**BÁO CÁO**

**Môn học: An toàn Mạng**

**Tên chủ đề: BANDIT -** [**OverTheWire: Level Goal: Bandit Level 0**](https://overthewire.org/wargames/bandit/bandit0.html)

*GVHD: Nghi Hoàng Khoa*

*Ngày báo cáo: 31/10/2023*

**Nhóm: 07 (ghi số thứ tự nhóm)**

1. **THÔNG TIN CHUNG:**

*(Liệt kê tất cả các thành viên trong nhóm)*

Lớp: NT140.O11.ANTN.1

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Họ và tên** | **MSSV** | **Email** |
| 1 | Hà Thị Thu Hiền | 21522056 | 21522056@gm.uit.edu.vn |
| 2 | Phạm Ngọc Thơ | 21522641 | 21522641@gm.uit.edu.vn |

1. **NỘI DUNG THỰC HIỆN:1**

|  |  |  |
| --- | --- | --- |
| **STT** | **Công việc** | **Kết quả tự đánh giá** |
| 1 | 34 LEVEL BANDIT | 100% |

**KẾT QUẢ**

bandit0-1: NH2SXQwcBdpmTEzi3bvBHMM9H66vVXjL

bandit1-2: rRGizSaX8Mk1RTb1CNQoXTcYZWU6lgzi

bandit2-3: aBZ0W5EmUfAf7kHTQeOwd8bauFJ2lAiG

bandit3-4: 2EW7BBsr6aMMoJ2HjW067dm8EgX26xNe

bandit4-5: lrIWWI6bB37kxfiCQZqUdOIYfr6eEeqR

bandit5-6: P4L4vucdmLnm8I7Vl7jG1ApGSfjYKqJU

bandit6-7: z7WtoNQU2XfjmMtWA8u5rN4vzqu4v99S

bandit7-8: TESKZC0XvTetK0S9xNwm25STk5iWrBvP

bandit8-9: EN632PlfYiZbn3PhVK3XOGSlNInNE00t

bandit9-10: G7w8LIi6J3kTb8A7j9LgrywtEUlyyp6s

bandit10-11: 6zPeziLdR2RKNdNYFNb6nVCKzphlXHBM

bandit11-12: JVNBBFSmZwKKOP0XbFXOoW8chDz5yVRv

bandit12-13: wbWdlBxEir4CaE8LaPhauuOo6pwRmrDw

bandit13-14: fGrHPx402xGC7U7rXKDaxiWFTOiF0ENq

bandit14-15: jN2kgmIXJ6fShzhT2avhotn4Zcka6tnt

bandit15-16: JQttfApK4SeyHwDlI9SXGR50qclOAil1

bandit16-17: ssh -i sshkey17.private bandit17@bandit.labs.overthewire.org -p 2220

bandit17-18: hga5tuuCLF6fFzUpnagiMN8ssu9LFrdg

bandit18-19: awhqfNnAbc1naukrpqDYcF95h7HoMTrC

bandit19-20: VxCazJaVykI6W36BkBU0mJTCM8rR95XT

bandit20-21: NvEJF7oVjkddltPSrdKEFOllh9V1IBcq

bandit21-22: WdDozAdTM2z9DiFEQ2mGlwngMfj4EZff

bandit22-23: QYw0Y2aiA672PsMmh9puTQuhoz8SyR2G

bandit23-24: VAfGXJ1PBSsPSnvsjI8p759leLZ9GGar

bandit24-25: p7TaowMYrmu23Ol8hiZh9UvD0O9hpx8d

bandit25-26: c7GvcKlw9mC7aUQaPx7nwFstuAIBw1o1

bandit26-27: YnQpBuifNMas1hcUFk70ZmqkhUU2EuaS

bandit27-28: AVanL161y9rsbcJIsFHuw35rjaOM19nR

bandit28-29: tQKvmcwNYcFS6vmPHIUSI3ShmsrQZK8S

bandit29-30: xbhV3HpNGlTIdnjUrdAlPzc2L6y9EOnS

bandit30-31: OoffzGDlzhAlerFJ2cAiz1D41JW1Mhmt

bandit31-32: rmCBvG56y58BXzv98yZGdO7ATVL5dW8y

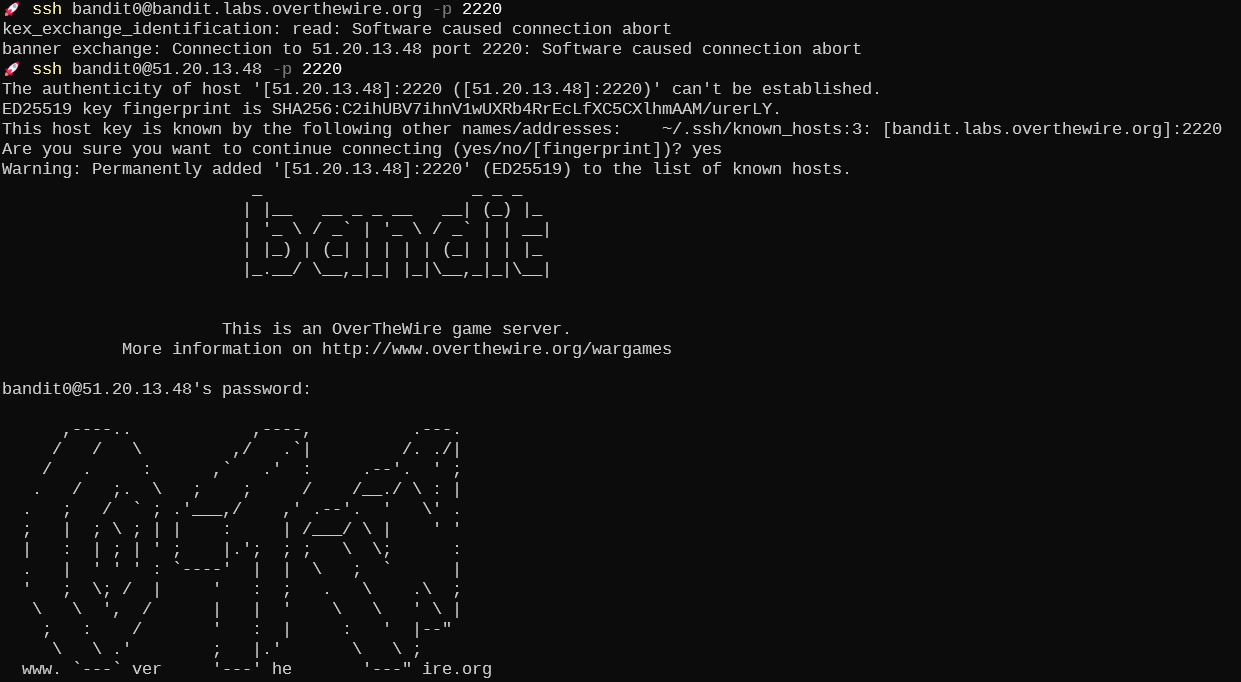
bandit32-33: odHo63fHiFqcWWJG9rLiLDtPm45KzUKy

