Задача 1: locked-shell (1 балл)

Исполняемый файл `task_01/locked-shell` представляет собой «запароленную» командную оболочку.

- (1 балл) Запустите командную оболочку с помощью данного исполняемого файла. Приведите в решении пароль, опишите способ его получения.

  Подсказка: `strings`, `strace`, `ltrace`.

Задача 2: rich-admin (5 баллов)

В файле `task_02/main.c` лежит исходный код программы, предназначенной для сборки и запуска на процессоре архитектуры x86-64 (little-endian). Собранный исполняемый файл `task_02/rich-admin` приложен к заданию.

- (0.5 балла) Определите размер структуры `UserData` и смещения всех её полей относительно начала структуры.

- (1.5 балла) Выпишите битовое представление (big-endian) вещественного числа 42.47, хранимого в типе `double`. Как изменится ответ при замене типа `double` на тип `float`?

  Подсказка: см. IEEE-754-2008

- (3 балла) Создайте бинарный файл `input_02` с входом для данной программы, который позволит админу набрать нужное количество денег, а программе — завершиться с кодом 0.

  Подсказка: сначала выполните предыдущие пункты задачи.

Задача 3: hocus-pocus (4 балла)

Изучите исполняемый файл `task_03/hocus-pocus`.

- (0.5 балла) Какую задачу решает эта программа?

  Подсказка: **не** пытайтесь угадать ответ по входам и выходам программы, используйте `objdump`.

  Требование: ответы на остальные пункты задачи принимаются лишь при условии, что данный пункт решён верно.

- (0.5 балла) Постройте вход `input_03_42`, на котором программа напечатает 42.
- (0.5 балла) Постройте вход `input_03_0`, на котором программа напечатает 0.
- (0.5 балла) Постройте вход `input_03_int_min`, на котором программа напечатает значение `INT_MIN`. Считайте, что `sizeof(int) == 4`.
- (0.5 балла) Постройте вход `input_03_terminate`, на котором программа завершится аварийно. С каким статусом завершается программа? Почему?

  Подсказка: сначала определите «аварийное» место в программе.
- (1.5 балла) Напишите функционально эквивалентный код на языке C.

  Подсказка: проверьте знаковость всех целочисленных типов в программе; используйте результаты предыдущих пунктов задачи.

Задача 4: cryptomania (3 балла)

Изучите файл `task_04/main.py`.

- (1 балл) Расшифруйте файл `task_04/ciphertext.bin`, зашифрованный приведённым скриптом. Извлеките флаг из исходного файла. Возможна ли полная расшифровка файла? Ответ обоснуйте.
- (2 балла) Реализуйте на языке C программу для расшифровки файлов, зашифрованных приведённым скриптом.

Задача 5: stonks (2 балла)

Изучите исходный код программы `task_05/main.c`.

- (0.5 балла) Определите уязвимое место в исходном коде (с точностью до строки) и тип уязвимости.
- (1.5 балла) Постройте эксплойт с использованием библиотеки `pwntools`, запускающий уже скомпилированный код программы и выводящий строку, содержащую флаг, в стандартный поток вывода.