Code-based physical layer secret key generation in passive optical networks

Marco Baldi, Franco Chiaraluce, Lorenzo Incipini

Dipartimento di Ingegneria dell'Informazione
Università Politecnica delle Marche
Ancona, Italy

Marco Ruffini

CONNECT telecommunications research centre
The University of Dublin, Trinity College
Dublin, Ireland

Abstract

To guarantee secure transmissions is an important target of passive optical networks (PONs). Modern standards for PONs, however, impose the adoption of symmetric encryption algorithms in downstream but do not do the same in upstream, where the secret keys may be transmitted in clear. Because of non-ideal optical network components, this exposes the PON to the risk of eavesdropping. In this paper, a novel technique for securely generating and sharing secret keys in passive optical networks is proposed. It exploits randomness at the physical layer and key distillation based on coding techniques. The main attack strategies are considered and the design parameters of the proposed protocol are discussed, both in analytical terms and through numerical examples. The cost in terms of complexity with respect to standard approaches affected by possible vulnerabilities is also assessed.

Keywords: Error correcting codes, passive optical networks, physical layer security, secret key generation, XG-PON.

2010 MSC: 94A05, 94A60, 94B05

Email addresses: m.baldi@univpm.it, f.chiaraluce@univpm.it, l.incipini@pm.univpm.it (Marco Baldi, Franco Chiaraluce, Lorenzo Incipini), marco.ruffini@scss.tcd.ie (Marco Ruffini)

1. Introduction

Passive optical networks (PONs) are a desirable choice for delivering high-speed and reliable data communications at minimum cost and complexity. The so-called 10-Gigabit Passive Optical Network (XG-PON) [1], in particular, is regarded as one of the key technologies for future Internet access networks. Whilst, on one hand, PONs offer a number of advantages, also in terms of reduced costs for deployment and maintenance, on the other hand they are inherently exposed to security threats which can mine their widespread applicability [2], [3]. Among the several security risks a PON may suffer, in this paper we focus attention on eavesdropping.

In Fig. 1 the basic elements of a typical PON are sketched. They are: an optical line terminal (OLT) at the service provider's central office (hub) and a number of optical network units (ONUs), located at the user's premises. Eavesdropping occurs when an ONU is able to intercept data sent to or by another ONU. In downstream, i.e., from the OLT to the ONUs, this is naturally possible as the signals coming from the OLT are broadcast to multiple ONUs through the beam splitter (BS) they are connected to.

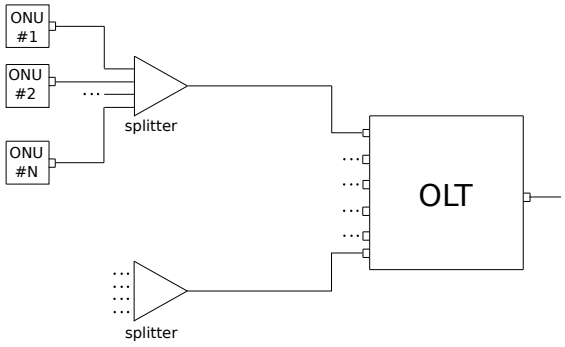

Figure 1: Typical setup of a PON.

1.1. Motivation

To prevent downstream eavesdropping, the standard requires the adoption of encryption by using symmetric-key cryptography based on the advanced encryption standard (AES), with keys of at least 128 bits. Instead, in upstream, though suggested, encryption is not mandatory. This is motivated by the assumption that upstream traffic of an ONU cannot be observed by other ONUs due to the high directionality of splitters. However, often this assumption does not hold in practice. Even though a directivity not less than 55 dB is achieved by most commercial splitters, as also addressed by the standard [4], an unexpectedly high signal leakage towards other ONUs may appear. This is due to a number of reasons, including dirtiness or degradation of connectors [5] and improper coupler termination [6]. We have verified this phenomenon through laboratory experiments, observing that the amount of reflected power received by a malicious ONU may be much greater than expected, although it strongly depends on the type of termination used for the coupler in the upstream direction. Since, according to the standard, data for generating the AES secret key are sent by each ONU to the OLT during the transmission initialization phase, it follows that, if the upstream transmission is in clear, a malicious user can employ a sensitive receiver to eavesdrop the key from the reflected signal and then decrypt the downstream transmission of the victim ONU.

The above considerations suggest the need for more robust key distribution techniques for PONs, and several solutions have been proposed for such a purpose [7]. A brief overview is reported in Section 2; all these schemes exhibit some limitation, mainly as regards their practical implementation within the current standard XG-PON framework.

1.2. Our contribution

The aim of this paper is to propose a novel, simple approach for key generation and distribution in PONs based on physical layer security, and exploiting error correcting codes (ECCs). To the best of our knowledge, the idea of using physical layer secret key distillation techniques in commercial PONs has never

appeared in the literature. The basic idea is to create the conditions for the appearance of errors in the transmission from the OLT to the ONUs, for a very short time interval during which relevant information on the secret key is sent. Because of their random nature, errors appear in different positions at the various ONUs. Using well known concepts of ECC theory, the OLT can be put in the condition to know exactly the error pattern affecting each ONU, and to use it to distill the secret key for the AES-encrypted downstream transmission. On the other hand, being specific of any communication pair, the error pattern received by an ONU cannot be recovered by another ONU and this provides the desired level of security. An important merit of this approach is that it can be easily implemented in current devices. In fact, the only change it requires with respect to the standard framework is the inclusion of simple and already consolidated decoding algorithms at ONU level, which operate according to the procedure described in Section 4. The employment of ECCs in PONs is not critical, as they are already used for improving the reliability of transmissions [8]. Nevertheless, this comes at some cost in terms of complexity, which is estimated in Section 5.2. Such a complexity increase, however, is not expected to affect the overall performance of the PON, and is acceptable for overcoming an important vulnerability of the standard key derivation procedure.

