                   Additional Diffie-Hellman Tests
        for the Internet Key Exchange Protocol Version 2 (IKEv2)

Abstract

   This document adds a small number of mandatory tests required for the
   secure operation of the Internet Key Exchange Protocol version 2
   (IKEv2) with elliptic curve groups. No change is required to IKE
   implementations that use modular exponential groups, other than a few
   rarely used so-called Digital Signature Algorithm (DSA) groups. This
   document updates the IKEv2 protocol, RFC 5996.

Status of This Memo

   This is an Internet Standards Track document.

   This document is a product of the Internet Engineering Task Force
   (IETF). It represents the consensus of the IETF community. It has
   received public review and has been approved for publication by the
   Internet Engineering Steering Group (IESG). Further information on
   Internet Standards is available in Section 2 of RFC 5741.

   Information about the current status of this document, any errata,
   and how to provide feedback on it may be obtained at
   http://www.rfc-editor.org/info/rfc6989.

Table of Contents

1. Introduction

   IKEv2 [RFC5996] consists of the establishment of a shared secret
   using the Diffie-Hellman (DH) protocol, followed by authentication of
   the two peers. Existing implementations typically use modular
   exponential (MODP) DH groups, such as those defined in [RFC3526].

   IKEv2 does not require that any tests be performed by a peer
   receiving a public Diffie-Hellman key from the other peer. This is
   fine for the common case of MODP groups. For other DH groups, when
   peers reuse DH values across multiple IKE sessions, the lack of tests
   by the recipient results in a potential vulnerability (see
   Section 4.1 for more details). In particular, this is true for
   Elliptic Curve (EC) groups, whose use is becoming ever more popular.
   This document defines such tests for several types of DH groups.

   In addition, this document describes another potential attack related
   to the reuse of DH keys: a timing attack. This additional material
   is taken from [RFC2412].

   This document updates [RFC5996] by adding security requirements that
   apply to many of the protocol's implementations.

1.1. 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].

2. Group Membership Tests

   This section describes the tests that need to be performed by IKE
   peers receiving a Key Exchange (KE) payload. The tests are
   RECOMMENDED for all implementations but only REQUIRED for those that
   reuse DH private keys (as defined in [RFC5996], Section 2.12). The
   tests apply to the recipient of a KE payload and describe how it
   should check the received payload. They are listed here according to
   the DH group being used.

2.1. Sophie Germain Prime MODP Groups

   These are currently the most commonly used groups; all these groups
   have the property that (p-1)/2 is also prime; this section applies to
   any such MODP group. Each recipient MUST verify that the peer's
   public value r is in the legal range (1 < r < p-1). According to
   [Menezes], Section 2.2, even with this check there remains the
   possibility of leaking a single bit of the secret exponent when DH
   keys are reused; this amount of leakage is insignificant.

   See Section 5 for the specific groups covered by this section.

2.2. MODP Groups with Small Subgroups

   [RFC5114] defines modular exponential groups with small subgroups;
   these are modular exponential groups with comparatively small
   subgroups, and all have (p-1)/2 composite. Section 2.1 of [Menezes]
   describes some informational leakage from a small-subgroup attack on
   these groups if the DH private value is reused.

   This leakage can be prevented if the recipient performs a test on the
   peer's public value; however, this test is expensive (approximately
   as expensive as what reusing DH private values saves). In addition,
   the NIST standard ([NIST-800-56A], Section 5.6.2.4) requires that
   test; hence, anyone needing to conform to that standard will need to
   implement the test anyway.

   Because of the above, the IKE implementation MUST choose between one
   of the following two options:

   o It MUST check both that the peer's public value is in range (1 < r
      < p-1) and that r^q = 1 mod p (where q is the size of the
      subgroup, as listed in the RFC defining the group). DH private
      values MAY then be reused. This option is appropriate if
      conformance to [NIST-800-56A] is required.

   o It MUST NOT reuse DH private values (that is, the DH private value
      for each DH exchange MUST be generated from a fresh output of a
      cryptographically secure random number generator), and it MUST
      check that the peer's public value is in range (1 < r < p-1).
      This option is more appropriate if conformance to [NIST-800-56A]
      is not required.

   See Section 5 for the specific groups covered by this section.

2.3. Elliptic Curve Groups

   IKEv2 can be used with elliptic curve groups defined over a field
   GF(p) [RFC5903] [RFC5114]. According to [Menezes], Section 2.3,
   there is some informational leakage possible. A receiving peer MUST
   check that its peer's public value is valid; that is, the x and y
   parameters from the peer's public value satisfy the curve equation,
   $y^2 = x^3 + ax + b \bmod p$ (where for groups 19, 20, and 21, a=-3 (mod
   p), and all other values of a, b, and p for the group are listed in
   the RFC defining the group).

   We note that an additional check to ensure that the public value is
   not the point at infinity is not needed, because IKE (see Section 7
   of [RFC5903]) does not allow for encoding this value.

   See Section 5 for the specific groups covered by this section.

