Quantum Qwyit™

Discussion

Version 1.2 Aug 1, 2023

Copyright Notice

Abstract

This paper provides a discussion of the Cryptography industry; especially in relation to the upcoming standardization of algorithms for Post-Quantum Cryptography (P-QC). This point in time is a critical juncture for the entire science – and it doesn't seem the industry is prepared. What is being touted as preparation for the future, seems to be nothing more than repetition of the past. All of this, and how Qwyit™ fits into it, is discussed.

Contents

Quantum Qwyit®

This document provides a discussion of Post-Quantum Cryptography (P-QC) (also called Quantum Safe, Quantum Resistant, etc.), and its relation to the *Qwyit*™ protocol. For this document, as well as in general discussions/communications, *Qwyit*® refers to the Company, *Qwyit*™ refers to the complete Authentication and Data Encryption protocol and *QCy*™ refers to the encryption cipher. Definitions and operation of all of our security technology can be found in their respective, current *Reference Guides* available from Qwyit®; go to www.qwyit.com.

Introduction

Cybersecurity needs Jerry Maguire: a heart-felt moral epiphany, eloquently expressed about not missing this crucial time-is-now opportunity to fix itself…especially since there's zero awareness within the industry that it needs fixing.

This paper is an attempt to do just that: Somehow convey to those in the Cybersecurity industry that at this juncture between 'current-soon-to-be-made-insecure' industry methods and the blind-faith acceptance of the selection and imposition of 'next generation Post-Quantum Cryptography' (P-QC), if there isn't a sincere, fully committed revisit to the fundamental purpose of the entire science, the industry will continue in overly complex expert-ism and another entire generation of *imperfect* methodologies. If this happens, there'll be no stopping the mind-boggling, ever-escalating cybercrime ($6*TRILLION* a year in 2022…and rising).

Since the Digital age inception 30+ years ago, isn't continued adherence to methodologies, that haven't actually worked because of performance, application, complexity and other issues, enough to open your eyes to other possibilities? Yes the methods weren't 'broken'…but they most certainly weren't *complete*. And if it seems like a good idea to you to select and impose another set of incomplete methods, and then mindlessly stumble into a new generation of platform problems leading to another 30+ years of failure…

Then no need to read any further: Jerry (and I) are looking for folks who want to actually succeed with cryptography. This paper will take you back to the beginning, articulate exactly how the Cryptography Train got off the tracks, show how it has continued full steam ahead…and is about to tear through another technology realm without any introspection or consideration of what actually is necessary to fix all of the incomprehensible cybercrime.

Incredibly, the fix isn't complicated. It isn't hard to do. The answer's been hiding in plain sight…and, lo and behold, there's actually an example methodology instantiation that works – and can be fully investigated – because it's all available to use. Right Now.

If you'll just set aside everything you think you know…let me walk you through the past – and into the future with *the real time, real world solution to Post-Quantum Cryptography…and beyond.*

Cryptography History – Part 1

We'll begin with a quick recap of cryptography's history:

- Everyone wants to send private information publicly
- Throughout history, thousands of methods (ciphers) have been tried; some longer-lasting than others…though they all eventually fell victim to cryptanalysis. They all yield a way to break the encryption in the then-current definition of 'real time'
- The search continued until 1917, when Gilbert Vernam published the first version of what shortly became the One Time Pad (OTP)
- This cipher proved unbreakable; and in 1948, Claude Shannon published a paper (Communication Theory of Secrecy Systems) that mathematically proved the OTP as an example of what he termed, and is now called "Perfect Secrecy": no matter how much material an eavesdropper collects, they can never break the cipher because multiple different plaintexts encrypt to identical ciphertext
    - *This means that the OTP, Perfect Secrecy (PS), is not computationally bound – it doesn't matter the 'platform' or method – PS was, is, and will always be, 100% safe under any Binary, Quantum, Artificial Intelligence, or future-unknown computing system* (Yes, that's a bit of a spoiler…and slightly ahead of ourselves, but…better to tag that while we're here…and we'll refer back to it shortly :)
- Perfect Secrecy (also called Perfect Security) completed the endless cryptographic search
- But…since you have already seen the movie (as you're now in it!), you know that Cryptography didn't stop in 1948, did it?

We'll slow the historic progress notation down a bit right here…because this is when the CryptoTrain went off the tracks. This is when The Current Status Quo began; and the addition of 75+ years of layer upon layer of 'complex incomprehensible stuff' is exactly why we need Jerry's (this) Magic Paper to right all the 'wrongs' and put the train *not only back on the tracks, but pull it into the station and actually, finally End The Cryptography Story*.