1.3. Paper organization

The organization of the paper is as follows. In Section 2 we present some related work. In Section 3, we remind the approach currently used for establishing the keys in PONs and discuss non-ideality of the components and the channel. In Section 4, we introduce the proposed key generation protocol and study some possible attacks against it, along with the relevant countermeasures. In Section 5, we design the system parameters based on an analytical model, and we assess the performance and complexity of the proposed approach. Finally, in Section 6, we draw some conclusions.

2. Related work

As in other types of networks, confidentiality in a PON can rely on either asymmetric or symmetric cryptosystems. Both these solutions require that the relevant encryption keys are distributed over the network. For this purpose, two alternative approaches are viable in PONs, based on

1. classical key distribution protocols for symmetric or asymmetric ciphers, or
2. quantum key distribution techniques for symmetric ciphers.

Actually, asymmetric ciphers are not well suited for the use in PONs. This is because key distribution for asymmetric ciphers requires setting up a public key infrastructure (PKI) [9], which is not a practical solution in PONs. In fact, a PKI would allow distributing public keys, e.g., through X.509 certificates, to be used with asymmetric ciphers. However, any ONU must be provided with a valid public key of the certification authority in order to verify authenticity of the certificate and, moreover, Internet connectivity is required in order to check the validity of any public key certificate at the time of its usage. In fact, it has to be verified that the certificate has not been revoked. However, Internet connectivity is normally unavailable at the time of a PON setup, and this makes the approach exploiting a PKI unfeasible in practice, as also witnessed by the lack of literature on this topic.

The use of symmetric encryption is more consolidated in PONs, and even recommended by the XG-PON standard [1]. In order to distribute fresh keys to be used with symmetric ciphers, the standard recommends that a fresh secret key is generated for each connection, based on proper data sent by each ONU to the OLT during the initial transmission phase. Details are provided in Section 3.1. This protocol, however, is subject to the vulnerability following from non-ideal splitters, as observed in [6] and further discussed afterwards. This poses an important threat on approaches relying on uplink transmissions isolation, and motivates the search for new key distribution techniques that may overcome such a vulnerability.

Opposed to the aforementioned approaches relying on classical key distribution protocols, quantum key distribution (QKD) techniques leverage the quantum description of signals to achieve unconditional security [10]. Quantum cryptography is based on certain quantum physics properties which guarantee that it is not possible to observe the state of a photon without changing it (Heisenberg's uncertainty principle). This has inspired well known QKD protocols like BB84 [11] and its simplified version B92 [12]. However, the deployment of QKD brings many practical issues, requiring sophisticated devices which increase the complexity of the system design and implementation. In short, it is recognized that QKD is very expensive, and suitable only for point-to-point connections and special applications. In fact, as shown in [13], several challenges have to be overcome before QKD can be used for securing everyday interactions, including cost. There are technological advances in the direction of addressing those challenges, but they are still at an early stage. In commercial PONs, instead, low-cost devices have to be used, which are technologically unsuitable to implement QKD schemes.

To overcome the aforementioned limitations, we propose an approach stemming from the area of physical layer security (PLS) [14]. PLS relies on the differences between the channels experienced by authorized and unauthorized users, without the need of any pre-shared secret key. Practical PLS schemes can be designed for wireless transmissions [15], able to provide a substrate helping to reduce the complexity of cryptographic techniques working at higher layers. The feasibility of physical layer key distribution over wireless channels with fading has already been verified [16, 17], as reciprocity of these channels provides the basis for key establishment.

The application of PLS to optical systems is less obvious, as the physical layer underlying optical communications is dramatically different than in wireless communications. However, significant examples already exist. Some of them are based on the phenomenon of identical chaos synchronization and exploit delay-coupled semiconductor lasers [18]. Some others use correlated random bit sequences implemented by adopting common random-signal induced synchro-

nization of cascaded semiconductor lasers [19]. Another interesting option relies on fiber index fluctuations induced by environmental instabilities [20]. Based on similar concepts, in [21], a large-scale Mach-Zehnder interferometer is used for measuring phase fluctuations in the fiber links between the communicating parties, exploiting them as a shared source of randomness to generate identical secret keys.

Although the feasibility of these solutions has been proved through a number of laboratory tests, their implementation in practical PONs remains difficult and often requires significant changes in current devices and architectures. Our aim is instead to propose a physical layer key generation mechanism that exploits the randomness of the distribution of channel-induced errors, with the advantage of working on a bit-level basis and thus not requiring significant changes in the hardware and software of commercial PON devices.

3. Current key exchange procedure and rationale of the proposal

In this section, first we shortly remind the mechanism which is currently used for managing the secret keys of the AES-encrypted downstream communication. Description is limited to the essential aspects; more details can be found in the standard [1] and in previous literature (e.g., in [22]). Then we discuss the non-ideality of the BSs which is the reason for the appearance of security threats. Finally, we remind some elements of the bit error rate (BER) analysis, since the presence of errors is at the basis of our PLS proposal.

3.1. Key exchange

In order to encrypt transmitted data, XG-PONs use several keys as schematically shown in Fig. 2. All they are obtained through an AES cipher-based message authentication code (AES-CMAC). AES-CMAC relies on the AES encryption algorithm as its main building block. Differently from plain AES, which encrypts a fixed-length input block into a fixed-length ciphertext using a secret key, AES-CMAC works on variable length input messages, and returns a fixed-length bit string as the result of encryption, depending on the input message

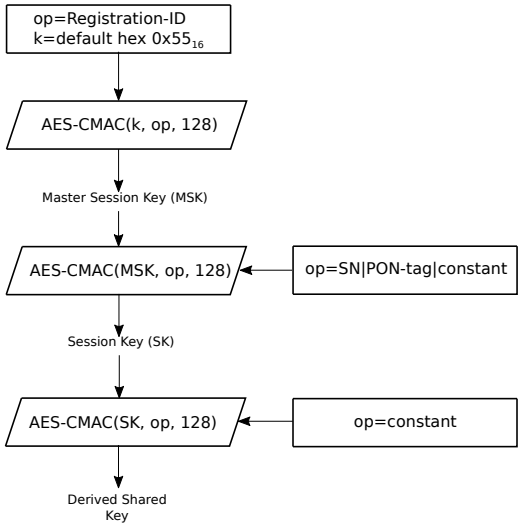

Figure 2: Keys derivation for XG-PON; the value of the constants is specified in the standard; symbol | denotes byte concatenation.

and a secret key. This is basically obtained through an iterated application of AES on concatenated blocks of the input message. As defined in the XG-PON standard [1], AES-CMAC has the following syntax:

$$\text{Result} = \text{AES} - \text{CMAC}(\text{key}, \text{operand}, 128) \tag{1}$$

which means that the 128-bit result is obtained by applying the AES-CMAC algorithm to a key and an operand [23].

