COMPUTING

The security of knowing nothing

Bernard Chazelle

'Zero-knowledge' proofs are all about knowing more, while knowing nothing. When married to cryptographic techniques, they are one avenue being explored towards improving the security of online transactions.

Modern scientists, not unlike medieval monks, keep their knowledge firmly grounded in trust and authority. Unless it is part of their job description to probe such matters, they take it on faith that Genghis Khan defeated the Khwarezmid Empire, that praying mantids moonlight as sexual cannibals, and that man landed on the Moon. There is only so much a person can check: trust is the oxygen of the scientific community.

Unless, of course, it has to do with online shopping. Computer transactions tend to bring out the paranoid in all of us. By all accounts, that is a healthy reaction. We might be putting our faith in online auctions, e-voting, computer authentication and privacy-preserving data mining, but — with very different aims in mind — so are the bad guys. More than a decade ago, Oded Goldreich, Silvio Micali and Avi Wigderson showed how to make virtually any cryptographic task secure[1]. But unfortunately, their otherwise remarkable scheme breaks down in 'concurrent' settings, which is another way of saying that it fails where it really shouldn't — namely, on the Internet. Work by Boaz Barak and Amit Sahai in recent years, however, offers a way out of this bind[2].

The secret to secure online transactions is the mastery of 'zero knowledge': the art of proving something without giving anything else away. Can I convince you that I am the better chess player without ever playing a game, that I am younger than you without divulging my age, or that I can prove a hard theorem without revealing my hand about the proof? Can a referendum take place on the Internet that leaks no information about voters' preferences? The concept of zero knowledge[3], introduced in the mid-1980s, helps us to formalize these questions.

To illustrate the principle, let us say Petra (the prover) and Virgil (the verifier) are shown the subway map of a large metropolitan area (Fig. 1). Blessed with superior mental powers, Petra claims to see right away that it is possible to visit every stop exactly once without leaving the subway system, thus forming what is called a hamiltonian path. Poor Virgil sees nothing of the sort — the reason being his inability to solve conundrums like the one at hand, known as NP-complete problems.

Such problems have solutions that can be verified in a number of steps proportional to a polynomial in the size of the input data. Whether all NP-complete problems can actually also be solved within that same time is an open question, arguably one of the most pressing in all science. The answer is believed to be no: this is why Virgil badly needs Petra's help if he is to be convinced of her claim.

A zero-knowledge proof takes the form of a question-and-answer session between Petra and Virgil that will leave Virgil convinced of the