2.4. Transition

   Existing implementations of IKEv2 with Elliptic Curve Diffie-Hellman
   (ECDH) groups may be modified to include the tests described in the
   current document, even if they do not reuse DH keys. The tests can
   be considered as sanity checks and will prevent the code having to
   handle inputs that it may not have been designed to handle.

   ECDH implementations that do reuse DH keys MUST be enhanced to
   include the above tests.

2.5. Protocol Behavior

   The recipient of a DH public key that fails one of the above tests
   must assume that the sender is either truly malicious or has a bug in
   its implementation. The behavior defined below attempts to balance
   resistance to attackers that are trying to disrupt the IKE exchange,
   against the need to help a badly implemented peer by providing useful
   error indications.

   If this error happens during the IKE_SA_INIT exchange, then the
   recipient MUST drop the message that contains an invalid KE payload
   and MUST NOT use that message when creating the IKE security
   association (SA).

   If the implementation employs the DoS-resistant behavior proposed in
   Section 2.4 of [RFC5996], it may simply ignore the erroneous request
   or response message, and continue waiting for a later message
   containing a legitimate KE payload.

   If DoS-resistant behavior is not implemented and the invalid KE
   payload was in the IKE_SA_INIT request, the implementation MAY send
   an INVALID_SYNTAX error notification back and remove the in-progress
   IKE SA; if the invalid KE payload was in the IKE_SA_INIT response,
   then the implementation MAY simply delete the half-created IKE SA and
   re-initiate the exchange.

   If the invalid KE payload is received during the CREATE_CHILD_SA
   exchange (or any other exchange after the IKE SA has been
   established) and the invalid KE payload is in the request message,
   the Responder MUST reply with an INVALID_SYNTAX error notification
   and drop the IKE SA. If the invalid KE payload is in a response, the
   Initiator getting this reply MUST immediately delete the IKE SA by
   sending an IKE SA Delete notification as a new exchange. In this
   case, the sender evidently has an implementation bug, and dropping
   the IKE SA makes it easier to detect.

3. Side-Channel Attacks

   In addition to the small-subgroup attack, there is also a potential
   timing attack on IKE peers when they are reusing Diffie-Hellman
   secret values. This is a side-channel attack, which means that it
   may or may not be a vulnerability in certain cases, depending on
   implementation details and the threat model.

The remainder of this section is quoted from [RFC2412], Section 5, with a few minor clarifications. This attack still applies to IKEv2 implementations, and both to MODP groups and ECDH groups. We also note that more efficient countermeasures are available for EC groups represented in projective form, but these are outside the scope of the current document.

Timing attacks that are capable of recovering the exponent value used in Diffie-Hellman calculations have been described by Paul Kocher [Kocher]. In order to nullify the attack, implementors must take pains to obscure the sequence of operations involved in carrying out modular exponentiations.

One potential method to foil these timing attacks is to use a "blinding factor". In this method, a group element, r, is chosen at random, and its multiplicative inverse modulo p is computed, which we'll call r_inv. r_inv can be computed by the Extended Euclidean Method, using r and p as inputs. When an exponent x is chosen, the value r_inv^x is also calculated. Then, when calculating (g^y)^x, the implementation will calculate this sequence:

```
A = r*g^y
B = A^x = (r*g^y)^x = (r^x)(g^(xy))
C = B*r_inv^x = (r^x)(r^(-1*x))(g^(xy)) = g^(xy)
```

The blinding factor is only necessary if the exponent x is used more than 100 times.

4. Security Considerations

   This entire document is concerned with the IKEv2 security protocol and the need to harden it in some cases.

4.1. DH Key Reuse and Multiple Peers

   This section describes one variant of the attack prevented by the tests defined above.

   Suppose that IKE peer Alice maintains IKE security associations with peers Bob and Eve. Alice uses the same secret ECDH key for both SAs, which is allowed with some restrictions. If Alice does not implement these tests, Eve will be able to send a malformed public key, which would allow her to efficiently determine Alice's private key (as described in Section 2 of [Menezes]). Since the key is shared, Eve will be able to obtain Alice's shared IKE SA key with Bob.

4.2. DH Key Reuse: Variants

   Private DH keys can be reused in different ways, with subtly
   different security implications. For example:

   1. DH keys are reused for multiple connections (IKE SAs) to the same
       peer and for connections to different peers.

   2. DH keys are reused for multiple connections to the same peer
       (e.g., when the peer is identified by its IP address) but not for
       different peers.

   3. DH keys are reused only when they had not been used to complete
       an exchange, e.g., when the peer replies with an
       INVALID_KE_PAYLOAD notification.

   Both the small-subgroup attack and the timing attack described in
   this document apply at least to options #1 and #2.

4.3. Groups Not Covered by This RFC

   There are a number of group types that are not specifically addressed
   by this RFC. A document that defines such a group MUST describe the
   tests required by that group.