What happened next, sadly, happened *in Claude's paper!* And like any good entertainment that keeps you hanging onto the edge of your seat…we're going to pause right here…and *flash forward* to what's happening *NOW* in cryptography that is opening a future chasm so wide, we'll never get back on track if we don't immediately, and forcefully, stop the train. We've got to stop going down the same mistaken (*fatal*) path, because we might never get back if we don't.

Post-Quantum Cryptography (P-QC)

You might be aware that 'all the current security stuff is going to be useless – completely broken – when those new-fangled *Quantum Computers* (QC) arrive'. You might also have heard 'not to worry, cuz those crypto experts are creating new Quantum-Safe stuff that works as good as what we've got now' before QCs are on the shelf at the Walmart.

This is the crux of the matter – and the reason for this paper: 'as good as what we've got right now' isn't actually very good. Cyber attacks are at an all time high, and cybercrime is too - $6*TRILLION* is a lot of crime! So the first inkling as to how far off the tracks we've gotten is that *IF* the new stuff is being pursued:

- By the same Crypto experts
- Using the same Crypto approach
- With the same Crypto foundation

We're all in a heap of trouble. Because when Artificial Intelligence (AI) is running the network, doing the 'security work', and it's being attacked by AI-created weapons…$6Trillion will be a drop in the bucket – the bucket of pain, suffering and insecurity as expert systems battle expert systems…and the only losers will be us, the users.

The tragic part? Cryptographers don't care that $6Trillion happens every year – and they most certainly (correctly) know that *none of that $6T is because the underlying cryptography is being broken!* They'll accurately point out that 'networks are complicated – our stuff isn't the reason for the vulnerabilities'.

And here's where we need our first bit of 'magic' to happen in the field:

We need Cryptography – run by those expert cryptographers – to take responsibility for '*all the security in the digital world*' even though that isn't the same as 'the cryptographic fundamentals'. There's identity theft, network break-ins, stolen crypto-currency, etc., ETC. But…we need this to happen for two crucial reasons:

1. The cryptographic fundamentals are the *basis for all of the security in the digital world*
2. If the cryptographic fundamental properties were such that they could be used *everywhere, in everything and on all the time*, then all of the other 'security issues' would evaporate – instantly – and all those vulnerable attack points wouldn't exist
   a. Those properties are:
      i. Speed – fast enough to operate in every transaction
      ii. Size – small enough in code space, bandwidth, etc. to fit in every network device
      iii. Flexibility – apply to any and every trust model, to work in any business process
      iv. Security – Perfectly Secret, provide mathematically guaranteed protection, forever

Just imagine if every single digital transmission – *every bit sent by any device* – was authentic and 100% encrypted!! What does the Digital World look like? Just a mass of random bits. Everywhere, in everything, and on all the time. There's nowhere to break in…the laughable 'zero-trust definition' *is what all network security is supposed to have been all along!* Everything is authentic, everything is Perfectly Securely encrypted: *and we need Cryptography to take responsibility for the fundamental definition, creation and implementation of it all* in order to make it happen.

While I've battled this exact point for several years; presenting Qwyit™ against the backdrop of 'What we have is Good Enough'…the technology world has caught up to us all and changed the game. It's not just me saying what we have isn't good enough: P-QC has all of Cryptography admitting that it isn't.

This time, my message isn't a battle of Qwyit versus the Status Quo of Current – it's now an opportunity to wake up the industry to listen to Qwyit's value proposition in terms of a P-QC solution. As the industry is poised to make a uniform choice of New Algorithms that withstand the New Computing Platforms.

And, sadly, they're using their same old method of selection: making *assumptions* about the platforms' future capabilities, the network architectures and applications, the measure of strength of the proposed algorithms, the users, etc. You can read lots of places about what, exactly, P-QC is…here's an example. (If not familiar with it, please do go acquaint yourself…the future is about to happen!)

A simple question should be enough to get cryptography to stop, before it's too late, and change their approach:

> Can you find a single example of a document written about the current algorithms as they were being proposed in the 1970's that correctly envisioned the cryptographic topography of today?

You don't have to look…there aren't any. Doesn't that make you stop and instantly recognize that *right now isn't the time to make the exact same 30+ year mistake?*

Cryptography History – Part 2

Since Cryptography is completely deluded from 75+ years of thinking they know what they're doing, we'll need to lend more credence to the need to stop than that fact-slap.

