Quantum Key Distribution Without Sifting

Extended abstract for QCrypt 2017. Full version attached.

A. B. Price\textsuperscript{1,2,*}, J. G. Rarity\textsuperscript{1} and C. Erven\textsuperscript{1}

\textsuperscript{1}Centre for Quantum Photonics, University of Bristol, UK.  
\textsuperscript{2}Quantum Engineering Centre for Doctoral Training, University of Bristol, UK.  
\textsuperscript{*}alasdair.price@bristol.ac.uk

April 26, 2017

Abstract

We propose a novel quantum key distribution protocol that uses AES to expand an initial secret, allowing us to individually authenticate every qubit, with tags that are efficient to construct. In exchange for an increase in the amount of classical data to be transmitted, the tags can be handled such that they allow secure key generation from two photon states, and make BB84 exactly 100\% efficient. This can be implemented as part of a software patch on pre-existing devices as no hardware modification is required. The scheme is secure so long as AES cannot be broken, therefore it is ideal for real-world implementations that use encryption schemes other than the one-time pad.

Background and Motivation

Assuming it is implemented perfectly, BB84 \cite{1} is an unconditionally secure way of distributing cryptographic keys. It is of particular value for schemes that themselves offer unconditional security, such as the one time pad, but which have no intrinsic method for generating a shared secret. However, for day-to-day real-world communications, BB84-with-one-time-pad is not fast enough to be useful, so the quantum key distribution is better supplying keys to practically computationally secure applications, such as AES-256 GCM instead. In this case, BB84 still surpasses modern cryptographic alternatives, as it offers eavesdropper detection, and is secure against quantum computers. Yet a number of issues remain. Half of the qubits transmitted from Alice to Bob are discarded during sifting, rendering BB84 only 50\% efficient, and photon number splitting (PNS) attacks exploit the use of weak coherent pulses in place of a single photon source.

In this more realistic system, AES GCM is now the weakest link with regards to mathematical attacks (although it is still strong in absolute terms), so we ask whether a reduction in the theoretical security of BB84 can be leveraged to counter the issues above? In addition, can this new protocol be constructed in such a way that it is vulnerable only to mathematical attacks that would break the data encryption scheme, thus giving no additional benefit to attackers who target the key generation? PNS resistance and (asymptotic) efficiency improvements are already provided by the decoy state \cite{2} and biased basis \cite{3} protocols respectively, both by modifying the quantum hardware. Therefore, our protocol must not require any physical changes be made to the BB84 setup. SARG04 \cite{4} helps mitigate against PNS attacks without making hardware changes, so our protocol must overall be better than this.
Figure 1: Block diagram showing the steps taken by Alice and Bob in order to implement quantum key distribution without sifting.

The Protocol

Our protocol is outlined in figure 1 and can be summarised as follows:

1. As in BB84, Alice prepares her first qubit by encoding a random bit $B_i$ (0 or 1) in a random basis $b_i$ ($X$ or $Z$).

2. The qubit is delayed for a short period of time while an “authentication” tag $f_{k_H}(b_i) \oplus \text{AES}_{k_C}(v_i)$ is sent to Bob. This takes the form of a computationally secure version of the authentication tag used in BB84, although it is no longer being used to authenticate a public message. $f_{k_H}(b_i)$ is a universal hash function, keyed with $k_H$, that takes the chosen basis as its input. AES$_{k_C}(v_i)$ is the Advanced Encryption Standard cipher, that “expands” a key $k_C$ by operating on a nonce (arbitrary one-time number) $v_i$. Both can be computed in parallel and in advance, meaning that XORing the two together is the only operation that must happen in real time.

3. Bob compares the tag he receives with $f_{k_H}(X) \oplus \text{AES}_{k_C}(v_i)$. If they are the same then, when he receives Alice’s qubit, he will measure in the $X$ basis. Otherwise, he measures in the $Z$ basis. This removes the need for Alice and Bob to publicly compare their bases after the qubits have been exchanged.

4. Steps 1 to 3 are repeated for the remaining $N - 1$ qubits sent from Alice to Bob. Error correction and privacy amplification are carried out as in standard BB84.

Security and Impact

If AES is indistinguishable from a random permutation, our protocol is secure (the 128-bit tags provide confidentiality and are difficult to forge) for up to $2^{64}$ qubits, given a single $k_H$, $k_C$ pair [5]. Our protocol is designed to distribute keys for use in AES-based encryption schemes, so if AES can be broken, there is no advantage to an attacker targeting the authentication tags, or indeed any aspect of the key exchange. Therefore, we claim not to have weakened the security in comparison to conventional QKD paired with data encryption that incorporates AES. A full analysis of the security and attack vectors can be found in [5].
With regards to fulfilling the design criteria, we observe first that Alice and Bob do not publicly announce their bases, so a traditional PNS attack will not work. A specialised form of the attack [5] will have limited success when applied to three-photon pulses, but is still no better than guesswork for two-photon terms. Secondly, as Bob always knows in advance which basis to measure in, the efficiency of the scheme is 100%. Therefore, our protocol offers higher key rates and increased resilience against photon number splitting, without modifying the standard BB84 hardware. These improvements are independent of characteristics that affect the quantum channel, such as transmission distance.

