Faster ECDL

D. J. Bernstein
University of Illinois at Chicago

Tanja Lange
Technische Universiteit Eindhoven

Peter Schwabe
Technische Universiteit Eindhoven

ECDL = complete ECC break,
computing user's secret key
given user's public key.

The Certicom challenges

1997: ECCp-79 broken.

1997: ECC2-79 broken.

1998: ECC2-79 broken.

1998: ECC2-89 broken.

1998: ECCp-97 broken.

1998: ECC2K-95 broken.

1999: ECC2-97 broken.

2000: ECC2K-108 broken.

2002: ECCp-109 broken.

2004: ECC2-109 broken.

2009–: ECC2K-130 in progress;
many optimizations; still
$10\times$ harder than RSA-768.

Challenges too widely spaced!

Latest ECDL record

2009.07 Bos–Kaihara–
Kleinjung–Lenstra–Montgomery
"PlayStation 3 computing
breaks 2^{60} barrier:
112-bit prime ECDLP solved".

Successful ECDL computation
for a standard curve over $\mathbf{F}_p$
where $p = (2^{128} - 3)/(11 \cdot 6949)$.

Latest ECDL record

2009.07 Bos–Kaihara–Kleinjung–Lenstra–Montgomery "PlayStation 3 computing breaks 2^{60} barrier: 112-bit prime ECDLP solved".

Successful ECDL computation for a standard curve over $\mathbf{F}_p$ where $p = (2^{128} - 3)/(11 \cdot 6949)$.

"We did not use the common negation map since it requires branching and results in code that runs slower in a SIMD environment."

2009.07 Bos–Kaihara–Kleinjung–Lenstra–Montgomery "On the security of 1024-bit RSA and 160-bit elliptic curve cryptography":

Group order $q \approx p$;
"expected number of iterations" is "$\sqrt{\frac{\pi \cdot q}{2}} \approx 8.4 \cdot 10^{16}$"; "we do not use the negation map"; "456 clock cycles per iteration per SPU"; "24-bit distinguishing property" $\Rightarrow$ "260 gigabytes".

"The overall calculation can be expected to take approximately 60 PS3 years."

2009.09 Bos–Kaihara–Montgomery "Pollard rho on the PlayStation 3":

"Our software implementation is optimized for the SPE . . . the computational overhead for [the negation map], due to the conditional branches required to check for fruitless cycles [13], results (in our implementation on this architecture) in an overall performance degradation."

"[13]" is 2000 Gallant–Lambert–Vanstone.

2010.07 Bos–Kleinjung–Lenstra
"On the use of the negation map
in the Pollard rho method":

"If the Pollard rho method is
parallelized in SIMD fashion,
it is a challenge to achieve any
speedup at all. . . . Dealing with
cycles entails administrative
overhead and branching, which
cause a non-negligible slowdown
when running multiple walks in
SIMD-parallel fashion. . . .
[This] is a major obstacle
to the negation map
in SIMD environments."

2010 Bernstein–Lange–Schwabe:
Our software solves
random ECDL on the same curve
(with no precomputation)
in 35.6 PS3 years on average.

For comparison:
Bos–Kaihara–Kleinjung–Lenstra–
Montgomery code
uses 65 PS3 years on average.

2010 Bernstein–Lange–Schwabe:
Our software solves
random ECDL on the same curve
(with no precomputation)
in 35.6 PS3 years on average.

For comparison:
Bos–Kaihara–Kleinjung–Lenstra–Montgomery code
uses 65 PS3 years on average.
Computation used 158000 kWh
(if PS3 ran at only 300W),
wasting >70000 kWh,
unnecessarily generating >10000
kilograms of carbon dioxide.
(0.143 kg CO2 per Swiss kWh.)

Several levels of speedups,
starting with fast arithmetic
and continuing up through rho.

Most important speedup:
We use the negation map
in a reasonable way.
Speedup very close to $\sqrt{2}$.

We also save time by using
better integer representation,
better multiplication methods,
adapting ideas from Curve25519.

Paper will be online very soon.

Advertisement

Fourteenth annual Workshop on Elliptic-Curve Cryptography is embedded into a larger Workshop on Elliptic Curves and Computation.

18–22 October 2010, Redmond, Washington, USA

http://eccworkshop.org