A short proof of the unpredictability of cipher block chaining

Daniel J. Bernstein *

Department of Mathematics, Statistics, and Computer Science (M/C 249)
The University of Illinois at Chicago
Chicago, IL 60607–7045
djb@cr.yp.to

Abstract. Let u be a uniform random function from b-bit strings to b-bit strings. Fix $m \geq 1$. Define

$$u_m^+(g_1, g_2, \ldots, g_m) = u(u(\cdots u(u(g_1) + g_2) + \cdots) + g_m).$$

This paper presents a short proof that u_m^+ is unpredictable: specifically, if A is an algorithm that performs at most q oracle queries, and v is a uniform random function from mb-bit strings to b-bit strings, then the A-distance from u_m^+ to v is at most $mq(mq-1)/2^{b+1}$. It was already known that u_m^+ was unpredictable, but previous proofs were much more complicated.

Keywords: mode of operation, CBC, provable security

1 Introduction

Let u be a uniform random function from b-bit strings to b-bit strings; in other words, let $u(0), u(1), u(2), \ldots, u(2^b - 1)$ be independent uniform random b-bit strings. Define

$$u^+(g_1, g_2, \ldots, g_m) = u_m^+(g_1, g_2, \ldots, g_m) = u(u(\cdots u(u(g_1) + g_2) + \cdots) + g_m)$$

for each integer $m \geq 0$ and each mb-bit string $(g_1, g_2, \ldots, g_m)$. For example, $u^+() = u_0^+() = 0$, and $u^+(g_1, g_2) = u_2^+(g_1, g_2) = u(u(g_1) + g_2)$.

This paper presents a short proof that u_m^+ is unpredictable for $m \geq 1$—i.e., u_m^+ is indistinguishable from a uniform random function from mb-bit strings to

* The author was supported by the National Science Foundation under grant CCR–9983950, and by the Alfred P. Sloan Foundation. Date of this document: 2005.01.09. Permanent ID of this document: 24120a1f8b92722b5e15fbb6a86521a0. This is a preliminary version meant to announce ideas; it will be replaced by a final version meant to record the ideas for posterity. There may be big changes before the final version. Future readers should not be forced to look at preliminary versions, unless they want to check historical credits; if you cite a preliminary version, please repeat all ideas that you are using from it, so that the reader can skip it.

b-bit strings. More precisely, if A is an algorithm that performs at most q oracle queries, and v is a uniform random function from mb-bit strings to b-bit strings, then the A-distance from u_m^+ to v is at most $mq(mq-1)/2^{b+1}$. Here the **A-distance from u_m^+ to v** is $|\Pr[A(u_m^+) = 1] - \Pr[A(v) = 1]|$, where $A(f)$ means the output of A using an oracle for f.

The heart of the proof—see Section 2—is that u_m^+ has large interpolation probabilities: if $x_1, x_2, \ldots, x_k$ are distinct mb-bit strings, and $y_1, y_2, \ldots, y_k$ are b-bit strings, then $(u_m^+(x_1), u_m^+(x_2), \ldots, u_m^+(x_k)) = (y_1, y_2, \ldots, y_k)$ with probability at least $(1-\epsilon)/2^{bk}$ where $\epsilon = mk(mk-1)/2^{b+1}$. The rest of the proof—see Section 3—is a broad principle having nothing to do with the details of u_m^+: any random function with large interpolation probabilities is unpredictable. Section 4 discusses a few standard consequences of the unpredictability of u_m^+.

History

The construction of u_m^+ is called "cipher block chaining." The unpredictability of CBC is not a new result: Bellare, Kilian, and Rogaway proved in [2, Theorem 3.1] that the q-query distance from u_m^+ to v is at most $3m^2q^2/2^{b+1}$. Their proof is vastly more complicated than the proof here.

In reaction to a draft of [2], I wrote [3], explaining a much simpler way to prove unpredictability. [3, Theorem 3.1] is the same as Theorem 3.1 in this paper. I illustrated the theorem with a construction different from CBC, but commented at the end of [3, Section 5] that the theorem would also allow an easy proof of unpredictability for CBC. This paper presents that proof.

