SHA-1 Hash Function Replacement

Uri Blumenthal
CTech Lab
Intel Corporation

Charanjit Jutla
Watson Research Center
IBM Corporation

November 10, 2005

What is the problem?

- **All crypto hashes deployed today – broken**
  - **To variable extent, but still…**
- **Strongest hash currently deployed – SHA-1**
  - **Design based on intuition, not science**
  - **Practical attacks are expected within one year**
    - **Either attack complexity will improve to 2^{50} (predicted lower bound)**
    - **Or distributed (and/or supercomputer) search will succeed at 2^{63} attack**
    - ***Or both?***

Why not SHA256?

- **Ultimately SHA256 planned as solution *but***
  - **Again intuition-based design that failed twice (SHA0 & SHA1), no known lower bounds**
  - **Requires radically new implementation**
  - **Different parameter size, etc.**
  - **Deployment expected by 2010 – not soon enough!**
  - **Performance sucks (and will for a few years)**
  - **Shares design weakness with SHA-1, plus**
    - **Non-linear code – security by confusion, not science**
    - **Non-linearity without analysis can lead to disastrous attacks**
  - **If truncated to 160 bits – problems compound**

Replace SHA-1 with what?

- Leading attack against SHA-1 – differential
  - Offers practical method of finding collisions
  - All other practical attacks rely on this one
- Math shows why differential attack possible
  - Weak key schedule (message expansion)
  - Low minimum Hamming distance
- Math also shows how to foil this attack
  - And therefore invalidate other attacks as well
- SHA1-IME is implementation of this defense

Structure of SHA1-IME

- **Same as SHA-1 in FIPS 180-1 and FIPS 180-2**
- **Minor change to message expansion**
  - **Old code (part of message expansion):**

```
1. for(t = 16; t < 80; t++)
2.     W[t] = ROL1(W[t-3] ^ W[t-8] ^ W[t-14] ^ W[t-16]);
```

  - **New code:**

```
1.  for(t = 16; t < 36; t++)
2.      W[t] = (W[t-3] ^ W[t-8] ^ W[t-14] ^ W[t-16]) ^
3.                 ROL13 (W[t-1] ^ W[t-2] ^ W[t-15]);
4.  for(t = 36; t < 80; t++)
5.      W[t] = (W[t-3] ^ W[t-8] ^ W[t-14] ^ W[t-16]) ^
6.                 ROL13, (W[t-1] ^ W[t-2] ^ W[t-15] ^ W[t-20]);
```

- **Provably secure against differential attacks**

Deploy SHA1-IME because

- SHA1-IME leaves API and PKCS unchanged (same input and output size)
- Performance hit minor – about 5% in software (possibly 10% in hardware)
- SHA1-IME is provably secure – proven lower bound on collision probability
  - Differential attack estimated 2^{150} probability
- SHA1-IME – easiest to get FIPS certification if you already certified SHA1
  - As it is a small change to already-certified FIPS 180-1, process much faster
- Code change miniscule – easier to do
  - Both software and firmware (ASIC may be in trouble ☺)

NO PATENTS!

References

- Uri Blumenthal uri.blumenthal@intel.com
- Charanjit Jutla csjutla@watson.ibm.com
  - ☞Please direct math questions to Charanjit ☺
- Anindya Patthak patthak@gmail.com
- Specification in draft-irtf-cfrg-sha1-ime-00.txt
  - URL to follow, also being submitted to CFRG

THANK YOU!