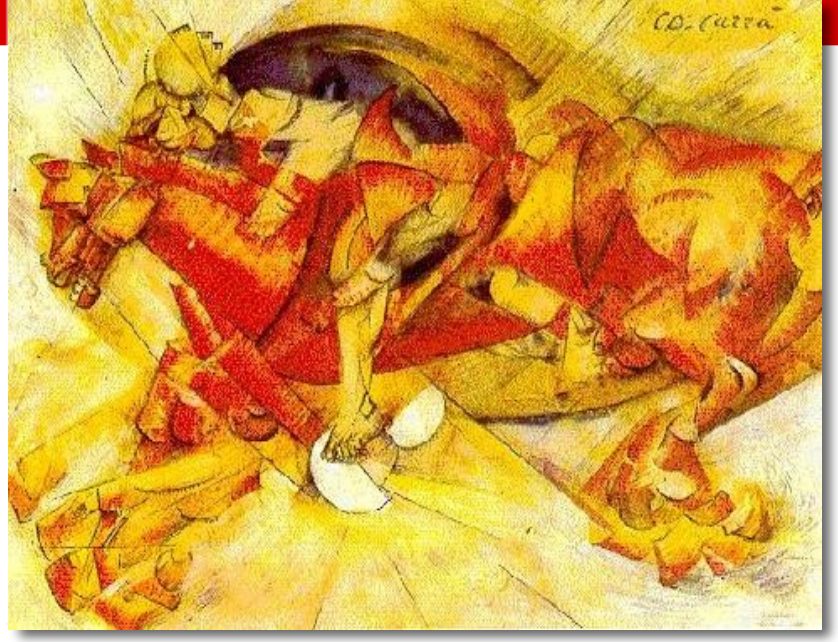
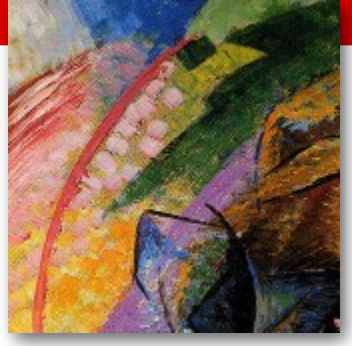


Linux /dev/random
BSI-Studie zur Entropiebewertung

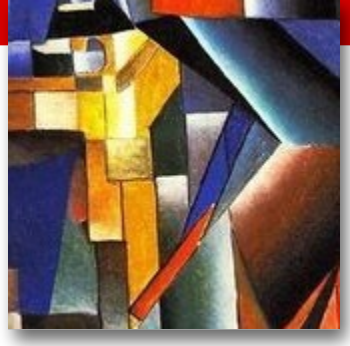
Stephan Müller <stephan.mueller@atsec.com>



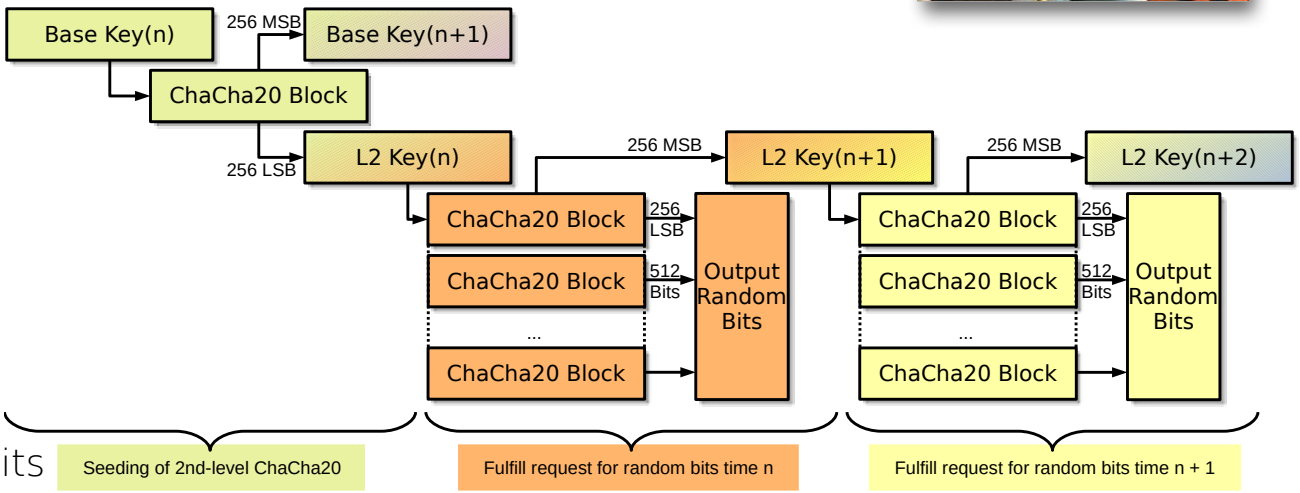
Agenda

- Architektur
- Rauschquellen
- AIS20/31 (2011)
- AIS20/31 (2023 Entwurf)

