Πανεπιστημιο Πατρων, Τμημα Ηλεκτρολογων Μηχανικων και Τεχνολογιας Υπολογιστων

Title

Ηλίας Ουζούνης
up1083749

20 Δεκεμβρίου 2024

Περιεχομενα

1 Random Bit Generators

In cryptography it is essential to be able to generate random values. They are the most secure from attackers as any attempts to guess them are futile. From the passwords we all use to access our accounts to keys used to encrypt data, random values are needed everywhere to create robust and secure systems.

Unfortunately, humans are notoriously bad at creating random values and introduce a certain amount of bias [1]. This is why we had to rely on machines and algorisms to achieve high entropy on the values they generate. However, machines are completely deterministic and truly random values are impossible to create. To compensate for this, pseud-random number generators (PRNGs) were created. They work by utilizing a seed as the input to a deterministic function to generate a sequence of values that appear random. This function has to be chosen carefully to ensure the generated values are as close to random as possible and impossible to predict.

We can model the PRNG as a function f that maps an input from $\mathfrak{R}$ to a finite set of values S. [2]

$$f : \mathfrak{R} \to S \tag{1.1}$$
$$\tag{1.2}$$

The seed is the input to the function and the output is the generated value. The function f should be a one-way function [3] to prevent attackers from reconstructing the seed from the generated values. Additionally, there are other metrics to evaluate the suitability of the function f as a PRNG such as the period of the generated values and the correlation between them [2]. Nonetheless, PRNGs simply move the issue of creating random values to picking a random seed as the input to f, the main problem still remains.

The method of selecting a seed is equally important as the function f itself. Nowadays, most computer systems use the inherent randomness found in nature as the seeds to their PRNGs. Examples Include the time of day, system temperature, cache lookup times or even user ouse movements. These values are rich in entropy and are additionally passed through a hash function to ensure they are more evenly distributed [2]. Even temperature values that are simar to each other will be transformed into completely different seeds. However, generating seeds this way is slow and not suitable for application stat require a high amount of random values. To combat this issue, a seed is not only used to generate a random value as the output of f but is also transformed through a similar function into a new seed that is used for the next value of the sequence. Periodically, when the truly random values from nature are available, they overwrite the seed to start a new chain of pseudo-random values.

$$f : S_1 \to S_2 \tag{1.3}$$
$$g : S_1 \to S_1 \tag{1.4}$$
$$s_0 = \text{seed} \tag{1.5}$$
$$v_1 = f(s_0), s_1 = g(s_0)$$
$$v_2 = f(s_1), s_2 = g(s_1)$$
$$\vdots$$

Here, S_1 is the set of all random seeds and S_2 is the set of all generated values.

There are a plethora of well constructed PRNGs that are used in practice. The Linear Congruential Generator (LCG) and the Mersenne Twister are some of the most widely used PRNGs for general purposes. For cryptography though, some more specialized and more advanced algorisms have been developed. The focus of this report is on the Dual_EC_DRBG algorithm. An Elliptic Curve based algorithm that was proposed by the National Institute of Standards and Technology (NIST) in 2007 [?]. Before analyzing the algorism and the controversy surrounding it, we will first examine the the theory behind Elliptic Curves and how they are used in cryptography.

References

[1] Willem A. Wagenaar. Generation of random sequences by human subjects: A critical survey of literature. Psychological Bulletin, 77(1):65–72, 1972.

[2] Tom Kennedy. Math 577-002 - monte carlo methods, 2016. Available at https://math.arizona.edu/~tgk/mc/book_chap3.pdf.

[3] Wikipedia contributors. One-way function. https://en.wikipedia.org/wiki/One-way_function, 2024. Accessed: 2024-12-20.