According to the standard [1], the shared key used for encryption is obtained starting from the knowledge of three operands, that are:

- Registration-ID,
- ONU Serial Number (SN),
- PON-TAG.

More precisely, for the derivation of the master session key (MSK), the AES-CMAC algorithm is fed with an operand that coincides with the 36-byte string known as Registration-ID. Then, in the second stage shown in Fig. 2, the

session key (SK) is obtained by applying AES-CMAC to a 24-byte operand formed by the concatenation of three elements: the ONU SN, the PON-TAG, and the hexadecimal representation of the ASCII string "SessionK". Finally, in the last stage shown in Fig. 2, the encryption key is obtained by performing AES-CMAC on a 128-bit input string, which is the hexadecimal representation of the ASCII string "KeyEncryptionKey". Whilst the PON-TAG is transmitted downstream in clear, and is known to all nodes of the network, the Registration-ID and the SN are specific of each ONU and are transmitted in upstream. If these quantities, relative to an ONU, are disclosed to the eavesdropper, he/she is able to decipher all the downstream traffic to this ONU.

Based on the above description, it results that the whole key derivation procedure, as shown in Fig. 2, requires the computation of AES-CMAC on a series of operands with overall length of 76 bytes. This number will be used in the complexity assessment reported in Section 5.2.

3.2. Weaknesses of the current solution

The schematic representation of a 1×4 BS is shown in Fig. 3. In particular, the figure refers to the case of port A fed by an upstream power P_1. Ideally, this power should be routed entirely to port E, so that $P_0 = P_1$ and $P_2 = P_3 = P_4 = 0$. In reality, this does not happen and a fraction of power leaks towards ports B, C and D. This is firstly due to the fact that the BS has a finite directivity. According to [4], however, commercial devices are expected to exhibit a directivity not less than 55 dB. This implies that by assuming $P_1 = 10$ dBm, the corresponding power at ports B, C and D should not be larger than -45 dBm. Indeed, measuring the actual power at those outputs, it is very likely to find larger values.

A numerical example is shown in Table 1, which provides the values of the output power at the various ports of a 2×4 BS we have characterized in the laboratory. More precisely a power of 0.35 dBm, obtained attenuating by 9.65 dB a power of 10 dBm emitted by a tunable laser source, has been applied to the input indicated as the "source port", and the levels at the other ports

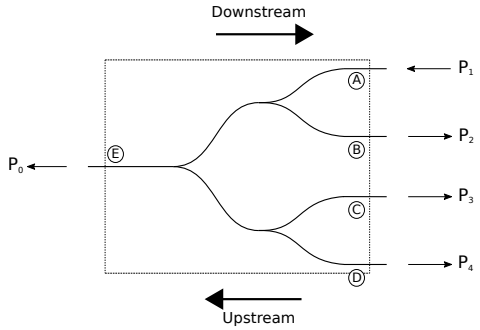

Figure 3: 1×4 BS with upstream traffic.

Table 1: Example of output measured power in a 2×4 BS. The input power is 0.35 dBm.

	P_{out} (dBm)					
Source Port	O1	O2	O3	O4	I1	I2
O1	/	−23.5	−25.1	−25.3	−7.0	−6.2
O2	−22.9	/	−23.1	−23.0	−6.9	−6.1
O3	−22.9	−30.3	/	−22.9	−6.2	−6.9
O4	−24.0	−23.4	−24.3	/	−6.4	−7.1
I1	−7.2	−7.9	−6.5	−6.5	/	−23.8
I2	−6.5	−6.7	−7.4	−7.3	−24.1	/

have been measured. In interpreting the values in the table, as regards the power balance, also the intrinsic losses of the components must be taken into account. It must be noticed that the results depend on the way we terminate the upstream connector of the passive coupler. In our experiment, we have considered the worst-case scenario with a strong mismatch at the termination, for example due to imperfect fibers coupling. In these conditions (but also for better matched scenarios) relatively high power levels, significantly above the receiver sensitivity, might be received by the eavesdropper. In such a scenario, and in absence of proper countermeasures, the malicious user can be in the condition to reveal the (theoretically) secret keys transmitted in clear from each ONU to the OLT.

Figure 4: Example of packet affected by errors; the payload is highlighted and the erred bits are circled.

3.3. Noise model

Let us suppose that receivers in the PON use an avalanche photodiode characterized by a multiplication factor M (the particular case of a PIN photodiode can be obtained by setting $M = 1$). The useful signal at the receiver is affected by thermal noise and shot noise, with root mean square values denoted by σ_t and σ_s, respectively. A simplified expression for the signal-to-noise ratio (SNR) at the receiver [24] is as follows

$$Q = \frac{2MR_dP_{rec}}{(\sigma_s^2 + \sigma_t^2)^{1/2} + \sigma_t} \tag{2}$$

where R_d is the responsivity of the photodiode and P_{rec} is the received signal power. Starting from Q in (2), the BER can be computed as

$$\mathrm{BER} = p = \frac{1}{2}\mathrm{erfc}\left(\frac{Q}{\sqrt{2}}\right). \tag{3}$$

The validity of (2) and (3) has been proved through a comparison with experimental data (see [25], for example). If the values of σ_t and σ_s are known, these equations can be used to determine the value of P_{rec} which is required to reach a given BER. We have experimentally verified the possibility to transmit at sufficiently low power levels (after the insertion of suitable attenuators) to yield high BERs. In Fig. 4, as an example, we show the screenshot of a received 128-bytes packet, catched by using Wireshark [26], where the 90-bytes payload is affected by 3 errors.

