Barak-Halevi pseudo-random number generator model extended with applications to `/dev/random`

Sylvain Ruhault (Ecole Normale Supérieure and Oppida)*

Generating random numbers is an essential task in cryptography. Random numbers are necessary not only for cryptographic keys, but are also often needed in steps of cryptographic algorithms or protocols. When needed, random numbers are mainly generated by a pseudo-random number generator, that can be an internal component of a cryptographic library or part of the operating system on which the library is installed. In this work, we focus on the Linux operating system pseudo-random number generator `/dev/random`.

A pseudo-random generator is an algorithm that expands short seeds into longer sequences, whose distribution is indistinguishable from uniform. It produces sequences that ”look random” and are suitable for applications that use them. A formal model for a robust pseudo-random number generator was proposed in 2005 by Barak and Halevi [1]. This model involves an internal state that is refreshed with a potentially biased external random source and a cryptographic function that outputs uniform random numbers from the internal state. It offers *resilience, forward security* and *backward security* against an attacker interacting with the generator for output and influencing the data used to refresh the internal state. We extend this model with an additional expected property of *entropy preservation*: the generator preserves computationally the entropy of the internal state after the output and refresh operations. In other words, the entropy of the internal state is computationally guaranteed not to decrease.

The Linux pseudo-random number generator `/dev/random` is part of the kernel and is used in operating system security services or transparently in cryptographic libraries that rely on this generator when calls to their internal random number generator are made. A first analysis of the generator was given in 2006 by Gutterman, Pinkas and Reinman [2], which was completed in 2012 by Lacharme, Röck, Strubel and Videau [3]. It involves a complex combination of cryptographic and non cryptographic functions, for which no formal security arguments or proof has been stated, nor its conformance with any security model. In particular, the random numbers extraction function involves a refresh of the internal state with a cryptographic transform of the extracted numbers that has not been previously formally assessed. We prove that the Linux pseudo-random number generator possesses the *entropy preservation* property and use this property to give computational results on this generator. In addition, we propose a new simple architecture for a pseudo-random generator.

* Join work with Yevgeniy Dodis (New York University), David Pointcheval (Ecole Normale Supérieure) and Damien Vergnaud (Ecole Normale Supérieure).

References

[1] B. Barak and S. Halevi, “A model and architecture for pseudo-random generation with applications to /dev/random,” in *ACM Conference on Computer and Communications Security*, pp. 203–212, 2005.

[2] Z. Gutterman, B. Pinkas, and T. Reinman, “Analysis of the linux random number generator,” in *IEEE Symposium on Security and Privacy*, pp. 371–385, 2006.

[3] P. Lacharme, A. Röck, V. Strubel, and M. Videau, “The linux pseudorandom number generator revisited.” Cryptology ePrint Archive, Report 2012/251, 2012.