[MS-PKCA]:
Public Key Cryptography for Initial Authentication (PKINIT) in Kerberos Protocol Specification

---

Revision Summary

Date	Revision History	Revision Class	Comments
03/14/2007	1.0		Version 1.0 release
04/10/2007	1.1		Version 1.1 release
05/18/2007	1.2		Version 1.2 release
06/08/2007	1.2.1	Editorial	Revised and edited the technical content.
07/10/2007	1.2.2	Editorial	Revised and edited the technical content.
08/17/2007	1.2.3	Editorial	Revised and edited the technical content.
09/21/2007	1.2.4	Editorial	Revised and edited the technical content.
10/26/2007	2.0	Major	Converted document to unified format.
01/25/2008	2.1	Minor	Updated the technical content.
03/14/2008	2.1.1	Editorial	Revised and edited the technical content.
06/20/2008	2.1.2	Editorial	Revised and edited the technical content.
07/25/2008	2.1.3	Editorial	Revised and edited the technical content.
08/29/2008	2.1.4	Editorial	Revised and edited the technical content.
10/24/2008	2.1.5	Editorial	Revised and edited the technical content.
12/05/2008	2.2	Minor	Updated the technical content.
01/16/2009	2.2.1	Editorial	Revised and edited the technical content.
02/27/2009	2.2.2	Editorial	Revised and edited the technical content.
04/10/2009	2.2.3	Editorial	Revised and edited the technical content.
05/22/2009	2.2.4	Editorial	Revised and edited the technical content.
07/02/2009	2.3	Minor	Updated the technical content.
08/14/2009	2.4	Minor	Updated the technical content.
09/25/2009	2.5	Minor	Updated the technical content.
11/06/2009	3.0	Major	Updated and revised the technical content.
12/18/2009	3.1	Minor	Updated the technical content.
01/29/2010	3.2	Minor	Updated the technical content.
03/12/2010	3.3	Minor	Updated the technical content.

Date	Revision History	Revision Class	Comments
04/23/2010	4.0	Major	Updated and revised the technical content.
06/04/2010	5.0	Major	Updated and revised the technical content.
07/16/2010	5.1	Minor	Clarified the meaning of the technical content.
08/27/2010	6.0	Major	Significantly changed the technical content.
10/08/2010	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
11/19/2010	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
01/07/2011	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
02/11/2011	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
03/25/2011	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
05/06/2011	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
06/17/2011	6.1	Minor	Clarified the meaning of the technical content.

Contents

1 Introduction

The Public Key Cryptography for Initial Authentication in Kerberos (PKINIT) protocol [RFC4556] enables the use of public **key** cryptography in the initial authentication exchange (that is, in the **Authentication Service (AS) exchange**) of the Kerberos protocol [MS-KILE]. This specification describes the Public Key Cryptography for Initial Authentication in Kerberos (PKINIT): Microsoft Extensions protocol (PKCA) and how the Microsoft Windows® implementation of PKINIT differs from what is specified in [RFC4556].

In an implementation of [RFC4120] or KILE, the security of the AS exchange depends on the strength of the password used to protect it. This also affects the security of subsequent protocol requests.

By using public key cryptography to protect the initial authentication, the Kerberos protocol [MS-KILE] is substantially strengthened and can be used with already existing public key authentication mechanisms such as smart cards.

This document references the PKINIT methods and data formats [RFC4556] and [RFC5349], that the client and the **KDC** can use both to mutually authenticate during the AS exchange with public and private key pairs and to negotiate the AS-REP key, which allows the KDC to encrypt the AS-REP key sent to the client.

1.1 Glossary

The following terms are defined in [MS-GLOS]:

Authentication Service (AS)
Authentication Service (AS) exchange
authenticator
authorization data
certificate authority (CA)
elliptic curve cryptography (ECC)
key
Key Distribution Center (KDC)
object identifier (OID)
Pre-authentication
privilege attribute certificate (PAC)
public key infrastructure (PKI)
realm
service
session
session key
ticket
ticket-granting service (TGS)
ticket-granting ticket (TGT)

The following terms are specific to this document:

Principal: A unique, individual account known to the **KDC**. Often a user, but it can be a **service** offering a resource on the network.

MAY, SHOULD, MUST, SHOULD NOT, MUST NOT: These terms (in all caps) are used as described in [RFC2119]. All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

1.2 References

References to Microsoft Open Specification documents do not include a publishing year because links are to the latest version of the documents, which are updated frequently. References to other documents include a publishing year when one is available.

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information. Please check the archive site, http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624, as an additional source.

