.crt, is.credicard.com.ve.crt, is.credicard.com.ve.crt, serviciosbancos.credicard.com.ve.crt, servicios.credicard.com.ve.crt, wsmpos.credicard.com.ve.crt”.

Realizamos una consulta del “TrueStore” para certificar su estado.

|  |
| --- |
| keytool -v -list -keystore ./client-truststore.jks -storepass J4P4nFYT1125LOJAeFE2 |

Al tener armado completamente el “TrueStore” con los certificados, la llave primaria y la CA de Credicard, procedemos a realizar una copia del certificado actual y sustituirlo por el nuevo certificado creado en los pasos anteriores

|  |
| --- |
| cp -p /opt/wso2/wso2esb-4.9.0/repository/resources/security/client-truststore.jks /opt/wso2/wso2esb-4.9.0/repository/resources/security/BackUp-Certs/ |

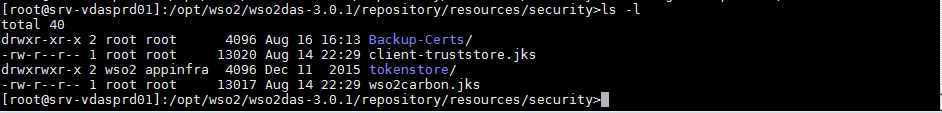
Antes de realizar la copia se debe verificar que el respaldo anterior fuera exitoso, copiamos el nuevo certificado.

|  |
| --- |
| cp -p /tmp/client-truststore.jks /opt/wso2/wso2esb-4.9.0/repository/resources/security |

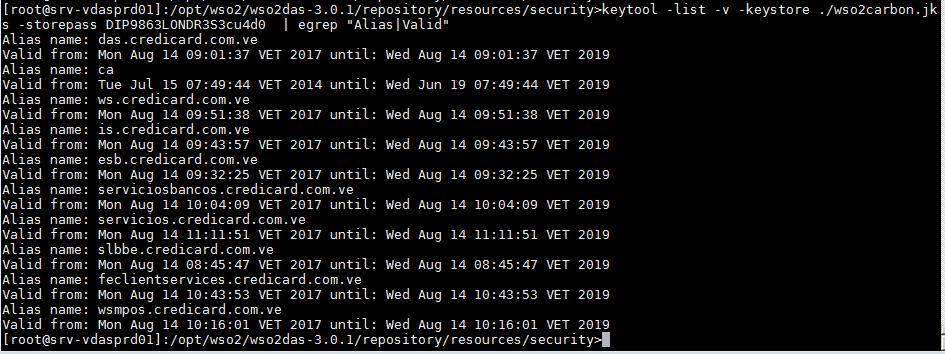
Reiniciamos el servicio de wso2 ESB en los servidores.

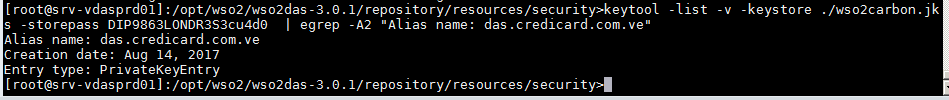
|  |
| --- |
| /etc/init.d/wso2das\_server restart |

Evidencias que se ejecutó la actividad:

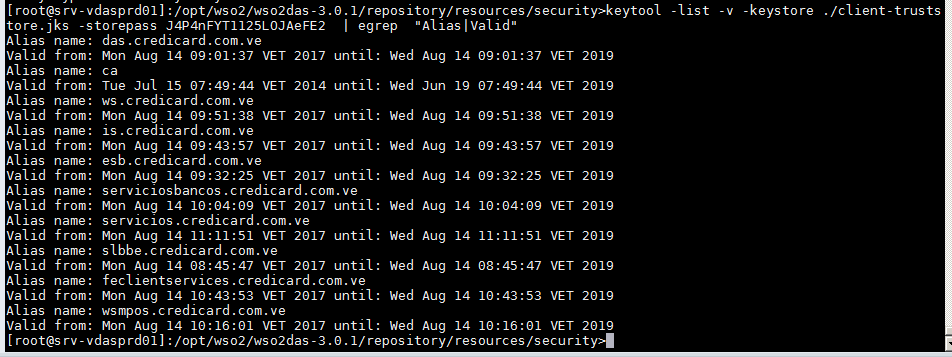


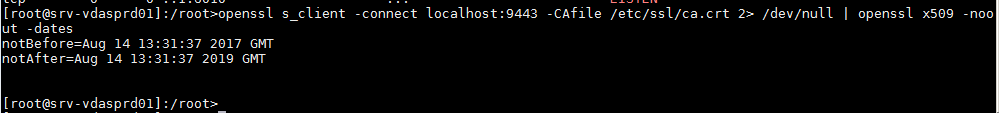
srv-vdasprd01 – wso2carbon.jks





srv-vdasprd01 – client-truststore.jks





IMPORTANTE recordar nuevamente > en los IS despues de cambiar el KeyStore y el TrueStore y reiniciar el servicio de wso2is\_server , debemos actualizar (encriptar con el nuevo jks) en el user store de la console admin para IS

MAIN > User Stores > List > Edit

Usuario configurado en el USER STORE usrwso2 u$rP0rt4l (LDAP)

Se debe hacer en ambos servidores

https://10.124.3.42:9443/carbon/admin/login.jsp

https://10.124.3.43:9443/carbon/admin/login.jsp