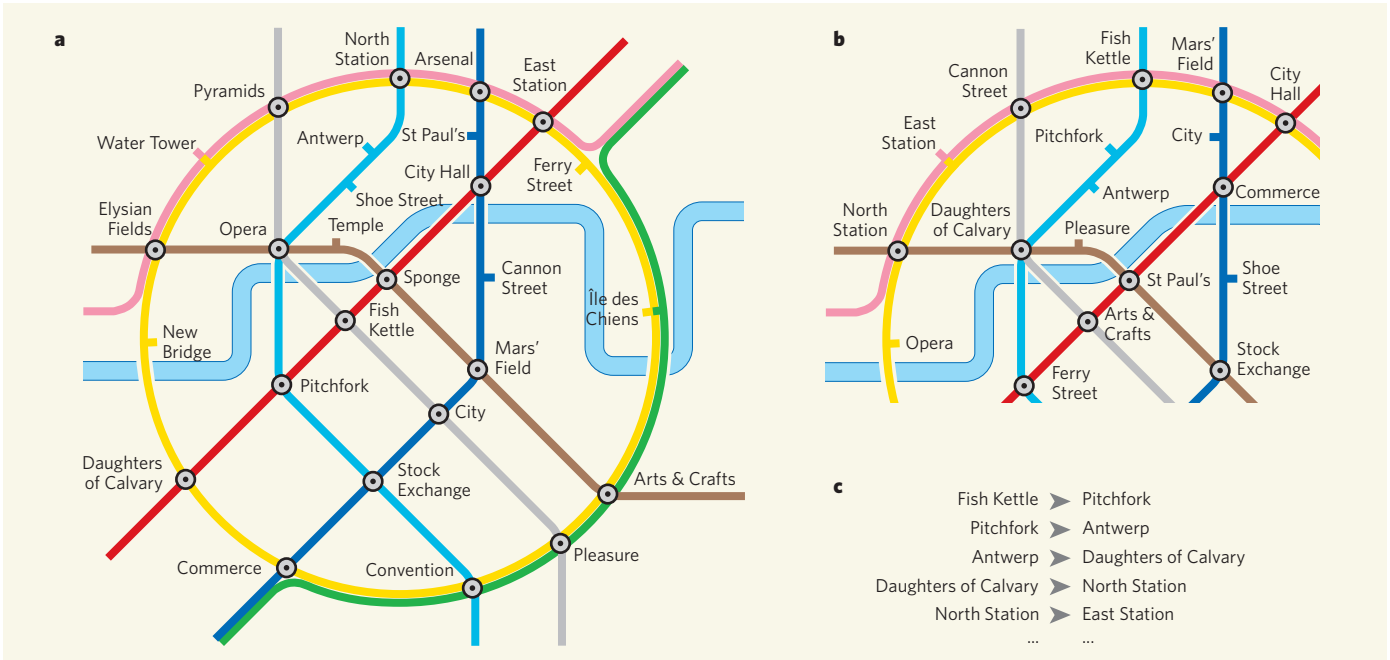

Figure 1 | Tube tour. After decades of steady investment, Orbiville's underground-railway system (**a**) is finally blessed with its own hamiltonian path, allowing one to visit every station exactly once without leaving the system. Or so Petra believes. To convince Virgil of that development, she renames the stations on the Orbiville metro map at random (**b**): North Station becomes Fish Kettle, Antwerp becomes Pitchfork, Shoe Street becomes Antwerp, and so on. She then hides the renamed map and the permutation table in a safe. Next, Virgil tosses a coin: heads, Petra opens the locked box and Virgil checks that the map was properly renamed, with each station appearing exactly once; tails, Petra reveals only the pairs of names on the relabelled map (**c**) that form the hamiltonian path. She also grants Virgil restricted access to the safe by revealing to him only the path on the map formed by the pairs of names. In this way, Virgil can check that the alleged path not only visits each station once, but also actually exists in the subway system — all the while being denied access to the rest of the map. If Petra can perform this test a few times without raising eyebrows (using freshly permuted names at each round), Virgil will leave utterly assured of the existence of a hamiltonian path, yet clueless about its layout. Why? If Petra did not have a valid path or cheated in the renaming, she would have at least a 50% chance of getting caught at each round: either the encryption or the path would have to be faulty. Heads could catch the first case, and tails the second one. Repeating the test boosts Virgil's confidence in Petra's claim beyond any reasonable doubt — or shatters it, as the case may be. At each round, Virgil learns the encrypted map or the alleged hamiltonian path with the renamed stations, but never both. This ensures zero knowledge.

existence of a hamiltonian path, but that will reveal nothing about it[4]. If Petra is mistaken or in a cheating mood, Virgil will spot a contradiction in her responses and reject her 'proof'. If Petra is correct, however, not only will Virgil believe her claim, but he will learn nothing else that he could not figure out on his own without Petra's help. Better still, no amount of deviousness in posing the questions on Virgil's part can alter that fact. Zero knowledge is a mechanism for enforcing honest behaviour on both sides: cheating would bring no benefits to Virgil and it would expose Petra to the embarrassment of getting caught. Little wonder that zero knowledge is often held up as the 'holy grail' of secure computing.

But do zero-knowledge proofs even exist? Under widely accepted assumptions, the answer is yes for any NP-complete problem[5]. Were Petra to hand over the hamiltonian path, Virgil would learn more than the mere truth of her claim: he would know the itinerary itself, something, we just argued, he could not find on his own. To achieve zero-knowledge status requires, as so often in life, dialogue, commitment, and a bit of luck. To keep Petra from deceiving him with inconsistent answers, Virgil will ask her to commit to her claim once and for all. This is the software equivalent of hiding the hamiltonian path in a locked box that neither party can access until both jointly decide to open it. In fact, the path must be relabelled randomly, so that merely looking at it will not help Virgil trace it in the original map. Why so much suspicion? Because distrust is the very ailment that zero knowledge seeks to cure: if Virgil trusted Petra, after all, her good word alone would be sufficient and no proof would be needed.

The surprise is that, to enforce honesty among distrustful parties, randomness must be thrown into the mix. A nasty side effect is that the conversation might go awry and lead Virgil wrongly to conclude that a hamiltonian path exists when none is to be found. This should happen with a probability of below 50%. (Correct claims will never be rejected, however.) Isn't this being inordinately lax? No: by repeating the dialogue a few dozen times, one can easily reduce the error probability to one in a trillion.

