# STANFORD SPLASH: ZERO-KNOWLEDGE PROOFS

NITYA MANI, ANDY CHEN

## 1. Introduction to Zero-Knowledge Proofs

Generally in mathematics, the goal of a proof is for the writer to convey knowledge and understanding to the reader. (However, you may doubt this claim based on some books or papers you have read or talks you have attended, although surely not any of mine.) But from time to time, the prover *wants* to obfuscate some key pieces of information. As we shall see, being able to obfuscate some key pieces of information in the proof has important cryptographic implications.

Let us start with a toy example, which is taken from the charming paper "Applied Kid Cryptography, or How To Convince Your Children You Are Not Cheating" by Naor, Naor, and Reingold.[1] Many of us, when we were very young, did "Where's Waldo" puzzles. The goal is to find Waldo in a picture, which may contain similarly-dressed decoys. Suppose you have found Waldo, and you want to convince me that you know where he is. You *could* just point at him, but where's the fun in that? Now you have just ruined a perfectly good puzzle for me.

**Figure 1.** Waldo

Instead, what you want to do, somehow, is to convince me, beyond a reasonable doubt, that you know where he is, while still not giving me any information about where to find him. How can you do this?

One possibility is that you could just *declare* that you have found him. But this is not convincing evidence: you could just as easily claim that you know where Waldo is when you are, in fact, just as clueless as I.

In fact, there is a way that you can convince me that you have found Waldo, without giving away any information about where he is.

One possibility is as follows: you take the *Where's Waldo* book, make a photocopy of the page, cut out the Waldo in the photocopy (as civilised people, we do not cut up actual books), and show me the Waldo cut-out.

---

*Date*: April 10, 2016.

[1] `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.46.9932&rep=rep1&type=pdf`

But I fundamentally don't trust you. I suspect that you have photocopied a *different* page of the *Where's Waldo* book, one which is much easier, and cut out the Waldo from that page. You could do that with a bit of sleight of hand, and I would be none the wiser.

So, I don't accept that solution. But there is another solution that would satisfy even me, at my most stubborn. What you can do is to take a giant piece of paper, much larger than the *Where's Waldo* book, cut out a small Waldo-sized hole in it, and put the hole over the Waldo. Then I can verify that you have actually found Waldo, since I can easily verify that the character in question is actually Waldo.

Finally, one more step you should take: pull the book out from under the paper (while covering the hole), so that I can be satisfied that you haven't flipped the page while navigating Waldo to the hole.

An interesting consequence of all this is that, although I am now entirely convinced that *you* know where Waldo is, and indeed that the page does in fact contain a Waldo, I will struggle to convince anyone else of this fact. I could videotape the entire the whole process of you putting the paper over the Waldo book and showing me the Waldo, point the camera at the Waldo once it has been covered by the hole, and so forth. But a skeptical third-party would not be satisfied: I could have paused the video at several points in the middle and flipped to a different page in the *Where's Waldo* book, and someone replaying the video would have no method of detecting the trick. Video editing software is powerful!

## 2. What is a Zero-Knowledge Proof?

Let us now abstract this idea a bit and talk about what a zero-knowledge proof is supposed to do, in general. As in standard cryptography, we have some characters, but they aren't the same characters we met earlier. In zero-knowledge proofs, we typically have two characters, named Peggy (the prover) and Victor (the verifier). Peggy knows some piece of information, and she wants to convince Victor that she knows that particular piece of information, while giving away absolutely nothing else. Typically, this piece of information is hard to come up with in the first place. (There isn't much point in having Peggy convince Victor that she knows the answer to a problem, when he can solve the problem easily himself. We're all mature people here; we don't have to tease and examine each other.) As we have already discussed quite a lot, factoring appears to be a difficult problem. If Peggy knows the prime factorization of some large number $n$, can she somehow convince Victor that she knows it, without giving away any details of the factorization? (For example, she should not be willing to give away any information about the *number* of prime factors, or how many digits the smallest prime factor is, or any other related information.)

