Radboud University

Bookmarks for Cryptographers

Benoît Viguier

(λ x y. x@y.nl) benoit viguier

CHES 2017 – Rump Session

Taipei, 26th September 2017

Institute for Computing and Information Sciences – Digital Security
Radboud University Nijmegen

Tikz for Cryptographer

Facts

- Papers/presentations using **Figures** can only be better.
  - They illustrate textual arguments.
  - Complex ideas can often be simply explained using pictures.
  - People prefer pictures over text anyway.

Facts

- Papers/presentations using **Figures** can only be better.
  - They illustrate textual arguments.
  - Complex ideas can often be simply explained using pictures.
  - People prefer pictures over text anyway.

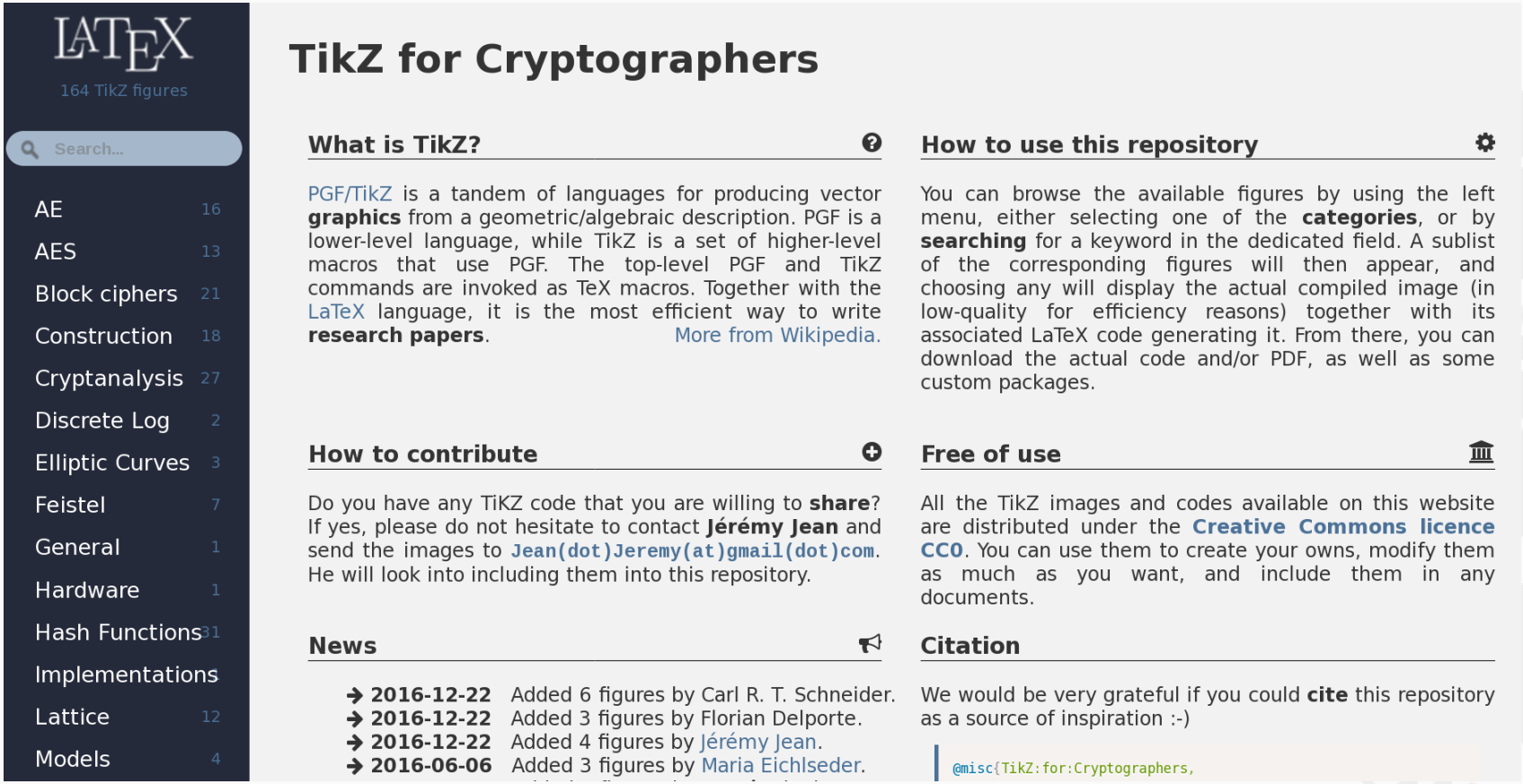
- However, drawing them can be
  - tedious,
  - frustrating,
  - time consuming