Cryptography must take a brand new approach to selecting a new fundamental algorithm to successfully navigate a completely unknown future – and the evidence and facts necessary to get this to happen are already in Crypto's past: so let's jump right back where we left off in our recap of Cryptography's history…because that's *exactly where the new approach was already identified*:

- Claude Shannon's 1948 Secrecy Systems paper (Communication Theory of Secrecy Systems – See Appendix A for a Table of Contents), was composed of three parts:
  - *Mathematical Structure of Secrecy Systems* – He sets up and defines how he's going to present and prove Secrecy
  - *Theoretical Secrecy* – He takes a journey through systems, arriving at the definition and math proof of "Perfect Secrecy"…the 100%, unbreakable secret cipher (an example of which is the OTP)
  - *Practical Secrecy* – Uh-oh…this is where Cryptography has lost 75+ years! Because Claude himself assumed that since the definition of Perfect Secrecy requires a key as long as the message – and if you have to securely pre-share that long of a key – *it (supposedly) wouldn't be "practical" because you might as well share the message instead of the key!*

- 1948 – Today: Massive growth in the expert-ism (and number of experts) in attempting to define, create, deliver and standardize on *Practical Security…because 100% Perfect Secrecy was (supposedly) not 'practical'!*

Buried in the above is actually *the definition of Practical Perfect Secrecy* – Claude wrote it himself. When you read his paper (or just look at the Chapter headings in Appendix A), you'll see he got to Perfect Secrecy in Part 2 *Theoretical Secrecy* in Chapter 10.

But there are *10 more chapters*; which continue to pursue and define more *practical* ciphers and secrecy systems under the theoretical heading. He defines these *finite key* systems (obviously much more *practical* than Perfect Secrecy with its endless key)…and eventually (sadly, and it turns out *wrongly*), he leaves these theoretical systems, because in his words (this from his opening *Introduction and Summary* Chapter as the last paragraph of the *Theoretical Secrecy* part):

> *"*It is possible to construct secrecy systems with a finite key for certain "languages" in which the equivocation does not approach zero as *N*→∞. In this case, no matter how much material is intercepted, the enemy still does not obtain a unique solution to the cipher but is left with many alternatives, all of reasonable probability. Such systems we call *ideal systems*. It is possible in any language to approximate such behavior—i.e., to make the approach to zero of *H(N)* recede out to arbitrarily large *N.*"

Boom. Those highlighted practical, finite key *Ideal Systems* are defined as:

> "We will define an "ideal" system as one in which $H_E(K)$ and $H_E(M)$ do not approach zero as $N\to\infty$. A "strongly ideal" system is one in which $H_E(K)$ remains constant at $H(K)$."

This isn't a math or technical discussion – but those are there to let you be absolutely certain that Shannon *defined and discussed that Perfect Secrecy Systems could be made using finite keys*. They meet the provable Perfect Secrecy standard: *Strongly Ideal Systems* leave no possible decryption no matter how much material is captured. These systems are *not computationally bound*.

Those 10 theoretical chapters are the answer to all of modern Cryptography's problems…and they've been ignored for all these years because…well…Claude himself summarized it in the sentence that follows the above Summary:

> "However, such systems have a number of drawbacks, such as complexity and sensitivity to errors in transmission of the cryptogram.*"*

So off went Claude to writing about less-than-Perfect *Practical Secrecy* systems in Part 3; and all of Cryptography went off the rails with him.

[heavy, deep sigh]…

There's a simple reason for this so-far fatal error – and the one we *must* avoid going into P-QC: *none of these Cryptographers were Engineers*. The "drawbacks" and "complexity" are nothing more than solving the 'How To' part of the science: *What are new ways to design, build and test Finite Key Strongly Ideal Systems that work?!* Then we'd have *Perfect Secrecy – forever, in finite key (practical) ciphers!* No other science quits after finding some 'difficult' facts…and then scatters away because there's *engineering work to be done to make it all real!* We wouldn't have almost *every bit of today's technology without engineering.*

If someone engineers a practical, Strongly Ideal System that is forever not computationally bound, that meets the real-world definition of Perfect Secrecy (multiple plaintexts encrypt to identical ciphertext), then we'd have what was mentioned at the start of this paper as the 'spoiler':

> *Cryptography that is 100% safe under any Binary, Quantum, Artificial Intelligence, or future-unknown computing system*

There'd be no need to 'search' or 'submit' any new special P-QC algorithm: Cryptography would have the universal, forever algorithm: Cryptography's history would, finally, end:

Gilbert, Claude…Qwyit™.

