

OpenBSD: Where crypto is heading?

Mike Belopuhov

.vantronix secure systems

mikeb@openbsd.org

Moscow, December 14 2013

THIS AYN RANDOM NUMBER GENERATOR YOU WROTE CLAIMS TO BE FAIR, BUT THE OUTPUT IS BIASED TOWARD CERTAIN NUMBERS.

WELL, MAYBE THOSE NUMBERS ARE JUST INTRINSICALLY BETTER!



<http://xkcd.com/1277>

Pseudo-random number generator

rand (ANSI C, POSIX)

Pseudo-random number generator

rand (ANSI C, POSIX)

*rand48 (POSIX)

Pseudo-random number generator

rand (ANSI C, POSIX)

*rand48 (POSIX)

random (POSIX)

Pseudo-random number generator

rand (ANSI C, POSIX)

*rand48 (POSIX)

random (POSIX)

/dev/[au]random (Linux)

Pseudo-random number generator

rand (ANSI C, POSIX)

*rand48 (POSIX)

random (POSIX)

/dev/[au]random

arc4random (OpenBSD)

Pseudo-random number generator

rand (ANSI C, POSIX)

*rand48 (POSIX)

random (POSIX)

/dev/[au]random

arc4random in Linux! (libbsd)

Pseudo-random number generator

rand (ANSI C, POSIX)

*rand48 (POSIX)

random (POSIX)

/dev/[au]random

arc4random (RC4?)

RC4 security

2001 Fluhrer, Mantin and Shamir attack ¹

2005 Klein attack ²

2013 AlFardan, Bernstein, Paterson, et. al. ³

¹Weaknesses in the Key Scheduling Algorithm of RC4

²Attacks on the RC4 stream cipher

³On the Security of RC4 in TLS and WPA

Pseudo-random number generator

rand (ANSI C, POSIX)

*rand48 (POSIX)

random (POSIX)

/dev/[au]random

arc4random (ChaCha?!)

ChaCha20 stream cipher

<http://cr.yp.to/chacha.html>

Based on Salsa20 (in eSTREAM portfolio)

Used in BLAKE (SHA-3 finalist)

4 cpb on modern x86

128/256 bit key

ChaCha versus Salsa

Improved diffusion

But no performance hit

IETF Crypto Forum Research Group (CFRG) “is confident that the analysis was sufficiently thorough that ChaCha is an acceptable alternative to SALSA-20.”⁴

⁴Synopsis of CFRG discussions on new stream ciphers and MACs for TLS

Pseudo-random number generator

rand (ANSI C, POSIX)

*rand48 (POSIX)

random (POSIX)

/dev/[au]random

arc4random

libottery

Pseudo-random number generator

rand (ANSI C, POSIX)

*rand48 (POSIX)

random (POSIX)

/dev/[au]random

arc4random

goodrandom?

SSL/TLS

SSL/TLS ciphers

| | |
|---------|----------|
| RC4 | SSL 2.0+ |
| AES-CBC | SSL 3.0+ |
| AES-GCM | TLS 1.2 |

SSL/TLS in Chrome

Undocumented option `--cipher-suite-blacklist`

```
0x0004  TLS_RSA_WITH_RC4_128_MD5
0x0005  TLS_RSA_WITH_RC4_128_SHA
0x000a  TLS_RSA_WITH_3DES_EDE_CBC_SHA
0x0032  TLS_DHE_DSS_WITH_AES_128_CBC_SHA
0xc007  TLS_ECDHE_ECDSA_WITH_RC4_128_SHA
0xc011  TLS_ECDHE_RSA_WITH_RC4_128_SHA
```

Values from RFC 2246

AES-GCM

NIST standard of authenticated encryption

AES-CTR + GHASH

NSA Suite B, SSH, TLS, IPsec, MACsec, FC-SP, WiGig

Experimental support in the OpenBSD IPsec stack

AES-NI and CLMUL

Available in Intel Westmere and newer

7 new SSE instructions

Implemented in OpenSSL and OCF

FPU “locks” in the kernel

CBC, CTR, XTS, GCM

ChaCha20-Poly1305 for TLS draft

Google have proposed [draft-agl-tls-chacha20poly1305](#)

