Tạp chí Khoa học Đại học Quốc gia Hà Nội, Khoa học Tự nhiên và Công nghệ 24 (2008) 238-247

Mô phỏng mật mã lượng tử theo giao thức BB84

Lê Minh Thanh*

Học viện Công nghệ Bưu chính Viễn thông
122 Hoàng Quốc Việt, Hà Nội, Việt Nam

Nhận ngày 28 tháng 3 năm 2008

Tóm tắt. Mô phỏng mật mã lượng tử theo giao thức BB84 là hệ mô phỏng cách làm việc thực tế của hệ phân phối khóa lượng tử. Hệ thực hiện theo giao thức BB84 cho mật mã lượng tử bao gồm việc sử dụng xung phân cực ánh sáng để truyền thông tin trên kênh lượng tử và điều chỉnh quá trình phân bố thông tin đó bằng cách trao đổi một cách công khai trong một môi trường hoàn toàn mở. Phần mềm này cho phép người sử dụng thử nghiệm giao thức và kết quả nghiên cứu này nhằm chứng minh mật mã lượng tử có tính ưu việt hơn mật mã hiện nay [1].

1. Mục đích mô phỏng mật mã lượng tử

Mật mã lượng tử là loại mật mã dựa vào các tính chất của vật lý lượng tử, do đó đây là loại mật mã không thể tấn công bằng các sức mạnh tính toán.Vì vậy mật mã lượng tử có thể sẽ trở thành loại mật mã được sử dụng nhiều trong tương lai. Việc kết hợp phương pháp phân tích lí thuyết và công cụ lập trình Java để thiết kế chương trình mô phỏng nhằm kiểm chứng lại lý thuyết, làm cơ sở cho các thí nghiệm đánh giá tính an toàn của giao thức là minh chứng cho khả năng tạo ra sản phẩm mật mã lượng tử [2].

2. Giao thức truyền khóa lượng tử [3]

Trong mật mã lượng tử, các trạng thái phân cực khác nhau của photon được sử dụng để mã hoá và giải mã. Nếu chúng ta đo phân cực của một photon thông qua hệ đo phân cực theo đường thẳng thì các kết quả đo sẽ chỉ ra rằng photon đó phân cực thẳng đứng hay nằm ngang. Hoàn toàn tương tự như vậy cho hệ phân cực chéo.

Ta quy ước các ký hiệu như sau:

$\oplus$: thiết bị đo phân cực thẳng.

$\updownarrow$: phân cực thẳng đứng.

$\leftrightarrow$: phân cực thẳng ngang.

$\otimes$: thiết bị đo phân cực chéo.

$\nwarrow\!\!\searrow$: phân cực chéo hướng trái.

$\swarrow\!\!\nearrow$: phân cực chéo hướng phải.

Mã hoá và giải mã lượng tử thực hiện dựa trên trạng thái phân cực của photon. Trạng thái các photon khi đi qua các hệ đo phân cực khác nhau như sau :

- Một photon trong hệ phân cực thẳng thì có thể là phân cực thẳng đứng hoặc ngang

Photon 1 $\leftrightarrow \oplus \leftrightarrow$

Photon 2 $\updownarrow \oplus \updownarrow$

* ĐT: 84-4-8583186.
E-mail: minhthanh69@yahoo.com

- Nếu một photon được gửi liên tiếp qua các hệ đo phân cực giống nhau thì cho kết quả không đổi.

Photon 1: ↔ ⊕ ↔ . ⊕ ↔ ⊕ ↔

Photon 2 : ↕ ⊕ ↕ ⊕ ↕ ⊕ ↕

Một photon phân cực thẳng hoặc ngang nếu truyền qua hệ đo phân cực chéo sẽ cho kết quả là phân cực chéo trái hoặc phải (↕ ⊗ ↖↘ hoặc ↕ ⊗ ↙↗).

Kết quả hoàn toàn tương tự như vậy khi ta truyền một photon phân cực chéo qua một thiết bị phân cực thẳng.

Giao thức BB84 (do Bennett và Brassard giới thiệu năm 1984) dựa trên tính chất bất định và không thể sao chép được các trạng thái lượng tử. Kẻ nghe trộm trên đường truyền (Eve) không thể đọc thông tin mà không làm thay đổi các trạng thái lượng tử, vì vậy nếu Eve cố tình đọc thông tin thì sẽ để lại dấu vết và bị phát hiện. Sau khi truyền khoá xong, nếu phát hiện có kẻ nghe trộm thì có thể huỷ bỏ khoá đó và thực hiện truyền thông tin với một khoá khác, không làm ảnh hưởng đến thông tin cần bảo mật.

