Network Working Group K.W. Burgin
Internet Draft K.M. Igoe
Intended Status: Informational National Security Agency
Expires: December 10, 2011 June 8, 2011

# Suite B in Kerberos 5
draft-burgin-kerberos-suiteb-00

## Abstract

The United States Government has published guidelines for "NSA Suite B Cryptography" dated July, 2005, which defines cryptographic algorithm policy for national security applications. This document specifies the conventions for using Suite B algorithms in the Kerberos 5 protocol specification.

## Status of this Memo



## Copyright Notice

Table of Contents

## 1. Introduction

The Fact Sheet on National Security Agency (NSA) Suite B Cryptography [NSA] states:

> A Cryptographic Interoperability Strategy (CIS) was developed to find ways to increase assured rapid sharing of information both within the U.S. and between the U.S. and her partners through the use of a common suite of public standards, protocols, algorithms and modes referred to as the "Secure Sharing Suite" or S.3 The implementation of CIS will facilitate the development of a broader range of secure cryptographic products which will be available to a wide customer base. The use of selected public cryptographic standards and protocols and Suite B is the core of CIS.
>
> In 2005, NSA announced Suite B Cryptography which built upon the National Policy on the use of the Advanced Encryption Standard (AES) to Protect National Security Systems and National Security Information. In addition to the AES algorithm, Suite B includes cryptographic algorithms for key exchanges, digital signatures and hashing. Suite B cryptography has been selected from cryptography that has been approved by NIST for use by the U.S. Government and specified in NIST standards or recommendations.

This document specifies the use of the United States National Security Agency's Suite B algorithms [NSA] in Kerberos 5. Symmetric key encryption algorithms and checksum types are specified for use in the protocol. Additionally, the use of elliptic curve cryptography in the initial authentication protocol (PKINIT) is specified.

## 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. Suite B Requirements

Suite B requires that key establishment and signature algorithms be based upon Elliptic Curve Cryptography and that the encryption algorithm be AES [FIPS197].





Suite B includes [NSA]:

| | |
|---|---|
| Encryption: | Advanced Encryption Standard (AES) [FIPS197] (key sizes of 128 and 256 bits) |
| Digital Signature: | Elliptic Curve Digital Signature Algorithm (ECDSA) [FIPS186-3] (using the curves with 256- and 384-bit prime moduli) |
| Key Exchange: | Elliptic Curve Diffie-Hellman (ECDH) [SP800-56A] (using the curves with 256- and 384-bit prime moduli) |
| Hashes: | SHA-256 and SHA-384 [FIPS180-3] |

The two elliptic curves used in Suite B each appear in the literature under two different names. For sake of clarity, we list both names below:

| Curve | NIST Name | SECG Name | OID [FIPS186-3] |
|---|---|---|---|
| nistp256 | P-256 | secp256r1 | 1.2.840.10045.3.1.7 |
| nistp384 | P-384 | secp384r1 | 1.3.132.0.34 |

## 4. Minimum Levels of Security (minLOS)

Suite B provides for two levels of cryptographic security, namely a 128-bit minimum level of security (minLOS_128) and a 192-bit minimum level of security (minLOS_192). Each level defines a minimum strength that all cryptographic algorithms must provide.

### 4.1. Non-signature Primitives

We divide the Suite B non-signature primitives into two columns as shown in Table 1.

| | Column 1 | Column 2 |
|---|---|---|
| Encryption | AES-128 | AES-256 |
| Key Agreement | ECDH on P-256 | ECDH on P-384 |
| Hash for PRF/MAC | SHA-256 | SHA-384 |

Table 1: Suite B Cryptographic Non-Signature Primitives

At the 128-bit minimum level of security:

- the non-signature primitives MUST either come exclusively from Column 1 or exclusively from Column 2.





At the 192-bit minimum level of security:

- the non-signature primitives MUST come exclusively from Column 2.

## 4.2. Digital Signatures and Certificates

Digital signatures using ECDSA MUST be used for authentication by Suite B compliant implementations. To simplify notation, ECDSA-256 will be used to represent an instantiation of the ECDSA algorithm using the P-256 curve and the SHA-256 hash function, and ECDSA-384 will be used to represent an instantiation of the ECDSA algorithm using the P-384 curve and the SHA-384 hash function.

