Lecture 12

- More on adaptive security
- Towards CCA secure cryptosystems

\[ \text{Alice (PK)} \xrightarrow{c \leftarrow \text{Enc}(PK, m)} \text{Bob (PK, SK)} \xleftarrow{C_i \leftarrow \text{Enc}(PK, m_i)} \text{Eve (PK)} \]

**GM cryptosystem:**
\[
\begin{align*}
\text{Enc}(PK, 0) & : \quad c \leftarrow \mathbb{G}_m \\
\text{Enc}(PK, 1) & : \quad c \leftarrow \mathbb{G}_m 
\end{align*}
\]

\[
\text{Eve(PK, c)}: \text{sample random } x \text{ in } \mathbb{G}_m \text{ and } c' = x \cdot c
\]
send \( c' \) to Bob, receive its decryption \( b \)
output \( b \).

**General Idea:** use non-interactive zero-knowledge proofs of knowledge of the plaintext.

**NIZK**

Key Generation:
- set up \((PK', SK')\) for semantically secure cryptosystem
- set up params for NIZK, \( G \)

\[ PK = (PK', 0), SK = SK' \]

Encryption: \((PK, m)\):
\[
C \leftarrow \text{Enc}(PK, m)
\]
proof \( \Pi \in \text{NIZK} \)

Decryption:
verify \( \Pi \)
decrypt \( c \) to obtain \( m \)
Zero-knowledge

Prover
\((G, H)\)
- pick a random permutation \(\rho\)

\(G_\rho: \) matrix obtained from \(G\) by permuting all the vertices

\(H_\rho: \) corr. Ham. cycle

commitments \(\{c_{ij}\}\) to entries of \(G_\rho\)

If \(b=0\), reveal commitments
reveal perm. \(\rho\)
\(\rightarrow\) verify that commitments open to \(G_\rho\)

If \(b=1\), open commitments corresponding to 1's in \(H_\rho\)
\(\rightarrow\) verify that \(H_\rho\) is a cycle

Completeness: If Pr. and Ver. are following the protocol, then Ver. accepts.

Soundness: If prover can respond to both \(b=0\) and \(b=1\), then \(G\) is Ham.

Zero-knowledge: \(\text{Sim}(V^*, G, X^*)\): pick \(b' \in \{0, 1\}\)
- If \(b'=0\), follow P's protocol
- Else if \(b'=1\), pick random \(H_\rho\)
commit to its matrix description.
Run \(V^*\), send to \(V^*\) the commitments \(\{c_{ij}\}\)
\(V^*\) responds with \(b, b'=b\), respond to \(V^*\).
End try again, output \(V^*\)'s output.

A protocol \((P, V)\) is a zero-knowledge proof for a language \(\Pi\) if \(\exists \text{PPT } V^*\)
the following \(\text{Dreal}(X^*)\), \(\text{Dsim}(X^*)\) are indistinguishable. \(\forall x \in \Pi:\)

\(\text{Dreal}(X^*): \) Output of \(V^*\) when interacting with \(P(x, \omega)\)
\(\text{Dsim}(X^*): \) Output of \(\text{Sim}(X^*)\) \(\text{Sim}(V^*, X, \omega)\)
The simulator needs to run in poly-time. Simulator's output has to work correctly.

Let $b \neq b'$ w.prob $\frac{1}{2} + \epsilon(z)$ for $(G, H)$.

Consider:

$H_0(z^i)$: prover, $b' \leftarrow 0, 1^j$, $b = b'$ w.p. $\frac{1}{2}$

$H_0(z^i)$: simulator

Diagram:

1. pos $(i,j)$: has 0 in both $G_p, H_p$
2. pos $(i,j)$: has 1 in both $G_p, H_p$
3. pos $(i,j)$: has 1 in $G_p$, 0 in $H_p$

Suppose that there are $l$ cells of type 3.