Generally, a zero-knowledge proof will consist of several rounds of information exchange. For example, Peggy may do some secret pre-computation and give the result of the pre-computation to Victor. Victor can then use the result of this pre-computation to issue a challenge to Peggy, a challenge that Victor knows how to solve. (Or, if he can't solve it on his own, he has some way of verifying after the fact that Peggy did it correctly.) Using her secret information, Peggy solves Victor's challenge problem and reports the answer. This exchange can go on for several rounds, until Victor is convinced that Peggy does indeed possess the secret information.

Now, Victor may be skeptical, but he is also a reasonable and decent human being. (On the other hand, we do *not* assume that Peggy is a reasonable and decent human being. It is not a good idea to trust people who can do magic calculations and come up with answers to

computationally difficult problems.) Victor does not demand absolute certainty that Peggy has the secret information; for example, she might just be able to guess the answers to all of Victor's challenges completely by accident, without needing to know the secret information. Or, she might attempt to cheat by guessing Victor's challenges and solving easier problems that will allow her to give seemingly-correct answers to the challenges. (We'll see an example of this later.) But, assuming that Victor chooses suitable challenges, Peggy's chances of guessing correctly every time are very small, small enough that Victor realizes that it is *much* more likely that she is being honest, than that she is lying and getting lucky every time. If Peggy claims that she can flip a coin 1000 times in a row and get heads every time, *and then she does so*, it's more likely that she has a two-headed coin than it is that that happened just by chance, isn't it?

We expect a zero-knowledge proof to satisfy three important properties:

**Completeness:** If Peggy is telling the truth, then Victor will be convinced of this.
**Soundness:** If Peggy is lying, then she cannot convince Victor that she is telling the truth, except with some very small probability.
**Zero-knowledge:** If Peggy is telling the truth, then Victor learns nothing other than the fact that she is telling the truth.

*Remark* 2.1. Formally, the zero-knowledge property is quite subtle. We have the notion of an *honest verifier*, who follows the proof protocol with the prover, as well as a *cheating verifier*, who creates a transcript that could potentially be that of the communication between an honest verifier and the prover. The zero-knowledge property says that a real transcript, created by an honest verifier and the prover, is indistinguishable from a transcript that is entirely fabricated by a cheating verifier. The reason this is a desirable property is that we do not wish to allow the verifier to pretend to be the prover at some later point.

*Remark* 2.2. In fact, even Remark 2.1 is *still* not fully rigorous, because we have not stated what we mean by two transcripts being "indistinguishable." One possibility is that the fake transcript could actually *be* a real transcript. But a weaker notion that is sometimes useful is a computational version of indistinguishability: it might not be possible for the fake transcript to occur as a result of a conversation between an honest verifier and an honest prover, but it takes a third party a very long time to detect the forgery. This is usually also acceptable to us.

Let us see more examples of zero-knowledge proofs

## 3. Ali Baba's Cave

In addition to playing Where's Waldo as kids, we also learn about Ali Baba. He followed a band of thieves to learn of a treasure-filled cave with an entrance that would open upon hearing the magic words "Open Sesame!"

In the zero-knowledge story of Ali Baba's Cave, as told by Quisquater and Guillou in their also-charming paper "How to Explain Zero-Knowledge Protocols to your Children,"[2] there is a cave with two entrances. (See Figure 2.) There is a door connecting the two entrances, and this door will open to a person who says the words "Open Sesame!" and otherwise it remains shut.

[2] `http://pages.cs.wisc.edu/~mkowalcz/628.pdf`

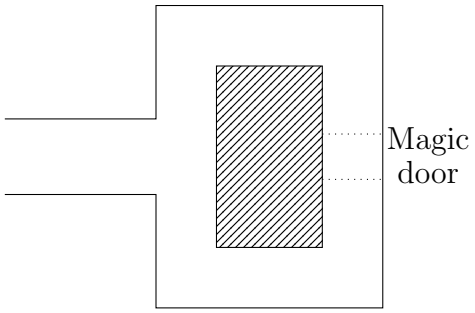


**Figure 2.** Ali Baba's Cave

Ali Baba brings a friend, Vera, to the cave and shows her the two entrances. He tells her that there are certain magic words he can say to open a door between the entrances. Naturally, Vera is skeptical. (Wouldn't you be, especially if you were living in the Middle Ages and there was no such thing as voice recognition software?) So, she asks Ali Baba to indulge her in a verification. Ali Baba is to enter the cave through one of the two entrances, of his choosing. Then Vera flips a coin, and depending on the result of the coin flip, she either calls out "Come out through the left entrance" or "Come out through the right entrance." Since Ali Baba can get through the door, he can come out through the desired entrance, regardless of which side he entered.