Dạ mấy câu đầu em chỉ trình bày cách giải và càng về sau em càng giải thích chi tiết ạ.

# LEVEL 0

A screenshot of a computer

Description automatically generated



A screen shot of a computer

Description automatically generated

A black background with white text

Description automatically generated

A screen shot of a computer

Description automatically generated

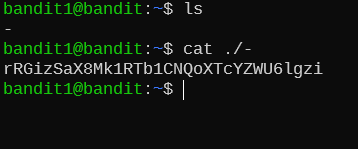
A screen shot of a computer

Description automatically generated

# LEVEL 1-2

A screenshot of a computer

Description automatically generated





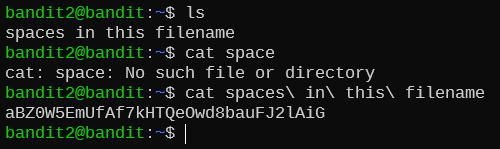
A screen shot of a computer

Description automatically generated

# 2. LEVEL 2-3

A screenshot of a game

Description automatically generated





A screen shot of a computer

Description automatically generated

# 3. LEVEL 3-4

A screenshot of a computer game

Description automatically generated

A screen shot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

A computer screen shot of a computer program

Description automatically generated

A screen shot of a computer

Description automatically generated

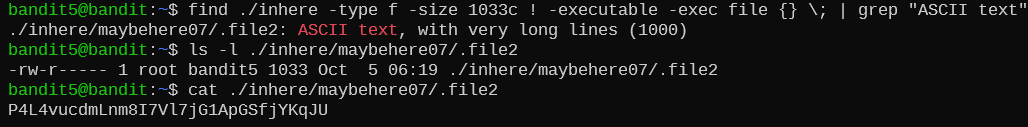
A screen shot of a computer

Description automatically generated

# 5. LEVEL 5-6

A screenshot of a black screen

Description automatically generated



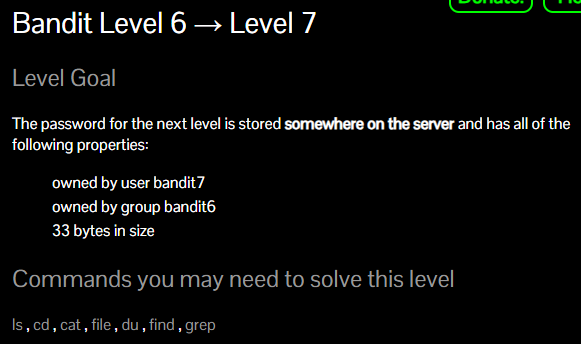
* **./inhere** là đường dẫn đến thư mục "inhere".
* **-type f** chỉ tìm kiếm tệp, không phải thư mục.
* **-size 1033c** tìm các tệp có kích thước là 1033 byte.
* **! -executable** đảm bảo rằng tệp không thể thực thi.
* **-exec file {} \;** sử dụng lệnh **file** để kiểm tra xem tệp có chứa dữ liệu đọc được bởi con người hay không.
* **grep "ASCII text"** sẽ lọc ra các tệp có nội dung là văn bản đọc được bởi con người.

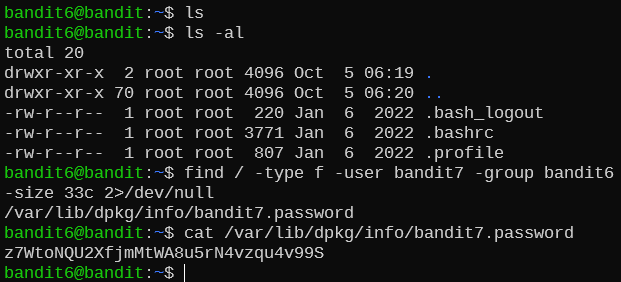


A screen shot of a computer

Description automatically generated

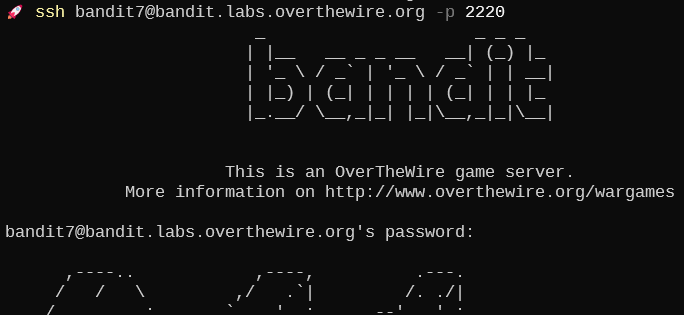
# 6. LEVEL 6-7





* **/** là thư mục gốc để tìm kiếm trên toàn bộ hệ thống.
* **-type f** chỉ tìm kiếm các tệp (không phải thư mục).
* **-user bandit7** chỉ tìm các tệp thuộc sở hữu của người dùng "bandit7".
* **-group bandit6** chỉ tìm các tệp thuộc sở hữu của nhóm "bandit6".
* **-size 33c** chỉ tìm các tệp có kích thước 33 byte.
* **2>/dev/null** dùng để ẩn thông báo lỗi nếu không có quyền truy cập vào một số thư mục.