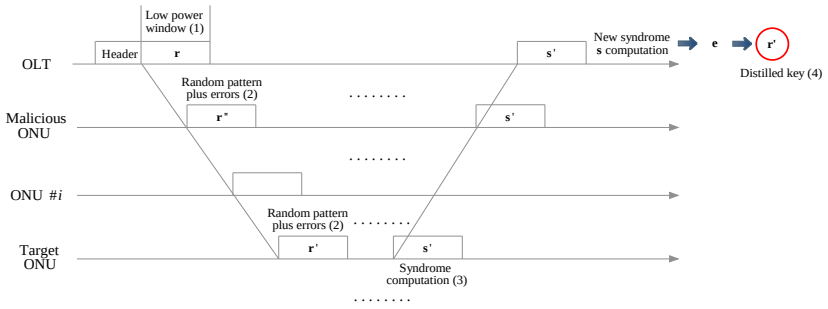

Figure 5: Messages exchange for the secret key distillation between the OLT and a target ONU.

4. Proposed method

In the protocol we propose, the randomness that the OLT and an ONU share in order to generate identical secret keys consists of the errors appearing when transmission occurs at very low power levels. Each pair OLT-ONU must be able to share an error pattern affecting their transmission, which is expected to be different from those occurred over any other channel and affecting any other transmission. As regards the eavesdropper, coherent with classical PLS formulations, his/her channel is assumed to have a quality not better than that of the legitimate channel. Indeed, solutions could be conceived to face also the case in which the quality of the eavesdropper's channel is better than that of the legitimate channel, but this is left for future works.

When key distillation starts, the OLT initiates a low power transmission window to send the next payload at a sufficiently low power to permit the appearance of transmission errors. This step is shown as Action (1) in Fig. 5. The transmission power during this phase must be properly set, also taking into account the distance of the ONUs, according to the desired value of the BER. It is important to stress that the low power window must begin after transmission of the packet's header. This is needed to avoid possible loss of synchronization, due to the presence of errors, that would prevent correct data recovery at the receiver.

During the low power transmission window, the OLT transmits a binary

random pattern $\mathbf{r}$, of length n, which is received by all the ONUs, but affected by different error patterns. Let us denote by $\mathbf{e}$ the binary error pattern at the target ONU, that therefore receives $\mathbf{r}' = \mathbf{r} \oplus \mathbf{e}$, where $\oplus$ denotes XOR. This step is shown as Action (2) in Fig. 5.

Let us denote by t the Hamming weight of the vector $\mathbf{e}$. We suppose that all network terminals are equipped with a binary ECC having rate $R = k/n$, which is able to correct a number of bit errors $t_2 \geq t$. Let us denote by $\mathbf{H}$ the $(n-k) \times n$ parity-check matrix of this code. The target ONU computes the syndrome $\mathbf{s}' = \mathbf{H} \cdot \mathbf{r}'^T$, where ^{T} denotes transposition, and transmits it to the OLT. This step is shown as Action (3) in Fig. 5.

Transmission of $\mathbf{s}'$ occurs at normal (high) power, such that the probability that $\mathbf{s}'$ is received in error by the OLT is negligible. Optionally, some integrity check mechanism can also be employed over $\mathbf{s}'$ to ensure that it is received correctly. On the other hand, it is important to stress that in this protocol the ONU does not need to correct the errors and recover $\mathbf{r}$. Moreover, a decoding attempt performed by the ONU would probably fail, as $\mathbf{r}$ is a random-like vector normally at a large distance from any codeword (hence decoding does not permit to find $\mathbf{e}$).

Once having received $\mathbf{s}'$, the OLT computes a new syndrome $\mathbf{s} = \mathbf{H} \cdot \mathbf{r}^T \oplus \mathbf{s}' = \mathbf{H} \cdot (\mathbf{r} \oplus \mathbf{r}')^T = \mathbf{H} \cdot \mathbf{e}^T$ and, by using syndrome decoding, recovers the error pattern $\mathbf{e}$. At this point, the OLT can compute $\mathbf{r}' = \mathbf{r} \oplus \mathbf{e}$, that therefore can be used to generate the shared secret key for downstream transmission towards the target ONU. This step is shown as Action (4) in Fig. 5. A malicious user connected to the same BS of the target ONU also receives $\mathbf{r}$, during Action (2), but affected by a different error pattern $\bar{\mathbf{e}}$, i.e., he/she receives $\mathbf{r}'' = \mathbf{r} \oplus \bar{\mathbf{e}}$. He/she can try two kinds of attacks, that are described next.

For the sake of convenience, in the following we will denote the OLT as Alice, the target ONU as Bob and the eavesdropping ONU as Eve.

4.1. First attack strategy

As a first attack, Eve can attempt to recover $\mathbf{r}$ by removing $\bar{\mathbf{e}}$. Similarly to Bob, she cannot exploit decoding but she can succeed, in principle, by brute force, which however is feasible only if the number of errors is small. On the other hand, if this attack is successful, Eve is not only able to recover the secret key but even to impersonate Alice. Therefore, the attack is particularly dangerous. To prevent it, we must fix a minimum weight for $\bar{\mathbf{e}}$, denoted as $t_{\text{min}}^{(E)}$, such that recovering $\mathbf{r}$ for the attacker is too difficult, and we must require, through a suitable choice of the design parameters, that the probability that the weight of $\bar{\mathbf{e}}$ becomes smaller than $t_{\text{min}}^{(E)}$ is sufficiently low.

