Surveillance et cryptographie: peut-on faire confiance aux courbes elliptiques ?

gapz+conf@dud-t.org - https://residus.eu.org

Pas Sage en Seine

29 juin 2017

Plan

La confiance dans les standards : retour sur Dual_EC_DRBG

Rappel sur les courbes elliptiques

Transparence & courbes elliptiques

Qui fabrique et "standardise" la cryptographie ?

Côté théorique :

- Universités
- Entreprises privées
- Agences gouvernementales

Côté recommandations/standards :

- NIST (USA), GOST (Russie), KISA (Corée du sud)
- IETF/CFRG
- Industriels (PKCS, etc)

(pas du tout exhaustif)

Manipulation des standards : le cas Dual_EC_DRBG

2013, projet SIGINT de la NSA rendu public

- insert vulnerabilities into commercial encryption systems, IT systems, networks, and endpoint communications devices used by targets
- Influence policies, standards and specification for commercial public key technologies

Dual_EC_DRBG

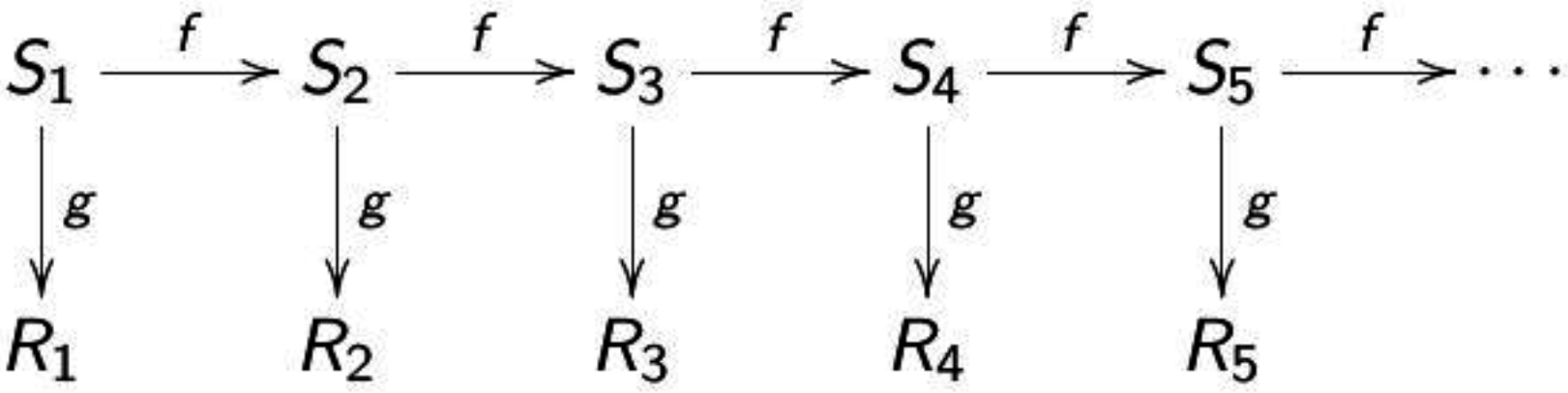
- générateur de nombres pseudo-aléatoires
- ≪ backdoor ≫ NOBUS

Manipulation des standards : le cas Dual_EC_DRBG

Quelques dates choisies

- 1997, développement de PRNG "backdoorés" basés sur la théorie des nombres
- 2004, première présentation lors d'un workshop du NIST
- 2005, brevets par CERTICOM sur un système de ≪ key escrow ≫
- 2007, backdoor découverte par deux chercheurs de chez Microsoft lors de la conf Crypto (pendant les rumps sessions)
- 2014, mise en pratique de Dual_EC par des chercheurs pour déchiffrer un flux TLS
- 2014, abandon de Dual_EC_DRBG par le NIST (et mise en garde)