ChaCha20-Poly1305 for TLS draft

Google have proposed [draft-agl-tls-chacha20poly1305](#)

E5-2690 2.9GHz

| | |
|-------------------------|----------|
| AES-128-GCM | 131 MB/s |
| AES-128-GCM with AES-NI | 311 MB/s |
| ChaCha20+Poly1305 | 420 MB/s |

Cortex-A9 1.2GHz

| | |
|-------------------|---------|
| AES-128-GCM | 27 MB/s |
| ChaCha20+Poly1305 | 78 MB/s |

ChaCha20-Poly1305 for TLS draft

Google have proposed [draft-agl-tls-chacha20poly1305](#)

E5-2690 2.9GHz

| | |
|-------------------------|----------|
| AES-128-GCM | 131 MB/s |
| AES-128-GCM with AES-NI | 311 MB/s |
| ChaCha20+Poly1305 | 420 MB/s |

Cortex-A9 1.2GHz

| | |
|-------------------|---------|
| AES-128-GCM | 27 MB/s |
| ChaCha20+Poly1305 | 78 MB/s |

Expected in Chrome 32, bug was filed against NSS, Firefox

Salsa20-SHA1 for TLS draft

RedHat et al. [draft-josefsson-salsa20-tls](#)

| Revision | Changes |
|----------|-------------------------------------|
| 01 | Salsa20/12 with 128-bit key removed |
| 02 | UMAC-96 added |
| 03 | UMAC-96 removed |

Poly1305 MAC

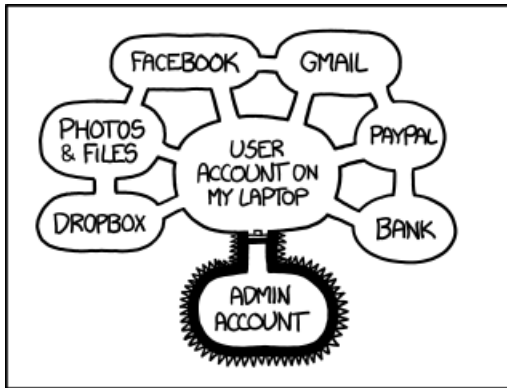
<http://cr.yp.to/mac.html>

“Poly1305 can be written in a tweet” ⁵

About 4 cpb w/o the cipher

Security mainly depends on a chosen cipher
(AES, ChaCha, etc.)

⁵Salsa20 and Poly1305 in TLS



IF SOMEONE STEALS MY LAPTOP WHILE I'M
LOGGED IN, THEY CAN READ MY EMAIL, TAKE MY
MONEY, AND IMPERSONATE ME TO MY FRIENDS,
BUT AT LEAST THEY CAN'T INSTALL
DRIVERS WITHOUT MY PERMISSION.

<http://xkcd.com/1200>

NIST curves

2013 Bernstein, Lange “Security dangers of NIST curves” ⁶

<http://safecurves.cr.yp.to/>

⁶Security dangers of the NIST curves

Curve25519 Diffie-Hellman

<http://cr.yp.to/ecdh.html>

Does not infringe Certicom patents

Executes in constant time

32 byte private and public keys

Ed25519 EdDSA signatures

<http://ed25519.cr.yp.to/>

Comparable to RSA3072, NIST P-256

32 byte private and public keys

64 byte signatures

Uses PRF (SHA512)

NIST-free cryptography in OpenSSH

Support in OpenBSD-current:

| | |
|--------------|----------------------------------|
| Cipher/MAC | chacha20-poly1305@openssh.com |
| Key exchange | curve25519-sha256@libssh.org |
| Public keys | ssh-ed25519-cert-v01@openssh.com |

IPsec/IKEv2 possibilities

It's possible to use "Private Range" in IKEv2

ChaCha20-Poly1305 AEAD for ESP

Questions?

If this story leaves you confused, join the club.

Bruce Schneier