IBM Software Group

SSL Certificate and Key Management

Brett Ostrander (bretto@us.ibm.com)
Software Engineer
June 12, 2012

WebSphere® Support Technical Exchange

Agenda

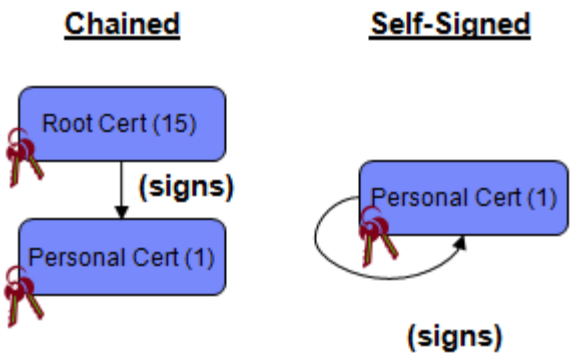
- Chained Certificates
- Renewing Certificates
- Personal Certificate Requests
- KeyStores
- Certificate Expiration Monitor

Chained Certificates

- Default chained certificates are created by root certificates with a long life span of 15 years.
- Using a root signer, the servers and client can establish and keep trust by exchanging the long lived root certificate signer as opposed to the short lived (1 year) signer of the personal certificates.
- Additionally the chained certificate provides flexibility surrounding the scalability issues in the flexible management configurations.

Chained Certificates (cont.)

- A chained certificate is a certificate signed by another certificate other than yourself, known as a root certificate.



Chained Certificates (cont.)

- Chained Certificate Attributes
  - Self-Signed Root Certificate
    - **Alias:** root
    - **DN :** CN=${hostname}, OU= Root Certificate, OU=<cell name>, OU=<node name>, O=IBM,C=US
    - **Validity Period:** 20 years
    - **KeyStore:** DmgrDefaultRootStore
    - NodeDefaultRootStore
  - Chained Certificate (Signed by root certificate)
    - **Alias:** default
    - **Issued to DN:** CN=${hostname}, OU=<cell name>, OU=<node name>,O=IBM,C=US
    - **Issued by DN:** CN=${hostname}, OU= Root Certificate, OU=<cell name>, OU=<node name>, O=IBM,C=US
    - **Validity Period:** 1 year
    - **KeyStore:** CellDefaultKeyStore
      NodeDefaultKeyStore

Chained Certificates (cont.)

- New console panels and task provided to create chained certificates
  - SSL certificate and key management > Key stores and certificates > <keyStore> > Personal certificates > Create Chained Certificate
- Chained certificate can only be signed by a certificate in the DefaultRootStore.

Chained Certificates (cont.)

SSL certificate and key management > **Key stores and certificates**

Defines keystore types, including cryptography, RACF(R), CMS, Java(TM), and all truststore types.

Keystore usages

Root certificates keystore

- All
- SSL keystores
- RSA token keystores
- Key set keystores
- Root certificates keystore
- Deleted certificates keystore
- Default signers keystore

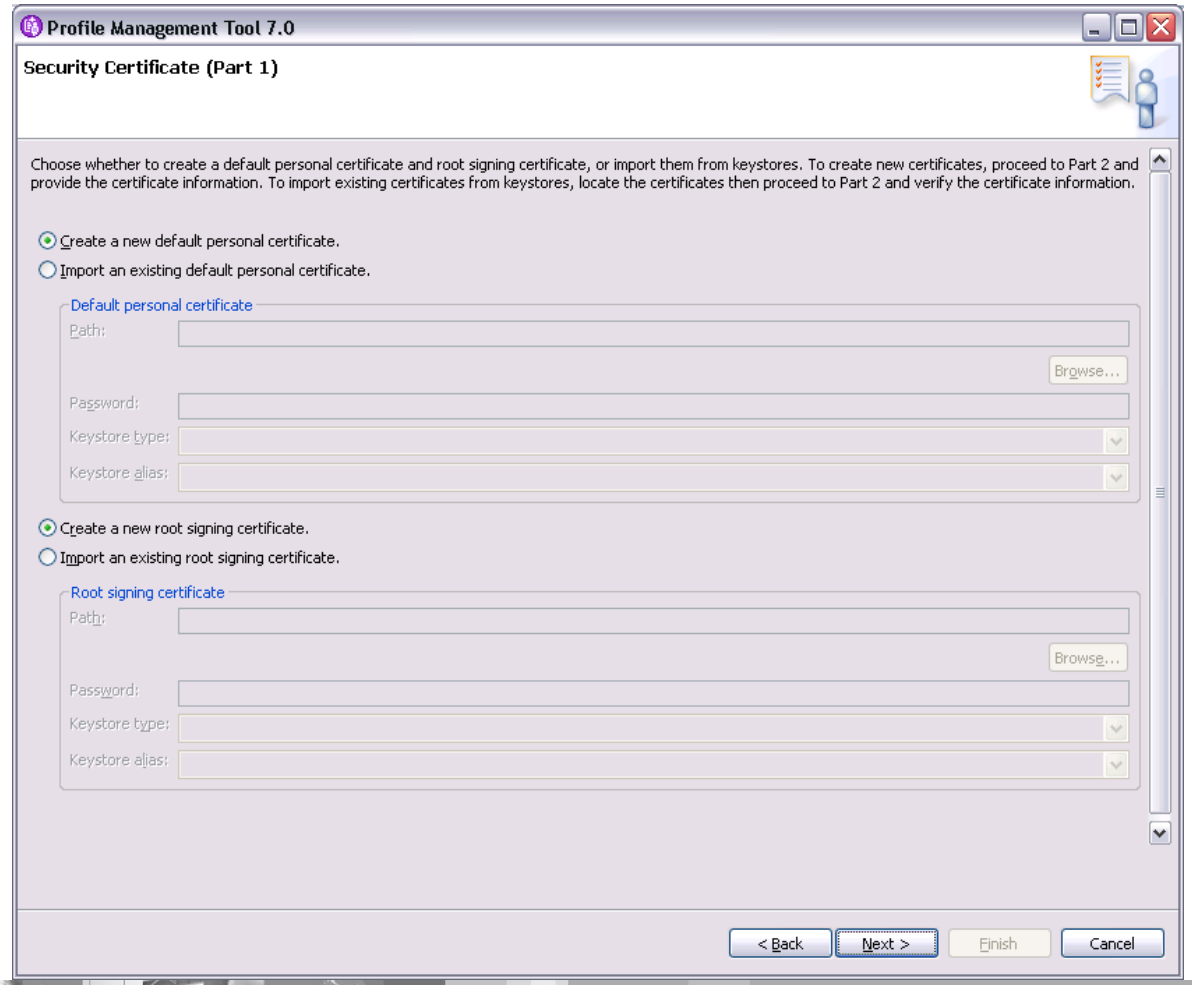
Exchange signers...