As our protocol offers advantages over BB84 and SARG04 [5], it could be implemented by means of a software update on commercial systems already in the field, which are capable of running either or both of these protocols. Although the classical channel must be able to transmit 128x the number of bits transferred over the quantum channel (as a 128-bit tag arrives before every qubit), this requirement is not unrealistic. For example, ID Quantique’s Clavis\textsuperscript{2} emits laser pulses clocked at 5 MHz [6], so will require a 640 Mbit/s classical channel. The Bristol and UK quantum networks, on which the Clavis\textsuperscript{2} systems are being deployed, have SFP+ and QSFP channels with capacities of 10 Gbit/s and above, which is more than adequate for our needs. In the case of newer, pre-commercial devices running at super-GHz clock speeds [7], some minor modifications can be made to avoid having to multiplex two transceivers together. Reducing the tag length to 64 bits, as is possible with a UMAC [8], brings the classical data rate down by enough to be handled with QSFP28 or CFP4 transceivers. Given the bounds derived in [9], this remains secure for up to $2^{32}$ qubits, which is still greater than the minimum number required to overcome finite key effects [10]. Thus, with the QKD technology described in [7], it is possible to implement our protocol with an optical device the size of two transceivers (one for the quantum channel, one for the classical), which is no different to the physical size of that required for BB84.

References

[1] C. Bennett and G. Brassard, “Quantum Cryptography: Public Key Distribution and Coin Tossing,” in Proc. IEEE Int. Conf. on Computers, Systems and Signal Processing, 1984.

[2] H.-K. Lo et al., “Decoy State Quantum Key Distribution,” Physical Review Letters, 2005.

[3] H.-K. Lo et al., “Efficient Quantum Key Distribution Scheme and a Proof of its Unconditional Security,” Journal of Cryptology, 2005.

[4] V. Scarani et al., “Quantum Cryptography Protocols Robust Against Photon Number Splitting Attacks for Weak Laser Pulse Implementations,” Physical Review Letters, 2004.

[5] See full version (attached).

[6] ID Quantique SA, “Quantum Key Distribution System Clavis\textsuperscript{2} User Guide (v 3.0),” 2013.

[7] P. Sibson et al., “Chip-based Quantum Key Distribution,” Nature Communications, 2017.

[8] J. Black et al., “UMAC: Message Authentication Code using Universal Hashing,” Network Working Group, The Internet Society, 2006.

[9] D. Bernstein, “Stronger Security Bounds for Wegman-Carter-Shoup Authenticators,” Ann. Int. Conf. on the Theory and Applications of Cryptographic Techniques (EUROCRYPT), 2005.

[10] V. Scarani and R. Renner, “Quantum Cryptography with Finite Resources: Unconditional Security Bound for Discrete-Variable Protocols with One-Way Postprocessing,” Physical Review Letters, 2008.
1 Canonical BB84 [1]

1.1 Overview

Conventionally, Alice will transmit a series of quantum bits \( \bigotimes_{j=1}^{N} |\psi_j\rangle \in \{ |0\rangle, |1\rangle, |+\rangle, |-\rangle \} \) to Bob, where each qubit is set to represent one of the two classical bit values. Once she is suitably convinced Bob has measured each qubit and eavesdropper Eve does not have a copy stored in a quantum memory (which can happen through a photon number splitting attack - see attack 2 - when weak coherent pulses are used in place of a single photon source), she will broadcast a message \( m_j \in M \) telling him the bases in which she prepared the quantum states. He will respond with message \( m'_j \in M \) telling her which qubits he measured correctly, enabling appropriate sifting of their keys\(^1\). \( m_j^{(t)} \) is accompanied by an authentication tag \( t_j^{(t)} \in T \) defined by a secret key \( k_j^{(t)} \in K \). That is,

\[
t_j^{(t)} = \text{Auth}_{k_j^{(t)}} \left( m_j^{(t)} \right)
\]

This allows Bob (Alice) to verify \( m_j^{(t)} \) came from Alice (Bob) and has not been tampered with. Therefore, the security of the scheme relies on Eve being unable to impersonate either party by forging or modifying the authentication tags. If the authentication scheme can be broken, the following attack is possible:

**Attack 1** Eve intercepts the quantum bits, measures in a random basis and resends the result she observed in the basis she measured. She conceals this by modifying Alice’s bases announcement and Bob’s response, along with the authentication tags for each. Eve can now read all communications encrypted using the key she shares with Alice, and forward them with or without modification\(^2\), having re-encrypted using the key she shares with Bob.

As we alluded to above, efficient single photon sources are yet to be realized in the real world, so state-of-the-art QKD implementations rely on weak coherent pulses. This enables the following attack to be mounted:

\(^1\)Bob can just as easily announce his bases and Alice confirm which are correct, removing the risk that she may publicly declare her bases before he receives the qubits. However, if both approaches are implemented correctly, they are equally secure. We have chosen the example where Alice is the one to reveal her bases as this will aid the explanations in section 2.

\(^2\)QKD can and should be used for the generation of both data encryption and data authentication keys, although in schemes such as GCM, these are the same.
**Attack 2** Eve performs a quantum nondemolition measurement on the number of photons in each pulse. She blocks all single photon terms, and splits those containing multiple photons. She retains at least one photon in a quantum memory, and allows the remainder to carry on towards Bob. When Alice announces her preparation bases, Eve measures the stored photons, returning the same raw key as Alice (assuming zero errors). This can be sifted correctly when Bob publicly responds to Alice’s original announcement.

To mitigate the risk posed by a photon number splitting (PNS) attack, Alice must ensure the mean photon number of her coherent states is sufficiently low. This will reduce the probability of a multi-photon state being emitted, such that attack 2 can more easily be detected. Naturally, a smaller mean photon number will negatively impact the secret key rate, so more advanced protocols have been developed that are PNS-resistant (see section 6).