   One specific type of group would be an even-characteristic elliptic
   curve group. Now, these curves have cofactors greater than 1; this
   leads to a possibility of some information leakage. There are
   several ways to address this information leakage, such as performing
   a test analogous to the test in Section 2.2 or adjusting the ECDH
   operation to avoid this leakage (such as Elliptic Curve Cryptography
   Cofactor Diffie-Hellman (ECC CDH), where the shared secret really is
   hxyG). Because the appropriate test depends on how the group is
   defined, we cannot document it in advance.

4.4. Behavior upon Test Failure

   The behavior recommended in Section 2.5 is in line with generic error
   treatment during the IKE_SA_INIT exchange, per Section 2.21.1 of
   [RFC5996]. The sender is not required to send back an error
   notification, and the recipient cannot depend on this notification
   because it is unauthenticated and may in fact have been sent by an
   attacker trying to launch a DoS attack on the connection. Thus, the
   notification is only useful to debug implementation errors.

On the other hand, the error notification is secure in the sense that
no secret information is leaked. All IKEv2 Diffie-Hellman groups are
publicly known, and none of the tests defined here depend on any
private key. In fact, the tests can all be performed by an
eavesdropper.

The situation when the failure occurs in the CREATE_CHILD_SA exchange
is different, since everything is protected by an IKE SA. The peers
are authenticated, and error notifications can be relied on. See
Section 2.21.3 of [RFC5996] for more details on error handling in
this case.

5. IANA Considerations

   IANA has added a column named "Recipient Tests" to the Transform
   Type 4 - Diffie-Hellman Group Transform IDs registry for IKEv2
   [IANA-IKEv2-Registry].

   This column has been initially populated as follows.

   +-----------------------------------+----------------------+
   | Number | Recipient Tests |
   +-----------------------------------+----------------------+
1, 2, 5, 14, 15, 16, 17, 18	RFC 6989, Section 2.1
22, 23, 24	RFC 6989, Section 2.2
19, 20, 21, 25, 26, 27, 28, 29, 30	RFC 6989, Section 2.3
   +-----------------------------------+----------------------+

   Groups 27-30 are defined in [RFC6954].

   Future documents that define new DH groups for IKEv2 are REQUIRED to
   provide this information for each new group, possibly by referring to
   the current document.

6. Acknowledgements

   We would like to thank Dan Harkins, who initially raised this issue
   on the IPsec mailing list. Thanks to Tero Kivinen and Rene Struik
   for their useful comments. Much of the text in Section 3 is taken
   from [RFC2412], and we would like to thank its author, Hilarie Orman.

   The document was originally prepared using the lyx2rfc tool, created
   by Nico Williams.

7. References

7.1. Normative References

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

   [RFC5996]   Kaufman, C., Hoffman, P., Nir, Y., and P. Eronen,
               "Internet Key Exchange Protocol Version 2 (IKEv2)",
               RFC 5996, September 2010.

7.2. Informative References

   [IANA-IKEv2-Registry]
               IANA, "Internet Key Exchange Version 2 (IKEv2)
               Parameters",
               <http://www.iana.org/assignments/ikev2-parameters/>.

   [Kocher]   Kocher, P., "Timing Attacks on Implementations of Diffie-
               Hellman, RSA, DSS, and Other Systems", December 1996,
               <http://www.cryptography.com/timingattack/paper.html>.

   [Menezes]   Menezes, A. and B. Ustaoglu, "On Reusing Ephemeral Keys In
               Diffie-Hellman Key Agreement Protocols", December 2008,
               <http://www.cacr.math.uwaterloo.ca/techreports/2008/
               cacr2008-24.pdf>.

   [NIST-800-56A]
               National Institute of Standards and Technology (NIST),
               "Recommendation for Pair-Wise Key Establishment Schemes
               Using Discrete Logarithm Cryptography (Revised)", NIST PUB
               800-56A, March 2007.

   [RFC2412]   Orman, H., "The OAKLEY Key Determination Protocol",
               RFC 2412, November 1998.

   [RFC3526]   Kivinen, T. and M. Kojo, "More Modular Exponential (MODP)
               Diffie-Hellman groups for Internet Key Exchange (IKE)",
               RFC 3526, May 2003.

   [RFC5114]   Lepinski, M. and S. Kent, "Additional Diffie-Hellman
               Groups for Use with IETF Standards", RFC 5114,
               January 2008.

   [RFC5903] Fu, D. and J. Solinas, "Elliptic Curve Groups modulo a
              Prime (ECP Groups) for IKE and IKEv2", RFC 5903,
              June 2010.

   [RFC6954] Merkle, J. and M. Lochter, "Using the Elliptic Curve
              Cryptography (ECC) Brainpool Curves for the Internet Key
              Exchange Protocol Version 2 (IKEv2)", RFC 6954, July 2013.

Authors' Addresses

   Yaron Sheffer
   Porticor

   EMail: yaronf.ietf@gmail.com

   Scott Fluhrer
   Cisco Systems
   1414 Massachusetts Ave.
   Boxborough, MA 01719
   USA

   EMail: sfluhrer@cisco.com