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

Revision Summary

Date	Revision History	Revision Class	Comments
10/22/2006	0.01		MCPP Milestone 1 Initial Availability
01/19/2007	1.0		MCPP Milestone 1
03/02/2007	1.1		Monthly release
04/03/2007	1.2		Monthly release

Date	Revision History	Revision Class	Comments
05/11/2007	1.3		Monthly release
06/01/2007	1.3.1	Editorial	Revised and edited the technical content.
07/03/2007	1.3.2	Editorial	Several editorial changes made.
07/20/2007	1.3.3	Editorial	Revised and edited the technical content.
08/10/2007	1.3.4	Editorial	Revised and edited the technical content.
09/28/2007	1.3.5	Editorial	Revised and edited the technical content.
10/23/2007	1.3.6	Editorial	Revised and edited the technical content.
11/30/2007	1.3.7	Editorial	Revised and edited the technical content.
01/25/2008	1.4	Minor	Updated the technical content.

Table of Contents

1 Introduction

The Public Key Cryptography for Initial Authentication (PKINIT) in Kerberos Protocol enables the use of public **key** cryptography in the initial authentication exchange (that is, in the **Authentication Service (AS) exchange**) of the Kerberos Protocol, as specified in [RFC4556]. This document specifies the Windows implementation of PKINIT where it differs from what is specified in [RFC4556].

In the initial authentication exchange (or AS exchange), the Kerberos Protocol uses passwords that are shared between the client and the **Key Distribution Center (KDC)** to derive a key (the AS-REP key) that is used to encrypt Kerberos **ticket-granting ticket (TGT)** and **ticket-granting service (TGS)** requests, as specified in [RFC4120] and in [MS-KILE]. Using this scheme, the Kerberos Protocol encryption strength is tied to the strength of the passwords that are used and affects the security of subsequent protocol requests.

By using public key cryptography to protect the initial authentication, the Kerberos Protocol 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, as specified in [RFC4556], that the client and the Key Distribution Center (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 sent to the client.

1.1 Glossary

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

Authentication Service (AS)
Authentication Service (AS) Exchange
Authorization Data
Certificate Authority (CA)
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:

Authenticator: A record sent with a **ticket** to a server to help certify the client's knowledge of the encryption **key** in the **ticket**, to help the server detect replay attacks by proving that the **authenticator** is recently constructed, and to help the two parties select additional encryption **keys** for a particular connection authenticated by the Kerberos Protocol. The use of **authenticators**, including how **authenticators** are validated, is specified in [RFC4120] section 5.5.1. For more information, see [KAUFMAN].

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

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-DTYP] Microsoft Corporation, "Windows Data Types", January 2007.

[MS-GLOS] Microsoft Corporation, "Windows Protocols Master Glossary", March 2007.

[MS-KILE] Microsoft Corporation, "Kerberos Protocol Extensions", January 2007.

[MS-NLMP] Microsoft Corporation, "NT LAN Manager (NTLM) Authentication Protocol Specification", June 2007.

[MS-PAC] Microsoft Corporation, "Privilege Attribute Certificate Data Structure", January 2007.

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

[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

[KAUFMAN] Kaufman, C., Perlman, R., and M. Speciner, "Network Security: Private Communication in a Public World, Second Edition", Prentice Hall, 2002, ISBN: 0130460192.

1.3 Protocol Overview (Synopsis)

Public Key Cryptography for Initial Authentication (PKINIT) in Kerberos Protocol (PKCA) is a security protocol that authenticates entities on a network with 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]. 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 document specifies the variations from [RFC4556] in the Windows implementation of PKINIT.

1.4 Relationship to Other Protocols

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

1.5 Prerequisites/Preconditions

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

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

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

3. Each user MUST be issued an [X509] certificate suitable for use with PKINIT. Details on such a certificate are as specified in [RFC4556] Appendix C.

Details on general Kerberos assumptions are as 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 for issuing [X509] certificates.

