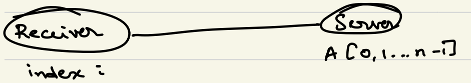
Oblivious Transfer (OT)



Q) Can receiver obtain A[i] without revealing i to the server whilst server doesnt reveal A[j], $j \neq i$ to the receiver.

Perfect solutions don't exist, server can see which A[i] it has sent to find i.

OT Protocol:

Step #1: Receiver sends a random array $R[0,1,\ldots n-1]$ where $R[i]$ is encrypted using servers public key. Others are not encrypted (where $R[j], j \neq i$)

Step #2: Server decrypts the entire array R[] to obtain $S[0,1\ldots n-1]$.

Step #3: Server sends $Z[] = A[] \oplus S[]$ to receiver

Step #4: Receiver obtains $A[i] = Z[i] \oplus R[i]$

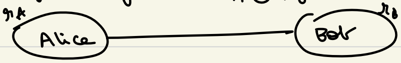
But, if receiver encrypts more indices, they can get more entries of A.

(Passive Adversary)
Semi-Honest Adversary Model
Adv. runs code delegated to them and otherwise they can do whatever they want

Active / Byzantine Adversary - Adv. can run whatever code it wants.

Semi Honest Adversary:

$$f(\vec{x},\vec{y}) = r_A \oplus r_B$$



$\vec{x} = x_1 x_2 \ldots x_n$ $\vec{y} : y_1 y_2 \ldots y_n$

$\vec{x}$ isnt revealed to Bob, $\vec{y}$ isnt revealed to Alice until output is revealed

A fixes r_A at random

$r_B = r_A \oplus f(\vec{x}, ?)$

$R = [\, . \, - - - \,]$ $OT(R, y)$

All possible values of y

$R[i] = r_A \oplus f(\vec{x}, i)$