Foolproof Authentication Via Quantum Digital Signatures

Qifan "qxi" Xi (20262492 - CS467)

April 15, 2013

Abstract

Authenticity is a vital pillar in information security - that is, to be able to verify that a message sent from a user was not forged or modified by an attacker. Modern classical security implementations employ one-way hashing functions for this purpose. It can be shown that while classical message authentication schemes are computationally difficult to attack, they are not information-theoretically secure. Furthermore, the computational requirements of a successful attack can be reduced via cryptanalysis[2].

Gottesman and Chuang's paper Quantum Digital Signatures describes an authentication scheme using quantum bits to provide an information-theoretically secure means verifying messages.

1 Introduction

Digital signatures are analgous to traditional signatures in that they present a proof of the authenticity of a document. Most digital signature schemes also attempt to guarantee that a document has not been tampered with by an attacker.

On the Internet, establishing a secure channel between two parties can be done by using an authenticated channel to exchange private-public key pairs. The existing public key infrastructure (PKI) is built around a series of trusted certificates- means by which to verify the signatures of messages to verify authenticity.

These authentication schemes depend on an attacker's inability to solve a difficult mathematical problem. Chiefly, the problem of large number factoring is commonly used[1] but is vulnerable to an attack by a quantum computer[3]. Other schemes[4] offer resistance to attacks by known quantum algorithms, but still provide only computational security.

Gottesman and Chung give a cumbersome, but effective, method of message verification using quantum bits that is information theoretically secure. A proof of its safety against forgery is provided. A method by which to distribute the keys necessary for signature verification is also presented. Ideas for further exploration and additional readings presented as well.

2 Preliminaries

2.1 A Few Reminders...

- **No-Cloning Theorem**: It is not possible to create identical copies of an arbitrary unknown quantum state[5].
- **Holevo's Theorem**: Measuring a single quantum bit yields at most one classical bit of information from that state[6].

2.2 Lamport's Signature Scheme

To begin, we will describe Lamport's Signature, a classical method for constructing a digital signature. Suppose we have a function f where it easy to compute $f(x)$ given x, but very difficult to reverse this procedure. We consider this to be a one-way function. Alice wishes to send messages to Bob of length L bits. From i between 1 to L, Alice chooses $x_{i,0}$ and $x_{i,1}$ from the domain of f randomly and then computes $z_{i,0} = f(x_{i,0})$ and $z_{i,1} = f(x_{i,1})$.

Alice's private key, which she uses to sign messages, are the $2L$ pairs $x_{i,a}, x_{i,b}$. Her public key, which Bob (and others) can use to verify her message's authenticity, are the $2L$ pairs $z_{i,a}, z_{i,b}$.

To sign each bit m_i of a message $m = m_1, m_2...m_L$, Alice presents either $x_{i,0}$ (if $m_i = 0$) or $x_{i,1}$ (if $m_1 = 1$) for the signature bit s_i. She then concatenates all L signature bits $s = s_1, s_2...s_L$ and sends it along with the message m.

Bob can verify that Alice was the author of m by checking that $f(s_i) = z_{i,m_i}$. If an attacker wants to forge or modify a message, she would have

to reverse the function f to recover the original $x_{i,0} and x_{i,1}$ usable to sign a different message.

Unfortunately, since Alice's signatures s_i are directly sourced from her private key x_i, it would be best for her not to resuse the same private key. If she does, an attacker can keep track of her signatures s_i and which bit in m_i they correspond to, eventually reconstructing Alice's private key in full. This is an accepted limitation of Lamport's Signature.

2.3 Quantum Fingerprinting

We also introduce the concept of Quantum Fingerprinting as described by Buhrman, Cleve, Watrous, and de Wolf[8]. Classsical fingerprinting is a technique to shorten the length of messages into fingerprints, while still enabling equality tests of the original messages via only the fingerprints. A good fingerprinting scheme aims to considerably reduce the size of the original message required for equality testing while introducing only a small probability for error. They also serve as good one-way functions, a property that we will later exploit.

The quantum fingerprinting scheme described by Buhrman et al is as follows: Alice and Bob are holding length L messages a and b, respectively, and wish to verify that $a = b$ or $a \neq b$. However, L is very big, they must send information only to a third party, Cleopatra, who will perform the verification for them, and they wish to be economical. To minimize the amount of information sent to Clepatra, Alice and Bob each compute a string of qubits of length $O(log_2(L))$ as fingerprints and send them to Cleopatra.