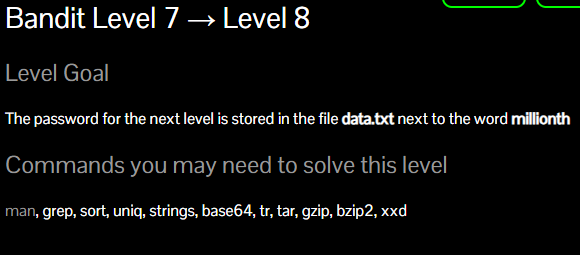
Lệnh **find** sẽ tìm và liệt kê tất cả các tệp thỏa mãn các điều kiện này trên toàn bộ hệ thống. Bạn cần kiểm tra từng tệp trong danh sách để tìm mật khẩu cho Bandit Level 7.

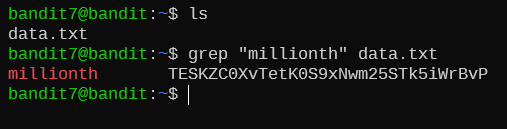


A screen shot of a computer

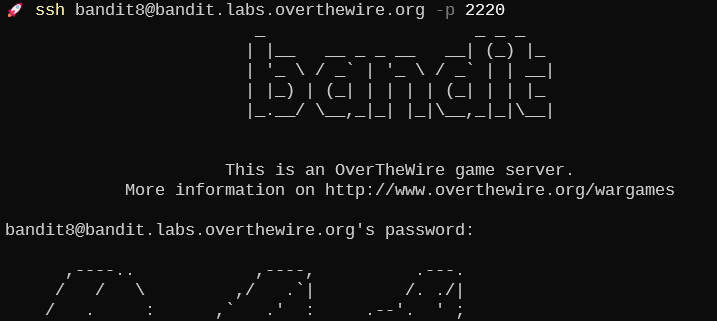
Description automatically generated

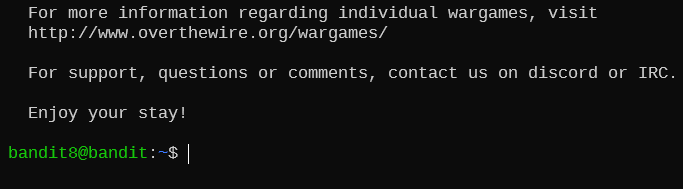
# 7. LEVEL 7-8



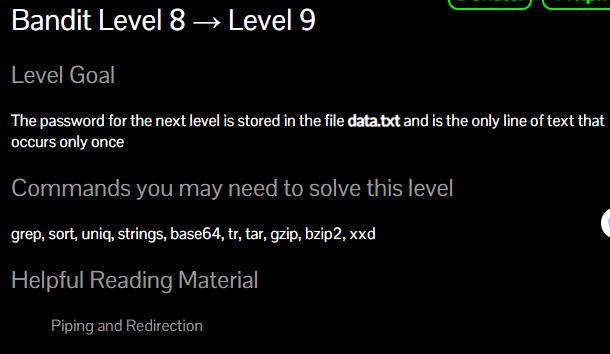


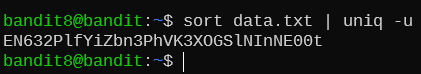
Lệnh này sẽ tìm kiếm từ "millionth" trong tệp "data.txt" và hiển thị dòng chứa từ đó. Mật khẩu cho Bandit Level 8 sẽ nằm trong dòng này.





# 8. LEVEL 8-9





* Đầu tiên, lệnh **sort** được sử dụng để sắp xếp các dòng trong tệp "data.txt" theo thứ tự từ điển.
* Sau khi tệp đã được sắp xếp, lệnh **uniq -u** (với tùy chọn **-u**) được sử dụng để hiển thị chỉ những dòng không trùng lặp, tức là chỉ dòng duy nhất trong tệp.

Lý do **sort** được sử dụng là để chuẩn hóa tệp và đảm bảo rằng tất cả các dòng giống nhau nằm gần nhau. Sau đó, **uniq -u** có thể tìm và hiển thị dòng duy nhất một cách dễ dàng. Nếu tệp không được sắp xếp trước, **uniq** có thể không hoạt động như mong đợi, vì nó chỉ loại bỏ các dòng liên tiếp giống nhau.

A screen shot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

# 9. LEVEL 9-10

A screenshot of a computer

Description automatically generated

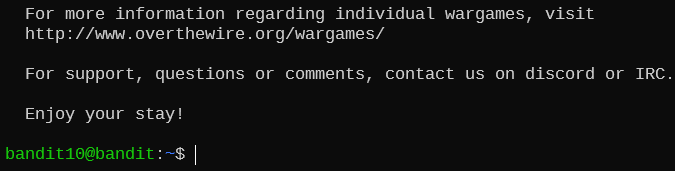
A black screen with white text

Description automatically generated

* Lệnh **strings** trích xuất các chuỗi có thể đọc được từ tệp "data.txt".
* Lệnh **grep "==="** tìm kiếm chuỗi nào có chứa một loạt ký tự "=", và mật khẩu sẽ nằm sau chuỗi này.

Khi chạy lệnh này, nó sẽ hiển thị chuỗi có "=" và mật khẩu cần tìm.

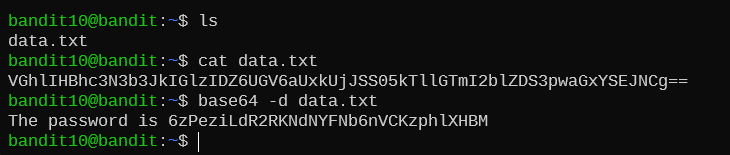




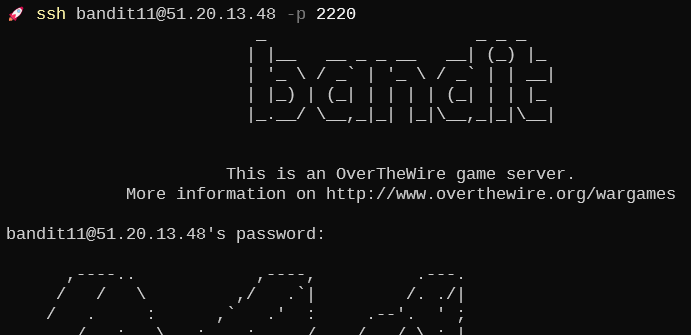
# 10. LEVEL 10-11

A screenshot of a black screen

Description automatically generated



* Lệnh “base64 -d data.txt” sẽ giải mã nội dung của tệp "data.txt" và hiển thị nó trên màn hình.



