PKI Support in Windows 2000

Secorvo White Paper

Version 1.1
12. February 2002

Holger Mack

Secorvo Security Consulting GmbH
Albert-Nestler-Straße 9
D-76131 Karlsruhe

Tel. +49 721 6105-500
Fax +49 721 6105-455

E-Mail info@secorvo.de
Internet http://www.secorvo.de

Table of contents

Acronyms

ADS	Active Directory Service
ADSI	Active Directory Service Interface
AIA	Authority Information Access
ANSI	American National Standard Institute
CA	Certification Authority
CDP	Certificate Distribution Point
COM	Common Object Model
CRL	Certificate Revocation List
CryptoAPI	Cryptographic Application Programming Interface
CSP	Cryptographic Service Provider
CTL	Certificate Trust Lists
DB	Database
DLL	Dynamic Link Libraries
DNS	Domain Name Service
EFS	Encrypting File System
HSM	Hardware Security Module
IE	Internet Explorer
IETF	Internet Engineering Task Force
IIS	Internet Information Server
IPSec	Internet Protocol Security
ISO	International Standardisation Organisation
IT	Information Technology
LDAP	Lightweight Directory Access Protocol
MMC	Mananagement Console
NT	New Technology
OCSP	Online Certificate Status Protocol
PC/SC	Personal Computer/Smart Card
PKCS	Public Key Cryptography Standard
PKI	Public Key Infrastructure
PKIX	X.509-based Public Key Infrastructure
RFC	Request for Comment
SDK	Software Development Kit

SSL	Secure Socket Layer
TLS	Transport Layer Security
XP	Experience

Acknowledgement

We would like to thank the European Central Bank (ECB) for providing the English translation of this White Paper.

1 Abstract

Microsoft has made public key infrastructure (PKI) functionality a core component of its security architecture in Windows 2000. While this is undoubtedly an important step, the focus of the PKI functionality is clearly on integrated support in a Microsoft environment. Microsoft also provides some neat solutions (e.g. for distributing trustworthy CA certificates) to issues that can often only be resolved with a great deal of difficulty in other environments.

Microsoft follows a similar path to that of Lotus Notes a few years ago, with the difference that Microsoft's PKI is more open and in greater conformity with standards than the solution offered by Lotus Notes. This support of standards makes it possible to use the functionality outside a pure Windows environment or to integrate it with other environments and applications. However, it is essential to check closely that all the requirements are met to ensure this kind of support.

From a critical point of view, it has to be said that the available PKI functionality is not yet fully advanced. The development of other CA products has demonstrated that it can take some time to produce an advanced PKI product. The main problems are the lack of flexibility and functionality, particularly when working outside a Windows 2000 environment.

With Windows XP it is evident that PKI is an important module of Microsoft's .NET strategy. Great progress appears to have been made in terms of PKI functionality in Windows XP or .NET: some important functions that were missing from Windows 2000 have been implemented here. However, it remains to be seen whether these functions will meet expectations or whether Microsoft will expand the Certificate Service in such a way that the functions for issuing certificates outside the Windows environment will also be improved.

Furthermore, certain functions, such as the automatic loading of new trustworthy certificates in the background, need to be examined more closely to ensure that there is no risk that they could enable a hacker to use the PKI functionality to prepare more extensive attacks. Since PKI is to play a significant role in the future Microsoft strategy (e.g. .NET architecture, Passport service), the security of the PKI functionality is an important issue.

In sum, it can be said that the PKI functionality in Windows 2000 offers all the basic functions of a PKI. Windows 2000, like the other PKI products on the market, has both strengths and weaknesses. It is therefore to be recommended that Microsoft be considered in the product selection process. If the basic conditions are right (e.g. the operational applications are primarily implemented in a Windows environment), Microsoft is a serious alternative to other specialised products. Other products generally have the advantage that they can be implemented more flexibly even in heterogeneous environments. Since heterogeneous environments dominate in practice, and PKI-based security functions are to be used not only in internal networks, but primarily with external partners and customers, a combination of a Windows 2000 PKI and other products or certification service providers may well be worthwhile.

2 Introduction

With Windows 2000 Microsoft has introduced a large number of new features by comparison with the previous version, Windows NT 4.0. Microsoft has clearly made particular efforts in the area of security in order to shake off its bad reputation in this field: many of the security functions in Windows 2000 have been revised, enhanced or provided with completely new functionality.

The integration of public key technology into the operating system is particularly important in this respect. Public key technology is used in Windows 2000 to improve existing security mechanisms (e.g. the integration of certificate-based authentication), but also to support new security mechanisms directly in Windows 2000 (e.g. file encryption, IPSec).

The PKI support in Windows 2000 has received a lot of attention, primarily because many organisations are currently concerned with the implementation of PKI solutions. The question of if and how Microsoft's Windows 2000 fits into the PKI strategy of an organisation is raised ever more frequently in PKI projects.

There are two main reasons for this. First, Microsoft provides certain functionality "free of charge" as part of the operating system which would have to be bought separately and at great cost from other specialised manufacturers of PKI software. Second, on account of its world-wide distribution and leading market position, Microsoft Windows always plays a significant role when IT projects are to be implemented. Understandably, when implementing an IT project, most companies endeavour to ensure that it is compatible with the technology supported by Microsoft, either because Windows 2000 is already used throughout the company or a migration is planned, or because they wish to avoid technical problems when working with companies that use Microsoft.

Against this background, it is important to establish what the Windows 2000 PKI functionality really has to offer, and to what extent the PKI functionality should be included in the planning of PKI or IT projects. The technical details of a PKI product and the effects of its technology on a PKI strategy are not always immediately apparent or to be found in the documentation. Particularly in the PKI field it has been demonstrated that if a function is described in the documentation of two different manufacturers (e.g. support of standards), it does not necessarily follow that the two products can work together in practice. The differences are often only very small, but they can have a considerable effect on the implementation. The respective conditions must also always be taken into account (e.g. technical environment, special requirements, security requirements, etc.).

This White Paper describes and analyses the functionality of Microsoft's Certificate Service, the certification authority (CA) component of Windows 2000. It examines how Microsoft's Certificate Service can assist in the construction of a PKI and which secondary issues are to be considered. It aims to provide the reader with a better understanding of Microsoft's Certificate Service so that he or she can judge its suitability.

However, Microsoft's PKI support does not just cover the CA functionality of the Certificate Service, but also includes client functionality, such as the certificate management that is integrated into the operating system.

3 PKI support in Windows 2000

The PKI support in Windows 2000 extends to many areas of the operating system. The most important components are illustrated in figure 1. The Certificate Service plays a central role, taking on the function of a certification authority (CA), i.e. issuing and revoking certificates.

