
ORACLE®

/dev/random and Your FIPS 140-2 Validation Can Be Friends

Yes, Really

Valerie Fenwick
Manager, Solaris Cryptographic Technologies team
Oracle
May 19, 2016

Photo by CGP Grey, http://www.cgpgrey.com/ Creative Commons

ORACLE

Copyright © 2016, Oracle and/or its affiliates. All rights reserved.

Not All /dev/random Implementations Are Alike

- Your mileage may vary
  - Even across OS versions
  - Solaris 7‘s /dev/random is nothing like Solaris 11’s
  - Which look nothing like /dev/random in Linux, OpenBSD, MacOS, etc
  - Windows gets you a whole ‘nother ball of wax…
- No common ancestry
  - Other than concept

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

/dev/random vs /dev/urandom

- On most OSes, /dev/urandom is a PRNG (Pseudo-Random Number Generator)
  - In some, so is their /dev/random
- Traditionally, /dev/urandom will never block
  - /dev/random will block
- For fun, on some OSes /dev/urandom is a link to /dev/random

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

FreeBSD: /dev/random

- /dev/urandom is a link to /dev/random
- Only blocks until seeded
- Based on Fortuna

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

OpenBSD: /dev/random

- Called /dev/arandom
- Does not block
- Formerly based on ARCFOUR
  - Now based on ChaCha20
  - C API still named arc4random()

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

MacOS: /dev/random

- /dev/urandom is a link to /dev/random
- 160-bit Yarrow PRNG, uses SHA1 and 3DES

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

Linux: /dev/random

- Blocks when entropy is depleted
- Has a separate non-blocking /dev/urandom

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

Solaris: /dev/random

- Blocks when entropy is depleted
- Has a separate non-blocking /dev/urandom

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

FIPS 140: Deep dive into Solaris and Linux

Let's dig in!

Photo by Tomi Tapio K, Creative Commons

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

Linux: /dev/random Noise Sources

- Disk I/O
- Human Interface Devices (HID)
- Interrupts
- HW RNG
  - Linux 3.16 and newer

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

Linux /dev/random: SP800-90A DRBG?

- Linux has a deterministic random bit generator (DRBG) plugin that's not built into /dev/random
- Available as a kernel plugin as of Linux kernel 3.17
  - Also in libgcrypt as of Feb 2016
  - http://www.chronox.de/drbg.html
- Ideally, a future project would hook the DRBG code into /dev/random.

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

Solaris /dev/random: Entropy Sources

- Actually gets all of its random bits from the Solaris Kernel Cryptographic Framework….

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

Solaris Kernel Cryptographic Framework: Entropy Sources

- swrand
  - Detecting changes in blocks of physical memory
  - Time it takes to detect if changes occurred or did not occur
  - Seeded with high resolution time since boot and current time of day, initial state of physical memory and the number of blocks & above sources.
- intelrd (Intel only)
  - RDRAND and RDSEED
- n2rng (Oracle SPARC only)
  - Hardware-based entropy source

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

Solaris /dev/random: SP800-90A DRBG

- Solaris kCF uses Hash_DRBG with SHA512
- Additional DRBG in the Userland Cryptographic Framework
  - PKCS#11 stack
  - ucrypto

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

Therefore…

- Solaris's /dev/random is SP800-90A compliant
  - Security strength of 256
  - Prediction Resistance enabled
- Can be used by any consumer as a DRBG

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

Other APIs

All the choices!

Photo by Daniel Dionne, Creative Commons

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

arc4random()

- Generally ***not*** implemented with ARCFOUR anymore
- Available on Solaris, OpenBSD, FreeBSD, NetBSD
- What's inside varies, most not using DRBG at this time
- But…