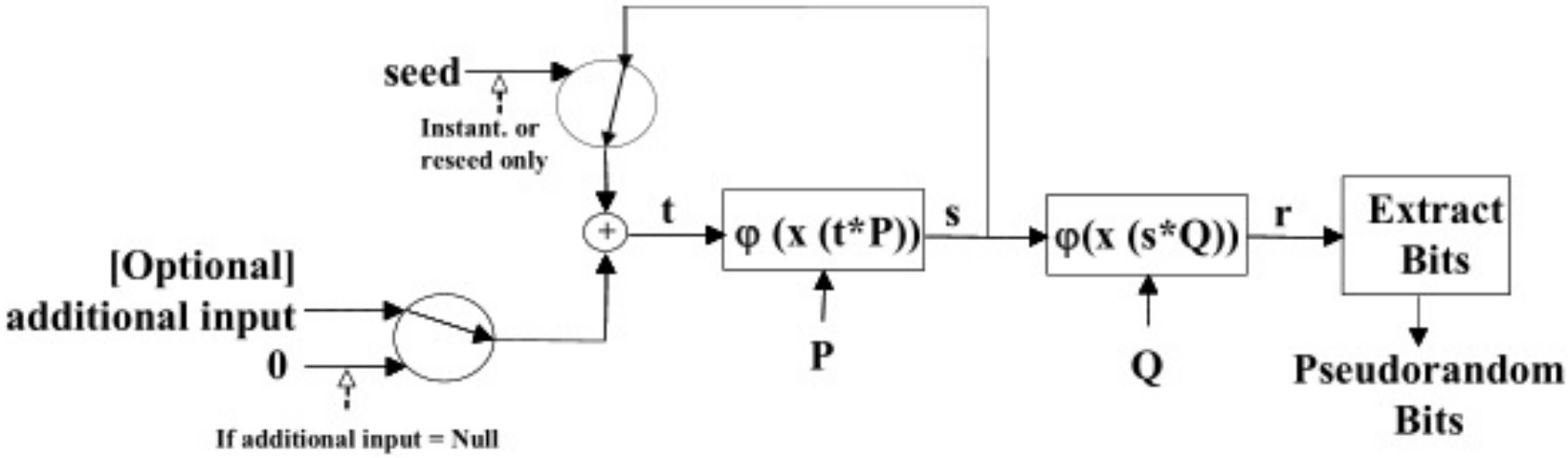
Manipulation des standards : le cas Dual_EC_DRBG (2)



Une porte-dérobée dans un PRNG ?

- trouver l'état interne en se basant sur les sorties précédentes
- déterminer ensuite les futures sorties/nombres qui seront générés

Manipulation des standards : le cas Dual_EC_DRBG (3)



La porte dérobée

- preuve de concept basée sur seulement deux « rounds »
- relation *bien choisie* entre P et Q

Manipulation des standards : le cas Dual_EC_DRBG (4)

Une affaire de constante

---------- Original Message ----------
Subject: RE: Minding our Ps and Qs in Dual_EC
From: "Don Johnson" <DJohnson@cygnacom.com>
Date: Wed, October 27, 2004 11:42 am
To: "John Kelsey" <john.kelsey@nist.gov>

John,

P = G.
Q is (in essence) the public key for some random private key.

It could also be generated like a(nother) canonical G, but NSA kyboshed this idea, and I was not allowed to publicly discuss it, just in case you may think of going there.

Don B. Johnson

Manipulation des standards : le cas Dual_EC_DRBG (5)

```
% ./dual_ec_poc
[*] Candidates generation (based on one round of the DRBG)
[*] Testing candidates (based on one more round of the DRBG)
[*] Success! Future output:
    -> next output 1:
b7446786f160ad81b9e4d92695624d53a68309a87eb9cbdf71adb04d02aa
    -> next output 2:
a7691b4fe1b78df0866954685ef436937ce7e5db815e7dbd0007e2bf3eb6

[*] Dual_EC output 1:
b7446786f160ad81b9e4d92695624d53a68309a87eb9cbdf71adb04d02aa
[*] Dual_EC output 2:
a7691b4fe1b78df0866954685ef436937ce7e5db815e7dbd0007e2bf3eb6
```

Manipulation des standards : le cas Dual_EC_DRBG (6)

Mais ça ne fonctionnera jamais en pratique !

