An Investigation of Certificate Authority / PKI Capability in Microsoft .NET Server

Paper Submitted in Partial Fulfillment of the Requirements for the SANS/GIAC GSEC Certification, Version 1.4, Option 1

Michael Lynn Devore

Referenced web resources were verified as current on September 3, 2002.

Windows, .NET, XP, and NT are trademarks of Microsoft Corporation.
eToken is a trademark of Aladdin Knowledge Systems.

Introduction

The Microsoft.NET Server enterprise environment includes enhanced Certificate Authority functionality, improved integration into the forest and domain, and easier use for the end user.

This paper surveys the available literature, identifying the product capabilities and differences compared with earlier Windows products. Live lab work with the Release Candidate-1 software confirms and illustrates product details.

Many significant new features depend on use of the XP Professional client (not the “Home” version) and .NET Enterprise Server (rather than the entry-level Server), and this paper is presented in the context of those products.

Microsoft PKI History

A Certificate Authority (CA) was present in both Microsoft Windows NT4 and in Windows 2000. As with most maturing products, both ease of implementation / use and the technical capabilities have been refined as later releases have been offered.

Across the life of the NT4 product, service pack updates provided fixes and expanded the capability of the CA. Of particular interest, Service Pack 6 (SP6) provided the capacity for the NT4 Certificate Server to join a non-Microsoft X.509 certificate hierarchy, showing an increasingly enterprise scope for the service.[1]

[1] Certificate Server Updated to Use non-Microsoft X.509 Certificate Authorities, TechNet, Microsoft Corporation (undated)
http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q242031&

It was already possible at the release level of NT4 to use a non-Microsoft CA in a Windows domain.[2]

Another important enhancement was the coupling of the Certificate Authority with the Active Directory, which was introduced along with the initial Windows 2000 product. Providing the capability automatically to publish certificate information into the enterprise directory was a big step in ensuring its easy use in the corporate world.

As a further example of the integration of this technology by Microsoft, PKI / smart cards are also now supported on the Windows CE .NET devices, bringing compatible technology to small mobile devices.[3]

.NET / XP PKI Enhancements -- Overview

With the Microsoft .NET Certificate Authority and accompanying XP desktop come a number of enhancements to the product. These include:

Automatic issuance of digital certificates to users

In the pure Windows 2000 environment, certificates could be automatically issued to machines to support such purposes as IPSEC and L2TP/IPSec VPN connections between machines. However, user certificates were not supplied in this manner.

In the .NET / XP environment, a .NET Certificate Authority operating in a forest incorporating the .NET schema updates can automatically issue a certificate to a user who is performing a logon from an XP desktop.[4] The automatic issuance is controlled by Group Policy Objects, with permissions assigned to certificate templates and user membership in Security Groups, and thus provides a good granularity of control over the identification of subsets of users for inclusion in the process.

It is especially useful that .NET domain controllers are not required for this feature to operate. It is fully operational in a forest comprising Windows 2000 Domain Controllers, once the .NET schema extensions have been applied. The .NET Server running Certificate Services and the XP client desktop both access

[2] How to Use the Directory Services Store Tool to Add a Non-Windows 2000 Certification Authority (CA) to the PKI in Windows 2000, TechNet, Microsoft Corporation (undated) http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q313197&

[3] What's New in Microsoft Windows CE .NET. Microsoft Corporation, July 31, 2002 http://www.microsoft.com/windows/embedded/ce.net/evaluation/whatsnew/default.asp

[4] Cross, David B., Certificate Autoenrollment in Windows XP. Microsoft Corporation, January 2002 http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/winxppro/maintain/certenrl.asp

the extended schema information to carry out this process.

User auto-enrollment significantly reduces one of the major costs involved in a standard PKI implementation, and directly enables use of a large suite of certificate-enabled activities. Of course, configuration allows manual or automatic request of certificates by the client, as well as manual or rule-based approval of the request at the server. Renewal of certificates may also be accomplished automatically.

The feature is especially important in providing certificate-based services to large user populations, because the logistics of manual certificate requests do not scale well to enterprise use. When combined with the automatic publishing to Active Directory, user acceptance for processes such as encrypted email within an organization should be high.

Version 2 Certificate Templates are available in the .NET server product to extend the range of functionality for auto-issued certificates, and to allow customization of the templates for the specific business and application requirements of a business.

Email and Secure Web Access

Use of email client data encryption and digital signatures is as simple as selecting a return receipt for a message for the end user. The Exchange Server Key Management System (KMS), which provided key services for the Exchange 5.5 product, was unneeded once the Exchange Global Address Book (GAL) was integrated with the single Active Directory at the Exchange 2000 product level.

Client-side certificates allowing secure client authentication for SSL web access can be automatically provided to the client without any requirement for action by the user.

True Cross-Certification and Qualified Subordination

The Windows 2000 certificate authority did not provide a cross-certification capability. A mechanism known as Certificate Trust Lists could be used by a CA to identify specific other certification authorities and certificate usages with which it would interoperate.

