The Dual Elliptic Curve Deterministic RBG

Background, Specification, Security and Notes

Joshua E. Hill

Department of Mathematics, University of California, Irvine

Math 235C Mathematical Cryptography
June 5, 2013
`http://bit.ly/DECDRBG`

v1.1

Talk Outline

Introduction Outline

Introduction

- We'll talk about Number Theoretic RBGs.
- We'll talk about the desired goals of any reasonable RBG.
- We'll provide a specification for the Dual Elliptic Curve Deterministic RBG.
- We'll discuss the relevant problems.
- We'll describe some attacks on this DRBG

Background

I mean, what's the point?

- There are many quick, well-designed RBGs in the world.
- They are generally based on ad-hoc assumptions and their security is dependent on some underlying security primitive.
- We would ideally have some RBG that was as secure as some very difficult problem.
- The "I'd have bigger problems" design ideal.
- Such algorithms do exist!

That's HARDCORE!

Definition

A hardcore bit (also called "hardcore predicate") is a single bit associated with a one way function. Guessing this bit with any significant advantage is equivalent to reversing the associated one-way function.

BBS my Heart!

- We have already discussed one such RBG whose security analysis uses this notion: The Blum-Blum-Shub RBG.

Definition

Seed the RBG with $2 < x_0 < n - 1$ such that $(x_0, n) = 1$. Future states are calculated as $x_j = x_{j-1}^2 \pmod{n}$. The jth output, r_j, is a hardcore bit, generally the parity of x_j.

So, “Presentation Accomplished”?

- One bit per modular squaring is not exactly quick...
- Security bounds are a killer...
  - 128 bit security requires a 3072 bit modulus.
  - 256 bit security requires a 15360 bit modulus.
- If the modulus is k bits long, these multiplications each take at least $O(k \log k \log \log k)$.

So, "Presentation Accomplished"?

- One bit per modular squaring is not exactly quick…
- Security bounds are a killer…
  - 128 bit security requires a 3072 bit modulus.
  - 256 bit security requires a 15360 bit modulus.
- If the modulus is k bits long, these multiplications each take at least $O(k \log k \log \log k)$.
- … per bit output…

Blum-Micali Random Bit Generator

A more closely related design to the deterministic RBG that we are looking at today is:

Definition

The Blum-Micali Number Generator is specified by a (large) prime p, a generator g of multiplicative order $p-1$ and an initial value x_0. The jth value is then $x_j = g^{x_{j-1}} \pmod{p}$. The jth output bit, r_j, is 1 if $x_j < \frac{p-1}{2}$ and 0 otherwise.

- Surely no performance problem here![1]
- If the modulus is k bits long, modular exponentiation occurs in $O(k^2 \log k \log\log k)$.

[1]This bullet point is intended as sarcasm.

Blum-Micali Random Bit Generator

A more closely related design to the deterministic RBG that we are looking at today is:

Definition

The Blum-Micali Number Generator is specified by a (large) prime p, a generator g of multiplicative order $p-1$ and an initial value x_0. The jth value is then $x_j = g^{x_{j-1}} \pmod{p}$. The jth output bit, r_j, is 1 if $x_j < \frac{p-1}{2}$ and 0 otherwise.

- Surely no performance problem here![1]
- If the modulus is k bits long, modular exponentiation occurs in $O(k^2 \log k \log\log k)$.
- … per bit output…

[1] This bullet point is intended as sarcasm.

GOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOALS!

Cryptographic Random Bit Generator

Definition

A cryptographic random bit generator, with security bound L bits, produces sequences of random bits $(R_1, R_2, \ldots, R_n)$ such that

1. The generator is unbiased: $\Pr\left(R_j = 0\right) = \frac{1}{2}$.
2. The bits are uncorrelated: $\Pr\left(R_j = 0 | R_1, R_2, \ldots, R_{j-1}\right) = \frac{1}{2}$.
3. Negligible advantage: An attacker can't distinguish between a "true" random bit generator and the cryptographic random bit generator without performing at least 2^L operations.