4.2. Second attack strategy

As a second attack, Eve can try to exploit syndrome decoding to recover the overall error pattern $\mathbf{e} \oplus \bar{\mathbf{e}}$. This is possible because Eve receives $\mathbf{s}'$, and can then compute $\mathbf{s}'' = \mathbf{H} \cdot \mathbf{r}''^T \oplus \mathbf{s}' = \mathbf{H} \cdot (\mathbf{r}'' \oplus \mathbf{r}')^T = \mathbf{H} \cdot (\mathbf{e} \oplus \bar{\mathbf{e}})^T$. By using syndrome decoding, she could recover $\mathbf{e} \oplus \bar{\mathbf{e}}$ from $\mathbf{s}''$.

Let us suppose that the supports of $\mathbf{e}$ and $\bar{\mathbf{e}}$ are disjoint, that is, they do not have any entry equal to 1 in the same position. This condition allows us to simplify the analysis, and it is likely to occur since we consider large values of n and very sparse error vectors. Nevertheless, this hypothesis can be removed by resorting to a more elaborated analysis, that is left for future works. Let us denote by $t_{\text{min}}^{(B)}$ the minimum Hamming weight of $\mathbf{e}$. We hence impose that $t_{\text{min}}^{(B)} \leq t \leq t_2$. This can be checked by Bob. In fact, once having received $\mathbf{s}'$,

- if Bob's decoding fails, it means that $t > t_2$, and the distilled key is different between Alice and Bob, resulting in decryption errors;
- if Bob's decoding is successful, Bob can compute the weight of $\mathbf{e}$ and check whether $t < t_{\text{min}}^{(B)}$.

In both these cases, Alice and Bob recognize that something was wrong, and they restart the procedure, until these conditions are satisfied.

Under the above assumption of disjoint supports of $\mathbf{e}$ and $\bar{\mathbf{e}}$, the weight of $\mathbf{e} \oplus \bar{\mathbf{e}}$ is $t_{\min}^{(B)} + t_{\min}^{(E)}$ or more. So, in order to avoid successful decoding for Eve, it is necessary to have $t_{\min}^{(B)} + t_{\min}^{(E)} > t_2$. In this case, however, Eve could still try to randomly compensate $t_{\min}^{(B)} + t_{\min}^{(E)} - t_2$ errors before performing decoding. Therefore, we also require that this yields an infeasible complexity for Eve.

The values of the parameters must be designed to satisfy all previous requirements. The relevant relationships are presented in Section 5 together with a numerical example. As regards security, we refer to computational security which means fixing the number of attempts the eavesdropper should make, on average, to recover the secret. We speak of S_l-bit security if the number of attempts is $\geq 2^{S_l}$. So, the proposed approach relies on a computational security paradigm, which is common in classical cryptography, opposed to unconditional security achievable through QKD techniques. As a counterpart, differently from QKD, the proposed approach can be implemented with low-cost commercial devices.

The above considerations assume that the information on the key is contained into a single transmitted block but, in the most general case, we can assume that $L \geq 1$ blocks are used for such a purpose. Once having fixed the codeword length n and the BER, as given by (3), based on the chosen level of security we can find the optimal number of blocks to securely share the secret key. A measure of the actual length of the distilled secret key is then given by its entropy. These quantitative aspects are clarified and discussed in the next section.

5. Parameters design and complexity assessment

In this section we design system parameters to achieve some given security level, and assess the complexity of the proposed approach, compared to the standard key derivation procedure.

5.1. Parameters design

Let us consider a code with codewords of length n bits each, able to correct t_2 errors, and a channel BER equal to $p = t_2/n$ for both Bob and Eve. So, the quality of the two channels is the same. The probability that one of them receives a codeword affected by i errors is

$$P_i = \binom{n}{i} p^i \cdot (1-p)^{n-i}. \tag{4}$$

Coherent with Section 4, we assume that the transmission of L consecutive blocks is used to generate the secret key. We require that Eve needs 2^{S_l} attempts or more to recover $\mathbf{r}$ through the attack described in Section 4.1. This occurs when Eve experiences a number of errors per block $\geq t_{\min}^{(E)}$, with

$$\binom{n}{t_{\min}^{(E)}}^L \geq 2^{S_l}. \tag{5}$$

In order to preserve security, the occurrence of less than $t_{\min}^{(E)}$ errors per block at Eve's must have a probability P_E such that

$$(P_E)^L = \left(\sum_{i=0}^{t_{\min}^{(E)}-1} P_i \right)^L \leq 2^{-S_l}. \tag{6}$$

Eqs. (5) and (6) allow to find pairs $\left(L, t_{\min}^{(E)}\right)$ that satisfy such security requirements. In addition, in order to avoid the attack described in Section 4.2, it must be

$$t_{\min}^{(B)} > t_2 - t_{\min}^{(E)} \tag{7}$$

which ensures that Eve is not able to decode successfully $\mathbf{e} \oplus \bar{\mathbf{e}}$. As mentioned in Section 4.2, Eve could still try to correct the errors exceeding t_2 by brute force. So, we must also impose that

$$\binom{n}{t_{\min}^{(B)} + t_{\min}^{(E)} - t_2}^L \geq 2^{S_l}. \tag{8}$$

Eq. (8) permits us to compute the minimum value of $t_{\min}^{(B)}$.

If t is out of the range $\left[t_{\min}^{(B)}, t_2\right]$ (as described in Section 4.2 the system is able to recognize when this occurs), key distillation fails and the procedure is

restarted. The probability that $t_{\min}^{(B)} \leq t \leq t_2$ is given by

$$P_B = \sum_{i=t_{\min}^{(B)}}^{t_2} P_i. \tag{9}$$

Considering that L blocks are used to distill and share the secret key, the probability that Bob experiences an acceptable number of errors in each block and secret key distillation is completed successfully is P_B^L. Hence, the procedure must be restarted by Alice with probability $1 - P_B^L$.

Finally, the binary entropy of the distilled key is

$$h \geq L \cdot \log_2 \binom{n}{t_{\min}^{(B)}}, \tag{10}$$

where the right-hand side term represents the binary entropy due to the random choice of L binary vectors having length n and weight $t_{\min}^{(B)}$. In order to distill a secret key, the L sequences shared between Alice and Bob are given as input to a hash function with digest length $\leq h$. Its output provides the secret key shared between Alice and Bob.

