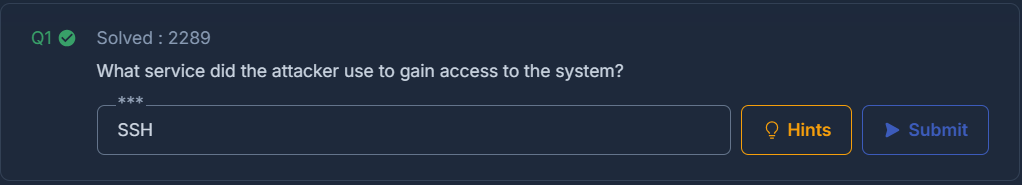
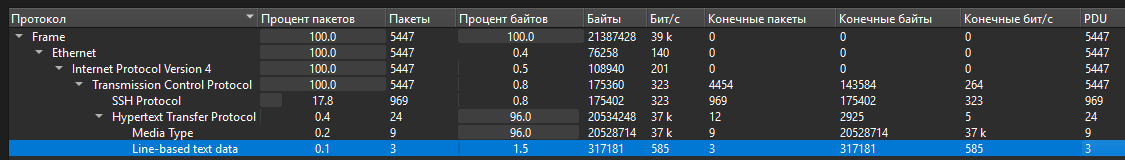
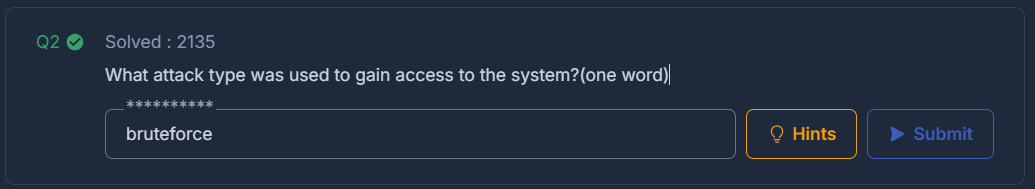
1.Для атаки был использован протокол SSH

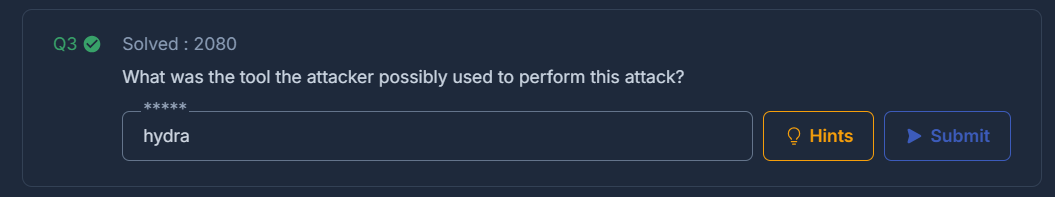




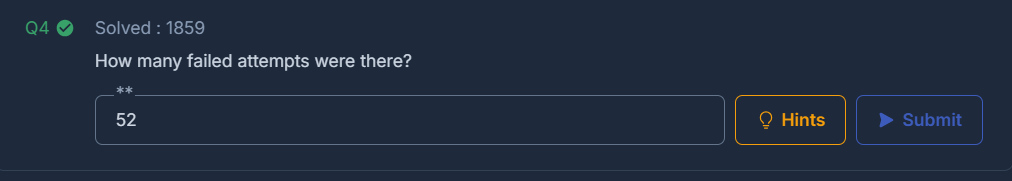
2.Тип атаки называется bruteforce, по большому количеству SSH запросов видно, что злоумышленник множество раз перебором пробовал произвести подключение.