Quy ước Alice là người gửi thông tin, Bob là người nhận thông tin, còn Eve là người nghe trộm thông tin. Các bước của giao thức BB84 (bảng 1) để xác định khóa chung giữa người gửi và người nhận như sau:

Bước 1: Alice sẽ chọn ngẫu nhiên các photon theo cả hệ đo phân cực thẳng và hệ đo phân cực chéo.

Bảng 1. Minh hoạ giao thức BB84

TT	**Mô tả**	1	2	3	4	5	6	7	8	9	10	11	12
1	Hệ đo mà Alice sử dụng để đo các photon	⊕	⊕	⊗	⊕	⊗	⊗	⊗	⊕	⊕	⊕	⊗	⊗
2	Kết quả các phép đo của Alice gửi đến	↕	↔	↖↘	↔	↙↗	↖↘	↙↗	↕	↕	↕	↙↗	↖↘
3a	Các hệ đo Bob sử dụng	⊕	⊕	⊕	⊕	⊗	⊗	⊗	⊗	⊕	⊗	⊗	⊕
3b	Kết quả phép đo của Bob	↕	↔	↔	↔	↙↗	↖↘	↙↗	↖↘	↔	↖↘	↙↗	↔
4	Bob thông báo cho Alice các hệ đo Bob đã sử dụng	⊕	⊕	⊕	⊕	⊗	⊗	⊗	⊗	⊕	⊗	⊗	⊕
5	Alice thông báo cho Bob biết hệ đo đúng	đúng	đúng	sai	đúng	đúng	đúng	đúng	sai	đúng	sai	đúng	sai
6	Alice và Bob ghi lại dữ liệu từ các phép đo đúng	1	0	X	0	1	0	1	X	0	X	1	X

Bước 2: Alice ghi lại các trạng thái của các photon rồi gửi cho Bob.

Bước 3 : Bob nhận các photon và đo trạng thái phân cực một cách ngẫu nhiên theo hệ đo phân cực thẳng hoặc hệ đo phân cực chéo. Bob ghi lại hệ đo sử dụng để đo phân cực và kết quả các phép đo phân cực. Chú ý là kết quả này có thể khác kết quả của Alice nếu như hai người không sử dụng hệ đo giống nhau.

Bước 4: Bob thông báo cho Alice biết các hệ đo phân cực mà mình đã sử dụng, nhưng không thông báo kết quả các phép đo.

Bước 5: Alice thông báo cho Bob biết hệ đo nào là đúng. (Hệ đo đúng là hệ đo mà Alice và Bob cùng sử dụng để đo phân cực).

Bước 6: Alice và Bob sẽ loại bỏ các dữ liệu từ các phép đo không đúng. Các dữ liệu từ phép đo đúng sẽ được chuyển thành chuỗi các bít, theo các quy ước sau:

- Chéo trái ↖↘ : 1 , chéo phải ↙↗ : 0.
- Thẳng ngang ↔ : 1, thẳng đứng ↕ : 0.

Bảng 1 là một ví dụ minh hoạ giao thức BB84. Giả thiết Alice gửi đi 12 photon và Bob nhận được đầy đủ. Cuối cùng, Alice và Bob đã có một chuỗi các bit là: 1 0 0 1 0 1 0 1.

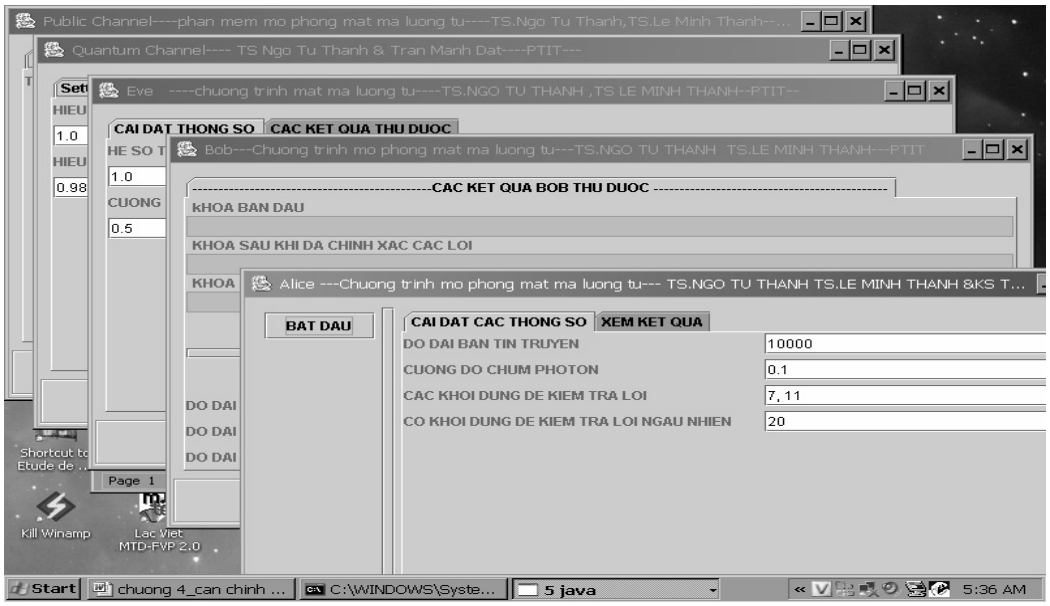
3. Các thành phần chính của phần mềm mô phỏng [4]