True enterprise cross-certification becomes available in the .NET context, between target CA's in two organizations.[5] A range of qualifications may be applied to the cross-certification, limiting the authority based on namespace

[5] Komar, Brian, Planning and Implementing Qualified Subordination using Windows .NET Enterprise Server, Microsoft Corporation, March 2002: 1 – 5

(DNS, IP address, e-mail address, User Principal Name or UPN), supported applications, issuance policy, and the path-length for a certificate chain. Federal bridge requirements are also met by the product, allowing equivalence between two certification authorities.[6]

Delta CRL's

Efficiency in the checking of certificate validity is important if the CA is to scale to enterprise size. Large organizations can accumulate many revoked certificates, and allowing modifications (or "delta's") to the Certificate Revocation List (CRL) to be provided the client, can make the certification verification process much more efficient, at significantly lower network cost.

In many instances, updates can be transmitted for validation purposes, rather than transmitting the whole list. The ability to publish Delta CRL's as defined in RFC 2459 is supported by the Microsoft .NET CA. Accordingly, the XP client always checks for the availability of a suitable Delta CRL first during validation chaining.[7]

Use of Smart Card Authentication in a Thin Client / Terminal Services Environment

Thin Client use provides flexibility in access in Windows systems, and is often used for specialized applications, such as systems administration tools used to control domain function.

While smart card logon was supported in Windows 2000 environments, it was only in the context of logons at the local machine. By extending the capability to Terminal Services operation, Microsoft has enabled multi-factor authentication to be used in these additional contexts.[8]

Additionally, in the .NET/XP products, the *net.exe* and *runas.exe* utilities also recognize smart card authentication, further extending the ability of administrators to operate effectively from different physical and logical locations in an enterprise.

[6] Freeman, Dr. Trevor, Windows XP PKI Client Windows .NET Server Public Key Infrastructure, Microsoft Corporation: 4, 7.
http://www.microsoft.com/usa/presentations/Freeman_SecuritySummitWest.ppt

[7] Komar, Brian, Troubleshooting Certificate Status and Revocation, Microsoft Corporation, October 2001: 40 -- 48
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/winxppro/support/tshtcrl.asp

[8] Cross, David, PKI Enhancements in Windows XP Professional and Windows .NET Server, Microsoft Corporation, July 2001: 11
http://www.microsoft.com/windowsxp/pro/techinfo/planning/pkiwinxp/PKIEnhancements.doc

Encrypted File System (EFS) Enhancements

When certificates issued by a CA are used in the EFS mechanism, the XP client is able to check the validity chain of those certificates. EFS is also available in conjunction with WebDAV, allowing secure sharing of data through http connections.

The XP client also allows multiple user accounts, both on the local machine and in Active Directory, to share access to encrypted files. This provides both flexibility of use and an alternate recovery path in the event of loss of the original account access.

User Key Escrow

While security of the private key is a cornerstone of a quality PKI implementation, it may be required in an enterprise context to be able to escrow the keyset at a high administrative level, to address future contingencies.

Users leaving the company, and user errors in operation / configuration of their desktops provide two instances in which significant company-owned data may not be recoverable without escrowed keys. This capability now exists in the .NET product.[9] The original keys may be provided to a user who has lost them, or an administrator may assume the identity of the user for purposes of data access.

[9] Cross, David, PKI Enhancements in Windows XP Professional and Windows .NET Server, Microsoft Corporation, July 2001: 19 -- 20
http://www.microsoft.com/windowsxp/pro/techinfo/planning/pkiwinxp/PKIEnhancements.doc

Lab Work

To verify information obtained from the cited web and published resources, lab work with the .NET Certificate Authority was conducted between March and August 2002.

Initially, the Beta-3 version of .NET was used for the testing; Release Candidate 1 (RC-1) was used after its release in late July to verify results of previous tests and to extend the testing of the models.

