The First Chosen-Prefix Collision on SHA-1

Gaëtan Leurent[1] and Thomas Peyrin[2]

[1] Inria, France
[2] Nanyang Technological University, Singapore

Abstract. The SHA-1 hash function was designed in 1995, and has been widely used during two decades. A theoretical collision attack was first proposed in 2004 [WYY05], but due to its high complexity it was only implemented in practice in 2017, using a large GPU cluster [SBK+17].
More recently, we have described a *chosen-prefix* collision attack against SHA-1 [LP19], a more powerful attack that allows to build colliding messages with two arbitrary prefixes. In this talk, we will announce the computation of the first chosen-prefix collision for SHA-1, and its impact on real world security with a PGP/GnuPG key-certification forgery.
Speaker: Gaëtan Leurent.

Keywords: SHA-1 · Cryptanalysis · Chosen-prefix collision · HPC · GPU · PGP · GnuPG

1 Introduction

Cryptographic hash functions are used in countless security applications and protocols. Recent standards such as SHA-2 or SHA-3 are believed to be secure, but their predecessor SHA-1 has been broken by a theoretical collision attack in 2004 [WYY05]. However, due to its high computational complexity this attack was only implemented in practice in 2017, using a large GPU cluster [SBK+17]. Moreover collision attacks are hard to exploit in practice, because the attacker has little control over the value of the actual colliding messages, where the differences are inserted. Because of this, the SHA-1 deprecation process has been quite slow in practice and one can still observe many uses of SHA-1 in the wild.

A stronger and easier to exploit attack is the so-called *chosen-prefix* collision attack (introduced for MD5 hash function in [SLW07]). The attacker is first challenged with two message prefixes P and P', and its goal is to compute two messages M and M' such that $H(P\|M) = H(P'\|M')$. Because the prefixes can be chosen arbitrarily, they can contain meaningful information, and this type of attack has been used to create colliding X.509 certificates, or even a rogue certificate authority [SSA+09].

Such collisions can be found generically with 2^{80} computations for a 160-bit hash function like SHA-1. Yet, we have recently described [LP19] a *chosen-prefix* collision attack against SHA-1 that requires an estimated complexity between $2^{66.9}$ and $2^{69.4}$ SHA-1 computations. It works with a two-phase strategy: given the challenge prefix and the random differences on the internal state it will induce, the first part of the attack uses a birthday approach to limit the internal state differences to a not-too-big subset (as done in [SLW07; Ste13]). From this subset, reusing basic principles of the various collision search advances on SHA-1, one slowly add successive message blocks to come closer to a collision, eventually reaching the goal after a dozen blocks.

Even though these advances put the chosen-prefix collisions within practical reach for well-funded entities, it remains very expensive to conduct and also very difficult to deploy as the attack contains many very technical parts.

2 Improvements to the Chosen-Prefix Collision Attack

While the work of [LP19] was mostly about high-level techniques to turn a collision attack into a chosen-prefix collision attack, we are now looking at the low-level details. We have a better understanding of the use of boomerangs in the attack, which leads to a complexity estimate of $2^{67.2}$ `SHA-1` computations, instead of a range of $2^{66.9}$ to $2^{69.4}$. We have also found several improvements to further reduce the cost down to $2^{63.7}$:

- Improvements to the near-collision search that reduces the cost of an identical-prefix attack from $2^{64.7}$ to $2^{61.5}$, through better use of degrees of freedom (message modifications and boomerangs) and GPU code improvements. The complexity estimates are obtained by measuring GPU code on a GTX 970, starting with the code of [SBK+17].
- Improvements to the CPC attack to reduce the gap between the cost of identical-prefix attack and a chosen-prefix one. In particular, we use a very large set of allowed differences (of size roughly $2^{37.6}$, which required an important implementation work), and we have a better control of the number of near-collision blocks.

3 Running a 2^{64} Computation on a Budget

Performing such a large-scale computation is still quite expensive, but can be performed using an academic budget. More precisely, we estimate that it would cost around 250k$ by renting GPUs from a cloud provider such as Amazon or Google. Alternatively, we can can rent cheaper GPUs from providers that use gaming or mining cards in consumer-grade PCs, rather that the datacenter-grade hardware used by big cloud providers. Services like gpuserversrental.com and Hostkey have GTX 1060 or GTX 1080 for a price below 5 cents per month per CUDA core; this gives a total cost between 50k$ and 100k$ to compute a chosen-prefix collision.

We have succesfully run the computation during two month last summer, using 900 GTX 1060 GPUs. We paid 75k$ to rent the GPUs from gpuserversrental.com, but actual price could be smaller because we lost some time tuning the attack. There is also a large variability depending on luck, and GPU rental prices flutuate together with cryptocurrency prices...

4 PGP/GnuPG key-certification Forgery

Our demonstration of a chosen-prefix collision targets the PGP/GnuPG Web of Trust. This trust model relies on users signing each other's identity certificate, instead of using a central PKI. For compatibility reasons the legacy branch of GnuPG (version 1.4) still uses `SHA-1` by default for key-certification signatures.

Therefore, we can forge key-certification signatures using `SHA-1` chosen-prefix collisions. More precisely, our goal is to create two PGP keys with different UserIDs, so that key B is a legitimate key for Bob (to be signed by the Web of Trust), but the signature can be transferred to key A which is a forged key with Alice's ID. This will succeed if the hash values used for the signature of the keys collide, as in previous attacks against X.509 MD5-based certificates [SSA+09]. However, due to details of the PGP/GnuPG certificate structure, our attack can reuse a single collision to target arbitrary users Alice and Bob: for each victim, the attacker only needs to create a new key embedding the collision, and to collect a `SHA-1` signature. This is arguably the first practical attack against a real world security application using weaknesses of `SHA-1`.

References

[LP19] Gaëtan Leurent and Thomas Peyrin. “From Collisions to Chosen-Prefix Collisions Application to Full SHA-1”. In: *EUROCRYPT 2019, Part III.*

[SBK+17] Marc Stevens, Elie Bursztein, Pierre Karpman, Ange Albertini, and Yarik Markov. “The First Collision for Full SHA-1”. In: *CRYPTO 2017, Part I.*

[SLW07] Marc Stevens, Arjen K. Lenstra, and Benne de Weger. “Chosen-Prefix Collisions for MD5 and Colliding X.509 Certificates for Different Identities”. In: *EUROCRYPT 2007.*

[SSA+09] Marc Stevens, Alexander Sotirov, Jacob Appelbaum, Arjen K. Lenstra, David Molnar, Dag Arne Osvik, and Benne de Weger. “Short Chosen-Prefix Collisions for MD5 and the Creation of a Rogue CA Certificate”. In: *CRYPTO 2009.*

[Ste13] Marc Stevens. “New Collision Attacks on SHA-1 Based on Optimal Joint Local-Collision Analysis”. In: *EUROCRYPT 2013.*

[WYY05] Xiaoyun Wang, Yiqun Lisa Yin, and Hongbo Yu. “Finding Collisions in the Full SHA-1”. In: *CRYPTO 2005.*