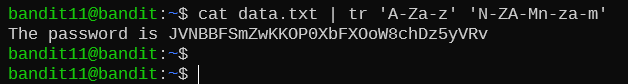
A screen shot of a computer

Description automatically generated

# 11. LEVEL 11-12

A screenshot of a computer

Description automatically generated



Lệnh **cat data.txt** đọc nội dung của tệp "data.txt" và xuất nó ra stdout (đầu ra tiêu chuẩn).

Lệnh **tr 'A-Za-z' 'N-ZA-Mn-za-m'** là lệnh sử dụng chương trình **tr** (translate or delete characters) để thực hiện việc thay đổi (hoặc dịch) các ký tự từ kiểu chữ cái tiếng Anh sang kiểu chữ cái ROT13.

* **'A-Za-z'** đại diện cho tất cả các ký tự chữ cái tiếng Anh (kể cả chữ in hoa và in thường) trong tầm từ 'A' đến 'Z' và sau đó từ 'a' đến 'z'.
* **'N-ZA-Mn-za-m'** đại diện cho các ký tự sau khi đã được ROT13. Nó sẽ biến đổi các ký tự 'A' thành 'N', 'B' thành 'O', 'C' thành 'P', và tương tự cho các chữ cái khác.

Kết hợp cả hai lệnh trên bằng dấu "pipe" (|), nó cho phép đầu ra từ lệnh **cat** (nội dung tệp "data.txt") trở thành đầu vào cho lệnh **tr**. Kết quả là lệnh **tr** thực hiện ROT13 trên nội dung tệp và xuất nó ra stdout, mà bạn có thể thấy trên màn hình.

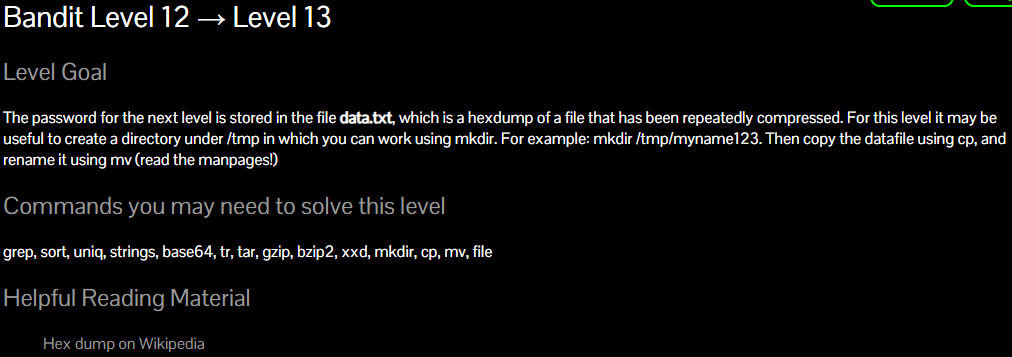
A screenshot of a computer screen

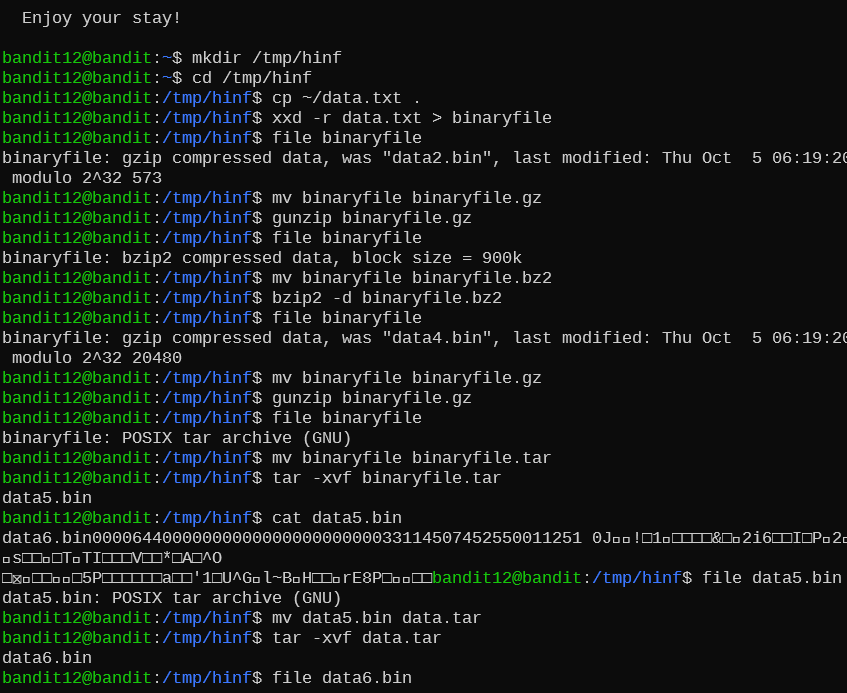
Description automatically generated

A screen shot of a computer

Description automatically generated

# 12. LEVEL 12-13





* Tạo một thư mục tạm thời trong /tmp bằng mkdir và điều hướng đến thư mục đó:
  + Mkdir /tmp/hinf
  + Cd /tmp/hinf
* Sau đó copy file data.txt vào thư mục hiện tại.
* Tiếp đến, sử dụng xxd -r data.txt > binaryfile để đảo ngược hexdump về dạng nhị phân ban đầu.
* Sau đó xem file binaryfile thuộc loại file nào và tiến hành đổi tên giải nén lần lượt đến khi thu được file: ASCII TEXT từ đó lấy được mật khẩu.