If configured at a minimum level of security of 128 bits, a Suite B Kerberos implementation MUST use either ECDSA-256 or ECDSA-384 for authentication. It is allowable for one party to authenticate with ECDSA-256 and the other party to authenticate with ECDSA-384. This flexibility will allow interoperability between a client and a server that have different sizes of ECDSA authentication keys.

Clients and servers in a Suite B Kerberos implementation configured

at a minimum level of security of 128 bits MUST be able to verify ECDSA-256 signatures and SHOULD be able to verify ECDSA-384 signatures unless it is absolutely certain that the implementation will never need to verify certificates from an authority which uses an ECDSA-384 signing key.

If configured at a minimum level of security of 192 bits, ECDSA-384 MUST be used by both parties for authentication.

Clients and servers in a Suite B Kerberos implementation configured at a minimum level of security of 192 bits MUST be able to verify ECDSA-384 signatures.

The client and server, at both minimum levels of security, MUST each have an X.509 certificate that complies with the Suite B End Entity Signature Certificate profile as defined in [RFC5759].

## 5. PKINIT

This section specifies the use of Suite B algorithms for integrating public key cryptography into the initial authentication protocol (PKINIT). The use of public key certificates and signature schemes allows the client and KDC to mutually authenticate in the Authentication Service (AS) request and reply. Furthermore, without PKINIT, the security strength of the AS reply key is usually determined by the strength of the user's password. Using a public key cryptography key exchange eliminates the





dependency of the AS reply key on a password, enhancing the security of the Kerberos protocol.

The original protocol extensions which include public key cryptography are described in PKINIT [RFC4556] and specifications for using elliptic curve cryptography are presented in ECC for PKINIT [RFC5349]. The majority of the conventions needed for Suite B are in those two documents and only the necessary details are provided here.

In Suite B, public key cryptography (PKINIT) MUST be used in the initial authentication protocol to avoid the need for password-based authentication. As defined in [RFC4556], one of the following pre-authentication data elements MUST be included in the AS_REQ and AS_REP messages.

| padata-type | Name | Included in |
|---|---|---|
| 16 | PA_PK_AS_REQ | AS_REQ |
| 17 | PA_PK_AS_REP | AS_REP |

The specific requirements for using ECDH and ECDSA in PKINIT are presented in Sections 5.2 and 5.3 respectively.

## 5.1. Algorithm Agility

PKINIT [RFC4556] has several dependencies on SHA-1 as a checksum algorithm. The first occurrence is the paChecksum field of the PKAuthenticator structure in the authentication request which is defined to contain the SHA-1 checksum of the KDC-REQ-BODY. PKINIT also requires SHA-1 in the key derivation function used to derive the AS reply key from the shared secret value generated by the Diffie-Hellman key exchange. Since Suite B requires SHA-256 or SHA-384 for hashing, the client and KDC need a method to negotiate the hash algorithm used in PKINIT.

[alg-agility] updates PKINIT by allowing the client and KDC to negotiate a KDF from [SP800-56A] which will provide integrity of the request body as well as a cryptographic binding between the client's pre-authentication data and the corresponding request body. This is achieved as described in Section 6 of [alg-agility] by including the AS-REQ and PA-PK-AS-REP messages and the ticket from the KDC in the OtherInfo input parameter to the KDF.

Choosing a KDF from [SP800-56A] that uses SHA-256 or SHA-384 as the hash function therefore eliminates the need for the paChecksum field. In Suite B, the client MUST NOT include the SHA-1 checksum of the KDC-REQ-BODY in the paChecksum field of the authentication request since the KDF will provide the desired





cryptographic binding and integrity protection. The KDC MUST NOT return a KRB-ERROR message due to the absence of the paChecksum field when validating the client's request since the paChecksum field is optional syntactically in [RFC4556]. Section 6 of [alg-agility] describes the new structures and fields included in the AS request and reply messages.

In Suite B, one of the following KDFs defined in [alg-agility] MUST

be used to derive the AS reply key from the Diffie-Hellman shared secret.

| Key Derivation Function | OID [alg-agility] |
|---|---|
| id-pkinit-kdf-ah-sha256 | 1.3.6.1.5.2.3.6.2 |
| id-pkinit-kdf-ah-sha384 | TBD |

## 5.2. ECDH Key Exchange

The use of elliptic curve cryptography in PKINIT is described in [RFC5349]. This section describes the Suite B requirements for using Elliptic Curve Diffie-Hellman (ECDH) to generate the AS reply key.