Quantum Qwyit™

Let's summarize all of the above:

- Cryptography has Perfect Secrecy (PS), 100% unbreakable encryption – it's *not computationally bound and therefore forever safe under any computing platform*

- It was deemed impractical – so all of Cryptography went chasing imperfect (thought-to-be 'practical') systems (that don't really work; evidenced by $6Trillion in cybercrime)

- But it turns out PS was already detailed, exampled and proven possible in *practical* Strongly Ideal Systems – that are also unbreakable encryption…but no one has ever engineered one

- In order to meet today's *practical*, it isn't enough for the system to be Strongly Ideal (Perfectly Secure)…it needs to have specific properties: Fast, Efficient and Flexible

- Cryptography is again about to go chasing imperfect systems into a P-QC world that no one can even imagine…and do it in the same way they mistakenly handled the last 75 years: full of hubris and flat-out wrong assumptions

There's an *answer* to all of the above:

Someone (Qwyit!) *has built a Strongly Ideal System* that meets all of Shannon's requirements for Perfect Secrecy. We did it by making brilliant engineering achievements with simple cryptographic primitives that collect up all of Cryptography's properties (authentication, encryption, integrity, assurance, etc.), solve all of Crypto's vulnerabilities (any and every attack profile), additionally stop all of cybercrimes 'edge' attacks (identity theft, network break-ins, stolen crypto-currency, etc.) and can perform in a single machine cycle (Fast) in less than 300 lines of VHDL fitting on any and every network device profile (HW/SW/FW) (Efficient) for any federated trust model (Flexible).

Sounds too good to be true…but (and *even Jerry wouldn't believe it without seeing it*):

You can test it, because we've instantiated our protocol (providing Authentication and Encryption) on: an FPGA, built several SW apps (QFone – a 100% secure video&voice app for example), had it independently tested and verified – even submitted and presented at the first 2015 NIST Lightweight Cryptography conference, have reference documents…and patents.

There's no need to waste any more words on whether Qwyit™ is what it says it is…or whether it is what Cryptography needs to end this millennial search for safe public transmission of private information – and what it needs *right now for P-QC standardization*:

All Cryptography has to do is take a look. As this paper has outlined, hopefully in a Jerry Maguire-definitive way, what is needed right now is available. Gilbert, Claude…Qwyit™. Stop the runaway train and take a look. We'll provide anything anyone wants in order to make their own decision. A quick empirical 'proof' is provided in Appendix B.

Lastly, Patents

NIST is a US government agency; they're the ones doing the P-QC standardization. In order to submit an algorithm for this standardization process, the author(s) have to sign a release giving up all the rights to it. This means if one has patented their technology, it has to be assigned to the US Government. But there is precedent of standardizing on a patented technology – and no reason not to do that again, since in Cryptography it has already been established: the first version of SSL was already in a copy-written, patented, for-profit product (Netscape's Lotus Notes) when it was established as a 'standard security algorithm'. Doing it again with Qwyit™ isn't an issue for any users; we'll negotiate with everyone towards their best interests.

Last question: If the government only accepts 'free' work, do they actually get the best – or even just worthwhile – technology to consider?

Probably not…as you'll see:

Author's Note

Wouldn't it be wonderful if this paper actually Jerry Maguire'd Cryptography?! I'd love to read comments: "Well done! A succinct encapsulation of The Fall of Cryptography – with a complete solution for the Rise ready and waiting." But…more than likely, I'll get 'Nice Try – we're experts…leave this stuff to us.' About the only thing to do then, is to present one last Dire Warning That Ought To Be Heeded:

When NIST (the cryptographic experts) proposed the search for and submission of P-QC algorithms (from other cryptographic experts) for their P-QC Standardization Process, they wound down their 6-year analysis effort by selecting four algorithms. Awesome. Then when they announced them, one of them was…well…

> "A team of scientists report they were able to defeat one of the post-quantum safe algorithms that is still under consideration as part of the National Institute of Standards and Technology's (NIST) PQC program — and it only took one computational core on a PC working for about an hour." – *The Quantum Insider*, Matt Swayne 8/5/22

That didn't go well…did it?! Whatever the 'experts' said, or say, is irrelevant. Cryptography simply ***must not repeat the mistakes of the past***.

*There's only one sure way to present an algorithm to the future and have it maintain its security…and that's to have it be Perfectly Secret **now**.* Qwyit is the only candidate for acceptance in the P-QC world – there aren't any others. And as luck would have it…it's all ready for Cryptography to test, evaluate, benchmark and proliferate.

Conclusion

The purpose of this paper was to make a series of unequivocal points with relation to Post-Quantum Cryptography:

- There is only one absolutely sure way to design a P-QC algorithm guaranteed to work and be forever safe under any computing platform – it must not be *computationally bound*
- There is only one absolute definition of any cryptography that is not computationally bound: *Perfect Secrecy*.
- Cryptography has only one absolute definition of *Perfect Secrecy* in *practical* finite keyed systems: *Shannon Defined Strongly Ideal Systems*
- *Practical* includes real world *Speed*, *Efficiency* and *Flexibility*
- There is only one available *Practical Shannon Defined Strongly Ideal System*: Qwyit™. It is ready for inspection and standardization.

These unequivocal points were made by Cryptography itself – reviewing the science's history brings to light all of the supporting structure, data and events for proof.

