**CHỦ ĐỀ 1:**

Chương 2: Các công cụ phát triển phần mềm

2.1. Khái niệm về GIT

**Git** là một**hệ thống kiểm soát phiên bản phân tán code nguồn mở**. Nó được thiết kế để xử lý các dự án từ nhỏ đến lớn với tốc độ và hiệu quả cao. Nó được phát triển để điều phối công việc giữa các developer. Kiểm soát phiên bản cho phép chúng ta theo dõi và làm việc cùng với các thành viên trong nhóm của chúng ta tại cùng một không gian làm việc.

**2.1.2. Các câu lệnh GIT**

Một số câu lệnh đơn giản của GIT:

- Kiểm tra phiên bản của Git:  $ git --v

- Định cấu hình các biến cấu hình chung:

+ $ git config --g user.name "Dev name"

+ $ git config --g user.email "Dev email"

+ $ git config --list

- Nếu bạn cần hỗ trợ, hãy sử dụng các lệnh:

+ $ git help -a

+ $ git help --all (Hướng dẫn bạn có thể làm được những gì, tất cả các lệnh có thể)

+ $ git config --help

+ $ git help config - (Đưa bạn tới trang hướng dẫn chính thống của Git)

+ $ git command -help . (Xem tất cả các tùy chọn có sẵn cho lệnh cụ thể)

- Khởi tạo một thư mục dự án : git init

- Thêm các file này vào staging area: git add <filename\_one>

- Chạy lệnh sau để commit các thay đổi của bạn đối với các file này: git commit <filename\_one>

- Push các thay đổi : git push

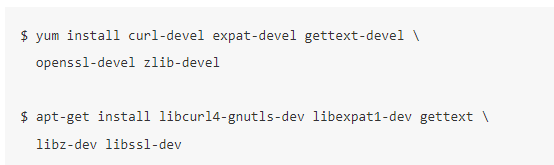
**2.1.3. Ứng dụng của GIT**

### **Cài đặt git trên máy tính cá nhân**

Để sử dụng Git đương nhiên chúng ta sẽ phải cài đặt Git lên máy tính cá nhân của chúng ta để thao tác. Bản chất của việc cài đặt Git là chúng ta sẽ có một Server Git Local để thao tác.

##### Cài đặt từ mã nguồn

Để cài đặt được Git, bạn cần có các thư viện mà Git sử dụng như sau: curl, zlib, openssl, expat, và libiconv. Ví dụ như bạn đang sử dụng một hệ điều hành có sử dụng yum (như Fedora) hoặc apt-get (như các hệ điều hành xây dựng trên nền Debian), bạn có thể sử dụng một trong các lệnh sau để cài đặt tất cả các thư viện cần thiết:



Khi đã cài đặt xong tất cả các thư viện cần thiết, bước tiếp theo là tải về phiên bản mới nhất của Git từ website của nó:



Sau đó, dịch và cài đặt:



Sau khi thực hiện xong các bước trên, bạn cũng có thể tải về các bản cập nhật của Git dùng chính nó như sau:



##### Cài Đặt Trên Linux

Nếu như bạn muốn cài đặt Git trên Linux thông qua một chương trình cài đặt, bạn có thể làm việc này thông qua phần mềm quản lý các gói cơ bản đi kèm với hệ điều hành của bạn. Nếu bạn đang sử dụng Fedora, bạn có thể dùng yum:



Còn nếu bạn đang sử dụng một hệ điều hành dựa trên nhân Debian như Ubuntu, hãy dùng apt-get:



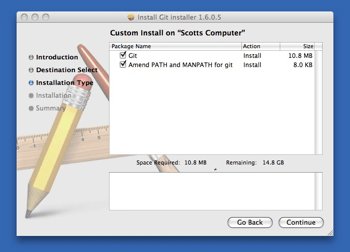
Với CentOS



##### Cài Đặt Trên Mac

Có hai cách đơn giản để cài đặt Git trên Mac. Cách đơn giản nhất là sử dụng chương trình cài đặt có hỗ trợ giao diện, bạn có thể tải về từ trang web của Google Code



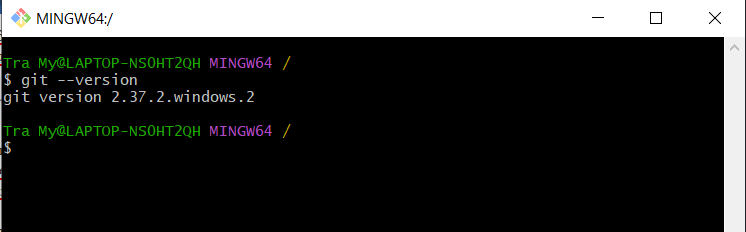


Cách khác để cài đặt Git là thông qua MacPorts. Nếu như bạn đã cài đặt MacPorts, Git có thể được cài đặt sử dụng lệnh sau:



##### Cài Đặt Trên Windows

Cài đặt Git trên Windows rất đơn giản. Mặc định bạn sẽ làm việc với Git ở giao diện dòng lệnh cmd, đối với Git trên Windows mà bạn cài như trên bạn cũng có thể sử dụng Git Bash - git-bash.exe trong thư mục C:\Program Files\Git, để mở giao diện dòng lệnh tiện ích hơn là cmd mặc định của Windows. Nó có giao diện như dưới



Các tác vụ với Git thực hiện bằng lệnh git với tham số thích hợp.