Recall that given a single qubit, we can construct an arbitrarily large set of distinct quantum states using only that qubit. Instead, we will endeavour to create a set of 2^N states using $O(log_2(N))$ qubits with the restriction that each state is very nearly orthogonal to another.

A complete description of the process is tangential to the purpose of this paper, but the summary is as follows: assume that for a fixed $c > 1$ and $\delta < 1$ we have an error correcting code $E : \{0,1\}^n \to \{0,1\}^m$ for each n where $m = cn$ and such that the distance between distinct codewords $E(x)$ and $E(y)$ is at least $(1 - \delta)m$. Now, for any choice of n, we define the state $|f_x\rangle$ as

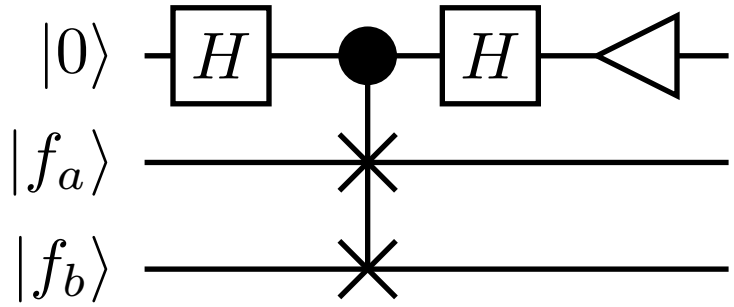
$$|f_x\rangle = \frac{1}{\sqrt{m}} \sum_{i=1}^{m} |i\rangle \, |E_i(x)\rangle$$

$|f_x\rangle$ is our signature for x. In general, we have a function that takes n classical bits and outputs $O(log_2(n))$ quantum bits.

2.4 Quantum Fingerprint verification

Now that we can produce quantum fingerprints for a and b, $|f_a\rangle$ and $|f_b\rangle$, how can Cleopatra compare them to verify that in fact $a = b$? Buhrman, Cleve, Watrous, and de Wolf present a quantum circuit that can perform such a test using a controlled Fredkin (swap) gate. This so-called "swap test" is illustrated in Figure 1.

Figure 1: Swap test for checking qubit equivalence



By tracing out the execution of this circuit, we get the final the state before the measurement as:

$$\frac{1}{2}|0\rangle(|f_a\rangle|f_b\rangle + |f_b\rangle|f_a\rangle) + \frac{1}{2}|1\rangle(|f_a\rangle|f_b\rangle - |f_b\rangle|f_a\rangle)$$

Measuring the first qubit of this state results in 0 if $|f_a\rangle = |f_b\rangle$. Additionally, it also results in 0 with probability at most $(1+\delta^2)/2$ if $|\langle f_a|f_b\rangle| \leq \delta$. Otherwise, the result of the first qubit is 1. This only happens if $|f_a\rangle \neq |f_b\rangle$. Ergo Cleopatra has an equality tests that works on the fingerprints, although with error probability $1/2(1+\delta^2)$.

3 Quantum Signature Scheme

Suppose we make use of the results of Quantum Fingerprints to generate a quantum state $|f_k\rangle$ of n qubits for each classical bit string k of length L. Thanks to Buhrman et al, we know that L can be as large as $O(2^{2^n})$

which nicely bounds the number of qubits required. We also have a means of equality checking via the aforementioned swap test. We now have everything needed to begin describing a secure signature scheme.

3.1 A Naive Approach Using Lamport's Scheme

One obvious approach is simply to use our mapping of k to $|f_k\rangle$ and drop it in place of a classical hash function in Lamport's Scheme.

We fix message length L, per-bit signature length n, and mapping k to $|f_k\rangle$ and distribute these properties to all parties as part of the protocol. As usual, Alice generates her private keys $x_{i,0}$ and $x_{i,1}$. She then publishes $|f_{x_{i,0}}\rangle$ and $|f_{x_{i,1}}\rangle$ as her public keys.

To sign a message $m = m_1, m_2...m_L$, she signs bit m_i with $s_i = x_{i,m_i}$. This is identical to our description of our earlier description of Lamport's Scheme. Bob, wishing to verify m with the signature s, uses the verification test to see if $|s_i\rangle |f_{x_{i,m_i}}\rangle$ maps back to $|s_i\rangle |0\rangle$.