Now, one attempt doesn't prove much. So, they repeat this experiment 40 times. Every time, Ali Baba comes out through the correct entrance. If he didn't have access to the door, then the chances of his being able to do this would be $\frac{1}{2^{40}}$, which is very small. Vera believes that Ali Baba does indeed know the magic words.

Maybe you object to this procedure: wouldn't it be easier if Vera were to *watch* Ali Baba enter the cave through the left side and then come out through the right side? That would eliminate the need for repeating the test over and over again: if Ali Baba can enter through the left side and exit through the right side even once, then he *must* know the magic words. (We're ignoring the possibility of quantum tunneling here.)

Yes, this would be easier, but the other version is better. Why? If they adopted this second procedure, and Vera were to videotape the process, then she could show her videotape to a third party. (Suppose they haven't invented video editing software or anything similar yet.) Vera could then convince this third party that Ali Baba knows the magic words. But we don't want to allow her to do that. In terms of honest and cheating verifiers, this procedure would allow her to create a "transcript" of the event that she could not make on her own, without Ali Baba's help.

In the first version, a videotape of the experiment is not convincing evidence that Ali Baba knows anything. The entire process could have been staged, so that Ali Baba and Vera agreed in advance which side Ali Baba should exit (and hence enter).

But what about the coin flips? Aren't those enough to make it random and hence convincing? First of all, with a bit of practice, one can flip a coin in such a way that one knows what the outcome will be in advance. (One way to do this is to "flip" the coin so that it only wobbles and never actually flips over. It is not so easy for a spectator to tell the difference.)[3]

[3]See the paper "Dynamical Bias in the Coin Toss" by Persi Diaconis, Susan Holmes, and Richard Montgomery at `http://statweb.stanford.edu/~cgates/PERSI/papers/dyn_coin_07.pdf`. There's a lot of hard work that goes into studying coin flipping!

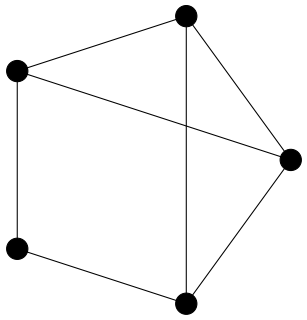

**Figure 3.** A graph

But more seriously, in cryptography, "random" numbers are not really so random: they are the result of a deterministic process that starts with a given "seed" value. So, it is possible that Ali Baba and Vera have set their seed value in advance, so that they know what all the "coin flips" are going to be. There *are* sources of true randomness such as radioactive decay, but your computer most likely isn't using them. Sometimes this can be a serious problem; for example, you can read about how some people managed to cheat an online poker server through bad random number generation, among other serious errors.[4]

## 4. Three-colorability

Let us now look at some less frivolous and more potentially relevant zero-knowledge proofs. In mathematics, the term "graph" has several different meanings, and they are unrelated to each other. For us, a graph will consist of a set $V$ of vertices, and a set $E$ of edges. An edge consists of an unordered pair of distinct vertices. Frequently, we represent a graph as a picture, with vertices pictured as dots and edges pictured as lines or curves connecting their two vertices. (See Figure 3.)

Graph theory is a large subject, and there are many things that are known about graphs. One of the most famous results about graphs is the infamous four-color theorem. We are interested in coloring each vertex of the graph with one of several colors, in such a way that if two vertices are connected by an edge, then they must be colored using different colors. See Figure 4 for a valid coloring with three colors, as well as an invalid coloring. A graph is said to be $k$-colorable if it can be colored in this way using (at most) $k$ colors. A graph is said to be *planar* if it is possible to draw it on the plane without any edge crossings. A famous example of a non-planar graph is the graph known as $K_{3,3}$, pictured in Figure 5.

