Phys 396 2018

Lecture 12

Tues: Turn in HW

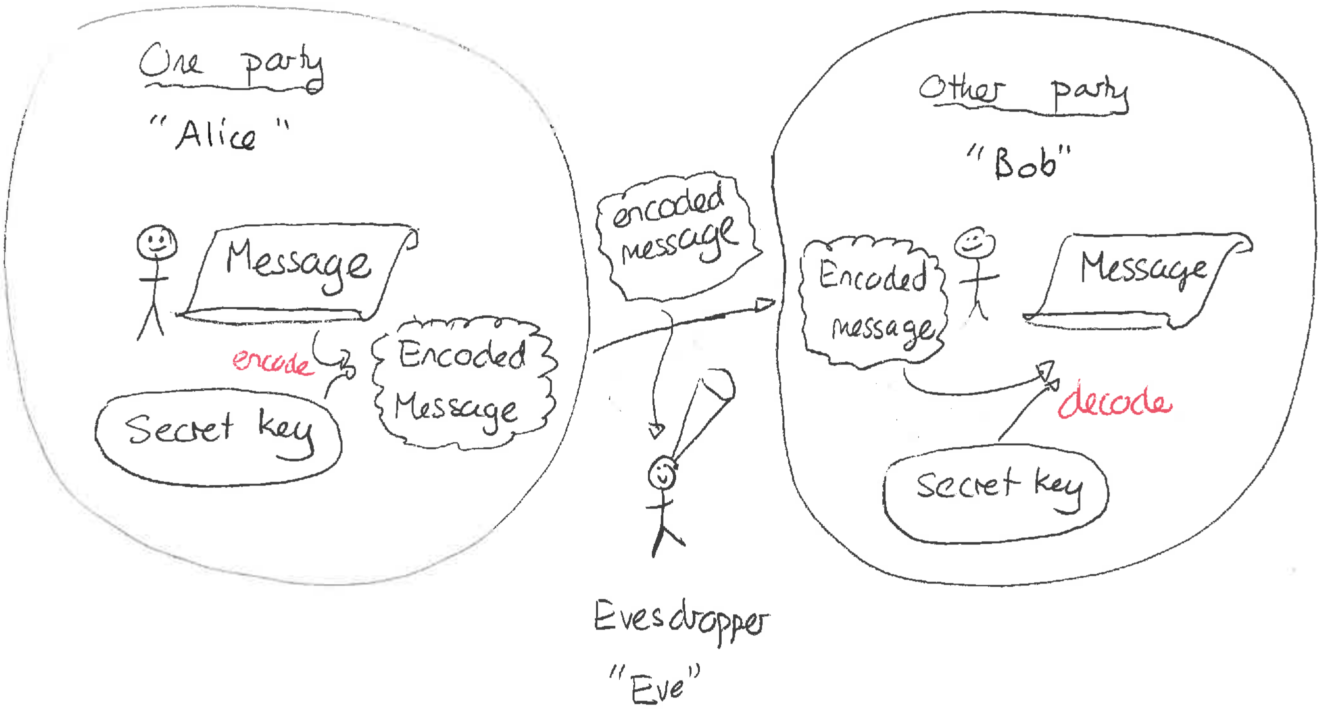
Thurs: Seminar

Tues 9 Oct : Exam

Covers up to latest HW

Cryptography B Ch4.1

Cryptography involves sending a message between two parties such that no evesdropper can intercept + understand the message. The basic scenario is



The true message is called the "plaintext." The encoded message is called the "cypher text." We shall explore one particular encoding scheme and see that it relies on a shared secret key.

Traditionally secret keys can be intercepted by an evesdropper and she can use these to decode and retransmit the cypher text without being detected and also learning the plaintext message.