Unfortunately, we have a problem with this naive approach. Namely, Bob would like to show Cleopatra that Alice's message is authentic- but Alice's public key is consumed (collapsed) in the quantum verification test! Furthermore, the verification test fails anyways with a small probability! We quickly abandon this approach and dedicate the rest of this paper describing Gottesman and Chuang's quantum signature protocol.

3.2 The Quantum Signature Protocol

3.2.1 Definitions

We define a one-use digital signature as the following: Alice holds a private key and distributes corresponding public keys to her recipients. She can then use her private key to sign a single message m with the signature $s = f(m)$. Her recipients, upon receiving (m, s), can evaluate the message and signature with their public keys to come to one of three conclusions:

- **1-ACC**: The message is authentic. It can also be shown to another verifier who will come to the same conclusion.
- **0-ACC**: The message is authentic. However, another verifier may not come to the same conclusion.

- **REJ**: The message is not authentic.

We require that any recipient of an authentic (m, s) pair reaches conclusion 1-ACC.

3.2.2 Security Criteria

Classical signature schemes are typically required to provide the following features, which we will prove that the quantum protocol also provides.

- **Authentication**: A receiver must be able to verify the sender of a message. A signature cannot be easily forged by an attacker to falsely authenticate arbitrary messages.
- **Integrity**: The signature must protect the integrity of the message. An attacker must not be able to modify a signed message and have the signature remain valid.
- **Non-repudiation**: The sender, after having signed a message, should not be able to later deny having signed it.

3.2.3 Protocol Specification

Alice begins by choosing M pairs of L-bit strings $\{k_0^i, k_1^i\}$ randomly, with $1 \leq i \leq M$. We will be using M keys to sign each bit, with k_0^i to sign 0-bits and k_1^i to sign-1 bits. This is Alice's private key.

She produces her public key $\{|f_{k_0^i}\rangle, |f_{k_1^i}\rangle\}$ for each i using our quantum one-way function described earlier. The public keys are distributed to all of Alice's potential recipients. We may assume that Eve, who wishes to attack this protocol, also manages to receive a genuine version of this public key. Alice ensures that she distributes at most T public keys, with $T < L/n$. This limits the size of Alice's audience, but is necessary for our protocol's security proof.

The recipients of Alice's messages, including Bob and Cleopatra, know how to implement our quantum one-way function. They also know values c_1 and c_2, which are used in the protocol as thresholds for accepting and rejecting a message. So far, our protocol sounds very similar to the naive approach using Lamport's scheme.

To send a message $m = m_1, m_2 ... m_L$, Alice signs each bit m_i with bits $k_{m_i}^i$ for $1 \leq i \leq M$.

To verify a message m, Bob can compute $|f_{k^i_{m_i}}\rangle$ for each i and use the swap test to compare them to his copy of the public keys. He counts the number of keys, p, that pass the swap test. If $p \leq c_1 M$, then Bob regards the message as valid and transferable (1-ACC). If $p \geq c_2 M$, then the message is invalid (REJ). If $c_1 M < p < c_2 M$, then the message is valid but not necessarily transferable to other recipients (0-ACC). All used and unused keys are discarded.

Recall that our quantum equality test, the swap test, will with a small probability report that two qubits are equal even though they are distinct. The c_2 threshold is designed to prevent an accidental mis-classification of a valid message. The c_1 threshold can be used if the protocol is used across a noisy channel, or can be set to $c_1 = 0$ if no transmission errors will occur. As we will show in our proof, c_2 can protect us against forgeries while $c_2 - c_1$ prevents repudiation by Alice.

3.3 Key Distribution

Before continuing with a proof of the protocol meeting our security criteria, we give a method for distributing Alice's public keys to her recipients. As with public key distribution on the Internet, we can assume the existence of a trusted third party with a secure link to each recipient that can forward Alice's keys to them. We look at ways in which Alice may attempt to cheat her recipients by distributing different keys, or special keys, in an attempt to sabotage the non-repudiation requirement of our protocol. Finally, we present ways in which our trusted third party can prevent these cheating measures.

