# Attacks on SHA-1

Initiative for Open AuTHentication

initiative for open authentication

http://www.openauthentication.org

# Attacks on SHA-1

Prepared by
By Mihir Bellare

Dept. of Computer Science & Engineering
University of California at San Diego
9500 Gilman Drive, La Jolla, California 92093, USA
March 2, 2005

# Table of Contents

# 1.   Introduction

This document addresses the impact of the recent attacks on SHA-1 on the security of the HMAC-SHA-1 based HOTP. We begin with some discussion of the situation of SHA-1 and then discuss the relevance to HMAC-SHA-1 and HOTP. Cited references are at the bottom of the document.

# 2.   SHA-1 status

A collision for a hash function h means a pair x,y of different inputs such that h(x)=h(y). Since SHA-1 outputs 160 bits, a birthday attack finds a collision in $2^{80}$ trials. (A trial means one computation of the function.) This was thought to be the best possible until Wang, Yin and Yu announced on February 15, 2005 that they had an attack finding collisions in $2^{69}$ trials.

Is SHA-1 broken? For most practical purposes we would say probably not, since the resources needed to mount the attack are huge.  Here is one way to get a sense of it: I estimate it is about the same as the time we would need to factor a 760-bit RSA modulus, and this is currently considered out of reach.

Burr of NIST is quoted [1] as saying ``Large national intelligence agencies could do this in a reasonable amount of time with a few million dollars in computer time.'' However, the computation may be out of reach of all but such well-funded agencies.

One should also ask what impact finding SHA-1 collisions actually has on security of real applications such as signatures.  To exploit a collision x,y to forge signatures, you need to somehow obtain a signature of x and then you can forge a signature of y. How damaging this is depends on the content of y: the y created by the attack may not be meaningful in the application context. Also, one needs a chosen-message attack to get the signature of x. This seems possible in some contexts, but not others. Overall, it is not clear the impact on the security of signatures is significant.

The press hypes it up. You read that SHA-1 is ``broken,'' [2], that encryption and SSL are ``broken'' [3], and so on, and you panic. But the media lives on hype. It would hardly be interesting to announce in the news that a team of cryptanalysts did very interesting theoretical work in attacking SHA-1, but there is no immediate cause for worry, would it?

Cryptographers are excited too. But largely because this is such an important theoretical breakthrough, not because they feel there is an immediate threat in practice.

So don't recall your software yet.  But stay tuned. Attacks can get better with time: once you make a dent in something, you can beat it more and more and maybe get a bigger hole. If there is a really practical break, you will hear about it.

# 3.   HMAC-SHA-1 status

The new attacks on SHA-1 have no impact on the security of HMAC-SHA-1. The best attack on the latter remains one needing a sender to authenticate $2^{80}$ messages before an adversary can create a forgery. Why?

HMAC is not a hash function. It is a message authentication code (MAC) that uses a hash function internally. A MAC depends on a secret key, while hash functions don't. What one needs to worry about with a MAC is forgery, not collisions.  HMAC was designed so that collisions in the hash function
(here SHA-1) do not yield forgeries for HMAC.

Recall that HMAC-SHA-1(K,x) = SHA-1(K_o,SHA-1(K_i,x)) where the keys K_o,K_i are derived from K. Suppose the attacker finds a pair x,y such that SHA-1(K_i,x)=SHA-1(K_i,y). (Call this a hidden-key collision.) Then if it can obtain the MAC of x (itself a tall order), it can forge the MAC of y. (These values are the same.) But finding hidden-key collisions is harder than finding collisions, because the attacker does not know the hidden key K_i.  All it may have is some outputs of HMAC-SHA-1 with key K. To date there are no claims or evidence that the recent attacks on SHA-1 extend to find hidden-key collisions.

Historically, the HMAC design has already proven itself in this regard.  MD5 is considered broken in that collisions in this hash function can be found relatively easily. But there is still no attack on HMAC-MD5 better than the trivial 2^{64} time birthday one. (MD5 outputs 128 bits, not 160.) We are seeing this strength of HMAC coming into play again in the SHA-1 context.

## 4.    HOTP status

Since no new weakness has surfaced in HMAC-SHA-1, there is no impact on HOTP. The best attacks on HOTP remain those described in the document, namely to try to guess output values.

The security proof of HOTP requires that HMAC-SHA-1 behave like a pseudorandom function. The quality of HMAC-SHA-1 as a pseudorandom function is not impacted by the new attacks on SHA-1, and so neither is this proven guarantee.

## 5.    References

[1] Crack in SHA-1 code 'stuns' security gurus
http://www.eetimes.com/showArticle.jhtml?articleID=60402150

[2] Bruce Schneier. SHA-1 broken. February 15, 2005.
http://www.schneier.com/blog/archives/2005/02/sha1_broken.html

[3] Researchers: Digital encryption standard flawed
http://news.com.com/Researchers+Digital+encryption+standard+flawed/2100
-1002-5579881.html?part=dht&tag=ntop&tag=nl.e703