Linux-RNG Architektur DRNG



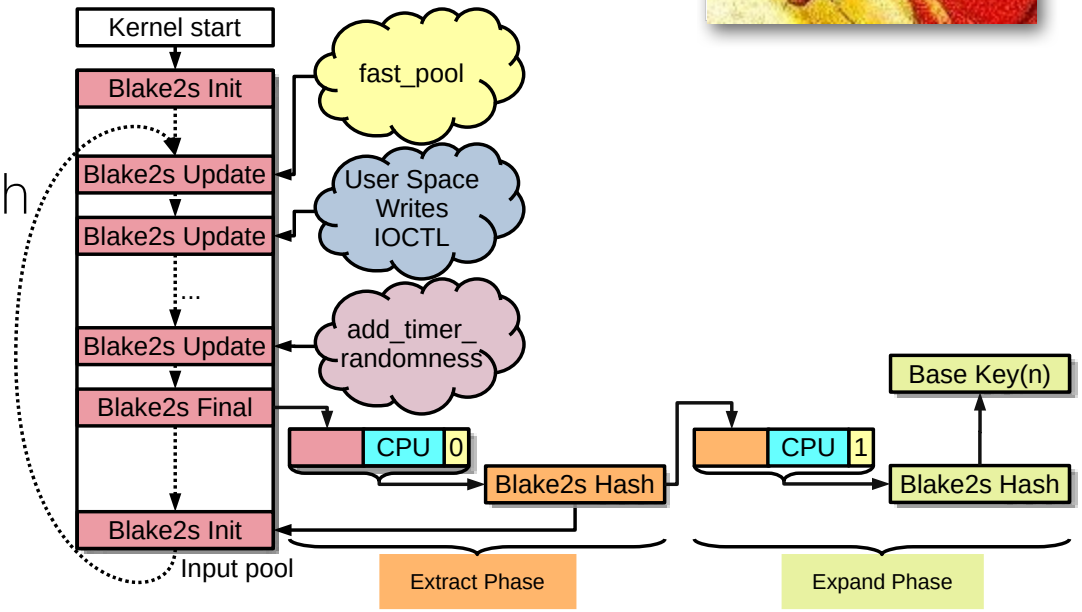
- Zustandsvariable:
  - 256-Bit Schlüssel
- Fast Key Erasure Konzept:
  - Instantiierung ChaCha20 Block Algo mit Schlüssel
  - Generierung ChaCha20 Block 512 Bits
    - 256 MSB → neue Zustandsvariable
    - 256 LSB → Zufallszahl



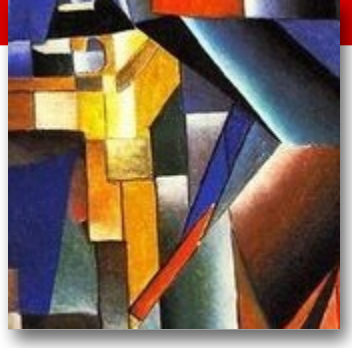
- Zufallszahl entspricht Schlüsselstrom der ChaCha20 Stromchiffre
- Ein primärer DRNG, pro-CPU sekundäre DRNG Instanzen
  - Primärer DRNG erhält Seed-Data vom Entropiepool
  - Sekundäre DRNGs erhalten Seed-Daten vom primären DRNG
- Keine Starttests oder anderweitige Tests

Linux-RNG
Architektur Entropiepool

- Blake2s Zustand ist Entropiepool
- Zufuhr von Daten via Hash Update
- Extraktion:
  - Hash Final → Hashwert
  - KDF Extraktion mit Hashwert, RDSEED, counter → Hashwert initialisiert neuen Entropiepoolzustand
- Expansionsfunktion erzeugt neuen primären ChaCha20 Schlüssel
- Keine Starttests oder anderweitige Tests