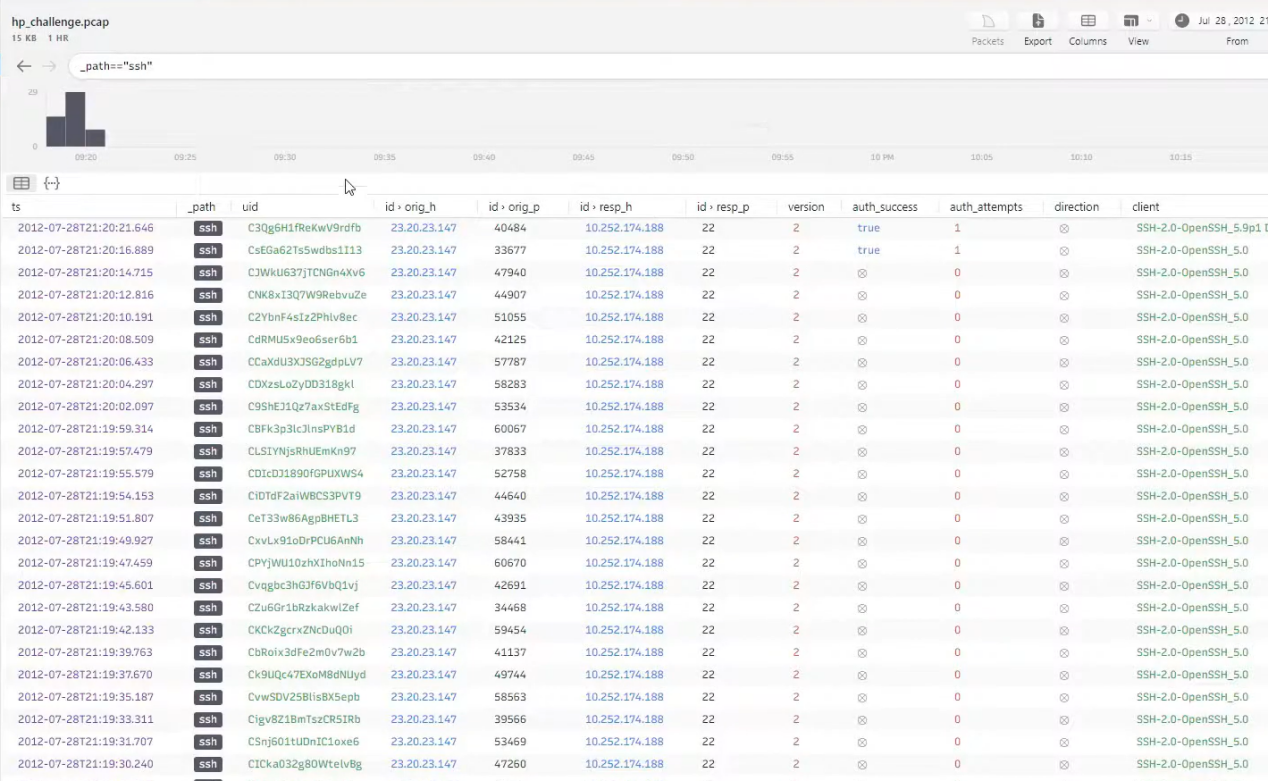


3.Hydra — один из самых популярных инструментов для автоматизации brute-force атак по SSH. Это предположение основано на характере сетевой активности: большое количество быстрых попыток авторизации.

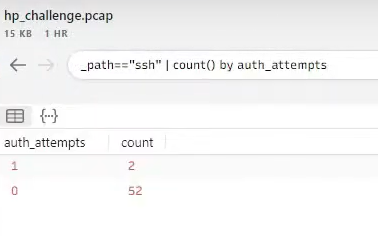


4.Всего неудачных попыток подключения было 52(Brim)





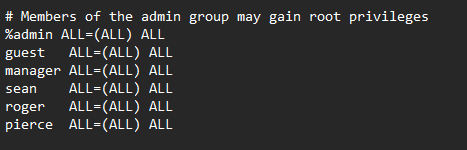
Общее количество считает выражением вида