Phần mềm gồm 3 phần mô phỏng chính:

- Mô phỏng người gửi và người nhận.
- Mô phỏng người muốn lấy cắp thông tin.
- Mô phỏng kênh lượng tử.

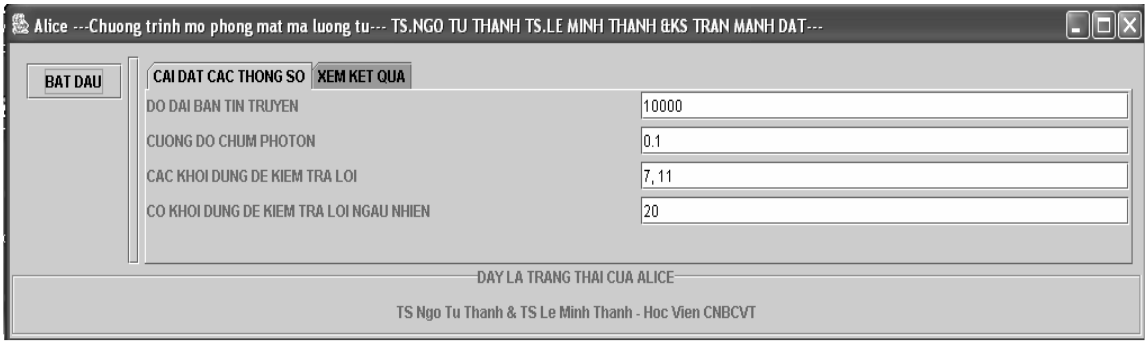
Người gửi và người nhận sẽ trao đổi thông tin với nhau đúng theo các giao thức mà người gửi và người nhận thông tin trong thực tế sử dụng. Hiển thị trên màn hình sẽ là các cửa sổ với các thông số tương ứng của từng nhân vật. Họ sẽ truyền thông cho nhau một cách trực tiếp trên một kênh lượng tử và với một môi trường hoàn toàn mở. Người sử dụng phần mềm có thể thay đổi các thông số cho từng nhân vật để tiến hành thí nghiệm bằng các dòng thông tin có độ dài khác nhau và các khoá có độ dài khác nhau.

Khi chạy sẽ xuất hiện các cửa sổ như sau:



Hình 1. Các cửa sổ thu được khi bắt đầu chạy chương trình.

Người nghe trộm thông tin là Eve có cửa sổ giao diện tương tự như giao diện người gửi (Alice) và người nhận (Bob). Khi sử dụng phần mềm này, có thể thay đổi các thông số tấn công



Hình 2. Cửa sổ cài đặt các thông số của Alice.

của Eve vào kênh lượng tử. Do đó ta dễ dàng thấy được những phản ứng khác nhau của kênh lượng tử nhằm bảo đảm an toàn cho thông tin truyền trên kênh này trước sự tấn công của Eve. Kênh lượng tử được sử dụng để xử lí trong quá trình mã hóa nhằm hiển thị trạng thái của kênh lượng tử. Ngoài ra nó còn cho phép người sử dụng có thể thí nghiệm trên phần mềm bằng cách cài đặt các thông số khác nhau cho kênh lượng tử. Qua đó chúng ta sẽ biết các mức tấn công khác nhau của Eve cùng mức độ mất dữ liệu và nhiễu của hệ thống.

4. Phân tích các chức năng cơ bản trong phần mềm

Cửa sổ CAI DAT CAC THONG SO (Hình 2)

Độ dài bản tin truyền: Thông số nhập vào phần này phải lớn hơn 10000 và nhỏ hơn 1000000.

