Photuris: Extended Schemes and Attributes

Status of this Memo

This document defines an Experimental Protocol for the Internet
community. It does not specify an Internet standard of any kind.
Discussion and suggestions for improvement are requested.
Distribution of this memo is unlimited.

Copyright Notice

Abstract

Photuris is a session-key management protocol. Extensible Exchange-
Schemes are provided to enable future implementation changes without
affecting the basic protocol.

Additional authentication attributes are included for use with the IP
Authentication Header (AH) or the IP Encapsulating Security Protocol
(ESP).

Additional confidentiality attributes are included for use with ESP.

Table of Contents

1. Additional Exchange-Schemes

   The packet format and basic facilities are already defined for
   Photuris [RFC-2522].

   These optional Exchange-Schemes are specified separately, and no
   single implementation is expected to support all of them.

   This document defines the following values:

   (3)   Implementation Optional. Any modulus (p) with a recommended
           generator (g) of 3. When the Exchange-Scheme Size is non-zero,
           the modulus is contained in the Exchange-Scheme Value field in
           the list of Offered-Schemes.

           An Exchange-Scheme Size of zero is invalid.

           Key-Generation-Function   "MD5 Hash"
           Privacy-Method   "Simple Masking"
           Validity-Method   "MD5-IPMAC Check"

           This combination of features requires a modulus with at least
           64-bits of cryptographic strength.

   (4)   Implementation Optional. Any modulus (p) with a recommended
           generator (g) of 2. When the Exchange-Scheme Size is non-zero,
           the modulus is contained in the Exchange-Scheme Value field in
           the list of Offered-Schemes.

           When the Exchange-Scheme Size field is zero, includes by
           reference all of the moduli specified in the list of Offered-
           Schemes for Scheme #2.

           Key-Generation-Function   "MD5 Hash"
           Privacy-Method   "DES-CBC over Mask"
           Validity-Method   "MD5-IPMAC Check"

           This combination of features requires a modulus with at least
           64-bits of cryptographic strength.

   (5)   Implementation Optional. Any modulus (p) with a recommended
           generator (g) of 5. When the Exchange-Scheme Size is non-zero,
           the modulus is contained in the Exchange-Scheme Value field in
           the list of Offered-Schemes.

           An Exchange-Scheme Size of zero is invalid.

```
                Key-Generation-Function    "MD5 Hash"
                Privacy-Method             "Simple Masking"
                Validity-Method            "MD5-IPMAC Check"
```

            This combination of features requires a modulus with at least
            64-bits of cryptographic strength.

   (6)   Implementation Optional. Any modulus (p) with a recommended
            generator (g) of 3. When the Exchange-Scheme Size is non-zero,
            the modulus is contained in the Exchange-Scheme Value field in
            the list of Offered-Schemes.

            When the Exchange-Scheme Size field is zero, includes by
            reference all of the moduli specified in the list of Offered-
            Schemes for Scheme #3.

```
                Key-Generation-Function    "MD5 Hash"
                Privacy-Method             "DES-CBC over Mask"
                Validity-Method            "MD5-IPMAC Check"
```

            This combination of features requires a modulus with at least
            64-bits of cryptographic strength.

   (7)   Implementation Optional. Any modulus (p) with a variable
            generator (g). When the Exchange-Scheme Size is non-zero, the
            pair [g,p] is contained in the Exchange-Scheme Value field in
            the list of Offered-Schemes. Each is encoded in a separate
            Variable Precision Integer (VPI). The generator VPI is
            followed by (concatenated to) the modulus VPI, and the result
            is nested inside the Exchange-Scheme Value field.

            An Exchange-Scheme Size of zero is invalid.

```
                Key-Generation-Function    "MD5 Hash"
                Privacy-Method             "Simple Masking"
                Validity-Method            "MD5-IPMAC Check"
```

            This combination of features requires a modulus with at least
            64-bits of cryptographic strength.

            When more than one modulus is specified for a given kind of
            Scheme, the Size of the modulus MUST be unique, independent of
            the Size of the generator.

   (8)   Implementation Optional. Any modulus (p) with a recommended
            generator (g) of 2. When the Exchange-Scheme Size is non-zero,
            the modulus is contained in the Exchange-Scheme Value field in

the list of Offered-Schemes.

When the Exchange-Scheme Size field is zero, includes by reference all of the moduli specified in the list of Offered-Schemes for Schemes #2 and #4.

```
Key-Generation-Function       "SHA1 Hash"
Privacy-Method                "DES-EDE3-CBC over Mask"
Validity-Method               "SHA1-IPMAC Check"
```

This combination of features requires a modulus with at least 112-bits of cryptographic strength.

