EE 720: An Introduction to Number Theory and Cryptography (Spring 2020)

Lecture 6 — January 30, 2020

Instructor: Saravanan Vijayakumaran *Scribe: Saravanan Vijayakumaran*

1 Lecture Plan

- Define pseudorandom generators.
- See example of stream ciphers used in practice.
- Construct a fixed-length private-key encryption scheme that has indistinguishable encryptions in the presence of an eavesdropper.
- Prove the security of the above scheme assuming the existence of a pseudorandom generator.

2 Pseudorandom Generators

- Pseudorandomness is a property of a *distribution* on strings.
- Some desirable properties of a pseudorandom generator:
  - Any bit of the output should be equal to 1 with probability close to $\frac{1}{2}$.
  - The parity of any subset of the output bits should be equal to 1 with probability close to $\frac{1}{2}$.
- A good pseudorandom generator should pass all efficient statistical tests, i.e. for any efficient statistical test or *distinguisher* D, the probability that D returns 1 given the output of the pseudorandom generator should be close to the probability that D returns 1 when given a uniform string of the same length.

Definition. *Let l be a polynomial and let G be a deterministic polynomial-time algorithm such that for any n and $s \in \{0,1\}^n$, the result $G(s)$ is a string of length $l(n)$. We say that G is a* ***pseudorandom generator*** *if the following conditions hold:*

1. ***Expansion:*** *For every n it holds that $l(n) > n$.*
2. ***Pseudorandomness:*** *For any PPT algorithm D, there is a negligible function $\mathtt{negl}$ such that*
$$\left|\Pr\left[D\left(G(s)\right)=1\right]-\Pr\left[D(r)=1\right]\right| \leq \mathtt{negl}(n),$$
where the first probability is taken over uniform choice of $s \in \{0,1\}^n$ and the randomness of D, and the second probability is taken over uniform choice of $r \in \{0,1\}^{l(n)}$ and the randomness of D.

We call l the ***expansion factor*** *of G.*

- Example of a *non-pseudorandom generator*: Define $G : \{0,1\}^n \to \{0,1\}^{n+1}$ as $G(s) = s \| \left(\oplus_{i=1}^{n} s_i\right)$.
- What happens if remove the restriction that D is polynomial time?
- There is no known way to prove the unconditional existence of pseudorandom generators. We will see some constructions of stream ciphers which we hope are pseudorandom generators.

3 Stream Ciphers

- Stream ciphers are practical systems which behave like pseudorandom generators. However, there are no proofs available that a particular stream cipher is in fact a pseudorandom generator.
- Stream ciphers can be designed for either efficient hardware implementation or efficient software implementation.
- Hardware-oriented stream ciphers are based on feedback shift registers (FSRs).
- Linear feedback shift registers (LFSRs) are FSRs where the feedback function is linear.
- Example: Consider a four-bit shift register where the feedback is the XOR of all the four bits. If we initialize the state to 1100, then we get a cycle of period 5. The states are 1100, 1000, 0001, 0011, 0110.
- The output depends on the state of the LFSR. Once a state repeats, the output repeats. If an LFSR has n bits, then the period of the output sequence can be at most $2^n - 1$.
- Each LFSR can be associated with a feedback polynomial. If the feedback polynomial is primitive, then the period is maximal. A polynomial of degree n over GF(2) is primitive if it is irreducible and the smallest value of m for which the polynomial divides $X^m + 1$ is $m = 2^n - 1$. Example: $1 + X^3 + X^4$.

3.1 A5/1

- Used to provide voice encryption in the GSM cellular system.
- Developed in 1987. Reverse engineered in 1999.
- Uses three LFSRs of lengths 19, 22, and 23.
- More details at `https://en.wikipedia.org/wiki/A5/1`.

4 A Secure Fixed-Length Encryption Scheme

- Let G be a pseudorandom generator with expansion factor l. Define a private-key encryption scheme for messages of length l as follows:
  - `Gen`: On input 1^n, choose k uniformly from $\{0,1\}^n$.

- $\texttt{Enc}$: Given $k \in \{0,1\}^n$ and message $m \in \{0,1\}^{l(n)}$, output the ciphertext

$$c := G(k) \oplus m.$$

- $\texttt{Dec}$: Given $k \in \{0,1\}^n$ and ciphertext $c \in \{0,1\}^{l(n)}$, output the message

$$m := G(k) \oplus c.$$

Theorem. *If G is a pseudorandom generator, then the above construction is a fixed-length encryption scheme that has indistinguishable encryptions in the presence of an eavesdropper, i.e. for any PPT adversary $\mathcal{A}$ there is a negligible function $\texttt{negl}$ such that*

$$\Pr\left[PrivK^{eav}_{\mathcal{A},\Pi}(n) = 1\right] \leq \frac{1}{2} + \texttt{negl}(n).$$

Proof. Note that if a one-time pad is used instead of the pseudorandom generator $G(k)$, the system is EAV-secure. The key idea is that if a PPT adversary $\mathcal{A}$ can distinguish between the encryptions of m_0 and m_1, then it can distinguish between $G(k)$ and a uniformly random bitstring.

Distinguisher D: D is given a string $w \in \{0,1\}^{l(n)}$ (assume n can be determined from $l(n)$)

1. Run $\mathcal{A}(1^n)$ to obtain a pair of messages $m_0, m_1 \in \{0,1\}^{l(n)}$.
2. Choose a uniform bit $b \in \{0,1\}$. Set $c := w \oplus m_b$.
3. Give c to $\mathcal{A}$ and get b'. If $b = b'$ output 1 and output 0 otherwise.

If $\mathcal{A}$ succeeds, D decides that w is a pseudorandom string and if $\mathcal{A}$ fails D decides w is a random string.

Rest of proof done in class. □

5 References and Additional Reading

- Sections 3.2, 3.3 from Katz/Lindell