On the Practical Exploitability of Dual EC in TLS Implementations

Stephen Checkoway,[1] Matthew Fredrikson,[2] Ruben Niederhagen,[3] Adam Everspaugh,[2]
Matthew Green,[1] Tanja Lange,[3] Thomas Ristenpart,[2]
Daniel J. Bernstein,[3,4] Jake Maskiewicz,[5] and Hovav Shacham[5]
[1] *Johns Hopkins University,* [2] *University of Wisconsin,* [3] *Technische Universiteit Eindhoven,*
[4] *University of Illinois at Chicago,* [5] *UC San Diego*

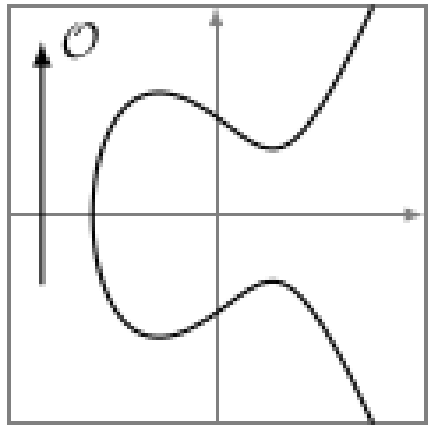
- sous certaines conditions, on récupère les clefs de session TLS en moins d'une seconde (attaque passive)

Manipulation des standards : le cas Dual_EC_DRBG (7)

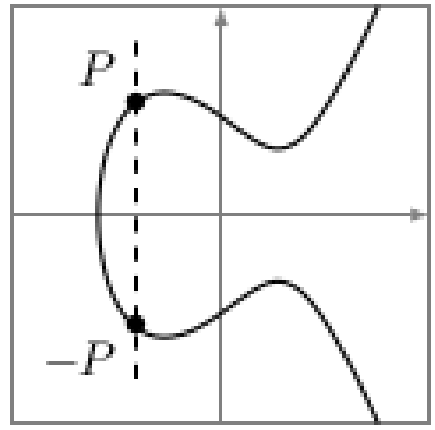
Conclusions

- réputation du NIST
- questionnement par rapport à la présence officielle de certaines agences au sein d'organismes de standardisation et/ou recommandation
- confiance dans d'autres standards issus du même processus
- peut-on apporter plus de transparence dans le choix des constantes ?

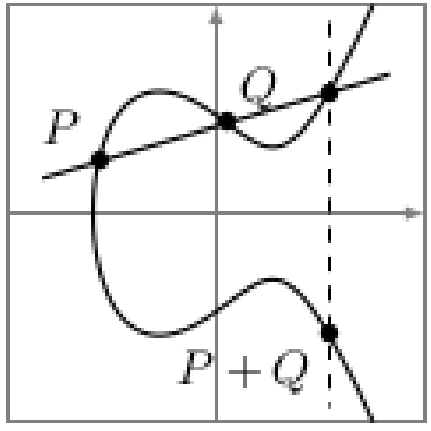
Qu'est-ce que "les courbes elliptiques" ?