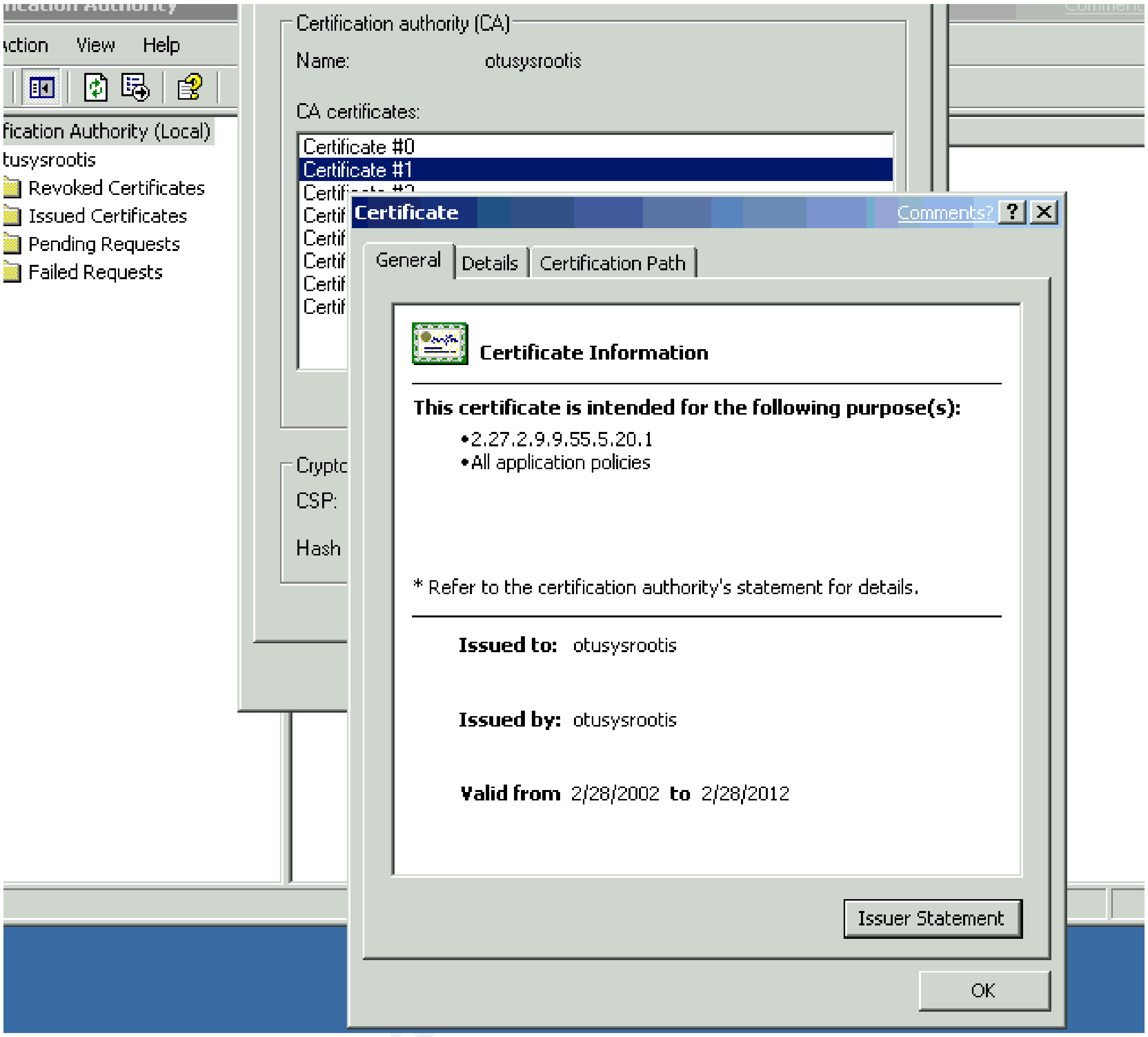
Single Forest Model

Testing began with the configuration of an offline root certificate authority running on .NET Enterprise Server. The Enterprise version of the product was chosen since it has the full feature set, compared to the baseline Server version.

A *policy.inf* file was used to supply the initial configuration for the new root server. When this file was read during initialization of the certificate authority, the processes initialized with proper configuration details.[10]

The root CA certificate is self-signed, with the “Issued to” and “Issued by” fields identical, and with usage for all application policies, as shown in Figure 1 below.

[10] Cross, David B., Best Practices for Implementing a Windows .NET PKI. Microsoft Corporation, April, 2002: 26 – 32.



Certification authority (CA)

Name: otusysrootis

CA certificates:

Certificate #0
Certificate #1

Certificate Information

This certificate is intended for the following purpose(s):

- 2.27.2.9.9.55.5.20.1
- All application policies

* Refer to the certification authority's statement for details.

