

Network Working Group  
Request for Comments: 4537  
Updates: 4120  
Category: Standards Track

L. Zhu  
P. Leach  
K. Jaganathan  
Microsoft Corporation  
June 2006

## Kerberos Cryptosystem Negotiation Extension

### Status of This Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

### Copyright Notice

Copyright (C) The Internet Society (2006).

### Abstract

This document specifies an extension to the Kerberos protocol as defined in RFC 4120, in which the client can send a list of supported encryption types in decreasing preference order, and the server then selects an encryption type that is supported by both the client and the server.

### Table of Contents

1. Introduction . . . . .	2
2. Conventions Used in This Document . . . . .	2
3. Negotiation Extension . . . . .	2
4. Security Considerations . . . . .	4
5. Acknowledgements . . . . .	4
6. Normative References . . . . .	4

## 1. Introduction

Under the current mechanism [RFC4120], the Kerberos Distribution Center (KDC) must limit the ticket session key encryption type (encatype) chosen for a given server to one it believes is supported by both the client and the server. If both the client and server understand a stronger encatype than the one selected by the KDC, they cannot negotiate it. As the result, the protection of application traffic is often weaker than necessary when the server can support different sets of encatypes depending on the server application software being used.

This document specifies an extension to the Kerberos protocol to allow clients and servers to negotiate use of a different and possibly stronger cryptosystem in subsequent communication.

This extension utilizes an authorization data element in the authenticator of the AP-REQ message [RFC4120]. The client sends the list of encatypes that it supports to the server; the server then informs the client of its choice. The negotiated subkey is sent in the AP-REP message [RFC4120].

## 2. Conventions Used in This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

## 3. Negotiation Extension

If the client prefers an encatype over that of the service ticket session key, then it SHOULD send a list of encatypes in decreasing preference order to the server. Based on local policy, the client selects encatypes out of all the encatypes available locally to be included in this list, and it SHOULD NOT include encatypes that are less preferable than that of the ticket session key in the service ticket. In addition, the client SHOULD NOT include negative (local-use) encatype numbers unless it knows a priori that the server has been configured to use the same negative encatype numbers for the same encatypes.

The client sends the encatype list via the authorization-data of the authenticator in the AP-REQ [RFC4120]. A new authorization data element type AD-ETYPE-NEGOTIATION is defined.

AD-ETYPE-NEGOTIATION

129

This authorization data element itself is enclosed in the AD-IF-RELEVANT container; thus, a correctly implemented server that does not understand this element should ignore it [RFC4120]. The value of this authorization element contains the DER [X680] [X690] encoding of the following ASN.1 type:

```
EtypeList ::= SEQUENCE OF Int32
  -- Specifies the encypes supported by the client.
  -- This encype list is in decreasing preference order
  -- (favorite choice first).
  -- Int32 is defined in [RFC4120].
```

If the EtypeList is present and the server prefers an encype from the client's encype list over that of the AP-REQ authenticator subkey (if that is present) or the service ticket session key, the server **MUST** create a subkey using that encype. This negotiated subkey is sent in the subkey field of AP-REP message, and it is then used as the protocol key or base key [RFC3961] for subsequent communication.

If the encype of the ticket session key is included in the encype list sent by the client, it **SHOULD** be the last on the list; otherwise, this encype **MUST NOT** be negotiated if it was not included in the list.

This negotiation extension **SHOULD NOT** be used when the client does not expect the subkey in the AP-REP message from the server.

A note on key generation: The KDC has a strong Pseudo-Random Number Generator (PRNG); as such, the client can take advantage of the randomness provided by the KDC by reusing the KDC key data when generating keys. Implementations **SHOULD** use the service ticket session key value as a source of additional entropy when generating the negotiated subkey. If the AP-REQ authenticator subkey is present, it **MAY** also be used as a source of entropy.

The server **MAY** ignore the preference order indicated by the client. The policy by which the client or the server chooses an encype (i.e., how the preference order for the supported encypes is selected) is a local matter.

#### 4. Security Considerations

The client's enctype list and the server's reply enctype are part of encrypted data; thus, the security considerations are the same as those of the Kerberos encrypted data.

Both the EtypeList and the server's sub-session key are protected by the session key or sub-session key used for the AP-REQ, and as a result, if a key for a stronger enctype is negotiated underneath a key for a weaker enctype, an attacker capable of breaking the weaker enctype can also discover the key for the stronger enctype. The advantage of this extension is to minimize the amount of cipher text encrypted under a weak enctype to which an attacker has access.

#### 5. Acknowledgements

The authors would like to thank the following individuals for their comments and suggestions: Ken Raeburn, Luke Howard, Tom Yu, Love Hornquist Astrand, Sam Hartman, and Martin Rex.

#### 6. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3961] Raeburn, K., "Encryption and Checksum Specifications for Kerberos 5", RFC 3961, February 2005.
- [RFC4120] Neuman, C., Yu, T., Hartman, S., and K. Raeburn, "The Kerberos Network Authentication Service (V5)", RFC 4120, July 2005.
- [X680] ITU-T Recommendation X.680 (2002) | ISO/IEC 8824-1:2002, Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation.
- [X690] ITU-T Recommendation X.690 (2002) | ISO/IEC 8825-1:2002, Information technology - ASN.1 encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER).

**Authors' Addresses**

Larry Zhu  
Microsoft Corporation  
One Microsoft Way  
Redmond, WA 98052  
US

EMail: lzhu@microsoft.com

Paul Leach  
Microsoft Corporation  
One Microsoft Way  
Redmond, WA 98052  
US

EMail: paulle@microsoft.com

Karthik Jaganathan  
Microsoft Corporation  
One Microsoft Way  
Redmond, WA 98052  
US

EMail: karthikj@microsoft.com

## Full Copyright Statement

Copyright (C) The Internet Society (2006).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

## Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at [ietf-ipr@ietf.org](mailto:ietf-ipr@ietf.org).

## Acknowledgement

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).