1.7 Versioning and Capability Negotiation

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

1.8 Vendor-Extensible Fields

PKCA contains the capability to add new encryption algorithms, as specified in [RFC4556] section 3.1.<2>

1.9 Standards Assignments

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

2 Messages

The following sections specify how PKCA messages are transported and message syntax.

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.<3>

Public Key Cryptography for Initial Authentication (PKINIT) in Kerberos Protocol Specification MAY support these variations based on an earlier draft of [RFC4556] for interoperability. <4>

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

- PA_PK_AS_REQ_WINDOWS_OLD 15
- PA_PK_AS_REP_WINDOWS_OLD 15

PA_PK_AS_REQ_WINDOWS_OLD corresponds to PA_PK_AS_REQ and PA_PK_AS_REP_WINDOWS_OLD corresponds to PA_PK_AS_REP.

Note that PA_PK_AS_REP is used for both the request and the reply; therefore, 15 is in both the request and the reply.<5>

A KDC MAY process both the PA-PK-AS-REQ and the PA-PK-AS-REQ-WINDOWS-OLD pre-authentication data identifiers<6>, and it MAY discard the PA-PK-AS-REQ-WINDOWS-OLD if both PA-PK-AS-REQ and the PA-PK-AS-REQ-WINDOWS-OLD are received.<7>

Kerberos clients can process both the PA-PK-AS-REP-WINDOWS-OLD and the PA-PK-AS-REP pre-authentication data identifiers in the reply; however, if both are received, they only process the PA_PK_AS_REP and discard the PA-PK-AS-REP-WINDOWS-OLD.<8>

The digest algorithm in Cryptographic Message Syntax (CMS) messages, as specified in [RFC3852], is md5WithRSAEncryption instead of md5, as specified in [RFC3370] sections 3.2 and 2.2.<9> The support of sha-1WithRSAEncryption is added, as specified in [RFC3370].<10>

2.2.1 PA-PK-AS-REQ-WINDOWS-OLD

The data for the PA-PK-AS-REQ-WINDOWS-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 (as specified in [ITUX680]) description of the message that is used in place of the message format specified in [RFC4556] section 3.2.1 follows.<11>

```
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 [RFC 3852]
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 the same content of the signedAuthPack, 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 than 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-REQ

The PA-PK-AS-REQ message format is specified in [RFC4556] section 3.2.1.<12>

2.2.3 PA-PK-AS-REP

The PA-PK-AS-REP message format is specified in [RFC4556] section 3.2.3.<13> The returned **ticket** does not include the AD-INITIAL-VERIFIED-CAS in the **authorization data**. The content of the SignedData 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, as specified in [RFC2315] section 7. In Windows 2000 and Windows Server 2003 SP1, the KDC responds with PA-PK-AS-REP-WINDOWS-OLD pre-authentication data that contains the following ASN.1 encoded structure instead of what is specified in [RFC4556] section 3.2.3.2.<14>

```
--
-- 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 contains a padata of type KRB5-PADATA-AS-CHECKSUM (132) with no corresponding data field (padata-value is an empty OTCET STRING), then the PA-PK-AS-REP-WINDOWS-OLD pre-authentication data contains the same data as specified in [RFC4556] section 3.2.3.2.

3 Protocol Details

The following sections specify protocol details including an abstract data model and message processing rules.

3.1 Common Details

3.1.1 Abstract Data Model

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

3.1.2 Timers

There are no timers.

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

There are no higher-layer triggered events in common to all parts of PKCA.

3.1.5 Message Processing Events and Sequencing Rules

There are no message processing events or sequencing rules specific to PKCA.

3.1.6 Timer Events

There are no timer events.

3.1.7 Other Local Events

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

4 Protocol Examples

PKCA uses the same protocol as specified in [MS-KILE].

5 Security

The following sections specify security considerations for implementers.

5.1 Security Considerations for Implementers

PKCA security considerations are specified in [RFC4556].

5.2 Index of Security Parameters

Security Parameter	Section
Encryption and signature algorithms	1.8

6 Appendix A: Windows Behavior

The information in this specification is applicable to the following versions of Windows:

- Windows Server 2008
- Windows Server 2003
- Windows Vista
- Windows XP
- Windows 2000

Exceptions, if any, are noted below. Unless otherwise specified, any statement of optional behavior in this specification prescribed using the terms SHOULD or SHOULD NOT implies Windows behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that Windows does not follow the prescription.

