Overview of AIS 20/31

Werner Schindler
Bundesamt für Sicherheit in der Informationstechnik (BSI)
Bonn, Germany

Random Bit Generation Workshop 2023

May 31, 2023
AIS 20 and AIS 31

- AIS 20 and AIS 31
  - are evaluation guidelines for RNGs for cryptographic applications.
  - have been effective in the German certification scheme (Common Criteria) since 1999, resp. since 2001.
  - both refer to a joint mathematical-technical reference
    - for short usually also called AIS 20, AIS 31, or AIS 20/31 (depending on the context).
    - We follow this convention.

- The current version of the mathematical-technical reference has been effective since 2011.
New AIS 20/31

- The mathematical-technical reference AIS 20/31 is currently being updated.
- September 2022: Draft version 2.35 published (co-authored by Matthias Peter and Werner Schindler)
- Comment period ended on February 15, 2023. Thank you for all comments!
- Intermediate document (version 2.36, considers numerous comments) will be online by Friday: https://www.bsi.bund.de/dok/randomnumbertengenerators
- A hybrid workshop will be held next week in Bonn (June 5 - June 7).
BSI and NIST have been in an ongoing process of harmonizing AIS 20/31 and SP 800-90[A,B,C].

BSI and NIST: joint presentations at ICMC 2021, ICMC 2022, and ICMC 2023.

Currently, BSI and NIST are working on a joint document that compares central features of AIS 20/31 and SP 800-90[A,B,C].

- explains similarities and differences between functionality classes (AIS 20/31) and RBG constructions (SP 800-90 series)
- More information is provided in the following presentation
  Kerry McKay: Bridging the Gap Between the SP 800-90 Series and AIS 20/31
AIS 20/31: Basic Philosophy

- The AIS 20 and the AIS 31 are technology neutral.
- The AIS 20 and the AIS 31 do not specify approved designs.
- Instead, functionality classes are defined.
  - Security requirements are specified that RNGs shall fulfil in order to comply.
  - The applicant for a certificate (usually the developer) and an accredited evaluation lab have to give evidence that the RNG meets the class-specific requirements.
Classification of RNGs

- **DRNGs** deterministic RNGs
  - the random numbers depend on
    - the seed,
    - (optional): + on reseeding + additional input

- **PTRNGs** physical true RNGs (short: physical RNGs)
  - physical noise source
    - exploits physical phenomena from dedicated hardware designs or physical experiments

- **NPTRNGs** non-physical true RNGs
  - non-physical noise source
    - no dedicated hardware design
    - typically, exploits system data (timing values, RAM data, etc.) or user’s interaction (mouse movement etc.)
New AIS 20/31: Functionality classes

DRNGs

DRG.2

DRG.3

DRG.4

PTG.2

PTG.3

NTG.1

Deterministic RNGs

Physical RNGs

Non-Physical True RNGs

→ Increasing requirements
DRNG: functionality classes

- The functionality classes again ensure
  - DRG.2: backward secrecy and forward secrecy
  - DRG.3: + enhanced backward secrecy
  - DRG.4: + enhanced forward secrecy (requires hybrid DRNGs)

- Comparison with SP 800-90: Rough correspondences
  - enhanced backward secrecy $\cong$ backtracking resistance
  - enhanced forward secrecy $\cong$ prediction resistance
DRNGs: Important new features

- The notion of **request** has been introduced.
- The notion of **effective internal state** has been introduced.
- **Under suitable conditions** ‘DRNG seeding DRNG’ is permitted.
Effective internal state

- Part of the internal state of a DRNG that an adversary does not know and cannot determine or guess (with significantly larger probability than for blind guessing) even if the adversary has seen many random numbers.

- **Entropy** (effective internal state) after (re-)seeding / after additional high-entropy input: \( \geq 240 \text{ bits min-entropy} \)
  - Under suitable conditions, alternatively \( \geq 250 \text{ bits Shannon entropy} \) can be claimed.

- **Goal**: Shall prevent multi-target attacks and attacks by quantum computers (Grover’s algorithm).
DRNG seeding DRNG: seed tree

- Usually, DRNGs are seeded by true RNGs (recommended, if possible)
- **There are scenarios in which no true RNG is available.**
  Example: The DRNG of the operating system has been seeded by an NPTRNG, and the applications call this DRNG for seed material to seed their own DRNGs.
- When using seed material from another DRNG additional problems have to be considered.
- For all (possible) reseeding procedures each DRNG must use the same seeding DRNG as for the seeding procedure.
  - This requirement prevents seed cycles!
  - This implicitly defines a seed tree. Its root (‘root DRNG’) is seeded by a true RNG.
- The requirements of BSI and NIST are rather similar.
DRNG seeding DRNG: seed tree
90A-compliant DRBGs: Conformity proofs

- AIS 20/31 contains a conformity proof for the Hash_DRBG to class DRG.3 (algorithmic properties).
- AIS 20/31 contains a conformity proof for the HMAC_DRBG to class DRG.3 (algorithmic properties).
  - For the central points of the proof AIS 20/31 refers to a paper of John Kelsey (NIST); paper to be published
- Applicants for certificates can simply refer to these conformity proofs.
The stochastic model is the ‘core’ of each PTRNG evaluation (PTG.2, PTG.3)

Random numbers are interpreted as realizations of random variables.

