When Alice and Bob want to communicate a secret message (such as Bob’s online banking details) over an insecure channel (such as the internet), its essential to encrypt the message.

If Alice and Bob want to use Eve’s classical communication channel to share their key, it is impossible to tell if Eve has made a copy of this key for herself- they must place complete trust in Eve that she is not listening. If, however, Eve provides a quantum communication channel, Alice and Bob no longer need to trust Eve at all- they will know if she tries to read Bob’s message before it gets to Alice.

**Simple Process:**

- Step 1

Alice chooses a string of random bits, e.g.:

1000101011010100

And a random choice of basis for each bit:

ZZXZXXXZXZXXXXXX

Alice keeps these two pieces of information private to herself.

- Step 2

Alice then encodes each bit onto a string of qubits using the basis she chose, this means each qubit is in one of the states |0⟩

, |1⟩, |+⟩ or |−⟩

, chosen at random. In this case, the string of qubits would look like this:

|1⟩|0⟩|+⟩|0⟩|−⟩|+⟩|−⟩|0⟩|−⟩|1⟩|+⟩|−⟩|+⟩|−⟩|+⟩|+⟩

This is the message she sends to Bob.

- Step 3

Bob then measures each qubit at random, for example, he might use the bases:

XZZZXZXZXZXZZZXZ

And Bob keeps the measurement results private.

- Step 4

Bob and Alice then publicly share which basis they used for each qubit. If Bob measured a qubit in the same basis Alice prepared it in, they use this to form part of their shared secret key, otherwise they discard the information for that bit.

- Step 5

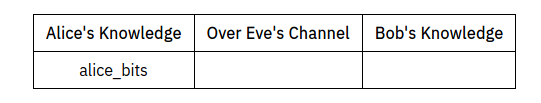
Finally, Bob and Alice share a random sample of their keys, and if the samples match, they can be sure (to a small margin of error) that their transmission is successful

## **Example: Without Interception**

### **Step 1:**

Alice generates her random set of bits.

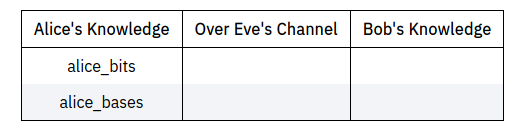
At the moment, the set of bits 'alice\_bits' is only known to Alice. We will keep track of what information is only known to Alice, what information is only known to Bob, and what has been sent over Eve's channel in a table like this:



### Step 2:

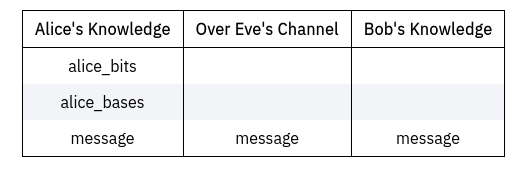
Alice chooses to encode each bit on qubit in the *X* or *Z*-basis at random, and stores the choice for each qubit in alice\_bases. In this case, a 0 means "prepare in the *Z*-basis", and a 1 means "prepare in the *X*-basis".

Alice also keeps this knowledge private.



Alice then creates a list of QuantumCircuits, each representing a single qubit in Alice's message.

This message of qubits is then sent to Bob over Eve's quantum channel:



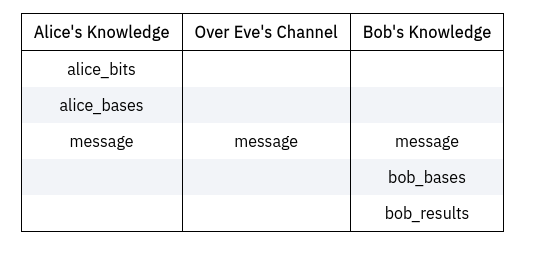
### Step 3:

Bob then measures each qubit in the *X* or *Z*-basis at random and stores this information:

Bob stores his choice for which basis he measures each qubit in.

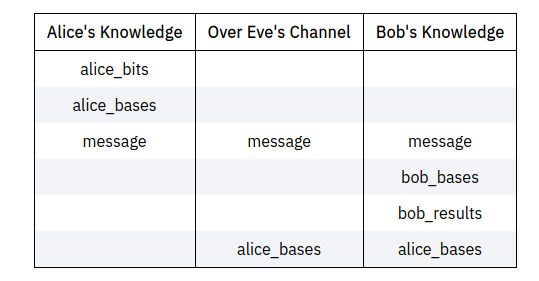
Bob then applies the corresponding measurement and simulates the result of measuring each qubit and stores it.

Bob keeps his results private.

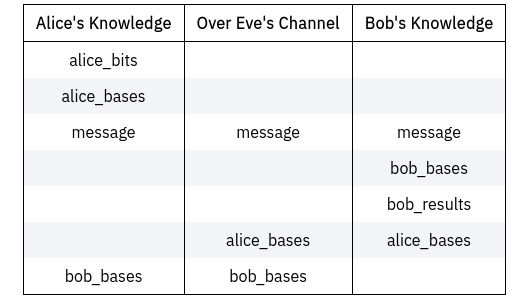


### Step 4:

After this, Alice reveals (through Eve's channel) which qubits were encoded in which basis:

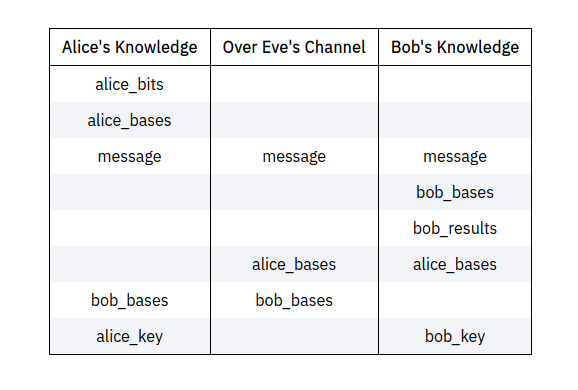


And Bob reveals which basis he measured each qubit in:



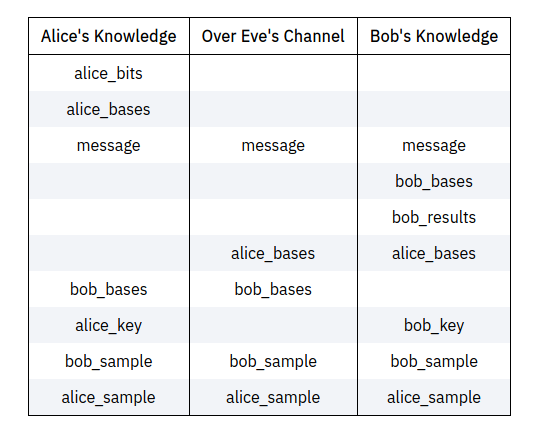
If Bob happened to measure a bit in the same basis Alice prepared it in, this means the entry in bob\_results will match the corresponding entry in alice\_bits, and they can use that bit as part of their key. If they measured in different bases, Bob's result is random, and they both throw that entry away.

Alice and Bob both discard the useless bits, and use the remaining bits to form their secret keys:



Finally, Bob and Alice compare a random selection of the bits in their keys to make sure the protocol has worked correctly.

Alice and Bob both broadcast these publicly, and remove them from their keys as they are no longer secret:



If the protocol has worked correctly without interference, their samples should match.

If their samples match, it means (with high probability) alice\_key == bob\_key. They now share a secret key they can use to encrypt their messages!

