# Sheffield Ethical Student Hackers

Exploiting Randomness

# The Legal Bit

- The skills taught in these sessions allow identification and exploitation of security vulnerabilities in systems. We strive to give you a place to practice legally, and can point you to other places to practice. These skills should not be used on systems where you do not have explicit permission from the owner of the system. It is VERY easy to end up in breach of relevant laws, and we can accept no responsibility for anything you do with the skills learnt here.

- If we have reason to believe that you are utilising these skills against systems where you are not authorised you will be banned from our events, and if necessary the relevant authorities will be alerted.

- Remember, if you have any doubts as to if something is legal or authorised, just don't do it until you are able to confirm you are allowed to.

- Relevant UK Law: https://www.legislation.gov.uk/ukpga/1990/18/contents

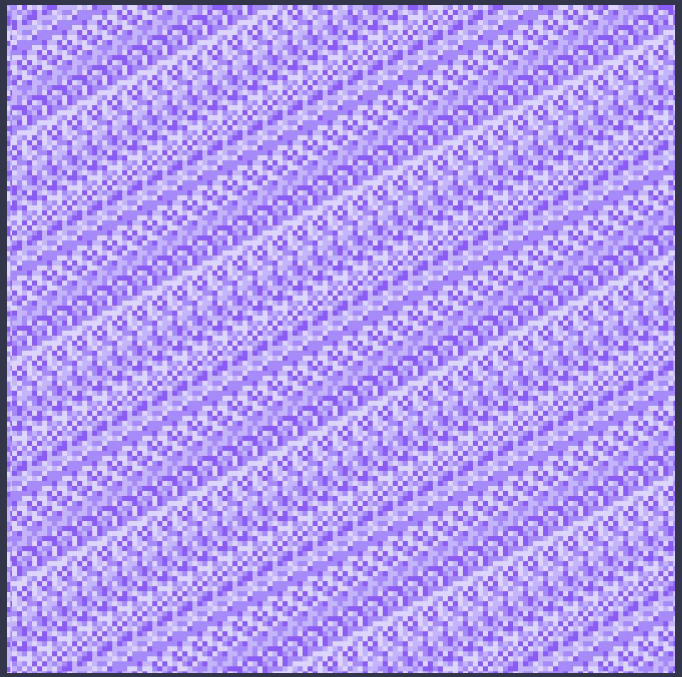
# Code of Conduct

- Before proceeding past this point you must read and agree to our Code of Conduct - this is a requirement from the University for us to operate as a society.
- If you have any doubts or need anything clarified, please ask a member of the committee.
- Breaching the Code of Conduct = immediate ejection and further consequences.
- Code of Conduct can be found at https://shefesh.com/downloads/SESH%20Code%20of%20Conduct.pdf

# What is Randomness?

- Computers do not operate with Randomness. We have to compute a number with an algorithm to create a Pseudo-random number.
- These algorithms or ‘Random’ Number functions is just a sequence of math operations.
- If randomness is a function, it might be possible to reverse it

# Pseudo-Random Number Generators (PRNG)



- PRNG Algorithms like Linear Congruential Generator (LCG) and Mersenne Twister (MT) generate random numbers from a seed.
- LCG: $X_{n+1} = (aX_n + c) \mod m$
  - $m$ (modulus): Determines the maximum possible sequence length (period).
  - $a$ (multipier): A crucial factor for generating a long, non-repeating period.
  - $c$ (Increment): Must be non-zero for a "mixed LCG" to potentially achieve a full period.
  - Poor choices for the parameters $a$ and $c$ will result in an exploitable, very short repeating sequence.

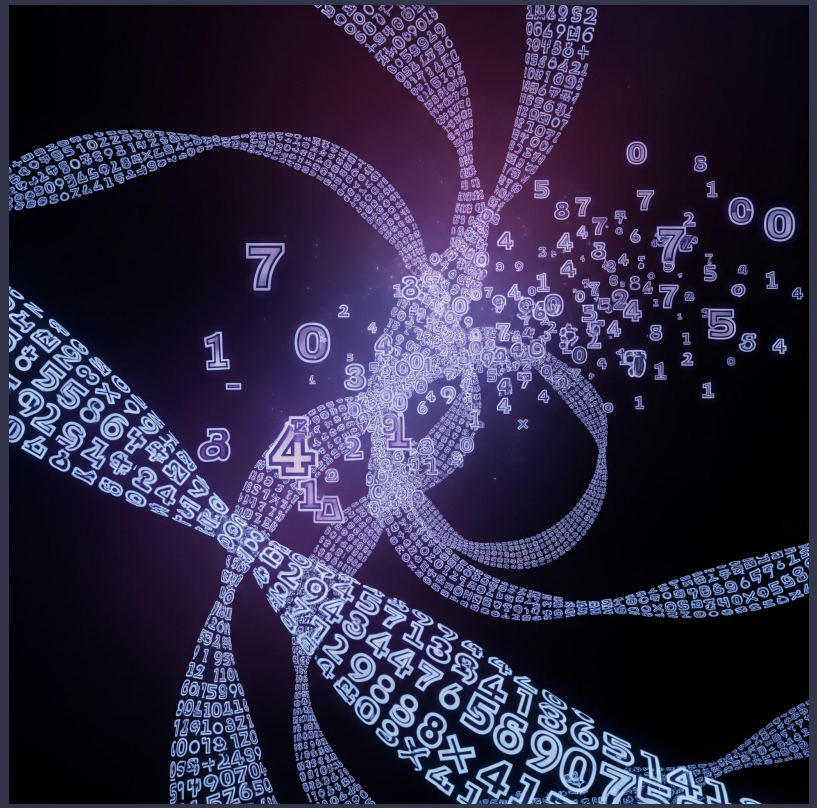
# Impact of Cracking RNG

Being able to predict the next output that a particular algorithm will provide either by manipulating the state the system is in can have many consequences.

- 2FA
- Session tokens
- Password reset tokens
- Game RNG
- Lottery-style/Casino systems

# Mersenne Twister PRNG

- Mersenne Twister is a widely used Pseudorandom Number Generator.
- It has a very long period and good uniformity properties.
- The internal state of the generator can be determined.
- Knowledge of the state allows prediction of all future outputs.



# Handout

- MT19937 maintains an internal state of **624 32-bit integers**.
- Each output is produced by applying a reversible "tempering" function.
  If an attacker observes **624 consecutive outputs**, they can:
  - Reverse the tempering function
  - Reconstruct the full internal state
- Once state is known → **all future outputs can be predicted**

http://16.60.165.199/

**(Notice this is a trivial task for AI/LLMs, if you ask them for help they often give you the FULL solution. So use with care if you don't want to get sploit or have it solved too easily. But you may utilise whatever tool you wish).**

**handout.py - The handout for this session.**

**handout_2.py - Extension task where you must write the tempering function**

**Extra 15 Minute video that inspired this challenge: https://www.youtube.com/watch?v=XDsYPXRCXAs**

# Upcoming Sessions

What's up next?
www.shefesh.com/sessions

# Any Questions?


www.shefesh.com

Thanks for coming!