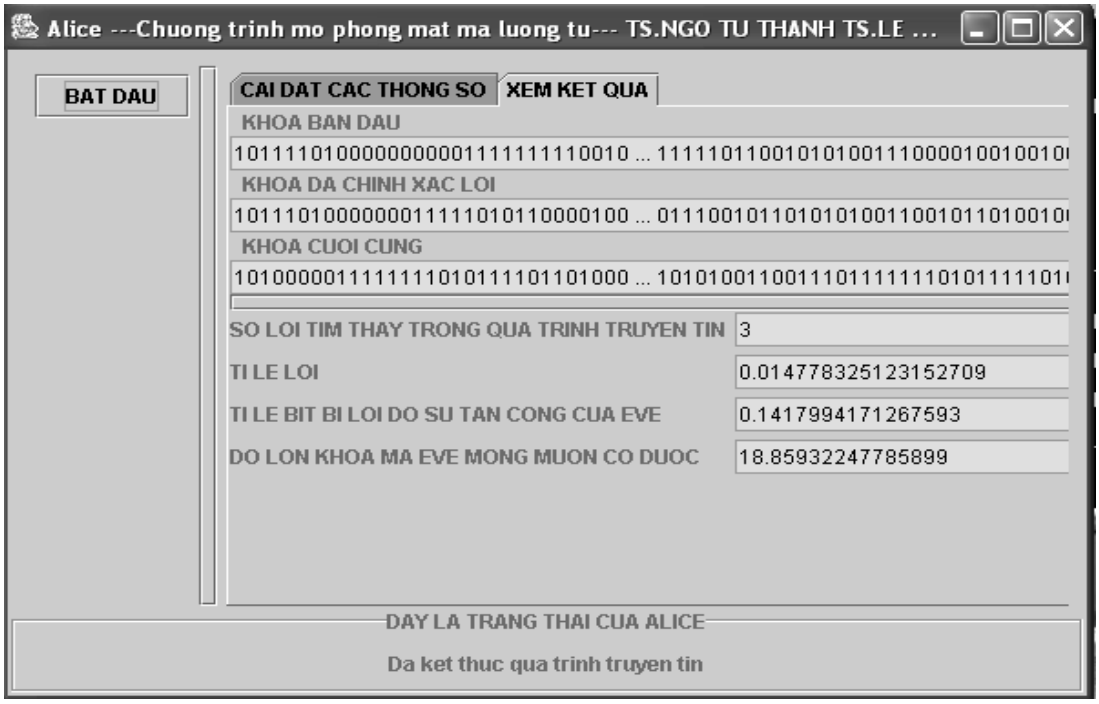
Vai trò của Alice chỉ rõ số lượng bít thông tin sẽ được truyền qua kênh lượng tử. Số lượng bít này được giới hạn bởi 1000000 bít.

Cường độ chùm photon: Thông số này phải là 1 số thực lớn hơn 0 và nhỏ hơn 1. Nó là tỉ lệ giữa số photon mà Bob và Eve đã nhận được trên tổng số photon trong 1 xung.

Các khối dùng để kiểm tra lỗi: Đây chính là số bít mà Alice sẽ sử dụng trong quá trình kiểm tra lỗi của chuỗi thông tin. Với mỗi số k này Alice sẽ kiểm tra từng cụm k bit của chuỗi thông tin cho đến hết, sau đó đem kết quả đó để so sánh tính chẵn lẻ với Bob.

Cỡ của khối ngẫu nhiên để kiểm tra lỗi: Trường này phải là một số nguyên lớn hơn 0. Để tăng khả năng giám sát và sửa lỗi thì sau mỗi lần Alice thực hiện kiểm tra lỗi thông thường xong sẽ thực hiện kiểm tra lỗi một lần nữa với một khối có độ lớn được chọn ngẫu nhiên. Có thể không tìm thấy bất cứ lỗi nào trong khối này sau khi hoàn tất quá trình kiểm tra.

***Cửa sổ KET QUA* của Alice** (Hình 3)



Hình 3. Hiển thị các kết quả của Alice.

Khóa ban đầu *:Đây chính là tóm tắt về các thông tin lượng tử mà Alice gửi trên kênh lượng* tử mà Bob đọc thành công. Nếu như hệ thống có thể đảm bảo Eve không nghe trộm được thông tin thì có thể đây sẽ là khoá cuối cùng (Final Key).

Khóa đã chính xác lỗi*: Đây là bản tóm tắt của khoá sau khi đã sửa các lỗi, đây là khoá được* sự đồng ý giữa Alice và Bob. Nó cho kết quả giống với kết quả của trường này trong cửa sổ của Bob.