**Theorem 4.1** (Four-Color Theorem)**.** *Every planar graph is 4-colorable.*

The four-color theorem is notoriously difficult, and it was only proven in 1976 by Kenneth Appel and Wolfgang Haken after an extensive computer search, perhaps the first major theorem whose proof relied on computers in an essential way.

Now, *some* planar graphs can be colored using only three colors, but not all of them. Furthermore, given a graph $G$, it does not appear to be easy to determine whether $G$ is 3-colorable. (In fact, this problem is NP-complete.)

In general, hard problems like this are good candidates for zero-knowledge proofs. In fact, any NP problem has a zero-knowledge proof. (Once we exhibit a zero-knowledge protocol for

[4]See "How We Learned To Cheat in Online Poker: A Study in Software Security," `https://www.cigital.com/papers/download/developer_gambling.pdf`

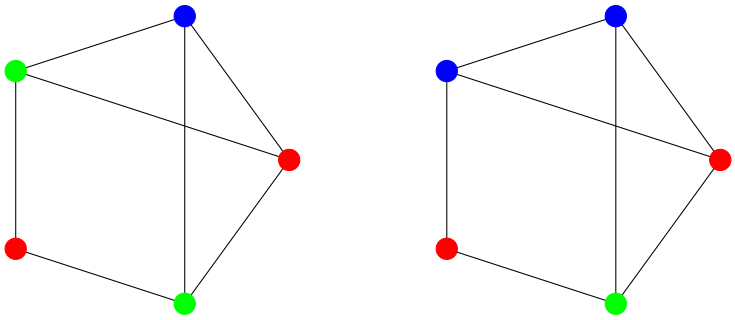

**Figure 4.** Left: A valid coloring with three colors. Right: An invalid coloring, since there are two adjacent blue vertices.

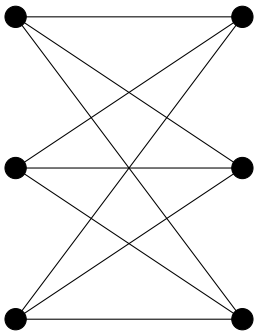

**Figure 5.** This graph, known as $K_{3,3}$, is not a planar graph.

3-colorability, the proof that 3-colorability is NP-complete will imply that any NP problem has a zero-knowledge proof. However, it also appears that problems that are not in NP can have zero-knowledge proofs; see problem 6 for an example of a zero-knowledge proof for a problem that does not appear to be in NP.) Let us see how it works in this case.

We assume that a graph $G$ *is* three-colorable, and that Peggy knows a three-coloring. She would like to convince Victor that she knows a three-coloring, without giving away any information about how the coloring works. In particular, she does not want to tell him *how* to do the three-coloring.

Why might she not want to tell? Well, perhaps she eventually wants to sell him the three-coloring. She doesn't want to tell him how it works before he pays, because once he knows the three-coloring, he has no incentive to pay for it. On the other hand, if Victor pays up front, then he will be unhappy if Peggy is lying and doesn't actually know how to three-color the graph, or if the graph turns out not to be three-colorable at all. From a legal perspective, it is possible to set up an escrow system to deal with such practical issues, but isn't it just more fun to solve all our problems using math?

So, now for our procedure. For each edge in the graph, Victor needs to know that the two vertices it connects are colored differently. So, he can pick an edge and ask Peggy for the colors of its two vertices. If the two colors are ever the same, then he knows that Peggy is lying.

However, there are several reasons why this is not yet a satisfactory solution. From Victor's perspective, it is not satisfactory, because Peggy can just say whatever she wants, regardless of the truth. From Peggy's perspective, it is also not satisfactory, because Victor is getting information about the coloring.

We can improve the situation from Victor's perspective by having Peggy first color the graph and then cover all the vertices. Then Victor gets to point to an edge, and Peggy uncovers the two vertices that it connects, so that Victor can see that the two vertices are colored differently.

But this is still not satisfactory from Peggy's point of view, since she is giving away information about the coloring. However, note that if she has a coloring of the graph, say with red, green, and blue vertices, then she can easily generate several more colorings. For example, she gets a new coloring by swapping all the red and blue vertices, and keeping all the green ones fixed.

