https://dx.doi.org/10.22161/ijaems.4.6.6 ISSN: 2454-1311

Performance Evaluation of Secure Key Distribution Based on the B92 Protocol

Brahim Ouchao[1], Abdeslam Jakimi[2]

[1]Department of Computer Science Faculty of Sciencs and Technics Errachidia, RO&I Team, Moualy Ismail University
ouchaob@gmail.com

[2]Department of Computer Science, GL&ISI Team, Faculty of Sciencs and Technics Errachidia, Moualy Ismail University
ajakimi@yahoo.fr

***Abstract*— *Quantum Cryptography (QKD) uses the laws of Quantum Mechanics to create new cryptographic primitives. Precisely, this technique is based on two laws of quantum mechanics, namely uncertainty principle and no-cloning theorem. The best known QKD is the BB84 protocol published by Bennett and Brassard in 1984. Since then, various QKD protocols have been developed for example a B92 protocol. This latter is similar to that of BB84but this time using only two non-orthogonal states of the four states of BB84. In this work, we develop the java simulation B92 protocol and we give some details of the implementation. Next, we show by an example how Alice (Initiator) can configure the protocol parameters before execution. Finally, we give the statistical results allowing to compare this protocol with that of BB84.*

Keywords*— *QKD, Qubit, BB84, B92.

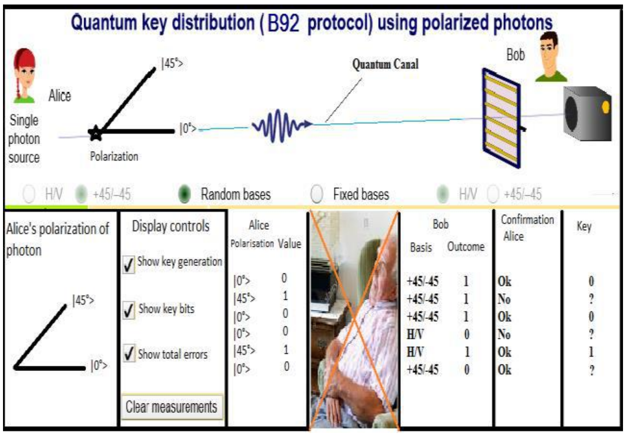
I. INTRODUCTION

The basic idea of the security of quantum transmissions was the instability of the quantum elements during the measurement. The B92 protocol is a simpler version of BB84 [1,2]. It is a two-state protocol (it uses two non-orthogonal quantum states) invented by Charles Bennett in 1992. It is based on the fact that two non-orthogonal quantum states are sufficient to guarantee the detection of an eavesdropper. Since no measurement can distinguish two non-orthogonal quantum qubit [3], it is impossible to identify the qubit with certainty. In addition, any attempt to learn the qubit will alter the qubit in a notable way. The coding scheme B92 is sometimes easier to implement. The coding scheme B92 uses a one-to-one correspondence between the conventional qubits and the quantum states. To send the qubit b, Alice prepares a photon in the following quantum state: 0° Horizontal polarization codes 0 and +45° Diagonal polarization for coding 1.

The rest of the paper is organized as follows: Section 2 presents and introduces our approach. Section 3 gives a predict the ear of a spy. Section 4 illustrates and presents the comparison of BB84 and B92. Section 5 concludes the paper and provides an outlook into future work.

II. CONCEPTION

Fig.1 shows the interface of our application that provides an implementation explaining a quantum transmission simulation based on the B92 protocol. Although it provides only a secret key distribution, the architecture can exploit to ensure a more secure transmission. Our application consists of designing a protocol running on a single computer or on two computers (servers) connected remotely via a socket connection [6].