Khóa cuối cùng*: Đây là bản tóm tắt của khoá riêng chia sẻ giữa Alice và Bob. Nó phải cho* kết quả giống như trường này trong cửa sổ của Bob.

Số lỗi tìm thấy trong quá trình truyền*: Đây là số lỗi mà Alice có thể phát hiện trong quá* trình truyền.

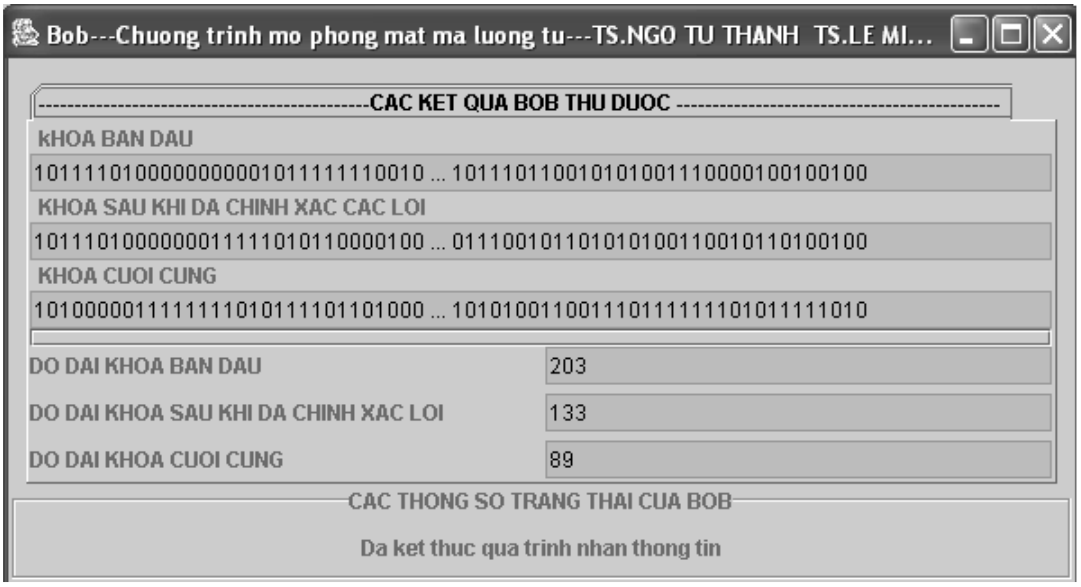
Tỉ lệ lỗi*: Là tỉ lệ của phần thông tin lượng tử ban đầu bị lỗi. Đây là lỗi gây ra do kênh lượng* tử chứ không phải do Bob.

Tỉ lệ lỗi bít do sự tấn công của Eve*: Là tỉ số bít lỗi do Eve tấn công vào kênh lượng tử trên* tổng số bít truyền qua kênh lượng tử.

Cỡ khóa mà Eve mong muốn có được*: Là số bít thông tin chính xác mà Eve mong muốn* được sở hữu. Đó là số bít tối thiểu mà dựa vào đó Eve có thể thu được các thông tin mà Alice gửi cho Bob.

Cửa sổ của Bob (Hình 4)

Là cửa sổ hiển thị các kết quả mà Bob thu được. Trong thực tế các trường khóa ban đầu, khóa đã chính xác lỗi, khóa cuối cùng của Bob sẽ cho các kết quả giống như các trường này trong cửa sổ của Alice. Các trường độ dài khóa ban đầu, độ dài khóa đã chính xác lỗi, độ dài khóa cuối cùng đơn giản chỉ là các thông số thể hiện độ dài của các khoá tương ứng.



Hình 4. Cửa sổ hiển thị các kết quả của Bob.