In fact, starting from one 3-coloring, Peggy can very easily produce five more, for a total of six. We can use this to modify the earlier attempt: instead of always using the same coloring each time Victor asks for an edge, she picks a random one of the six colorings before each request.

More precisely: Peggy takes her six 3-colorings of the graph. They then repeat the following procedure several times: Peggy picks one of her six 3-colorings at random and covers all the vertices. Then Victor points to an edge. Peggy removes the covers from the two vertices of that edge. Victor checks that the two vertices are colored differently. After enough queries, Victor is confident that Peggy does in fact have a 3-coloring. He has learned nothing about the coloring, because if he sees that two vertices are colored (say) red and blue, then all that means is that there is a 3-coloring in which they *could* be colored red and blue. But this is obvious, because the specific colors have no special meaning.

If Peggy is cheating, meaning that she doesn't actually have a 3-coloring, then Victor will eventually figure this out, because he will point to an edge and see two adjacent vertices with the same color. Or perhaps Peggy has cheated by not coloring all the vertices. Victor will also eventually notice this.

Suppose that the graph has $m$ edges. Then, if Peggy is lying, then there must be one "faulty" edge, with two vertices of the same color. So, if Peggy is lying and Victor picks a random edge, then his chance of exposing the lie is at least $\frac{1}{m}$. If they repeat the experiment $m$ times, then his chance of ever exposing the lie is at least $1 - \left(1 - \frac{1}{m}\right)^m \approx 1 - \frac{1}{e}$. If they instead repeat it $mk$ times, for some $k$, then his chance of finding out is at least $1 - \left(1 - \frac{1}{m}\right)^{mk} \approx 1 - \frac{1}{e^k}$, which becomes very close to 1 for large $k$. It is reasonable for them to be willing to do $mk$ trials, since this is a polynomial function of $m$, the number of edges in the graph.

## 5. Sudoku

This time, Victor is trying to do a Sudoku and is having trouble. As usual, Peggy knows how to do it. Instead of giving a hint, she wants to rub it in and convince Victor that she already knows the solution.

Here are the rules of sudoku. There is a $9 \times 9$ grid of squares, divided into a $3 \times 3$ grid of $3 \times 3$ squares. Initially, some of the squares are filled in with numbers from 1 to 9. The goal is to fill in the remaining squares, in such a way that each row, column, and $3 \times 3$ square contains each of the digits from 1 to 9, exactly once. See Figure 6 for a challenging example.[5]

How does Peggy demonstrate to Victor that she knows how to solve the sudoku? Since sudoku (at least, in its $n^2 \times n^2$ generalization) is an NP problem (in fact, like 3-colorability,

[5]Source: http://www.telegraph.co.uk/news/science/science-news/9359579/Worlds-hardest-sudoku-can-you-crack-it.html

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 8 | | | | | | | | |
| | | 3 | 6 | | | | | |
| | 7 | | | 9 | | 2 | | |
| | 5 | | | | 7 | | | |
| | | | | 4 | 5 | 7 | | |
| | | | 1 | | | | 3 | |
| | | 1 | | | | | 6 | 8 |
| | | 8 | 5 | | | | 1 | |
| | 9 | | | | | 4 | | |

**Figure 6.** A difficult sudoku problem

it is NP-complete), we can convert the sudoku problem to a 3-colorability problem and use the zero-knowledge proof we just looked at for 3-colorability. But such a proof would be difficult to understand.

Instead, we want a zero-knowledge proof that has more of a sudoku "flavor" to it. Here is one. (Find another one in problem 11.) First, Peggy fills out her solution on a separate sheet, that she does not show to Victor. Now that she has this one filled-out grid, it is easy for her to generate others. Let $\sigma$ be any permutation of the numbers $\{1, \ldots, 9\}$, i.e. a bijective function $\sigma : \{1, \ldots, 9\} \to \{1, \ldots, 9\}$. Recall that this means that for each $y \in \{1, \ldots, 9\}$, there is a *unique* $x \in \{1, \ldots, 9\}$ such that $\sigma(x) = y$. To generate a new completed sudoku grid, she simply replaces every $x$ on the grid with $\sigma(x)$. Since there are $9! = 362880$ permutations of $\{1, \ldots, 9\}$, this gives her 9! filled out sudoku grids.