Facts

- Papers/presentations using **Figures** can only be better.
  - They illustrate textual arguments.
  - Complex ideas can often be simply explained using pictures.
  - People prefer pictures over text anyway.
- However, drawing them can be
  - tedious,
  - frustrating,
  - time consuming
- But: there exist tools to draw them straight from LaTeX
  - **TikZ !**
  - The results usually look really good.
  - It can produce reusable PDF images.

Contribution

An online repository of TikZ figures.



https://www.iacr.org/authors/tikz/

Example

You look for the round function of the PRESENT block cipher.

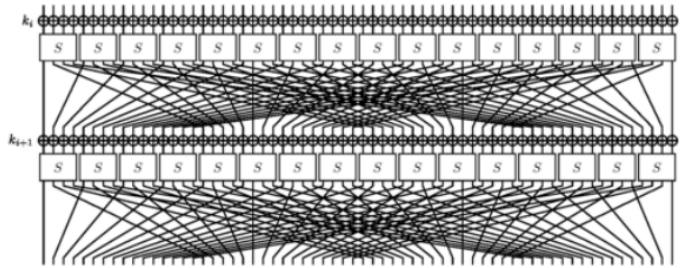
LaTeX
101 TikZ figures

Search...

- AE 7
- AES 13
- Block ciphers 12
- Construction 11
- Cryptanalysis 20
- Feistel 4
- General 1
- Hash Functions 26
- Models 4
- Modes 3

- Block cipher black-box representation
- Key-alternating cipher
- LBlock F-function
- LBlock round function
- LED-128 block cipher
- LED-64 block cipher
- PRESENT block cipher
- PRINCE block cipher
- Piccolo F-function
- Piccolo RP Permutation
- Piccolo block cipher
- TWINE round function

Title: PRESENT block cipher.
Author: Jérémy Jean.
Date: February 2015.
Section: Block ciphers.
Required packages : crypto symbols • custom arrows.
Download files: PNG • PDF • TEX.

