Entrust®
Securing Digital Identities & Information

EntrustSecurity
@Entrust

Zero to Dual_EC_DRBG in 30 minutes

A look into the suspect Dual-EC DRBG cryptographic mechanism

Entrust.com
1-888-690-2424
Entrust@Entrust.com

Entrust®
Securing Digital Identities & Information

Table of Contents

Introduction

The U.S. National Institute of Standards and Technology (NIST) is one of the oldest physical science laboratories in the United States. NIST helps promote domestic innovation by advancing, monitoring and evaluating technical standards and measurement sciences.

Core to its mission, NIST produces special publications to disseminate policies, standards and findings to related security and technology communities.

However, special publication 800-90, first published in 2006, came under heavy criticism from the media, who claimed that security vendor RSA and the National Security Agency (NSA) created a deal to make the dual-EC (elliptic curve) variant the default deterministic random-bit generator algorithm, or DRBG, in its commercial toolkit product. RSA denied the allegation.

The claims introduce serious questions about the security of the algorithm. Random-bit generation is a critical foundation of every security protocol. The presence of a backdoor would have serious implications for security everywhere the algorithm is used.

Fortunately, the NIST specification describes three alternative algorithms, all of which were based on well-established cryptographic principles. By “well-established” we mean that the academic community had examined their security properties over many years and was satisfied that they were cryptographically sound.

Because of the critical role they play in every security protocol, Entrust pays close attention to the design of random-bit generators, and it does not use NIST’s Dual-EC DRBG in any of its products or services.

Indeed, with the April 2014 announcement by NIST that it is withdrawing support for Dual-EC DRBG, implementations of the algorithm have become non-compliant.

This white paper provides an introduction to the elliptic-curve DRBG mechanism and why the design approach is suspect.

Building Block

The basic building block of all elliptic-curve cryptographic mechanisms is integer-point multiplication (see White Paper: "Zero to ECC in 30 minutes"). In subsequent diagrams we use the symbol shown in **Figure 1** to denote this operation.

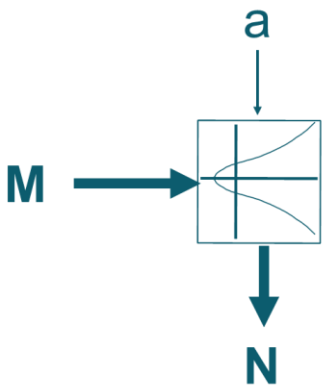

Figure 1: ECC Building Block

Bold upper-case characters and thick lines denote curve points, and normal font and thin lines denote integers. Integer-point multiplication is one-way, because, while, given *a* and **M**, it is easy to calculate **N**, given **M** and **N**, it is computationally infeasible to calculate *a*. Calculating *a* is known as the elliptic-curve discrete logarithm problem.

> *"Random-bit generation is a critical foundation of every security protocol."*

NIST's Dual_EC_DRBG

The NIST Dual_EC_DRBG is described in NIST Special Publication 800-90A. One round of the procedure is shown in **Figure 2**.