Aim: Verification of a lower entropy bound per internal random bit (= random bits after postprocessing)
Stochastic model (II)

- The raw random numbers shall be (time-locally) stationarily distributed.
  - Slow drifts of the parameters are permitted as long as the entropy remains sufficiently large.
- Ideally, the stochastic model specifies a class of distributions that contains the (unknown) true distribution of the (usual case) raw random numbers during the lifetime of the PTRNG.
Stochastic model (III):
Toy example in a nutshell

- A coin is tossed $N$ times; ‘1’ $\sim$ ‘head’ and ‘0’ $\sim$ ‘tail’
  - outcome: $x_1, \ldots, x_N \in \{0, 1\}$
- $x_1, \ldots, x_N \sim$ realizations of random variables $X_1, \ldots, X_N$.
  - Coins have no memory.
  - $\implies X_1, \ldots, X_N$ may be assumed to be independent and identically $B(1, p)$-distributed (Bernoulli distribution)
  - parameter $p := \text{Prob}(X_j = 1)$ is unknown

Stochastic model: $X_1, \ldots, X_N$ are independent and identically $B(1, p)$-distributed with $p \in [0, 1]$.

- The stochastic model fits to other coins, too, and would tolerate drifts of $p$ for the same coin in course of time.
- Estimate $p$ on the basis of $x_1, \ldots, x_n$
- Substitute its estimate $\tilde{p}$ into the (1-dimensional) entropy formula.
The applicant has to give evidence that the stochastic model fits to the physical noise source (includes digitization).

- The stochastic model shall be based on the understanding of the noise source.
- The argumentation should be supported by engineering or physical arguments, by findings from the literature, by tests on empirical data etc.
- AIS 20/31 discusses in detail several stochastic models of real-world physical noise sources.

Presentation: Johannes Mittmann (BSI): Use of stochastic models in RBG standards
Online test and total failure test

- The online test shall detect non-tolerable weaknesses (sufficiently) soon.
  - The online test shall be tailored to the stochastic model.
- The total failure test shall detect total failures of the noise source very fast.
  - The output of weak random numbers must be prevented.
  - The justification shall be supported by engineering arguments (failure analysis).
PTRNG: Functionality class PTG.2

- ‘Pure’ PTRNG
  - algorithmic postprocessing (e.g., XOR)
  - ‘no postprocessing’ and cryptographic postprocessing are also permitted
- raw random numbers:
  - time-locally stationarily distributed
- Entropy (one or both claims are possible [selection])
  - Shannon entropy / output bit $\geq 0.9998$
  - Min-entropy / output bit $\geq 0.98$
- Effective online test and total failure test, startup test
PTRNG: Functionality class PTG.3

- Physical RNG with
  - strong, well-understood physical noise source
  - effective online test and total failure test, startup test
  - cryptographic postprocessing with memory
    - If the postprocessing algorithm runs autonomously it can be viewed as a DRG.3-compliant DRNG.
PTG.3: typical design

The evaluation can be divided into two separate steps:

- **PTG.2-compliance** of the ‘inner’ PTRNG
- **PTG.3-compliance** of the entire RNG

(possibly at a later date, with another applicant)

Different companies can be involved in these evaluations.
PTG.3: Entropy claim

- The applicant (developer) can apply for Shannon entropy, for min-entropy, or for both [selection].
- (as before) The input rate of the cryptographic postprocessing is $\geq$ than its output rate.
- Furthermore, now individual entropy claims are possible (requires data compression!).
  - (Shannon entropy) claim $\nu_S \in [0.9998, 1 - 2^{-32}]$
  - (min-entropy) claim $\nu_M \in [0.98, 1 - 2^{-32}]$
- Important special cases are treated in AIS 20/31. Explicit formulae are provided.
PTG.3: Example (entropy claim)

PTG.2-compliant RNG generates the intermediate bits. Assumptions:
- min-entropy $\geq 0.98$ per bit
- output size (cryptographic postprocessing): 256 bits
- The cryptographic postprocessing can be modelled by a random mapping (depends on the algorithm!)

Then: input size (intermediate random numbers) $\geq 256 + 64 + 7 = 327$ bits
$\rightarrow$ min-entropy / output bit $\geq 1 - 2^{-32}$. 

- PTG.2, PTG.3, NTG.1
- AIS 20/31 Workshop
NPTTRNGs

- Main differences to PTRNGs
  - problem: designer / evaluator cannot control the environment where the NPTTRNG is operated (typically run on PCs, servers, etc.)
  - usually does not allow precise stochastic modeling
  - instead: conservative entropy estimates and a large data compression rate
    - goal: derive a trustworthy lower entropy bound under conservative (realistic) assumptions on the noise source and the abilities of potential attackers
NTG.1

- min-entropy claim $v_M \in [0.98, 1 - 2^{-32}]$ per output bit
- No random numbers are output until at least two different noise sources have provided 220 bits min-entropy. The two noise sources shall employ different principles to provide randomness.
Overview of AIS 20/31

Schindler

Introduction

Functionality classes

DRNGs

Stochastic model

Online test, total failure test

PTG.2, PTG.3, NTG.1

AIS 20/31 Workshop

New AIS 20/31
(Structure of the document)

1. Introduction
2. AIS 20 and AIS 31 — scope, limits, and concepts [informative]
3. Functionality classes [normative]
   class requirements, application notes, general explanations
4. Mathematical Background [mainly informative]
5. Examples [mainly informative]
6. Glossary [normative]
AIS 20/31 Workshop

- **Date:** Monday, June 5 - Wednesday, June 7
  (2:30 pm - 7:00 pm CEST (‘German time’))
- **Type:** hybrid (physical, virtual)
- **Registration:** by e-mail: ais-20-31@bsi.bund.de
- **Registration deadline:** extended to June 2, 2023
- participation is free of charge
- We are glad about your participation.
Contact

Bundesamt für Sicherheit in der Informationstechnik (BSI),
Bonn, Germany

Werner Schindler
P.O. Box 200363, 53133 Bonn, Germany
Tel.: +49 (0)228-9582-5652
Werner.Schindler@bsi.bund.de
https://www.bsi.bund.de