○ H/V ● +45/–45 ● Random bases ○ Fixed bases ● H/V ○ +45/–45

☑ Show key generation
☑ Show key bits
☑ Show total errors

Alice Polarisation	Alice Value	Bob Basis	Bob Outcome	Confirmation Alice	Key
|0°>	0	+45/-45	1	Ok	0
|45°>	1	+45/-45	1	No	?
|0°>	0	+45/-45	1	Ok	0
|0°>	0	H/V	0	No	?
|45°>	1	H/V	1	Ok	1
|0°>	0	+45/-45	0	Ok	?

Fig. 1: Simulation of the B92 protocol without spy.

These servers can be used either to simulate a quantum channel, or to interact the two ends of a real quantum channel. The output generated by a protocol execution is a collection of HTML [7] files providing feedback for all relevant phases of the protocol. The software architecture consists of 4 main steps:

- Alice randomly prepares each photon with ether 0° Horizontal polarization assigned a qubit of value (|0>) or |+45> Diagonal polarization assigned a qubit of value (|1>) [4, 5].
- Alice sends the polarized photon to Bob. This one is equipped with a polarization analyzer that measures randomly along the two basis. that is to say according to the following four directions (0°, +45°, 90° or -45°)

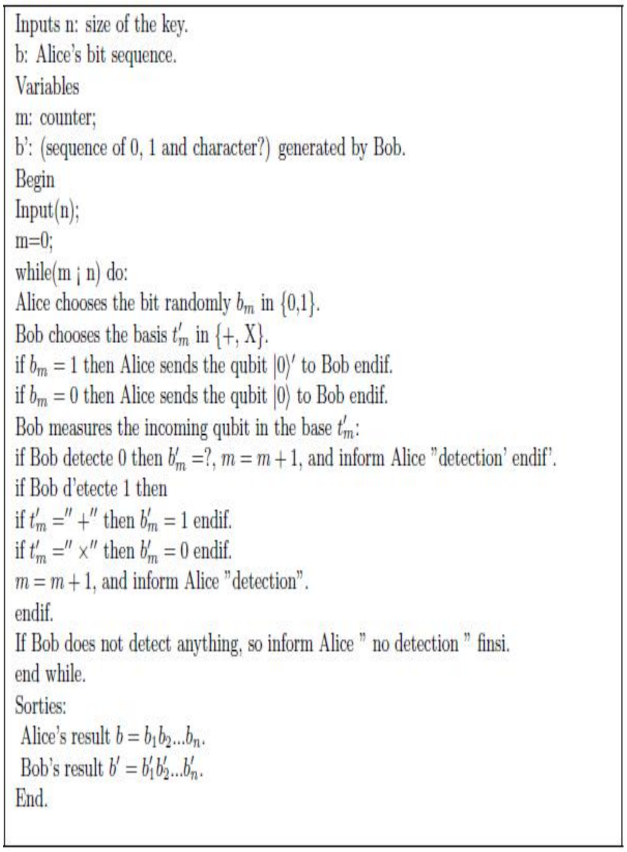
- Bob informs Alice via an authenticated public channel the basis used, but not the polarization of the photon.
- Alice and Bob keep the polarizations of the photons (qubits) sent and received in directions 45° and 90° and reject those received in directions 0° and 45°.

1.1. Example

For example, if Bob detects a photon by measuring with a polarized filter on the 90° direction, he knows that Alice has sent a polarization photon 45° (it cannot be 0° polarization) and therefore it encodes this value by the bit 1. Bob can therefore assign the detections 90° the binary values 1, and the detections 45° the value 0. If Bob does not detect the photon, he cannot be certain which state Alice sent. Thus, Alice and Bob keep only those measurements where Bob detected a photon. This sequence of 0 and 1 bits values forms the key.