As in any Windows 2000 domain, the integrated Active Directory Service (ADS) plays an important role in the Windows 2000 PKI. Depending on the mode of the CA (see below), the Active Directory is used for publishing certificates and certificate revocation lists, registering participants and centrally controlling the PKI functionality on the clients in a Windows 2000 domain.

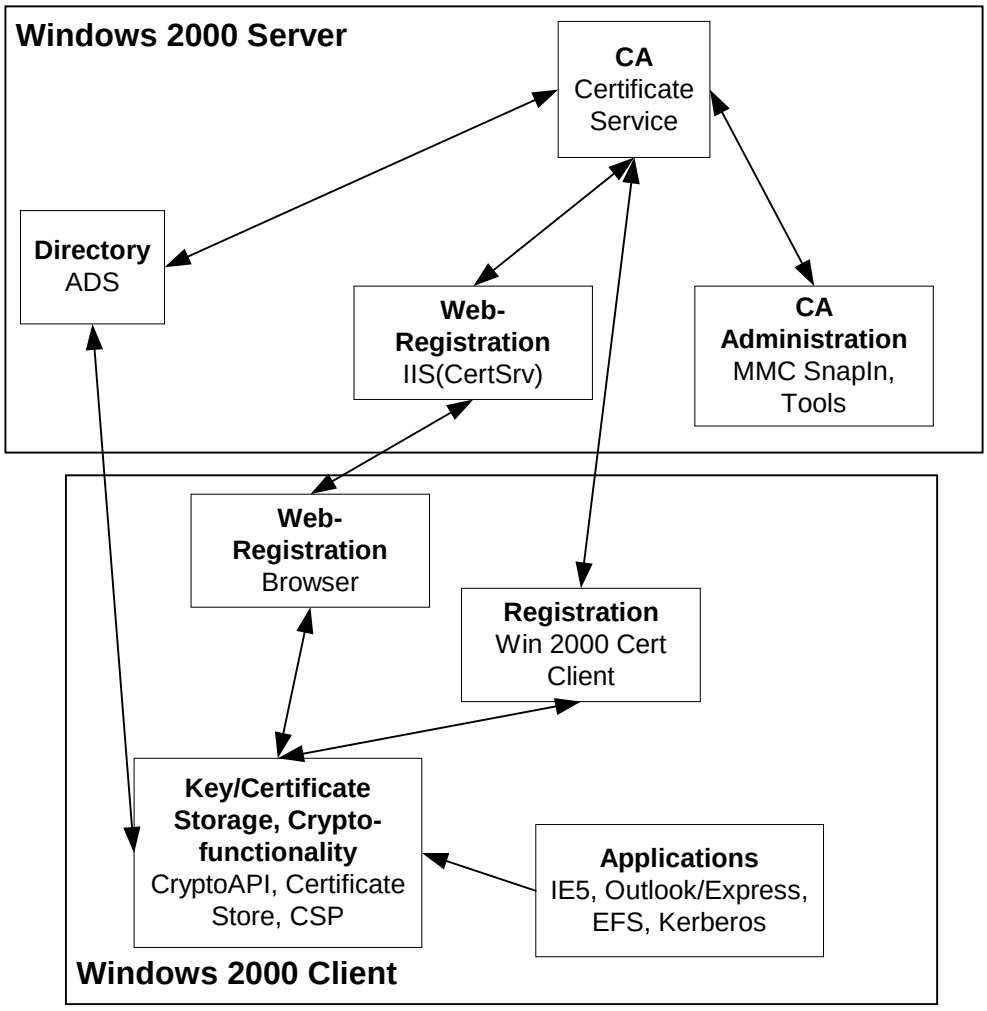

Figure 1: Komponenten Windows 2000 PKI

Functions for managing certificates, certificate revocation lists (CRL) and keys as well as for validating certificates and certificate chains are integrated into the operating system for the certificate user. Using the appropriate interfaces (e.g. CryptoAPI) these functions can be integrated into applications by developers. This functionality makes it possible to provide users with PKI functionality in a uniform manner. Parts of the user's certificate management can be administered and predefined from a central point in a Windows 2000 domain. Some Microsoft applications, such as Outlook and Internet Explorer, already use this functionality.

Cryptographic service providers (CSP) – libraries that enable the operating system to access cryptographic operations via a defined interface – can also be used to enhance the standard functionality provided in Windows 2000, e.g. for supporting cryptographic hardware.

The sections below focus primarily on the CA component of Windows 2000, the Certificate Service. This component is in competition with other products on the market, from companies such as Entrust or Baltimore who specialise in CA components.

4 Architecture

The Certificate Service is made up of a large number of modules which perform different certificate management tasks. Figure 2 illustrates the architecture of the Certificate Service with related components.