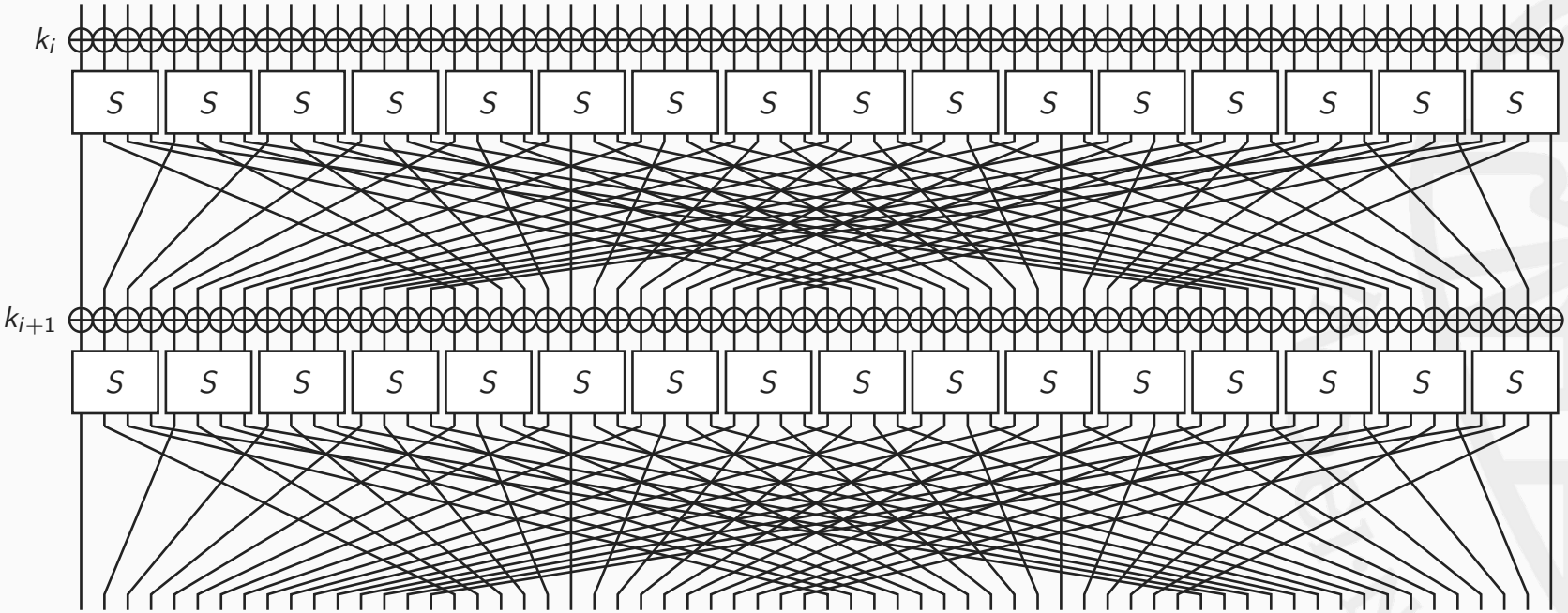



```
\documentclass{standalone}

%% Common TikZ libraries
\usepackage{tikz}
\usetikzlibrary{calc}
\usetikzlibrary{positioning}

%% Custom TikZ addons
\usetikzlibrary{crypto.symbols}
\tikzset{shadows=no}        % Option: add shadows to XOR, ADD, etc.

%% Document
\begin{document}
\begin{tikzpicture}
```

Example

```
\begin{tikzpicture}

    %% Subkey XORs
    \foreach \z in {0,...,63} {
        \node[XOR, scale=0.8] (xor\z) at ($\z*(0.75em, 0)$) {};
        \node[XOR, scale=0.8] (xorr\z) at ($\z*(0.75em, 0)+(0,-9em)$) {};
    }

    %% Nodes positions
    \foreach \z in {0,...,63} {
        \node (i\z) [above = 0.75em of xor\z] {};
        \node (o\z) [below = 2.5em of xor\z] {};
        \node (ii\z) [above = 0.25em of xorr\z] {};
        \node (oo\z) [below = 3em of xorr\z] {};
        \node (t\z) [below = 4em of oo\z] {};
        \draw[thick] (i\z) -- (xor\z);
    }

    %% Permutation layer
    \foreach \z [evaluate=\z as \zz using {int(mod(16*\z,63))}] in {0,...,62} {
       \draw[thick] (xor\z)  -- (o\z.center)  -- (ii\zz.center) -- (xorr\zz) -- (oo\zz);
       \draw[thick] (oo\z.north)  -- (t\zz.south) -- +(0,-0.5em);
    }
    \draw[thick] (xor63) -- (o63.center) -- (ii63.center) -- (xorr63) -- (oo63);
    \draw[thick] (oo63.north) -- (t63.south) -- +(0,-0.5em);

    %% SBoxes
    \foreach \z in {0,...,15} {
          \node[draw,thick,minimum width=2.75em,minimum height=2em,fill=white] (p4) at ($\z*(3em,0) + (1.1em,-2em)$) {$S$};
          \node[draw,thick,minimum width=2.75em,minimum height=2em,fill=white] (p4) at ($\z*(3em,0) + (1.1em,-11em)$) {$S$};
    }

    \node[left = 0em of xor0] {$k_{i}$};
    \node[left = 0em of xorr0] {$k_{i+1}$};

\end{tikzpicture}
```

Example



Details

- Already online

`https://www.iacr.org/authors/tikz/`

Details

- Already online

  `https://www.iacr.org/authors/tikz/`

- Currently
  - About 100 different pictures.
  - All share (almost) the same look.
  - Mostly symmetric-key related content.