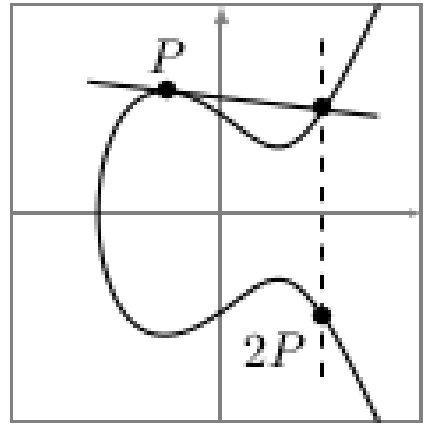
Neutral element $\mathcal{O}$



Inverse element $-P$

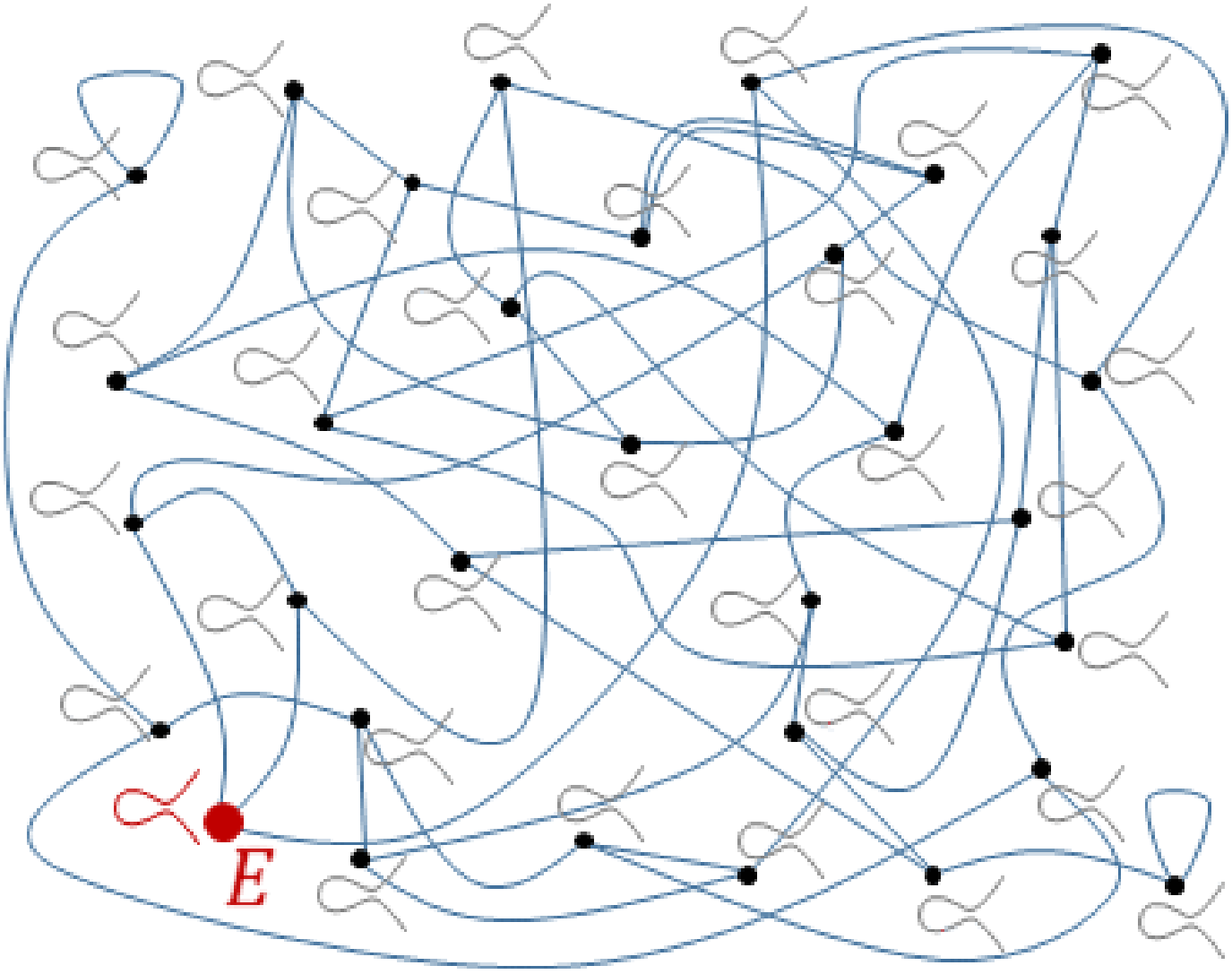


Addition $P + Q$
"Chord rule"



Doubling $P + P$
"Tangent rule"

Qu'est-ce que "les courbes elliptiques" ?



Qu'est-ce que "les courbes elliptiques" ?

ssl_ciphers "EECDH+AESGCM :EDH+AESGCM :ECDHE-RSA-AES128-GCM-SHA256 :AES256+EECDH

...