A subsequent Bellare-Rogaway preprint "The Game-Playing Technique," now at Draft 0.4 after the correction of some serious errors, presents (among other things) another proof of unpredictability for CBC. The authors describe their proof as "elementary"; I agree that it is an improvement over the proof in [2], but it is still much more complicated than necessary.

Bellare and Rogaway say that their approach "can lead to more easily verified, less error-prone proofs than those grounded in more conventional probabilistic language." I see no justification for that claim. I see many cryptographic proofs that are unnecessarily complicated because the authors simply don't *know* the standard language of probability theory,[1] let alone how to competently use it;[2] but the obvious solution is to educate people, not to reinvent the wheel.

[1] There's much more to the language than the simplified concepts of "event" (subset of a finite universe) and "probability" (subset size divided by universe size) that we teach to undergraduates. Most importantly, the concept of a "random variable" has had a standard mathematical definition for seventy years and is a tremendous time-saver in probabilistic definitions, theorems, and proofs. Warning to undergraduates: "random" does not imply "uniform" or "discrete" or "independent of everything else." For definitions see, e.g., [3].

[2] For example, many cryptographers appear to believe that figuring out the success probability of a protocol requires separately analyzing the success probability of the first step, the conditional success probability of the second step, etc. See, e.g., the CBC proofs in [2] and [7].

2 CBC has large interpolation probabilities

Theorem 2.1. *Let G be a finite commutative group. Let u be a uniform random function from G to G. Define $u^+(g_1, \ldots, g_i) = u(u(\cdots u(g_1) + \cdots) + g_i)$ for all $(g_1, \ldots, g_i) \in G^0 \cup G^1 \cup G^2 \cup \cdots$. Let m and k be integers with $m \geq 1$ and $k \geq 0$. Let $x_1, x_2, \ldots, x_k$ be distinct elements of G^m. Let $y_1, y_2, \ldots, y_k$ be elements of G. Then $(u^+(x_1), u^+(x_2), \ldots, u^+(x_k)) = (y_1, y_2, \ldots, y_k)$ with probability at least $(1 - \epsilon)/\#G^k$ where $\epsilon = mk(mk - 1)/2\#G$.*

In other words, every k-interpolation probability of u_m^+ is at least $(1-\epsilon)/\#G^k$.

Proof. Define $S = \{x_1, x_2, \ldots, x_k\}$. Define $P \subseteq G^1 \cup \cdots \cup G^m$ as the set of nonempty prefixes of $x_1, x_2, \ldots, x_k$. Note that $\#P \leq mk$.

Each element of P can be written uniquely as (q, g) with $g \in G$ and $q \in G^0 \cup P$. Define $\operatorname{chop}(q, g) = q$ and $\operatorname{last}(q, g) = g$.

Define a function $f : G^0 \cup P \to G$ as **admissible** if $f() = 0$, $f(x_i) = y_i$ for all i, and the function $p \mapsto f(\operatorname{chop} p) + \operatorname{last} p$ from P to G is injective. Define f as being **compatible with** u if $u(f(\operatorname{chop} p) + \operatorname{last} p) = f(p)$ for every $p \in P$.

Observe that each admissible function f has probability $1/\#G^{\#P}$ of being compatible with u. (Proof: $p \mapsto u(f(\operatorname{chop} p)+\operatorname{last} p)$ is a uniform random function from P to G, so it has probability $1/\#G^{\#P}$ of matching f.) Furthermore, if an admissible function f is compatible with u, then $(u^+(x_1), u^+(x_2), \ldots, u^+(x_k)) = (y_1, y_2, \ldots, y_k)$; in fact, $u^+(p) = f(p)$ for every $p \in G^0 \cup P$. (Proof: $u^+() = 0 = f()$. For $p \in P$, assume inductively that $u^+(\operatorname{chop} p) = f(\operatorname{chop} p)$. Then $u^+(p) = u(u^+(\operatorname{chop} p) + \operatorname{last} p) = u(f(\operatorname{chop} p) + \operatorname{last} p) = f(p)$.)