Собственно говоря понимаем что удачный вход или нет по строчке в Brim



5-6.Тут смотрим кто имеет sudo права в sudoers.log



В shadows.log смотрим какой у нас хэш

{424FC764-20E0-4466-A68B-C7C319DFA9B8}

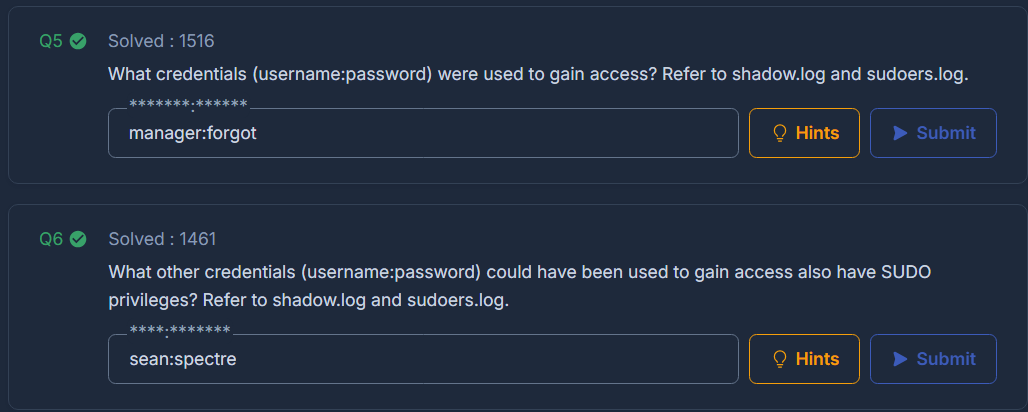
По hashcat(https://hashcat.net/wiki/doku.php?id=example\_hashes) определяем что это у нас sha512crypt $6$



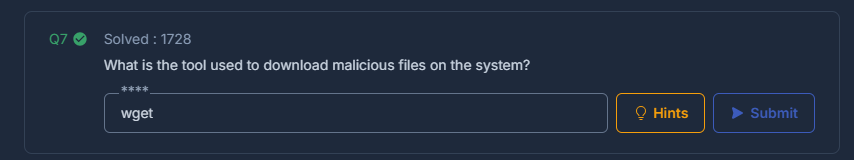
Для hashcat пишем

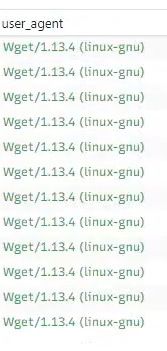
hashcat -m 1800(выбираем тип хэша) <название\_файла\_с\_хэшем> <словарь\_с\_распространёнными\_паролями> -o(флажок\_для\_сохранения\_результата\_в\_файл>)= <имя\_файла\_для\_сохранения>

hashcat -m 1800 gg.txt /usr/share/wordlists/rockyou.txt -o cracked.txt

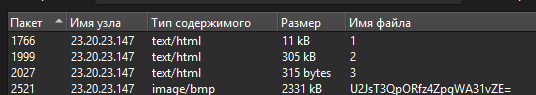


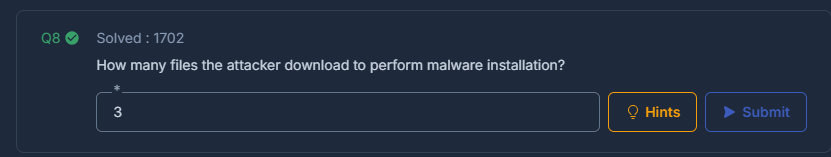
Если посмотрим наши http запросы(Brim), то в поле useragent можем увидеть wget



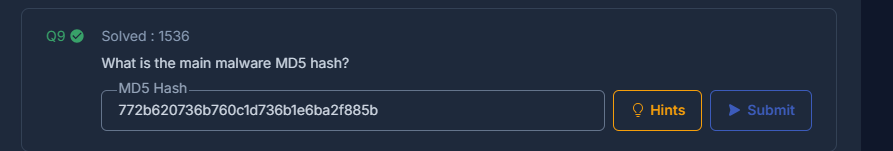


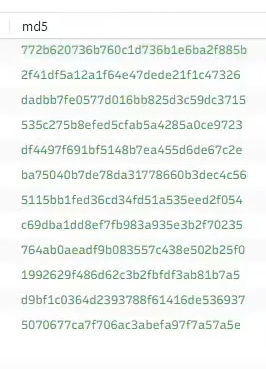
8.А вот у нас как раз 3 файлика в Wireshark