(10) Implementation Optional. Any modulus (p) with a recommended generator (g) of 5. When the Exchange-Scheme Size is non-zero, the modulus is contained in the Exchange-Scheme Value field in the list of Offered-Schemes.

When the Exchange-Scheme Size field is zero, includes by reference all of the moduli specified in the list of Offered-Schemes for Scheme #5.

```
Key-Generation-Function       "MD5 Hash"
Privacy-Method                "DES-CBC over Mask"
Validity-Method               "MD5-IPMAC Check"
```

This combination of features requires a modulus with at least 64-bits of cryptographic strength.

(12) Implementation Optional. Any modulus (p) with a recommended generator (g) of 3. When the Exchange-Scheme Size is non-zero, the modulus is contained in the Exchange-Scheme Value field in the list of Offered-Schemes.

When the Exchange-Scheme Size field is zero, includes by reference all of the moduli specified in the list of Offered-Schemes for Schemes #3 and #6.

```
Key-Generation-Function       "SHA1 Hash"
Privacy-Method                "DES-EDE3-CBC over Mask"
Validity-Method               "SHA1-IPMAC Check"
```

This combination of features requires a modulus with at least 112-bits of cryptographic strength.

(14) Implementation Optional. Any modulus (p) with a variable generator (g). When the Exchange-Scheme Size is non-zero, the pair [g,p] is contained in the Exchange-Scheme Value field in

the list of Offered-Schemes. Each is encoded in a separate
Variable Precision Integer (VPI). The generator VPI is
followed by (concatenated to) the modulus VPI, and the result
is nested inside the Exchange-Scheme Value field.

When the Exchange-Scheme Size field is zero, includes by
reference all of the moduli specified in the list of Offered-
Schemes for Scheme #7.

```
Key-Generation-Function      "MD5 Hash"
Privacy-Method               "DES-CBC over Mask"
Validity-Method              "MD5-IPMAC Check"
```

This combination of features requires a modulus with at least
64-bits of cryptographic strength.

When more than one modulus is specified for a given kind of
Scheme, the Size of the modulus MUST be unique, independent of
the Size of the generator.

(20)   Implementation Optional. Any modulus (p) with a recommended
       generator (g) of 5. When the Exchange-Scheme Size is non-zero,
       the modulus is contained in the Exchange-Scheme Value field in
       the list of Offered-Schemes.

       When the Exchange-Scheme Size field is zero, includes by
       reference all of the moduli specified in the list of Offered-
       Schemes for Schemes #5 and #10.

```
Key-Generation-Function      "SHA1 Hash"
Privacy-Method               "DES-EDE3-CBC over Mask"
Validity-Method              "SHA1-IPMAC Check"
```

       This combination of features requires a modulus with at least
       112-bits of cryptographic strength.

(28)   Implementation Optional. Any modulus (p) with a variable
       generator (g). When the Exchange-Scheme Size is non-zero, the
       pair [g,p] is contained in the Exchange-Scheme Value field in
       the list of Offered-Schemes. Each is encoded in a separate
       Variable Precision Integer (VPI). The generator VPI is
       followed by (concatenated to) the modulus VPI, and the result
       is nested inside the Exchange-Scheme Value field.

       When the Exchange-Scheme Size field is zero, includes by
       reference all of the moduli specified in the list of Offered-
       Schemes for Schemes #7 and #14.

```
      Key-Generation-Function       "SHA1 Hash"
      Privacy-Method                "DES-EDE3-CBC over Mask"
      Validity-Method               "SHA1-IPMAC Check"
```

This combination of features requires a modulus with at least 112-bits of cryptographic strength.

When more than one modulus is specified for a given kind of Scheme, the Size of the modulus MUST be unique, independent of the Size of the generator.

2.   Additional Key-Generation-Function
2.1. SHA1 Hash

SHA1 [FIPS-180-1] is used as a pseudo-random-function for generating the key(s). The key(s) begin with the most significant bits of the hash. SHA1 is iterated as needed to generate the requisite length of key material.

When an individual key does not use all 160-bits of the last hash, any remaining unused (least significant) bits of the last hash are discarded. When combined with other uses of key generation for the same purpose, the next key will begin with a new hash iteration.

