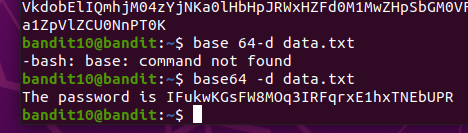
Level 10->11

-비밀번호는 data.txt파일에 Base64로 인코딩 되어 있다.

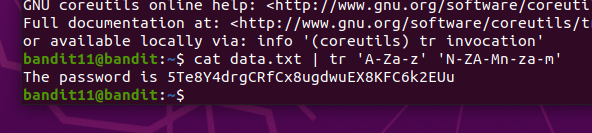
-base 64로 인코딩된 내용을 디코딩 해주면 된다.



Level 11->12

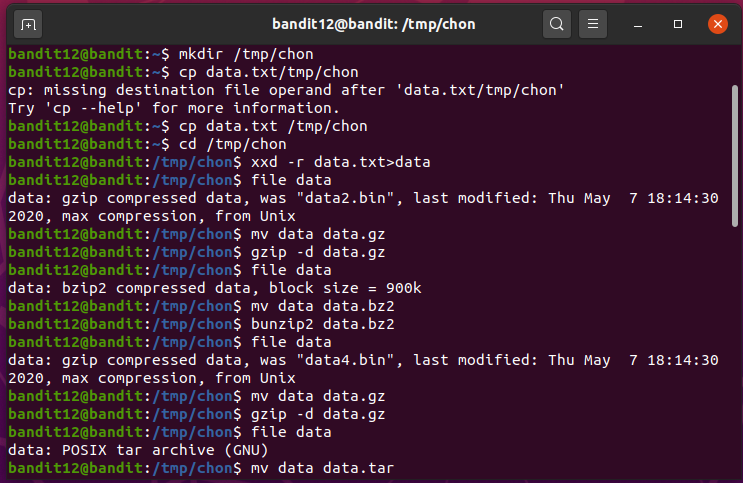
-data.txt파일에 rot13으로 쓰여진 암호를 해독해야 한다.

-tr 명령어를 이용하면 지정한 문자를 다른 문자로 바꾸거나 또는 특정 문자를 제거할 수 있다.

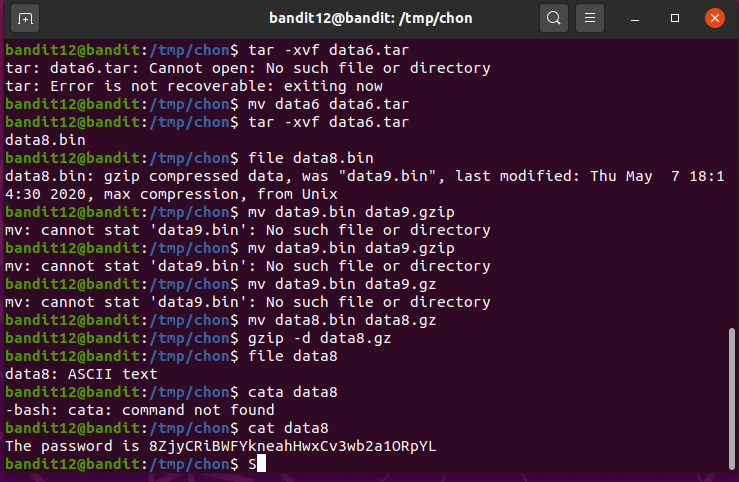


Level 12->13

-data.txt 파일에 hexdump 파일이 반복적으로 압축되어 있어 압축되어져 있는 데이터 파일을 압축을 풀어주는 과정을 계속 반복해야 한다.

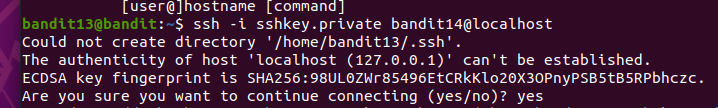


결과적으로 아래와 같이 패스워드를 알 수 있었다.



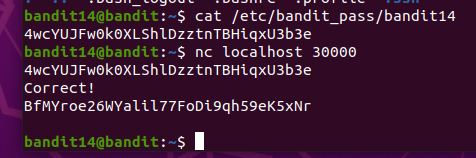
Level 13->14

-ssh에서 -i옵션으로 키를 가지고 bandit14에 접속할 수 있다.



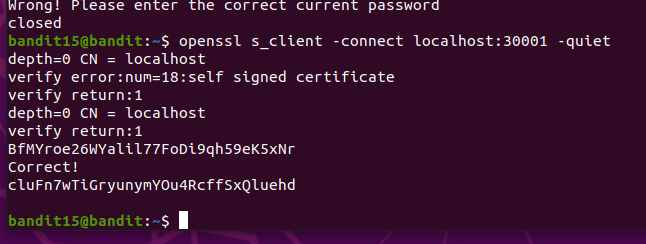
Level14->15

-14번에서 암호가 숨겨져있는 파일을 알려주었으므로 암호를 찾은 뒤에 포트 30000에 들어가서 암호를 입력하면 된다.



Level15->16

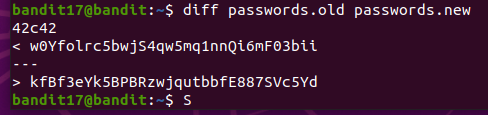
Openssl을 사용해서 저번 문제의 비밀번호를 그대로 30001번 포트로 옮겨주면 된다.



Level 16->17

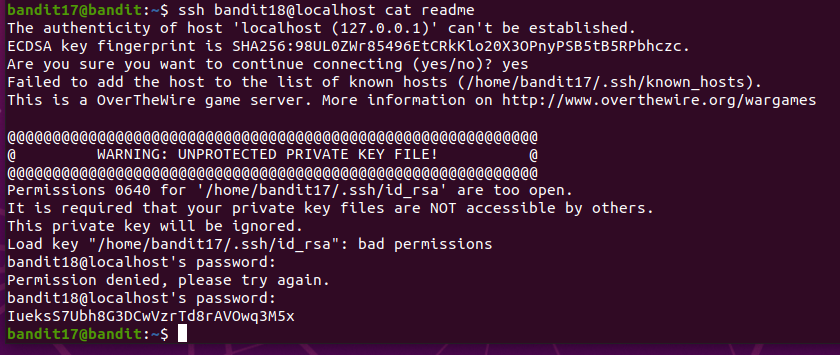
Level 17->18

-diff 명령어를 이용해 passwords.old와 passwords.new 두 파일에서 다른 점을 찾으면 된다.



Level 18->19

-bandit18 readme에 비밀번호가 저장되어 있지만 누군가 bashrc의 내용을 바꿔서 bandit18을 통해 로그인 하면 로그아웃되어 버린다. 그래서 bandit17을 통해서 바로 bandit18의 readme 파일을 읽어와야 한다.



Level 19->20

-홈 디렉토리의 setuid를 이용해야만 패스워드가 들어있는 파일을 읽을 수 있다.

-ls 로 bandit20-do 파일을 찾은 후 이 파일의 권한을 빌려서 패스워드가 들어있는 파일을 읽을 수 있다.