In Suite B, ephemeral-ephemeral ECDH MUST be used as the AS reply key agreement method. In a Suite B Kerberos system configured at a minimum level of security of 128 bits, ephemeral-ephemeral ECDH MUST be used with the SHA-256 KDF and the P-256 elliptic curve or used with the SHA-384 KDF and the P-384 elliptic curve. In a Suite B Kerberos system configured at a minimum level of security of 192 bits, ephemeral-ephemeral ECDH MUST be used with the SHA-384 KDF and the P-384 elliptic curve. A detailed description of the uses of the ECDH key exchange in PKINIT is provided in [RFC5349].

The client MUST include its encoded ECDH ephemeral public key value and domain parameters in the clientPublicValue field of the AuthPack structure as described in [RFC4556]. The clientPublicValue field MUST comply with the SubjectPublicKeyInfo guidance in [RFC5759] Section 4.4. The domain parameters in the clientPublicValue field MUST be for one of the following curves. Since the curves appear under two different names, both names are listed below.

| Curve | NIST Name | SECG Name | OID [FIPS186-3] |
|---|---|---|---|
| nistp256 | P-256 | secp256r1 | 1.2.840.10045.3.1.7 |
| nistp384 | P-384 | secp384r1 | 1.3.132.0.34 |

A description of the two curves can be found in [FIPS186-3] or [SEC2].





The KDC MUST include its encoded ECDH ephemeral public key value in

the subjectPublicKey field of the KDCDHKeyInfo structure in the authentication reply. Section 2.2 of [RFC5480] provides guidance on the format of the subjectPublicKey field. The KDC MUST NOT reuse its DH keys, even if the client includes the clientDHNonce field. Section 5.6.4.3 of [SP800-56A] states that an ephemeral private key MUST be used in exactly one key establishment transaction, SHOULD be generated as close to its time of use as possible and MUST be zeroized after its use. Section 5.8 of [SP800-56A] states that the Diffie-Hellman shared secret MUST be zeroized immediately after its use. Suite B Kerberos implementations MUST follow the mandates in SP800-56A.

The ECDH shared secret value (Z) is calculated using the ECSVDP-DH primitive described in Section 7.2.1 of [IEEE1363]. Note this primitive is also described in Section 5.7.1.2 of [SP800-56A] under the name ECC CDH.

The AS reply key is derived from the ECDH shared secret value using a negotiated key derivation function from [SP800-56A] with the method described in Section 6 of [alg-agility]. The KDF based on SHA-256 MUST be used when ECDH is used with the 256-bit prime modulus elliptic curve and the KDF based on SHA-384 MUST be used when ECDH is used with the 384-bit prime modulus elliptic curve. Additional guidance on implementing the Ephemeral Unified Model Key Agreement Scheme for Suite B is provided in [IG].

## 5.3. ECDSA and X.509 Certificates

The use of elliptic curve signature schemes in PKINIT is described in [RFC5349]. This section describes the use of digital signatures and certificates that are compatible with Suite B.

The signatureAlgorithm field of the SignerInfo data type in both the AS request and reply messages MUST contain one of the following signature algorithm identifiers:

| Signature Algorithm | OID [FIPS186-3] |
| --- | --- |
| ecdsa-with-Sha256 | 1.2.840.10045.4.3.2 |
| ecdsa-with-Sha384 | 1.2.840.10045.4.3.3 |

If configured at a minimum level of security of 128 bits, a Suite B Kerberos client MUST list one or both of ecdsa-with-sha256 and ecdsa-with-sha384 in the supportedCMSTypes field of the authentication request as the only acceptable signature algorithms for the server's response. If configured at a minimum level of security of 192 bits, a Suite B Kerberos client MUST list ecdsa-with-sha384 in the supportedCMSTypes field of the

authentication request as the only acceptable signature algorithm for the server's response.

The corresponding digestAlgorithm field of the SignerInfo data type MUST contain one of the following hash algorithm identifiers.

| Hash Algorithm | OID [FIPS180-3] |
|---|---|
| id-sha256 | 2.16.840.1.101.3.4.2.2 |
| id-sha384 | 2.16.840.1.101.3.4.2.3 |

id-sha256 MUST be used with ecdsa-with-Sha256 and id-sha384 MUST be used with ecdsa-with-Sha384, as noted in [RFC5349].

## 6. Encryption and Checksum Types

Encryption and checksum types for Kerberos 5 are described in [RFC3961] and specifications for using AES in Kerberos 5 are detailed in [RFC3962]. The dependencies of those types on SHA-1 make them inappropriate choices for Suite B. [AES-CBC-SHA2] defines the encryption and checksum types required by Suite B.