Select	Name	Description	Management Scope	Path
You can administer the following resources:				
☐	DmgrDefaultRootStore	Root certificate key store for brettoCellManager01	(cell):brettoCell01: (node):brettoCellManager01	${CONFIG_ROOT}/cells /brettoCell01/nodes /brettoCellManager01/root-key.p12
Total 1				

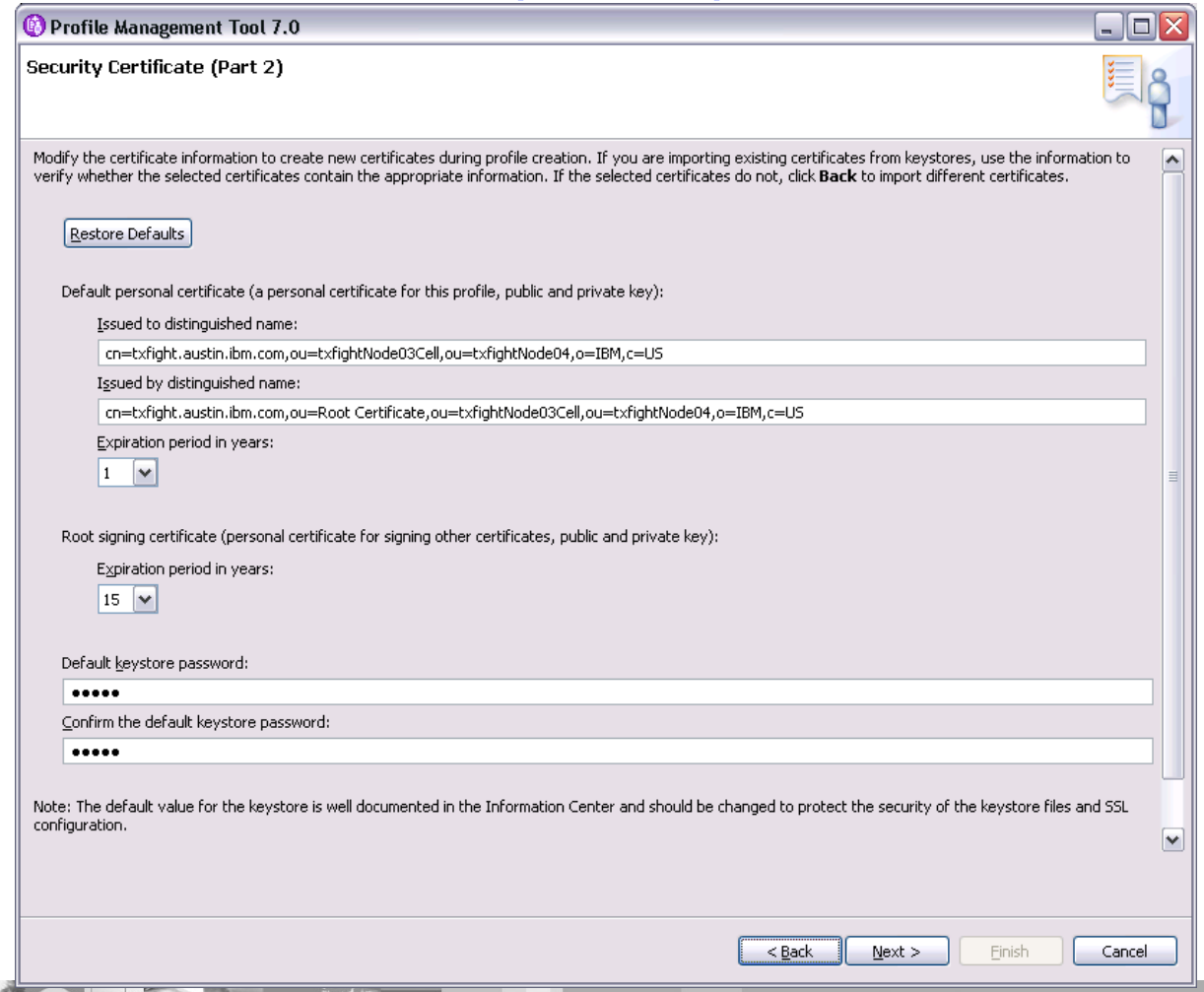
Chained Certificates (cont.)

