# Lecture Two

# Randomness

## 1. Introduction

The security of many cryptographic systems depends upon the generation of unpredictable quantities. Examples include the keystream in the one-time pad, the secret key in the DES encryption algorithm, the primes p, q in the RSA encryption and digital signature schemes, the private key a in the DSA. In all these cases, the quantities generated must be of sufficient size and be "random" in the sense that the probability of any particular value being selected must be sufficiently small to preclude an adversary from gaining advantage through optimizing a search strategy based on such probability. For example, the key space for DES has size $2^{56}$. If a secret key k were selected using a true random generator, an adversary would on average have to try $2^{55}$ possible keys before guessing the correct key k. If, on the other hand, a key k were selected by first choosing a 16-bit random secret, and then expanding it into a 56-bit key k using a complicated but publicly known function L, the adversary would on average only need to try $2^{15}$ possible keys (obtained by running every possible value for the sequence through the function $f$ ).

This lecture introduces basic concepts relevant to random and pseudorandom bit generation, considers a technique as a sample for pseudorandom bit generation, and describes statistical tests designed to measure the quality of a random bit generator.



## 2. Background

**Definition(2.1):** A **random bit generator** is a device or algorithm which outputs a sequence of statistically independent and unbiased binary digits.

**Remark(2.1):** (random bits vs. random numbers) A random bit generator can be used to generate (uniformly distributed) random numbers. For example, a random integer in the interval [0,n] can be obtained by generating a random bit sequence of length $\log_2^{|n+1|}$ bits, and converting it to an integer; if the resulting integer exceeds n, one option is to discard it and generate a new random bit sequence.

**Definition(2.2):** A **Pseudo Random Bit Generator** (PRBG) is a deterministic algorithm which, given a truly random binary sequence of length k, outputs a binary sequence of length L , k which "appears" to be random. The input to the PRBG is called the **seed**, while the output of the PRBG is called a **pseudorandom bit sequence**.

A minimum security requirement for a pseudorandom bit generator is that the length k of the random seed should be sufficiently large so that a search over $2^k$ elements (the total number of possible seeds) is infeasible for the adversary.