If two different admissible functions f, f' are compatible with u then $f(p) = u^+(p) = f'(p)$ for every $p \in G^0 \cup P$, contradiction. I will show in a moment that there are at least $(1 - \epsilon)\#G^{\#P-k}$ admissible functions f. Therefore, with probability at least $(1 - \epsilon)\#G^{-k}$, some admissible function f is compatible with u, and in particular $(u^+(x_1), u^+(x_2), \ldots, u^+(x_k)) = (y_1, y_2, \ldots, y_k)$ as claimed.

To count admissible functions, consider a uniform random function $f : G^0 \cup P \to G$. Each of the conditions $f() = 0$, $f(x_1) = y_1$, $\ldots$, $f(x_k) = y_k$ is satisfied with probability $1/\#G$. These conditions are independent, since $x_1, \ldots, x_k$ are distinct and $m \geq 1$; thus f satisfies all the conditions with probability $\#G^{-1-k}$.

If p, p' are distinct elements of P then $f(\operatorname{chop} p) + \operatorname{last} p = f(\operatorname{chop} p') + \operatorname{last} p'$ with conditional probability at most $1/\#G$. (If $\operatorname{chop} p = \operatorname{chop} p'$ and $\operatorname{last} p = \operatorname{last} p'$ then $p = p'$, contradiction. If $\operatorname{chop} p = \operatorname{chop} p'$ and $\operatorname{last} p \neq \operatorname{last} p'$ then $f(\operatorname{chop} p) + \operatorname{last} p$ cannot equal $f(\operatorname{chop} p') + \operatorname{last} p'$. If $\operatorname{chop} p \neq \operatorname{chop} p'$ then at least one of $\operatorname{chop} p, \operatorname{chop} p'$, let's say $\operatorname{chop} p$, is distinct from $()$; thus $f(\operatorname{chop} p)$ is conditionally uniform, so it equals $f(\operatorname{chop} p') + \operatorname{last} p' - \operatorname{last} p$ with probability $1/\#G$. Note that requiring G to be a commutative group is overkill here.)

Hence the conditional probability of any collisions in $p \mapsto f(\operatorname{chop} p) + \operatorname{last} p$ is at most $\#P(\#P - 1)/2\#G \leq \epsilon$; i.e., f is admissible with probability at least $(1-\epsilon)\#G^{-1-k}$; i.e., there are at least $(1-\epsilon)\#G^{-1-k}\#G^{\#P+1} = (1-\epsilon)\#G^{\#P-k}$ admissible functions f. $\square$

Example

Say $G = \mathbf{Z}/10^6$, $m = 3$, $k = 3$, $x_1 = (1, 2, 3)$, $x_2 = (1, 2, 4)$, and $x_3 = (3, 1, 4)$. Then $S = \{(1, 2, 3), (1, 2, 4), (3, 1, 4)\}$ and

$$P = \{(1), (3), (1, 2), (3, 1), (1, 2, 3), (1, 2, 4), (3, 1, 4)\}.$$

There are at most $mk = 9$ elements of P: in fact, only 7, since $(1, 2, 3)$ and $(1, 2, 4)$ share some prefixes.

A function $f : G^0 \cup P \to G$ is admissible if and only if $f() = 0$, $f(1, 2, 3) = y_1$, $f(1, 2, 4) = y_2$, $f(3, 1, 4) = y_3$, and the seven quantities

$$f() + 1, f() + 3, f(1) + 2, f(3) + 1, f(1, 2) + 3, f(1, 2) + 4, f(3, 1) + 4$$

are distinct. There are $\#G^4$ functions satisfying the equations (i.e., $\#G^4$ choices of $f(1), f(3), f(1, 2), f(3, 1)$), and there are $7(7 - 1)/2 = 21$ inequalities each eliminating at most $\#G^3$ functions, so there are at least $\#G^4 - 21\#G^3$ admissible functions.