1.2. Algorithm of B92

The algorithm of the B92 protocol is:



```
Inputs n: size of the key.
b: Alice's bit sequence.
Variables
m: counter;
b': (sequence of 0, 1 and character?) generated by Bob.
Begin
Input(n);
m=0;
while(m ¡ n) do:
Alice chooses the bit randomly $b_m$ in {0,1}.
Bob chooses the basis $t'_m$ in {+, X}.
if $b_m = 1$ then Alice sends the qubit $|0\rangle'$ to Bob endif.
if $b_m = 0$ then Alice sends the qubit $|0\rangle$ to Bob endif.
Bob measures the incoming qubit in the base $t'_m$:
if Bob detecte 0 then $b'_m =?$, $m = m + 1$, and inform Alice "detection' endif'.
if Bob d'etecte 1 then
if $t'_m ='' +''$ then $b'_m = 1$ endif.
if $t'_m ='' \times''$ then $b'_m = 0$ endif.
$m = m + 1$, and inform Alice "detection".
endif.
If Bob does not detect anything, so inform Alice " no detection " finsi.
end while.
Sorties:
 Alice's result $b = b_1 b_2 ... b_n$.
 Bob's result $b' = b'_1 b'_2 ... b'_n$.
End.
```

III. PREDICT THE EAR OF A SPY

1.3. Case: without spy

Alice and Bob communicate in an authenticated public channel to determine which photons have been detected. To do this, they exchange a small number of their bit values which are then rejected (because they are no longer secure) to check for errors. Assuming no spy has intervened, Alice and Bob will have the same qubit sequences. Unlike the BB84 protocol which extracts a sample from the shared key, the B92 protocol even exploits the rejected qubits to calculate the probability of transmission error. It leaves the key intact. After publicly communicating, Alice and Bob find half of these qubits rejected common. Which shows the absence of espionage listening. This is the ideal case for negligence and errors due to the impurity of the transmission channel. Fig 2. illustrates the results obtained

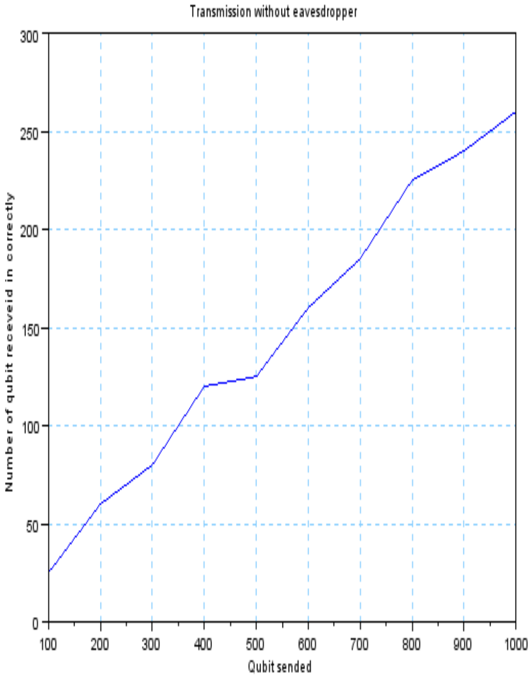
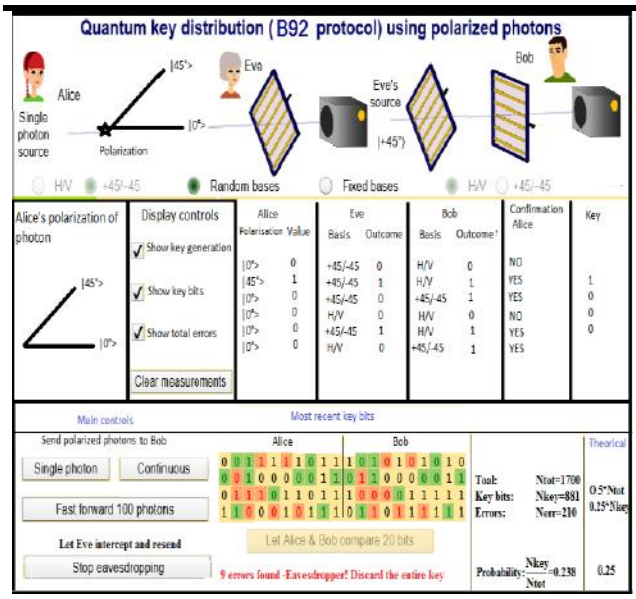


Fig. 2: Average number of qubits received correctly in the absence of the Spy.