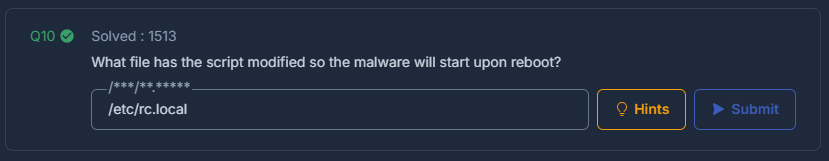


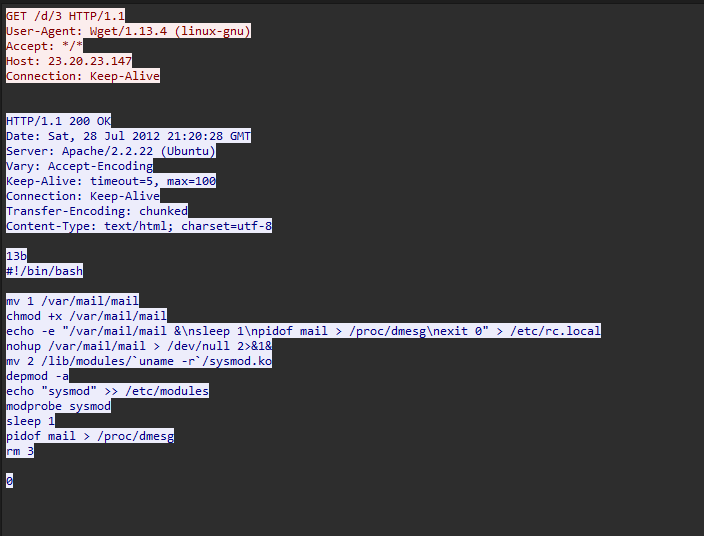


В Brim можно посмотреть хэш md5 по файлам

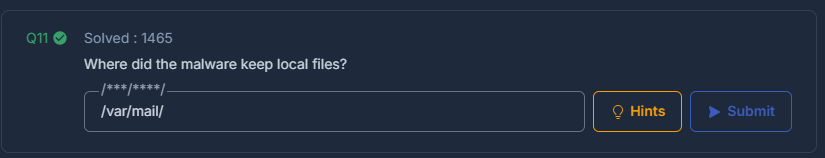




За автозапуск на Linux отвечает файл rc.local 

В данном случае видим что файлы троян хранил в /var/mail

11.



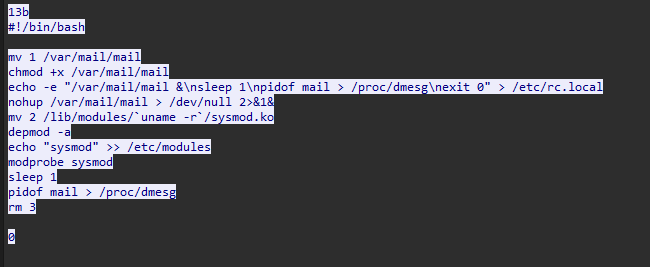
12.

nohup /var/mail/mail > /dev/null 2>&1&

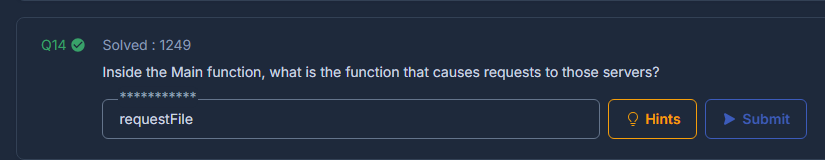
Если откроем ps.log то как раз /var/mail/mail там не обнаружим. Потому что запись в лог отключена /dev/null 2>&1&