3.   Additional Privacy-Methods
3.1. DES-CBC over Mask

As described in [RFC-2522] "Privacy-Key Computation", sufficient privacy-key material is generated to match the message length, beginning with the next field after the SPI, and including the Padding. The message is masked by XOR with the privacy-key.

Then, the Key-Generation-Function is iterated to generate a DES key. The most significant 64-bits (8 bytes) of the generated hash are used for the privacy-key, and the remainder are discarded. Although extremely rare, the 64 weak, semi-weak, and possibly weak keys [Schneier95, pages 280-282] are discarded. The Key-Generation-Function is iterated until a valid key is obtained.

The least significant bit of each key byte is ignored (or set to parity when the implementation requires).

The 64-bit CBC IV is zero. Message encryption begins with the next field after the SPI, and continues to the end of the data indicated

   by the UDP Length.

3.2. DES-EDE3-CBC over Mask

   This is "Triple DES" outer-CBC EDE encryption (and DED decryption)
   with three 56-bit keys [KR96].

   As described in [RFC-2522] "Privacy-Key Computation", sufficient
   privacy-key material is generated to match the message length,
   beginning with the next field after the SPI, and including the
   Padding. The message is masked by XOR with the privacy-key.

   Then, the Key-Generation-Function is iterated (at least) three times
   to generate the three DES keys. The most significant 64-bits (8
   bytes) of each generated hash are used for each successive privacy-
   key, and the remainder are discarded. Each key is examined
   sequentially, in the order used for encryption. A key that is
   identical to a previous key MUST be discarded. Although extremely
   rare, the 64 weak, semi-weak, and possibly weak keys [Schneier95,
   pages 280-282] MUST be discarded. The Key-Generation-Function is
   iterated until a valid key is obtained before generating the next
   key.

   In all three keys, the least significant bit of each key byte is
   ignored (or set to parity when the implementation requires).

   The 64-bit CBC IV is zero. Message encryption begins with the next
   field after the SPI, and continues to the end of the data indicated
   by the UDP Length.

4. Additional Validity-Method
4.1. SHA1-IPMAC Check

   As described in [RFC-2522] "Validity Verification", the Verification
   field value is the SHA1 [FIPS-180-1] hash over the concatenation of

      SHA1(key, keyfill, data, datafill, key, mdfill)

   where the key is the computed verification-key.

   The keyfill and datafill use the same pad-with-length technique
   defined for mdfill. This padding and length is implicit, and does
   not appear in the datagram.

   The resulting Verification field is a 160-bit Variable Precision
   Integer (22 bytes including Size). When used in calculations, the

    Verification data includes both the Size and Value fields.

5. Additional Attributes

    The attribute format and basic facilities are already defined for
    Photuris [RFC-2522].

    These optional attributes are specified separately, and no single
    implementation is expected to support all of them.

    This document defines the following values:

       Use   Type
       AEI   6   SHA1-IPMAC
       AEI   7   RIPEMD-160-IPMAC
        E   8   DES-CBC
        E   9   Invert (Decryption/Encryption)
        E   10   XOR

        A   AH Attribute-Choice
         E   ESP Attribute-Choice
          I   Identity-Choice
           X   dependent on list location

5.1. SHA1-IPMAC

    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | Attribute | Length |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

    Attribute   6

    Length   0

5.1.1. Symmetric Identification

   When selected as an Identity-Choice, the immediately following
   Identification field contains an unstructured Variable Precision
   Integer. Valid Identifications and symmetric secret-keys are
   preconfigured by the parties.

   There is no required format or content for the Identification value.
   The value may be a number or string of any kind. See [RFC-2522] "Use
   of Identification and Secrets" for details.

   The symmetric secret-key (as specified) is selected based on the
   contents of the Identification field. All implementations MUST
   support at least 62 bytes. The selected symmetric secret-key SHOULD
   provide at least 80-bits of cryptographic strength.

   As described in [RFC-2522] "Identity Verification", the Verification
   field value is the SHA1 [FIPS-180-1] hash over the concatenation of:

      SHA1(key, keyfill, data, datafill, key, mdfill)

   where the key is the computed verification-key.

   The keyfill and datafill use the same pad-with-length technique
   defined for mdfill. This padding and length is implicit, and does
   not appear in the datagram.

   The resulting Verification field is a 160-bit Variable Precision
   Integer (22 bytes including Size). When used in calculations, the
   Verification data includes both the Size and Value fields.

   For both [RFC-2522] "Identity Verification" and "Validity
   Verification", the verification-key is the SHA1 [FIPS-180-1] hash of
   the following concatenated values:

    + the symmetric secret-key,
    + the computed shared-secret.