### 1.2 Public Channel Authentication

In 1981, Wegman and Carter presented a new class of hash functions, which allow the construction of information-theoretically secure authentication tags [2]. Using $k_j$, a function $f_j \in \mathcal{F}$ can be selected, where $f_j : \mathcal{M} \rightarrow \mathcal{T}$. It is required that an attacker in possession of $m_j$ and $t_j$ can obtain no information about $f_j$, and that knowing $f_j$ reveals no information about $f_j \pm x$, where $x$ is an arbitrary integer. However for this to be true, the same function cannot be used for multiple messages without modifying our definition of the tag to mask the information leaked through reuse.

To create a more practical system, we redefine the key such that

$$k_j = k_H \parallel k_{j,M}$$  \hspace{1cm} (2)

where $\parallel$ is used to indicate a concatenation. Now $k_H$ (the hash key) selects a function $f \in \mathcal{F}$, where $f : \mathcal{M} \rightarrow \mathcal{G}$. As $f$ no longer changes with $j$, $g_j \in \mathcal{G}$ can be one-time pad encrypted with $k_{j,M}$ (the mask key) such that $\oplus : \mathcal{G} \rightarrow \mathcal{T}$, or more specifically

$$t_j = f_{k_H}(m_j) \oplus k_{j,M}$$ \hspace{1cm} (3)

Note it is still important to hash $m_j$ to prevent a trivial key recovery attack, whereby Eve simply XORs the message out from the tag. In addition, the initial shared secret between Alice and Bob should correspond to $k_1$, as opposed to just $k_H$, to prevent an attack of the type outlined in [3].

Authentication tags for the public channel in quantum key distribution often take the form of equation 3 [4, 5, 6], although for the purposes of providing an example, we will focus on the SECOQC implementation [4, 7]. Even though we no longer need to select a new hash function for each tag, the use of a one-time pad means that authenticating every message at the time of communication will consume valuable quantum keys to no advantage. Instead, a delayed authentication scheme is used where all messages are transmitted between Alice and Bob in the clear and without tags. Copies of the messages are stored in a buffer local to the transmitter and concatenated, so that once a secret key has been generated, a 96-bit tag can be produced which authenticates every message sent up until that point. Note this means we require a 96-bit $k_{j,M}$. If the public channel authenticates successfully, the new quantum key can be used by an application, however if it fails, the key should be discarded. Hash functions map arbitrarily long bit strings to bit strings of a fixed size. At the cost of wasting significant
time if it fails\textsuperscript{3}, a delayed authentication scheme minimises the amount of key that must be used in each round of QKD, because $g_j$ will always be the same length regardless of whether $m_j$ is a single announcement or a concatenation of $N$ announcements. The hash function chosen for SECOQC utilises Horner’s rule, the structure of which allows both Alice and Bob to begin computing $g_j$ before the entire message is available, so reducing the time spent authenticating once the final quantum key has been generated.

2 Proposed Modifications to BB84

2.1 Overview

We first split the bases announcement into $N$ equally sized segments, such that each segment $s_i$ announces a single basis. Each $s_i$ can be authenticated using $\text{Auth}_{k_i}(s_i)$, a tag of the form given by equation 1. To avoid consuming more secret key than is generated by the QKD protocol, we require

**Assumption 1** There exists an authentication scheme that safely allows $k_i$ to be reused, such that $k_{i+1} = k_i$ but $t_{i+1} \neq t_i$ for $1 \leq i \leq N - 1$.

**Definition 1** We consider a QKD authentication scheme to be safe if it is at least as difficult to break as the (authenticated) data encryption scheme in which the quantum key is used.

This restricts us to using practically computationally secure encryption as opposed to the one-time pad. However, as information-theoretically secure encryption is impractical for day-to-day real-world applications, this is not particularly concerning.

It should be observed that for the purposes of this work, safety and security are separate concepts. Under definition 1, there is nothing to prevent a scheme that is unsafe from being considered secure in absolute terms.

Now it is possible to transmit $(s_i, t_i)$ in parallel with $|\psi\rangle_{i+y+1}$, where $y$ is the number of bits corresponding to the amount of time needed to be sure Eve has not stored $|\psi\rangle_i$ (or a copy of $|\psi\rangle_j$) in a quantum memory. However, depending on which basis $s_i$ is set to represent, $t_i$ can be one of only two possible bit strings, both of which will change unpredictably with $i$, in addition to their dependency on $s_i$. Therefore, if Bob is willing to devote extra processing power to calculating both possible tags for each segment, he can compare these with the $t_i$ he receives in order to sift the quantum key, meaning Alice no longer needs to transmit the plaintext basis announcement.

If the bases are no longer being publicly announced, we can discard the requirement whereby Bob measures before comparing authentication tags. Instead, $t_i$ can be transmitted immediately after $|\psi\rangle_i$ such that, by applying a short delay to the quantum channel, Bob can check in which basis he should measure before making the measurement. As a result, he measures in the correct basis 100% of the time, thereby doubling the pre-error-corrected key rate at the cost of some extra processing.

2.2 Attack Vector Analysis

2.2.1 Authentication Breaks

If the authentication scheme can be broken as in conventional BB84, attack 1 is still valid, with only the tags needing to be forged. However, the break may be of such a nature that the\textsuperscript{3} intentionality forcing a delayed authentication scheme to fail can form the basis of a reliable denial of service (DoS) attack.