Vernam cypher

Consider a piece of text converted into ASCII. For example

GJ ≡

71 $= \underline{0}\,\underline{1}\,\underline{0}\,\underline{0}\,\underline{0}\,\underline{1}\,\underline{1}\,\underline{1}$

64 32 16 8 4 2 1

74 $= 0\,\underline{1}\,\underline{0}\,\underline{0}\,\underline{1}\,\underline{0}\,\underline{1}\,\underline{0}$

64 8

use IBM binary ASCII table

We can encode this bit-by-bit as follows:

1) List the plain text digits via bits as:

$p_N p_{N-1} \dots p_1$ e.g. 01000111 ⁏ 01001010

2) List the key digits as

$k_N k_{N-1} \dots k_1$ e.g. 10101101 ⁏ 11110101

3) Perform bit-wise addition to generate a secret key:

$S_N S_{N-1} \dots S_1$

where $S_j = p_j \oplus k_j$.

In the example we would need a 16 digit secret key. The resulting cypher text is:

$$\begin{array}{c|c|c|c|c}
\vec{p} & 0100 & 0111 & 0100 & 1010 \\
\vec{k} & 1010 & 1101 & 1111 & 0101 \\
\hline
\bar{s} & 1110 & 1010 & 1011 & 1111
\end{array}$$

The cypher text can be decoded via bit wise addition with the same key.

$$\begin{array}{c|c|c|c|c} \bar{S} & 1110 & 1010 & 1011 & 1111 \\ \bar{k} & 1010 & 1101 & 1111 & 0101 \\ \hline & \multicolumn{2}{c}{\underbrace{0100 \quad 0111}_{\equiv G}} & \multicolumn{2}{c}{\underbrace{0100 \quad 1010}_{\equiv J}} \end{array}$$

We can see that the procedure always encodes + decodes correctly. This is called the Vernam cypher and is provably secure provided that:

1) the key is secret
2) the key is not reused.

Thus we need to generate shared secret keys.

1 Vernam cypher

a) "Alice" picks a short secret word, expresses it in terms of Ascii code (the plain text) and keeps this secret.

b) "Alice" generates a secret binary key and shares it with "Bob."

c) "Alice" does bitwise addition between the secret key and the plain text and sends the resulting cypher text to "Bob."

d) "Bob" does bitwise addition between the secret key and the cypher text. Does this match the original plain text? Does Eve know the plain text?

Quantum systems for cryptography

We could use qubits to transmit a potential secret key. One simple strategy might be for Alice to send a stream of photons, each either in the $|\rightarrow\rangle$ or $|\uparrow\rangle$ state to Bob. Alice would use a random number generator to choose which to send. Bob would measure (using a PBS) and record the result. They could then assemble a key.

In general terms let

$$0 \leftrightsquigarrow \rightarrow \qquad \text{so} \qquad |0\rangle \equiv |\rightarrow\rangle$$

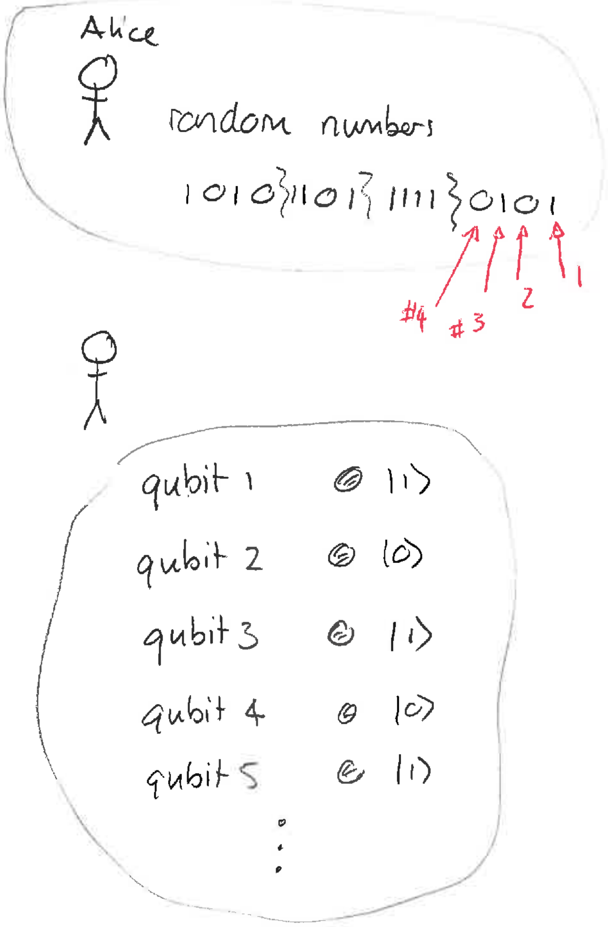
$$1 \leftrightsquigarrow \uparrow \qquad\qquad\qquad |1\rangle \equiv |\uparrow\rangle.$$