1.4. Case: with spy

Figure3 illustrates the interface of the application in the presence of a spy. The analysis of the results obtained shows that the increase in the measurement rate of the spy disrupts the transmission. Indeed, if the probability of error is better than 0:25, then the transmission is no longer secure.



Quantum key distribution (B92 protocol) using polarized photons

Display controls

☑ Show key generation

☑ Show key bits

☑ Show total errors

Clear measurements

Alice		Eve		Bob		Confirmation Alice	Key
Polarisation	Value	Basis	Outcome	Basis	Outcome		
|0°>	0	+45/-45	0	H/V	0	NO	
|45°>	1	+45/-45	1	H/V	1	YES	1
|0°>	0	+45/-45	0	+45/-45	1	YES	0
|0°>	0	H/V	0	H/V	0	NO	0
|0°>	0	+45/-45	1	H/V	1	YES	0
|0°>	0	H/V	0	+45/-45	1	YES	

Main controls

Send polarized photons to Bob

Single photon | Continuous

Fast forward 100 photons

Let Eve intercept and resend

Stop eavesdropping

Most recent key bits

Alice										Bob									
0	0	1	1	1	1	1	0	1	1	1	0	1	0	1	0	1	0	1	0
0	0	1	0	0	0	0	0	1	1	0	1	1	0	0	0	0	0	1	1
0	1	1	1	0	1	1	0	1	1	1	0	0	0	0	1	1	1	1	1
1	1	0	0	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1	1

Let Alice & Bob compare 20 bits

9 errors found -Eavesdropper! Discard the entire key

		Theorical
Toal:	Ntot=1700	0.5*Ntot
Key bits:	Nkey=881	0.25*Nkey
Errors:	Nerr=210	
Probability: $\frac{Nkey}{Ntot}$=0.238		0.25

Fig. 3: Simulation of B92 protocol with spy.

1.5. Interpretation

Fig. 4 illustrates the results obtained. It can be seen that the average number of qubits measured between Alice and Bob decreases compared to the previous case (case without spy). Certainly, as long as the error rate is below a threshold ($P_{error} < 0:25$) [4] the spy remains deaf.

IV. COMPARISON OF BB84 AND B92

1.6. View on BB84 and B92

With some simple modifications in BB84, one can adopt it in model B92. In this last protocol Alice does not encode the qubit randomly. If the generated bit is zero then it sends the polarized photon 0° if the bit is equal to one then the polarized photon 45° is exchanged. With some modifications in the code of BB84, we can adopt it in model B92.

1.7. Attack "Intercept Resend"

In terms of indiscreet detection, B92 is better than BB84. This is well explained in figure5 and figure6.

1.8. Analysis and interpretation

The probability of detection of spy converges faster to 1 when the number of quits exchanged is large. Thus the number of correct measurements between Alice and Bob is reduced; this is due to the listening of the spy's indiscreet ear.

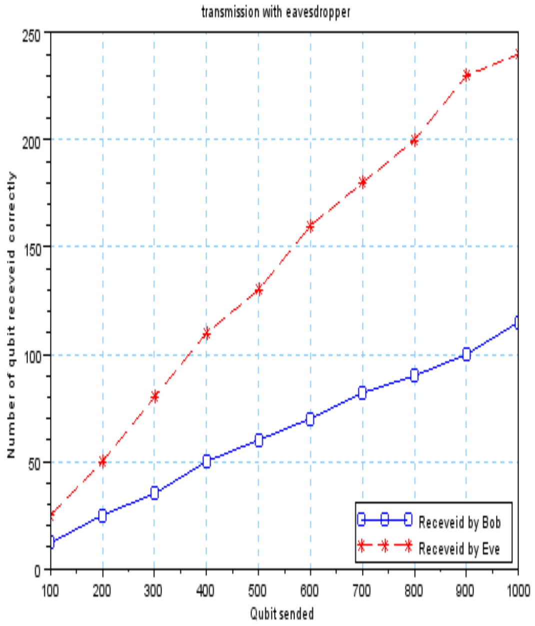


Fig. 4 :Average number of qubits received in the same databases in thepresence of a only spy.