\textsuperscript{3}It should be noted that intentionally forcing a delayed authentication scheme to fail can form the basis of a reliable denial of service (DoS) attack.
following must be considered.

**Attack 3** Eve intercepts the quantum bits and authentication tag. She reads off the bases in which Alice prepared the bits, measures accordingly and resends the result she observed in the basis she measured. Eve can now read all communications encrypted using the key she shares with Alice and Bob, and forward them with or without modification.

To know if our new way of announcing the bases is as secure as before, it may appear that we need to establish whether attack 3 is of equal or greater difficulty than attack 1. Specifically, if it is at least as difficult to determine a message from its authentication tag as it is to forge or modify said tag, then there is no reason for an attacker to attempt the former in a world free of implementation flaws. However, while this is an interesting consideration in itself, it is not necessary under the operating conditions for which our protocol is designed. According to definition 1, the important metric is whether attack 3 is at least as difficult as breaking the confidentiality of the data encryption scheme. If, in the construction of our authentication scheme, we use the block cipher on which our data encryption is based, it is possible to fulfill this condition. We discuss this further in sections 3 and 4.

Note that because we have not changed any underlying principles of QKD beyond the nature of the classical communication channel, our tags are still dependent on the measurement bases rather than the bit values themselves. This means we retain eavesdropper detection and a limit on the time over which breaking the authentication scheme is useful. One may observe that this time is now fundamentally limited by the performance of Eve’s quantum memory, whereas a flawed implementation of conventional QKD technically has no such bound if its designer fails to program a maximum wait time for the authentication tag to be received by Bob.

### 2.2.2 Photon Number Splitting

Attack 2 is only possible if Alice and Bob publicly compare their bases. Without this step, Eve has no way of obtaining the information she needs to correctly measure the states stored in her quantum memory. Therefore, our protocol invalidates the traditional PNS attack, as Alice and Bob’s bases remain private throughout. Any break in the authentication scheme which allows Eve to carry out attack 2 will also allow her to carry out the much simpler attack 3 or (by section 2.2.1) break the data encryption scheme directly, meaning there is no advantage to attempting attack 2. While this allows us to increase our mean photon number, yielding a higher secure key rate than in BB84 (see section 4.3), it does not mean we can do so indefinitely. A new, less powerful PNS attack can be mounted as follows:

**Attack 4** Eve performs a quantum nondemolition measurement on the number of photons in each pulse. She blocks all single and two-photon terms, but splits those containing three or more photons. She retains at least two photons in a quantum memory, and allows the remainder to carry on towards Bob. Eve then performs unambiguous state discrimination on the qubits in her possession and returns a fraction of Alice’s raw key.

For three-photon terms, Eve can unambiguously determine 50% of Alice’s key using basic linear optics [8]. Assuming that Eve randomly guesses the values of the bits for which state discrimination proves inconclusive, she can expect to learn 75% of the key overall, compared to 50% through randomly guessing and 100% in a traditional PNS attack. However, for the purposes of this work, we assume that 75% is sufficient to compromise the security of our scheme.

### 2.2.3 Randomness

The final point to check is whether Bob’s failure to inject additional random numbers has an adverse effect on the entropy of the final key. The short answer is no, and it is important to
realise that any answer to the contrary would also apply to the case where Alice and Bob both randomly generate the same set of bases with probability \( \frac{1}{N} \). If Alice is using an ideal quantum random number generator (QRNG) then the key she transmits will have maximum entropy. In conventional QKD, Bob’s random bit deletion now becomes a matter of practicality rather than doing anything to further mitigate Eve’s ability to guess the key, assuming he also uses an ideal QRNG.

However, it is important to consider what happens if an insecure or backdoored random number generator (RNG) is used at either end for basis selection. While the outcome is trivial if the same RNG is used for Alice’s bit selection, this is not enforced, hence we stick with a more general implementation where difference RNGs are used for Alice’s bits, Alice’s bases and Bob’s bases.

**Attack 5** If Eve can predict Alice’s random sequence, she will be able to intercept the qubits, measure in the correct basis and resend. Her measurements return the same raw key as Alice (assuming zero errors) which can be correctly sifted when the bases are publicly compared.

Similarly,

**Attack 6** If Eve can predict Bob’s random sequence, she will be able to intercept the qubits, measure using his set of bases and resend. Her measurements return the same raw key as Bob (assuming zero errors) which can be correctly sifted when the bases are publicly compared.

In our new QKD scheme, attack 6 reduces to attack 5 without sifting. As Bob is not generating any extra randomness himself, the predictability of his measurement bases is determined by Alice’s RNG. Therefore, Bob needs to trust Alice has made sensible implementation decisions, but as attack 5 exists in conventional QKD as well, this is nothing new. If Alice is using a maximum entropy RNG, Bob’s measurement bases should still be highly unpredictable from Eve’s perspective, as they are correlated only with the preparation bases. More precisely, Alice and Bob’s mutual information is now maximised, but Alice and Eve’s mutual information is still at a minimum, hence the mutual information between Bob and Eve is also minimised.