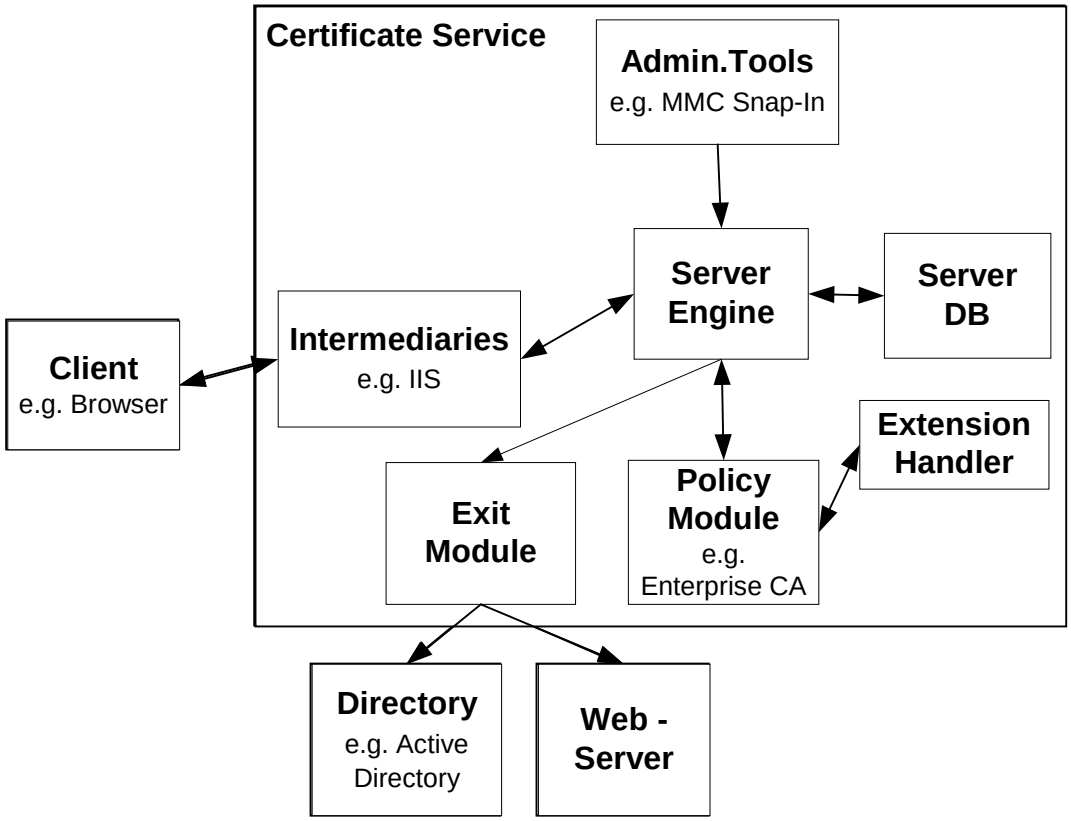

Figure 2: Windows 2000 Certificate Service architecture

The server engine is the central component of this architecture. It is responsible for issuing certificates and certificate revocation lists. Only limited functionality is integrated into the server engine itself (that is, the actual certificate generation). A significant proportion of the PKI functionality is implemented in the different modules used by the server engine:

- *Policy module*: Functions such as the validation and authorisation of a certificate request, naming and the content of a certificate (use of attributes and their values) are implemented here.
- *Exit module*: Functions for publishing certificate revocation lists and certificates, e.g. in a directory service, are implemented here.

- *Extension Handler:* Certificate extensions for use in certificates are defined here.
- *Intermediaries*: These accept certificate requests from applications and pass them on to the server engine.

All these modules are linked to one another via defined interfaces, but are otherwise set up independently of one another in the form of Dynamic Link Libraries (DLL). As a result, they can be adjusted and exchanged. The modules communicate with the server engine mainly by means of common object model (COM) interfaces.

The modular set-up provides a high level of flexibility and offers the possibility of creating individual solutions, but in practice this involves a certain amount of work. The main reason for this is that individual modules can only be exchanged in their entirety, and the modules for implementing modifications have to be completely reprogrammed. The relevant functions are contained in the Microsoft Software Development Kit (SDK) [MSDN_01] and can be used in the programming languages C++ and Visual Basic. A number of modules are already contained in the standard version of Windows 2000. In certain cases, it is explicitly recommended that these are not exchanged (e.g. policy module for Enterprise CA).

The two policy modules Enterprise CA and Stand-Alone CA, which are contained in the standard Microsoft package, are of greatest significance for the PKI functionality. The decision as to which of these two policy modules is to be used is made at the point of installation. The main criterion in the decision is the purpose of the CA:

- The *Enterprise CA* is very highly integrated in the Windows 2000 environment including Active Directory and requires a Windows 2000 domain and Active Directory. The Enterprise CA can only to be used for the certification of users and computers within a domain.
- The *Stand-Alone CA*, by contrast, is largely independent of other components (such as Active Directory) and its operability is not dependent on a Windows 2000 domain. Certification is not dependent on domain accounts.

The sections below examine the different aspects of the two policy modules.

5 Criteria for comparison

In practice, the assessment of a PKI product is very strongly influenced by important conditions: the type of use, the applications to be supported, the technical environment and the required level of security are just some of the criteria that need to be considered in an assessment of this nature.

The examination in this chapter is not based on a specific scenario; rather, it attempts to be as general as possible. The functionality of the Windows 2000 PKI should be judged within this framework on the basis of the most important criteria for a CA product. These criteria are:

- Trust models
- Support of standards
- Registration and key/certificate distribution
- Flexibility
- Administration
- Directory support (publication of certificates and certificate revocation lists)

The following sections describe these criteria in detail.

5.1 Trust model

In addition to the option of operating a Windows 2000 CA independently, a hierarchical trust model (e.g. integration in or construction of a PKI hierarchy) is also supported. CA products from other manufacturers can be combined with Windows 2000 CAs as required. For example, a Windows 2000 CA can function under an external CA as a subordinate CA, but Windows 2000 can also issue certificates for subordinate CAs outside the Windows 2000 environment. The number of levels in the hierarchy is not limited. Certifications in a hierarchy are requested and processed via the standard formats PKCS#10 [PKCS_10] and PKCS#7 [PKCS_7], which are supported by virtually all manufacturers and providers.

Cross-certification [HAM_01] is not supported by Windows 2000 as a second method of constructing a trust model, neither through the CA nor through the client components. However, it is expected to be supported in Windows XP (see also Chapter 7).

In addition to the hierarchical model, Windows 2000 offers a further possibility for achieving trust with other CAs, namely by using Certificate Trust Lists (CTL). A CTL is a signed list of trusted CA certificates. It works on a similar principle as a certificate revocation list, with the difference that a CTL does not contain revoked certificates, but rather CA certificates from trusted CAs. In a Windows 2000 environment, this list is signed by a trustworthy person (such as a PKI administrator) with a key/certificate from the own hierarchy. The list can be distributed to the clients in a domain and deleted using the Active Directory and the Windows 2000 policy mechanism.

In this way, CAs can be declared or defined, from a central place, as trusted within a domain. Programs using the Windows 2000 client functionality will automatically recognise certificates from CAs in a CTL as trustworthy.

The (proprietary) format of CTLs also offers two possibilities for limiting the trust in the CA certificates contained in the list:

- Like CRLs and certificates, CTLs have a limited lifetime, i.e. a validity period can be defined.
- The use of the CA certificates can be limited. It is possible to specify the use (e.g. object signing) for which the CAs in the CTL are trusted. The certificates will thus be recognised in the client as trustworthy for these uses only.

The CTL mechanism is a proprietary solution from Microsoft and does not correspond to any standards. For this reason, CTLs are currently only supported by Microsoft. Although CTLs can be exported and distributed as a file, they can only be evaluated and used in a Windows 2000 environment.

A problem is represented by the fact that there is currently no provision for revoking CTLs in Windows 2000. When a certificate is no longer to be seen as trusted, the CTL must be deleted using Windows 2000 mechanisms and a new CTL issued and distributed. If CTLs are distributed beyond a centrally administrated Windows 2000 environment, the absence of a revoking option is critical.

5.2 Support of standards

Just as Windows is generally opening up to established IETF, ISO and ANSI standards in many areas (e.g. DNS), the PKI functionality in Windows 2000 is now also based to a large extent on international standards. The most important of these are:

- X.509v3 [X509_97] and PKIX RFC 2459 [RFC2459] for certificate and certificate revocation list formats
- PKCS for signature formats [PKCS_1] and exchange formats [PKCS_7], [PKCS_10], [PKCS_12]
- LDAPv3 [RFC2251]
- PC/SC for smart card integration [PC/SC_97]

5.2.1 Certificates

Microsoft's certificate formats are adapted to X.509v3 and the certificate and certificate revocation list profiles defined in RFC2459. In principle, the architecture of the Certificate Service permits flexible certificate content. However, these options are extremely limited with the policy modules contained in the standard Windows 2000 installation. The content and layout of the certificates are predefined using certificate templates and can only be adapted to a very limited extent.[1] There is a range of application-specific certificate templates covering most of the standard applications. These templates are managed in the Active Directory, and they cannot currently be adapted or redefined.