An admissible function f is compatible with u if and only if $u(f()+1) = f(1)$, $u(f() + 3) = f(3)$, $u(f(1) + 2) = f(1, 2)$, $u(f(3) + 1) = f(3, 1)$, $u(f(1, 2) + 3) = f(1, 2, 3)$, $u(f(1, 2) + 4) = f(1, 2, 4)$, and $u(f(3, 1) + 4) = f(3, 1, 4)$, This occurs with probability exactly $1/\#G^7$ for each f, and if it does occur then $u^+(1, 2, 3) = y_1$, $u^+(1, 2, 4) = y_2$, $u^+(3, 1, 4) = y_3$. It cannot occur for two f's simultaneously, so it occurs with probability at least $(\#G^4 - 21\#G^3)/\#G^7 = (1 - 21/\#G)/\#G^3$.

3 Large interpolation probabilities imply unpredictability

Theorem 3.1. *Let φ be a random function from a set S to a finite set T. Let q be an integer with $q \ge 0$. Let A be an algorithm that performs at most q distinct oracle queries. Assume, for all $k \in \{0, 1, 2, \ldots, q\}$, all $y_1, y_2, \ldots, y_k \in T$, and all distinct $x_1, x_2, \ldots, x_k \in S$, that $(\varphi(x_1), \varphi(x_2), \ldots, \varphi(x_k)) = (y_1, y_2, \ldots, y_k)$ with probability at least $(1 - \epsilon)/\#T^k$. Then the A-distance between φ and uniform is at most ϵ.*

In other words, if every k-interpolation probability of φ is at least $(1-\epsilon)/\#T^k$ for all $k \in \{0, 1, 2, \ldots, q\}$, then φ cannot be predicted with probability larger than ϵ by an algorithm that performs at most q oracle queries. Note that this is an information-theoretic statement: the run time of the algorithm is irrelevant.

Theorem 3.1 appears in my paper [3]. I have included a (slightly shorter) proof here for completeness.

Proof. For each $k \in \{0, 1, 2, \ldots, q\}$, each $y = (y_1, y_2, \ldots, y_k) \in T^k$, and each $x = (x_1, x_2, \ldots, x_k) \in S^k$ with $x_1, x_2, \ldots, x_k$ distinct, first define $\alpha(x, y)$ as the conditional probability that A's distinct oracle queries are exactly $x_1, x_2, \ldots, x_k$ and A's output is 1, given that the oracle responses are $y_1, y_2, \ldots, y_k$.

In other words, $\alpha(x, y)$ is the chance that A decides to issue oracle query x_1, then—given response y_1—to issue oracle query x_2, and so on.

Next define $\beta_f(x, y)$ as the probability that $(f(x_1), \ldots, f(x_k)) = (y_1, \ldots, y_k)$. Then $\alpha(x, y)\beta_f(x, y)$ is the probability that, when A uses f as an oracle, its distinct oracle queries are $x_1, x_2, \ldots, x_k$, the oracle responses are $y_1, y_2, \ldots, y_k$, and A's output is 1. Sum over all x, y to obtain the overall probability that A prints 1: namely, $\Pr[A(f) = 1] = \sum_{x,y} \alpha(x, y)\beta_f(x, y)$.

By hypothesis $\beta_\varphi(x, y) \geq (1 - \epsilon)/\#T^k = (1 - \epsilon)\beta_v(x, y)$ where v is a uniform random function from S to T. Hence $\Pr[A(\varphi) = 1] = \sum_{x,y} \alpha(x, y)\beta_\varphi(x, y) \geq (1 - \epsilon) \sum_{x,y} \alpha(x, y)\beta_v(x, y) = (1 - \epsilon) \Pr[A(v) = 1] \geq \Pr[A(v) = 1] - \epsilon$. Similarly $\Pr[A(\varphi) \neq 1] \geq \Pr[A(v) \neq 1] - \epsilon$. Thus the A-distance between φ and v is at most ϵ. □

Theorem 3.2. *Let m and q be integers with $m \geq 1$ and $q \geq 0$. Let G be a finite commutative group. Let u be a uniform random function from G to G. Define*

$$u_m^+(g_1, g_2, \ldots, g_m) = u(u(\cdots u(u(g_1) + g_2) + \cdots) + g_m)$$

for all $(g_1, g_2, \ldots, g_m) \in G^m$. Let A be an algorithm that performs at most q distinct oracle queries. Then the A-distance between u_m^+ and uniform is at most $mq(mq - 1)/2\#G$.