If Alice uses our signature protocol to broker a contract between herself and Bob, with Cleopatra as a neutral observer, she can try to repudiate any offers that she sends to Bob with her signature. One way to do this is to simply send different keys to Bob and Cleopatra, claiming them both to have received her valid public key. Alice signs a document with her private key and later recants, arguing to Cleopatra that her signature was forged. When Bob checks the signature against his copy of the public key, the measurement returns 1-ACC. However, Cleopatra checks it with her own (false) key and measures REJ. Alice has cheated her way into backing out of a deal.

Using a neutral third party, Dan, to distribute Alice's public keys, can prevent this form of cheating. Before handing Alice's public keys to Bob and Cleopatra, Dan simply needs to perform the swap test between the keys to

check if they're identical. If p pieces of the keys fail with $p \geq c_2 M$, then Dan concludes that the keys are not identical and cannot trusted.

Alice may then attempt to fool Dan as well. She is free to prepare the state $|\psi\rangle_B |\phi\rangle_C + |\phi\rangle_B |\psi\rangle_C$ which is symmatric and invariant under swaps. Dan will claim that the keys are identical and Bob and Cleopatra will be none the wiser. Fortunately, Alice cannot be certain who of Bob and Cleopatra receives the valid key and who receives the invalid key. Additionally, recall that we use a key of M parts, of which the valid one of each part is randomly distributed between Bob and Cleopatra. Bob tests all M of his keys with a message bit and sees p_B passing; likewise, Cleopatra sees P_C. When M is very large, the difference $|p_B - p_C|$ is $O(\sqrt{M})$ with high probability. For Bob and Cleopatra to claim definitive, but different, results requires that $p_B < c_1 M$ (1-ACC) and $p_C > c_2 M$ (REJ). $|p_B - p_C| > |c_1 M - c_2 M|$ is *very* unlikely to occur.

In their paper, Gottesman and Chuang also propose a method of distributed key verification without a third party Dan. In this version of the protocol, Alice distributes two keys each to Bob and Cleopatra, with a total of four keys that are ostensibly Alice's identical public key. Bob and Cleopatra first use the swap test to check that their own two keys are the same, and then exchange one of their two keys with each other. They then perform another swap test to complete the verification. If any of the swap tests fail, the protocol is aborted. This ensures that all four keys that Alice distributed are identical. Optionally, the extra keys are destroyed to reduce the number of copies in existence for a forgery attack (the importance of this is explained later).

Given an honest Alice and a malicous Bob or Cleopatra, the worst case scenario results in a protocol abort every time. Although the signature scheme is not compromised, it is an effective denial of service attack that prevents Alice from communicating with her recipients. If Alice is the malicious individual, the distributed verification process aborts the protocol and prevents her from using it to repudiate her signed messages. Only if Alice and Bob are colluding against Cleopatra can this distributed verification process fail: Alice gives one valid and one invalid key to Bob, and two valid keys Cleopatra. Bob lies about the result of his swap tests, but exchanges his valid key with Cleopatra so that both of Cleopatra's swap tests come back clean. However, we can accept that this protocol cannot provide security if all participants are conspiring to swindle a single individual.

3.4 Proofs Of Security

3.4.1 Against Forgeries

Suppose it is Eve's objective to make a forgery of Alice's signature and sign an arbitrary message of her choosing. She cannot reverse the quantum one-way function f with a single copy of the public key, but we assume that she manages to obtain all T copies of Alice's public keys.

Holevo's theorem tells us that Eve can acquire at most n classical bits of information from measuring n quantum bits. Therefore, Eve acquires Tn bits of information about each private key string $k^i_{m_i}$. But Alice made sure that she only distributed T copies of public keys with $T < L/n$. Eve has insufficient information to reconstruct Alice's private key.

Thus, we have shown that our protocol covers two of our three security criteria: authenticity and message integrity.

3.4.2 Against Repudiation

As mentioned in our discussion of key distribution methods, a good defense against repudiation is the use of swap tests on Alice's public keys- either by a third party or in a distributed manner. Recall our previous statement: that for Alice to fool Bob and Cleopatra requires $|p_B - p_C| > |c_1 M - c_2 M|$, which we said is "very unlikely to occur". We want to determine exactly how unlikely occurance actually is.