[FIPS140] National Institute of Standards and Technology, "Federal Information Processing Standards Publication 140-2: Security Requirements for Cryptographic Modules", December 2002, http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf

[MS-KILE] Microsoft Corporation, "Kerberos Protocol Extensions".

[MS-NLMP] Microsoft Corporation, "NT LAN Manager (NTLM) Authentication Protocol Specification".

[MS-PAC] Microsoft Corporation, "Privilege Attribute Certificate Data Structure".

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, http://www.rfc-editor.org/rfc/rfc2119.txt

[RFC2315] Kaliski, B., "PKCS #7: Cryptographic Message Syntax Version 1.5", RFC 2315, March 1998, http://www.ietf.org/rfc/rfc2315.txt

[RFC3280] Housley, R., Polk, W., Ford, W., and Solo, D., "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 3280, April 2002, http://www.ietf.org/rfc/rfc3280.txt

[RFC3370] Housley, R., "Cryptographic Message Syntax (CMS) Algorithms", RFC 3370, August 2002, http://www.ietf.org/rfc/rfc3370.txt

[RFC3852] Housley, R., "Cryptographic Message Syntax (CMS)", RFC 3852, July 2004, http://www.ietf.org/rfc/rfc3852.txt

[RFC4120] Neuman, C., Yu, T., Hartman, S., and Raeburn, K., "The Kerberos Network Authentication Service (V5)", RFC 4120, July 2005, http://www.ietf.org/rfc/rfc4120.txt

[RFC4556] Zhu, L., and Tung, B., "Public Key Cryptography for Initial Authentication in Kerberos", RFC 4556, June 2006 http://www.ietf.org/rfc/rfc4556.txt

[RFC5349] Zhu, L., Jaganathan, K., and Lauter, K., "Elliptic Curve Cryptography (ECC) Support for Public Key Cryptography for Initial Authentication in Kerberos (PKINIT)", RFC 5349, September 2008, http://www.ietf.org/rfc/rfc5349.txt

[X509] ITU-T, "Information Technology - Open Systems Interconnection - The Directory: Public-Key and Attribute Certificate Frameworks", Recommendation X.509, August 2005, http://www.itu.int/rec/T-REC-X.509/en

Note There is a charge to download the specification.

[ITUX680] ITU-T, "Abstract Syntax Notation One (ASN.1): Specification of Basic Notation", Recommendation X.680, July 2002, http://www.itu.int/ITU-T/studygroups/com17/languages/X.680-0207.pdf

1.2.2 Informative References

[MS-GLOS] Microsoft Corporation, "Windows Protocols Master Glossary".

1.3 Overview

The PKINIT protocol is a security protocol that authenticates entities on a network using public key cryptography. Kerberos is a security protocol that mutually authenticates entities on a network and can provide user credential delegation after authentication is complete. Kerberos is specified in [RFC4120] and [MS-KILE], and PKINIT is specified in [RFC4556]. [RFC5349] specifies the use of **elliptic curve cryptography (ECC)** within the framework of PKINIT. PKINIT is a **pre-authentication** extension that extends the Kerberos Protocol to use public key cryptography and **ticket-granting ticket (TGT)** data signing during the initial AS exchange.

This specification indicates the variations from [RFC4556] and [RFC5349] in the Microsoft Windows® implementation of PKINIT.

1.4 Relationship to Other Protocols

PKCA is defined as a Kerberos pre-authentication extension ([RFC4120] section 3.1.1). This extension is used in the Kerberos AS exchange [RFC4556], and therefore PKCA relies on a working Kerberos infrastructure and a **certificate authority (CA)** for issuing [X509] certificates. PKCA includes the use of elliptic curve cryptography (ECC). ECC support [RFC5349] relies upon a CA issuing ECC certificates. Applications already using Kerberos can use PKCA without modifications.

In order to support NTLM authentication [MS-NLMP] for applications connecting to network services that do not support Kerberos authentication, when PKCA is used, the KDC returns the user's NTLM **one-way function (OWF)** in the **privilege attribute certificate (PAC)** PAC_CREDENTIAL_INFO buffer ([MS-PAC] section 2.6.1).

1.5 Prerequisites/Preconditions

PKCA assumes the following, in addition to any assumptions specified in [MS-KILE]:

1. The key distribution center (KDC) has an X.509 public key certificate [X509], issued by a certificate authority (CA) and trusted by the clients in the Kerberos realm. For ECC support, the KDC has an ECC public key certificate issued by a CA and trusted by clients in the Kerberos realm. The issuing of these [X509] certificates is not addressed in this protocol specification.

