Cryptography: a crisis revealed – a resolution solved

A case study

In 2012 the American National Institute for Standards and Technology (NIST) will announce the winners of the "SHA-3" competition on electronic security and hash functions. But the story began at the Isaac Newton Institute in 1996.

Hash functions, security and digital signatures
At the heart of computer systems such as browsers, mobile phones or chip-and-pin payment cards, is a *cryptographic hash function*, which is an algorithm that maps arbitrary data to relatively small numbers. (For example, a hash function might map a multi-megabyte digital movie to its *hash value*, an integer in the range between zero and 2^{160}, say.) The input data can be arbitrarily large but the hash value has to be of relatively small fixed size. For use in security systems, two properties must hold. First, it must be "computationally hard" to identify input data from its hash value (*pre-image resistance*) and, second, it must be "computationally hard" to identify two inputs that hash to the same value (*collision resistance*).

Cryptographic hash functions have an important role in technologies that rely upon *digital signatures*. Like their hand-written counterparts, digital signatures authenticate the source of a message or document and confirm its integrity and completeness. But in practice, it is its hash value that is signed rather than the document itself. Thus, if different documents have the same hash value, a recipient may be led to believe that one document had been signed when in fact the sender had signed a different document with the same hash value, with consequent breach of security.

...if different documents have the same hash value, a recipient may be led to believe that one document had been signed when it fact the sender had signed a different document...

The mathematics behind hash functions
Designing secure hash functions, and in particular ones that satisfy collision resistance, is mathematically challenging. Since the input data can be arbitrarily large and since the output is of small fixed length, it is inevitable that there will be multiple inputs that share the same hash value – an exhaustive search would eventually find such values but this brute force approach is computationally impracticable. Thus, the only issue is whether collisions can be *efficiently* found.

A standard approach to the design of cryptographic hash functions is to construct a "compression function" and iterate it, the so-called *Merkle–Damgård construction*. Crucially, Merkle and Damgård proved that if the compression function is collision-resistant then the resulting hash function is also collision-resistant.

The hash function crisis: "secure systems" are no longer secure
In 1996, the Isaac Newton Institute held a six-month programme on *Computer Security, Cryptology and Coding Theory*, during which a participant, Hans Dobbertin, announced the paradigm-shifting discovery that there are collisions

in the compression function used inside the algorithm MD5, upon which most browsers relied [1]. Although Dobbertin had not found a technique for identifying collisions in MD5 itself, his result meant that Merkle and Damgård's Theorem did not apply (the pre-condition did not hold) and hence the presumed guarantee of security for this ubiquitous algorithm was invalidated.

This was a crisis: cryptographers asserted that MD5 should no longer be considered secure but many practitioners continued to use it in their products. They argued that MD5 itself had not been broken, only the underlying compression function upon which it was based.

However, as Dobbertin predicted might happen, in 2004 a group of Chinese researchers led by Xiaoyun Wang found a technique for finding collisions in the full MD5 algorithm. This was quickly followed by attacks on real world systems and collisions were found in X509 certificates used to authenticate web sites on the internet. Thus the MD5 algorithm was well and truly broken and alternative algorithms were needed. The obvious candidates were SHA-1 and SHA-2. Unfortunately, although their compression functions were more complex than that for MD5, they came from the same family as MD5, and an attack employing techniques similar to Wang's was plausible. Thus confidence in the security of hash functions had come to a low ebb.

NIST steps in

To address this crisis, in November 2007 NIST announced a competition to find a replacement for MD5, SHA-1 and SHA-2. The new algorithm would be called SHA-3, and researchers from around the world were invited to make proposals. Fifty one algorithms were submitted to the first round; by July 2009 the list had been reduced to fourteen and in December 2010 to five. The result will be announced in the Autumn of 2012.

The goal of the workshop was to break down the divide between theory and practice; and to create a true dialogue so we can learn the lessons from the past.

Return to the Isaac Newton Institute

In January 2012 cryptographers returned to the Institute for a workshop entitled *Is Cryptographic Theory Practically Relevant?* as part of the programme *Semantics and Syntax: A Legacy of Alan Turing*. The goal of the workshop was to break down the divide between theory and practice; and to create a true dialogue so we can learn the lessons from the past. They explored current best practice and observed that the discrepancy between theory and practice still remains.

Fittingly, in the room where 16 years earlier Hans Dobbertin predicted security breaches in MD5 (an algorithm that continues to be used - in 2005, 800 reported uses of MD5 in Microsoft Windows remained), speakers and participants spanning Government, academia and industry discussed the strengths and weaknesses of the remaining five candidates for SHA-3. Nigel Smart (University of Bristol) co-organiser of the programme *Semantics and Syntax: A Legacy of Alan Turing* observed that "the time is now ripe for a deeper, more fruitful, interaction between theoreticians and practitioners".

Speakers at the 2012 workshop included J Beric (Mastercard International), M Bond (Cryptomathic), C Cachin (IBM Research, Zurich), L Chen (Hewlett-Packard Laboratories), C Cremers (ETH Zurich), G Danezis (Microsoft), G French (Barclays), J Groth (UCL), R Horne (Barclays and UK Cabinet Office), A Kiayias (Athens), H Krawczyk (IBM Research, USA), D McGrew (Cisco), D Naccache (ENS Paris), C Paar (Ruhr University Bochum), B Preneel (KU Leuven), T Ristenpart (Wisconsin, Madison), H Shacham (UCSD), T Shrimpton (Portland State), G Steel (ENS Cachan), S Vaudenay (EPFL), M Ward (Mastercard International) and D Wikström (KTH - Royal Institute of Technology).

References

Isaac Newton Institute programme: *Computer Security, Cryptology and Coding Theory*. Jan–Jun 1996. Organisers: R Anderson, P Farrell, P Landrock and R Needham. Programme webpage: www.newton.ac.uk/programmes/CCC/

Isaac Newton Institute workshop: *Is Cryptographic Theory Practically Relevant?* Workshop webpage: www.newton.ac.uk/programmes/SAS/sasw07

[1] H Dobbertin. Cryptanalysis of MD4. In D Gollmann, editor, *Lecture Notes in Computer Science*, vol. 1039, pages 53–69. Springer, 1996.