The procedure would be:

Alice uses a stream of random numbers to prepare a key.

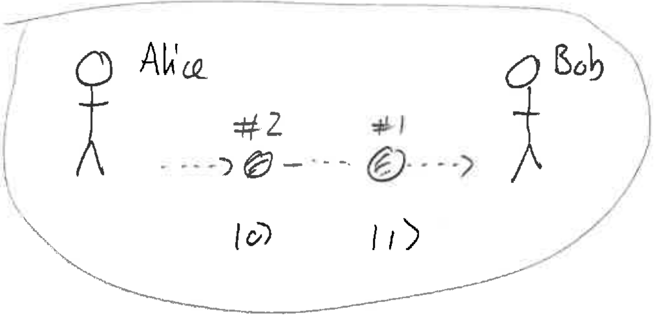
Alice prepares one qubit (e.g. photon) for each random digit according to

random digit	state
0	$|0\rangle \equiv |\rightarrow\rangle$
1	$|1\rangle \equiv |\uparrow\rangle$



qubit 1	⊛	$|1\rangle$
qubit 2	⊛	$|0\rangle$
qubit 3	⊛	$|1\rangle$
qubit 4	⊛	$|0\rangle$
qubit 5	⊛	$|1\rangle$
⋮		

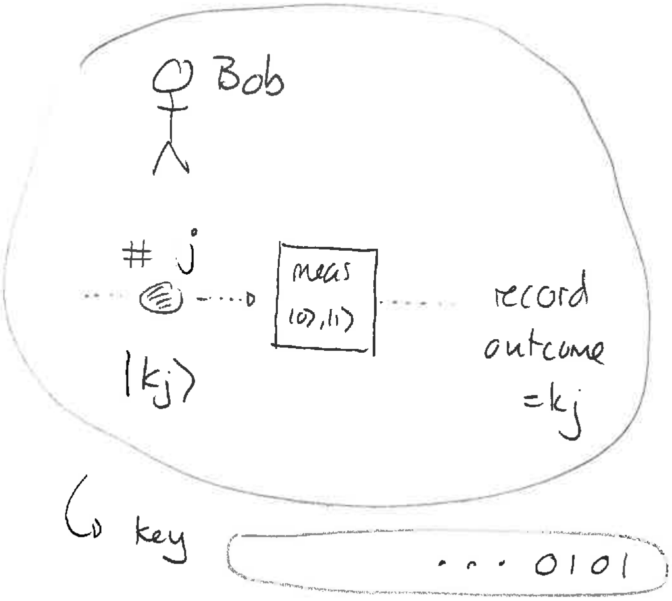
Alice transmits the qubits one at a time to Bob. They and possible evesdroppers agree on which is #1 #2,...



