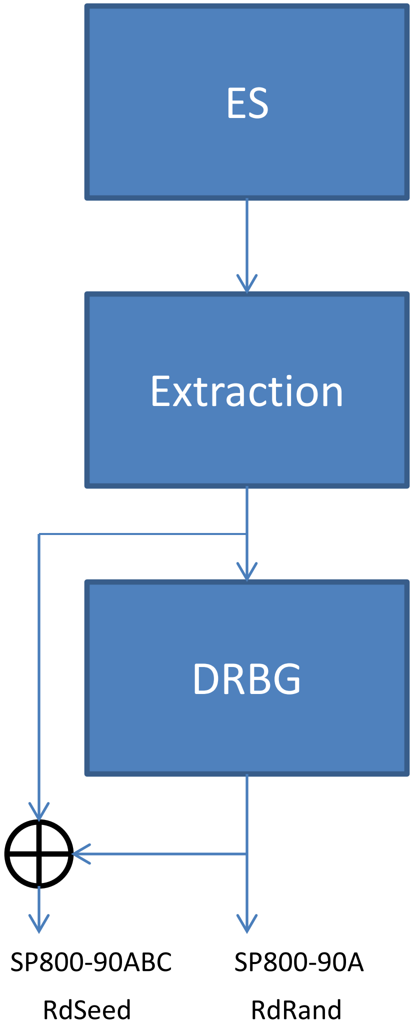
Intel and Random Numbers

David Johnston
dj.johnston@intel.com

NIST Random Number Workshop

Inconvenient Truths

(about building crypto-secure random number sources)



ES

Extraction

DRBG

SP800-90ABC

RdSeed

SP800-90A

RdRand

Entropy Sources electronic circuits coupling noise to binary data. Fast ones (3+ Gbps) are bit serial. Feedback is necessary for manufacturing robustness.

This implies all real-world OTS sources are biased, correlated and non stationary.

Hard/impossible to show independence of sources. Impossible to source perfect helper data → so no universal hash functions or 3 input extractors. So we are use LHL and CBC extraction proofs that use (relatively) expensive functions (like AES) where smaller, lower power functions might otherwise suffice. See SP800-90B for examples.

Lack of AES symmetry (except at 128 bits) make 256 bit wide DRBGs messy to use, inefficient and hard to export. MDC-2 is not an option.
The hash-drbg is messy in hardware. Non power-of-two data widths.
XOR Construction is pointless. Implies OHT is not effective. Bounding with reseeds is a scheduling nightmare when sharing crypto block function.

SP800-90 Looks like a software spec:

Instantiation : Firing doping atoms into a silicon substrate
Authentication : Built on the same die. Key distribution is silly.

What is Changing?

(in Intel products, that is pertinent to the problem to hand)

Basic cryptographic elements need to move:

1) To hardware.
    1) Contstant time
    2) Limited side channels
    3) Cheap and fast to software at all levels
2) To non device models (like instructions)
    1) Smaller attack surface
    2) Bypsses OS, drivers, APIs and layers of bugs
    3) Works in VMs.

Intel is doing this:

AES-NI
RdRand
RdSeed
…

(tools you can use approach)

SHA-3 ? Poor scalability makes Keccak a problem.

Features for key management a WIP.

Wouldn't it be nice if SP800-90 and FIPS-140:

- Explicitly permit conformant RNGs as output only devices
- Define entropy quality statistically (because that is all the physical world gives you)
- Expect conformant implementations to make statistical entropy quality and availability claims (because that is all the physical world gives you)
- Allow construction on the same die to imply implicit authentication.
    - Establishing and distributing keys on die to authenticate access to an on die RNG is a chicken and egg problem.