2. A cryptographic-strength random-number generator is available for generating keys and other cryptographically sensitive information.<1>

3. Each user has an [X509] certificate suitable for use with PKINIT. Details about such a certificate are specified in [RFC4556] Appendix C.

Details about general Kerberos assumptions are specified in [RFC4120] section 1.6.

1.6 Applicability Statement

PKCA is used only in environments that use Kerberos, and it requires the deployment of a **Public Key Infrastructure (PKI)** for issuing [X509] certificates.

1.7 Versioning and Capability Negotiation

PKCA does not have explicit versioning; it is tied to the Kerberos protocol [MS-KILE] versioning mechanisms, as specified in [RFC4120] section 7.5.6. Capability negotiation is as specified in [RFC4556] sections 3.3 and 3.4.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

There are no standards assignments in PKCA beyond what is specified in [RFC4556] and [RFC5349].

2 Messages

2.1 Transport

Messages are carried in the Kerberos AS exchange as pre-authentication data, as specified in [RFC4120] section 5.2.7.

2.2 Message Syntax

The message syntax is as specified in [RFC4556] section 3.2.<2>

PKCA MAY support these variations based on an earlier draft of [RFC4556] for interoperability.<3>

An earlier draft of [RFC4556] supported a different pre-authentication data identifier:

- PA-PK-AS-REP_OLD 15 <4>

The algorithm identifier in Cryptographic Message Syntax (CMS) messages, as specified in [RFC2315] and [RFC3852], is md5WithRSAEncryption instead of md5 ([RFC3370] sections 3.2 and 2.2).<5> The support of sha-1WithRSAEncryption is added [RFC3370].<6> The support of ecdsa-with-Sha1, ecdsa-with-Sha256, ecdsa-with-Sha384, and ecdsa-with-Sha512 ([RFC5349] section 3) is also added.<7>

The following ECC curves ([RFC5349] section 5) are supported:<8>

- ECPRGF256Random | groupP-256 | secp256r1
- ECPRGF384Random | groupP-384 | secp384r1
- ECPRGF521Random | groupP-521 | secp521r1

2.2.1 PA-PK-AS-REP_OLD

The data for the PA-PK-AS-REP_OLD pre-authentication data identifiers is based on an earlier draft of [RFC4556]; therefore, there are some differences in the message format. The ASN.1 [ITUX680] description of the message that is used in place of the message format specified in [RFC4556] section 3.2.1 follows.<9>

```
PKINIT DEFINITIONS EXPLICIT TAGS ::=
BEGIN
--EXPORTS ALL--
IMPORTS
KerberosTime, PrincipalName, Realm, EncryptionKey
FROM KerberosV5Spec2
{ iso(1) identified-organization(3) dod(6) internet(1) security(5)
         kerberosV5(2) }
-- Different from [RFC4556] Appendix A
ContentInfo, EnvelopedData, SignedData, IssuerAndSerialNumber
FROM CryptographicMessageSyntax2004
{ iso(1) member-body(2) us(840) rsadsi(113549)
pkcs(1) pkcs-9(9) smime(16) modules(0) cms-
2004(24) }
-- Same as defined in [RFC3852]
AlgorithmIdentifier
FROM PKIX1Explicit88
{ iso(1) identified-organization(3) dod(6) internet(1) security(5)
```

```
          mechanisms(5) pkix(7) id-mod(0) id-pkix1-explicit(18) };
-- From [RFC3280] (Same as defined in [RFC4556] Appendix A)
--
-- PKINT data types
--
PA-PK-AS-REQ ::= SEQUENCE {
-- PA TYPE 15
signedAuthPack [0] IMPLICIT OCTET STRING
}


AuthPack::= SEQUENCE {
    pkAuthenticator  [0] PKAuthenticator
}


--
-- PK-AUTHENTICATOR - Different from [RFC4556]
-- Appendix A, PKAuthenticator.
--
PKAuthenticator::= SEQUENCE {
    kdc-name   [0] PRINCIPAL-NAME,
    kdc-realm  [1] REALM,
-- name and realm of the KDC issuing the ticket
    cusec      [2] INTEGER,
    ctime      [3] KerberosTime,
    nonce      [4] INTEGER
}
END
```

PA-PK-AS-REQ field:

- signedAuthPack: Contains content identical to the content of the signedAuthPack field, as specified in [RFC4556] section 3.2.1.

AuthPack field:

- pkAuthenticator: Contains a PKAuthenticator structure, as defined in this document. This variation of the AuthPack structure is different from the one specified in [RFC4556].