In all but a few details, the certificate contents defined in the certificate templates correspond to the formats defined in important standards. However, these details can play an important role in practice. A distinction should be made between the following two cases:

- The Microsoft Certificate Service is used to issue certificates for non-Windows products.

[1] The certificate templates do not just contain specifications regarding the content of the certificates, but also information that is necessary for their issuance (checks, etc.).

- A CA product from another vendor or service provider is to be used to issue certificates for Windows 2000 applications.

The cases described below have different effects depending on which of these scenarios is relevant.

In addition to the certificate extensions defined in the standards, Microsoft has defined its own private extensions, which are primarily necessary for original Microsoft applications (e.g. Encrypting File System (EFS)). The standard X.509 explicitly permits the definition of own extensions of this type, but there may be problems in practice if applications cannot interpret these extensions or if products from third parties do not support the functionality related to an extension. However, many PKI manufacturers have now built support for Microsoft extensions into their current products, so that these products can also be used to issue certificates with the extensions defined by Microsoft, for example for certain Windows 2000 applications (such as EFS). Since none of these extensions are flagged as “critical”, other client products should, according to the standard, at worst ignore them. In practice, however, there are sometimes problems such as program crashes. In the case of doubt, therefore, the usability of certificates with Microsoft-specific extensions in non-Microsoft products should be tested.

Furthermore, in its predefined certificates Microsoft does not always follow the recommendations in the standards with regard to the marking of certificate extensions as “critical”. When extensions are used in predefined certificate templates, they are never marked as “critical”. This is also the case for extensions such as key usage, which the standards recommend be flagged as “critical”.[2]

However, it should be noted that, in principle, the Certificate Service already supports the issuance of critical extensions; however, this functionality is not used in the predefined certificate formats.

A third problem that the certificate formats used by Microsoft can cause is that Microsoft applications have strict requirements with regard to the presence and precise format of certain certificate extensions (e.g. Certificate Distribution Points (CDP)). This is of particular significance in the case of certificate validity checks. If they are not available as expected by a Microsoft product, the client functionality may be restricted (when searching and importing certificate revocation lists, for example).

Overall, within the framework of different (not complete) tests, other manufacturers’ products were generally able to import certificates issued by Windows 2000, and it was also possible to use certificates issued by CA products from other manufacturers in Windows 2000. However, caution is recommended to ensure that the certificate details will not cause any functional or security-related restrictions. This may be the case particularly if existing infrastructures are to work together with Windows 2000. The well-known case of a fake Verisign certificate for Microsoft [MAC1_00] clearly demonstrated the extent of the problems that can occur in this context.

5.2.2 Certificate revocation lists

The standard Windows 2000 installation supports Certificate Revocation Lists (CRL) as a mechanism for revoking certificates. Certificate revocation lists are used according to the X.509 standard [X509_97], which meets with the standard generally in current use. These

[2] Unfortunately, this is a very common practice used by some suppliers to avoid interoperability problems at the expense of security functionality.

are always complete certificate revocation lists, i.e. the CA creates a certificate revocation list containing all the revoked certificates by a CA which have not yet expired. More extensive mechanisms, which are provided for in the standard and now supported by many CA products, such as the differentiation between CRLs and ARLs (Authority Revocation Lists), Delta CRLs or OCSP [FOX_99], are not supported in Windows 2000, either on the CA side or on the client side.

It is important to note that Windows 2000 clients can only find certificate revocation lists in directories if the CDP extension is contained in the certificate with the relevant information in the correct format. If this extension is not contained (which is primarily the case with older certificates), Windows 2000 can only run checks against locally imported certificate revocation lists.[3]

5.2.3 Exchange formats

In addition to the standards for certificates and certificate revocation lists described above, Windows 2000 supports a number of standards from the PKCS series for the exchange of certificate requests, keys and certificates. The supported standards are:

- PKCS#10 for certificate requests [PKCS_10]
- PKCS#7 for exchanging certificates and certificate chains [PKCS_7]
- PKCS#12 for exchanging private keys [PKCS_12].

These standards are supported by virtually all other PKI products.

5.3 Directory support

The Certificate Service only provides direct directory support if the Enterprise Policy and related exit module are used. If this is the case, certificates and certificate revocation lists are automatically published in the Active Directory (via ADSI). Automatic publication in other directories via LDAP is not supported. Direct integration with a directory is not supported in the stand-alone mode.

Active Directory supports LDAPv3 in such a way that applications can access Active Directory and the certificates and certificate revocation lists via LDAPv3. Applications from other manufacturers can also access certificates and certificate revocation lists, but only if the clients support the Certificate Distribution Point (CDP) and Authority Information Access (AIA) extensions for finding certificate revocation lists or CA certificates in the Active Directory.

5.4 Flexibility

On account of the various modules, the architecture of the Certificate Service offers a relatively high degree of flexibility in principle. As described in chapter 4, however, this flexibility can only be exploited in many areas if a considerable amount of programming is carried out. The possibilities for configuration are limited with the standard policy modules for the Enterprise and Stand-Alone CAs. The options for configuring the PKI functionality are

[3] However, checks against locally imported certificate revocation lists are not supported in all cases or by all applications (see [MAC1_00]).

limited to a small number of parameters (CDPs), which can be adjusted accordingly. The attributes for the creation of the distinguished name of CAs and users are also pre-set.

On the client side, flexibility is provided by exchangeable CSPs, which mainly enable the adjustment of the cryptographic functions and key storage.

5.5 Registration

The procedures for registering users and computers differ greatly depending on which policy module is used. The two modules are therefore discussed separately below.

5.5.1 Enterprise CA

In an Enterprise CA, a user or computer is registered when an account is created in the Windows 2000 domain. If a user is registered here, he or she can use the Certification Manager in the Management Console (MMC) or the CA's registration website (assisted by the Internet Information Servers (IIS)) to request a certificate. A website is provided by Microsoft for this purpose. The user is authenticated by means of his or her Windows 2000 account using the information stored in the Active Directories, and the certificate is then issued automatically. There is no limit to the number of certificates with which a user can be issued in this way, but the types of certificate that a user can request can be limited and controlled via the access rights to the certificate templates in the Active Directory. Access to the certificate website can also be controlled using the standard IIS mechanisms (password, SSL/TLS).

This form of enrollment does not take place for the Encrypting File System (EFS). The first time a user tries to encrypt a file, a corresponding key will be generated and signed by the Enterprise CA. This happens automatically and is not seen by the user.