### 3 Addressing the Choice of Authentication Scheme so as to Satisfy Assumption 1 and Definition 1

The ideal way to satisfy assumption 1 and definition 1 is if there were an authentication scheme that could only be defeated if the cipher used in the quantum-keyed encryption scheme were broken. Brassard proposed in [9] that \( k_{j,M} \) could be defined as the output of a random function, keyed with Alice and Bob’s shared secret. It does not take much to realise this function could be the cipher used in the data encryption (see section 4.1 for the reasoning behind this), independently keyed with \( k_C \), meaning the authentication tag becomes

\[
t_j = f_{k_H}(m_j) + \text{AES}_{k_C}(v_j)
\]

where \( v_j \) is a nonce and + can be any abelian group operation, including XOR. This is still of Wegman-Carter form, except

\[
k_{j,M} = \text{AES}_{k_C}(v_j)
\]

A number of efficient authentication schemes are based on this idea, such as UMAC [10] and poly1305-AES [11]. So long as \( v_j \) never repeats for any given \( k_C \) (this could be ensured by introduction of a counter), the same key can be used for securely authenticating up to \( 2^{64} \) messages [12]. For comparison, \( \Omega(10^5) \) raw bits must be exchanged and processed for finite
key security [13]. If each basis were to be declared using an authentication tag, $O(10^{14})$ secret keys could be generated before our authentication scheme would need to be rekeyed\(^4\).

Therefore, we can redefine the overall authentication key as

$$k = k_H \| k_C$$  \hspace{1cm} (6)

noting that the index $j$ has been dropped because $k_j = k_{j+1}$ for $j < 2^{64}$.

So long as the quantum keys are used in an AES encryptor such as AES-GCM\(^5\), we can use this to authenticate the classical QKD channel, the security of which is discussed in section 4. While $k_C$ will be 160 bits longer than the $k_{j,M}$ used in SECOQC, this is radically offset by our scheme’s higher key rate (see section 4.3), because $k$ will ultimately be refreshed with a quantum key.

The final question that we will explore in this section is whether using information theoretically secure data encryption with practically computationally secure QKD authentication could present any kind of advantage. The answer to this is yes. While the short term security of the system cannot be said to be any better, the long term security may benefit. Under the arguments presented here and in section 4, breaking AES allows Eve to decrypt a user’s data as easily as she can attack the QKD authentication. However, data which needs long-term security could be encrypted with a quantum-keyed one-time pad so that it does not become vulnerable if such a break is discovered after it has been sent across a public network. The idea that authentication weaknesses can only be exploited during QKD, as opposed to after a key has been established, is also the basis for claims that QKD with public key authentication is still superior relative to contemporary key distribution protocols [14].

4 Security

4.1 Confidentiality

The authentication tags in conventional QKD are designed to provide both message integrity and authenticity. While this is still important for our protocol, as we are technically authenticating an out-of-band message, we also have an additional requirement: confidentiality. Luckily, it is easy to show this is provided by the MAC recommended herein.

Stream ciphers compute the ciphertext

$$c_\iota = p_\iota \oplus r_\iota$$  \hspace{1cm} (7)

where $r_\iota$ is a pseudorandom stream, $p_\iota$ is the $\iota$th plaintext block, and $c_\iota$ is the $\iota$th ciphertext block. Equation 4 is of this form and, more specifically, can be viewed as an implementation of AES-CTR (equation 8), where $p_\iota$ represents the output of $f_{k_H}(m_\iota)$.

$$c = p_\iota \oplus \text{AES}_{k_C}(v_\iota)$$  \hspace{1cm} (8)

The security of CTR using a pseudorandom function (PRF) to generate $r_\iota$ is presented in Theorem 13 of the full version of [15]. Block ciphers are considered strong pseudorandom permutations (PRPs) and all PRPs are PRFs [16]. A PRP should, by definition, be indistinguishable

\(^4\) Although doing so more often (for example after each round of QKD) may be preferable.

\(^5\) The data authentication scheme could of course be the same as that used for the classical QKD channel, however it is important to note that this does not have to be the case for security.
from a random permutation, therefore so long as AES fulfils its design criteria, AES-CTR is secure in that it provides an acceptable level of confidentiality. Thus, the MAC described in section 3 provides the same.

Not only does this show that the secrecy required for our basis announcements can be achieved, it also demonstrates that in order to break the confidentiality of our scheme, Eve must compromise the same construct (AES) that will break the confidentiality of the data encryption. Therefore, in this regard, there is no advantage to attacking the QKD authentication over the data encryption, in accordance with definition 1 and the discussion surrounding attack 3.

4.2 Unforgeability

Bernstein showed in [12] that authenticators of the Wegman-Carter form are secure up to $\sqrt{\#K_M}$ messages, where $k_{i,M} \in K_M$ and # used to represent the cardinality of the set. Given equation 5, our scheme is secure with regards to forgeability for up to $2^{64}$ qubits\(^6\), under the conditions that our hash function has small differential properties\(^7\) and AES cannot be distinguished from a uniform random one-to-one function. Theorem 5.4 of [12] proves this.

If a distinguishing attack can be mounted on AES, the quantum-keyed encryption scheme will also become vulnerable. Once again, Eve gains no clear advantage from attacking the QKD authentication scheme over the encrypted data.