In order to provide a numerical example, let us consider a BCH(8191, 7294) code, which is able to correct up to $t_2 = 70$ errors. We suppose that the transmission power is adapted in such a way as to achieve an average channel BER equal to $p = t_2/n = 0.0085$ that, according to (4), determines the value of P_i. Let us fix a target security level of $S_l = 80$ bits. Possible solutions for the parameters $\left(t_{\min}^{(E)}, L\right)$ following from (5)-(6) are reported in Fig. 6. Obviously, the curve has to be considered only for integer values of $t_{\min}^{(E)}$. We observe that the value of L increases with $t_{\min}^{(E)}$.

For each pair $\left(L, t_{\min}^{(E)}\right)$, (7) and (8) allow to compute $t_{\min}^{(B)}$, from which P_B follows through (9). We can then compute P_B^L, which represents the probability that the key distillation process is successful after transmission of a sequence of L blocks. For the considered choices of $t_{\min}^{(E)}$, the values of P_B^L obtained through this procedure are reported in Fig. 7. Since it is convenient to maximize P_B^L, the best result is achieved for $t_{\min}^{(E)} = 20$ and therefore, from Fig. 6, for $L = 2$. Correspondingly, we have $P_B^L = 0.2614$, which implies that about 4 attempts

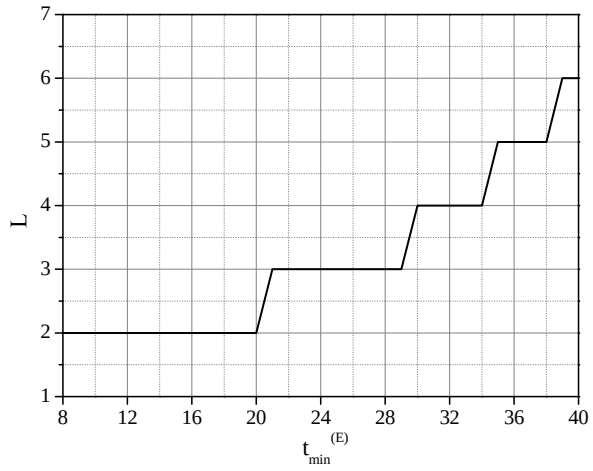

Figure 6: $\left(t_{\min}^{(E)}, L\right)$ valid pairs when using a BCH(8191, 7294) code able to correct $t_2 = 70$ errors, $p = t_2/n = 0.0085$ and $S_l = 80$ bits.

are required, on average, to distill a reliable and secure key. Because of the very high transmission rate, the latency required for completing the procedure is almost negligible and much smaller than any reasonable delay threshold set, if required, in the network.

The admissible pairs $\left(t_{\min}^{(B)}, t_{\min}^{(E)}\right)$ are shown, for the sake of completeness, in Fig. 8. From the figure we see that $t_{\min}^{(B)} = 54$ must be assumed for this specific example.

Finally, from (10) we have $h \geq 929.35$ bits, meaning that this procedure allows to distill and share secret keys that are significantly longer than those required by the AES-128 algorithm.

5.2. Complexity assessment

Let us assess the complexity of the proposed approach and compare it with that of the key derivation procedure recommended by the standard [1] and described in Section 3.1, which is affected by the vulnerability described in Section 3.2. For this purpose, we neglect the latency due to the exchange of key

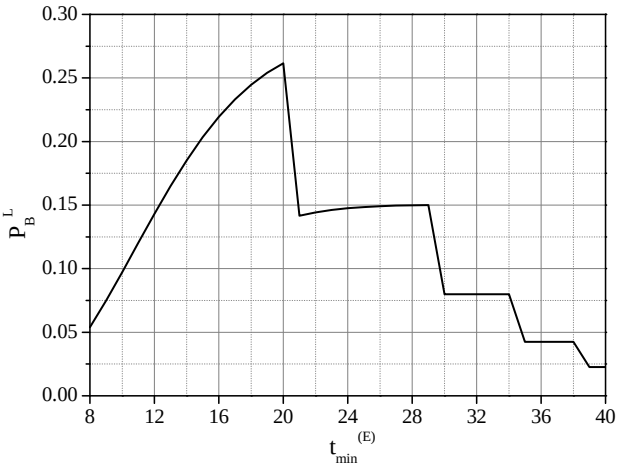

Figure 7: Probability of successful key distillation in L blocks, as a function of the minimum weight of $\bar{\mathbf{e}}$ $\left(t_{\min}^{(E)}\right)$, when using a BCH(8191, 7294) code able to correct $t_2 = 70$ errors, $p = t_2/n = 0.0085$ and $S_l = 80$ bits.

derivation messages, since communication in a PON occurs at very high data rates, while the bottleneck from the complexity standpoint is represented by computations.

Based on this assumption, complexity of the standard key derivation procedure is dominated by that of AES encryption. According to [27], the number of elementary operations required for performing one AES encryption can be estimated in

$$C_{\text{AES}} = (46N_bR - 30N_b)\,T_a + [31N_bR + 12\,(R-1) - 20N_b]\,T_o + [64N_bR + 96\,(R-1) - 61N_b]\,T_s,$$

where:

- N_b is the AES block length divided by 32,
- R is the number of AES rounds,
- T_a is the number of elementary operations required for computing a byte-wise AND,

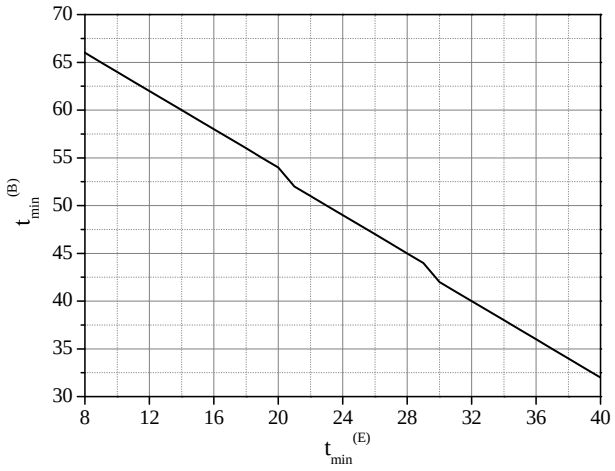

Figure 8: Minimum admissible number of errors at Bob's when using a BCH(8191, 7294) code able to correct $t_2 = 70$ errors, $p = t_2/n = 0.0085$ and $S_l = 80$ bits.