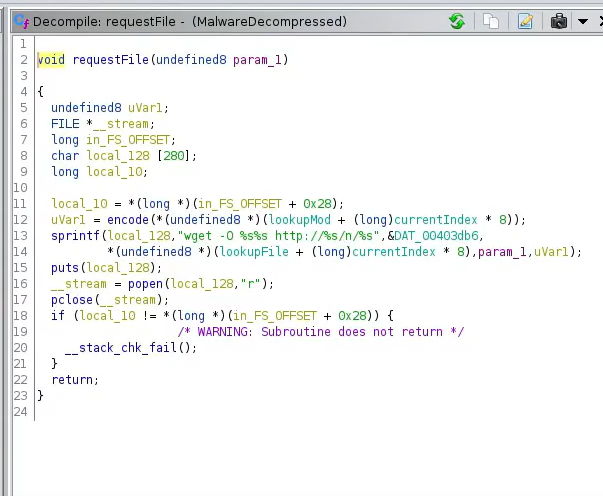


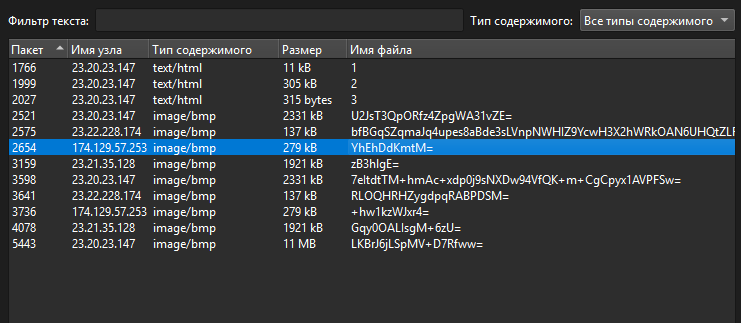
13.

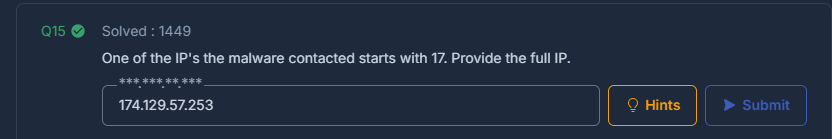
sysmod.ko — это вредоносный модуль ядра, который устанавливается и загружается злоумышленником для сокрытия его дальнейших действий на скомпрометированной машине.

14.Через reverse engineering analyse(Ghidra) можно увидеть, что функция называется requestFile

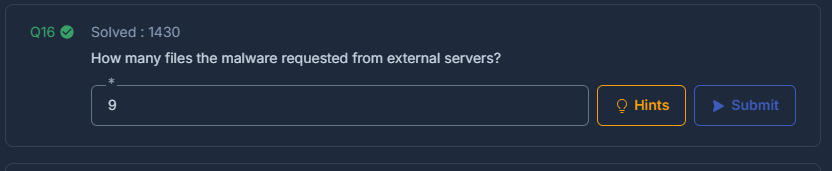
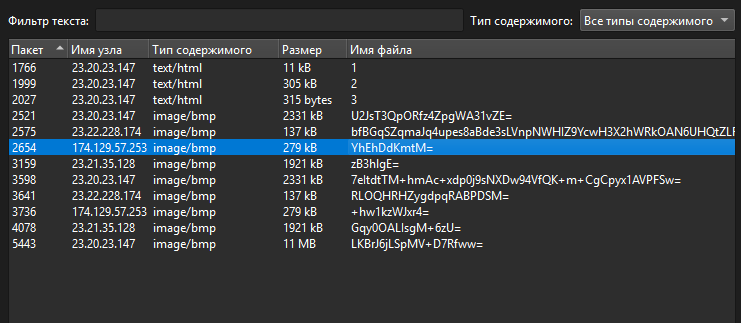




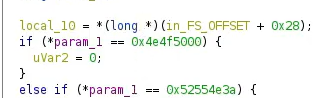
15.Через экспорт объектов можм увидеть подходящий под требования IP



16.Видимо что 9 файлов типа image/bmp



17.Через Ghidra получаем значения

\