   For [RFC-2522] "Session-Key Computation", the symmetric secret-key is
   used directly as the generation-key.

   The symmetric secret-key is used in calculations in the same fashion
   as [RFC-2522] "MD5-IPMAC Symmetric Identification".

5.1.2. Authentication

   May be selected as an AH or ESP Attribute-Choice, pursuant to [RFC-
   1852] et sequitur. The selected Exchange-Scheme SHOULD provide at
   least 80-bits of cryptographic strength.

   As described in [RFC-2522] "Session-Key Computation", the most
   significant 384-bits (48 bytes) of the Key-Generation-Function
   iterations are used for the key.

   Profile:

      When negotiated with Photuris, the transform differs slightly from
      [RFC-1852].

      The form of the authenticated message is:

         SHA1(key, keyfill, datagram, datafill, key, mdfill)

      where the key is the SPI session-key.

      The additional datafill protects against the attack described in
      [PO96]. The keyfill and datafill use the same pad-with-length
      technique defined for mdfill. This padding and length is
      implicit, and does not appear in the datagram.

5.2. RIPEMD-160-IPMAC

   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | Attribute | Length |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

   Attribute   7

   Length   0

5.2.1. Symmetric Identification

   When selected as an Identity-Choice, the immediately following
   Identification field contains an unstructured Variable Precision
   Integer. Valid Identifications and symmetric secret-keys are
   preconfigured by the parties.

   There is no required format or content for the Identification value.
   The value may be a number or string of any kind. See [RFC-2522] "Use
   of Identification and Secrets" for details.

   The symmetric secret-key (as specified) is selected based on the
   contents of the Identification field. All implementations MUST
   support at least 62 bytes. The selected symmetric secret-key SHOULD
   provide at least 80-bits of cryptographic strength.

   As described in [RFC-2522] "Identity Verification", the Verification
   field value is the RIPEMD-160 [DBP96] hash over the concatenation of:

       RIPEMD160(key, keyfill, data, datafill, key, mdfill)

   where the key is the computed verification-key.

   The keyfill and datafill use the same pad-with-length technique
   defined for mdfill. This padding and length is implicit, and does
   not appear in the datagram.

   The resulting Verification field is a 160-bit Variable Precision
   Integer (22 bytes including Size). When used in calculations, the
   Verification data includes both the Size and Value fields.

   For both [RFC-2522] "Identity Verification" and "Validity
   Verification", the verification-key is the RIPEMD-160 [DBP96] hash of
   the following concatenated values:

    + the symmetric secret-key,
    + the computed shared-secret.

   For [RFC-2522] "Session-Key Computation", the symmetric secret-key is
   used directly as the generation-key.

   The symmetric secret-key is used in calculations in the same fashion
   as [RFC-2522] "MD5-IPMAC Symmetric Identification".

5.2.2. Authentication

   May be selected as an AH or ESP Attribute-Choice. The selected
   Exchange-Scheme SHOULD provide at least 80-bits of cryptographic
   strength.

   As described in [RFC-2522] "Session-Key Computation", the most
   significant 384-bits (48 bytes) of the Key-Generation-Function
   iterations are used for the key.

   Profile:

      When negotiated with Photuris, the form of the authenticated
      message is:

         RIPEMD160(key, keyfill, datagram, datafill, key, mdfill)

      where the key is the SPI session-key.

      The additional datafill protects against the attack described in
      [PO96]. The keyfill and datafill use the same pad-with-length
      technique defined for mdfill. This padding and length is
      implicit, and does not appear in the datagram.