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

# 13. LEVEL 13-14

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

A computer screen with white text

Description automatically generated

# 14. LEVEL 14-15

A screenshot of a black screen

Description automatically generated



A screen shot of a computer

Description automatically generated

# 15. LEVEL 15-16

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer code

Description automatically generated

* Connect đến máy chủ trên port 30001 bằng openssl
* Sau đó, nhập pass của bandit gần nhất là bandit 15, sau đó nếu đúng pass thì hiển thị pass bandit 16.

# 16. LEVEL 16-17

A screenshot of a computer

Description automatically generated

* **Port scanning là phương pháp tìm các port đang mở trên máy chủ. Một port có thể được xem như một địa chỉ cho một dịch vụ cụ thể. Mọi máy tính đều có số port từ 0-65535. Một số dịch vụ có port mặc định như HTTP/80 hoặc SSH/22. Một port mở có nghĩa là máy chủ (**

**host) cung cấp dịch vụ mạng trên port này.**

* **Nmap là một máy quét mạng (network scanner). Nó có thể thực hiển khám phá máy chủ (host discovery), version detection (service detection),và hơn thế nữa. Đối với level bandit 17, -p cho phép chọn port nào để quét và theo mặc định thì Nmap quét top 1000 ports (không phải là 1000 ports đầu tiên). Sử dụng -p- để quét tất cả 65535 ports, -sV quét phát hiện phiên bản/ dịch vụ (version detection/ service detection). Nmap có thể thực hiện tất cả các lần quét có thể bằng cờ -A nhưng sẽ mất thời gian. Quét toàn bộ: nmap -p- -A <host>**

**<host>: IP address/ name**

* **Đầu tiên, chúng ta cần tìm các cổng mở trong khoảng từ 31000 đến 32000 trên localhost và kiểm tra xem dịch vụ nào đang chạy trên chúng.**

**A computer screen shot of a computer

Description automatically generated**

* **Nmap cho chúng ta biết rằng có 5 cổng đang mở. Chỉ có hai cổng (31518 và 31790) sử dụng SSL. Nmap cũng cho chúng ta biết rằng cổng 31518 chỉ chạy dịch vụ echo. Cổng 31790 chạy một dịch vụ không xác định.**
* **Bây giờ chúng ta sử dụng lại OpenSSL để kết nối với cổng 31790 trên localhost và gửi mật khẩu.**

**A computer screen shot of a program

Description automatically generated**

**A screenshot of a computer screen

Description automatically generated**

[What is Nmap and How to Use it – A Tutorial for the Greatest Scanning Tool of All Time (freecodecamp.org)](https://www.freecodecamp.org/news/what-is-nmap-and-how-to-use-it-a-tutorial-for-the-greatest-scanning-tool-of-all-time/)

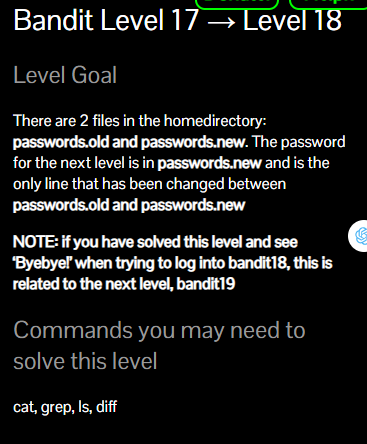
**A computer screen with white text

Description automatically generated**

**A screen shot of a computer

Description automatically generated**

# 17. LEVEL 17-18



* Lệnh diff in ra sự khác biệt giữa 2 files.

A screenshot of a computer

Description automatically generated

* Vì password.new chúng ta để ở sau nên password new cũng ở phần sau là hga5tuuCLF6fFzUpnagiMN8ssu9LFrdg

# 18. LEVEL 18-19

A screenshot of a computer game

Description automatically generated

A screenshot of a computer screen

Description automatically generatedA screenshot of a computer

Description automatically generatedA computer screen shot of a computer

Description automatically generated

* .bashrc là một tệp được chạy mỗi khi terminal được tải, điều này có nghĩa là nó cũng được chạy khi log in qua SSH vì thao tác này cũng tải terminal.
* SSH không chỉ cho phép chúng ta đăng nhập từ xa mà còn cho phép thực thi các lệnh từ xa bằng cách thêm các lệnh sau biểu thức SSH.

A screen shot of a computer screen

Description automatically generated

A screen shot of a computer screen

Description automatically generated

CÁCH 2:

A screen shot of a computer

Description automatically generated

* Ngoài ra, chúng ta có thể sử dụng cùng một phương pháp thực thi lệnh với SSH nhưng sử dụng /bin/bash làm lệnh để tạo bash shell hoặc sử dụng flag -t, cho phép 'pseudo-terminal' chạy trên máy đích, bằng cách này chúng ta có thể chạy \bin\sh. Điều này đặc biệt hữu ích nếu chúng ta phải thực hiện nhiều lệnh vì chúng ta không cần lặp lại câu lệnh SSH và mật khẩu.

# 19. LEVEL 19-20

A screenshot of a computer

Description automatically generated

* Setuid là một special permission. Nó thay thế cho x (excutable) của quyền user, đồng nghĩa với tệp nhị phân (binary) sẽ được chạy với tư cách là chủ sở hữu của tệp nhị phân chứ không phải là người thực thi nó. Để cấp quyền Setuid thì ta có cú pháp sau: chmode u+s <filename>.
* Đầu tiên, chúng ta kiểm tra xem ai là chủ sở hữu của tệp nhị phân setuid:

A screenshot of a computer

Description automatically generated

* Trong trường hợp này, chủ sở hữu là badit20 và nhóm (group) là bandit19, '-rwsr-x—' có nghĩa là người dùng bandit19 có thể thực thi tệp nhị phân, nhưng tệp nhị phân được thực thi với tư cách là người dùng bandit20.
* Chúng ta có thể truy cập tệp mật khẩu của người dùng bandit20 mà chỉ người dùng bandit20 mới có thể đọc được.