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

<2> Section 1.8: Windows supports the following encryption types: des-ede3-cbc and rc2-cbc; by default, it uses des-ede3-cbc. Windows does not implement the id-pkinit-san algorithm, as specified in [RFC4556] section 3.2.2.

<3> Section 2.2: Supported in Windows Vista and Windows Server 2008.

<4> Section 2.2: Supported by Windows 2000, Windows XP SP2, and Windows 2000 Server SP1

<5> Section 2.2: In Windows Vista and Windows Server 2008, 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 unaware that the KDC is running Windows Server 2008, it sends both.

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

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

Windows XP and Windows 2000 clients also send this additional padata type.

<6> Section 2.2: Supported in Windows Server 2008.

<7> Section 2.2: Windows Server 2008 will discard. However, a KDC that runs Windows Server 2003 or Windows 2000 always discards the PA-PK-AS-REQ and processes the PA-PK-AS-REQ-WINDOWS-OLD if that is received.

<8> Section 2.2: Windows Vista and Windows Server 2008 Kerberos clients. A Windows XP SP2 Kerberos client always discards the PA-PK-AS-REP pre-authentication data and processes the PA-PK-AS-REP-WINDOWS-OLD if that is received.

<9> Section 2.2: Supported by Windows 2000, Windows XP SP2, and Windows Server 2003 SP1. In Windows Vista and Windows Server 2008, 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 SP1, Windows Vista, and Windows Server 2008 all accept the correct OID.

<10> Section 2.2: Supported by Windows Vista and Windows Server 2008.

<11> Section 2.2.1: In Windows 2000, Windows XP SP2, and Windows Server 2003 SP1, SignedData (as specified in [RFC3852]) is encoded as specified in [RFC2315] section 9, not as specified in [RFC3852]. 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 and Windows Server 2008, the SignedData (as specified in [RFC3852]) is encoded as specified in [RFC3852]. Windows 2000, Windows XP SP2, Windows Server 2003 SP1, Windows Vista, and Windows Server 2008 all accept the SignedData, as specified in [RFC3852].

In Windows 2000, Windows XP SP2, and Windows Server 2003 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 and Windows Server 2008.

In Windows 2000, Windows XP SP2, and Windows Server 2003 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 and Windows Server 2008, the content-type field of the SignedData is id-pkinit-authData, as specified in [RFC4556]. Windows 2000, Windows XP SP2, Windows Server 2003 SP1, Windows Vista, and Windows Server 2008 all accept id-data in the PA-PK-AS-REQ-WINDOWS-OLD pre-authentication data.

<12> Section 2.2.2: Supported in Windows Vista and Windows Server 2008.

<13> Section 2.2.3: The RFC version of PA-PK-AS-REP is supported in Windows Vista and Windows Server 2008.

<14> Section 2.2.3: In Windows 2000, Windows XP SP2, and Windows Server 2003 SP1, the content-type field of the SignedData inside the EnvelopedData in the PA-PK-AS-REP-WINDOWS-OLD pre-authentication data is id-data, as specified in [RFC3852] section 4, instead of id-pkinit-rkey Data, as specified in [RFC4556]. However, in Windows Vista and Windows Server 2008, the content-type field is id-pkinit-rkey Data, as specified in [RFC4556]. Windows 2000, Windows XP SP2, Windows Server 2003 SP1, Windows Vista, and Windows Server 2008 all accept id-data in the SignedData contained in the PA-PK-AS-REP-WINDOWS-OLD pre-authentication data.

When PKCA is used, even though the user did not authenticate with NTLM, the KDC returns the NTLM credentials for that user in the **privilege attribute certificate (PAC)** PAC_CREDENTIAL_INFO buffer, as specified in [MS-PAC] section 2.6.1. These credentials allow an application to connect to some network service that does not accept Kerberos tickets and requires NTLM authentication.

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

7 Index