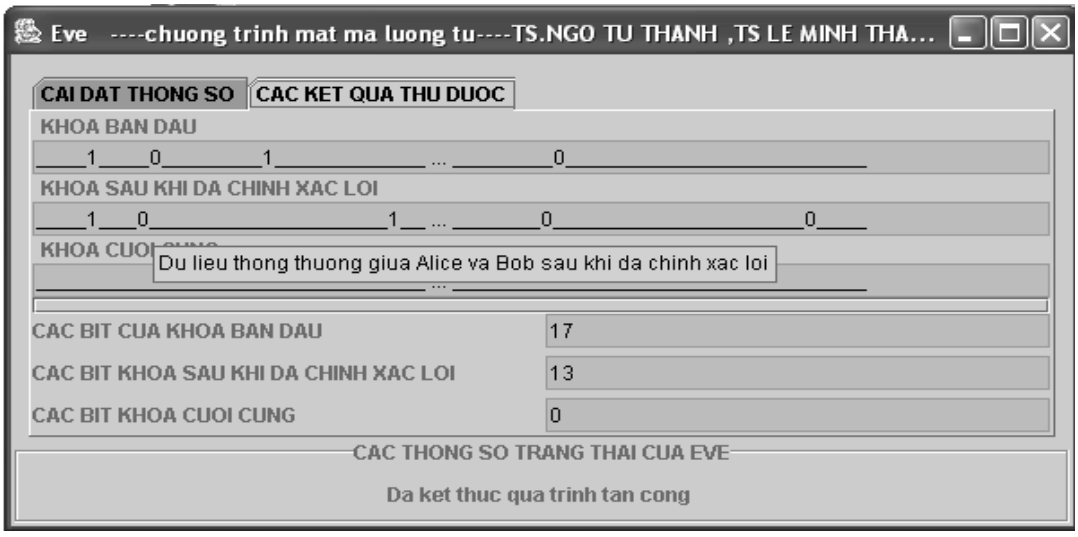
Cửa sổ của Eve (Hình 5)

Hình 5. Cửa sổ ban đầu của Eve – người nghe trộm thông tin.

Nếu cài đặt giá trị này thấp, ví dụ như bằng 0.1 thì có nghĩa là khả năng phân tách các xung là rất thấp. Ngược lại nếu cài đặt giá trị này là một số lớn, ví dụ như 0,9 thì hầu hết các xung đều sẽ bị phân tách.

Hệ số tách luồng: Trường này phải có giá trị nằm trong khoảng từ 0 đến 1.

Trường Kết quả (Hình 6)



Hình 6. Cửa sổ hiển thị các kết quả mà Eve thu được.

Cửa sổ kết quả của Eve cũng giống như cửa sổ kết quả của Bob, ba trường khóa ban đầu, khóa đã chính xác lỗi và khóa cuối cùng đơn giản chỉ là thể hiện các bit mà Eve thu được trong các khoá tương ứng. Trong đó Eve sẽ thể hiện các bít khoá mà mình không thu được thông qua một đường gạch ngang.

Các thông số: các bít khóa ban đầu, các bít khóa sau khi đã chính xác lỗi, các bít khóa cuối cùng đơn giản chỉ thể hiện số lượng các bít mà Eve biết được của các khoá tương ứng.

Kênh công khai (Hình 7)

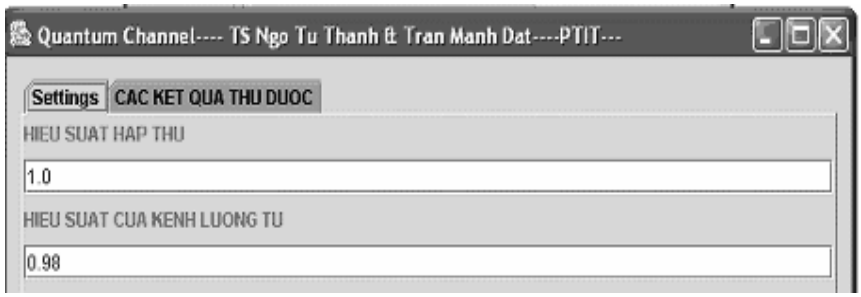
Cửa sổ 'kênh công khai' là cửa sổ đơn giản nhất trong các cửa sổ của chương trình. Tất cả những gì hiển thị trong cửa sổ này là một hộp thể hiện tổng số tín hiệu được truyền qua kênh. Mỗi một dấu hiệu truyền qua kênh được thể hiện bằng một đơn vị và do đó kết quả thu được chính là số tín hiệu mà Alice và Bob truyền cho nhau trong quá trình đàm phán để thiết lập khóa.

Hình 7. Cửa sổ hiển thị các kết quả thu được trên kênh công khai (công cộng).

Kênh lượng tử (Hình 8)

Cửa sổ cài đặt các thông số:

Hiệu suất hấp thụ: Trường này có giá trị lớn hơn hoặc bằng 0 và nhỏ hơn hoặc bằng 1.



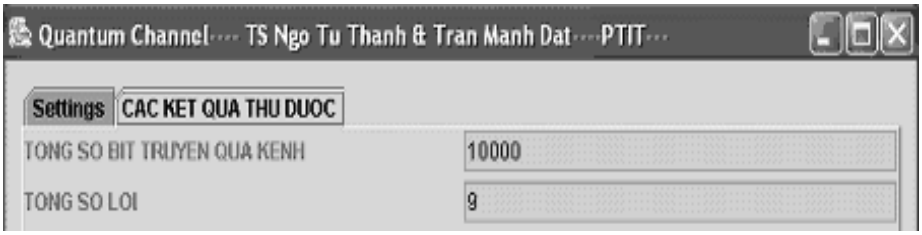
Hình 8. Cửa sổ ban đầu của kênh lượng tử.