Cryptography must not continue to disregard the obvious: the hubris, imperfect algorithms, convoluted and complex processes along with total disregard for the never-ending escalating cybercrime must stop. Now is a crucial juncture and opportunity to change course, get back on track, and deliver a real world solution to both the current out-of-control security infrastructure and the next generation Post-Quantum Cryptography brave new world. A solution exists for the entire past, present and future Cryptographic Universe; and it's 100% ready for inspection:

Qwyit™ is the first and only Shannon Strongly Ideal System:

- Shared, secret Authentication Keys
- One-way, underdetermined, never-ending, child key creating, linear mixing function (PDAF)
- Single machine cycle XOR of endless-key with every message
- 'No communication' Key Update anytime

Qwyit™ is a complete "Security System" authentication and encryption protocol. It has HW benchmarked "Real-Time" speed of 256-bit encryption in a single clock cycle. Any "Real-World" network data can be processed because the entire *QCy™* encryption engine can be flexibly modified for any bit configuration, size, length – including changing *on the fly* while in operation.

Qwyit™ has papers, benchmarks, descriptions, independent validation, prototypes (HW and SW) – even a phone app (QFone) demonstrating every aspect in Real-World, Real-Time implementation.

Qwyit™ delivers Perfect Security in a Shannon Strongly Ideal System – and provides Quantum (and every future computing platform) Safety.

Appendix A – Shannon's Paper

Here's a summary 'Table of Contents' for Shannon's Perfect Secrecy paper (he didn't have one in the original document; this is created from his content headings). The **bold typeface** highlights the sections pertinent to this document:

Communication Theory of Secrecy Systems

By C. E. SHANNON

Contents

PART I
MATHEMATICAL STRUCTURE OF SECRECY SYSTEMS

PART II
THEORETICAL SECRECY

PART III
PRACTICAL SECRECY

Appendix B – The small-key test

For those who want to 'jump to the end' of discovering/learning/realizing Qwyit really is Shannon's first, real Strongly Ideal System – and that such a system meets the Perfect Secrecy definition – simply perform the system in a small-key example. It's a stunning lapse of the entire science of cryptography that all proposed systems aren't presented in these encapsulated examples, since they instantly lay bare the vulnerabilities.

Here's a small key example of the RSA public key algorithm, showing that the security is 100% tied to the length of the keys – not to any actual security or mathematic protection in the method. It's uselessly bound to computational capability; as is any and every asymmetric key algorithm (and all current symmetric key systems also). This vulnerability makes them useless in a Post-Quantum Cryptography (P-QC) world:

> Google Search: small prime number example of RSA algorithm
> Result: **The RSA Algorithm 1. Select two large prime numbers p and q ...**

As shown previously, the new NIST selected algorithms fare no better – and we've pointed out the foolishness and fatal flaws in searching for algorithms that are computationally bound in any way.

So is QCy™ not computationally bound, and therefore Perfectly Secret (Secure) under any and every P-QC scenario – and without any need for any infrastructure change?

Here's a small-key test for QCy™:

QCy™ – A Strongly Ideal System

For review, the QCy™ method:

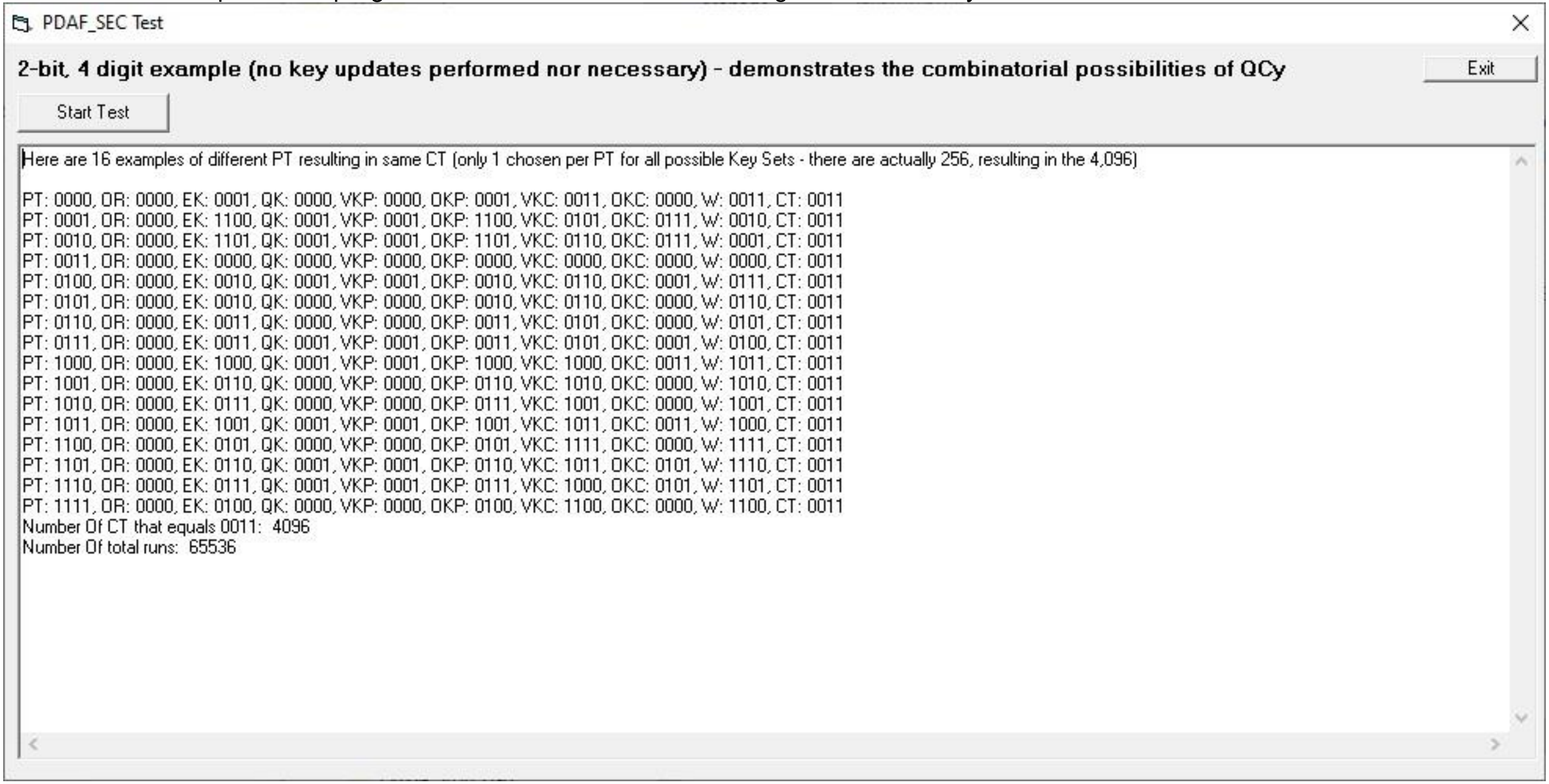
1. QK mod $\mathbf{OR} = VK^{P}$, PDAF(EK, VK^{P}) = VK^{C} and EK mod $\mathbf{OR} = OK^{P}$, PDAF(QK, OK^{P}) = OK^{C}
2. SELECTION: $VK^{Pc[1\ldots n]}{}_{Pv[1\ldots n]}$ mod $OK_{Po[1\ldots n]} = W_{1\ldots n^2}$ Where pointer Pv and Po increment +1 through the entire key length for each cycle pointer Pc. If repeating, substitute VK_N and OK_N for each new cycle*
3. CIPHER: $W_{1\ldots n^2} \oplus PT_{1\ldots n^2} = \mathbf{CT}_{1\ldots n^2}$ repeating w/next cycle selection if more PT
4. UPDATE: PDAF(OK^{C}, VK^{P}) = VK^{Next} and PDAF(VK^{C}, OK^{P}) = OK^{Next}

The method is underdetermined (consistent, with multiple solutions), mathematically unsolvable and complete – the security isn't theoretical, it's forever true.

The ‘Small Key’ example (2-bits) proves Perfect Security. The following is from the QCy™ Reference Guide, Appendix F:

“To prove that a scheme is perfectly secure, you must be able to show that for any pair of messages, the probability that they map to a given ciphertext is identical.” – CryptoProofs

Here’s a screen capture of a program written to execute a 2-bit, 4-digit version of QCy™:



2-bit, 4 digit example (no key updates performed nor necessary) - demonstrates the combinatorial possibilities of QCy

