Peter Gutmann
University of Auckland

ElGamal Profile for X.509 Certificates
<draft-rfced-info-pgutmann-00.txt>

Status of this Memo

Abstract

This document describes the ASN.1 encoding for an X.509 certificate profiled for use with the ElGamal public key cryptosystem [1]. It is intended to provide guidelines for those developing software that will be used to issue and use ElGamal certificates, and to ensure that ElGamal certificate and key information will be handled consistently throughout the public key infrastructure.

1. ASN.1 Definition of Certificate Elements

The abstract definition of X.509 certificates is given in [2]. The elements specific to ElGamal are the algorithm identifier, the public key information, and the signature data. These are as follows:

```
-- ElGamal may be used in conjunction with the SHA-1 and RIPEMD-160
-- hash algorithms.  The ASN.1 object identifiers used to identify
--- the ElGamal signature algorithm when used with these two hash
-- algorithms is:
```

```
elGamalWithSHA-1 OBJECT IDENTIFIER ::= {
    {iso(1) org(3) dod(6) internet(1) private(4) enterprise(1)
dds(3029)
     signature(3) 1}
    }

elGamalWithRIPEMD-160 OBJECT IDENTIFIER :;= {
    {iso(1) org(3) dod(6) internet(1) private(4) enterprise(1)
dds(3029)
     signature(3) 2}
    }

-- Some of the ElGamal parameters may be shared among a number of
-- users.  These are conveyed in the parameters component of the
-- ElGamal AlgorithmIdentifier, and are as follows:

elGamal-Params ::= SEQUENCE {
    p           INTEGER,
    g           INTEGER
    }

-- The remaining ElGamal parameter is the users public key:

elGamalPublicKey ::= SEQUENCE {
    y           INTEGER,
    }

-- The AlgorithmIdentifier and public key are then encoded into a
-- standard X.509 SubjectPublicKeyInfo:

SubjectPublicKeyInfo ::= SEQUENCE {
    algorithm   AlgorithmIdentifier,
    subjectPublicKey    BIT STRING
    }

-- Prior to the bitstring encoding of an ElGamal signature, the
-- signature components are encoded  as follows:

elGamal-Sig ::= SEQUENCE {
    r           INTEGER,
    s           INTEGER
    }
```

2. Use of ElGamal for Encryption

The ElGamal algorithm may also be used for encryption. In this case the message formatting rules follow the rules for RSA encryption as set
out in PKCS #1 [3], and use a message block type of 01. The object identifier for ElGamal encryption is:

```
elGamalEncryption OBJECT IDENTIFIER ::= {
    {iso(1) org(3) dod(6) internet(1) private(4) enterprise(1)
dds(3029)
     asymmetric-encryption(2) 1}
    }
```

The encrypted message consists of two components, the integers a = g^k mod p and b = My^k mod p (this is not intended as an explanation of the
ElGamal algorithm, but merely to indicate which integer is which). The
encoding of these integers is:

```
elGamalEncryptedMessage ::= SEQUENCE {
    a           INTEGER,
    b           INTEGER
    }
```

Decryption follows the ElGamal algorithm, and the decrypted message is again handled as per PKCS #1.

3. Security considerations

Although the use of the ElGamal algorithm for digital signature generation is not directly addressed in this document, it should be pointed out that some care needs to be taken with both the choice of keys and the use of the algorithm. Details on the safe use of ElGamal are given in [4]. A weakness of ElGamal when used for digital signatures, and workarounds to avoid the weakness, are given in [5].

Ongoing research into the security of ElGamal may reveal other factors which need to be taken into account to provide adequate security for signature and encryption applications, for example it is desirable that
g generate a large subgroup of Zp*; it is recommended that implementors
keep abreast of current research on the choice of parameters and use of
the algorithm in order to avoid potential security weaknesses.

3. References

[1] "A public-key cryptosystem based on discrete logarithms", Taher ElGamal, IEEE Transactions on Information Theory, Vol.31, No.4 (1985), p.469.

[2] ITU Recommendation X.509 (1993).

[3] Public-Key Cryptography Standard #1 (PKCS #1) v1.5, RSA Data Security Inc, November 1993.

[4] "Handbook of Applied Cryptography", Alfred Menezes, Paul van Oorschot, and Scott Vanstone, CRC Press, 1996.

[5] "Generating ElGamal signatures without knowing the secret key", Daniel Bleichenbacher, presented at EuroCrypt'96.

Author's Address

Peter Gutmann
University of Auckland
Private Bag 92019
Auckland
New Zealand

Phone: +64 9 373-7599
Email: pgut001@cs.auckland.ac.nz