Issued to: otusysrootis

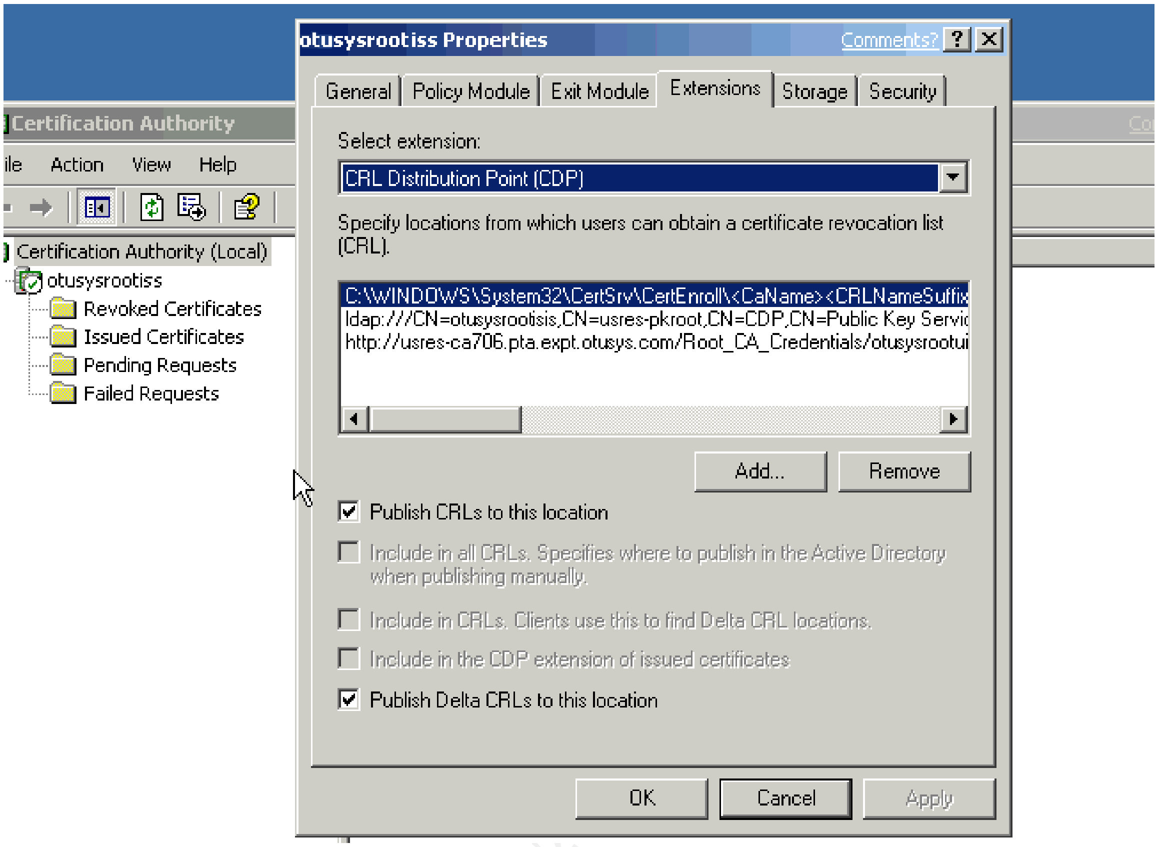
Issued by: otusysrootis

Valid from 2/28/2002 **to** 2/28/2012

Figure 1

Distribution points for the root Certificate Revocation List (CRL) were chosen both in ldap space on the forest’s Active Directory, and on filename space for access by requestors without directory access. The list can also be published in http space for access by web components.[11] Figure 2 shows the CRL Extensions employed by the root.

[11] Cross, David B., Best Practices for Implementing a Windows .NET PKI. Microsoft Corporation, April, 2002: 51.



Select extension:

CRL Distribution Point (CDP)

Specify locations from which users can obtain a certificate revocation list (CRL).

```
C:\WINDOWS\System32\CertSrv\CertEnroll\<CaName><CRLNameSuffix
ldap:///CN=otusysrootisis,CN=usres-pkroot,CN=CDP,CN=Public Key Servic
http://usres-ca706.pta.expt.otusys.com/Root_CA_Credentials/otusysrootui
```

☑ Publish CRLs to this location

□ Include in all CRLs. Specifies where to publish in the Active Directory when publishing manually.

□ Include in CRLs. Clients use this to find Delta CRL locations.

□ Include in the CDP extension of issued certificates

☑ Publish Delta CRLs to this location

Figure 2

Removable media were used to acquire certificates for the offline root CA for two Enterprise CA's. Note that Enterprise CA's are defined as those which are coupled to the Windows forest/domain structure. They are capable of automatic certificate issuance based on AD configuration and configurable certificate templates, and they can publish to Active Directory. This usage of the term "enterprise" may initially be a point of confusion, and is unrelated to whether they are based on the basic Server or Enterprise Server platforms, or even whether they are functioning in a business enterprise, though presence of a forest does imply some organizational hierarchy.

These subordinate issuing authorities were one step lower than the root in the hierarchy, as shown in Figure 3. A two-level hierarchy was chosen to illustrate the operation of the offline-rooted hierarchy, and to economize in the use of laboratory equipment; however, a three-level hierarchy would be the usual choice for deployment.