A computer screen shot of a computer

Description automatically generated

# 20. LEVEL 20-21

A screenshot of a computer screen

Description automatically generated

* Netcat hay nc để tạo máy chủ trên localhost, cần có flag -l (có nghĩa là đang nghe). Để chỉ định một cổng, cần có flag -p. Để tạo một "onetime server" (máy chủ dùng một lần), một server gửi một message rồi ngắt kết nối, chúng ta có thể sử dụng pipe (|) và echo để nhập message.
* Nếu một lệnh cần được chạy nhưng chúng ta không cần phải tương tác với nó trong một thời gian và muốn tiếp tục sử dụng cùng một terminal với các lệnh khác trong khi lệnh đang thực thi, ta có thể sử dụng &. Đây là một phần của quản lý quy trình Linux. Cụ thể, lệnh jobs hiển thị các processes/commands/jobs đang chạy ở chế độ background và foreground.
* Sử dụng 'netcat', chúng ta có thể tạo kết nối ở chế độ máy chủ (server mode) - chế độ này lắng nghe kết nối gửi đến. Để netcat gửi mật khẩu, ta sử dụng echo và chuyển nó vào netcat. Flag -n dùng để ngăn các ký tự dòng mới trong đầu vào. Cuối cùng, chúng ta để quá trình chạy ở background với &.

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

# 21. LEVEL 21-22

A screenshot of a computer program

Description automatically generated

* Cronjob là các chương trình chạy tự động đều đặn. Trong Linux, có nhiều thư mục có thể chứa các cronjob này: cron.d, cron.daily, cron.hourly, cron.monthly, crontab, cron.weekly. Các thư mục chứa các tệp có hướng dẫn về cách chạy chương trình. Nó bắt đầu với 5 cột đầu tiên, cho biết thời gian/khoảng thời gian mà chương trình nên được thực hiện. Tiếp theo là lệnh/chương trình sẽ được thực thi.

A computer screen with white text and numbers

Description automatically generated

A black background with white text

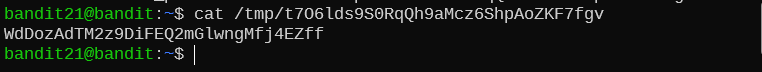
Description automatically generated

* Cronjob này chạy tệp /usr/bin/cronjob\_bandit22.sh với tư cách là người dùng bandit22. 5 ngôi sao cho biết nó được chạy mỗi phút, mỗi ngày. Để biết chính xác những gì được thực thi, chúng ta cần xem tệp bash.

A black screen with white text

Description automatically generated

* Ta có thể đọc được password ở file /tmp/t7O6lds9S0RqQh9aMcz6ShpAoZKF7fgv



# 22. LEVEL 22-23

A screen shot of a computer program

Description automatically generated

* Ở cấp độ trước ta về cronjob.
* Điều mới trong level này là các biến trong tập lệnh bash. Một biến giống chứa một giá trị. Để khai báo một biến trong tập lệnh bash, hãy sử dụng cú pháp sau: var\_name=var\_value. Có thể lưu đầu ra của lệnh vào một biến với cú pháp sau: var\_name=$(command). Truy cập giá trị của một biến hiện có như thế này: $var\_name

A computer screen shot of a program

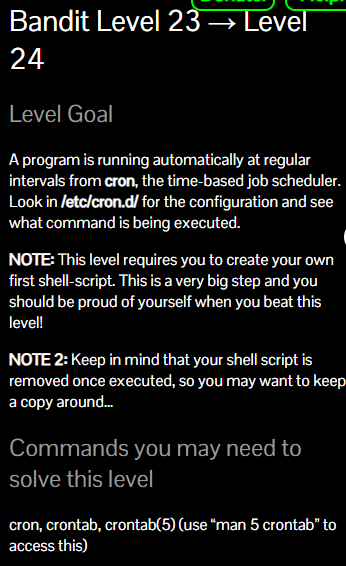
Description automatically generated

* Nhìn vào ‘/usr/bin/cronjob\_bandit23.sh’, dòng cuối cùng tương tự như level 22. Tập lệnh này chỉ giới thiệu các biến. Biến đầu tiên là 'myname' và lưu kết quả đầu ra từ lệnh whoami. Vì tập lệnh này sẽ được chạy dưới bandit23 nên lệnh whoami sẽ in ‘bandit23’. Vì vậy, dòng cuối cùng cho chúng ta biết rằng mật khẩu từ bandit23 sẽ được ghi vào một tệp trong thư mục ‘/tmp’. Tên tệp được tạo bởi dòng ‘echo I am user $myname | md5sum | cut -d ' ' -f 1’. Chúng ta chỉ cần thay $myname bằng bandit23, thực thi nó và kết quả là tên tệp.

A screenshot of a computer

Description automatically generated

# 23. LEVEL 23-24



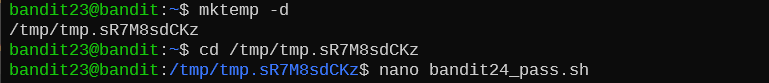
A screenshot of a computer program

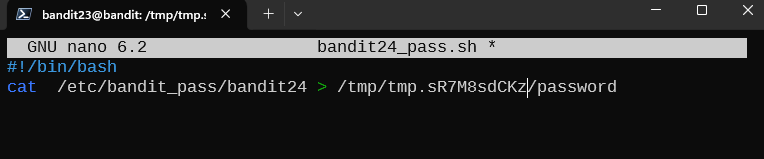
Description automatically generated

A screenshot of a computer program

Description automatically generated

* Tập lệnh thực thi và xóa tất cả các tệp trong thư mục ‘/var/spool/bandit24/foo’. Trường hợp này xảy ra vì nó được chạy với tư cách là người dùng bandit24, do đó biến ‘myname’ chứa giá trị ‘bandit24’. Vòng lặp for đi qua các tập tin. Câu lệnh if đầu tiên đảm bảo rằng các thư mục '.' và '..', đại diện cho các thư mục hiện tại và trước đó, bị bỏ qua. Bên trong câu lệnh if là đoạn mã để thực thi một tập lệnh, nhưng chỉ khi chủ sở hữu là bandit23. Sau đó tập tin sẽ bị xóa.
* Vì hiện tại chúng ta đang đăng nhập với tư cách là người dùng bandit23 nên chúng ta có thể tạo một script cung cấp cho chúng ta mật khẩu của bandit24. Đầu tiên, tạo một tệp trong thư mục 'tmp'. Điều này ngăn chặn việc xóa tệp sớm và ta có một bản sao trong trường hợp có sự cố. Sau đó di chuyển tệp vào thư mục ‘/var/spool/bandit24/foo’ và nó sẽ được thực thi.