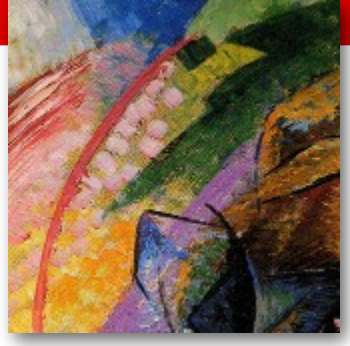


Linux-RNG
Rauschquelle: Interrupts

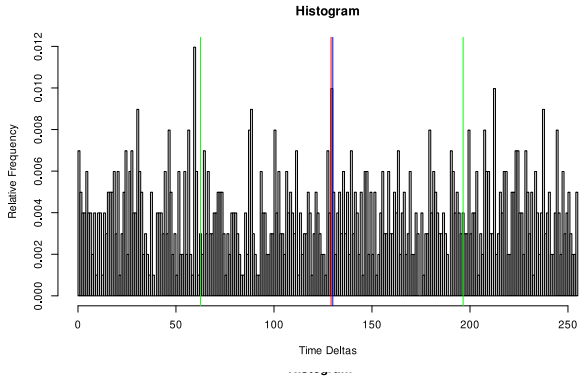


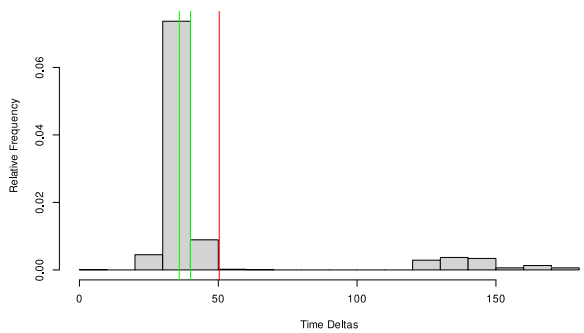
- Haupt-Rauschquelle
- 64-Bit Zeitstempel wird pro Interrupt als roher Entropiewert gespeichert und evtl. zum Blake2s hinzugefügt
- Keine Laufzeittests
- Keine Testschnittstellen zur Entropiebewertung:
  - Erstellung eines Kernel-Patches zur Extraktion roher Entropiedaten
  - Massive Unterbewertung vorhandener Entropie
  - Auswirkung der initialen Verarbeitung der Rohdaten auf Entropie unklar
- Durchführung von Analysen auf Hardware und in virtuellen Umgebungen

Linux-RNG
Rauschquelle: Scheduler



- Optionale Rauschquelle während Kern-Boot
- 64-Bit Zeitstempel wird pro Kontextwechsel eines speziellen Kernel-Threads zum Blake2s Zustand hinzugefügt
- Keine Laufzeittests
- Keine Testschnittstellen zur Entropiebewertung:
  - Erstellung eines Kernel-Patches zur Extraktion roher Entropiedaten
  - SP800-90B Spaltenweise Entropierate « Zeilenweise Entropierate → Problematische Rauschquelle

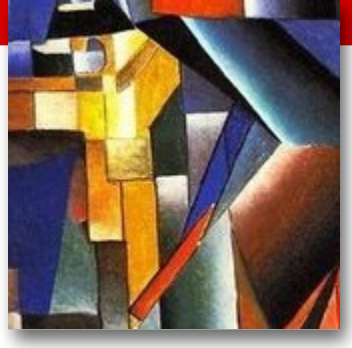




AIS20/31 (2011)

- NTG.1:
  - Jegliche Konsistenz ging mit Kern v5.18 verloren
  - Änderungen wurden in LTS-Kerne zurückportiert

    → NTG.1 Eigenschaft für “aktuelle” Kerne nicht mehr gegeben
- DRG.3:
  - Konsistenz wird im Bericht analysiert und bestätigt



AIS20/31 (2023 Entwurf)

- NTG.1 Anforderung:
  - Initialisierung des DRNGs mit 220 Bit Entropie jeweils aus 2 verschiedenen Rauschquellen
  - DRNG erzeugt Zufallsdaten nach erfolgreicher Initialisierung „frei-laufend“ – keine Blockierung mehr
- Linux-RNG hat keine Konsistenz mit NTG.1
  - Eine bewertete Rauschquelle: Interrupts
  - Eine als nicht ausreichend bewertete Rauschquelle: Scheduler
  - Eine CPU-basierte Rauschquelle ohne Entropieanalyse: RDSEED

  → Anforderung an Existenz von 2 Rauschquellen verletzt
- Verschiedene Lösungen zur NTG.1 Konsistenz wurden veröffentlicht

Vielen Dank!
Fragen?