The Enterprise CA also offers the option of using auto-enrollment for issuing certificates for computers. Once this has been configured (via group policies), certificates are automatically issued for computers when the computer registers in the domain. If auto-enrollment is not used, an administrator must carry out the certification manually for each separate computer.

Another special case is issuing of certificates for the Smart Card login supported in Windows 2000. In the standard system, these certificates cannot be requested directly by the user. The request must be made by a special administrator (such as a PKI officer), that is, an administrator with a special certificate, who then passes on the Smart Card to the user. In the standard system, the transaction is carried out via an appropriate website.

Looking at the Enterprise CA from a PKI perspective, the registration points are the points at which accounts are set up for users or computers. The security is thus heavily dependent on the process of setting up accounts in a domain. It may therefore be necessary to check whether this process satisfies the security requirements set for the certificates (or the related applications).

5.5.2 Stand-Alone CA

With the Stand-Alone CA there is no integration in a domain, and so certificates can only be requested via the IIS website. In the standard set-up the certificate requests are then passed onto the CA, where an operator explicitly approves (or rejects) the request. However, it is also possible to configure the system to automatically issue all incoming requests. However, apart from the very limited details contained in the certificate request, the administrator does not have any additional information with which to validate the request.

As with the Enterprise CA, access to the websites can be protected using the standard IIS protocols and mechanisms (e.g. TLS).

5.6 Administration

Along with the pure PKI functionality, the administration of a PKI plays an important role in practice. It is crucial to the work required to operate the PKI, and therefore to both the cost and the security of the PKI. As long as no additional special data or processes are necessary, the administrative investment required for the Enterprise CA can be kept relatively low as a result of its integration with the operating system and the use of existing information from the Active Directory.

Microsoft offers a range of tools for managing the PKI. The most important graphical tool is a snap-in for the MMC, which can be used to carry out the most fundamental CA functions, such as revoking certificates (see figure 3).

The visual set-up is like that of the file manager and is therefore relatively clear and simple. However, it can soon become confusing if there are a large number of certificates.

In addition to issuing and revoking certificates, this tool can also be used to perform a number of additional administrative functions, such as starting and stopping the Certificate Service, renewing a CA certificate[4] and saving and resetting the CA database.

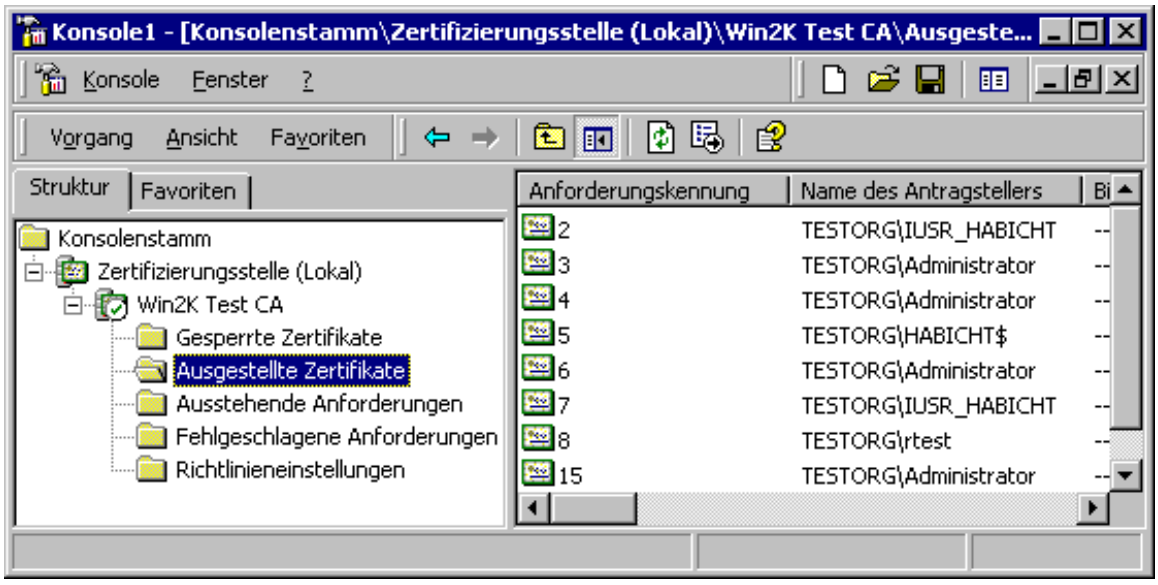

Figure 3: MMC snap in administration certification authority

In addition to this graphical interface, there are a number of very useful command line tools that can be used for administration. The two most important are *certutil.exe* and *dsstore.exe*.

- In principle, Certutil provides the most important functions of the graphical interface as well as some significant additional functions on a command line interface (see appendix).
- DSStore provides functions that are important for the interaction of the Active Directory and Enterprise CA. It is particularly helpful in solving PKI and Active Directory problems.

4 It is possible to renew the certificate for the existing keys or to generate a new keys.

Unlike Certutil, which is supplied with a Windows 2000 server, DSStore is only available as part of the Server Resource Kit (see appendix).

Since the Enterprise CA is highly integrated into the Active Directory, certain LDAP and Active Directory tools can be very useful for problem-solving. Some of these are provided with Windows 2000 and others are contained in the Resource Kit.