- During profile creation users will have the opportunity to make decisions about the servers default certificates.
  - Available on the profile creation advanced path
  - Customize the DN of the default signing certificate and default certificate.
  - Set the life span of the signing certificate and default certificate
  - Import a certificate to be the root signing certificate or default personal certificate
  - Provide a custom password for the key stores created during profile creation.
    - This password applies to ALL key stores created

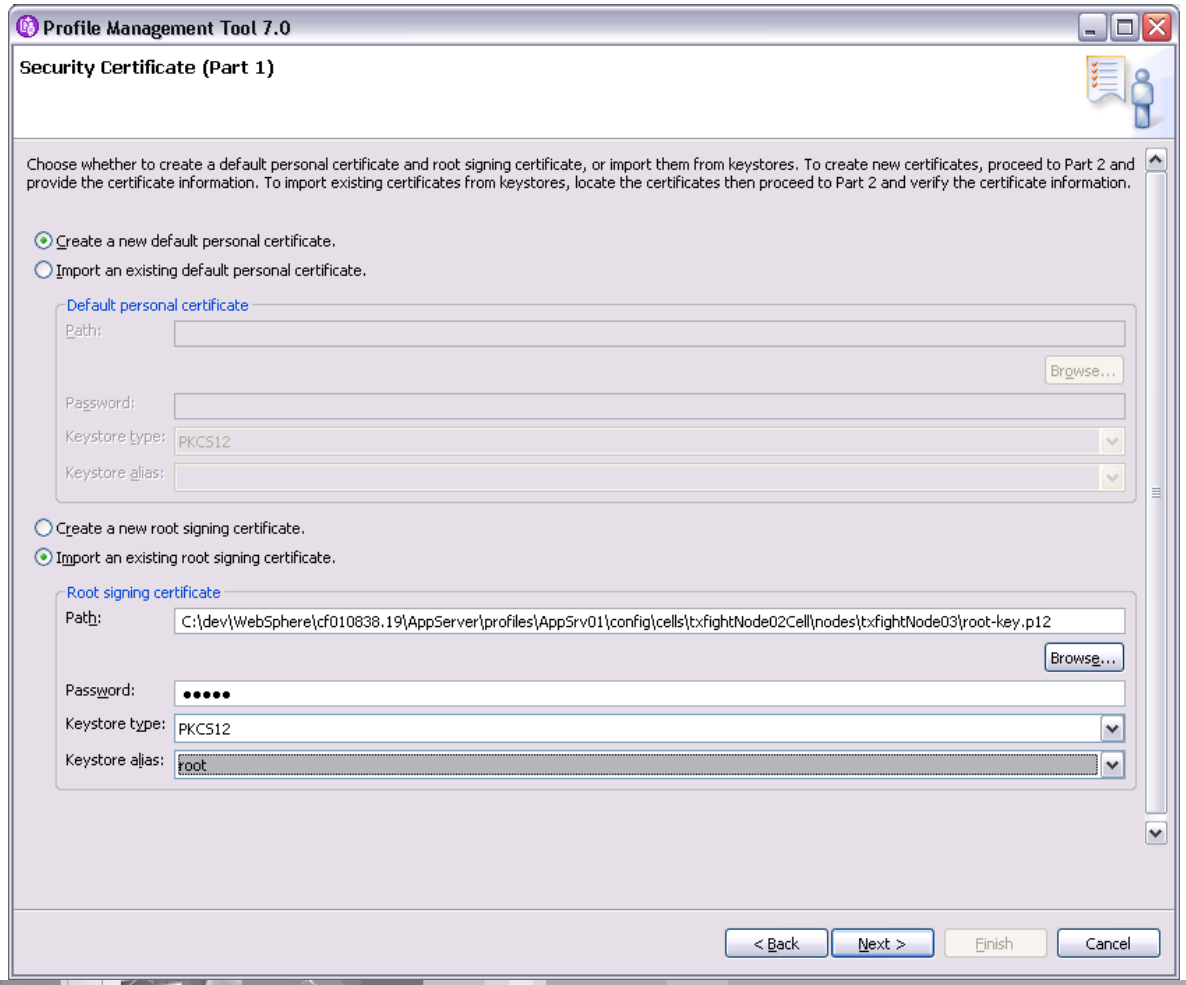
Chained Certificates (cont.)