PKAuthenticator fields:

- kdc-name: Contains the name portion of the ticket-granting service (TGS) name of the KDC that will service the request, as specified in [RFC4120] section 7.3.
- kdc-realm: Contains the realm portion of the TGS name of the KDC that will service the request, as specified in [RFC4120] section 7.3.
- cusec: Contains the same content of the corresponding, identically named field in the type PKAuthenticator, as specified in [RFC4556] section 3.2.1.
- ctime: Contains the same content of the corresponding, identically named field in the type PKAuthenticator, as specified in [RFC4556] section 3.2.1.
- nonce: Contains the same content of the corresponding, identically named field in the type PKAuthenticator, as specified in [RFC4556] section 3.2.1.

2.2.2 PA-PK-AS-REP_OLD

The data for the PA-PK-AS-REP_OLD pre-authentication data identifiers is based on an earlier draft of [RFC4556]; therefore, there are some differences in the message format. The ASN.1 [ITUX680] description of the message that is used in place of the message format specified in [RFC4556] section 3.2.3 follows.<10>

```
--
-- KERB-REPLY-KEY-PACKAGE - Different from [RFC4556]
-- Appendix A, ReplyKeyPack
--
KERB-REPLY-KEY-PACKAGE ::= SEQUENCE {
    replyKey [0] EncryptionKey,
-- Contains the session key used to encrypt the enc-part
-- field in the AS-REP, for example, the AS reply key.


nonce [1] INTEGER,
-- binds response to the request; must be same as the nonce
-- passed in the PK-AUTHENTICATOR.
    ...
} --#public-
```

KERB-REPLY-KEY-PACKAGE fields:

- replyKey: Contains the same content of the identically named field in the type ReplyKeyPack, as specified in [RFC4556] section 3.2.3.2.
- nonce: Contains the nonce from the PKAuthenticator structure in the PA-PK-AS-REQ request.

However, if the AS-REQ message contains a padata of type KRB5-PADATA-AS-CHECKSUM(132) with no corresponding data field (padata-value is an empty OCTET STRING), then the PA-PK-AS-REP_OLD pre-authentication data contains the same data as specified in [RFC4556] section 3.2.3.2.

2.2.3 PA-PK-AS-REQ

The PA-PK-AS-REQ message format is specified in [RFC4556] section 3.2.1.<11>

2.2.4 PA-PK-AS-REP

The PA-PK-AS-REP message format is specified in [RFC4556] section 3.2.3.<12> The returned **ticket** does not include the AD-INITIAL-VERIFIED-CAS type in the **authorization data**. The content of the SignedData field in the content of EnvelopedData is encoded, as specified in [RFC2315] section 7, not as specified in [RFC3852]. Therefore, the data is not wrapped in OCTET STRING; rather, it is wrapped in an ANY DEFINED BY content specific type, as specified in [RFC2315] section 7.

2.3 Directory Service Schema Elements

None.

3 Protocol Details

3.1 Common Details

3.1.1 Abstract Data Model

The abstract data model follows what is specified in [RFC4556].

3.1.2 Timers

None.

3.1.3 Initialization

During initialization, the [FIPS140]-compliant random-number generator for keys and nonces is initialized.

3.1.4 Higher-Layer Triggered Events

None.

3.1.5 Message Processing Events and Sequencing Rules

In addition to the required ([RFC4556] section 3.1.1) and recommended ([RFC4556] section 3.1.2) algorithms, PKCA supports the rc2-cbc ([RFC4556] section 3.1.4) algorithm. An implementer SHOULD specify des-ede3-cbc ([RFC4556] section 3.1.2) as the default algorithm.<13>

PKCA does not implement the id-pkinit-san algorithm ([RFC4556] section 3.2.2).

3.1.5.1 Client

The Kerberos client SHOULD send only a PA-PK-AS-REQ pre-authentication data identifier.<14> <15>

Kerberos clients can process either the PA-PK-AS-REP_OLD or the PA-PK-AS-REP pre-authentication data identifier in the reply, but not both.<16>

3.1.5.2 KDC

If the KDC receives both a PA-PK-AS-REQ and PA-PK-AS-REQ_OLD, the KDC should return KRB_ERROR_GENERIC.

The KDC SHOULD process the PA-PK-AS-REQ pre-authentication data identifier.<17> The KDC SHOULD respond with PA-PK-AS-REP.<18>

The KDC MUST return the user's unicodePwd attribute ([MS-ADA3] section 2.331) in the **NTLM_SUPPLEMENTAL_CREDENTIAL** buffer ([MS-PAC] section 2.6.4).