4.3 Secure Key Rate

As outlined in section 2.2.2, our new protocol invalidates the traditional photon number splitting attack. This means we can increase our mean photon number, $\mu$, such that we reduce the probability of Alice emitting a zero-photon state (equation 9 for $n = 0$), thus improving the secret key rate relative to BB84.

$$\text{Prob} (\gamma = n) = |\langle n | \alpha \rangle|^2 = e^{-\mu \mu^n} \frac{n^n}{n!}$$ (9)

Although each pulse is now more likely to contain a single photon, the chances of it containing two or more have also gone up. Our scheme can securely generate key from two-photon terms, so these are also able to contribute to the overall key rate, however we still require that our mean photon number fulfills the condition

$$\text{Prob} (\gamma \geq 3) \rightarrow 0^+$$ (10)

For BB84, GLLP gives the secure key rate as [17]

$$R_{BB84} \geq \frac{1}{2} \left\{ -Q_\mu H_2 (E_\mu) + Q_1 \left[ 1 - H_2 (e_1) \right] \right\}$$ (11)

where the factor of $\frac{1}{2}$ manifests itself because this is the fraction of pulses for which Alice and Bob can expect their bases will match. $Q_\mu$ is the probability of Bob getting a detection in a pulse for which he and Alice use identical bases, and $E_\mu$ is the quantum bit error rate (QBER). $H_2 (\cdot)$ is the binary entropy function, while $Q_1$ and $e_1$ are the gain and error rate respectively for single photon states.

\(^6\)We use up the equivalent of one message for every qubit transmitted, because one tag is required per qubit and generating $N$ tags is the same as authenticating $N$ messages.

\(^7\)This is also the case for the authentication tags used in conventional QKD as differential cryptanalysis is applicable to hash functions in general.
Table 1: Showing the probability of a bit flip error occurring between Alice and Bob depending both on whether or not Eve blindly modifies the authentication tag, and the bases chosen by each of the three parties.

| Alice’s Basis | Eve’s Basis | Forwarding Choice | Bob’s Basis | Prob(error) |
|---------------|-------------|--------------------|-------------|-------------|
| X             | X           | \( t_e = t_i \)    | X           | 0           |
| X             | X           | \( t_e \neq t_i \) | Z           | 0.5         |
| X             | Z           | \( t_e = t_i \)    | X           | 0.5         |
| X             | Z           | \( t_e \neq t_i \) | Z           | 0.5         |
| Z             | X           | \( t_e = t_i \)    | Z           | 0.5         |
| Z             | X           | \( t_e \neq t_i \) | Z           | 0.5         |
| Z             | Z           | \( t_e = t_i \)    | Z           | 0           |
| Z             | Z           | \( t_e \neq t_i \) | Z           | 0           |

For QKD without sifting, Alice and Bob use the same bases 100% of the time. We must also introduce a term to represent two-photon state contributions to the key rate \( R_2 \), the exact form of which is work in progress. Therefore,

\[
R_{QWS} \geq -Q_\mu H_2 (E_\mu) + Q_1 [1 - H_2 (e_1)] + R_2
\]  

(12)

As with SARG04 [18], extending our scheme to the six-state protocol is expected to enable secure key generation using pulses containing three or four photons.

5 Optimising the Classical Processing

While the protocol outlined herein offers a number of benefits, it is at the cost of increasing our processing requirements. We now look at where these can be reduced, so as to increase the feasibility of implementation.

First, \( g_i \) can be computed in advance for \( \forall m_i \), most likely while the QKD system is calibrating itself. This amounts to two unique bit strings and, assuming each basis can be described by \( N_b \) bits, takes the amount of data that needs to be hashed by Alice and Bob down from \( N_b N \) to \( 2N_b \) bits and \( 2N_b N \) to \( 2N_b \) bits respectively. Conventionally, they would each have been required to hash \( N_b N \) bits, although as indicated by section 1.2, a total of 160 fewer bits were present at the output.

The basis now determines only which of the two hashes gets XOR’d with each \( k_{i,M} \), allowing parallel computations to avoid the accumulation of any delays in the transmission of \( t_i \).

Although Bob can compute the two possible tags he expects before each qubit arrives, he will still end up having to calculate \( 2N \) tags. Let us say these are defined by the X and Z bases as in BB84, and he halves his workload by only evaluating the tags corresponding to a measurement in X. We designate \( t'_i \) to be the tag he receives and \( t^X_i \) to be the tag he expects for the case where \( |\psi\rangle_i \) should be measured in the X basis. If \( t'_i = t^X_i \) then Bob will measure in X, else he will measure in Z. Therefore, if Eve measures in Z, she need not be able to forge \( t^Z_i \). If she can correctly guess whether the tag corresponds to X or Z, she need only ensure \( t_e \neq t^X_i \), where \( t_e \) is the tag she forwards. It should be clear that this offers her no advantage. If she can correctly guess the basis, she would not need to modify the tag in the first place as she would measure correctly. However, for completeness, we demonstrate that without breaking the authentication scheme, Eve’s optimal strategy is still not to modify the authentication tags.
Table 1 shows all possible outcomes. As we are no longer performing sifting, instances where Alice and Bob’s bases differ can only be exposed during the error correction process, when they compare a string of check bits to estimate the QBER. As our concern is only whether Bob’s failure to check if \( t_e = t_i \) constitutes a weakness, it is important to remember that the authentication has not been broken, so the check bit comparison is still between Alice and Bob, not Alice and Eve/Eve and Bob.

We can expect a QBER of 31.25% if Eve randomly chooses both her basis and her forwarding choice. This can be contrasted with a conventional intercept/resend attack, where \( \text{Prob (error)} = 0 \) in half of all possible scenarios, meaning we expect the QBER to be 25% if Eve randomly chooses her basis. Therefore, the question presents itself as to whether a better strategy exists in this new scenario, with QBER \( \leq 25\% \). We see that if Eve measures and resends in X, the probability of Bob measuring a different bit to that which Alice sent is greater if \( t_e \neq t_i \) than if \( t_e = t_i \). If Eve measures and resends in Z, the probability of Bob measuring a different bit to that which Alice sent is independent of whether or not she modifies \( t_i \). Therefore, \( t_e \equiv t_i \) is an optimal strategy, with the same QBER as in conventional QKD.