Backtracking Resistance

Definition

Backtracking resistance is provided relative to time T if there is assurance that an adversary who has knowledge of the internal state of the DRBG at some time subsequent to time T would be unable to distinguish between observations of ideal random bitstrings and (previously unseen) bitstrings that were output by the DRBG prior to time T.

NIST SP 800-90A

Prediction Resistance

Definition

Prediction resistance is provided relative to time T if there is assurance that an adversary who has knowledge of the internal state of the DRBG at some time prior to T would be unable to distinguish between observations of ideal random bitstrings and bitstrings output by the DRBG at or subsequent to time T.

NIST SP 800-90A

- Note that this *requires* reseeding for any deterministic design.

Cycle Resistance

Definition

The random bit generator is said to have cycle resistance if there is a negligible probability that the generator enters a cycle when used as specified.

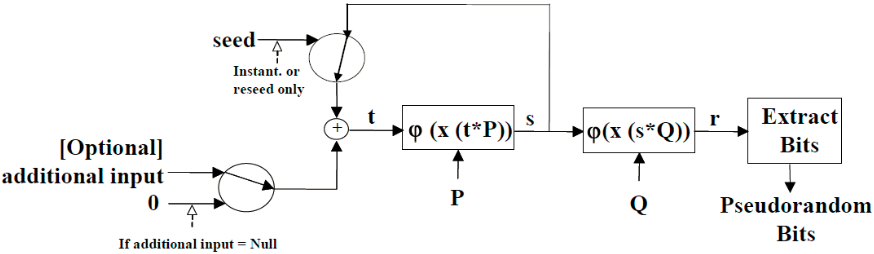
Here negligible probability means less than 2^{-40}.

Specifications of our Lives

Helper Functions

- $\varphi(\cdot)$ converts a field element to an integer in a canonical way.
- $x(\cdot)$ takes the x-coordinate in affine coordinates in the provided model for the EC.
- "Extract Bits" takes the rightmost (LSBs) of the value.

The Algorithm



NIST SP800-90A

Parameters

- The generator is intended to produce no more than 2^{32} blocks between reseeding events.
- P and Q are obviously very important to the security of this generator.
- Three curves (along with associated P and Q values) are provided.
- There is a procedure for generating your own values of P and Q.

When you ASSUME...

Elliptic Curve Decisional Diffie-Hellman Problem

Definition

Given an elliptic curve E and basepoint P, an attacker cannot distinguish between (qP, rP, qrP) and (qP, rP, zP), where q, r, and z are random values.

Truncated Point Problem

Definition

Let R be a random point and b a random bitstring matching the length of the output of the truncation function, t. The problem of distinguishing between $t(\varphi(x(R)))$ and b is the Truncated Point Problem.

See [Brown, Gjøsteen 2007]

x-logarithm Problem

Definition

Let E be an elliptic curve over $\mathbb{F}_q$, $P \in E(\mathbb{F}_q)$. Let $Z \in E(\mathbb{F}_q)$ be chosen uniformly at random and d a random integer in the range $[0, n-1]$. The x-logarithm problem is the problem of distinguishing between dP and $x(Z)P$.

See [Brown, Gjøsteen 2007]

No “There” There

Inadequate Truncation

- Due to [Schoenmakers, Sidorenko 2006]
- If too few bits are truncated, the generator has a predictor.
- This is as a result of modular arithmetic mod a prime.
- For k-bit random integers in $[0, 2^k - 1]$, the lth bit is random.
- If we restrict to some other (non-power of two length) range, this is no longer true.
- Thus, there is a small bias associated with the high order bits.
- Solution: remove at least 17 bits.

Too Much Truncation

- Asymptotic estimates of the distribution of x-coordinates by Shparlinski suggest that too much truncation may make a predictor possible as well.

The LSB for Binary Fields

- Due to [Brown, Gjøsteen 2007]
- A set of elliptic curves over binary fields are specified by NIST.
- B-409 and K-409 (both over $\mathbb{F}_{2^{409}}$) are such binary fields.
- These fields have the property that the LSB of the x value is fixed, so should be discarded.