Nếu như giá trị này đặt bằng 1 thì có nghĩa là nó sẽ không hấp thụ xung nào, ngược lại nếu như giá trị này đặt bằng 0 thì có nghĩa là nó sẽ hấp thụ tất cả các xung.

Hiệu suất kênh lượng tử*:* Trường này phải có giá trị lớn hơn hoặc bằng 0 và nhỏ hơn hoặc bằng 1. Trường này chỉ thị khả năng 1 chùm tia qua kênh lượng tử trong điều kiện không có nhiễu. Nếu như giá trị này được cài đặt bằng 1 thì có nghĩa là sẽ không có bất cứ bít nào bị đảo giá trị. Nếu như giá trị này bằng 0 thì có nghĩa là lỗi toàn kênh lượng tử, khi đó tất cả các bít đều bị đảo giá trị.

Trường kết quả (Hình 9)

.

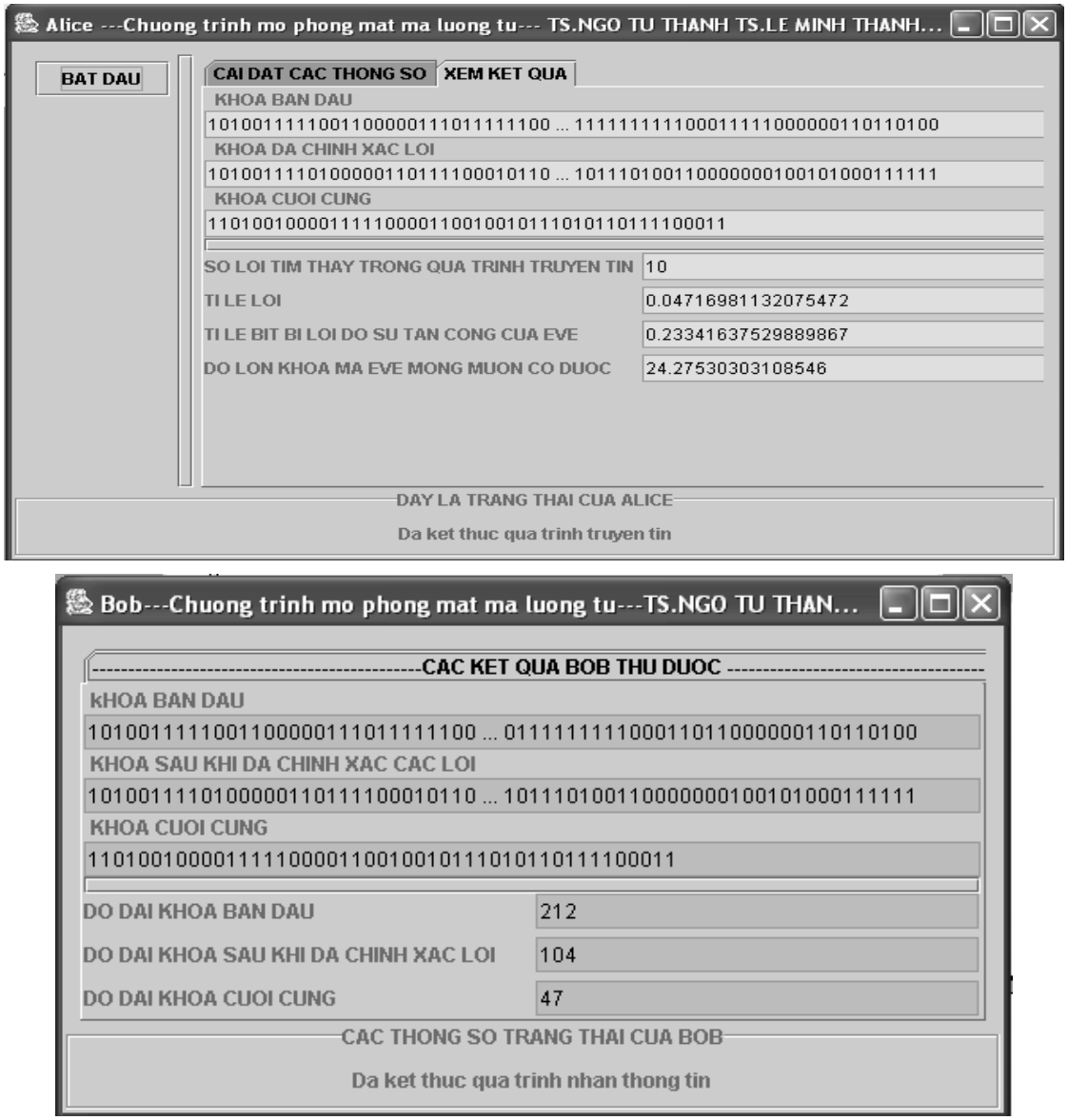


Hình 9. Các kết quả thu được của kênh lượng tử khi chạy chương trình.