In their paper, Gottesman and Chuang perform this analysis with the assumption that a distributed key verification method is used. The proof is a bit too lengthy and complex to include verbatim, so we will instead present only the major ideas in their proof.o

We want to compute q_{cheat}, the probability that Alice can pass all swap tests while $|p_B - p_C| > |c_1 M - c_2 M|$ - meaning that Bob and Cleopatra disagree vehemently about the validity of the message. We examine a global pure state $|\psi\rangle$, which describes all the public keys that Alice distributes, including any state entangled with those keys.

Assuming that Alice prepares $|\psi\rangle$ with the property that all keys pass the first (distributed) swap test, we consider what happens only after Bob and Cleopatra exchange keys and perform the second swap tests. For each set of keys in existence, the global state is in a superposition of two types of keys: type-1 passes the swap test, but results in Bob and Cleopatra agreeing to the

validity of the keys; type-2 fails the swap test, but measurements with those keys reveals a disagreement between Bob and Cleopatra.

We then decompose the global state $|\psi\rangle$ into $|\psi\rangle = |\psi_1\rangle + |\psi_2\rangle$ where, to respect the requirement to pass the first swap test, each $|\psi_1\rangle$ consists of at most r type-2 tensor factors, $r = cM$ for some $c > 0$. $|\psi_2\rangle$ consists of term with more than r type-2 factors.

For a sufficiently large enough M and small enough c, the probability that Bob and Cleopatra show drastically different results during verification using $|\psi_1\rangle$ is shown to be exponentially small in M. And by using a modification of the swap test (that is more lenient than the actual swap test) yet still fails each type-2 term with probability 1/2, we show $|\psi_2\rangle$ with r type-2 terms will pass with probability at most 2^{-r}. Ergo, it will pass with probability at most 2^{cM}, a number exponentially small in M.

Putting the results of the two together gives us a q_{cheat} of $O(d^{-M})$ for some $d > 1$. Thus, the protocol ensures that Alice cannot likely cheat the system and cause Bob and Cleopatra to come to opposite conclusions regarding the validity of her message.

4 Conclusion

We have successfully presented a digital signature scheme that provides information theoretical security. Applications are many. Establishing an authenticated channel is often the first step to building a secure link over an insecure medium. Signatures verifiable by third parties have utility in legal settings. Overall, digital signatures are a vital cryptographic primitive that benefits from a quantum equivalent.

Unfortunately, the protocol presented does have its downsides: security becomes compromised if too many public keys are distributed. Each signature's public/private key pair is only usable for one message. And finally, the length of keys scales linearly with the length of messages.

Nevertheless, Gottesman and Chuang's paper presents a good starting point for further developing the field of quantum cryptography. For more recent advancements in the field of quantum digital signatures, consider additional reading[9][10].

References

[1] R. Rivest, A. Shamir, L. Adleman *A method for obtaining digital signatures and public-key cryptosystems*. Communications of the ACM 21.2 (1978): 120-126.

[2] X. Wang, Y. L. Yin, H. Yu *Finding collisions in the full SHA-1*. Advances in Cryptology-CRYPTO 2005, Springer Berlin Heidelberg, 2005.

[3] P. W. Shor *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*. SIAM journal on computing 26.5 (1997): 1484-1509.

[4] D. J. Bernstein *Introduction to post-quantum cryptography*. Post-quantum cryptography. Springer Berlin Heidelberg, 2009. 1-14.

[5] W. K. Wootters, W. H. Zurek *A single quantum cannot be cloned*. Nature 299.5886 (1982): 802-803.

[6] A. S. Holevo *Bounds for the quantity of information transmitted by a quantum communication channel*. Problemy Peredachi Informatsii 9.3 (1973): 3-11.

[7] D. Gottesman, I. Chuang *Quantum digital signatures*. arXiv preprint quant-ph/0105032 (2001).

[8] H. Buhrman, R. Cleve, J. Watrous, R. de Wolf *Quantum fingerprinting*. Physical Review Letters 87.16 (2001): 167902.

[9] X. Lu, D. Feng *Quantum digital signature based on quantum one-way functions*. Advanced Communication Technology, 2005, ICACT 2005. The 7th International Conference on. Vol. 1. IEEE, 2005.

[10] X. Wen, X. Niu, L. Ji, Y. Tian *A weak blind signature scheme based on quantum cryptography*. Optics Communications 282.4 (2009): 666-669.