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

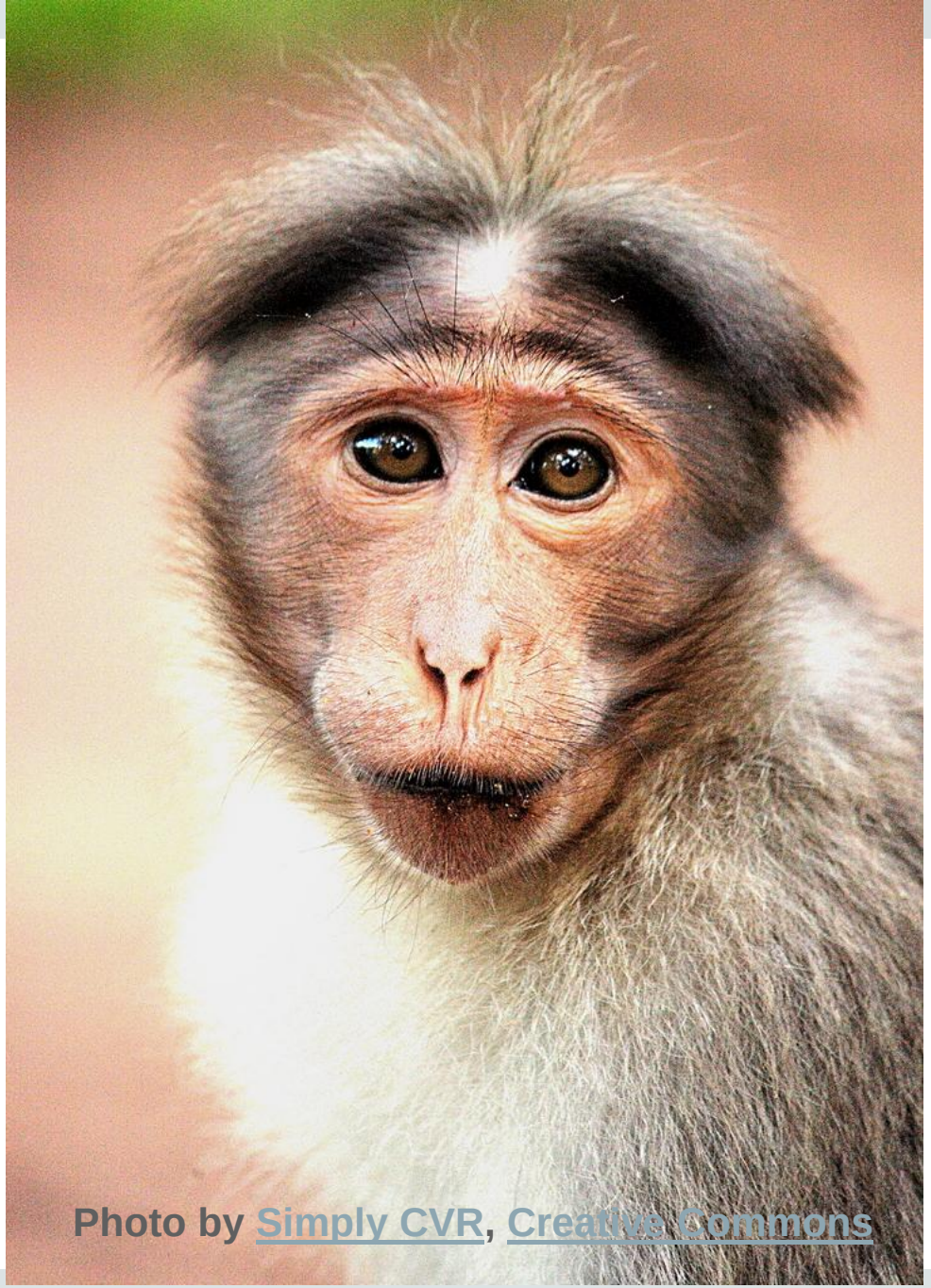
getrandom() system call

- Available in Solaris and Linux
- In Solaris this API leverages SP800-90A DRBG
  - In the future, Linux can and should do the same
- Not to be confused with getentropy()
  - getentropy() returns raw entropy – not mixed.
- arc4random() should simply call getrandom()
  - Problem solved!

Copyright © 2016, Oracle and/or its affiliates. All rights reserved

Same Name, Different Meanings

- /dev/random on one system could be SP800-90A compliant
  - Might even have a validation certificate!
- But, not on the next…



Copyright © 2016, Oracle and/or its affiliates. All rights reserved

References

- Stephan Mueller, ICMC 2015, “SP800-90B: Analysis of Linux /dev/random”
- Darren Moffat’s blog:
  - https://blogs.oracle.com/darren/entry/solaris_random_number_generation
  - https://blogs.oracle.com/darren/entry/solaris_new_system_calls_getentropy
- Krishna Yenduri’s Solaris RNG history: https://blogs.oracle.com/yenduri/entry/dev_random_in_solaris
- Wikipedia /dev/random article: https://en.wikipedia.org/wiki//dev/random
- Future work on Linux’s /dev/urandom: https://lwn.net/Articles/686033/

Copyright © 2016, Oracle and/or its affiliates. All rights reserved