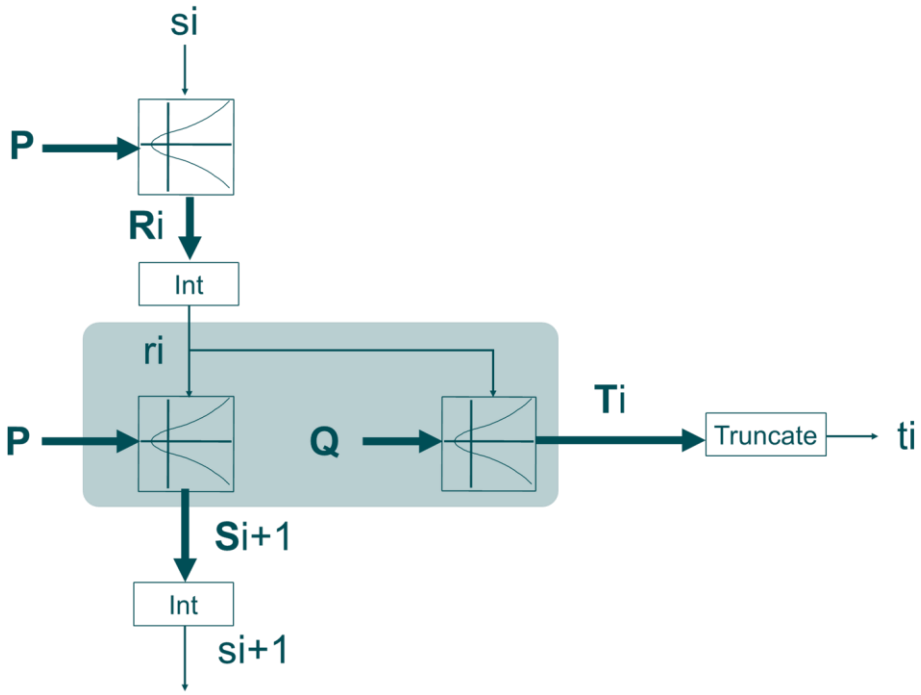

Figure 2: One Round

Note that all three integer-point multiplication operations shown here use the same elliptic curve with a base-point of either **P** or **Q**. The 'int' operation denotes taking the input point's x coordinate as an integer value. And the 'truncate' operation denotes taking the input point's x coordinate and removing the least significant 16 bits.

ti is the output random bit sequence. The output, *Si+1*, becomes the input for the next round, so that longer random sequences can be generated from one seed. Bear in mind that, while it is easy to calculate *si+1* and *ti*, it is computationally infeasible, given *ti*, to calculate the corresponding *ri*.

The initial round is seeded from a non-deterministic random-bit generator.

Points form a Group

Consider the example curve shown in **Figure 3**. This curve satisfies the equation:

$$Y^2 = x^3 + 5x + 1 \bmod 23$$

Obviously, this example is for illustrative purposes only; it has no cryptographic utility. The curve has group order 31; i.e., there are 31 points on the curve, including the additive identity, or point at infinity **(0)**.

31 is a prime number. So, every point on the curve is an integer multiple mod 31 of every other point except **0**. And 31**P** = **0**, regardless of which point is chosen for **P**.

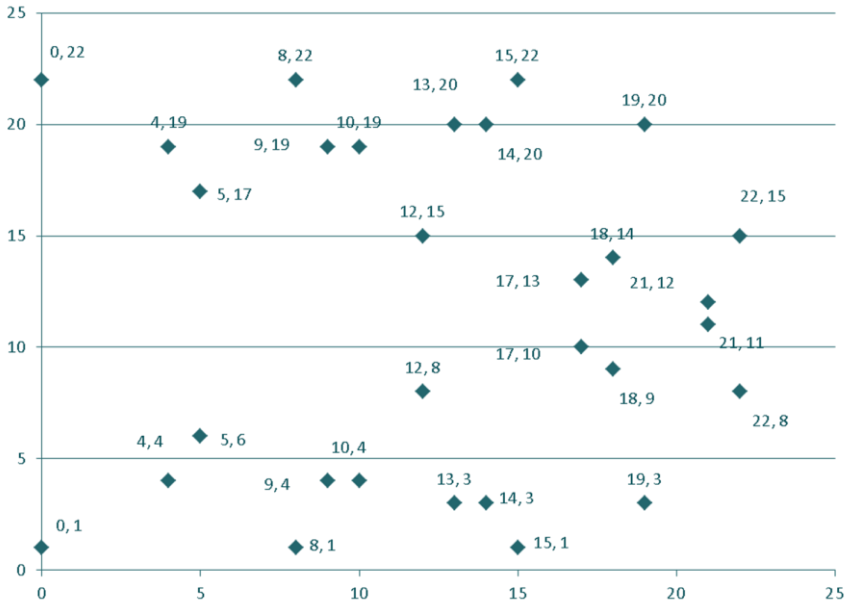

Figure 3: Example Curve

Suppose **Q** were the point (9,4), as shown in **Figure 4**. And suppose that **P** were the point (12,8).