Qu'est-ce que "les courbes elliptiques" ?

Quelques dates choisies

- 1985, les débuts avec Kobiltz et Miller
- 1990, ECDSA
- 1999, courbes du NIST
- 2005, suite B de la NSA
- 2005, implémentation de Curve25519 dans OpenSSL
- 2015, annonce d'un transition de la suite B vers des algos post-quantiques

Qu'est-ce que "les courbes elliptiques" ?

Aspects théoriques

- cryptographie asymétrique
- logarithme discret
- des plus petites clefs pour un même niveau de sécurité

Les courbes

- NIST-P256, SECP256R1, BRAINPOOLP256, FRP256V1, GOST256, Ed448-Goldilocks, etc

Les usages

- des schémas de signature : ECDSA
- échange de clef : ECDH
- schéma de chiffrement : ECIES

De quoi sont faites les courbes elliptiques ?

Des constantes

- équations, exemple classique : $y^2 = x^3 + ax + b$
- point de base
- un nombre premier (définis le corp dans lequel on est)
- cofacteur, ordre du groupe, ...

Mécanismes pour la génération des constantes

Critères de sélection

- sécurité
- performance
- généricité
- implémentation

Processus de génération

- déterministe
- aléatoire

Quid de la vérifiabilité/transparence du processus de génération ?

Mécanismes pour la génération des constantes (1)

Génération déterministe

- constantes dérivées d'un hash (NIST-P, $\sqrt{\frac{-27}{SHA1(s)}}$)
- NUMS (cos(1), sqrt(2))
- optimisation pour les performances (Curve25519)

Optimiser les performances ?

- nombre premier structuré ($2^{2s} - c$)
- optimisation de certaines opérations (racine carrée)

Mécanismes pour la génération des constantes (2)

Génération aléatoire

- source d'aléa public (loteries, Million Dollar Curve)
- « premier » aléatoire (BRAINPOOLP)
- constantes tombées du ciel

NOM	FRP256v1
p	F1FD178C0B3AD58F10126DE8CE42435B3961ADBCABC8CA6DE8FCF353D86E9C03
A	F1FD178C0B3AD58F10126DE8CE42435B3961ADBCABC8CA6DE8FCF353D86E9C00
B	EE353FCA5428A9300D4ABA754A44C00FDFEC0C9AE4B1A1803075ED967B7BB73F
x(P0)	B6B3D4C356C139EB31183D4749D423958C27D2DCAF98B70164C97A2DD98F5CFF
y(P0)	6142E0F7C8B204911F9271F0F3ECEF8C2701C307E8E4C9E183115A1554062CFB
q	F1FD178C0B3AD58F10126DE8CE42435B53DC67E140D2BF941FFDD459C6D655E1
i	1

Du côté des standards/recommandations

CFRG/IETF

- grand débat pour les recommandations pour TLS 1.3
- RFC 7748, ≪ Elliptic Curves for Security ≫ (Janvier 2016)
- les courbes retenues ont des paramètres ≪ transparents/vérifiables ≫

NIST

- ne revient pas sur les courbes de 1999 (NIST-P256,384)
- se pose la question d'en émettre de nouvelles

ANSSI

- publication de ≪ Diversity and Transparency for ECC ≫ lors d'un workshop NIST en 2015

Conclusions

- la transparence des paramètres en crypto est indispensable
- transparent ne veut pas dire infaillible
- la confiance ne se donne pas mais se *construit*
- les courbes existantes a priori ≪ sûres ≫

Remerciements

- pour leurs travaux : Tanja Lange, Daniel Bernstein, Matthew Green et leurs nombreux collaborateurs
- pour la FOIA : EFF

Illustrations

- le schéma d'un PRNG : https ://projectbullrun.org
- TikZ for cryptographers : https ://www.iacr.org/authors/tikz/
- le ≪ dessin ≫ des courbes : https ://www.esat.kuleuven.be/

Merci de votre attention.

Des questions ?

`https://residus.eu.org/articles.html`
`https://residus.eu.org/code/dual_ec_poc.go`