Creating 40-Bit Keys for DES

Status of this memo

1. Introduction

This document describes an method for shortening DES keys from 56 bits
to 40 bits. The shortened keys are generally known as "DES-40". The
motivation for this weakening is that some localities give export
preference to applications that use relatively weak encryption
algorithms. Some implementors want to use DES with 40-bit keys instead
of other algorithms with 40-bit keys because they already have DES
coded into their products. The weakened keys are then used with the DES
encryption algorithm in the same manner as full-strength keys.

There are many possible methods for reducing a 56-bit key to a 40-bit
key. The method in this draft was chosen because one method is needed
for interoperability. Further, this method has been known to
occasionally have been approved for export from the United States.

2. Creating 40-Bit Keys for DES

DES [DES] uses a 56-bit key. The key consists of eight 8-bit bytes;
however the last (eighth) bit of each byte is used for parity, leaving
56 bits of key.

To weaken the 8-byte, 56-bit key into a 40-bit key, you set to zero

the first four bits of every other byte in the key, starting with the
first byte. Stated a different way, you take the bitwise logical AND
of the key and the binary value:
    0000111111111110000111111111110000111111111110000111111111111

Another way to picture this is:

Bit positions:
    000000000011111111112222222222333333333344444444445555555555666
    012345678901234567890123456789012345678901234567890123456789011123
Use:
    zzzzKKKpKKKKKKKpzzzzKKKpKKKKKKKpzzzzKKKpKKKKKKKpzzzzKKKpKKKKKKKp

    Legend:
     z = zero bit
     K = key bit
     p = parity bit

Some implementations of DES require the parity bit of each byte to be
set correctly in order for the key to be accepted. DES requires that
the last bit of each byte be a parity bit. DES uses odd parity,
meaning that the number of 1 bits in each byte should be odd.
Therefore, to complete the transformation to a 40-bit key, the
software SHOULD cause the parity in each byte to be odd, changing the
last bit if necessary.

[3]. Security Considerations

Current computer technology makes a brute-force attack on ciphertext
that is encrypted with a 40-bit key fairly quick. This is true for any
encryption algorithms, not just DES. Thus, 40-bit keys result in only
weak security against decryption. As computers get faster, this weak
security will become even weaker. Thus, 40-bit keys should never be
used with data that has a high value if it is decrypted by an
adversary. However, encrypting data with 40-bit keys prevents passive
snoopers from immediately reading a message without using some
significant but not onerous decryption effort.

Because of the ease of a brute-force attack on 40-bit keys, the 56-bit
key from which a 40-bit key is derived must not also be used as a
56-bit key. This is due to a simple attack that first derives the
40-bit key, then fills in the remaining 16 bits by brute force.
Systems that produce 40-bit keys from 56-bit keys must assume that the
associated 56-bit key is only slightly harder to compromise than the
40-bit key.

Note that short keys (and 40 bits is generally considered short) are
subject to a variety of brute-force attacks that are not possible with
longer keys, thus making them even more dangerous. For example, if a
40-bit algorithm is used and encrypted text includes a block of bytes
known to the attacker, then the attacker can pre-compute all possible

encryptions of that block and do a rapid comparison against the pre-computed ciphertexts. Further, it is likely that more attacks on short keys will appear in the future, thereby rendering them even less suitable for protecting data.

The shortening method described in this draft causes a discernable pattern of zero bits in the resulting key. There is no known literature at this time that describes whether cyphertext encrypted with a key that has this pattern of zeros is easier to decrypt than cyphertext that has no pattern. However, because 40-bit keys are already inherently weak, a decrease in security from the pattern is not considered to be very important relative to the inherent weakness due to the short key length.

There are other methods for converting longer keys to shorter ones. For example, IBM has created a patented (and significantly more complex) method called "Commercial Data Masking Facility", or CDMF [CDMF]; other methods probably exist. These methods might result in keys that produce cyphertext that is harder (or easier) to determine through brute-force. A quick comparison of CDMF and DES-40 shows that the brute-force attack against CDMF require one additional DES operation. Saving one DES operation does not seem to warrant the additional complexity.

A. References

[CDMF] "Design of the Commercial Data Masking Facility Data Privacy Algorithm", 1st ACM Conference on Computer and Communications Security, ACM Press, 1993.

[DES] ANSI X3.106, "American National Standard for Information Systems-Data Link Encryption," American National Standards Institute, 1983.

B. Object Identifier

In general, the algorithm identifiers associated with DES are used with DES-40 since the algorithm is exactly the same except that 16 of the key bits have known values. However, there are a few instances (such as algorithm negotiation) where DES-40 needs to be specified. The following algorithm identifier has been assigned for these cases. Note that no mode of operation is associated with this algorithm identifier.

```
id-alg-des40 OBJECT IDENTIFIER ::= { iso(1) identified-organization(3)
          dod(6) internet(1) security(5) mechanisms(5) pkix(7) alg(6) 1 }
```

C. Authors' Addresses

Paul Hoffman
Internet Mail Consortium
127 Segre Place
Santa Cruz, CA 95060 USA

phoffman@imc.org

Russ Housley
SPYRUS
381 Elden Street, Suite 1120
Herndon, VA 20170 USA
housley@spyrus.com