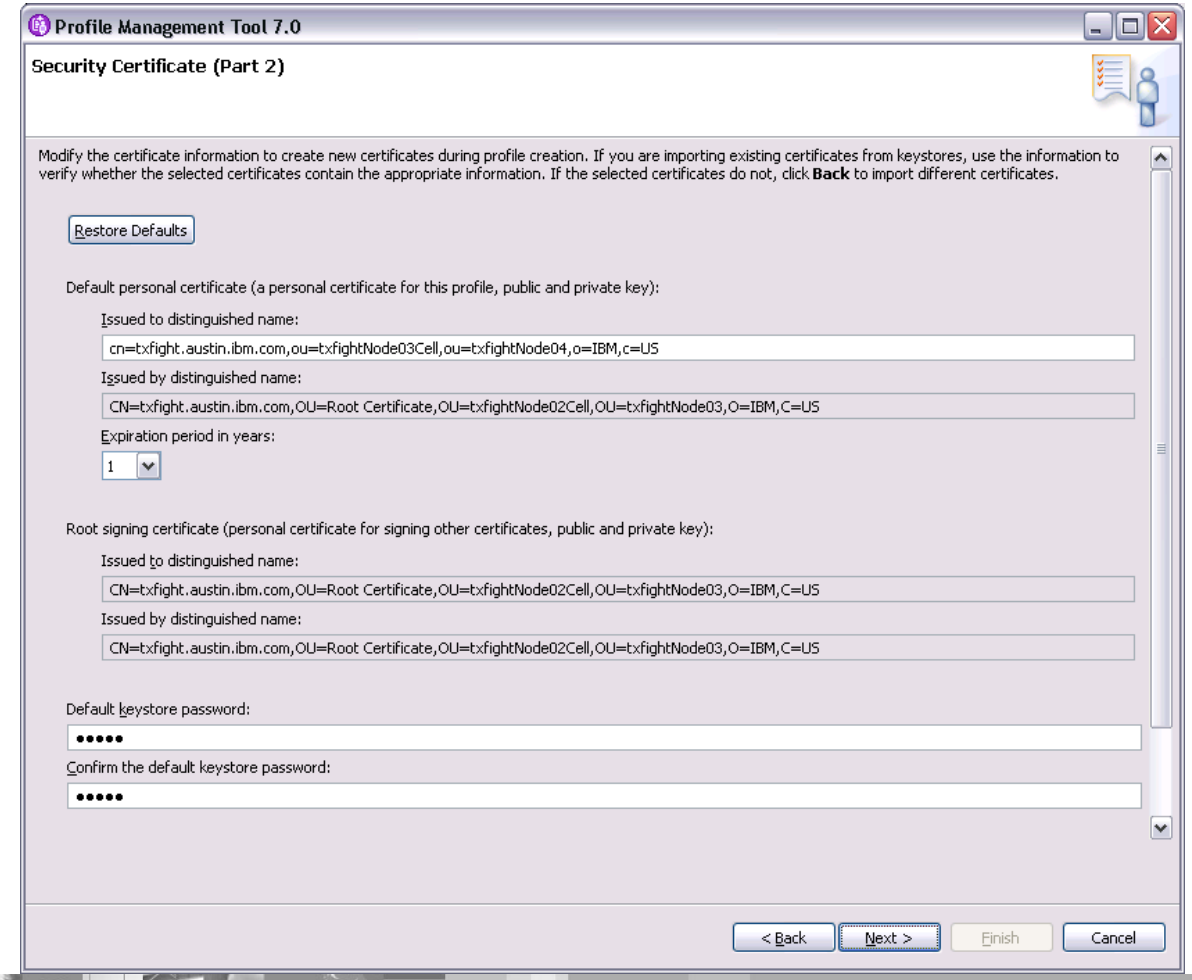
Chained Certificates (cont.)



Chained Certificates (cont.)



Chained Certificates (cont.)



Renewing Certificates

- Renews a certificate. Creates a new certificate with all the information used to create the original certificate (DN, keySize, Validity). Old signers found in the configuration are either replaced or kept in the configuration depending on the option specified.
- Renew can only be performed on self-signed certificates and chained certificates.
- Externally signed CA certificates must be renewed manually by an administrator.

Renewing Certificates (cont.)

- Can be done from the console
  - SSL certificate and key management > Key stores and certificates > <keyStore> > Personal certificates >
  - Select a personal certificate
  - Select renew
- For scripting the renewCertificate task can be used.
  - **AdminTask.renewCertificate('-keyStoreName *myKS***
    -certificateAlias *testCertificate*')

Renewing Certificates (cont.)