### 6.1. Suite B Requirements

If configured at a minimum level of security of 128 bits, a Suite B Kerberos implementation MUST use either the combination of aes128-cbc-hmac-sha256-128 for content encryption and hmac-sha256-128-aes-128 for message integrity or the combination of aes256-cbc-hmac-sha384-192 for content encryption and hmac-sha384-192-aes256 for message integrity.

If configured at a minimum level of security of 192 bits, a Suite B Kerberos implementation MUST use aes256-cbc-hmac-sha384-192 for content encryption and hmac-sha384-192-aes256 for message integrity.

If the Suite B Kerberos client is using ECDH P-256 for its ephemeral public key in its request, it MUST list aes128-cbc-hmac-sha256-128 in the etype field of the req-body in the initial request message. If the Suite B Kerberos client is using ECDH P-384 for its ephemeral public key in its request, it MUST list aes256-cbc-hmac-sha384-192 in the etype field of the req-body in the initial request message.

## 7. Security Considerations

The security considerations in [RFC4556] discuss PKINIT in general and the security considerations in [RFC5349] discuss the use of

elliptic curve cryptography (ECC).





## 8. IANA Considerations

No IANA considerations.

## 9. References

### 9.1. Normative References

[alg-agility]
Astrand, L. and L. Zhu, "PK-INIT algorithm agility", draft-ietf-krb-wg-pkinit-alg-agility-04, August 2008.

[IEEE1363] IEEE, "Standard Specifications for Public Key Cryptography", IEEE 1363, 2000.

[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.

[RFC3962] Raeburn, K., "Advanced Encryption Standard (AES) Encryption for Kerberos 5", RFC 3962, February 2005.

[RFC4556] Zhu, L. and B. Tung, "Public Key Cryptography for Initial Authentication in Kerberos (PKINIT)", RFC 4556, June 2006.

[RFC5349] Zhu, L., Jaganathan, K. and K. Lauter, "Elliptic Curve Cryptography (ECC) Support for Public Key Cryptography for Initial Authentication in Kerberos (PKINIT)", RFC 5349, September 2008.

[RFC5480] Turner, S., Brown, D., Yiu, K., Housley, R., and T. Polk, "Elliptic Curve Cryptography Subject Public Key Information", RFC 5480, March 2009.

[RFC5759] Solinas, J. and L. Zieglar, "Suite B certificate and Certificate Revocation List (CRL) Profile", RFC 5759,

January 2010.

[FIPS180-3] National Institute of Standards and Technology, "Secure Hash Standard", FIPS PUB 180-3, October 2008.

[FIPS186-3] National Institute of Standards and Technology, "Digital Signature Standard (DSS)", FIPS PUB 186-3, June 2009.





[FIPS197] National Institute of Standards and Technology, "Advanced Encryption Standard (AES)", FIPS PUB 197, November 2001.

## 9.2. Informative References

[IG] U.S. National Security Agency, "Suite B Implementers' Guide to NIST SP 800-56A", July 2009, [http://www.nsa.gov/ia/_files/SuiteB_Implementer_G-113808.pdf].

[NSA] U.S. National Security Agency, "Fact Sheet NSA Suite B Cryptography", January 2009, [http://www.nsa.gov/ia/programs/suiteb_cryptography/].

[SEC2] Standards for Efficient Cryptography Group, "SEC 2 - Recommended Elliptic Curve Domain Parameters", Ver. 1.0, 2000, [http://www.secg.org].

[SP800-56A] National Institute of Standards and Technology, "Recommendation for Pair-wise Key Establishment Schemes Using Discrete Logarithm Cryptography", NIST Special Publication 800-56A, March 2007.

[AES-CBC-SHA2] Burgin, K. and M. Peck, "AES-CBC Mode with HMAC-SHA2 For Kerberos 5", draft-burgin-kerberos-aes-cbc-hmac-sha2-00, (work in progress), June 2011.

## Appendix A. Acknowledgements

The authors would like to thank Mike Peck for his initial work on this document, useful discussions on AES modes and his thorough review of the final draft.

Authors' addresses

Kelley W. Burgin
National Security Agency

EMail: kwburgi@tycho.ncsc.mil

Kevin M. Igoe
NSA/CSS Commercial Solutions Center
National Security Agency

EMail: kmigoe@nsa.gov