Lệnh đầu tiên là hãy kiểm tra Version của Git đang dùng:



Hiện thị các lệnh Git bằng cách gõ:



### **Sử dụng các câu lệnh git**

Dưới đây là danh sách các câu lệnh Git cơ bản và phổ biến được sử dụng để quản lý phiên bản mã nguồn và tương tác với repository trên GitHub.

**Cấu hình ban đầu**

1. **Cấu hình tên người dùng và email:**

git config --global user.name "Your Name"

git config --global user.email "your.email@example.com"

**Khởi tạo và quản lý repository**

1. **Khởi tạo một repository mới:**

git init

1. **Clone một repository từ GitHub:**

git clone https://github.com/username/repository.git

**Quản lý tệp và thư mục**

1. **Thêm tệp vào staging area:**

git add <file> # Thêm một tệp cụ thể

git add . # Thêm tất cả các tệp đã thay đổi

1. **Kiểm tra trạng thái hiện tại của repository:**

git status

1. **Bỏ qua các tệp hoặc thư mục không cần thiết:**

Tạo tệp .gitignore và thêm các tệp hoặc thư mục mà bạn muốn bỏ qua.

Commit thay đổi

1. **Tạo commit với thông điệp mô tả:**

git commit -m "Your commit message"

1. **Commit với tất cả thay đổi trong một bước:**

git commit -a -m "Your commit message"

**Làm việc với các nhánh (branches)**

1. **Tạo và chuyển đến một nhánh mới:**

git checkout -b <branch-name>

1. **Chuyển đến nhánh khác:**

git checkout <branch-name>

1. **Liệt kê các nhánh hiện có:**

git branch

1. **Xóa nhánh:**

git branch -d <branch-name>

**Kết nối với remote repository**

1. **Thêm remote repository:**

git remote add origin https://github.com/username/repository.git

1. **Kiểm tra remote repository:**

git remote -v

**Push và Pull**

1. **Push các commit lên remote repository:**

git push -u origin master # Lần đầu tiên push

git push # Các lần sau

1. **Pull các thay đổi từ remote repository:**

git pull

**Merge và Rebase**

1. **Merge một nhánh vào nhánh hiện tại:**

git merge <branch-name>

1. **Rebase một nhánh:**

git rebase <branch-name>

# **Chương 2: Các công cụ phát triển phần mềm**

## **2.5. Giới thiệu về công cụ remote**

### **2.5.1. Giới thiệu về VPN**

**Khái niệm:**

VPN hay Mạng riêng ảo tạo ra kết nối mạng riêng tư giữa các thiết bị thông qua Internet. VPN được sử dụng để truyền dữ liệu một cách an toàn và ẩn danh qua các mạng công cộng. VPN hoạt động bằng cách ẩn địa chỉ IP của người dùng và mã hóa dữ liệu để chỉ người được cấp quyền nhận dữ liệu mới có thể đọc được.

**Công dụng của VPN:**

Dịch vụ VPN chủ yếu được sử dụng để gửi dữ liệu một cách an toàn qua Internet. 3 chức năng chính của VPN là:

**1. Quyền riêng tư**

Nếu không có mạng riêng ảo, dữ liệu cá nhân của bạn như mật khẩu, thông tin thẻ tín dụng và lịch sử duyệt web có thể bị ghi lại và rao bán bởi các bên thứ ba. VPN sử dụng mã hóa để giữ bí mật những thông tin này, đặc biệt là khi bạn kết nối qua mạng Wi-Fi công cộng.

**2. Tính ẩn danh**

Địa chỉ IP chứa thông tin về vị trí và hoạt động duyệt web của bạn. Tất cả các trang web trên Internet theo dõi dữ liệu này bằng cookie và công nghệ tương tự. Họ có thể nhận dạng bạn bất cứ khi nào bạn ghé thăm trang web của họ. Kết nối VPN sẽ ẩn địa chỉ IP của bạn, để bạn được ẩn danh trên Internet.

**3. Bảo mật**

Dịch vụ VPN sử dụng mật mã để bảo vệ kết nối Internet của bạn khỏi những truy cập trái phép. VPN cũng có thể hoạt động như một cơ chế tắt, hủy bỏ các chương trình được chọn trước đó phòng khi có hoạt động đáng ngờ trên Internet. Việc này làm giảm khả năng dữ liệu bị xâm phạm. Những tính năng trên cho phép các công ty cấp quyền truy cập từ xa cho người dùng được ủy quyền thuộc mạng lưới kinh doanh của họ.