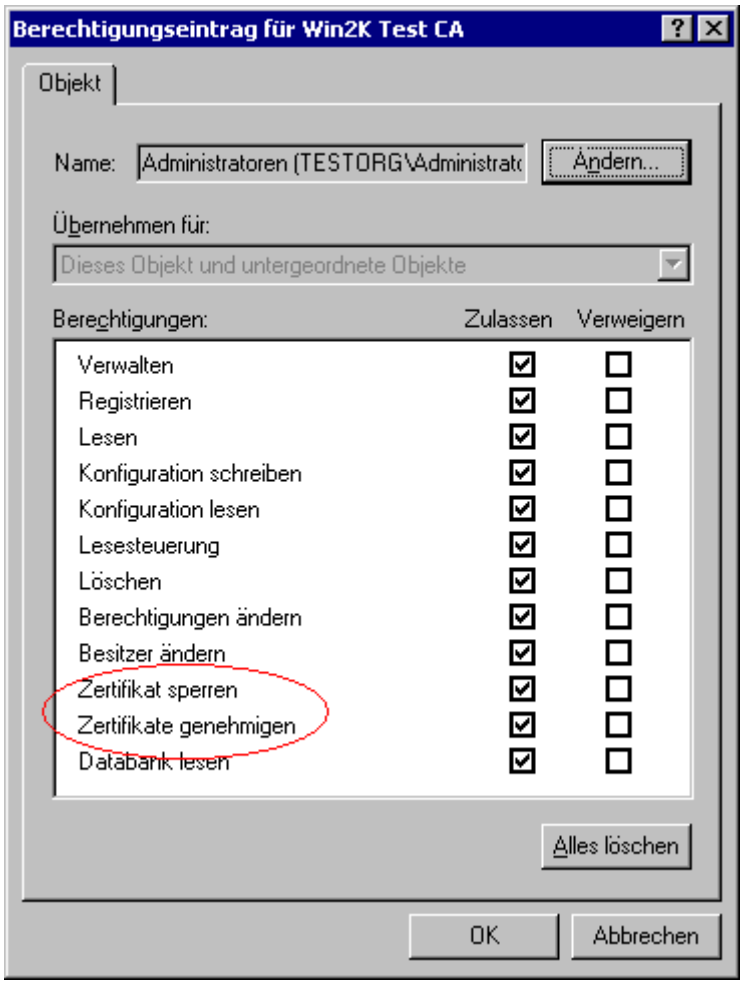

Figure 4: Certificate Service rights management

To control access to the CA functionality, Microsoft uses the access rights management model used in Windows 2000. The Certificate Service and certain important components are – as everything in a Windows 2000 environment – objects for which special access rights can be assigned. Access rights to the CA can be restricted (see figure 4) using the special authorisations for the CA object. There is also the option of adjusting the rights of the CA or the user by restricting access to the certificate template (part of the Group Policies in ADS). This method provides the option of configuring which types of certificate can be issued by which CA, and who can request which certificate types. An even more refined adjustment can be made by assigning rights to the various enrollment controls which are necessary for requesting certificates.

5.7 Special security measures (CA)

In addition to the standard measures for protecting Windows 2000 servers, other special measures are necessary to safeguard the CA services. The standard access rights granted for some components are often too generous. More details on secure configuration can be found in [NSA_00].

If particular requirements need to be met (such as the 4-eye-principle), this can only be done through organisational measures (e.g. split passwords) or additional functions of third party products (e.g. Hardware Security Modules).

However, the close integration of the operating system and the CA makes it virtually impossible to prevent administrators from also having far-reaching rights for the CA functionality. Therefore a clear segregation of duties cannot be reproduced in a Windows 2000 PKI.

6 Other features

This chapter describes some further characteristics of the Windows 2000 PKI that were not mentioned under any of the above topics.

6.1 Validity model

The Certificate Service issues nested validity periods for the certificates [BER_01]. This means that a CA only issues certificates which have a validity period that falls completely within the validity period of the CA certificate [MS_CS_00]. In practice this means that a CA whose certificate is only valid for a further six months, for example, can only issue certificates with a maximum validity period of six months. This fact must be taken into account when planning an update of the CA certificates. Older client versions Microsoft applications (such as Internet Explorer) checked for these nested validity periods and rejected certificates in the case of violations. However, more recent versions no longer appear to run this check, so Microsoft applications no longer enforce this validity model.

6.2 Integration with other products

Other PKI-component manufacturers have reacted quickly and integrated support for Windows 2000 into their products. The form of this integration ranges from the simple option of implementing the products to run on the Windows 2000 operating system to a more far-reaching integration into the functions of the operating system. Particularly the large manufacturers of CA products endeavour to have their products work with Windows 2000 in such a way that the customer sees the enhancement provided by these products vis-à-vis Windows 2000.

The support and integration differ from product to product. In principle, there are a number of different strategies and starting points for integration. The most important of these are:

- *Active Directory support*: Products can write certificates and certificate revocation lists directly to the Active Directory.
- *Certificate extension support*: The option of issuing certificates with the special Microsoft extensions and the certificate extensions in the form that Microsoft expects.
- *Certificate management*: Provision of a user certificate management via the CryptoAPI/CSP interface.

- *Integration into the PKI hierarchy:* The option of integrating other products and Windows 2000 CAs within one hierarchical structure.

If, therefore, a manufacturer claims to support Windows 2000, it is advisable to examine the form of that support in detail.

One frequently asked question is whether the Microsoft Certificate Service can be replaced completely by another product. For many Microsoft applications (such as Outlook) this is, in principle, possible, i.e. certificates from other CAs can also be used (e.g. via PKCS#12-Import). However, this does not allow the same degree of integration as the use of the Enterprise CA. The existence of an Enterprise CA is necessary for some applications, such as auto-enrollment. Such functions therefore cannot be supported if an external CA product alone is used.

6.3 Key management

Key pairs for users and computers are usually generated decentrally, that is, with the user. In Microsoft clients, the type and quality of key generation and storage therefore depends on the cryptographic service provider (CSP) used by the user. In the case of integrated CSPs, the standard Microsoft system allows keys to be generated and stored (only) in software. However, there are a number of manufacturers that provide the option of integrating CSPs with special characteristics, such as for generating and storing keys on smart cards or special hardware security modules (HSM).

A managed key recovery or key backup concept for storing secret participant keys is not currently supported. Such a concept would, for example, makes it possible to access encrypted data even if a key had been lost.

The automatic renewal of certificates is not currently integrated in Windows 2000 either. When a certificate expires, the user must apply for a new one, except in the case of auto-enrollment, where new certificates are automatically issued. There is a mechanism for extending CA certificates.

7 Windows XP

In autumn 2001 (client) respectively spring 2002 (server) Windows XP[5], the successor to Windows 2000, will come onto the market. This will build upon the technology introduced in Windows 2000. Microsoft will also provide a number of new features in the area of PKI support, but the basic architecture is the same as in Windows 2000. However, Microsoft has developed the individual components and added some new functionality.

The most significant new features in the PKI area are described briefly below:

- Both CA and client components will support cross-certification and Bridge CA scenarios.[6]
- Auto-enrollment and auto-update are supported for user certificates.

5 The server version will be called .NET Server.

6 This could not yet be tested in any more detail.

- Certificate templates can be adapted, meaning that a number of attributes can be configured, such as the certificate content, the policy for issuing certificates, validity, CSP used, etc.
- A key archive for the central storage of keys is integrated and can be configured via the certificate templates.
- Delta CRLs are supported.
- Defined roles for managing the CA will be introduced, such as CA administrator, operator, auditor, etc., with each role having appropriate rights. This division of roles is connected to Microsoft's efforts to achieve a common criteria certification for Windows XP.

The existing functionality was also enhanced in the applications; for example, EFS will allow several users to access one encrypted file. Windows XP will also support the validation of digital signatures on software packages to be installed.