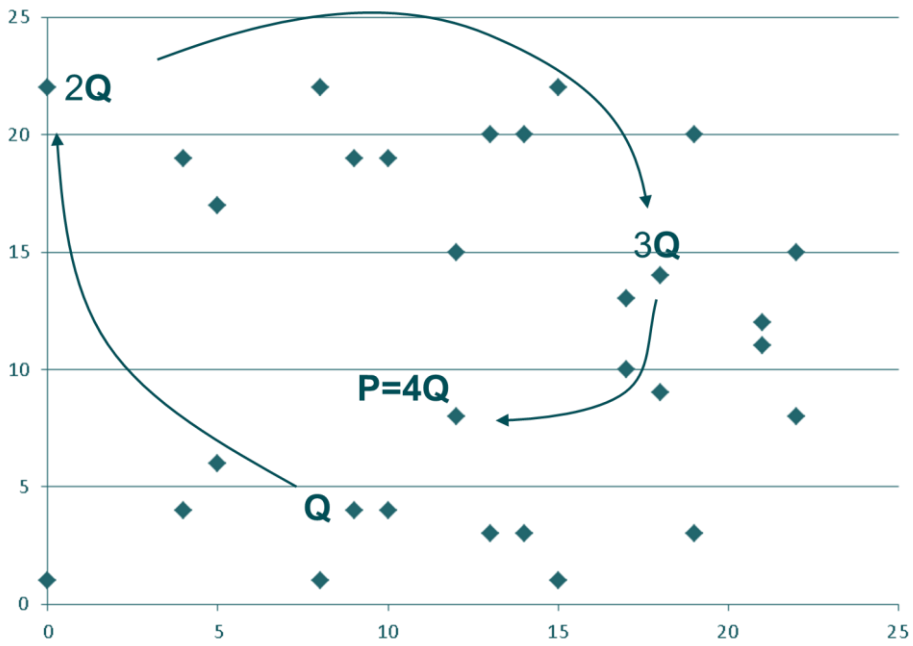

Figure 4: P = (12,8)

Then, the figure shows multiples of **Q** up to 4**Q**, and we see that **P** = 4**Q.** More generally, **P** = e**Q**.

The Attack

Multiplying interim state, *ri*, of the Dual_EC_DRBG by **Q** and then multiplying the resulting point by *e* produces the same result as multiplying the interim state by the point **P**. See **Figure 5**.

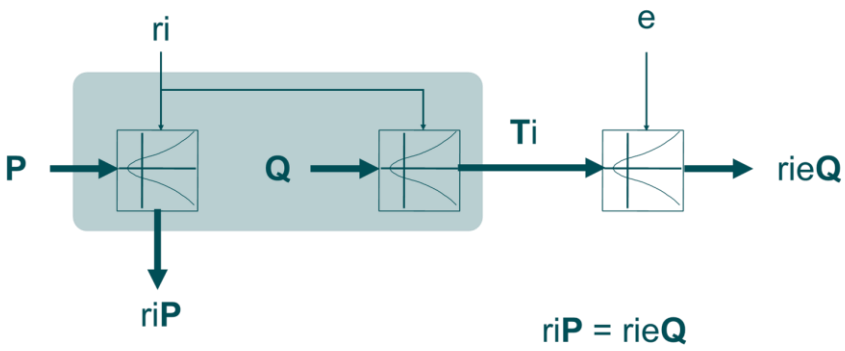

Figure 5: The Attack

In other words, the internal state can be calculated from the output by anyone who knows *e*. Clearly, the designers of the algorithm **COULD** know *e*, although it cannot be proved whether they do or not.

There is one final complication. Only a truncated version of the output curve point is known, *ti*, rather than **T**i.

In order to multiply the curve point **T**i by *e*, the attacker has to reconstruct the curve point that produced *ti*. See **Figure 7**.