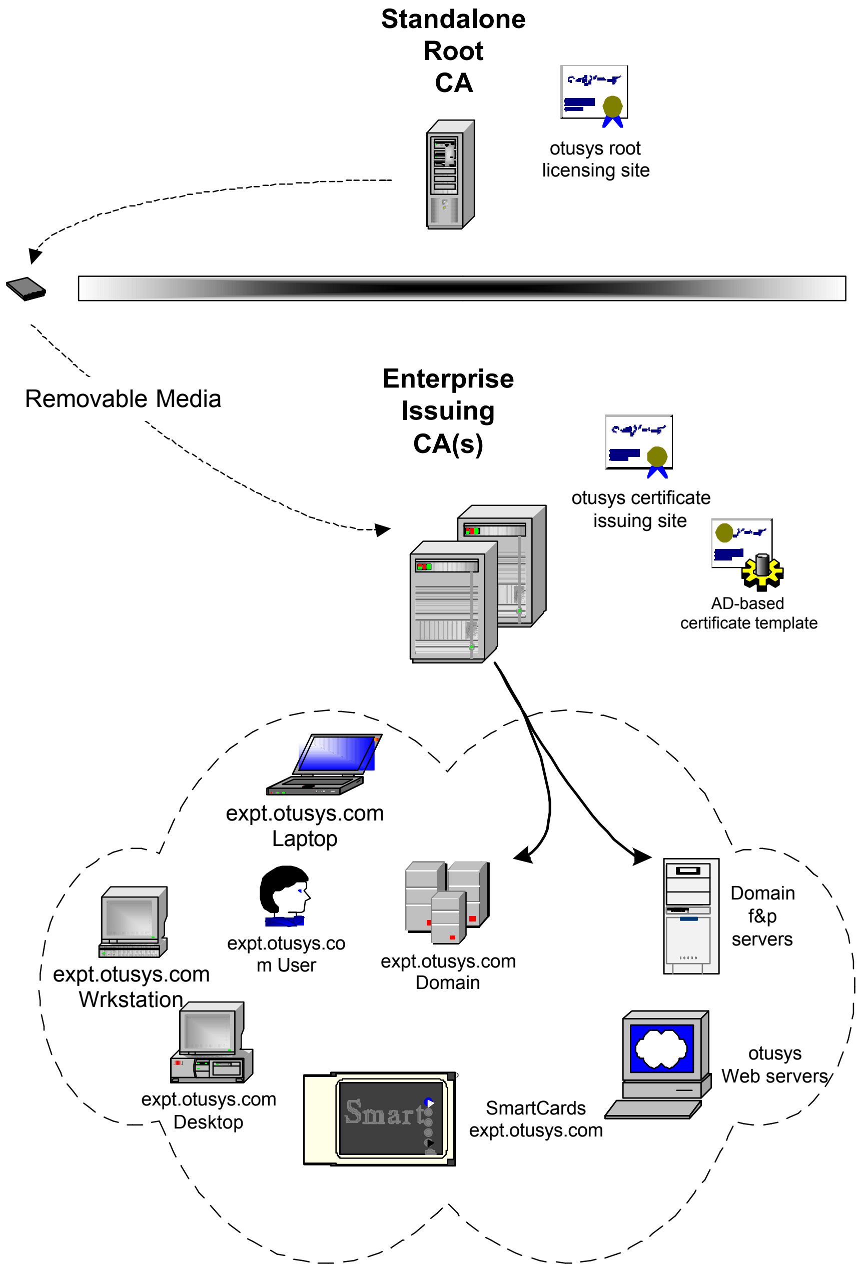
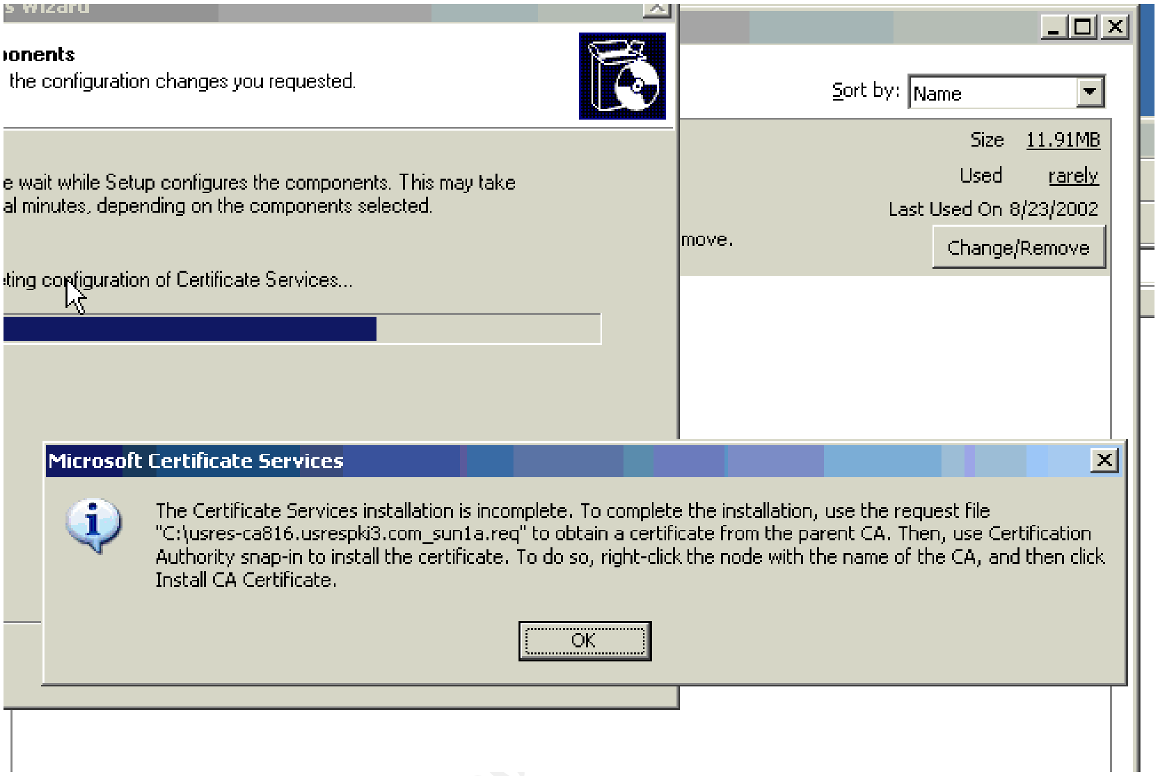


