RNG Summary

1. Szrek2Solutions' randomizer uses a combination of hardware and software methods.
2. It runs on Windows platform.
3. NIST certified tamper evident Hardware Security Module (HSM) generates the seed for the random numbers.
4. The seed is a RSA signature. Following are the features of the seed:
    - It does not have any cycle.
    - It is not predictable or repeatable. There are 3 elements that enforce unpredictability and un-repeatability:
        - Signing key (protected by HSM) is not readable outside of HSM and cannot be retrieved or deduced. RSA digital signatures without the knowledge of the private keys cannot be predicted. There is almost 40 years of literature analyzing it. RSA signatures is a digital signature standard in the US and Canada. Use of digital signature as RNG seed enforces unpredictability.
        - Any time the data is passed to be signed (to generate the seed), HSM appends to it internally controlled sequencer and RTC (Real Time Clock). Consequently the data that will be signed will be always different. This enforces non-repeatability of the signed data.
    - The seed is verified in real time, after the generation. After the digital signature is generated, it is verified using public key corresponding to private key. If the signature fails, the software will automatically switch to use another HSM. In any case no random number will be generated if the signature (hardware) fails. Verification is done using a different crypto provider (software from a different provider).
    - The seed can be verified any time after the generation to prove integrity of random data.
    - Every seed generation can be accounted for. Use of sequencer controlled by HSM allows accounting for every seed generation.
    - Seed generation time can be verified. RTC is signed and seed generation time is verifiable.
5. Algorithm:
    - One-way hash iterations are used to reduce number of bits without loosing entropy and to produce the raw data used for actual random generation. This data can be combined with some other sources of entropy without loosing audit capability.
    - It is ensured that no bias is introduced during the scaling process.

RNG Methodology

Szrek2Solutions' RNG (Random Number Generator) generates random outcomes for different types of games of chance with designed in ability to audit the generated numbers and time of generation. The key element of generating random number is the

use of digital signatures. A cryptographic processor[1] signs application data. Since it doesn't disclose the signing (private) key, signature itself is unpredictable and can be treated as a source of randomness. Corresponding public key allows for verification of generated random numbers.
Following sample types of random outcomes are supported:

- **Integer** random number M through N is generated with and without replacement for games like **Numbers (3-digit, 4-digit), Joker, Kicker, Bingo, Lotto, Keno, Raffle** etc. Currently supported maximum is 2 billion. Higher ranges can be supported if needed.
- **Table based distribution** for **card games, Bingo, instant winners betting**, etc. Table based distribution is a very general mechanism to generate different distributions of random numbers. This approach allows creation of outcomes for games with desired bias e.g. winning categories for instant winners betting. Random numbers are generated with or without replacement. This approach is very useful for games with strict pools liability.
- **Draw from file with transactions** – variety of methods are supported: draw from multiple files, support for a single winner, weighted odds for different entries, sorting, handling user defined fields, etc. Depending on the type of generation, support for 10,000,000 to 2 billion entries is provided. Used for second chance, raffle, and other games.

A few main advantages of using this approach are:

- **Ability to audit** random numbers created using this method by verification of the digital signature, using verification (public) key, and verifying random numbers recreated from the digital signatures.
- **Inability to attack** random number generation process without detection. The only secret element (private key used to generate digital signatures) is protected by tamper evident HSM. Signature sequencer and Real Time Clock are also protected by HSM.
- **Signing Real Time Clock** (RTC) enables verification of the time of random numbers generation.
- **Signing sequencer** enables accounting for every single random number generated

[1] Crypto processor is the cryptographic device that allows performing various cryptographic operations. For this application crypto processor has the following features:

- RSA signatures
- Private/public key pair is generated in a way that private key is never accessible outside the crypto processor.
- During the key generation crypto processor initializes a sequencer counter that is kept in the non-volatile memory. Sequencer is incremented only when a new signature is generated and reset when new pair of keys is created. There is no other way to modify sequencer value.
- Crypto processor has a battery backed up real time clock. This clock value, at the time of signing, is also signed, so the time of the generation of random number can be also audited.