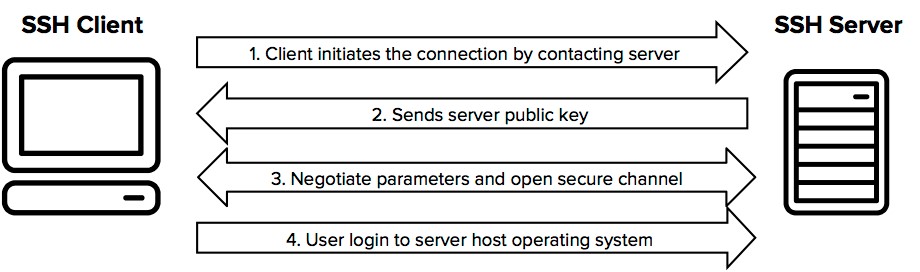
**Challenge 3**

1. **Các hoạt động của dịch vụ sshd và tiến trình ssh**

SSH tên đầy đủ là Secure Shell là một giao thức mạng mang tính bảo mật, cung cấp cách thức truy cập từ xa vào một thiết bị khác một cách an toàn thông qua các cơ chế mã hóa. Bằng cách sử dụng một số kỹ thuật mã hóa, xác thực mỗi bên với nhau để đảm bảo dữ liệu trao đổi giữa các máy trong phiên làm việc được mã hóa và giải mã nhằm mục đích giữ vững tính toàn vẹn của dữ liệu và bảo vệ an ninh thông tin.

Sshd (OpenSSH Daemon) là chương trình nền cho ssh. Các chương trình này cùng nhau thay thế rlogin và rsh, đồng thời cung cấp thông tin liên lạc được mã hóa an toàn giữa hai máy chủ không đáng tin cậy qua một mạng không an toàn.



***1.1. Quá trình xác thực của dịch vụ sshd***

* Bất kì khi nào 1 người dùng kết nối, daemon sẽ phản hồi bằng cách public host và server key.
* Người dung sẽ so sánh RSA host key với cơ sở dữ liệu của người dùng để xác định rằng key không đổi
* Sau đó người dung tạo 1 số ngẫu nhiên 256bit
* Daemon sẽ mã hóa số ngẫu nhiên này làm session key để mã hóa tất cả các giao thức khác trong session
* Cuối cùng máy chủ và người dùng sẽ đi vào bước authentication dialog khi người dùng sẽ tự xác thực bản thân bằng public key authentication password authentication
* Tài khoản có thể đăng nhập không thành công, không được phép truy cập nếu tài khoản nằm trong DenyUsers hoặc nhóm của nó nằm trong DenyGroup

***1.2. Quá trình đăng nhập của dịch vụ SSHD***

* Nếu đăng nhập trên tty và không có lệnh nào được thực thi, SSHD sẽ in thời gian đăng nhập cuối cùng và chạy lệnh /etc/motd
* Nếu đăng nhập trên tty, ghi lại thời gian đăng nhập
* Kiểm tra tệp /etc/nologin, nếu tệp tin đó tồn tại, nó in ra nội dung và thoát
* Thay đổi sang chế độ chạy với các đặc quyền của người dùng thông thường
* Thiết lập môi trường cơ bản
* Đọc tệp ~/.ssh/environment, nếu tệp tồn tại thì người dùng được phép thay đổi môi trường của họ
* Chuyển đến thư mục chính của người dùng
* Nếu ~/.ssh/rc tồn tại, chạy tệp. Nếu /etc/ssh/sshrc tồn tại, chạy tệp. Nếu không thì chạy auth
* Thực thi user’s shell hoặc command

***1.3. Tiến trình SSH***

+ Xác minh máy chủ bởi người dùng

* Người dùng khởi tạo kết nối SSH với máy chủ
* Máy chủ listens port mặc định 22 cho các kết nối SSH
* Danh tính máy chủ được xác minh bằng 2 trường hợp
  + TH1: Nếu người dùng truy cập máy chủ lần đầu, máy người dùng sẽ được yêu cầu xác thực máy chủ theo cách thủ công bằng cách xác minh public key của máy chủ
  + TH2: Nếu người dùng truy cập vào máy chủ không phải lần đầu, danh tính máy chủ sẽ được xác minh với thông tin đã được ghi trước đó trong tệp known\_hosts

+ Tạo session key để mã hóa tất cả những thông tin liên lạc

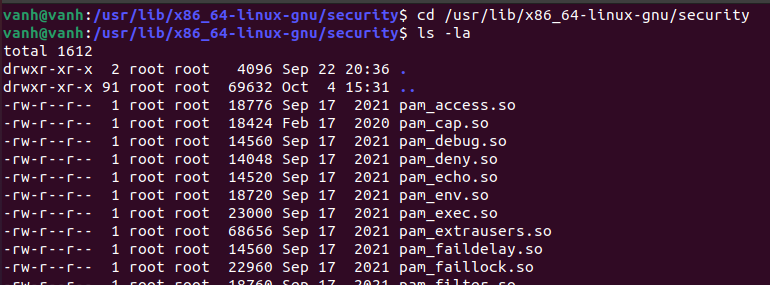
* Sau khi máy chủ được xác minh, cả 2 bên sẽ tạo session key sử dụng thuật toán Diffle Hellman
* Session key được tạo là khối đối xứng chung, tức là 1 khóa được sử dụng để mã hóa và giải mã