- T_o is the number of elementary operations required for computing a byte-wise OR,
- T_s is the number of elementary operations required for computing a byte-wise shift.

In our case, and according to Section 3.1, we can set $T_a = T_o = T_s = 8$, $N_b = 4$ and $R = 10$. By using these parameters, and taking into account that computing AES-CMAC of a 76-byte input sequence (see Section 3.1) requires 5 invocations of AES encryption, we obtain an overall complexity of the standard procedure in the order of

$$C_{\mathrm{PON}} = 5C_{\mathrm{AES}} = 246720 \tag{11}$$

elementary operations.

For what concerns complexity of the proposed method, it is clearly dominated by the complexity of BCH decoding. The latter can be estimated according to [28]. In fact, following [28, Table 2], we have that the total complexity of

syndrome-based decoding can be estimated in

$$C_{\mathrm{BCH}} = \left(3nt + 10t^2 - n + 6t\right) M + \left(3nt + 6t^2 - t\right) A + tI, \tag{12}$$

elementary operations, where n and t are the code length and number of correctable errors, while M, A and I are the costs of one multiplication, one addition and one inversion, respectively. Since BCH codes are defined as binary subfield subcodes of RS codes, we can consider $M = A = I = 1$ in (12). According to Section 5.1, we must consider $n = 8191$ and $t = 70$. In addition, we must consider that the proposed protocol requires performing decoding L/P_B^L times per key exchange, on average. According to Section 5.1, we have $L = 2$ and $P_B^L = 0.2614$. It follows that the number of elementary operations required by the proposed method based on PLS can be estimated in

$$C_{\mathrm{PLS}} = \frac{L}{P_B^L} C_{\mathrm{BCH}} = 26861890 \tag{13}$$

elementary operations, on average.

Based on the above considerations, we can conclude that the cost to be paid for overcoming the vulnerabilities of the standard XG-PON approach through the proposed PLS-based approach is an increase in complexity in the order of $C_{\mathrm{PLS}}/C_{\mathrm{PON}}$, that is, about two orders of magnitude. This may seem an important drawback of the proposed scheme. However, it has to be considered that both AES encryption and BCH decoding are very common tasks that can be executed in almost negligible time on modern hardware. Therefore, such a complexity increase, limited to the key derivation phase, may still not affect the overall system performance. Moreover, this solution has the merit to overcome the main limitations of the standard approach with an increase in complexity but without any significant change in the network architecture and devices. Other solutions, like those based on PKIs and QKD, instead require a more radical revision of the whole system.

6. Conclusion

Procedures based on PLS can provide a valid solution for secret key exchange in optical networks, overcoming the main limitations of standard approaches with a limited impact on complexity. We have presented a proposal exploiting the randomness in the distribution of errors due to very low power levels, that enables protected key generation and sharing through classical methods based on ECCs. This proposal appears interesting also in view of possible future updates of relevant standards.

References

[1] ITU-T G.987.3, *10-Gigabit-Capable Passive Optical Networks (XG-PON): Transmission Convergence (TC) Layer Specification*, 2014.

[2] K.-I. Kitayama, M. Sasaki, S. Araki, M. Tsubokawa, A. Tomita, K. Inoue, K. Harasawa, Y. Nagasako, and A. Takada, "Security in photonic networks: Threats and security enhancement," *J. Lightw. Technol.*, vol. 29, no. 21, pp. 3210–3222, 1 Nov. 2011.

[3] M. P. Fok, Z. Wang, Y. Deng, and P. R. Prucnal, "Optical layer security in fiber-optic networks," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 3, pp. 725–736, Sep. 2011.

[4] IEC 61300-3-20, *Fibre Optic Interconnecting Devices and Passive Components Basic Test and Measurement Procedures. Part 3-20: Examinations and Measurements Directivity of Fibre Optic Branching Devices*, First edition, 2001.

[5] V. O'Byme, "Verizons's fiber to the premises: Lessons learned," *Proc. Optical Fiber Communication Conference (OFC/NFOEC 2005)*, Anaheim, CA, 6-11 Mar. 2005, vol. 6, paper OWP6.

[6] D. Gutierrez, J. Cho, and L. G. Kazovsky, "TDM-PON security issues: Upstream encryption is needed," *Proc. Optical Fiber Communication Conference and Exposition and The National Fiber Optic Engineers Conference*, Anaheim, CA, 25 Mar. 2007, paper JWA83.

[7] H. Rohde and D. Schupke, *Securing Passive Optical Networks against Signal Injection Attacks*, in I. Tomkos, F. Neri, J. Sol Pareta, X. Masip Bruin, and S. Snchez Lopez (eds), Optical Network Design and Modeling. Lecture Notes in Computer Science, vol 4534, Springer, Berlin, Heidelberg, 2007.

[8] L. Schmalen, A. J. de Lind van Wijngaarden, and S. ten Brink, "Forward error correction in optical core and optical access networks," *Bell Labs Tech J.*, vol. 18, no. 3, pp. 39–66, Dec. 2013.

[9] W. Stallings, *Cryptography and Network Security: Principles and Practice*, 7th Edition, Pearson, 2016.

[10] O. Hirota, K. Katob, M. Shomac, and T. S. Usuda, "Quantum key distribution with unconditional security for all optical fiber network," *Proc. SPIE*, vol. 5161, pp. 320–331, 2004.

