

Первый тур заключительного этапа

Первый тур заключительного этапа прошел на специальной платформе в формате CTF task-based.

Особенности тура:

- Длительность тура: 4 астрономических часа.
- Интернет отсутствует.
- Для каждого участника был подготовлен ноутбук с предустановленной ОС Kali Linux
- Каждая задача по категориям оценивалась от 1 до 3 баллов в зависимости от уровня сложности.
- Платформа хостилась на локальной сети Университета, что ограничивало поиск подсказок в сети интернет.

Все задания можно отнести к следующим категориям:

- **ADMIN** (задания на администрирование операционных систем). Обычно задания, связанные с работой сисадмина: восстановление данных, виртуальные машины и так далее.
- **CRYPTO** (Криптография – задания на криптографические алгоритмы, как на старинные, так и на современные).
- **FORENSIC** (Компьютерная криминалистика – расследование инцидентов, исследование различных дампов (сетевых, памяти и прочее), восстановление архивов.
- **NETWORK** (Задачи на знания сетевых протоколов, сетевого оборудования, принципов работы с сетевым трафиком и отслеживание вредоносной сетевой активности)
- **PPC** (Задачи на программирование или автоматизацию обработки большого количества данных)
- **Privilege Escalation** (повышение привилегий)
- **PWN** (задачи на поиск и эксплуатацию уязвимостей в скомпилированных приложениях) Поиск и эксплуатация бинарных уязвимостей)
- **STEGANO** (Стеганография - поиск и обнаружение скрытых каналов передачи, а также их организация)
- **REVERSE** (Обратная разработка – исследование бинарных файлов (программ) без исходных кодов и изучение работы различных редких архитектур)
- **WEB** (Поиск и эксплуатация веб-уязвимостей)

Задания Первого тура заключительного этапа**1. Admin****Балл: 1****Условие:**

Если ты знаешь, что за ссылки "magnet:", что такое шары, как качать файлы, то ты видел интернет нулевых. Это совсем старые файлообменники, подключайся, и скачай флаг

2. Crypto**Балл: 1****Условие:**

Пришли Вы на ЕГЭ, а там олимпиадная очевка:

$$N = p * q$$

$$e = 65537$$

$$c = m^e \bmod N$$

$$\text{hint} = (p^5 + q^5) / (p + q)$$

3. Forensic**Балл: 1****Условие:**

Дан дамп диска, он, к сожалению, битый, чуть-чуть. Вам нужно правильно ответить на эти вопросы:

1. sha1 hash исправленного диска (почините, чтобы он перестал быть битым, и снимите хэш)
2. полный путь до файла (начиная с /) , включая имя файла, размер которого 44 байта
3. самый часто встречающийся ip в файле access.log.10.gz

Вы должны использовать исключительно возможности командной строки, написав такое правило (в одну строчку), можно использовать перенаправление буфера, pipe и т.д. Все три действия должны подтверждаться скриншотами.

Например, по условию задачи вас просят вывести содержимое файла readme.md. В ответ пишем содержимое, на скриншоте должна быть видна команда cat readme.md (либо с вкраплением find, если файл в другой директории), либо же открытое через nano. То, что вы нашли именно тот readme.md, который требовался условием, также нужно подтвердить.

4. Network

Балл: 1

Условие:

В задаче приведен дамп трафика самописного бинарного протокола и получение флага. Надо проанализировать этот протокол и повторить тоже самое на бою, забрав оригинальный флаг.

5. PPC

Балл: 1

Условие:

На входе вам дается png-файл в формате base64. но он битый. В ответ мы ожидаем номер чанка, который поврежден, а также корректную сумму CRC32 (она в оригинальном файле битая). Сделайте так 100 раундов таких png check'ов, и получите нормальную картинку.

Отправлять ответ в формате id:crc (hex), например

1:0xdeadbeef

Man в помощь

6. Privilege Escalation

Балл: 1-2-3

Условие:

Зайти на сайт, разработанный для удобства и обзора китайской техники.

Задача ожидает три флага, а именно:

- Пароль пользователя с ником vmedkov. Не хэш, а именно пароль. Без CTF{}
- Неопубликованное фото админа
- Содержимое таблицы flags базы flags

В данной задаче можно использовать dirbusting. Брут паролей не разрешается.

7. Pwn

Балл: 1

Условие:

20 миллионов мощи, ого! Откройте окно и проветрите, а то эти фабрики загрязняют мне воздух.

8. Stego

Балл: 1-3

Условие:

Мы написали свой генератор для случайности в LSB. Вот он

```
def lcg_prng(seed=, m=, a=, c=):  
    x = seed % m  
    while True:  
        x = (a * x + c) % m  
        r = x + 1  
        shift = r % 3  
        channel = (r // 3) % 4  
        yield r, shift, channel
```

В файлах - два примера, спрятали там одинаковый текст

9. Reverse

Балл: 1

Условие:

Числа, числа, числа, я что-то явно упускаю...

10. Web

Балл: 1

Условие:

Посмотрите на исходный код веб-сайта (он написан на C++) и докажите новому СТО, что переезд со старого кода сделает только хуже! Например, все деньги можно будет украсть.