3.1.6 Timer Events

None.

3.1.7 Other Local Events

There are no local events other than what is specified in [RFC4556].

4 Protocol Examples

The following sections describe three common scenarios to illustrate the function of the KILE.

4.1 Interactive Logon Using Smart Cards

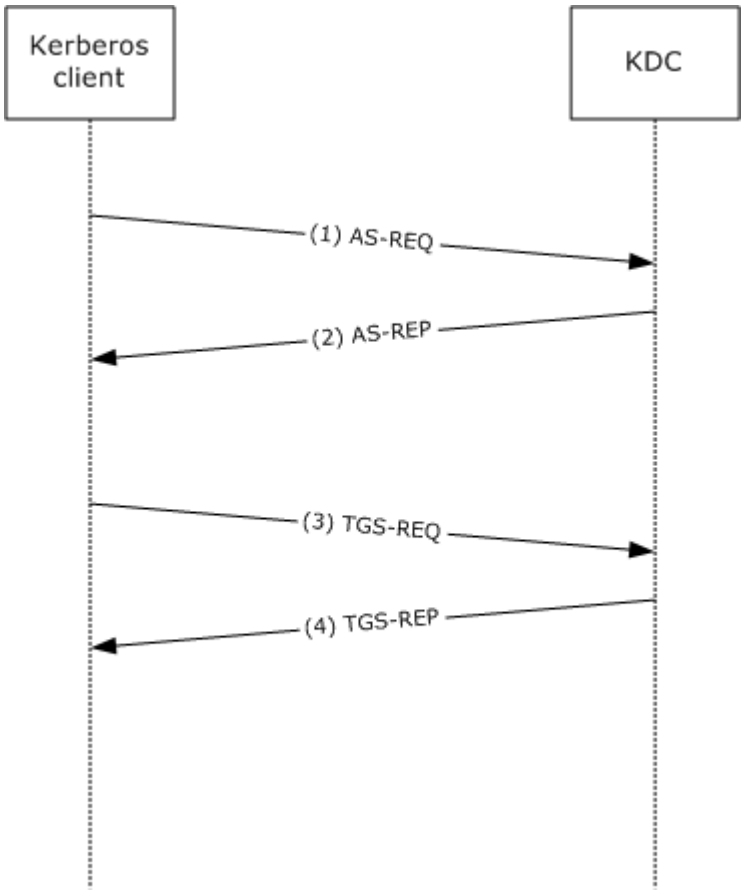

Figure 1: Interactive logon

Step 1: A user attempts to log on to a client. At the logon screen, the user selects the certificate and types the PIN. Using the PIN to unlock the smart card, the client generates an AS-REQ with PA-PK-AS-REQ pre-authentication data ([RFC4556] section 3.2.1) and sends the request to the KDC.

Step 2: The KDC validates the AS-REQ ([RFC4120] section 3.1.2), including verifying the user's signature and validating certificate ([RFC4556] section 3.2.2). If the AS-REQ is valid, the KDC generates an AS-REP ([RFC4556] section 3.2.3), with a PAC ([MS-KILE] section 3.3.5.3.2) in the authorization_data field of the TGT, and sends the reply to the client.

Step 3: The client validates the AS-REP ([RFC4556] section 3.2.4). For interactive logons, the client runtime requests authentication to host/hostname.domain, where hostname is the actual name of the client machine, and domain is the domain or realm of the client machine. If the AS-REP is valid, the client generates a TGS-REQ based on the TGT that is obtained in step 2 to obtain a service ticket for host/hostname.domain ([RFC4120] section 3.3.1) and sends the request to the KDC.

Step 4: The KDC validates the TGS-REQ ([RFC4120] section 3.3.2) ([MS-KILE] section 3.3.5.4.1). If the TGS-REQ is valid, the KDC adds Domain Local Groups to the PAC ([MS-KILE] section 3.3.5.4.3), generates a TGS-REP ([RFC4120] section 3.3.3), and sends the reply to the client.

The client validates the TGS-REP ([MS-KILE] section 3.3.4). If the TGS-REP is valid, the service ticket is then interpreted by the Kerberos runtime within the local workstation.