SSL certificate and key management > Key stores and certificates > NodeDefaultKeyStore > Personal certificates

Manages personal certificates.

Preferences

Create | Delete | Receive from a certificate authority... | Replace... | Extract... | Import... | Export... | Revoke... | Renew

Self-signed Certificate...
CA-signed Certificate...
Chained Certificate...

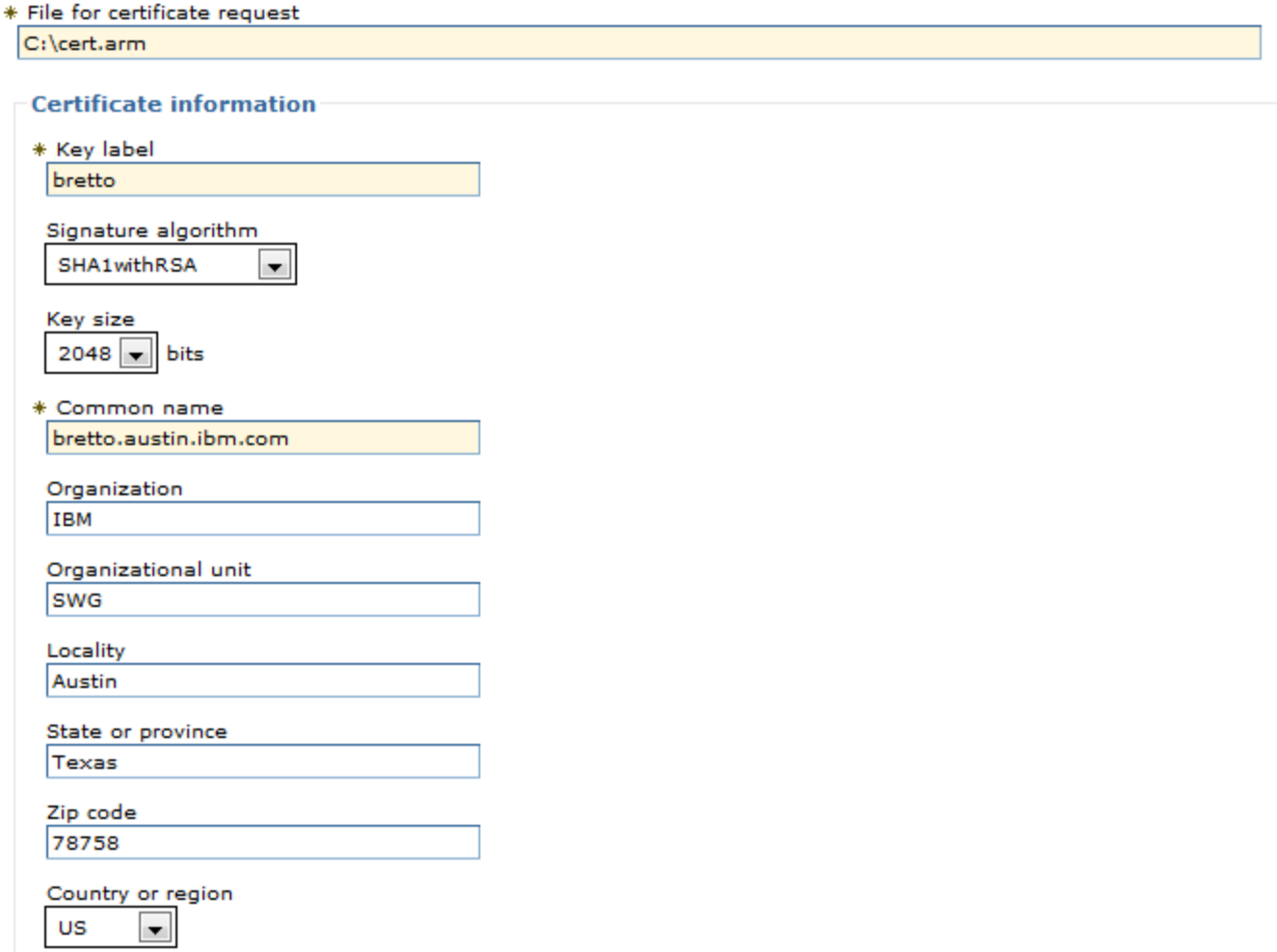
		Alias	Issued To	Issued By	Serial Number	Expiration
			ollowing resources:			
☐		default	CN=bretto, OU=brettoCell01, OU=brettoCellManager01, O=IBM, C=US	CN=bretto, OU=Root Certificate, OU=brettoCell01, OU=brettoCellManager01, O=IBM, C=US	27141016537256	Valid from Feb 29, 2012 to Feb 28, 2013.
			CN=bretto, OU=Root Certificate, OU=brettoCell01, OU=brettoCellManager01, O=IBM, C=US	CN=bretto, OU=Root Certificate, OU=brettoCell01, OU=brettoCellManager01, O=IBM, C=US	27139559901808	Valid from Feb 29, 2012 to Feb 25, 2027.

