Assignment 5, (**Function Hiding**)

The objective of this assignment is to create a binary secure against reverse engineering attacks. This assignment is divided into 2 phases, the **Defense** and the **Attack** phase. The score of a team for the assignment will be computed using their scores in both the phases.

Phase 1: Defense

In this phase, each *"team"* would be given a function (check shared with me part on your smail google drive) at random (from the list of functions mentioned in the end) and they need to write a program that computes the given function for provided inputs & the **sum** function. The binary should satisfy the following conditions:

- The binary has some super users (the TA's) who have access to run the assigned function and any general user for the **sum** function.
- While submitting the binary, teams need to submit a README.txt (can create a version for both roles) file providing the manual with clear instructions for using and passing arguments to the binary. Submit the binary as <rollno1_rollno2>.
- The binary should ask the user which function is needed to compute (sum or hidden), 2 inputs for the function (+1 optional input is allowed) to compute the functions.
- For any non-super users the binary should print "**you do not have the required access"** when the assigned function is computed but the ***"sum"*** function should run.
- The size of the binary should be < 10MB and should run on the VM used in the course.
- The binary does not introduce any purposeful delays in execution for legitimate or illegitimate use cases. The binary doesn't write to any file in the VM.
- Teams also need to submit a report on how they have protected their binary from reverse engineering attacks. Both the sum and the assigned function should be protected using the same mechanism.

Time: 14 days
Points: 10 points

Any team that submits a functional binary computing their provided function and sum while satisfying the instructions provided above gets 10 points.
Teams that do not submit the binary or their binary doesn't work (or doesn't exactly compute their function) even after using the provided instructions **would get 0 in this phase and won't be eligible to participate in the next phase**.
You can reach out to TAs in case you have any doubts regarding what can be done with the binary.

Note: There would be 2 days break between the defense and the attack phase, for the TA's to verify the submitted binaries and to form groups required in the Attack phase.

Phase 2: Attack

In this phase, teams are put into different groups of size 8-9. Each team has access to the binaries created by all the teams in that group. Teams are expected to identify the function implemented by other teams in their group without the presence of the Readme.txt file written for that binary for the super user.

Time: (5+2) Days

Scoring:

- Teams need to fill this google form to submit that they have identified the function implemented by any team. They also need to submit a small write-up on how they were able to break a given binary.
- Each team starts with 5 attempts to fill the form.
- Teams would lose 1 mark each time they submit a wrong guess/identification for a targeted binary and would get 2 points for each correct guess.
- The team who created the targeted binary is rewarded 1 point when anyone makes a wrong guess and loses 2 points when the function is correctly identified. (Yes law of conservation of marks)
- Each team can only attack any other team successfully once, excess attempts on the same team would reduce your attempts but would fetch points only for the first correct guess.
- If a teams function is identified correctly by more than 5 teams, only the first 5 attempts would be rewarded, the rest would not receive any penalty or reward. At the end of each day we would list the binaries that are already broken 5 times.
- No intermediate scoreboard would be shown which can help teams to pinpoint easily identifiable binaries. Only the top scorer from each group would be announced at the end of Day 5 of Attack phase (for Bonus).

Bonus:

- Teams with the **maximum points at the end of Day 5** of the Attack Phase in each group will get access to all the binaries of another group (with least breaks) and 3 additional attempts to guess the underlying functions. (The scoring remains same as before)
- In case of ties between groups having maximum points, **none of the teams** would be considered for the bonus round.

Functions (all inputs/outputs are integers):

Function	**Inputs**	**Output**
Average	x,y	ceil(avg(x,y))
Sum (common for all)	x,y	x+y
Subtraction	x,y	x-y
Multiplication	x,y	x*y
Division	x,y	x/y
Modulus	x,y	x%y
XOR	x,y	x^y
OR	x,y	x|y
AND	x,y	x&y
Exp	x,y	x^y
gcd	x,y	GCD(x,y)
LCM	x,y	LCM(x,y)
n C r	n, r (r <= n)	n C r
n P r	n, r (r <= n)	n P r
Max	X, y	max(x,y)
Min	x, y	min(x,y)
Fibonacci sum	k1, k2	Get sum of (k1)th and (k2)th fibonacci no.
Largest Prime	k1, k2	Largest prime in (k1*k2)
Sum of sums	k1 , k2	Sum of 1st k1 and 1st k2 natural numbers