For technical reasons, to carry on several such conversations at the same time, as might happen on the Internet, is a big no-no. The issue is subtle, but to see why concurrency facilitates deception is not. Ever care to 'beat' your local chess-club champ? Here is how you do it: arrange for Garry Kasparov to play a match with you while you play with the local champion simultaneously — and do it online so neither one can see what you're up to. The trick is to feed each player the other's moves: in all likelihood, you will lose to Kasparov, but win against your neighbourhood champion.

The beauty of Barak and Sahai's work[2] is that they are able to overcome the pitfalls of concurrency by deploying a sophisticated arsenal of cryptographic techniques. One of them is to relax the definition of zero knowledge by enhancing Virgil's power to simulate any potential dialogue with Petra. Another is to squeeze long messages into shorter ones without losing their security properties. Although fairly complex, the Barak–Sahai technology takes us one step closer to true security on the Internet.

In 1962, US President John F. Kennedy dispatched Dean Acheson to Paris to offer Charles de Gaulle photographic evidence of Soviet missiles in Cuba. The French president declined to see it, saying: "The word of the president of the United States is good enough for me." Zero knowledge is so blissfully easy in a climate of trust. The challenge is to deal with liars and cheaters. The continuing work of those such as Barak and Sahai[2] is giving us new tools to do that on the Internet. ■

Bernard Chazelle is in the Department of Computer Science, Princeton University, 35 Olden Street, Princeton, New Jersey 08540-5233, USA.
e-mail: chazelle@cs.princeton.edu

1. Goldreich, O., Micali, S. & Wigderson, A. *Proc. 19th ACM Symp. Theor. Comput.* 218–229 (1987).
2. Barak, B. & Sahai, A. *Proc. 46th IEEE Symp. Fut. Comput. Sci.* 543–552 (2005).
3. Goldwasser, S., Micali, S. & Rackoff, C. *SIAM J. Comput.* **18,** 186–208 (1989).
4. Goldreich, O. *Foundations of Cryptography* (Cambridge Univ. Press, 2001).
5. Goldreich, O., Micali, S. & Wigderson, A. *J. Ass. Comput. Machin.* **38,** 691–729 (1991).

CHEMICAL BIOLOGY

Dressed-up proteins

Gijsbert Grotenbreg and Hidde Ploegh

Proteins aren't just defined by their constituent amino acids — structural modifications can yield complex mixtures of protein forms. An approach that controls the addition of such modifications may help to define their role.

Polypeptides freshly minted from the cell's protein-assembly apparatus are by no means ready for active service. Both the peptide backbone and its side chains may need to be altered by post-translational modifications (PTMs), the covalent attachment of chemical groups that change the properties, and hence the function, of newly generated proteins. PTMs also control the degradation of aberrant proteins and those at the end of their lifespan. Such modifications dramatically expand the compositional and functional complexity of these molecules. Not satisfied with leaving everything to nature's whim, chemists are taking great strides in developing PTM-mimics. On page 1105 of this issue[1], van Kasteren *et al.* describe strategies that offer precise control over the attachment of carbohydrates and other PTM-mimics to polypeptides.

In nature, any given protein may exist as a mixture of forms, each incorporating different PTMs. Assessment of the contribution made by each PTM to that protein's function demands that the individual components of these mixtures should be isolated and identified. But separating the constituents from these complex — and possibly dynamic — mixtures is a dauntingly arduous task. The ability to uniformly modify proteins at specific sites in the molecule, using chemical approaches that mimic the original PTM, is therefore essential for progress in this area.

A few examples of this approach are already known. One clever and synthetically straightforward example was reported recently[2], in which the side chains of cysteine amino acids in proteins were selectively targeted to mimic a particular PTM — the modification of lysine amino-acid side chains with methyl groups. The resulting molecules were used to assess the role of this PTM in DNA–protein complexes found in cell nuclei.

One intricate form of PTM that remains a prize target for chemists is the attachment of carbohydrates — glycans — either to nitrogen atoms (N-linked) in the side chains of asparagine amino acids, or to oxygen atoms (O-linked) on the side chains of serine or threonine. In living cells, glycosidase and glycosyl transferase enzymes first trim N-linked glycans, then extend them with sugars that can branch in several directions, generating numerous variations on a theme. Glycosyl transferases also act on O-linked glycans, imposing similar extensions and modifications. To add to the complexity, each sugar can bear different chemical groups, and the linkages between sugars have specific orientations. Many biological processes, such as cellular differentiation and development, cell adhesion, immune surveillance and inflammation, rely to varying degrees on the correct decoration of proteins with such glycans.

Van Kasteren *et al.*[1] have now responded to the clarion call for improved synthetic methods to modify proteins. Not only have they installed carbohydrate PTM-mimics at specific locations in a protein, but they have also incorporated another important motif — a sulphated