Total 2

Personal Certificate Requests (CA signed)

- Personal certificate requests are temporary place holders for certificates that will be signed by a certificate authority (CA)
- The private key is generated during the certificate request generation, but only the certificate is sent to the CA. The CA generates a new certificate, signed by the CA.
  - SSL certificate and key management > Key stores and certificates > NodeDefaultKeyStore > Personal certificate requests > New...

Personal Certificate Requests (cont.)



Personal Certificate Requests (cont.)

Personal Certificate Requests (cont.)

-----BEGIN NEW CERTIFICATE REQUEST-----
MIIC7zCCAdcCAQAwejELMAkGA1UEBhMCVVMxDjAMBgNVBBETBTc4NzU4MQ4wDAYDVQQIEwVUZXhh
czEPMA0GA1UEBxMGQXVzdGluMQwwCgYDVQQKEwNJQk0xDDAKBgNVBAsTA1NXRzEeMBwGA1UEAxMV
YnJldHRvLmF1c3Rpbi5pYm0uY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAn8Ip
zN4NmLzYkIwvYYp/zv2P1wk3zcDwWgX1aWelcwJeyw0tbT6/TaCjGG0w+BvOkMvf3GF/8cV6vxZN
U1FW3ubs6zSpxrGIZhUuzWPxibtOs8W4Um4mN1WHEDYPnpDjv9/EqfAkki4sQsK69E2VKi1IXm1I
00tc/FLiuFyqVBoTe5+t3UxET18aNmzpx2AiAbDYyg/fdMbfbfUQY+7F9IpDUbGVMdzyLYBI1Zme
926nKLNQDab/UxDXBioooUW7Oo+J0SpOZLj2Y09w1ZCrQsHT+Qgk2fWvkB/U5YnvoE6quw4Ku5Tn
MtNIVqEDCjjKHvgwPRQUOn+sBUW61LwZ+QIDAQABoDAwLgYJKoZIhvcNAQkOMSEwHzAdBgNVHQ4E
FgQUiAlxceczNWGUFn63zeUiQK6dBJAwDQYJKoZIhvcNAQEFBQADggEBAFKjq9eAuSZzkMDr4M6i
9rU7+s+9Kc/7XcSHnFqLO3wWfP8ScoiOuiRPbadyGuyX0VVTwAtGiaJUS381ARl3s7BanY9BVH0y
+gtat/uky2uX+5iY5WyH8xLUQ8Vsdpsxk+ndndC6yRP8YyC7uugqyOUbwTp/iTu97zNEsOfnlJZh
dPR+Dmoyke+XAUfZWgHblNhcZ2ePHAwqKDPY+J4K5wiTTLiM0VPySD9ck11WO1gkxtY+s2j72AaA
lebB+JwJMAZ1sip1bkvPw1gMwRVA14ZgnYKDPaUM3hROKKpXgjBgNUICNwPzVQAf9BjlSjPc5tt/
6Nwt9KqmnZgDCPdBgeY=
-----END NEW CERTIFICATE REQUEST-----

Personal Certificate Requests (cont.)

- When a certificate authority (CA) receives a certificate request, it issues a new certificate that functions as a temporary placeholder for a CA-issued certificate. A keystore receives the certificate from the CA and generates a CA-signed personal certificate that WebSphere® Application Server can use for Secure Sockets Layer (SSL) security.

Personal Certificate Requests (cont.)

- WebSphere Application Server can receive only those certificates that are generated by a WebSphere Application Server certificate request. It cannot receive certificates that are created with certificate requests from other keystore tools, such as **iKeyman** and **keyTool**.
  - SSL certificate and key management > Key stores and certificates > NodeDefaultKeyStore > Personal certificates > Receive certificate from CA

KeyStores

- Cell (managed):
  profile_root/config/cells/*cell_name*/key.p12
  profile_root/config/cells/*cell_name*/trust.p12
- Node (managed):
  profile_root/config/cells/*cell_name*/nodes/*node_name*/key.p12
  profile_root/config/cells/*cell_name*/nodes/*node_name*/trust.p12
- Each profile also has (unmanaged):
  profile_root/etc/key.p12
  profile_root/etc/trust.p12

KeyStores (cont.)