The following fields from the KERB_VALIDATION_INFO field of the PAC ([MS-PAC] Section 2.5) are required by the interactive logon client runtime to authorize the user for local interactive logon, and to establish the necessary management profile for the user:

- LogonTime
- LogoffTime
- KickOffTime
- PasswordLastSet
- PasswordCanChange
- EffectiveName
- FullName
- LogonScript
- ProfilePath
- HomeDirectory
- HomeDirectoryDrive
- LogonCount
- BadPasswordCount
- LogonServer
- LogonDomainName
- UserAccountControl

4.2 Network Logon Using Smart Cards

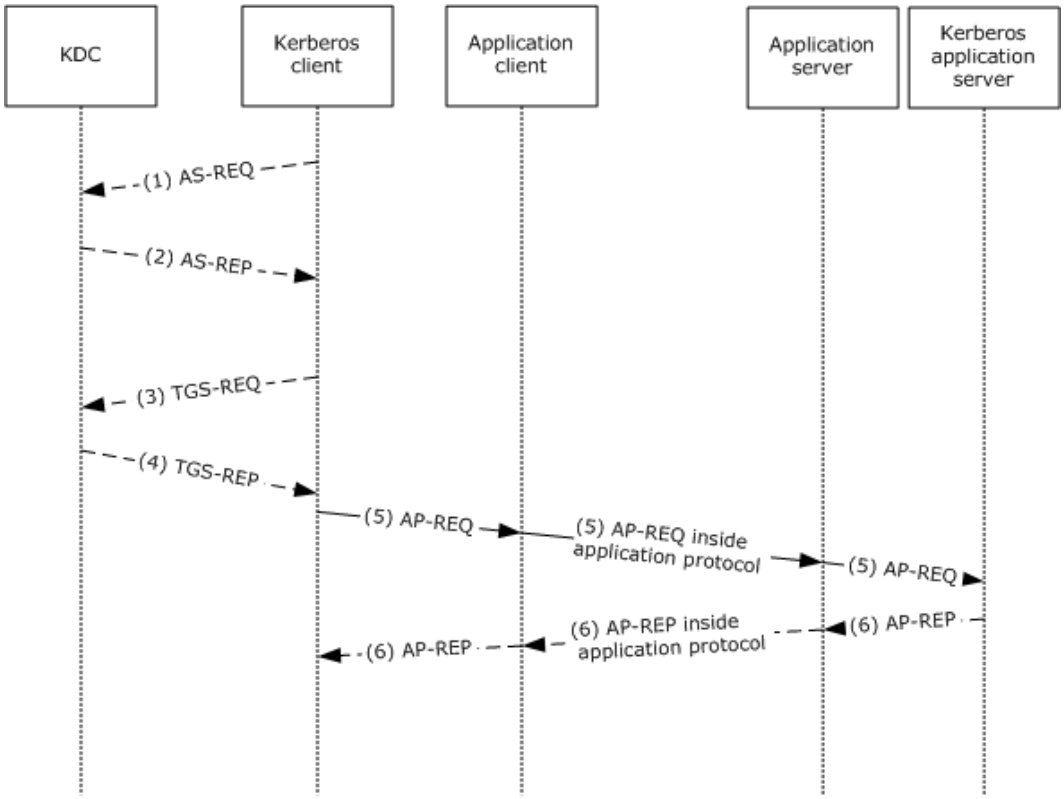

Figure 2: Network logon

When an application wants authentication, it calls GSS_Init_sec_context ([RFC2743] section 2.2.1) to either invoke KILE [MS-KILE] directly, or SPNEGO [MS-SPNG] which can invoke Kerberos.

Step 0: The application client calls GSS_Init_sec_context ([RFC2743] section 2.2.1).

When the client does not have a TGT, steps 1 through 2, as described in section 4.1, must be performed.

When the client does not have a service ticket for the application server, steps 3 and 4, as described in section 4.1, must be performed.

Step 5: The Kerberos client generates a GSS-API initial token ([RFC1964] section 1.1.1) containing an AP-REQ ([RFC4120] section 3.2.2) and returns it to the application.

Step 6: The application server calls GSS_Accept_sec_context ([RFC2743] section 2.2.2). The Kerberos application server validates the AP-REQ ([RFC4120] section 3.2.3). If the AP-REQ is valid and the client requested mutual authentication, the Kerberos application server generates a GSS-API response token ([RFC1964] section 1.1.2) containing an AP-REP ([RFC4120] section 3.2.4) and returns it to the application server. The Kerberos application server provides the authorization data from the ticket to the Microsoft Windows® system which creates a Windows-specific object that is known as an access token, which is used with the Windows system-provided authorization functions.

If mutual authentication was requested, the application client calls GSS_Init_sec_context ([RFC2743] section 2.2.1). The Kerberos client validates the AP-REP ([RFC4120] section 3.2.5). If the AP-REP is valid, the Kerberos client returns GSS_S_COMPLETE ([RFC2743] section 2.2.1).