5.3. DES-CBC

```
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   Attribute   |    Length     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   Attribute   8

   Length   0

   May be selected as an ESP Attribute-Choice, pursuant to [RFC-1829] et
   sequitur. The selected Exchange-Scheme SHOULD provide at least 56-
   bits of cryptographic strength.

   As described in [RFC-2522] "Session-Key Computation", the most
   significant 64-bits (8 bytes) of the Key-Generation iteration are
   used for the key, and the remainder are discarded. Although
   extremely rare, the 64 weak, semi-weak, and possibly weak keys
   [Schneier95, pages 280-282] MUST be discarded. The Key-Generation-
   Function is iterated until a valid key is obtained.

   The least significant bit of each key byte is ignored (or set to

parity when the implementation requires).

Profile:

When negotiated with Photuris, the transform differs slightly from [RFC-1829].

The 32-bit Security Parameters Index (SPI) field is followed by a 32-bit Sequence Number (SN).

The 64-bit CBC IV is generated from the 32-bit Security Parameters Index (SPI) field followed by (concatenated with) the 32-bit Sequence Number (SN) field. Then, the bit-wise complement of the 32-bit Sequence Number (SN) value is XOR'd with the first 32-bits (SPI):

   (SPI ^ -SN) || SN

The Padding values begin with the value 1, and count up to the number of padding bytes. For example, if the plaintext length is 41, the padding values are 1, 2, 3, 4, 5, 6 and 7, plus any additional obscuring padding.

The PadLength and PayloadType are not appended. Instead, the PayloadType is indicated by the SPI, as specified by the ESP-Attributes attribute (#2).

After decryption, if the padding bytes are not the correct sequential values, then the payload is discarded, and a "Decryption Failed" error is indicated, as described in [RFC-2521].

5.4. Invert (Decryption/Encryption)

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Attribute   |    Length     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Attribute   9

Length   0

May be selected as an ESP Attribute-Choice, immediately preceding an encryption choice. This indicates that the following attribute is inverted from encryption to decryption (or decryption to encryption) as the attributes are processed.

For example, the combination

    "DES-CBC",
    "Invert",
    "DES-CBC",
    "DES-CBC",

indicates "Triple DES" outer-CBC EDE encryption (and DED decryption)
with three keys [KR96] pursuant to [RFC-1851] et sequitur. The
selected Exchange-Scheme SHOULD provide at least 112-bits of
cryptographic strength.

As described in [RFC-2522] "Session-Key Computation", the Key-
Generation-Function is iterated (at least) three times to generate
the three independent keys, in the order used for encryption. The
most significant 64-bits (8 bytes) of each iteration are used for
each successive key, and the remainder are discarded.

Each key is examined sequentially, in the order used for encryption.
A key that is identical to any previous key MUST be discarded. Any
weak keys indicated for the algorithm MUST be discarded. The Key-
Generation-Function is iterated until a valid key is obtained before
generating the next key.

Profile:

    When negotiated with Photuris, the "DES-EDE3-CBC" transform
    differs slightly from [RFC-1851], in the same fashion as "DES-CBC"
    (described earlier).

5.5. XOR Whitening

    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | Attribute | Length |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

    Attribute   10

    Length   0

    May be selected as an ESP Attribute-Choice, pursuant to [XEX3] et
    sequitur. The combination

        "XOR",
        "DES-CBC",
        "XOR",

indicates "DESX" encryption with three keys [KR96]. The selected
Exchange-Scheme SHOULD provide at least 104-bits of cryptographic
strength.

As described in [RFC-2522] "Session-Key Computation", the Key-
Generation-Function is iterated (at least) three times to generate
the three independent keys, in the order used for encryption. The
most significant bytes of each iteration are used for each successive
key, and the remainder are discarded.

Note that this attribute may appear multiple times in the same ESP
attribute list, both before and after an encryption transform. For
example,

    "XOR",
    "DES-CBC",
    "XOR",
    "Invert",
    "DES-CBC",
    "XOR",
    "DES-CBC",
    "XOR",

would be one possible combination with Triple DES.

A. Exchange-Scheme Selection

   At first glance, there appear to be a large number of exchange-
   schemes. In practice, the selection is simple to automate.

   Each scheme indicates a needed strength. This strength is based upon
   the functions used in protecting the Photuris Exchanges themselves.

   Each keyed attribute also indicates a needed strength. This strength
   is based upon its cryptographic functions.

   Because the usage of these functions is orthogonal, the same strength
   value can select an appropriate scheme that meets the needs of both
   features.

A.1. Responder

   The attributes to be offered to the particular Initiator are
   examined. For each level of strength specified, a scheme that meets
   or exceeds the requirements is offered.

   For example, a Responder offering MD5-IPMAC and SHA1-IPMAC might
   offer scheme #2 with a 512-bit modulus and a 1024-bit modulus, and