"The Back Door"

- Due to [Shumow, Ferguson 2007]
- NIST Prime curves have prime order.
- Thus there is an integer e so that $eQ = P$.
- The Attack: An attacker knows e and the prior output R, and the number of bits the system truncates, m.
  - The attacker iterates through all 2^m possible values for x, say $x_1, \ldots, x_{2^m}$.
  - If $\hat{y}_j = x_j^3 + ax_j + b \pmod{p}$ is a square, then $(x_j, \pm\sqrt{\hat{y}_j})$ are points on our EC.
  - The correct point, A, must be in the resulting list.
  - We have $A = sQ$, so $eA = s(eQ) = sP$, so $\varphi(x(eA))$ is then the next *internal* state!
- This attack difficulty increases exponentially with the number of bits truncated.

"The Back Door"

- Due to [Shumow, Ferguson 2007]
- NIST Prime curves have prime order.
- Thus there is an integer e so that $eQ = P$.
- The Attack: An attacker knows e and the prior output R, and the number of bits the system truncates, m.
  - The attacker iterates through all 2^m possible values for x, say $x_1, \ldots, x_{2^m}$.
  - If $\hat{y}_j = x_j^3 + ax_j + b \pmod{p}$ is a square, then $(x_j, \pm\sqrt{\hat{y}_j})$ are points on our EC.
  - The correct point, A, must be in the resulting list.
  - We have $A = sQ$, so $eA = s(eQ) = sP$, so $\varphi(x(eA))$ is then the next *internal* state!
- This attack difficulty increases exponentially with the number of bits truncated.
- So, that would be bad then.

“The Back Door”

- Due to [Shumow, Ferguson 2007]
- NIST Prime curves have prime order.
- Thus there is an integer e so that $eQ = P$.
- The Attack: An attacker knows e and the prior output R, and the number of bits the system truncates, m.
    - The attacker iterates through all 2^m possible values for x, say $x_1, \ldots, x_{2^m}$.
    - If $\hat{y}_j = x_j^3 + ax_j + b \pmod{p}$ is a square, then $(x_j, \pm\sqrt{\hat{y}_j})$ are points on our EC.
    - The correct point, A, must be in the resulting list.
    - We have $A = sQ$, so $eA = s(eQ) = sP$, so $\varphi(x(eA))$ is then the next *internal* state!
- This attack difficulty increases exponentially with the number of bits truncated.
- So, that would be bad then.
- Does the NSA know e for the provided curves?

Computational Notes

- This generator is orders of magnitude slower than any of the common (non-number theoretic) RBG designs.
- It is considerably faster than any of the common number theoretic RBGs.
  - EC security (exponential) vs non-EC security (often sub-exponential).
  - Other EC generators output only a single bit per EC point scaling operation.

Section 7

Conclusion

Today's Conclusion

- Reseed often.
- Generate your own P, Q.
- Truncate aggressively, but not *too* aggressively.

Thank You!

Bibliography

- Barker, Elaine and Kelsey, John. NIST Special Publication 800-90A, "Recommendation for Random Number Generation Using Deterministic Random Bit Generators", January 2012. `http://csrc.nist.gov/publications/nistpubs/800-90A/SP800-90A.pdf`
- Brown, Daniel R.L. and Gjøsteen, Kristian. "A Security Analysis of the NIST SP 800-90 Elliptic Curve Random Number Generator", February 2007. `http://eprint.iacr.org/2007/048`
- Schoenmakers, Berry and Sidorenko, Andrey. "Cryptanalysis of the Dual Elliptic Curve Pseudorandom Generator", May 2006. `http://eprint.iacr.org/2006/190`
- Ferguson, Niels and Shumow, Dan. "On the Possibility of a Back door in the NIST SP800-90 Dual EC PRNG", August, 2007. `http://rump2007.cr.yp.to/15-shumow.pdf`

Colophon

- The principal font is Evert Bloemsma's 2004 humanist sans-serif font Legato. This font is designed to be exquisitely readable, and is a significant departure from the highly geometric forms that dominate most sans-serif fonts. Legato was Evert Bloemsma's final font prior to his untimely death at the age of 46.
- The URLs are typeset in Luc(as) de Groot's 2005 Consolas, a monospace font with excellent readability.