Bob measures in the $|0\rangle, |1\rangle$ basis (or H/V) and records result via

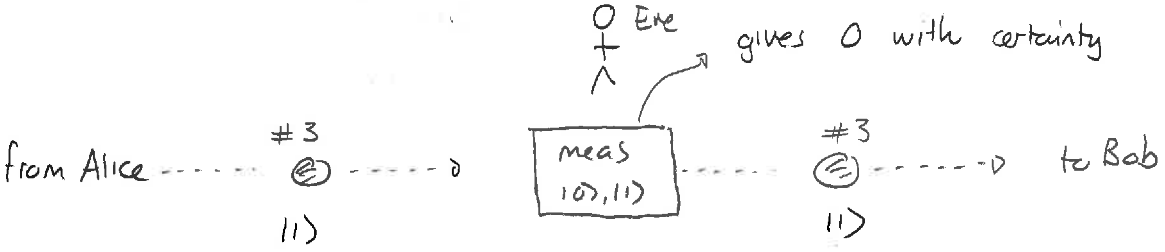
outcome 0 ⇝ bit = 0

outcome 1 ⇝ " = 1



Clearly whenever Alice transmits a $|0\rangle$, Bob's measurement yields 0 with certainty. Thus their keys will match perfectly.

The trouble is that any evesdropper could intercept the qubits, measure in the same basis, record the result and retransmit the qubit



Eve would learn the key perfectly without affecting the key that Bob and Alice obtain. They would not be able to detect her activity

Quantum cryptography: BB84 scheme

One way to avoid such evesdropping is the following scheme due to Giles Brassard + Charles Bennet (1984). The crucial idea is that

Alice / Bob use states and measuwements from either of the bases

$$\{|0\rangle, |1\rangle\}$$

or

$$\left\{ |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle+|1\rangle) \;,\; |-\rangle = \frac{1}{\sqrt{2}}(|0\rangle-|1\rangle) \right\}$$

Denote these bases via

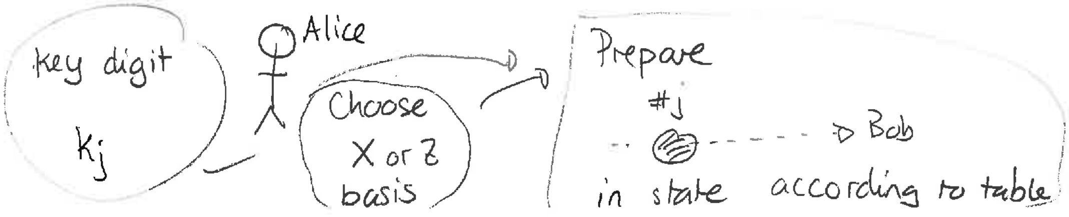
$$Z \equiv \{|0\rangle, |1\rangle\}$$

$$X \equiv \{|+\rangle, |-\rangle\}$$

Alice will use key bit values to prepare states via:

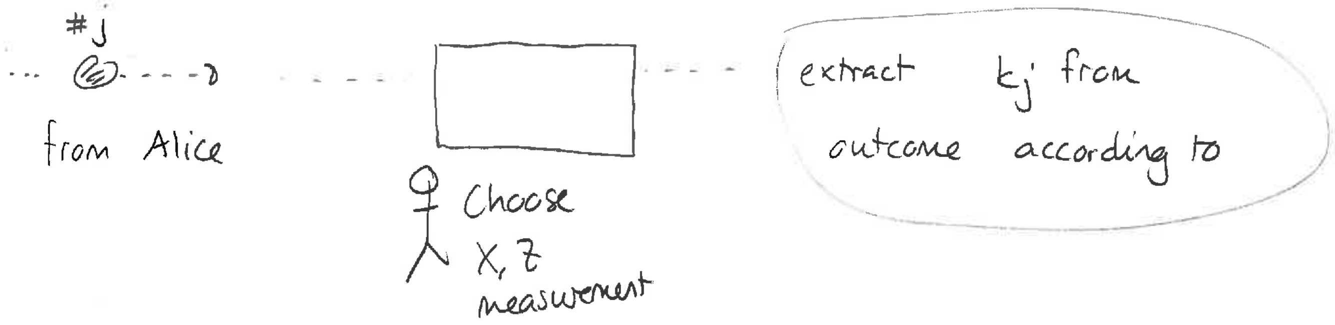
Key bits	basis choice	prepare qubit in state
0	Z	$\|0\rangle$
1	Z	$\|1\rangle$
0	X	$\|+\rangle$
1	X	$\|-\rangle$

So



Bob will extract key bit values via

Measurement choice	measurement outcome	bit value extracted
Z	0	0
Z	1	1
X	+	0
X	−	1



Consider this in the absence of any evesdropping. In the following exercise we see that:

1) whenever Alice and Bob's measurements coincide their bit values agree

2) whenever they do not coincide the bit values may not agree.