   scheme #4 with a zero Size (indicating moduli of #2).

A.2. Initiator

   The strength indicated by the application for the Security
   Association, together with the party privacy policy of the system
   operator, is used to select from the offered schemes. The strength
   indicates the minimal level to be chosen, while the party privacy
   policy indicates whether to choose the minimal or maximal level of
   available protection.

   For example, an application might indicate that it desires 80-bits of
   strength. In that case, only the 1024-bit modulus would be
   appropriate. The party privacy policy of the system operator would
   indicate whether to choose scheme #2 with "Simple Masking" or scheme
   #4 with "DES-CBC over Mask".

   Alternatively, an application might indicate that it desires 64-bits
   of strength. The party privacy policy of the system operator would
   indicate whether to choose scheme #2 with the 512-bit modulus, or
   scheme #4 with the 1024-bit modulus.

Security Considerations

Provision for multiple generators does not enhance the security of the Photuris protocol exchange itself. Rather, it provides an opportunity for novelty of moduli, by allowing more forms of moduli to be used. An abundance of moduli inhibits a determined attacker from pre-calculating moduli exchange values, and discourages dedication of resources for analysis of any particular modulus. That is, this protects the community of Photuris users.

In addition to preventing various attacks by protecting verification fields, the masking of the message plaintext before encryption is intended to obscure the relation of the number of parties and SPIs active between two IP nodes. The privacy mask dependency on the SPI and SPILT generates a different initial encrypted block for every SPI creation message.

This obscurement would be less effective when the SPI and SPILT are invariant or are not created for a particular exchange direction. The number of parties could be revealed by the number of exchanges with differences in the initial encrypted blocks.

Acknowledgements

Phil Karn was principally responsible for the design of party privacy protection, and provided much of the design rationale text (now removed to a separate document).

William Simpson was responsible for the packet formats, and additional Exchange-Schemes, editing and formatting. All such mistakes are his responsibity.

Use of encryption for privacy protection is also found in the Station-To-Station authentication protocol [DOW92].

Bart Preneel and Paul C van Oorschot in [PO96] recommended padding between the data and trailing key when hashing for authentication.

Niels Provos developed the first implementation with multiple schemes and multiple moduli per scheme (circa July 1997).

Special thanks to the Center for Information Technology Integration (CITI) for providing computing resources.

References

[DBP96]   Dobbertin, H., Bosselaers, A., and Preneel, B., "RIPEMD-
             160: a strengthened version of RIPEMD", Fast Software
             Encryption, Third International Workshop, Lecture Notes
             in Computer Science 1039 (1996), Springer-Verlag, pages
             71-82.

             See also corrections at
             ftp://ftp.esat.kuleuven.ac.be/pub/COSIC/bosselae/ripemd/.

[DOW92]   Whitfield Diffie, Paul C van Oorshot, and Michael J
             Wiener, "Authentication and Authenticated Key Exchanges",
             Designs, Codes and Cryptography, v 2 pp 107-125, Kluwer
             Academic Publishers, 1992.

[FIPS-180-1]
             "Secure Hash Standard", National Institute of Standards
             and Technology, U.S. Department Of Commerce, April 1995.

             Also known as: 59 Fed Reg 35317 (1994).

[KR96]   Kaliski, B., and Robshaw, M., "Multiple Encryption:
             Weighing Security and Performance", Dr. Dobbs Journal,
             January 1996.

[PO96]   Bart Preneel, and Paul C van Oorshot, "On the security of
             two MAC algorithms", Advances in Cryptology -- Eurocrypt
             '96, Lecture Notes in Computer Science 1070 (May 1996),
             Springer-Verlag, pages 19-32.

[RFC-1829]   Karn, P., Metzger, P., Simpson, W., "The ESP DES-CBC
             Transform", July 1995.

[RFC-1850]   Karn, P., Metzger, P., Simpson, W., "The ESP Triple DES
             Transform", September 1995.

[RFC-1851]   Metzger, P., Simpson, W., "IP Authentication using Keyed
             SHA", September 1995.

[RFC-2521]   Karn, P., and Simpson, W., "ICMP Security Failures
             Messages", March 1999.

[RFC-2522]   Karn, P., and Simpson, W., "Photuris: Session-Key
             Management Protocol", March 1999.

[XEX3]   Simpson, W., Baldwin, R., "The ESP DES-XEX3-CBC
             Transform", Work In Progress, June 1997.

Contacts

    Comments about this document should be discussed on the
    photuris@adk.gr mailing list.

    Questions about this document can also be directed to:

        Phil Karn
        Qualcomm, Inc.
        6455 Lusk Blvd.
        San Diego, California 92121-2779

            karn@qualcomm.com
            karn@unix.ka9q.ampr.org (preferred)

        William Allen Simpson
        DayDreamer
        Computer Systems Consulting Services
        1384 Fontaine
        Madison Heights, Michigan 48071

            wsimpson@UMich.edu
            wsimpson@GreenDragon.com (preferred)

Full Copyright Statement