[11] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," *Proc. IEEE International Conference on Computers, Systems and Signal Processing*, Bangalore, India, 10-12 Dec. 1984, pp. 175-179.

[12] C. H. Bennett, "Quantum cryptography using any two nonorthogonal states," *Phys. Rev. Lett.*, vol. 68, no. 21, pp. 3121–3124, May 1992.

[13] E. Diamanti, H.-K. Lo, B. Qi and Z. Yuan, "Practical challenges in quantum key distribution," *Npj Quantum Information*, vol. 2, art. no. 16025, 2016.

[14] M. Bloch and J. Barros, *Physical-Layer Security: From Information Theory to Security Engineering*, Cambridge University Press, Cambridge, 2011.

[15] M. Baldi, F. Chiaraluce, N. Laurenti, S. Tomasin, and F. Renna, "Secrecy transmission on parallel channels: Theoretical limits and performance of practical codes," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 11, pp. 1765–1779, Nov. 2014.

[16] K. Ren, H. Su, and Q. Wang, "Secret key generation exploiting channel characteristics in wireless communications," *IEEE Wireless Commun.*, vol. 18, no. 4, pp. 6–12, Aug. 2011.

[17] M. Baldi, M. Bianchi, N. Maturo, and F. Chiaraluce, "A physical layer secured key distribution technique for IEEE 802.11g wireless networks," *IEEE Wireless Commun. Letts.*, vol. 2, no. 2, pp. 183–186, Apr. 2013.

[18] X. Porte, M. C. Soriano, D. Brunner, and I. Fischer, "Bidirectional private key exchange using delay-coupled semiconductor lasers," *Opt. Letts.*, vol. 41, no. 12, pp. 2871–2874, Jun. 2016.

[19] H. Koizumi, S. Morikatsu, H. Aida, T. Nozawa, I. Kakesu, A. Uchida, K. Yoshimura, J. Muramatsu, and P. Davis, "Information-theoretic secure key distribution based on common random-signal induced synchronization in unidirectionally-coupled cascades of semiconductor lasers," *Opt. Express*, vol. 21, no. 11, pp. 17869–17893, Jul. 2013.

[20] B. Wu, Y.-K. Huang, S. Zhang, B. J. Shastri, and P. R. Prucnal, "Long range secure key distribution over multiple amplified fiber spans based on environmental instabilities," *Proc. 2016 Conference on Lasers and Electro-Optics (CLEO)*, San Jose, CA, 5-10 Jun. 2016, Paper SF1F.4.

[21] K. Kravtsov, Z. Wang, W. Trappe, and P. R. Prucnal, "Physical layer secret key generation for fiber-optical networks," *Opt. Express*, vol. 21, no. 20, pp. 23756–23771, Oct. 2013.

[22] D. Hood and E. Trojer, *Gigabit-Capable Passive Optical Networks*, Wiley, Hoboken, 2012.

[23] NIST Special Publication 800-38B, *Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication*, Oct. 6, 2016.

[24] G. P. Agrawal, *Fiber-Optic Communication Systems*, Fourth Edition, Wiley, Hoboken, 2010.

[25] M. Nakazawa, H. Kubota, K. Suzuki, E. Yamada, and A. Saharaa, "Ultrahigh-speed long-distance TDM and WDM soliton transmission technologies," *IEEE J. Sel. Topics Quantum Electron.*, vol. 6, no. 2, pp. 363–396, Apr. 2000.

[26] Wireshark network protocol analyzer, https://www.wireshark.org/.

[27] M. Razvi Doomun, K. M. Sunjiv Soyjaudah and D. Bundhoo, "Energy consumption and computational analysis of rijndael-AES," *Proc. 3rd IEEE/IFIP International Conference in Central Asia on Internet*, Tashkent, 2007, pp. 1-6.

[28] N. Chen and Z. Yan, "Complexity analysis of Reed-Solomon decoding over $GF(2^m)$ without using syndromes," *EURASIP Journal on Wireless Communications and Networking*, vol. 2008, article ID 843634, 2008.

Authors biographies

Marco Baldi received his Laurea degree (Hons.) in electronic engineering and his Ph.D. degree in electronic, computer science and telecommunications engineering from the Polytechnic University of Marche, Ancona, Italy, in 2003 and 2006, respectively. Since 2016, he has been a tenure-track Assistant Professor with the same university. He has co-authored over 140 scientific papers, one book, and three patents. His research is focused on coding and cryptography for information security and reliability. He serves as an Associate Editor for the EURASIP Journal on Wireless Communications and Networking and the IEEE Communications Letters.

Franco Chiaraluce received his Laurea degree (Hons.) in Electronics Engineering from the University of Ancona, Italy, in 1985. Since 1987 he is with the Polytechnic University of Marche and at present he is an Associate Professor at the Department of Information Engineering, where he is in charge of several courses in the area of Telecommunications. His research interests are currently focused on error correcting codes, physical layer security and cryptography. He has co-authored over 300 scientific papers, 2 books and 3 patents. He is a Senior Member of IEEE and Member of IEICE.

Lorenzo Incipini received his B.Sc. degree in Telecommunication Engineering and the M.Sc. degree (Hons.) in Electronic Eengineering from the Polytechnic University of Marche, Italy. Actually he is pursuing the Ph.D. degree at Polytechnic University of Marche. His research interests are in the areas of Internet of Things, Visible Light Communication, and network security.

Marco Ruffini received his Ph.D from Trinity College Dublin in 2007, working on packet/optical networks. Since 2010, he has been Assistant Professor (tenured 2014) at TCD. He is Principal Investigator (PI) of CONNECT Telecommunications Research Centre at TCD, and IPIC photonics integration centre at Tyndall institute. Prof. Ruffini is currently involved in several Science Foundation Ireland and H2020 projects and leads the Optical Network Architecture group at Trinity College Dublin. He has authored over 110 international publications, over 10 patents, contributed to the BroadBand Forum standardisation body and he is associated editor for the OSA JOCN journal.