Finally, because Bob will interpret any tag which does not match \( t_i^X \) as an instruction to measure in the Z basis, Alice could reduce her processing requirements by sending random data in place of \( t_i^Z \). This modification is only of use if Alice has an efficient (Q)RNG, or excess memory that can be sufficiently populated when an inefficient (Q)RNG would otherwise be idle. If it were to be implemented, care must be taken to ensure that the opportunity is not created for Eve to mount a simple timing attack.

6 A Comparison with PNS-resistant and Highly Efficient QKD Protocols

6.1 QKD with Biased Bases

It was first suggested in [19] that, by biasing the basis choices in QKD, one can increase the average efficiency of BB84, which is conventionally set at 50%. More precisely, if a qubit is sent in basis 1 with probability \( \text{Prob} (b_1) \) and basis 2 with probability \( 1 - \text{Prob} (b_1) \), where basis 2 is used for key generation and \( \text{Prob} (b_1) \) is constrained by the number of bits required to detect any eavesdropping, then the efficiency of BB84 asymptotically doubles.

The optimum value for \( \text{Prob} (b_1) \) is given by [19]

\[
\text{Prob} (b_1) = O \left( \sqrt{\frac{\log N_q}{N}} \right)
\]

meaning \( \text{Prob} (b_1) \to 0^+ \) as \( N \to \infty \). \( N_q \) is the final length of the quantum key. However, unlike our scheme, biased basis QKD can still never actually reach 100% efficiency, theoretically or in practice [20].

The advantage of biasing one’s bases is that the information-theoretic security of BB84 is retained for single photon implementations [19], so this may be preferable to our approach in certain bespoke situations. In contrast to QKD without sifting, biased basis BB84 is weaker than its vanilla counterpart against a PNS attack. An eavesdropper who does not possess a quantum memory, but is still capable of splitting multiphoton states, has to measure her stolen qubits before Alice and Bob compare their bases. So long as she makes all of her measurements in basis 2, she will obtain Alice’s raw key without introducing any errors.
Biased basis QKD is feasible to implement with the same level of technology as required for BB84, although minor modifications to the optical setup are required.

6.2 SARG04

In [21], a new protocol (SARG04) was proposed as a way of making QKD more robust against PNS attacks. It uses the same physical hardware as BB84, but changes the information encoding, meaning the bases never have to be directly announced. Unfortunately, a side effect of this is that, for zero errors, Alice and Bob retain only $\frac{1}{4}$ of their raw key bits after sifting. Therefore, the efficiency is only 25% of that of our protocol. Furthermore, it is thought that the retention rate of $\frac{1}{4}$ is responsible for SARG04 having a lower secret key rate than BB84 at short distances, although we also note that in this regime SARG04 is not proven secure [22]. At greater distances, it benefits from being able to securely extract key from two-photon pulses, however QKD without sifting also has this advantage.

Given our scheme is no different to BB84 at the quantum information level, we expect the same QBER for a channel of given visibility. However, the QBER of SARG04 is twice as large in single photon implementations [22], limiting its applicability.

One final (non-scientific) point is that SARG04 has been patented [23], reducing how widely and/or easily it can be adopted for real-world applications within the foreseeable future. The authors are unaware of any intellectual property claims that would affect QKD without sifting in this way.

6.3 Decoy State QKD

A comparison of our protocol with decoy state QKD [24] is still work in progress. However, we expect decoy state QKD to perform better with respect to PNS attacks, as it tightly bounds $Q_1$ and $e_1$ rather than applying worst-case assumptions even where they may not be appropriate. Additional optical components are required to generate the decoy states, and the efficiency of the underlying protocol remains unchanged unless combined with the biased basis approach (section 6.1). Even then, for BB84, simulations show the key rate improves only by 80% when the transmission loss is 0 dB, tending towards 45% as the transmission loss becomes greater [25]. If they do indeed offer significantly better protection against PNS attacks than QKD without sifting, decoy states could be added to our scheme in a similar way to take advantage of its superior efficiency.

7 Summary and Further Work

Quantum key distribution (QKD) was originally designed for information theoretically secure communications and, if this really is essential, it remains necessary for Alice and Bob to publicly declare their bases after the qubits have been transmitted. However, in the real world, unconditional security is rarely required, and practically computationally secure ciphers such as AES are widely used, since they consume fewer resources. Furthermore, attacks that real QKD implementations can be vulnerable to, such as photon number splitting, are far more easily exploited than a yet-to-be-discovered flaw in AES.

Here, we have presented a new protocol for QKD that authenticates a non-existent basis announcement with an AES-based variant of the tags used in BB84. By sending these tags in advance of each qubit, so Bob can always work out which basis he should measure in, it is possible to increase the efficiency of QKD to 100%. In addition, this invalidates the traditional photon number splitting attack, although unambiguous state discrimination will always be possible for any protocol using weak coherent pulses, if the mean photon number gets too high. Our
Table 2: Comparing QKD without sifting (QWS), SARG04 and biased basis BB84.

|                              | QWS       | SARG04    | Biased Basis BB84 |
|------------------------------|-----------|-----------|-------------------|
| Mathematical Security        | PC (short term) | -- (low loss) | IT (higher loss)  |
| Mathematical Security        | IT (long term)† | IT         | IT               |
| with One Time Pad            |           |           |                   |
| PNS Resistance               | Yes       | Yes       | No                |
| Efficiency                   | 100%      | 25%       | $\lim_{N \to \infty} = 100\%$ |
| Requires Hardware Changes    | No        | No        | Yes               |

† Here, long term security works under the assumption that the scheme was not broken at the time of key exchange.