Details

- Already online

  `https://www.iacr.org/authors/tikz/`

- Currently
  - About 100 different pictures.
  - All share (almost) the same look.
  - Mostly symmetric-key related content.

- Goals
  - Help the crypto community write better papers.
  - Gather all crypto-related pictures in a single place.
  - **Encourage you to submit and share your crypto figures!**

Details

- Already online

  `https://www.iacr.org/authors/tikz/`

- Currently
  - About 100 different pictures.
  - All share (almost) the same look.
  - Mostly symmetric-key related content.

- Goals
  - Help the crypto community write better papers.
  - Gather all crypto-related pictures in a single place.
  - **Encourage you to submit and share your crypto figures!**

Thanks to Jérémy Jean

Cryptography Stack Exchange

The Community

- New to crypto or already well versed ?
- wish to share your knowledge ?
- want to know more about other domains ?

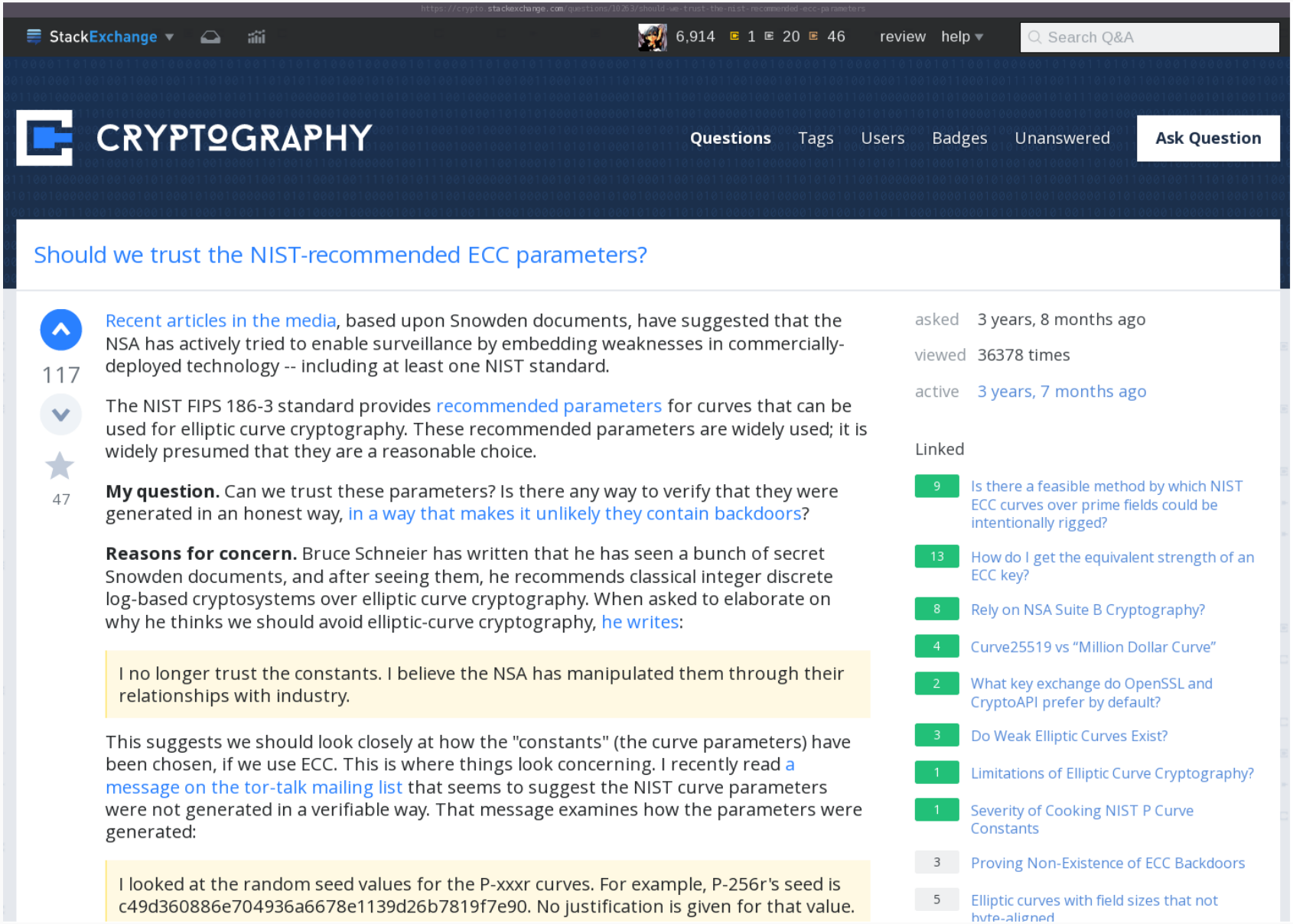
The Community

- New to crypto or already well versed ?
- wish to share your knowledge ?
- want to know more about other domains ?

Join **CRYPTO STACK EXCHANGE** and

- Ask questions
- Answer questions
- Bonus : Check that your students are not cheating ! :D

Website


StackExchange
6,914 1 20 46
review help
Search Q&A
CRYPTOGRAPHY
Questions Tags Users Badges Unanswered
Ask Question
Should we trust the NIST-recommended ECC parameters?
117
47
Recent articles in the media, based upon Snowden documents, have suggested that the NSA has actively tried to enable surveillance by embedding weaknesses in commercially-deployed technology -- including at least one NIST standard.
The NIST FIPS 186-3 standard provides recommended parameters for curves that can be used for elliptic curve cryptography. These recommended parameters are widely used; it is widely presumed that they are a reasonable choice.