Figure 3

Enterprise (Active Directory-coupled) CA's were established subordinate to the offline root. Since the root was offline, the requests were saved to a request file (*.req*) on removable media, which was then physically transported to the root CA. The subordinate request was then processed using the Certificate Authority snap-in on the root CA, by opening the request, processing and issuing it, and then exporting it as binary data back to the removable media as a **.cer* file format. Finally, this issued certificate was installed on the CA which originated the request. Figure 4 shows the advice from the subordinate server on obtaining a certificate from the parent.

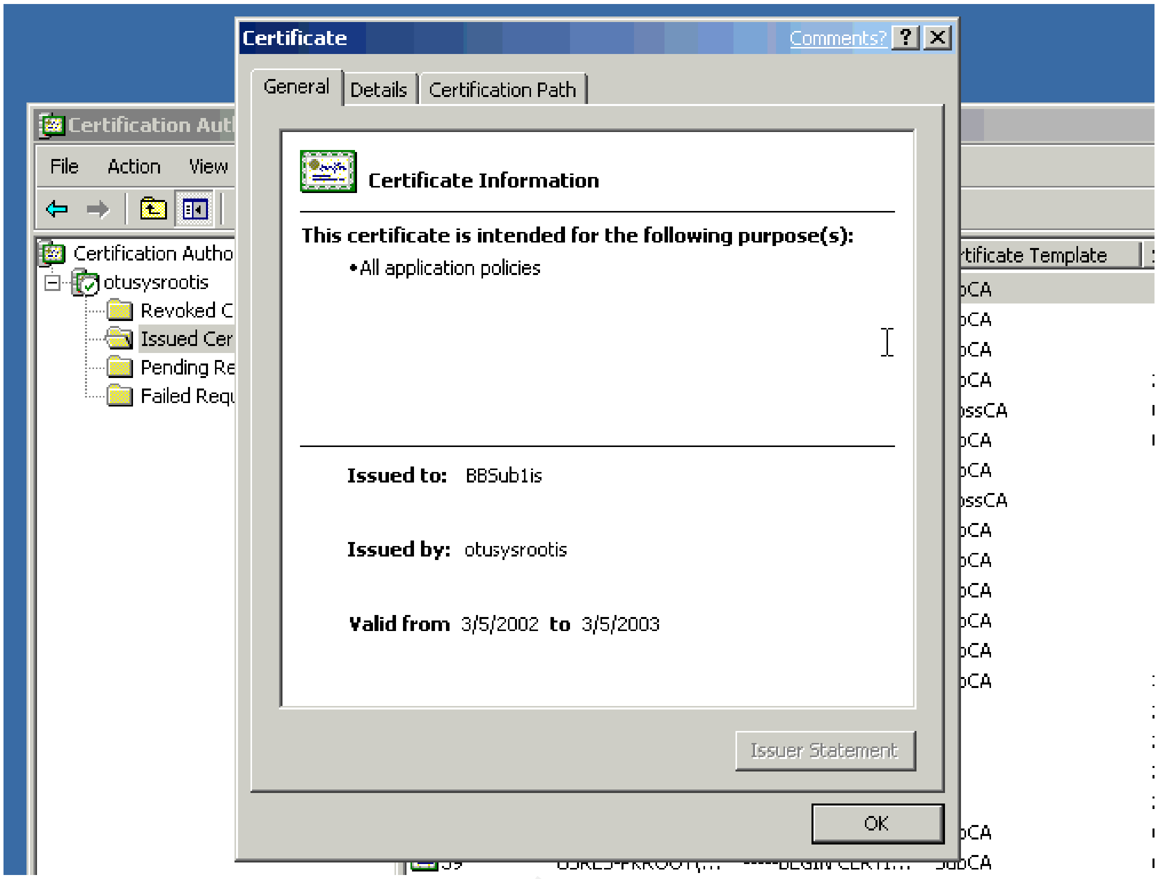


s wizard
ıonents
the configuration changes you requested.
e wait while Setup configures the components. This may take
al minutes, depending on the components selected.
ting configuration of Certificate Services...
Sort by: Name
Size 11.91MB
Used rarely
Last Used On 8/23/2002
move.
Change/Remove
Microsoft Certificate Services
OK

The Certificate Services installation is incomplete. To complete the installation, use the request file "C:\usres-ca816.usrespki3.com_sun1a.req" to obtain a certificate from the parent CA. Then, use Certification Authority snap-in to install the certificate. To do so, right-click the node with the name of the CA, and then click Install CA Certificate.

