Don't Play Dice With Random Numbers

H. Peter Anvin, Intel Open Source Technology Center
LinuxCon Europe 2012
Barcelona, Spain
Random numbers

• Random numbers are used in...
  – Games
  – Monte Carlo simulations
  – Security protocols

• Computers are not very random
  – Lots of effort goes into eliminating random behavior...

• “Good enough” randomness depends on the application
  – Security protocols have very high demands
  – Games usually not so much...
Randomness is subtle

• Improper use
  – A random number is only random once
  – Only random until the outcome is known

• There are no tests for randomness!
  – There are tests for *some types* of nonrandomness
  – General testing for randomness might be intractable \((P = \text{BPP} \text{ conjecture})\)
  – Need to understand the failure modes of the source
What could possibly go wrong?

- **Weak keys**
  - Several serious vulnerabilities in Linux distros already
- **Key disclosure**
  - Recent PS3 hack
- **Identifier collisions**
  - UUIDs are probabilistically unique
- ...
Pseudo-Random Number Generator

- Statistical properties
- Cycle length
- Resistance to analysis (“security”)
“God doesn't play dice.”

— Albert Einstein

“Wanna bet?”

— God
Hardware (true) Random Number Generator

Quantum events

Entropy source → Conditioner → CSPRNG → 83, 11, 78, ...

Integrity monitor → ERROR!

• Bandwidth
• Resistance to observation ("security")
• Failure modes
Intel Bull Mountain Technology (DRNG)
Linux Kernel Random Number Generator

Pool  →  CSPRNG

Pool  →  CSPRNG

Pool  →  CSPRNG

Arch hwrng

New in 3.6

Throttle

/dev/random

/dev/urandom

Kernel users
Linux Kernel Random Number Generator Inputs

- hwrng
- virtio
- TPM (new in 3.7)
- Arch hwrng

hwrng driver

Event timings

rngd (user space)

FIPS tests
rngd

- Necessary to get full benefit from a hardware or virtio RNG
- *Should be started as early as possible*
- Versions < 4 had significant problems
  - Hopefully all fixed now
- TPM harvesting conflicts with TrouSerS unless `rng-tpm` is available
  - Upstream in 3.7, probably an easy backport
  - TPM may need to be “provisioned”
  - If you don't need TrouSerS, don't run `tcsd`
rngd -r /dev/urandom
HAVEGE

• Claims to extract entropy from CPU indeterminism
• Some people swear by it...
• Unclear to what extent it actually works
  – “The source is so complex it is impossible to analyze”
  – Self-tests pass even with the timer readout removed
• It probably does provide some entropy
  – Consider to what degree you are willing to trust it
• Can be run in parallel with rngd
Administrator recommendations

• **Make sure that rngd is running**
  – Version 4 or higher strongly recommended
  – If not by default, please complain to your distribution
  – Run as early as possible
    • Avoid zero-entropy situation on boot

• **Make sure TPM is available**
  – May have to be provisioned
  – If you don’t need TrouSerS, don’t run `tcsd`

• **haveged** can be a complement, but not an alternative
  – Consider how much you trust it...
Application writer recommendations

• If you need *lots of randomness*:  
  – Use a cryptographic library (OpenSSL, etc.)  
  – A simple `librandom` may be available in the future

• If you need *a little randomness*:  
  – Use `/dev/random` if you would rather fail than be insecure  
  – Use `/dev/urandom` if you need “good enough for most things”

• Please conserve randomness  
  – Not everyone has a hardware random source yet...  
  – Don't use buffered I/O unless you really need to!

• Defer extraction as much as possible (especially daemons)  
  – Entropy may be scarce at boot
Future work

• Policy interface
  – Allow rngd bypass and possibly direct use of architectural hwrng
  – Discussed in principle at Kernel Summit 2012
  – Still being architected

• Finish virtio-rng
  – Kernel (guest) side complete since 2008
  – Host (Qemu/KVM) side still in progress
    • Got stalled several times
    • Hopefully will get committed to Qemu git this week or next
Copyright acknowledgments

  - © 2012 Darren Hester, Creative Commons Attribution license

- **Lava lamp**, http://www.flickr.com/photos/skyfaller/111857525/
  - © 2006 Nelson Pavlosky, Creative Commons Attribution – Share Alike license

- **Diagram of Intel DRNG Entropy Source**
  - © 2011 IEEE Spectrum