Key: IT = Information Theoretic; PC = Practical Computational; -- = Unproven.

The protocol is secure so long as AES is secure, meaning that in a cryptographic system that uses AES-based encryption (that is, in a real-world quantum-secure network), there is no advantage to an attacker targeting the key generation over the encrypted data.

The authentication tags are efficient to construct, so although a dedicated FPGA may be desirable in some situations, no hardware modification is required. This makes it possible to upgrade QKD systems already installed on networks, simply by means of a software patch.

Table 2 compares our protocol with two alternatives: SARG04 and BB84 with biased bases. We argue that, from a quantum perspective and when using AES for data encryption, our protocol outperforms both of these at the cost of increasing the amount of traffic on the classical communications channel. Relatively speaking, when using the one time pad, our protocol also suffers from a slight reduction in the mathematical security, except for where SARG04 is not proven secure. Further work will include an investigation of the space occupied by our protocol with respect to decoy state QKD. We will also consider whether our ideas can be applied to schemes that do not use the standard BB84 encodings.

8 Acknowledgements

The authors would like to thank Kenneth G. Paterson and David L. D. Lowndes for useful conversations.
References

[1] C. H. Bennett and G. Brassard, “Quantum Cryptography: Public Key Distribution and Coin Tossing,” in Proceedings of IEEE International Conference on Computers, Systems and Signal Processing, 1984.

[2] M. N. Wegman and J. L. Carter, “New Hash Functions and Their Use in Authentication and Set Equality,” Journal of Computer and System Sciences, 1981.

[3] A. Abidin, “On Security of Universal Hash Function Based Multiple Authentication,” in Information and Communications Security (ICICS) (T. W. Chim and T. H. Yuen, eds.), Springer, 2012.

[4] J. Bouda et al., “Encryption and Authentication in SECOQC.” http://www.fi.muni.cz/~xbouda1/, 2007 [Last Accessed: 07/11/16].

[5] M. Sasaki et al., “Field Test of Quantum Key Distribution in the Tokyo QKD Network,” Optics Express, 2011.

[6] D. Stucki et al., “Long-term Performance of the SwissQuantum Quantum Key Distribution Network in a Field Environment,” New Journal of Physics, 2011.

[7] O. Maurhart, “QKD Networks Based on Q3P,” in Applied Quantum Cryptography (C. Kollmitzer and M. Pivk, eds.), Springer, 2010.

[8] S. J. van Enk, “Unambiguous State Discrimination of Coherent States with Linear Optics: Application to Quantum Cryptography,” Physical Review A, 2002.

[9] G. Brassard, “On Computationally Secure Authentication Tags Requiring Short Secret Shared Keys,” in Advances in Cryptology: Proceedings of Crypto 82 (D. Chaum, R. L. Rivest, and A. T. Sherman, eds.), Springer, 1983.

[10] J. Black et al., “UMAC: Message Authentication Code Using Universal Hashing,” Network Working Group, The Internet Society, 2006.

[11] D. J. Bernstein, “The Poly1305-AES Message-Authentication Code,” International Workshop on Fast Software Encryption, 2005.

[12] D. J. Bernstein, “Stronger Security Bounds for Wegman-Carter-Shoup Authenticators,” Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT), 2005.

[13] V. Scarani and R. Renner, “Quantum Cryptography with Finite Resources: Unconditional Security Bound for Discrete-Variable Protocols with One-Way Postprocessing,” Physical Review Letters, 2008.

[14] D. Stebila et al., “The Case for Quantum Key Distribution,” arXiv:0902.2839v2, 2009.

[15] M. Bellare et al., “A Concrete Security Treatment of Symmetric Encryption,” in Proceedings of the 38th Symposium on Foundations of Computer Science, IEEE, 1997. Full version available from http://web.cs.ucdavis.edu/~rogaway/papers/sym-enc.pdf [Last Accessed: 07/04/17].

[16] J. Katz and Y. Lindell, “Introduction to Modern Cryptography,” Chapman & Hall/CRC, 2008.

[17] X. Ma, “Unconditional security at a low cost,” Physical Review A, 2006.

[18] K. Tamaki and H.-K. Lo, “Unconditional Secure Key Distillation From Multi-photons,” Physical Review A, 2006.
[19] H.-K. Lo et al., “Efficient Quantum Key Distribution Scheme and a Proof of its Unconditional Security,” *Journal of Cryptology*, 2005.

[20] C. Erven et al., “Entangled Quantum Key Distribution with a Biased Basis Choice,” *New Journal of Physics*, 2009.

[21] V. Scarani et al., “Quantum cryptography protocols robust against photon number splitting attacks for weak laser pulse implementations,” *Physical Review Letters*, 2004.

[22] C. Branciard et al., “Security of Two Quantum Cryptography Protocols Using the Same Four Qubit States,” *Physical Review A*, 2005.

[23] N. Gisin et al., “Quantum Cryptography Protocol,” *Worldwide Patent WO/2004/047359*, 2004.

[24] H.-K. Lo et al., “Decoy State Quantum Key Distribution,” *Physical Review Letters*, 2005.

[25] Z. Wei et al., “Decoy-state Quantum Key Distribution with Biased Basis Choice,” *Scientific Reports*, 2013.