Figure 4

The certificate supplied to the subsidiary is shown in Figure 5, and it is valid for all application policies.



Certificate Information

This certificate is intended for the following purpose(s):

- All application policies

Issued to: BBSub1is

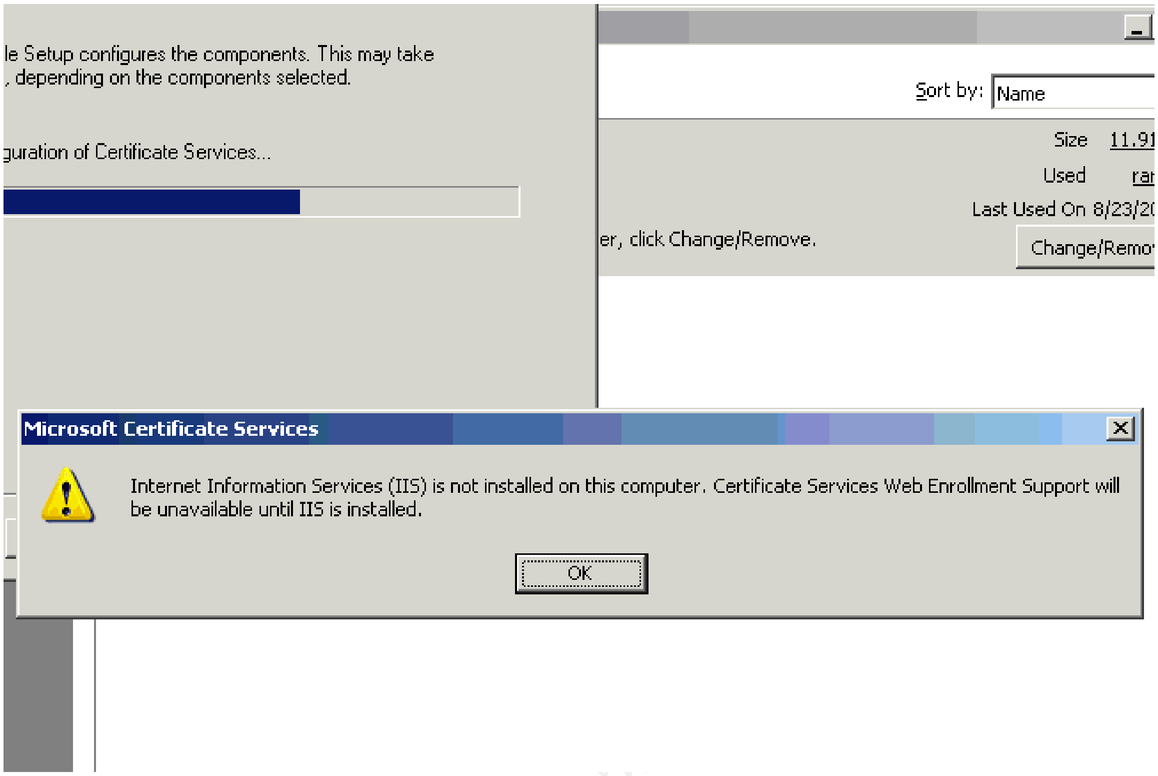
Issued by: otusysrootis

Valid from 3/5/2002 to 3/5/2003

Figure 5

The alternative processing mechanism, through the */certsrv* web enrollment page of the root CA, was also tested successfully.

No difficulties were encountered in either procedure. It is a best practice to install web services before configuring the Certificate Authority. The message in Figure 6 illustrates the warning produced if it has not been pre-installed; in addition some aspects of subsequent web administration work better if the correct order of installation is followed.



le Setup configures the components. This may take
, depending on the components selected.

guration of Certificate Services...

Microsoft Certificate Services

Internet Information Services (IIS) is not installed on this computer. Certificate Services Web Enrollment Support will be unavailable until IIS is installed.

Figure 6

The issuing enterprise CA's were established in two different domains in the forest, with a goal of testing cross-domain operations of the published certificates. Appropriate online distribution points were chosen for the Certificate Relocation List (CRL) and Authority Info Access (AIA) for the root server, since it was offline and vaulted for security.

Three distribution points were chosen in each case for the enterprise certificate authorities. One was an ldap pointer to the information published into Active Directory by the *certutil.exe -dspublish* command; two were file pointers to the subordinate CA's for clients not a part of the domain. In addition, the renewal key length and validity period of the certificate were pre-configured in the CAPolicy.inf file. The publishing mechanism is similar to that using the *dsstore.exe* command for directory publishing in the Windows 2000 environment.[12]

Following setup of the issuing subordinate enterprise servers, email operations were tested in the forest. Since the .NET schema extensions had not yet been

[12] How to Install a Windows 2000 Certificate Services Offline Root Certificate Authority, Tech Net, Microsoft Corporation, undated. http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q271386&

applied to the forest, it was necessary for each client needing a certificate to manually request one from the authority, typically by using the web-based certsrv request. Email operations using encrypted and digitally signed messages were possible among forest domains for those users who had acquired certificates.

