RNG with compromise recovery

Homework VIII.

PA193 – Secure coding

Marek Sýs
Faculty of Informatics, Masaryk University, Brno, CZ

CROCS
Centre for Research on
Cryptography and Security

Assignment 8: recoverable RNG

Design and implement your own secure RNG:

1. RNG provides method **generateData(byte[] buffer, int length);** which will fill buffer with required amount of pseudorandom data (length paramater)
2. RNG should be capable to recover from compromise of its internal state by an attacker. After recovery, should not be able to predict pseudorandom data produced by RNG.
3. RNG should recover fast but without blocking.
4. Test output of your RNG with NIST STS, Dieharder or TestU01 battery.

What to submit

- What to submit:
  - Your program (*.c, *.cpp, *.java, *.py,…)
  - Results.txt – results of randomness testing
  - Text description of your program, interpretation of results and RNG characteristics (recovery, speed, security). Discuss the properties of your recovery mechanism especially speed and security.
- When and where to submit
  - Submit before 28.11.2019 23:59 into IS HW vault
  - Soft deadline: -1.5 points for every started 24 hours