+ Xác thực người dùng

* Máy người dùng bắt đầu bằng cách gửi ID cho cặp khóa mà nó muốn xác thực tới máy chủ
* Máy chủ kiểm tra tệp authorized\_keys của tài khoản mà máy người dùng đang cố gắng đăng nhập để tìm ID của khóa
* Nếu tìm thấy public key có ID giống trong tệp, máy chủ sẽ tạo 1 số ngẫu nhiên và sử dụng public key để mã hóa số và gửi tin nhắn được mã hóa
* Nếu máy người dùng có private key đúng, nó sẽ giải mã tin nhắn đế lấy số ngẫu nhiên do máy chủ tạo ra
* Máy người dùng kết hợp số ngẫu nhiên thu được với session key được chia sẻ để tính toán giá trị MD5
* Sau đó máy người dùng sẽ gửi lại mã MD5 này cho máy chủ như 1 câu trả lời cho số đã được mã hóa
* Máy chủ sử dụng cùng 1 session key được chia sẻ và số ban đầu mà nó gửi cho người dùng để tự tính toán giá trị MD5
* Máy chủ so sánh kết quả của nó với kết quả mà khách hàng đã gửi lại. Nếu 2 giá trị này khớp nhau, điều đó chứng tỏ rằng máy người dùng đang sở hữu private key và người dùng được xác thực

1. **Tìm hiểu cơ chế xác thực pam của hệ điều hành linux.**

* Người dùng chạy 1 ứng dụng để truy cập vào dịch vụ mong muốn
* PAM-aware application (Chương trình giúp các privile-granting application làm việc với thư mục PAM), gọi thư viên PAM để thực hiện nhiệm vụ xác thực. Thư viện các module được lưu trong: /usr/lib/x86\_64-linux-gnu/security



* PAM Library sẽ load các module yêu cầu cho việc xác thực được đề cập trên
* Các modules này sẽ tạo 1 liên kết tới các hàm chuyển đổi trên chương trình
* Các hàm này dựa vào các modules mà đưa ra các yêu cầu với người dùng (vd ở login, chúng có thể yêu cầu người dùng nhập password)

1. **Tìm hiểu cách hoạt động của strace, ptrace.**

Strace là chương trình trong linux được build trên nền của ptrace, khi khởi tạo sẽ chạy câu lệnh chỉ định cho đến khi kết thúc. Strace sẽ chặn và ghi lại các câu lệnh gọi của hệ thống và các tín hiệu mà tiến trình thực hiện. Tên của các lệnh gọi hệ thống, các tham số truyền vào và các giá trị trả về được lưu vào trong stderr

* Là lệnh của shell, sử dụng ptrace để trace 1 tiến trình nào đó (có thể bao gồm cả những tiến trình con) đến khi kết thúc chương trình
* Strace sẽ log lại những syscall của trace, mỗi dòng tương ứng với 1 syscall theo thứ tự thời gian

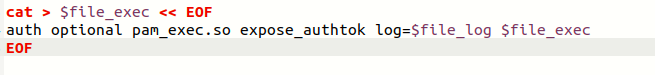
Ptrace là một hàm được sử dụng để ghi lại các lệnh gọi và các tín hiệu của một chương trình khác. Ptrace thường được sử dụng để tạo breakpoint debugging.

* Khi 1 câu lệnh được thực thi, strace sẽ tạo ra 2 process, 1 để thực hiện chương trình, một để theo dõi chương trình đó.
* Khi chương trình được theo dõi, nó sẽ thực hiện ptrace request tên ptrace\_traceme
* Tiến trình theo dõi sẽ bắt đầu theo dõi, được báo hiệu khi chương trình đang được thực thi gửi lại SIGSTOP
* Khi này chương trình được theo dõi sẽ chạy đến hoặc điểm khởi đầu hoặc thoát khỏi system calls
* Khi chương trình được theo dõi đến 1 system call, PTRACE\_PEEKUSER sẽ decode system call number

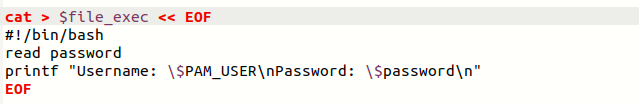
**Thực hành**

Bài 1: [sshtrojan1.sh](https://github.com/vvaannhh/VCS_Training/blob/main/prog3/sshtrojan1.shhttps:/github.com/vvaannhh/VCS_Training/blob/main/prog3/sshtrojan1.sh)

* Sửa file config của PAM và dùng module pam\_exec để thực hiện log password được login



* Dùng module auth để kết nối với expose\_authtok để chạy lệnh shell $file\_exec đọc password từ input nhập vào
* Dùng optional để không
* Kết quả sẽ được lưu vào file $file\_log



* PAM\_USER là biến môi trường của auth
* Password được lấy từ input nhập vào

Bài 2: [sshtrojan2.sh](https://github.com/vvaannhh/VCS_Training/blob/main/prog3/sshtrojan2.sh)

* Chương trình sử dụng hàm strace để theo dõi hệ thống.
* File sshtrojan2.sh theo dõi tiến trình login ssh và ghi file log
* Lấy được pid của tiến trình từ đó hàm strace theo dõi và ghi log vào 1 file tạm thời. Khi kết thúc tiến trình ssh, lấy từ kí tự password từ file log tạm thời