This suggests an attempted zero-knowledge proof. Peggy picks one of the 9! permutations at random and generates the corresponding filled-out grid. Victor needs to verify that this is a correct solution, so he needs to check that, in every row, every number from 1 to 9 is used exactly once. Similarly for every column and $3 \times 3$ square.

So, we allow Victor to name a row, column, or $3 \times 3$ square. Peggy then reveals all the entries in that row, column, or $3 \times 3$ square, and Victor can check that all the numbers are indeed different.

While this is enough to convince Victor that Peggy has a complete sudoku grid, it is not completely satisfactory. The reason is that Peggy could have chosen an entirely different (and much easier) sudoku puzzle to start with. Victor also needs to know that Peggy started with the same numbers filled in as he did.

So, we need to allow Victor one more option. In addition to being allowed to see all the entries in a row, column, or $3 \times 3$ square, he also gets to ask to see what's in all the squares that were originally filled in. He doesn't expect to see exactly the same numbers as in the

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | | | | | | | | |
| | | 4 | 8 | | | | | |
| | 9 | | | 2 | | 6 | | |
| | 5 | | | | 9 | | | |
| | | | | 7 | 5 | 9 | | |
| | | | 3 | | | | 4 | |
| | | 3 | | | | | 8 | 1 |
| | | 1 | 5 | | | | 3 | |
| | 2 | | | | | 7 | | |

**Figure 7.** A permutation of the original sudoku problem

original, but he expects to see the same *pattern* as in the original: if he knows of two squares that (say) both contain 4's, then he expects to see the same number in both squares, even though they might not be 4's. Similarly, if he knows of two squares that contain *different* numbers, then in the version Peggy shows, they should also show different numbers. A permutation of the original sudoku is shown in Figure 7.

After running this test several times, Victor is convinced that Peggy is telling the truth.

## 6. Time machines and prover tricks

One day, you receive an unsolicited letter in the mail, predicting the winners of five sports games to take place over the next month. Instead of putting it in the recycling bin, where it belongs, you keep the letter and are surprised to find out that all five predictions are correct. The following month, you get another letter from the same person, again predicting the results of five sports games to take place over the next month. Now you are curious, so again you keep the letter. Again, you find that all five are correct. The following month, you get yet another letter from the same person, asking if you would like to pay for more predictions. Since the predictor has a perfect 10/10 record, maybe this is a good idea?

Not so fast! What you should try to determine is whether other people have received very similar letters. Here is a possible scenario: for the first mailer, the sender prepared 32 copies of each of 32 different letters, containing all possible results of the five games. Then the sender selected 1024 people and mailed a letter to each of those 1024 people. So, 32 people, including you, got the letter with all correct predictions. For the second letter, the sender wrote just 32 letters, each one containing a prediction for all five games the following month. The sender then mailed one of these letters to each of the people who had received the perfect mailer the previous month. The third letter, offering to sell predictions, was only sent to you, since you were the only one who got the perfect predictions the first two times.

(Maybe the people who got letters with 9/10 right also got this third letter. After all, 9/10 is pretty good too!)

In other words, the sender has no ability whatsoever to predict. But you don't know that, because you only got to see one letter, out of all 1024. Therefore, you should be extremely suspicious of such advertising if you do not know the mechanism by which it was produced.

Or, take the following similar scenario. Let us suppose, for some rather unfathomable reason, I feel the need to make a lot of money in a short amount of time. My training as a mathematician has not taught me much about how to do this, so I do the logical thing under the circumstances: I plan to ask the experts. I assemble a list of many people who have made a lot of money in a short amount of time, and I arrange to interview all of them about their behavior and lifestyle choices. Then, I interview them all and look for patterns. And I find a very noticeable pattern in their behaviors shortly before they made their fortunes. Naturally, I choose to mimic their behavior: I buy a lottery ticket.