My question. Can we trust these parameters? Is there any way to verify that they were generated in an honest way, in a way that makes it unlikely they contain backdoors?
Reasons for concern. Bruce Schneier has written that he has seen a bunch of secret Snowden documents, and after seeing them, he recommends classical integer discrete log-based cryptosystems over elliptic curve cryptography. When asked to elaborate on why he thinks we should avoid elliptic-curve cryptography, he writes:
I no longer trust the constants. I believe the NSA has manipulated them through their relationships with industry.
This suggests we should look closely at how the "constants" (the curve parameters) have been chosen, if we use ECC. This is where things look concerning. I recently read a message on the tor-talk mailing list that seems to suggest the NIST curve parameters were not generated in a verifiable way. That message examines how the parameters were generated:
I looked at the random seed values for the P-xxxr curves. For example, P-256r's seed is c49d360886e704936a6678e1139d26b7819f7e90. No justification is given for that value.
asked 3 years, 8 months ago
viewed 36378 times
active 3 years, 7 months ago
Linked
9 Is there a feasible method by which NIST ECC curves over prime fields could be intentionally rigged?
13 How do I get the equivalent strength of an ECC key?
8 Rely on NSA Suite B Cryptography?
4 Curve25519 vs “Million Dollar Curve”
2 What key exchange do OpenSSL and CryptoAPI prefer by default?
3 Do Weak Elliptic Curves Exist?
1 Limitations of Elliptic Curve Cryptography?
1 Severity of Cooking NIST P Curve Constants
3 Proving Non-Existence of ECC Backdoors
5 Elliptic curves with field sizes that not

The People

Some of the members...

- Poncho *aka. Scott Fluhrer*
  Weaknesses in the Key Scheduling Algorithm of RC4
- Thomas Pornin
  BearSSL
- Yehuda Lindell
  Introduction to Modern Cryptography - Katz Lindell
- Samuel Neves
  NORX Designer
- ...

https://www.iacr.org/authors/tikz/

https://crypto.stackexchange.com