After every qubit has been processed, Alice + Bob communicate their basis choice. But this does not indicate the bit value and these are a secret.

	ALICE			BOB		Bases	Bit	
Qubit	Alice key	Alice basis choice		Bob basis choice	Outcome	Agree	A	B
1	1	X		X	1	✓	1	1
2	1	Z		Z	1	✓	1	1
3	1	X		X	1	✓	1	1
4	1	Z		Z	1	✓	1	1
5	0	Z		X	1			
6	0	X		Z	0			
7	0	Z		Z	0	✓	0	0
8	0	Z		X	0			
9	0	X		X	0	✓	0	0
10	0	X		Z	0			
11	0	X		X	0	✓	0	0
12	0	X		Z	0			
13	1	Z		Z	1	✓	1	1
14	0	Z		X	0			
15	1	X		Z	1			
16	1	X		Z	0			
17	1	X		X	1	✓	1	1
18	1	X		Z	1			
19	0	Z		X	0			
20	0	X		Z	1			

Alice and Bob then retain the bit values that they obtained when their bases choices agreed. They discard the rest. We see that they generate the same string of bits like this.

2 BB84 – no evesdropping

a) Alice generates a random string of key bits. Alice also generates a random string of choices of basis. Using a random number generator, produce both such choices for ten key bits and bases choices; record these. Write down the states of the qubits that Alice transmits to Bob.

b) Bob generates a random string of choices of basis; record these choices.

c) Bob measures successive qubits in the bases that he chose. When his basis choice matches that of Alice, write down bit value attained from the measurement outcome.

d) Compare the strings of bits that they generate when their bases choices agree. Do these match?

What could an evesdropper do? In this scheme Eve does not learn about Alice + Bob's basis choices until they have done their measuements and discarded their qubits.

Eve could

1) Intercept the qubit from Alice.

2) Randomly choose a basis, measure in this + retransmit the relevant state.

3) Only retain the qubits when Alice and Bob agree on their choices.

When all three of Eve, Alice and Bob make the same choice then Eve attains the same bit value as A and B without them detecting any interference

But what if Eve's choice is different to Alice + Bob's? We can analyze this for one case.

3 BB84 – with evesdropping

a) Suppose that Alice and Bob both choose to use the Z basis for a particular qubit. Suppose that Eve chooses the same basis. Will their bit values all match with certainty?

b) Suppose that Alice and Bob both choose to use the Z basis for a particular qubit. Suppose that Eve chooses the ~~same~~ X basis. Will their bit values all match with certainty? Analyze all possibilities.

	ALICE			EVE			BOB		AB match	Bit values	
Qubit	Alice key	Basis choice		Basis Choice	Outcome		Basis choice	Outcome		A	B
1	0	Z		X	1		X	1			
2	0	X		X	0		X	0	✓	0	0
3	1	X		X	1		Z	0			
4	0	X		Z	0		Z	0			
5	0	Z		Z	0		X	0			
6	0	Z		Z	0		Z	0	✓	0	0
7	1	X		Z	0		Z	0			
8	0	Z		Z	0		Z	0	✓	0	0
9	1	Z		X	1		Z	0	✓	1	0
10	1	X		X	1		Z	0			
11	1	X		Z	0		X	0	✓	1	0
12	0	Z		Z	0		X	0			
13	1	X		Z	0		Z	0			
14	0	X		X	0		Z	0			
15	0	Z		Z	0		X	0			
16	0	X		X	0		X	0	✓	0	0
17	0	X		Z	1		Z	1			
18	0	X		X	0		Z	0			
19	1	X		Z	1		X	1	✓	1	1
20	1	Z		X	0		X	0			

Note that they sometimes disagree