CSC 591 Cryptography	September 10th, 2018

Lecture 3 – Security proofs, pseudo-OTP, PRG

Lecturer: Alessandra Scafuro — *Scribe: Tyler Irons*

Topic/Problem

We previously learned that pseudo-OTP is secure, now we are going to prove this through a Security Proof. This class we went over the Proof by Reduction method.

Intuition

Pseudo-OTP: We describe one time pad cipher using a pseudorandom generator. Using the PRG with an input of a random seed k, we generate a string of random bits that is longer than k, called r. Encryption of a message is done through taking the message and XOR-ing with r. Decryption is very similar, take the encrypted message c and XOR with r to get the message.

$Gen(1^\lambda)$	Picks a random seed $k \leftarrow \{0,1\}^n$.
$Enc(k,m)$	Computes $r = G(k)$ and outputs $c := m \oplus r$.
$Dec(k,c)$	Computes $r = G(k)$ and outputs $m := c \oplus r$.

We describe the pseudo-OTP scheme prove its security in more details below.

Scheme

We need to fix some message length l and let G be a PRG that has an expansion factor of l (so $|G(k)| = l(|k|)$), which is a polynomial. Let G be a PRG that has an expansion factor of l. We set n as the security parameter.

- **Gen:** on the input of 1^n, we choose a uniformly random PRG seed $k \in \{0,1\}^n$
  - Outputs the PRG seed k as the key.
  - Sets up the message space to be $\{0,1\}^{l(n)}$
- **Enc:** When given a key $k \in \{0,1\}^n$ and a message $m \in \{0,1\}^{l(n)}$ it outputs the ciphertext
  - $c := G(k) \oplus m$
- **Dec:** When given a key $k \in \{0,1\}^n$ and ciphertext $c \in \{0,1\}^{l(n)}$ it will output the message
  - $m := G(k) \oplus c$

To prove the security of this scheme. We prove the following theorem:

Theorem 1 *Assuming G is a pseudorandom generator, then the pseudo-OTP scheme above is a private-key encryption scheme that has indistinguishable encryptions in the presence of an eavesdropper.*

Security Proof

We prove it towards a contradiction. We do this by assuming that there exists a PPT adversary A, where $Pr[A\, wins\, Exp_{pseudo-OTP}(1^n)] \geq \frac{1}{2} + p(n)$ where $p(n)$ is assumed to be non-negligible. We consider OTP experiment in which the challenger uses one-time pad to encrypt message and pseudo-OTP experiment in which the challenger uses pseudorandom generator to encrypt message. Then we construct a reduction D, given a string y, using A as a black-box to distinguish whether y is computed using pseudorandom genrator or is truly random string.

1. OTP experiment

   Using real OTP, $Pr[A\, wins\, Exp_{OTP}(1^n)] = \frac{1}{2}$

$Exp_{OTP}(1^n)$	$\overleftarrow{m_0, m_1}$	
1) pick $r \leftarrow \{0,1\}^{l(n)}$		
	A	$\overrightarrow{b'}$ and if $(b = b')$, A wins
2) pick b at random		
3) output $c := r \oplus m_b$	$\overrightarrow{c}$	

- Adversary A provides $Exp_{OTP}(1^n)$ with the messages m_0, m_1 where the length of these two messages are equal.
- The challenger of $Exp_{OTP}(1^n)$ will generate a random string r of length $l(n)$, and pick a random $b \in \{0, 1\}$. Computes the challenge ciphertext $c = r \oplus m_b$ and then sent c to A.
- A will output a bit b'.
- if $b = b'$, then the experiment outputs 1 (i.e., A wins), otherwise outputs 0.

2. Pseudo-OTP experiment

$Exp_{pseudo-OTP}(1^n)$	$\overleftarrow{m_0, m_1}$	
1) pick $k \leftarrow \{0,1\}^n$		
	A	$\overrightarrow{b'}$ where $Pr[b = b'] \geq \frac{1}{2} + p(n)$
2) pick b at random		
3) output $c := G(k) \oplus m_b$	$\overrightarrow{c}$	

- Adversary A provides $Exp_{pseudo-OTP}(1^n)$ with the messages m_0, m_1 where the length of these two messages are equal.
- The challenger of $Exp_{pseudo-OTP}(1^n)$ will generate a PRG seed k and compute the the key $r = G(k)$, and pick a random bit $b \in \{0, 1\}$. Computes the challenge ciphertext $c = r \oplus m_b$ and then sent it to A.

- A will output a bit b'.
- if $b = b'$, then output 1 (A succeeded), otherwise output 0.

3. Reduction

$D(y)$	$\overleftarrow{m_0, m_1}$	
We play as D		
	A	$\overrightarrow{b'}$ and if $(b = b')$, output 1; else 0
pick b at random		
output $c := r \oplus m_b$	$\overrightarrow{c}$	

The reduction D is given an input $y \in \{0,1\}^{l(n)}$. The goal of D is to distinguish y is computed either using pseudorandom generator (i.e., $r = G(k)$) or it is a truly random string (i.e., r is a truly random string).

(a) Adversary A provides D with a pair of messages $m_0, m_1 \in \{0,1\}^{l(n)}$

(b) D will pick a random uniform bit $b \in \{0,1\}$ and will set $c := r \oplus m_b$

(c) D will give the challenge ciphertext c to A, and will take adversary A's output bit b'. D will then output 1 if $b' = b$, otherwise it will output 0.

4. Analysis

If y is truly random ($y \leftarrow U^{l(n)}$), then $Pr[D(y)_{y \leftarrow U^{l(n)}} = 1] = Pr[A wins Exp_{OTP}] = \frac{1}{2}$

If y is pseudo-random ($y \leftarrow G(k)$) then $Pr[D(y) = 1] = Pr[A\, wins\, Exp_{pseudo-OTP}] = \frac{1}{2} + p(n)$

Then we have advantage $|Pr[D(y)_{y \leftarrow U^{l(n)}} = 1] - Pr[D(y)_{y \leftarrow G(k), k \leftarrow \{0,1\}^n} = 1]| = |\frac{1}{2} - \frac{1}{2} - p(n)| = p(n)$

Because $p(n)$ is non-negligible, this contradicts to our assumption that G is a PRG.