* Mục đích để ghi password từ bandit24 vào file trong thư mục /tmp.
* Tiếp theo chúng ta sẽ cấp quyền cho thư mục và file và di chuyển file vào đúng thư mục.

A computer screen with text and numbers

Description automatically generated

# 24. LEVEL 24-25

A screenshot of a game

Description automatically generated

* Vòng lặp for trong bash có cú pháp là:

For var in 1 2 … N

Do

#something

Done

* Nếu biến được lặp trong một phạm vi cụ thể thì việc chỉ viết các giới hạn có thể dễ dàng hơn. Ví dụ: nếu chúng ta muốn lặp từ 1 đến 10, cú pháp sẽ là {0..10}. Ngoài ra, nếu muốn mọi số có hai chữ số (00, 01, 02, … 10), chúng ta có thể thêm chữ số 0 vào số {00..10}.
* Đầu tiên, ta sẽ kết nối với port 30002 bằng netcat để xem script và phản hồi của nó.

A screenshot of a computer

Description automatically generated

* Như chúng ta thấy thì port không đúng, bây giờ ta sẽ brute-forcing bằng script:

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generatedA screen shot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

# 25. LEVEL 25-26

A screenshot of a game

Description automatically generated

* Mỗi user có một user shell mặc định. Điều này đặc biệt quan trọng khi sử dụng ssh, vì đây là shell sẽ được hiển thị. Thông tin về shell mặc định cho user có thể được tìm thấy ở cuối dòng cho người dùng trong tệp '/etc/passwd'.
* More là một lệnh shell cho phép hiển thị các file ở chế độ tương tác (interactive mode). Chế độ tương tác chỉ hoạt động khi nội dung của file quá lớn không thể hiển thị đầy đủ trong cửa sổ terminal. Một lệnh được cho phép trong chế độ tương tác là v (vim) – lệnh này sẽ mở trình soạn thảo vim.
* Vim là một trình soạn thảo văn bản, nó cũng cho phép chạy các lệnh shell. Có thể sử dụng vim để thoát ra khỏi môi trường bị hạn chế và tạo ra một lớp vỏ. Để sinh ra shell mặc định của user, lệnh ‘:shell’ được sử dụng. Để thay đổi shell thành ‘/bin/bash’, lệnh là ‘:set shell=/bin/sh.
* Đầu tiên, chúng ta cần kiểm tra xem người dùng bandit26 đã sử dụng shell nào. Kiểm tra bằng cách tìm dòng chính xác trong tệp 'passwd'.

A computer screen with white text and green text

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screen shot of a computer

Description automatically generated

* Chúng ta có thể thấy rằng nó đề cập đến một script có tên 'showtext' mở tệp có tên 'text.txt' bằng more.
* Tiếp theo, trong home directory của người dùng hiện tại, chúng ta tìm thấy khoá SSH private. Chúng ta có thể copy-paste private key vào 1 file trên máy mình. Để có thể đăng nhập, chúng ta cần giới hạn quyền đối với key.
* Khi cố gắng đăng nhập, chúng ta thấy kết nối bị đóng vì '/usr/bin/showtext' được thực thi.
* Chính xác thì chuyện gì đã xảy ra? Văn bản trong 'text.txt' rất ngắn, nghĩa là toàn bộ văn bản có thể được hiển thị ngay lập tức. more không cần phải chuyển sang chế độ tương tác. Nếu chúng ta làm cho cửa sổ terminal nhỏ hơn thì nhiều cửa sổ khác sẽ chuyển sang chế độ lệnh. Sau đó chúng ta có thể sử dụng v để vào vim. Bây giờ chúng ta có thể thay đổi kích thước cửa sổ.

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

* Vim hiện đã được mở với tên bandit26 và chúng ta có thể làm nhiều việc khác nhau để lấy lại mật khẩu. Với ‘:e /etc/bandit\\_pass/bandit26’ chúng ta có thể mở tệp mật khẩu và đọc mật khẩu. Nếu chúng ta muốn có một shell, chúng ta có thể thử lệnh ‘:shell’ mà vim đưa ra. Tuy nhiên, lệnh này sử dụng shell mặc định của người dùng. Thay vào đó, điều chúng ta cần làm là đặt shell mặc định của người dùng trong vim thành một shell hữu ích, như \bin\bash. Các lệnh trông giống như sau: và sau đó sử dụng ‘:shell’. Cuối cùng, chúng ta có shell và có thể lấy mật khẩu cho người dùng.

# 26. LEVEL 26-27

A screenshot of a game

Description automatically generated

* Sau khi giải được level 26 ta đã có được shell và tìm thấy file bandit27-do trong thư mục chính của bandit26.

A black and white screen with white text

Description automatically generated

A black screen with white lines

Description automatically generated

A computer screen with white text

Description automatically generated

* Nó chỉ chạy lệnh với tư cách một user khác (level 20). Bằng cách này, chúng ta lấy mật khẩu một cách đơn giản.

# 27. LEVEL 27-28

A screenshot of a computer

Description automatically generated

* Đầu tiên, chúng ta tạo một thư mục cho dự án, sau đó clone về.

A screenshot of a computer screen

Description automatically generated

* Ta thấy thư mục ‘.git' và tệp ‘README’. Vì đây là file duy nhất nên chúng ta kiểm tra nội dung và tìm mật khẩu cho level tiếp theo.