4.3 Non-RFC Kerberos Clients during AS-REQ

PKCA clients developed prior to finalizing RFC 4556 support a PKInit pre-authentication data based on an earlier draft of [RFC4556].

Step 1: A user attempts to log on to a client. At the logon screen, the user selects the certificate and types the PIN. Using the PIN to unlock the smart card, the client generates an AS-REQ with PA-PK-AS-REP_OLD pre-authentication data (section 2.2.1) and sends the request to the KDC.

Step 2: The KDC validates the AS-REQ ([RFC4120] section 3.1.2) including verifying the user's signature and validating certificate ([RFC4556] section 3.2.2). Since the PA-PK-AS-REP_OLD version of the pre-authentication data does not contain a paChecksum, the KDC does not return a KRB-ERROR with the code KDC_ERR_PA_CHECKSUM_MUST_BE_INCLUDED ([RFC4556] section 3.2.3). If the AS-REQ is valid, with the exception of the paChecksum checks, the KDC generates an AS-REP ([RFC4556] section 3.2.3) using the PA-PK-AS-REP_OLD, instead of the PA-PK-AS-REP with a PAC ([MS-KILE] section 3.3.5.3.2) in the authorization_data field of the TGT, and sends the reply to the client.

5 Security

5.1 Security Considerations for Implementers

PKCA security considerations are specified in [RFC4556]. PA-PK-AS-REP_OLD is the earlier version of PA-PK-AS-REQ and PA-PK-AS-REP, and has the same security considerations.

5.2 Index of Security Parameters

PKCA security parameters are specified in [RFC4556].

Security parameter	Section
PKAuthenticator	2.2.1
KERB-REPLY-KEY-PACKAGE	2.2.2

6 Appendix A: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Microsoft Windows® 2000 operating system
- Windows® XP operating system
- Windows Server® 2003 operating system
- Windows Vista® operating system
- Windows Server® 2008 operating system
- Windows® 7 operating system
- Windows Server® 2008 R2 operating system

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

<1> Section 1.5: Windows contains a FIPS-140-validated random-number generator, as specified in [FIPS140].

<2> Section 2.2: Supported in Windows 2000, Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2.

<3> Section 2.2: Supported in Windows 2000, Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2.

<4> Section 2.2: Windows 2000, Windows XP, and Windows Server 2003 sent PA-PK-AS-REP_OLD where [RFC4120] would have them send PA-PK-AS-REQ or PA-PK-AS-REP.

<5> Section 2.2: Supported by Windows 2000, Windows XP SP2, and Windows Server 2003 with SP1. In Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2, the **object identifier (OID)** has been updated to match CMS algorithms, as specified in [RFC3370] sections 3.2 and 2.2. Windows 2000, Windows XP SP2, Windows Server 2003 with SP1, Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2 accept the correct OID.

<6> Section 2.2: Supported by Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2.

<7> Section 2.2: ECC supported by Windows 7 and Windows Server 2008 R2.

<8> Section 2.2: Supported by Windows 7 and Windows Server 2008 R2.

<9> Section 2.2.1: In Windows 2000, Windows XP SP2, and Windows Server 2003 with SP1, SignedData (as specified in [RFC3852]) is encoded as specified in [RFC2315] section 9, not as

specified in [RFC3852] section 5. Therefore, the data is not wrapped in OCTET STRING; rather, it is wrapped in an ANY, as specified in [RFC2315] section 7. However, in Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2, the SignedData (as specified in [RFC3852]) is encoded as specified in [RFC3852]. Windows 2000, Windows XP SP2, Windows Server 2003 with SP1, Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2 accept the SignedData, as specified in [RFC3852]. In Windows 2000, Windows XP SP2, and Windows Server 2003 with SP1, the DHRepInfo form is not implemented; the Public Key Encryption style is used, as specified in [RFC4556] section 3.2.3.2. The Diffie-Hellman key delivery method, as specified in [RFC4556] section 3.2.3.1, is supported in Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2.

In Windows 2000, Windows XP SP2, and Windows Server 2003 with SP1, the content-type field of the SignedData in PA-PK-AS-REQ is id-data, as specified in [RFC3852] section 4, instead of id-pkinit-authData. However, in Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2, the content-type field of the SignedData is id-pkinit-authData, as specified in [RFC4556] section 3.2.3.2. Windows 2000, Windows XP SP2, Windows Server 2003 with SP1, Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2 accept id-data in the PA-PK-AS-REQ_OLD pre-authentication data.