By applying the .NET extensions to the Windows 2000 Active Directory, auto-enrollment of the domain users running Windows XP workstation clients becomes possible. Initially the *adprep.exe /forestprep* command returned error status relating to missing / incomplete information needed to process the schema upgrades.

The problem was eventually solved by running the *winnt32.exe /checkupgradeonly* command first. In determining whether the controller is a candidate for an upgrade to .NET build level, this command creates the proper file and directory structure for a subsequent *adprep* command to succeed. This will doubtless be fixed in the release level software.

Web

Secure web services were operated using both one way and two way SSL authentication. Client certificates were automatically assigned, and user identity was proved to the website without logon / password prompting.

Smart Cards

Smart card logon functions through a thin client interface for the first time in this environment. USB devices from Aladdin were particularly effective in implementing a logon in the Terminal Services environment. These cards also had no difficulty in processing autoenrollment tasks using certificate templates and the Aladdin eToken cryptographic service provider (CSP).

The user is presented with a “bubble” notice at the bottom of the screen, followed by a popup message and advice to insert the smart card device for auto-enrollment. This requires previous installation of the device-specific software (in this case the Aladdin Real Time Environment) on the workstation, and the presence of a USB port or smart card reader suitable for the token device.

The state of the Aladdin drivers and utilities was particularly mature.[13] Products from two other smart card vendors were unable to complete the authentication, probably because of the pre-release nature of the software and the necessarily beta condition of the associated client drivers software.

[13] Aladdin eToken Enterprise product description: http://www.ealaddin.com/etoken/default.asp?cf=tl

MultiForest Environment

An environment consisting of two independent forests was established to test cross-certification between organizations and inter-organizational traffic. In this scheme of extending the scope of validity of certificates, two Enterprise root CA's were certified, each with the other.

In the resulting environment, which Microsoft terms "Qualified Subordination," it is possible for certificates from either hierarchy to be verified back to the local root, providing great efficiency in business-to-business situations in which this relationship is possible.

Conclusions

The enhancements in the Windows PKI mechanisms present in XP Professional client and .NET Advanced Server are very significant. Both the underlying technical advances such as true enterprise cross-certification and the ease-of-use features like automatic user certificate acquisition will make the product mainstream for business use.

The total cost of ownership is further enhanced since the Certificate Authority is included, like DNS or DHCP, in the basic server license. Additionally, the certificate database is maintained locally on the server or integrated into Active Directory for availability and resiliency. There is no need to license expensive third-party software comprising a CA and database server in order to have a fully-featured certificate hierarchy.

When combined with PKI initiatives currently underway, such as the recent issuance of the one millionth digital certificate by the U.S. Department of Defense, this enhanced certificate capability will advance both the usage and acceptance of this important technology.

References:

Aladdin eToken Enterprise product description:
http://www.ealaddin.com/etoken/default.asp?cf=tl

Certificate Server Updated to Use non-Microsoft X.509 Certificate Authorities, TechNet, Microsoft Corporation (undated)
http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q242031&

Cross, David B., Certificate Autoenrollment in Windows XP. Microsoft Corporation, January 2002
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol

/winxppro/maintain/certenrl.asp

Cross, David, PKI Enhancements in Windows XP Professional and Windows .NET Server, Microsoft Corporation, July 2001. http://www.microsoft.com/windowsxp/pro/techinfo/planning/pkiwinxp/PKIEnhancements.doc

Cross, David B., Best Practices for Implementing a Windows .NET PKI. Microsoft Corporation, April, 2002.

Freeman, Dr. Trevor, Windows XP PKI Client Windows .NET Server Public Key Infrastructure, Microsoft Corporation. http://www.microsoft.com/usa/presentations/Freeman_SecuritySummitWest.ppt

How to Use the Directory Services Store Tool to Add a Non-Windows 2000 Certification Authority (CA) to the PKI in Windows 2000, TechNet, Microsoft Corporation (undated) http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q313197&

How to Install a Windows 2000 Certificate Services Offline Root Certificate Authority, Tech Net, Microsoft Corporation, undated. http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q271386&

Komar, Brian, Planning and Implementing Qualified Subordination using Windows .NET Enterprise Server, Microsoft Corporation, March 2002.

Komar, Brian, Troubleshooting Certificate Status and Revocation, Microsoft Corporation, October 2001. http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/winxppro/support/tshtcrl.asp

What's New in Microsoft Windows CE .NET. Microsoft Corporation, July 31, 2002 http://www.microsoft.com/windows/embedded/ce.net/evaluation/whatsnew/default.asp