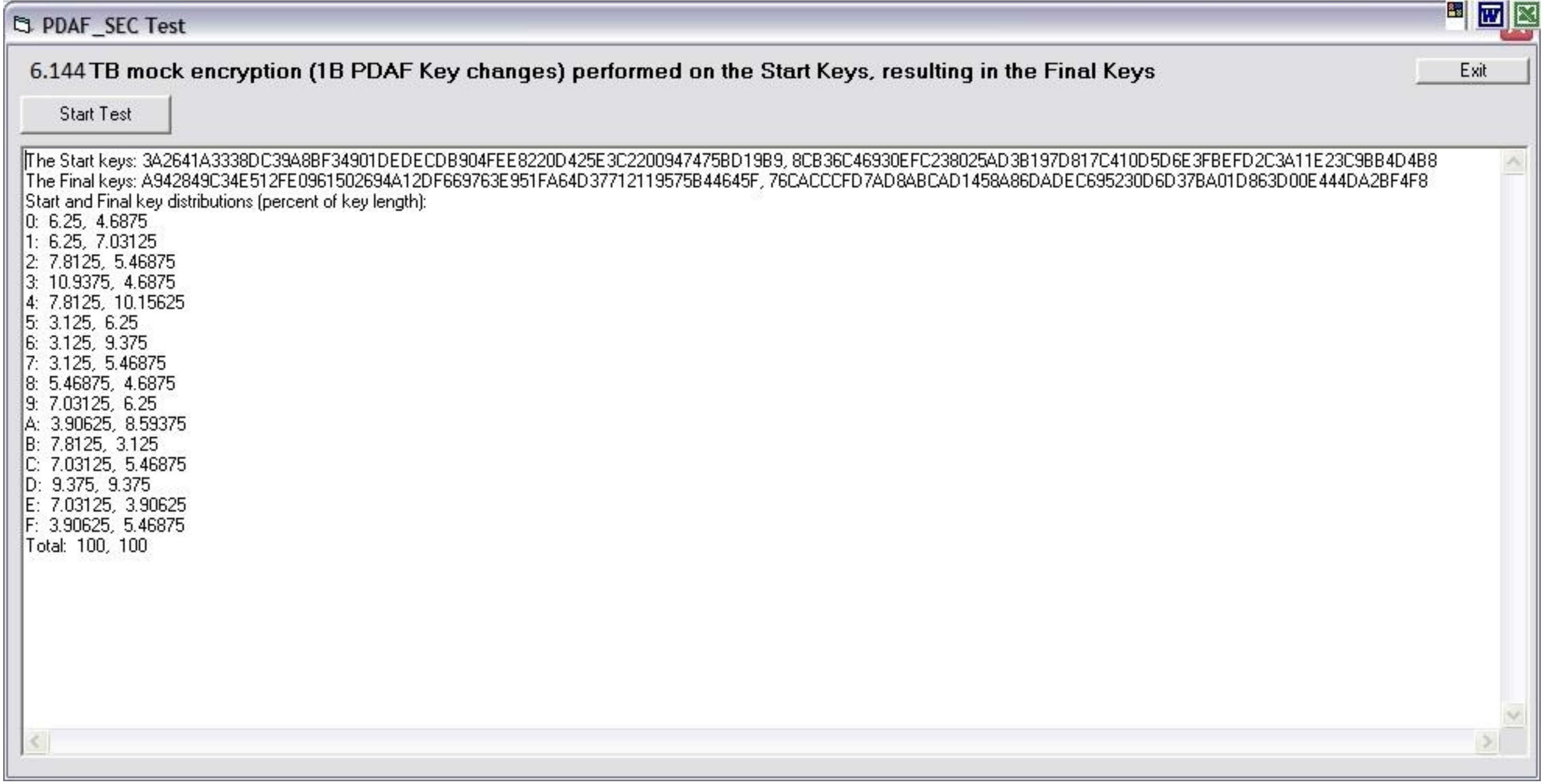
```
Here are 16 examples of different PT resulting in same CT (only 1 chosen per PT for all possible Key Sets - there are actually 256, resulting in the 4,096)

PT: 0000, OR: 0000, EK: 0001, QK: 0000, VKP: 0000, OKP: 0001, VKC: 0011, OKC: 0000, W: 0011, CT: 0011
PT: 0001, OR: 0000, EK: 1100, QK: 0001, VKP: 0001, OKP: 1100, VKC: 0101, OKC: 0111, W: 0010, CT: 0011
PT: 0010, OR: 0000, EK: 1101, QK: 0001, VKP: 0001, OKP: 1101, VKC: 0110, OKC: 0111, W: 0001, CT: 0011
PT: 0011, OR: 0000, EK: 0000, QK: 0000, VKP: 0000, OKP: 0000, VKC: 0000, OKC: 0000, W: 0000, CT: 0011
PT: 0100, OR: 0000, EK: 0010, QK: 0001, VKP: 0001, OKP: 0010, VKC: 0110, OKC: 0001, W: 0111, CT: 0011
PT: 0101, OR: 0000, EK: 0010, QK: 0000, VKP: 0000, OKP: 0010, VKC: 0110, OKC: 0000, W: 0110, CT: 0011
PT: 0110, OR: 0000, EK: 0011, QK: 0000, VKP: 0000, OKP: 0011, VKC: 0101, OKC: 0000, W: 0101, CT: 0011
PT: 0111, OR: 0000, EK: 0011, QK: 0001, VKP: 0001, OKP: 0011, VKC: 0101, OKC: 0001, W: 0100, CT: 0011
PT: 1000, OR: 0000, EK: 1000, QK: 0001, VKP: 0001, OKP: 1000, VKC: 1000, OKC: 0011, W: 1011, CT: 0011
PT: 1001, OR: 0000, EK: 0110, QK: 0000, VKP: 0000, OKP: 0110, VKC: 1010, OKC: 0000, W: 1010, CT: 0011
PT: 1010, OR: 0000, EK: 0111, QK: 0000, VKP: 0000, OKP: 0111, VKC: 1001, OKC: 0000, W: 1001, CT: 0011
PT: 1011, OR: 0000, EK: 1001, QK: 0001, VKP: 0001, OKP: 1001, VKC: 1011, OKC: 0011, W: 1000, CT: 0011
PT: 1100, OR: 0000, EK: 0101, QK: 0000, VKP: 0000, OKP: 0101, VKC: 1111, OKC: 0000, W: 1111, CT: 0011
PT: 1101, OR: 0000, EK: 0110, QK: 0001, VKP: 0001, OKP: 0110, VKC: 1011, OKC: 0101, W: 1110, CT: 0011
PT: 1110, OR: 0000, EK: 0111, QK: 0001, VKP: 0001, OKP: 0111, VKC: 1000, OKC: 0101, W: 1101, CT: 0011
PT: 1111, OR: 0000, EK: 0100, QK: 0000, VKP: 0000, OKP: 0100, VKC: 1100, OKC: 0000, W: 1100, CT: 0011
Number Of CT that equals 0011:  4096
Number Of total runs:  65536
```