<10> Section 2.2.2: In Windows 2000, Windows XP SP2, and Windows Server 2003 with SP1, the content-type field of the SignedData type inside the EnvelopedData type in the PA-PK-AS-REP_OLD pre-authentication data is id-data, as specified in [RFC3852] section 4, instead of id-pkinit-rkeyData, as specified in [RFC4556]. However, in Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2, the content-type field is id-pkinit-rkeyData, as specified in [RFC4556]. Windows 2000, Windows XP SP2, Windows Server 2003 with SP1, Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2 all accept id-data in the SignedData contained in the PA-PK-AS-REP_OLD pre-authentication data.

In addition, Windows 2000, Windows XP SP2, Windows Server 2003 with SP1, Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2 do not process id-pkinit-san in the client's [X509] certificate, if present, as specified in [RFC4556] section 3.2.4.

<11> Section 2.2.3: Supported in Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2.

<12> Section 2.2.4: The RFC version of PA-PK-AS-REP is supported in Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2.

<13> Section 3.1.5: In Windows with PKCA, the KDC supports both des-ede3-cbc and rc2-cbc. If both des-ede3-cbc and rc2-cbc are present, the KDC uses des-ede3-cbc.

<14> Section 3.1.5.1: In Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2, the PKINIT pre-authentication data identifiers have been updated to match what is specified in [RFC4556], with one addition (KRB5-PADATA-AS-CHECKSUM) as noted below. However, for backward-compatibility reasons, if the client is not detecting that the KDC is running Windows Server 2008, Windows 7, or Windows Server 2008 R2, it sends both.

In Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2, the client sends additional padata (KRB5-PADATA-AS-CHECKSUM) besides what is specified in [RFC4556]. This is padata that contains NO data.

```
#define KRB5_PADATA_AS_CHECKSUM         132 /* AS checksum */
```

Clients running Windows XP and Windows 2000 also send this additional padata type.

<15> Section 3.1.5.1: Windows 2000, Windows XP, and Windows Server 2003 clients send a PA-PK-AS-REP_OLD pre-authentication data identifier. Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2 clients send a PA-PK-AS-REP_OLD pre-authentication data identifier when all of the following are true:

- the user certificate has a smart card logon EKU, and
- the user certificate has a UPN in Subject Alternative Name.

<16> Section 3.1.5.1: Windows 2000 and Windows XP SP2 Kerberos clients can only process PA-PK-AS-REP-WINDOWS-OLD.

<17> Section 3.1.5.2: Windows 2000 and Windows Server 2003 KDCs always discard the PA-PK-AS-REQ data identifier and process the PA-PK-AS-REP_OLD data identifier, if present.

<18> Section 3.1.5.2: Windows 2000 and Windows Server 2003 KDCs respond with PA-PK-AS-REP_OLD.

7 Change Tracking

This section identifies changes that were made to the [MS-PKCA] protocol document between the May 2011 and June 2011 releases. Changes are classified as New, Major, Minor, Editorial, or No change.

The revision class **New** means that a new document is being released.

The revision class **Major** means that the technical content in the document was significantly revised. Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements or functionality.
- An extensive rewrite, addition, or deletion of major portions of content.
- The removal of a document from the documentation set.
- Changes made for template compliance.

The revision class **Minor** means that the meaning of the technical content was clarified. Minor changes do not affect protocol interoperability or implementation. Examples of minor changes are updates to clarify ambiguity at the sentence, paragraph, or table level.

The revision class **Editorial** means that the language and formatting in the technical content was changed. Editorial changes apply to grammatical, formatting, and style issues.

The revision class **No change** means that no new technical or language changes were introduced. The technical content of the document is identical to the last released version, but minor editorial and formatting changes, as well as updates to the header and footer information, and to the revision summary, may have been made.

Major and minor changes can be described further using the following change types:

- New content added.
- Content updated.
- Content removed.
- New product behavior note added.
- Product behavior note updated.
- Product behavior note removed.
- New protocol syntax added.
- Protocol syntax updated.
- Protocol syntax removed.
- New content added due to protocol revision.
- Content updated due to protocol revision.
- Content removed due to protocol revision.
- New protocol syntax added due to protocol revision.

- Protocol syntax updated due to protocol revision.
- Protocol syntax removed due to protocol revision.
- New content added for template compliance.
- Content updated for template compliance.
- Content removed for template compliance.
- Obsolete document removed.

Editorial changes are always classified with the change type **Editorially updated.**

Some important terms used in the change type descriptions are defined as follows:

- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.
- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

The changes made to this document are listed in the following table. For more information, please contact protocol@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
1.2 References	Added explanatory statement regarding the removal of the publishing year from Microsoft Open Specification document references.	N	Content updated.

8 Index