Formalities

Preliminary Title:
Improved Security Through Randomness Testing

Name:
Joel Gärtner
Email:
jgartner@kth.se
Date:
25 januari 2017

CSC Supervisor:
Douglas Wikström
Principal:
Omegapoint
Principal Supervisor:
Hannes Salin

Background and Objective

Several cryptographic protocols require some input to be unpredictable in order to provide security. Furthermore, any keys used in the protocols must also be generated in an unpredictable manner since these are crucial for the level of security, thus it must be hard for any adversary to guess or compute such keys. In order to get a source of unpredictability, random generators are used. These generators can potentially be truly random and generate numbers based on some phenomenon which is deemed unpredictable. The speed of these generators is however often limited and because of this it is more common that the generators are pseudo random, i.e. based upon deterministic algorithms which given a seed for the generator, produces an arbitrarily long sequence which looks random. Depending on the application different types of pseudo random generators are needed. For simulations and similar the most important properties is that it is fast to generate numbers and that they behave statistically in a way that seems random. For cryptographic purposes the generators also have to be unpredictable in the meaning that an adversary who sees part of the generated sequence won't be able to efficiently guess the next bit of the sequence with a probability significantly higher than if the sequence was truly random.

These cryptographically secure pseudo random generators are thus such that they expand a seed into a sequence which cannot be efficiently distinguished from a truly random sequence without knowledge of the seed. If the seed is known however, the whole sequence is easily reproducible as the algorithm generating the sequence is deterministic. As such that the pseudo random generator is cryptographically secure will not provide any security if the seed is predictable. Therefore it is of great importance for a process to select a seed of unpredictable bits which can then be expanded into arbitrarily much seemingly random data using a cryptographically secure pseudo random generator. In order to select this seed there is a requirement for a

source of unpredictability. This can for example be an actual true random generator or another process on the generating machine which behaves in an unpredictable manner. It is however not obvious how much data is needed from the source of unpredictability before enough unpredictability has been gathered to the seed. A measure of this unpredictability is the entropy of the sample received from the source. Entropy is a concept taken from thermodynamics which was first used in information theory by Shannon with Shannon entropy [7]. A sequence of bits from a source will have an entropy of 1 per bit if it is completely random and an entropy of 0 per bit if it is completely predictable. It is easy to estimate the entropy of data if the data consists of samples with n bit where each such sample is generated completely independent from any of the other samples. If this is the case then the unpredictability and thus the entropy only depends on the probabilities of the samples x and can be calculated by estimating the probabilities of the sample x with the relative frequency of x. In case the samples are not independent identically distributed then it is a lot harder to estimate the entropy.

Several test suites exists which try to determine when data does not behave randomly [6] [4]. These suites can to some extent identify sequences where the samples do not behave like identically distributed random variables. Most such tests do however only identify when there is incorrect behaviour in the sequence and does not give an estimate of the actual unpredictability of the sequence. As the sequence can somehow be distinguished from truly random data, it could be a potential weakness if used in cryptographic applications by itself. The sequence does however probably still contain some amount of unpredictability and this could potentially be extracted by transforming the sequence so that it statistically behaves more like random data. This can be done with multiple sources of entropy which can be combined in order to create a source of random data which behaves randomly and is unpredictable. This source could then be used to get a seed for a cryptographically secure random generator which could expand this source of unpredictable into as much random data as necessary.

An estimate of the entropy of data is thus useful in order to determine how much data is necessary to get enough unpredictability for your random generators. A recent approach to estimating the entropy of data is to use predictors. These estimators take the approach of trying to predict the data which is fed into them as good as possible. The entropy of the data can then be approximated via the success rate of the predictors. Several predictors were constructed in order to give good predictions and thus good estimations of the entropy. All entropy estimations will however have the problem that they only estimate the entropy of some specific types of distributions of data while other distributions will get wrong estimations. The constructed predictors were meant to be general and work in general but more specific

estimators will work better for most types of entropy sources. As such there is the possibility of a lot more entropy sources which better models more types of data sources.

Research Questions and Method

Evaluation and News Value

Pilot Study

Some study of the implementations of pseudo random number generators and cryptographically secure pseudo random number generators to get a better understanding of the context. Furthermore study of the Yarrow and Fortuna algorithms and their implementations may be of interest as these use entropy accumulators as integral parts of their algorithm to secure the algorithm even in cases where part of the state becomes known to an attacker.

There has also been guidelines produced by NIST related to random number generators and entropy sources. These documents NIST SP 800-90(A,B,C) [2] [3] [1] are guidelines related to how secure random generators should perform and how sources of entropy sources should behave and how they can be tested. Furthermore researchers at NIST have also produced the paper *Predictive Models for Min-entropy Estimation* [5] related to using predictors to producing better estimates for entropy. Other sources related to how entropy and randomness testing is performed is also of interest, including for example how suites such as TestU01 evaluates data.

Furthermore it will need to be investigated what sources of entropy exist today and how they work. This includes built in entropy generators in operating systems but also specific generators built for that specific purpose. Sites such as *random.org* which provide a service which provide random data as a service will also be of interest.

It may also be useful to study the different cryptographic protocols where randomness is useful in order to see what kind of impact too low entropy may have in relation to security. In combination with this it will also be of interest to investigate the weaknesses which have been discovered that were the result of lacking entropy when generating random numbers for cryptographic protocols.

Conditions and Schedule

Litteraturförteckning

[1] E Barker and J Kelsey. Recommendation for random bit generator (rbg) constructions (draft nist special publication 800-90c). *National Institute of Standards and Technology*, 2012.

[2] Elaine Barker and John Kelsey. Recommendation for random number generation using deterministic random bit generators (revised). *NIST Special Publication*, pages 800–90, 2011.

[3] Elaine B Barker and John Michael Kelsey. *(Second DRAFT) Recommendation for the Entropy Sources Used for Random Bit Generator*. US Department of Commerce, National Institute of Standards and Technology, 2016.

[4] Lawrence E. Bassham, III, Andrew L. Rukhin, Juan Soto, James R. Nechvatal, Miles E. Smid, Elaine B. Barker, Stefan D. Leigh, Mark Levenson, Mark Vangel, David L. Banks, Nathanael Alan Heckert, James F. Dray, and San Vo. Sp 800-22 rev. 1a. a statistical test suite for random and pseudorandom number generators for cryptographic applications. Technical report, Gaithersburg, MD, United States, 2010.

[5] John Kelsey, Kerry A. McKay, and Meltem Sönmez Turan. Predictive models for min-entropy estimation. *IACR Cryptology ePrint Archive*, 2015:600, 2015.

[6] Pierre L'Ecuyer and Richard Simard. Testu01: A c library for empirical testing of random number generators. *ACM Trans. Math. Softw.*, 33(4):22:1–22:40, August 2007.

[7] C. E. Shannon. A mathematical theory of communication. *SIGMOBILE Mob. Comput. Commun. Rev.*, 5(1):3–55, January 2001.