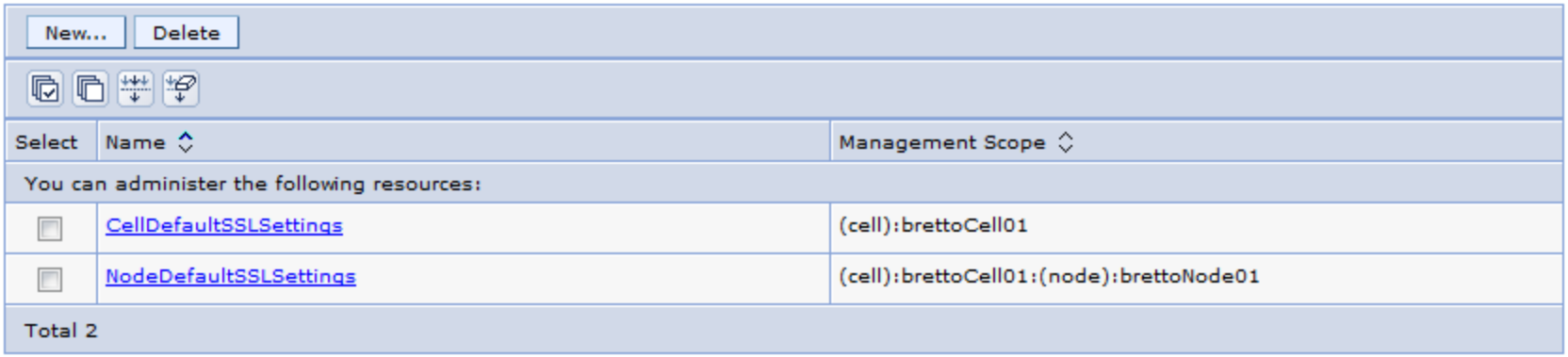
- SSL certificate and key management > Key stores and certificates

New... | Delete | Change password... | Exchange signers...

Select	Name	Description	Management Scope	Path
You can administer the following resources:				
☐	CellDefaultKeyStore	Default key store for brettoCell01	(cell):brettoCell01	${CONFIG_ROOT}/cells /brettoCell01/key.p12
☐	CellDefaultTrustStore	Default trust store for brettoCell01	(cell):brettoCell01	${CONFIG_ROOT}/cells /brettoCell01/trust.p12
☐	NodeDefaultKeyStore	Default key store for brettoNode01	(cell):brettoCell01: (node):brettoNode01	${CONFIG_ROOT}/cells /brettoCell01/nodes /brettoNode01/key.p12
☐	NodeDefaultTrustStore	Default trust store for brettoNode01	(cell):brettoCell01: (node):brettoNode01	${CONFIG_ROOT}/cells /brettoCell01/nodes /brettoNode01/trust.p12
Total 4				

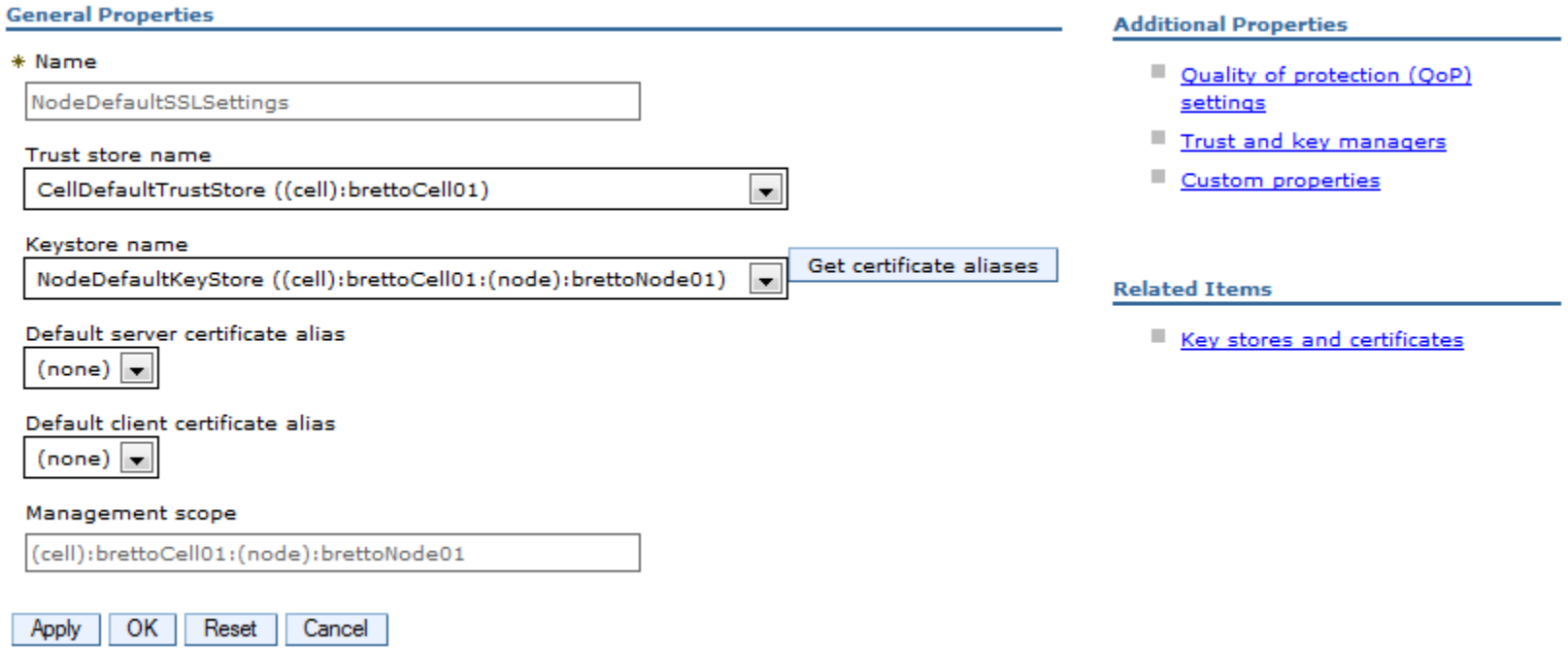
KeyStores (cont.)