Proof. If $k \in \{0, 1, \ldots, q\}$ then $(u_m^+(x_1), u_m^+(x_2), \ldots, u_m^+(x_k)) = (y_1, y_2, \ldots, y_k)$ with probability at least $(1 - mq(mq - 1)/2\#G)/\#G^k$ by Theorem 2.1. Apply Theorem 3.1. □

4 Standard consequences

From uniform to unpredictable

Say f is a uniform random permutation of the set of b-bit strings. It is difficult to distinguish f from u, so it is difficult to distinguish f_m^+ from u_m^+. More precisely, the q-query distance from f_m^+ to u_m^+ is at most the mq-query distance from f to u, which is at most $mq(mq - 1)/2^{b+1}$. Hence f_m^+ is unpredictable: the q-query distance from f_m^+ to uniform is at most $mq(mq - 1)/2^b$.

More generally, if f is a random function from b-bit strings to b-bit strings, and if f is unpredictable to all fast algorithms, then f_m^+ is unpredictable to all fast algorithms. For example, if k is a uniform random 128-bit string, then the random function AES_k from 128-bit strings to 128-bit strings is conjectured to be unpredictable, so the random function $(\mathrm{AES}_k)_{20}^+$ from 2560-bit strings to 128-bit strings is also conjectured to be unpredictable.

Message authentication

One way to securely authenticate a message t is to transmit it as $(t, v(t))$, where v is a secret uniform random function shared by the sender and receiver. This protocol remains secure when v is replaced with any unpredictable random function—in particular, u_m^+, or more generally f_m^+ when f is unpredictable.

Beware that it is not a good idea to use CBC to authenticate messages in practice:

- Old reason: f_m^+ takes inputs of a fixed positive length, namely mb bits, whereas most applications send variable-length messages. Switching from f_m^+ to f^+ is not safe: observe that $f^+() = 0$, for example, and $f^+(0) = f^+(0, -f^+(0))$. On the other hand, minor variants of f^+ are unpredictable.
- New reason: Other message-authentication codes are much faster and provide much stronger security guarantees. See, e.g., [4].

CBC nevertheless remains—thanks to its simplicity—an interesting test case for security-proof methodologies.

References

1. Mihir Bellare, Joe Kilian, Phillip Rogaway, *The security of cipher block chaining*, in [5] (1994), 341–358; see also newer version [2].
2. Mihir Bellare, Joe Kilian, Phillip Rogaway, *The security of the cipher block chaining message authentication code*, Journal of Computer and System Sciences **61** (2000), 362–399; see also older version [1]. ISSN 0022–0000. URL: `http://www-cse.ucsd.edu/~mihir/papers/cbc.html`.
3. Daniel J. Bernstein, *How to stretch random functions: the security of protected counter sums*, Journal of Cryptology **12** (1999), 185–192. ISSN 0933–2790. URL: `http://cr.yp.to/papers.html`.
4. Daniel J. Bernstein, *The Poly1305-AES message-authentication code*. URL: `http://cr.yp.to/papers.html#poly1305`. ID `0018d9551b5546d97c340e0dd8cb5750`.
5. Yvo Desmedt (editor), *Advances in cryptology—CRYPTO '94*, Lecture Notes in Computer Science, 839, Springer-Verlag, Berlin, 1994.
6. Lars Knudsen (editor), *Advances in cryptology—EUROCRYPT 2002: proceedings of the 21st International Annual Conference on the Theory and Applications of Cryptographic Techniques held in Amsterdam, April 28–May 2, 2002*, Lecture Notes in Computer Science, 2332, Springer-Verlag, Berlin, 2002. ISBN 3–540–43553–0.
7. Ueli Maurer, *Indistinguishability of random systems*, in [6] (2002), 110–133.