Oops. That wasn't what we wanted. My mistake was that I failed to interview all the people who did *exactly the same thing* but didn't win. I was the victim of survival bias.

In the context of zero-knowledge proofs, a dishonest Peggy may be able to pull a similar stunt. She doesn't know the secret she's supposed to know, but she wants to convince someone that she does anyway. So, she plays one of these probabilistic games, in which she can only trick Victor with small probability when she doesn't know the secret. Since she doesn't know it, she probably fails. But she is undeterred: she just tries again, this time with a different verifier. She probably fails again, so then she moves on to a third verifier. Eventually, just by chance, she manages to trick someone. Success!

Similarly, a Peggy with a time machine can pull off the stunt with against a single verifier. Whenever Victor rejects her claim of knowing the secret, she uses her time machine to "undo" the last interaction. We conclude that Victor's confidence in a zero-knowledge proof should depend on what he believes Peggy's power to be. If he thinks she has a time machine, then he won't accept the protocols we have already discussed, and they have to come up with a more convincing protocol. We leave it as an exercise to the reader to try to come up with time-machine-resistant zero-knowledge proofs. (See problem 14.)

## 7. Commitment schemes

Alice and Bob are having a dispute and would like to settle it by the canonical fair and random process: flipping a coin.[6] If they were in the same room, they could just flip a coin, with (say) Alice winning the dispute in case of heads, and Bob winning in case of tails. However, they are thousands of miles apart, and they are only talking on the telephone. Furthermore, they do not have fancy modern phones with videocameras.

What can they do? One option is for Alice (say) to flip a coin, and she wins if the coin lands on heads and loses if the coin lands on tails. But this is not satisfactory from Bob's perspective: there is nothing to stop Alice from *claiming* that the coin landed on heads, regardless of the actual result. In fact, she could easily make such a claim without even flipping a coin at all.

Another thing they can try is for Bob to decide (secretly) whether he picks heads or tails, and after Alice announces the result of the coin flip, he reveals to her which he has chosen. But now we have the opposite problem: Bob can lie. If the coin lands on heads, then he can

[6]They haven't learned about the hacks we discussed on page 4.

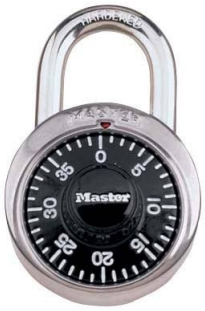


**Figure 8.** A combination padlock

claim to have chosen heads, and if the coin lands on tails, then he can claim to have chosen tails. So, this scheme is also not satisfactory.

Instead, we need a way for both Alice and Bob to lock down their choices *before* a certain key piece of information is revealed. We do this by relying on a computationally difficult problem. Many such computationally difficult problems admit similar commitment schemes, but we will discuss one based on factoring.

Rather than flipping coins, Alice picks two large prime numbers, $p$ and $q$, such that one of them is $1 \pmod 4$ and the other is $3 \pmod 4$. Alice computes $n = pq$ and reveals this number to Bob. However, she does not reveal $p$ or $q$. Since $n \equiv 3 \pmod 4$, Bob can already verify that one of $p$ and $q$ is $1 \pmod 4$ and the other is $3 \pmod 4$. (Well, sort of: he cannot verify that $n$ has only two prime factors. But more on that later.)

The number $n$ is Alice's part of the commitment scheme. Now, it's Bob's turn to do something. He knows that one of the prime factors of $n$ is $1 \pmod 4$ and the other is $3 \pmod 4$, but he doesn't know whether it is the smaller or larger one that is $3 \pmod 4$. So, he makes a guess. That guess is Bob's part of the commitment scheme.

Now, Alice reveals the two numbers $p$ and $q$, and both Alice and Bob can easily check whether Bob's guess was right or not. Furthermore, they can check whether Alice cheated or not. For example, Bob can verify that $pq = n$. He can also verify that $p$ and $q$ are both primes. (Remember that primality testing can be done in polynomial time, i.e. quickly!)