- SSL certificate and key management > SSL configurations



KeyStores (cont.)

- SSL certificate and key management > SSL configurations > NodeDefaultSSLSettings



New KeyStores

- **DefaultRootStore**
  - Key store to hold self signed root certificates. Chained certificate can only be created using root certificates from this key store.
  - root-key.p12
- **DefaultSignersStore**
  - Holds all signer certificates that get added to any key store created. By default the root signer is included.
  - default-signers.p12
- **DefaultDeletedStore**
  - Certificates deleted from other key stores are temporarily stored in this key store.
  - deleted.p12
- **RSA Key Stores**
  - rsatoken-root-key.p12
  - rsatoken-key.p12, rsatoken-trust.p12

New KeyStores (cont.)

- Default Signers Key Store
  - The DefaultSignersStore contains all signer certificates added by customers that they wish to be added to newly created keystores.
  - This can be use to establish trust at the time of creating a new key store, saving multiple import steps.
  - Dummy signers not included by default
    - Dummy signers still shipped with the product, can be added manually though not recommended

New KeyStores (cont.)

- DefaultDeletedStore
  - Created to hold deleted certificates.
  - Allows users to restore or permanently delete a certificate.
  - Requests to delete a certificate will move the certificate to the DefaultDeletedStore. The deleted certificate will be stored with the alias name "keystorename_alias_uniquenum".
  - Certificate Expiration Monitor will clean out the DefaultDeletedStore.

Certificate Expiration Monitor

- Add the ability to replace/renew the new certificate types.
  - Chained certificates that we have a root in the DefaultRootStore.
- Handle expiring root certificates.
  - Root certificate expiring will require renewing of all certificates that are created with that root.
- Clean out the deleted key store.

Certificate Expiration Monitor Output

**** Subject: Expiration Monitor ****;

Hostname: bretto
Profile UUID: Dmgr01-bretto
Process type: DeploymentManager

*** CERTIFICATES WITHIN THE 90 DAYS OF THE CERTIFICATE EXPIRATION THRESHOLD (MAY BE REPLACED WITHIN 90 DAYS) ***;

CWPKI0714I: The certificate expiration monitor has recently run and discovered that the certificates, which are listed in associated messages, will be replaced within the next 90 days. This replacement is based on the configured policy to automatically replace expiring self-signed certificates 60 days prior to expiration. This notification is informs you that problems might arise when the certificates are automatically replaced.
CWPKI0715I: In some cases, automatically replacing certificates can cause outages for Web server plug-ins operating on unmanaged nodes. In such a situation, the plug-in will be unable to contact the application servers over HTTPS because it will be using signers for certificates that have been replaced by the automatic replacement process. To prevent what may be and serious outage you should act before the scheduled replacement date and replace the expiring certificates and update the plug-in kdb to use the new signers.
CWPKI0719I: The test personal certificate in the "TestKeyStore((cell):brettoCell01)" keystore is due to expire on July 16, 2012 and might be replaced after the May 17, 2012 threshold date.

Summary

- Chained Certificates
- Renewing Certificates
- Personal Certificate Requests
- KeyStores
- Certificate Expiration Monitor

Additional WebSphere Product Resources

- Learn about upcoming WebSphere Support Technical Exchange webcasts, and access previously recorded presentations at:
  http://www.ibm.com/software/websphere/support/supp_tech.html
- Discover the latest trends in WebSphere Technology and implementation, participate in technically-focused briefings, webcasts and podcasts at:
  http://www.ibm.com/developerworks/websphere/community/
- Join the Global WebSphere User Group Community:
  http://www.websphere.org
- Access key product show-me demos and tutorials by visiting IBM ® Education Assistant:
  http://www.ibm.com/software/info/education/assistant
- View a webcast replay with step-by-step instructions for using the Service Request (SR) tool for submitting problems electronically:
  http://www.ibm.com/software/websphere/support/d2w.html
- Sign up to receive weekly technical My Notifications emails:
  http://www.ibm.com/software/support/einfo.html

Connect with us!

1. **Get notified on upcoming webcasts**
   Send an e-mail to wsehelp@us.ibm.com with subject line “wste subscribe” to get a list of mailing lists and to subscribe
2. **Tell us what you want to learn**
   Send us suggestions for future topics or improvements about our webcasts to wsehelp@us.ibm.com
3. **Be connected!**
   Connect with us on Facebook
   Connect with us on Twitter

Questions and Answers