A screen shot of a computer

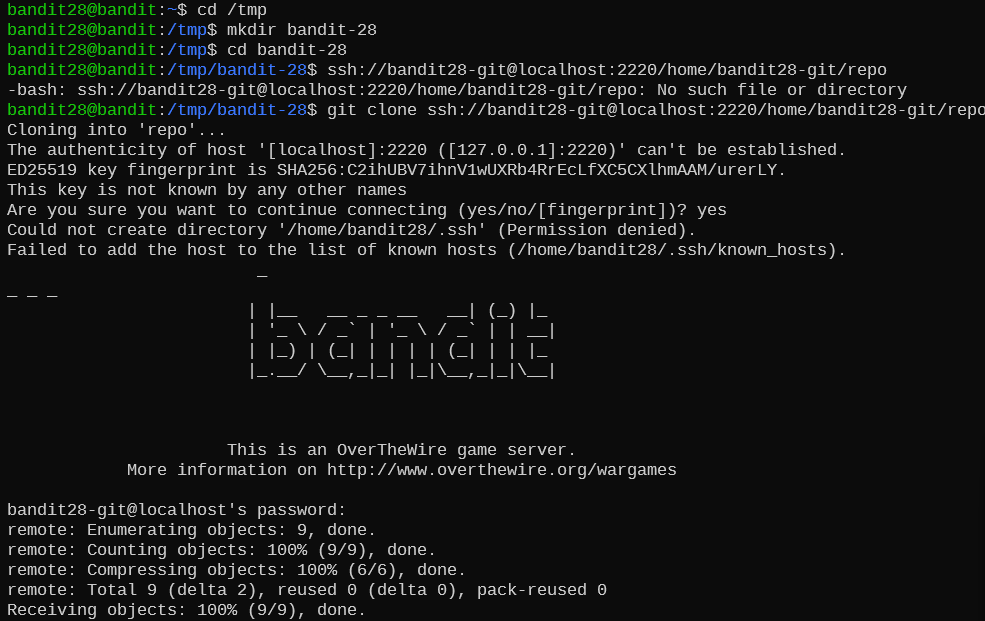
Description automatically generated

# 28. LEVEL 28-29

A screenshot of a computer

Description automatically generated

* Kiểm tra file README.md tương tự như level trước.



A screen shot of a computer

Description automatically generated

* Lần này, README có định dạng Markdown ('.md'). Nó đề cập nhưng không chứa mật khẩu. Chúng ta có thể xem lịch sử git để xem liệu phiên bản trước của tệp README có chứa mật khẩu hay không. Đầu tiên, chúng ta kiểm tra log.

A screenshot of a computer

Description automatically generated

* Nhìn vào ‘fix info leak’, hy vọng ta sẽ có thông tin về mật khẩu.
* Ở đây chúng ta thấy sự khác biệt giữa tệp README cũ và sau khi cập nhật. Lịch sử đã cho chúng ta thấy mật khẩu.

A screen shot of a computer

Description automatically generated

# 29. LEVEL 29-30

A screenshot of a computer

Description automatically generated

* Tương tự ta tạo một thư mục, clone git và kiểm tra README.md

A screenshot of a computer screen

Description automatically generatedA screenshot of a computer screen

Description automatically generated

* ‘no passwords in production!’ có vẻ như có thể có nhiều branch hơn. Kiểm tra từng branch.

A computer screen shot of a black screen

Description automatically generated

* Nhánh 'dev' chứa mật khẩu trong file 'README'.

# 30. LEVEL 30-31

A screenshot of a computer program

Description automatically generated

* Tương tự ta tạo một thư mục, clone git và kiểm tra README.md

A screenshot of a computer program

Description automatically generated

A screen shot of a computer

Description automatically generated

* README không cung cấp bất kỳ thông tin nào. Kiểm tra thẻ git, ta thấy ‘secret’.
* Git show secret ta thấy được mật khẩu.

A black screen with white text

Description automatically generated

# 31. LEVEL 31-32

A screenshot of a phone

Description automatically generated

* Tương tự ta tạo một thư mục, clone git và kiểm tra README.md

A screenshot of a computer screen

Description automatically generated

* 'README' nói rằng ta phải đẩy file vào kho lưu trữ repository. Vì vậy, trước tiên, ta tạo file:

A screenshot of a computer

Description automatically generated

* Bây giờ chúng ta có thể thử push file mới bằng cách trước tiên thực hiện một commit và sau đó push nó. Tuy nhiên, điều này sẽ không thực hiện được vì file ‘.gitignore’ trong repo. File này liệt kê files/-types mà git sẽ không push vào kho lưu trữ repository. Trong trường hợp này, chính xác là tất cả các tệp kết thúc bằng ‘.txt'.



* Để thêm file, chúng ta cần sử dụng git add. Sau đó commit và push.

A computer screen with white and blue text

Description automatically generatedA screenshot of a computer screen

Description automatically generated

# 32. LEVEL 32-33

A screenshot of a video game

Description automatically generated

* Khi sử dụng ssh để truy cập vào máy, chúng ta được chào đón bằng một shell hơi khác: WELCOME TO THE CAPS SHELL. Điều này đã cho chúng ta ý tưởng về chức năng của shell. Kiểm tra một lệnh đơn giản, ta thấy:

A black screen with white text

Description automatically generated

* Vì vậy, mọi thứ chúng ta gõ dường như được viết hoa. Tuy nhiên, các lệnh ta đã sử dụng cho đến nay đều là chữ thường và không hoạt động. Một thứ trong Linux được viết hoa là các biến. Cụ thể, biến $0 có tham chiếu đến shell.

A screenshot of a computer

Description automatically generated

Chúng ta có thể thấy rằng tệp ‘uppershell’ chạy dưới dạng bandit33 (thuộc sở hữu của người dùng ‘bandit33’ và SUID). Kiểm tra điều này, chúng ta có thể thấy rằng trên thực tế chúng ta là ‘bandit33’ và do đó, chúng ta có thể đọc file mật khẩu: odHo63fHiFqcWWJG9rLiLDtPm45KzUKy