Tổng số bít truyền qua kênh: Trường này hiển thị tổng số các bít được truyền qua kênh lượng tử. Giá trị này phải giống giá trị của trường Transmission Size trong cửa sổ của Alice.

Tổng số lỗi: Thể hiện tổng số lỗi trên kênh lượng tử, nhưng không bao gồm lỗi do Eve lập kế hoạch tấn công tạo nên.

5. Thử nghiệm và đánh giá kết quả của chương trình mô phỏng (Hình10)



Hình 10. Số liệu tương thích giữa 2 màn hình.

Thống nhất các thông số kỹ thuật giữa các màn hình

Trên màn hình (hình 10), cửa số Alice và Bob được sánh đôi nhau, dễ dàng cho thấy sự thống nhất về các thông số quan trọng giữa Alice và Bob như khóa cuối cùng, độ dài khóa. . .

Khảo sát sự thay đổi của các thông số kỹ thuật so với lý thuyết

Giữa lý thuyết và thực tế hoàn toàn tương thích nhau.

Thử nghiệm 1: nếu cường độ chùm photon là 0,5 thì độ dài từ mã là 264, nếu chọn cường độ chùm photon là 0,05 thì độ dài từ mã là 41.

Thử nghiệm 2: nếu độ dài bản tin là 100.000 thì độ dài từ mã là 1215, nếu chọn độ dài bản tin là 10.000 thì độ dài từ mã là 58.

Thử nghiệm 3: nếu hiệu suất lượng tử là 0 tức là nhiễu hoàn toàn thì độ dài từ mã là 0.

6. Kết luận

Từ việc phân tích các giao thức truyền khóa lượng tử, bằng công cụ lập trình Java phiên bản 1.3.1, chúng tôi đã xây dựng phần mềm mô phỏng mật mã lượng tử theo đúng giao thức BB84 [5]. Kết quả của chương trình phần mềm đã thiết kế được các giao diện chính, cho phép người dùng có thể cài đặt các thông số của Alice như : độ dài thông tin truyền đi, độ lớn các khối kiểm tra, các thông số của Eve như: khả năng tấn công kênh lượng tử,... Hệ thống sẽ đưa ra các khoá ban đầu và các khoá cuối cùng mà Alice và Bob thống nhất với nhau, nhưng quan trọng hơn là hệ thống đưa ra các khoá mà Eve thu được. Kết quả nghiên cứu này có thể được dùng cho việc nghiên cứu thử nghiệm về khả năng bảo mật vô điều kiện của mật mã lượng tử [6].

Tài liệu tham khảo

[1] Nguyễn Bình, *Giáo trình mật mã học,* Học viện Công nghệ BCVT, 2005.

[2] Vũ Đình Cự, Máy điện toán lượng tử, xu thế phát triển trong vài thập kỷ tới, *Kỷ yếu hội thảo ICT.rda'*, 2/2003.

[3] Đỗ Quang Hưng, Mật mã lượng tử, một hướng mới trong bảo mật, *Tạp chí Bưu chính Viễn thông*, 9/2005.

[4] David Cuthbertson, *A Quantum Cryptography Simulation,* School of Computer Science University of St Andrews, 23rd April 2000.

[5] Matthias Scholz, *Quantum key Distribution via BB84 An Advanced Lab Experiment*.

[6] Kyo Inoue - Kaoru Shimizu, *Quantum Crytography – Quantum Mechanics Open up a New Trend In Communication Security,* 2003.

Quantum Cryptography Simulation For BB84 Protocol

Le Minh Thanh

Post and Telecommunications Institute of Technology
122 Hoang Quoc Viet, Hanoi, Vietnam

Quantum Cryptography Simulation For BB84 Protocol (QCS84) is a system to simulate a practical working model of a quantum key distribution system. It implements the BB84 protocol for quantum cryptography involving distribution of information over a quantum channel using pulses of polarised light and reconciliation of that information by public discussion over an open channel.The software allows the user to experiment the protocol and the results show that quantum cryptography has better current Cryptography.