This small version execution shows that for every possible PT (there are 16: 0000, 0001, 0010, etc.), calculated with every possible OR (16), using every possible QK (16) and EK (16), there are 4,096 identical PT-to-CT mappings out of the 65,536 total possibilities.

The program shows one for every same PT, where there are 256; e.g., for every different PT against all possible key sets (OR, QK, EK, which is 16^3 or 4,096), there are 256 identical PT-to-CT mappings. Since there are 16 different possible PTs, there are 16*256 identical sets (4,096) out of the total 65,536 different PT/OR/QK/EK pairings. This empirical QCy™ execution demonstrates the combinatorial probabilities at any key size. The larger the key, the smaller the percentage valid set of possible answers. For any particular ciphertext, there is straightforward probability mapping of multiple plaintexts, meeting the Perfect Secrecy in an Ideal System definition. This is demonstrably true for any QCy™ Session Start, Selection Case and Cipher.

The Update (QCy™ *Random Rearrangement* property) provides a truly new random key set for the next cycle:



6.144 TB mock encryption (1B PDAF Key changes) performed on the Start Keys, resulting in the Final Keys

```
The Start keys: 3A2641A3338DC39A8BF34901DEDECDB904FEE8220D425E3C2200947475BD19B9, 8CB36C46930EFC238025AD3B197D817C410D5D6E3FBEFD2C3A11E23C9BB4D4B8
The Final keys: A942849C34E512FE0961502694A12DF669763E951FA64D37712119575B44645F, 76CACCCFD7AD8ABCAD1458A86DADEC695230D6D37BA01D863D00E444DA2BF4F8
Start and Final key distributions (percent of key length):
0:  6.25,  4.6875
1:  6.25,  7.03125
2:  7.8125,  5.46875
3:  10.9375,  4.6875
4:  7.8125,  10.15625
5:  3.125,  6.25
6:  3.125,  9.375
7:  3.125,  5.46875
8:  5.46875,  4.6875
9:  7.03125,  6.25
A:  3.90625,  8.59375
B:  7.8125,  3.125
C:  7.03125,  5.46875
D:  9.375,  9.375
E:  7.03125,  3.90625
F:  3.90625,  5.46875
Total:  100,  100
```

The above is a screen capture of a small program written to perform 1Billion PDAF *Random Rearrangements (RR)* using the PDAF function as called for in QCy™ key Update. As shown, if the keys are random to start, this PRNG process ends with random keys; the 1B key Updates represent 6.144TB of encrypted data (using all 3 Selection Cases). Key updates in this manner provide a provably mathematic one-way gate; and have long-lasting system capability, limiting the number of times a Static new key delivery is ‘required’. All of the PDAF key changes pass statistical Random tests (NIST Statistical Test Suite tested and verified).

Appendix C – Notes

- There are a great many technical aspects of the 'search for the next P-QC' algorithm. The point of this document is that all of them are irrelevant: Cryptography doesn't 'need a better Public Key Asymmetric algorithm': Qwyit™ performs authentication in every transmission – identical to the assurance of PKIs. 'Expert' dissection of anything in this discussion only needs to revolve around a single question:

If there is a Perfect Secrecy algorithm available in real time Shannon Defined Strongly Ideal System practicality, *that is not computationally bound such that it is forever safe in any computing environment*, why isn't it the Cryptographic standard?