In addition, the simplicity of implementing the B92 protocol and the Figure 5: BB84 and B92 comparison for Intercept Resend correct measurements by eavesdropper. Number of de-mined exchange bit makes this protocol more efficient compared to its BB84 ancestor.

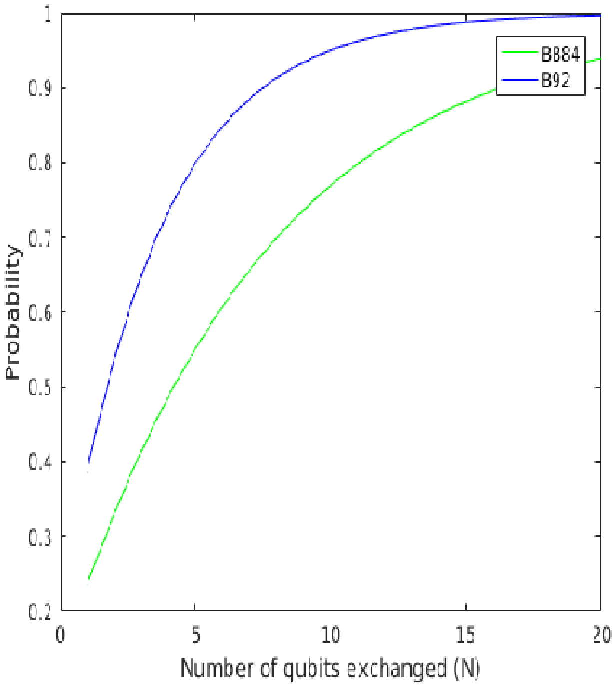


Fig. 5: BB84 and B92 Comparison for Intercept Resend eavesdropper detection.

V. CONCLUSION

In this work we have developed a java simulation of protocol B92 (BB84 version) for key quantum exchange. The analysis of the results obtained allowed us to define a security zone where the presence of the spy does not influence the transmission. This study also confirmed to us, after a statistical comparison, the importance of the B92 protocol in relation to its BB84 ancestor and this, even if their main steps are similar.

As future works, we will evaluate our approach on more complex systems as quantum teleportation. We will study the possibility of developing the java simulation, comparing and giving more details of the implementation.

REFERENCES

[1] Ch.H. Bennett, G. Brassard and A. Ekert, Quantum cryptography,Scienti_c Am. 267,pp. 26-33, 1992 (int. ed.).

[2] Ch.H. Bennett, G. Brassard, Quantum cryptography: Public key distribution and coin tossinh. In Proceedings of the IEEE International Conference on Computers, Systems, and Signal Processing, Bangalore, India, IEEE, New York, pp. 175-179, 1984.

[3] B. Ouchao and E. H. El Kinani, Statistical Analysis of Common Qubits between Alice and Bob in BB84 Protocol, Contemporary Engineering Sciences, 4, 8, 363-370, 2011.

[4] B. Ouchao and E. H. El Kinani, Analysis of the Inuence of the Rate of Spies'Measure on the Quantum Transmission, Science Journal of Mathematics and Statistics, 2012, 7

[5] B. Ouchao and E. H. El Kinani, Etude statistique de l'inuence de pr_esence des espions: Simulation du protocole BB84 en langage Java, African Journal of Mathematical Physics, 11, 41-46 2012.

[6] V. S. Bagad and I. A. Dhotre, Computer Networks (5th revised edition, 2010 ed.), Technical Publications Pune, p. 52, 2008.

[7] Matthew MacDonald, \HTML 5 - The Missing Manual", 2nd ed, OReilly. 2014.8