With Windows XP Microsoft is also introducing a further new feature in the form of the "*Microsoft Root Certificate Program"*[MS_TN_01].[7] In the past browsers and operating systems were delivered with a preconfigured list of "trustworthy CAs". This list will now be replaced with a mechanism by which, in the background, the browser automatically uploads new "trustworthy" CA certificates from a Microsoft server and integrates them into the browser. This mechanism is based on the CTLs described above (see Chapter 6.1). In addition to the mechanism for distributing CA certificates, Microsoft is also changing the criteria to be met by a CA operator in order to be included in the *Root Certificate Program*. These criteria range from those relating to the appearance of the certificates (e.g. valid until at least 2010, use of CDP extensions) to certification under the "*WebTrust for Certification Authorities Program"* [WEB_01].

8 Strengths and weaknesses

The strengths of the Windows 2000 PKI clearly lie in the high level of integration in the Windows 2000 environment. This integration allows a large degree of transparency or automation in many places, with the result that tasks that are often complicated in connection with PKIs, such as registration, distribution of certificates, etc., can be performed relatively easily. The administrative effort of using an Enterprise CA is thus reduced to a minimum. On account of the integration of applications, the Enterprise CA is also suitable for improving security within a Windows 2000 domain.

However, the high level of integration also has disadvantages. The links to the operating system functionality may mean that changes, updates and the incorporation of new functions are more difficult because the interaction with other operating system functions must be taken into account.

Very high security criteria can only be met if considerable efforts are made.

The Stand-Alone CA is really only suitable for issuing a small number of certificates (e.g. for SSL servers or as a root CA) or for experimenting with a PKI within the framework of limited pilot tests. When certificates are issued for a large number of participants, the inadequate management options and missing functionality (such as directory integration) are of great relevance.

---

[7] Appropriate updates will also be provided for older Windows versions (NT, ME, 98, 95).

One of the greatest shortcomings is the generally limited functionality and the lack of flexibility of the current implementation. The Windows 2000 CA demonstrates weaknesses particularly when required to operate outside a Windows 2000 environment. In this regard, it clearly lags behind the other products on the market.

Provided that Microsoft's default settings are suitable for a "standard" IT environment, the lack of flexibility is unlikely to pose a problem. In the case of a more varied solution, however, problems certainly can occur. Microsoft has announced its intention to develop a number of future improvements that should rectify these weaknesses.

One of the main recurring arguments in favour of the Certificate Service is the price. The Certificate Service comes free of charge with every Windows 2000 server version (in Windows XP the full range of functions is, however, only available with the Advanced Server edition). By contrast, CA products from other manufacturers create high additional costs or have licensed models which are dependent on the number of certificates issued. There is undeniably a significant difference in price. Depending on the type of PKI and its use, however, the cost of purchase tends to represent a very small proportion of the overall cost of setting up and operating a PKI. It is therefore important to consider to what extent the required concept can be fulfilled using a Windows 2000 PKI, and how much more work the latter will require by comparison with other products. In many cases, the better manageability of other products can certainly offset the higher purchase costs.

9 Bibliography

[BER_01] Bertsch, Andreas, *Digitale Signaturen*, Springer, 2001

[FOX_99] Fox, Dirk: *Zum Problem der Gültigkeitsprüfung von Schlüsselzertifikaten*. Tagungsband 6. Deutscher IT-Sicherheitskongreß des BSI 1999, SecuMedia Verlag, Ingelheim 1999, S. 215-230.

[HAM_01] Hammer, Volker, Cross-Zertifikate verbinden, DuD 2/2001, Verlag Vieweg

[MAC1_00] Mack, Holger: *Sperren von Zertifikaten in der Praxis – eine Fallanalyse*, DuD 8/2001, Verlag Vieweg,

[MSDN_01] MSDN Library, *Platform Software Developement Kit*, 2001, Microsoft Corporation www.msdn.microsoft.com

[MS_CS_00] *Windows 2000 Certificate Service*, Microsoft Corporation, 2000

[MS_TN_01] Microsoft TechNet, *Microsoft Root Certificate Program,* Microsoft Corporation, 2001

[NSA_00] S.Christman, *Guide to the Secure Configuration and Administration of Microsoft 2000 Certificate Services*, National Security Agency, 2000

[PC/SC_97] *Interoperability Specification for ICCs and Personal Computer Systems, Part 1. Introduction and Architecture Overview,* PC/SC Workgroup, 1997

[PKCS_1] *PKCS #1: RSA Encryption Standard*, v1.5, 1993, RSA Laboratories

[PKCS_7] PKCS #7 - Cryptographic Message Syntax Standard, v1.5, 1993, RSA Laboratories

[PKCS_10] *PKCS #10 v1.0: Certification Request Syntax Standard*, 1993, RSA Laboratories

[PKCS_12] *PKCS #12 v1.0: Personal Information Exchange Syntax*, 1999, RSA Laboratories

[RFC2251] M.Wahl u.a., *Lightweight Directory Access Protocol (v3) (RFC2251)*, 1997, IETF

[RFC2459] R. Housley u.a., *Internet X.509 Public Key Infrastructure Certificate and CRL Profile*, IETF, January 1999

[WEB_01] AICPA/CICA, *WebTrust Program for Certification Authorities*, Version 1.0, WebTrust

[X509_97] *ITU-T Recommendation X.509 „Information Technology-Open Systems Interconnection-The Directory: Authentication Framework“*, June 1997

Appendix: DSStore, CertUtil Options

DSSTORE

Dsstore Usage:
DS Certificate Management Options
dsstore <dn of root domain> [-del][-addcrl][-addroot]
Must specify DN of root domain as first param!
i.e. dsstore DC=ntdev,DC=microsoft,DC=com
-del will give you a list of roots, and you can choose 1 for deletion.
-display Display enterprise roots
-addcrl <.crl file> <CA Name> <Machine Name>
-addroot <.crt file> <CA Name>
-addaia <.crt file> <CA Name> (for intermediate CAs)

Other diagnostic options
dsstore [[-domain] [-dcmon]] [-tcainfo] [-pulse] [-entmon] [-macobj]
-domain <domain name> Modify target domain for DCMon
-dcmon Run KDC Certificate monitoring tool
-checksc Check on smart card certificate validity
-tcainfo Display information about Enterprise CAs on domain
-pulse Pulse autoenrollment event(s)

Following options use SAM style machine names, e.g. domain\machinename$
-entmon <SAM machine name> Examine PKI and autoenrollment on remote machine
-macobj <SAM machine name> List attributes on DS machine object of interest to PKI

CERTUTIL

Syntax:
CertUtil [Opt.] -dump
CertUtil [Opt.] [-config Konfig.] -getconfig

CertUtil [Opt.] -decodehex Eingabedatei Ausgabedatei
CertUtil [Opt.] -decode Eingabedatei Ausgabedatei
CertUtil [Opt.] -encode Eingabedatei Ausgabedatei