## 8. Problems

(1) Victor is colorblind. Peggy has two billiard balls, which are identical except for color: one is green and one is red. Peggy would like to convince Victor that they are different, without telling him which one is which. Design a protocol for them to use.
(2) Victor has two combination padlocks like the one in Figure 8, which are identical except for the combination used to open them. Peggy claims that she knows the combination to one of them. Design a procedure that allows her to convince Victor that she knows the combination to one, without him learning what the combination is, or which one she knows the combination for.
(3) Peggy has two identical padlocks, and she claims that they both have the same combination. Design a procedure whereby she can convince Victor of this, without him learning the combination.
(4) Peggy claims to know the number of coins in a large jar. Come up with a protocol whereby she can convince Victor of this. (You may suppose that she can instantly determine the number of coins in the jar at all times.)

(5) (a) Given a deterministic oracle that, for every $a$ and $n$, either returns some $t$ so that $t^2 \equiv a \pmod{n}$ or promises that no such $t$ exists. ("Deterministic" means that the oracle always returns the same answer given the same input.) Explain how to use this oracle to factor numbers quickly. (Hint: if $a$ is a square modulo $n$, how many square roots does $a$ have?)
   (b) Given an oracle that returns the prime factorization of a number $n$, explain how to determine, given $a$ and $n$, whether there is some $t$ such that $t^2 \equiv a \pmod{n}$, and if such a $t$ exists, to find an example. (You may assume that you know how to take square roots modulo $p$ when they exist, even if this is actually false. At any rate, there *are* polynomial-time algorithms for doing that.)

(6) Let $G = (V_1, E_1)$ and $H = (V_2, E_2)$ be two graphs, so that $E_i$ is a set of pairs of vertices in $V_i$, for $i = 1, 2$. We say that $G$ and $H$ are isomorphic if there is a bijective function $\sigma : V_1 \to V_2$ such that $\{v, w\} \in E_1$ if and only if $\{\sigma(v), \sigma(w)\} \in E_2$. (That is, $G$ and $H$ are "the same" up to relabeling the vertices.) Find a zero-knowledge proof for determining that $G$ and $H$ are *not* isomorphic. (You may assume that Peggy knows an actual isomorphism of the graph, and furthermore than she can solve the graph isomorphism problem for *every* pair of graphs.) This is interesting, because there does not seem to be a general way of producing a short proof that two graphs are not isomorphic, i.e. graph non-isomorphism is not believed to be an NP problem. By contrast, one can produce a short proof that two graphs *are* isomorphic by simply writing down an isomorphism.

(7) Find a zero-knowledge proof for showing that two graphs *are* isomorphic.

(8) Consider the following attempt at a commitment scheme. Alice chooses a prime $p$, a primitive root $g$ modulo $p$, a number $x$, and computes $h \equiv g^x \pmod{p}$. She then tells Bob $p$, $g$, and $h$, and asks Bob whether $x$ is even or odd. Why is this a bad idea?

(9) In the previous problem, suppose that Bob is only supposed to have a $1/3$ chance of winning the commitment, and Alice asks him to guess $x \pmod 3$. Is this a good commitment scheme?

(10) Modify the factorization-based commitment scheme to the case where Bob is supposed to have an $r/s$ chance of winning, where $r/s$ is a rational number in lowest terms.

(11) Find a different zero-knowledge proof for sudoku.

(12) Find a zero-knowledge proof for solving Rubik's cubes.

(13) A graph $G$ is said to have a Hamiltonian circuit if there is a sequence $v_1, v_2, \ldots, v_n, v_1$ of vertices, starting and ending with the same vertex, such that
   (a) each vertex of $G$ is on the list $v_1, \ldots, v_n$ exactly once,
   (b) for each $i$, there is an edge between $v_i$ and $v_{i+1}$, and also an edge between $v_n$ and $v_1$.

   Determining whether $G$ has a Hamiltonian circuit as an NP-complete problem. Find a zero-knowledge proof for determining whether $G$ has a Hamiltonian circuit. (You may assume that Victor cannot solve the graph isomorphism problem.)

(14) Come up with a zero-knowledge proof for graph 3-colorability that Peggy and Victor can enact if Victor believes that Peggy has a time machine.

Thanks to Simon Rubinstein-Salzedo for course materials