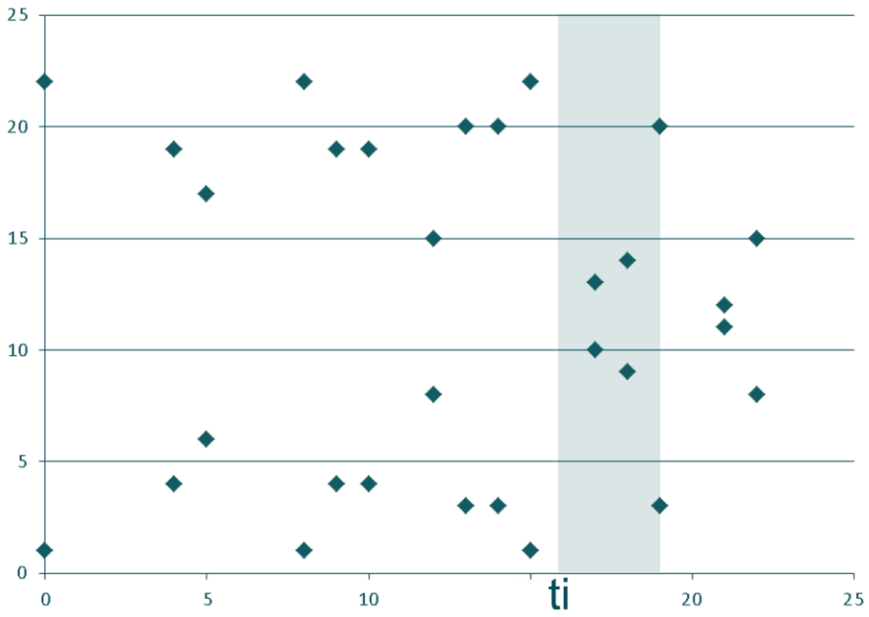

Figure 7: The Effect of Truncation

ti is calculated by taking the *x* coordinate of **T**i and truncating it by 16 bits. So, the attacker has to try all 65,536 corresponding *x* coordinates and find all those that have a solution on the curve.

In our example, if *ti* = 4 and we truncate by two bits, then the following points are candidate values for **T**i: (17,10), (17,13), (18,9), (18,14), (19,3) and (19,20).

Once the candidates have been identified, then subsequent output for each of them can easily be calculated. If subsequent output is used to generate keys, for instance, then the attacker has a small number of key values to search through in order to find the correct one.

This attack depends upon the RBG being used to produce a public random number (such as a nonce, a challenge or an initialization vector) and then subsequently using it to generate a secret value, such as a key.

Conclusion

Either the designers failed to appreciate that their design approach was vulnerable to this criticism, or they knew perfectly well, but didn't expect others to discover the vulnerability.

Both explanations seem equally improbable. But, one of them has to be true. Either way, the mechanism is now thoroughly discredited.

Entrust & You

More than ever, Entrust understands your organization's security pain points. Whether it's the protection of information, securing online customers, regulatory compliance or large-scale government projects,

Entrust provides identity-based security solutions that are not only proven in real-world environments, but cost-effective in today's uncertain economic climate.

Entrust offers software authentication platforms that strengthen security in a wide range of identity and transaction ecosystems. Government agencies, financial institutions and other enterprises rely on Entrust solutions to strengthen trust and reduce complexity for consumers, citizens and employees.

Now, as part of Datacard Group, Entrust offers an expanded portfolio of solutions across more than 150 countries. Together, Datacard Group and Entrust issue more than 10 million secure identities every day, manage billions of secure transactions annually and issue a majority of the world's financial cards.

For more information about Entrust products and services, call **888-690-2424**, email **entrust@entrust.com** or visit **entrust.com**.

Company Facts

Website: www.entrust.com
Employees: 359
Customers: 5,000
Offices: 10 Globally

Headquarters

Three Lincoln Centre
5430 LBJ Freeway, Suite 1250
Dallas, Texas 75240

Sales

North America: 1-888-690-2424
EMEA: +44 (0) 118 953 3000
Email: entrust@entrust.com