CertUtil [Opt.] [-config Konfig.] -deny Anforderungskennung
CertUtil [Opt.] [-config Konfig.] -resubmit Anforderungskennung
CertUtil [Opt.] [-config Konfig.] -setattributes Anford.-kennung Attribute
CertUtil [Opt.] [-config Konfig.] -setextension Anford.-kennung Erweiterungsname Flags {Lang | Datum | Zeichenkette | @InDatei}
CertUtil [Opt.] [-config Konfig.] -revoke Seriennummer [Grund]
CertUtil [Opt.] [-config Konfig.] -isvalid Seriennummer
CertUtil [Opt.] [-config Konfig.] -CRL [Ausgabedatei | -]
CertUtil [Opt.] [-config Konfig.] -GetCRL Ausgabedatei [Index]
CertUtil [Opt.] [-config Konfig.] -importcert Zertifikatsdatei [Flags]

CertUtil [Opt.] [-config Konfig.] -ca.cert AusgStellenZertifikatsdatei [Index]
CertUtil [Opt.] [-config Konfig.] -ca.chain AusgStellenAustauschKettendatei [Index]
CertUtil [Opt.] [-config Konfig.] -ping
CertUtil [Opt.] [-config Konfig.] -pingadmin
CertUtil [Opt.] [-config Konfig.] -shutdown

CertUtil [Opt.] [-config Konfig.] -installCert [Stellenzertifikatdatei]
CertUtil [Opt.] [-config Konfig.] -renewCert [Schlüssel wiederverwenden] [Anforderungsdatei]

CertUtil [Opt.] [-config Konfig.] -schema

```
    CertUtil [Opt.] [-config Konfig.] -ConvertMDB

    CertUtil [Opt.] [-config Konfig.] -backup Sicherungsverz. [Kennwort [Inkremental] [Protokoll]]
    CertUtil [Opt.] [-config Konfig.] -backupDB Sicherungsverz. [Inkremental] [Protkoll]
    CertUtil [Opt.] [-config Konfig.] -backupKey Sicherungsverz. [Kennwort]
    CertUtil [Opt.] [-config Konfig.] -restore Sicherungsverz. [Kennwort]
    CertUtil [Opt.] [-config Konfig.] -restoreDB Sicherungsverzeichnis
    CertUtil [Opt.] [-config Konfig.] -restoreKey Sicherungsverz. | PFX-Datei [Kennwort]
    CertUtil [Opt.] [-config Konfig.] -dynamicfilelist
    CertUtil [Opt.] [-config Konfig.] -databaselocations

    CertUtil [Opt.] -store [Zertifikatspeichername [Zert.-index [Ausgabedatei]]
    CertUtil [Opt.] -verifystore Zertifikatspeichername [Zert.-index]
    CertUtil [Opt.] [-config Konfig.] -verifykeys [Schlüsselcontainername StellenZertDatei [Anbietertyp]]
    CertUtil [Opt.] -verify Zertifikatsdatei [StellenZertDatei]

    CertUtil [Opt.] -vroot [Löschen]
    CertUtil [Opt.] -7f Zertifikatsdatei
    CertUtil [Opt.] -error Fehlercode
    CertUtil [Opt.] -getreg [{ca|restore|policy|exit}\[Prog.-kennung\]Reg.-wertname
    CertUtil [Opt.] -setreg [{ca|restore|policy|exit}\[ProgId\]Reg.-wertname Wert
    CertUtil [Opt.] -?

Befehle:
    -dump          -- Bildet Konfigurationsinformationen oder -dateien ab.
    -getconfig     -- Fragt die Standardkonfiguration ab.

    -decodehex     -- Decodiert eine hexadezimal-codierte Datei.
    -decode        -- Decodiert eine Base64-codierte Datei.
    -encode        -- Codiert eine Datei mit Base64.

    -deny          -- Verweigert die ausstehende Anforderung.
    -resubmit      -- Übermittelt die ausstehende Anforderung erneut.
    -setattributes -- Legt Attribute für die ausstehende Anforderung fest.
    -setextension  -- Legt Erweiterung für die ausstehende Anforderung fest.
    -revoke        -- Sperrt das Zertifikat.
    -isvalid       -- "IsValid"-Zertifikat
    -CRL           -- Veröffentlicht Sperrliste [optional in Datei].
    -GetCRL        -- Sperrliste lesen
    -importcert    -- Importiert eine Zertifikatsdatei in eine Datenbank.

    -ca.cert       -- Fragt das Zertifikat der Zertifizierungsstelle ab.
    -ca.chain      -- Fragt die Zertifikatskette der Zertifizierungsstelle ab.
    -ping          -- Sendet Signal zu den Zertifikatsdiensten.
    -pingadmin     -- Sendet Signal zur Adminschnittst. d. Zertifikatsdienste.
    -shutdown      -- Fährt die Zertifikatsdienste herunter.

    -installCert   -- Installiert das Zertifikat der Zertifizierungsstelle.
    -renewCert     -- Erneuert das Zertifizierungsstellen-Zertifikat.

    -schema        -- Bildet das Zertifikatschema ab.
    -ConvertMDB    -- Konvertiert die MDB-Datenbank in aktuelle Version.

    -backup        -- Sichert die Zertifikatsdienste.
    -backupDB      -- Sichert die Zertifikatsdienste-Datenbank.
    -backupKey     -- Sichert Dienstezertifikat und privaten Schlüssel.
    -restore       -- Stellt die Zertifikatsdienste wieder her.
    -restoreDB     -- Stellt die Zertifikatsdienste-Datenbank wieder her.
    -restoreKey    -- Stellt Dienstezertifikat und priv. Schlüssel wieder her.
    -dynamicfilelist -- Zeigt eine dynamische Dateiliste an.
    -databaselocations -- Zeigt den Datenbankpfad an.

    -store         -- Bildet Zertifikatspeicher ab.
```

```
   -verifystore    -- Bestätigt ein Zertifikat ein einem Speicher
   -verifykeys     -- Bestätigt das öffentlich-private Schlüsselpaar.
   -verify         -- Bestätigt die Zertifikatskette.

   -vroot          -- Erstellt/löscht virtuelle Webstammverz. und Freigaben.
   -7f             -- Überprüft das Zertifikat auf 0x7f-Längencodierungen.
   -error          -- Zeigt den Meldungstext des Fehlercodes an.
   -getreg         -- Zeigt den Registrierungswert an.
   -setreg         -- Legt den Registrierungswert fest.
   -?              -- Zeigt diese Syntaxmeldung an.

Optionen:
   -v              -- Führt den Vorgang ausführlich aus.
   -f              -- Erzwingt überschreiben.
   -idispatch      -- Verwendet IDispatch anstelle von COM.
   -user           -- Verwendet HKEY_CURRENT_USER oder Zertifikatspeicher.
   -gmt            -- Zeigt die Uhrzeit in GMT an.
   -silent         -- Verwenden Sie den automatischen Flag, um den Kryptografiekontext abzurufen.
```