**4.Lợi Ích của VPN**

1. Bảo Mật Dữ Liệu: VPN mã hóa dữ liệu truyền tải, bảo vệ thông tin cá nhân và nhạy cảm khỏi bị đánh cắp.
2. Quyền Riêng Tư: VPN che giấu địa chỉ IP của bạn, ngăn chặn các trang web và nhà cung cấp dịch vụ Internet (ISP) theo dõi hoạt động trực tuyến của bạn.
3. Truy Cập Nội Dung Bị Chặn: VPN cho phép bạn vượt qua các giới hạn địa lý và truy cập nội dung bị chặn hoặc giới hạn ở quốc gia của bạn.
4. An Toàn Khi Sử Dụng Wi-Fi Công Cộng: VPN bảo vệ bạn khỏi các mối đe dọa khi sử dụng mạng Wi-Fi công cộng, như quán cà phê, sân bay hay khách sạn.

Loại Hình VPN:

1. VPN Cá Nhân: Thường được cá nhân sử dụng để bảo vệ quyền riêng tư và an ninh trực tuyến.
2. VPN Doanh Nghiệp: Được các doanh nghiệp sử dụng để bảo vệ dữ liệu nội bộ và cho phép nhân viên truy cập từ xa vào hệ thống mạng của công ty.
3. Chính Phủ và Tổ Chức: VPN được sử dụng để bảo vệ thông tin nhạy cảm và duy trì kết nối an toàn trong các tổ chức chính phủ và các tổ chức lớn.

Các Loại Hình VPN

1. Remote Access VPN: Kết nối người dùng từ xa tới mạng công ty, giúp truy cập tài nguyên mạng một cách an toàn.
2. Site-to-Site VPN: Kết nối hai hoặc nhiều mạng LAN tại các địa điểm khác nhau, thường được sử dụng bởi các doanh nghiệp lớn có nhiều chi nhánh.
3. Personal VPN: Dành cho người dùng cá nhân, bảo vệ quyền riêng tư và bảo mật khi truy cập Internet.

### 2.5.2. Giới thiệu về Teamviewer

TeamViewer là phần mềm dùng để điều khiển máy tính từ xa hay còn gọi là remote từ một máy tính khác. TeamViewer cho phép máy tính của người khác kết nối vào máy tính của bạn nhằm mục đích để trao đổi, chia sẻ dữ liệu và người khác có thể sử dụng máy tính của bạn từ xa. Phần mềm này nổi tiếng với lợi ích giúp bạn có thể nhận sự hỗ trợ từ các chuyên gia công nghệ nếu như bạn gặp sự cố với chiếc máy tính để bàn hoặc laptop của mình.

TeamViewer có các phiên bản dành cho máy tính, laptop và cũng có thể là điện thoại thông minh, hỗ trợ nhiều hệ điều hành khác nhau như Windows, MacOS, Linux, IOS, Android,… Chỉ cần nhập đúng mã liên kết cùng với mật khẩu được đối phương cung cấp là người dùng có thể nhận được sự hỗ trợ giải quyết các vấn đề trong học tập và công việc một cách trực tuyến mà không cần gặp mặt trực tiếp.

Với các chức năng ưu việt trên cùng với giao diện thân thiện, dễ dàng làm quen và sử dụng mà TeamViewer từ lâu đã trở thành phần mềm điều khiển máy tính từ xa phổ biến và được ưa chuộng hàng đầu trên toàn thế giới.

**Cách thức hoạt động Teamviewer**

Cách thức hoạt động của TeamViewer là thông qua mã định danh của máy tính (ID) và mật khẩu được cung cấp (Password) mỗi lần kết nối. Để sử dụng TeamViewer, bạn cần phải cài đặt phần mềm và có kết nối với Internet.

Khi khởi động TeamViewer lên thì ở giữa giao diện ứng dụng sẽ cho bạn biết về mã định danh của máy tính ở phần Your ID, còn mật khẩu kết nối nằm ở phần Password. Bạn chỉ cần gửi cho người bạn muốn kết nối vào máy tính mình hai đoạn mã trên là người đó có thể đăng nhập vào máy tính của bạn rất dễ dàng và nhanh chóng. Bạn cũng có thể làm điều tương tự với đối phương nếu như bạn có mã định danh máy tính và mật khẩu kết nối của người đó bằng cách sử dụng nút Connect.

Cần lưu ý rằng mã định danh của máy tính sẽ luôn luôn cố định nhưng mật khẩu kết nối sẽ thay đổi khác nhau ở mỗi lần kết nối.

Ưu điểm của Teamviewer

* Giao diện thân thiện giúp người dùng dễ dàng làm quen và sử dụng trong thời gian ngắn. các chứ năng trực năng, dễ hiểu và sắp xếp khoa học, thuận tiện thao tác.
* Kết nối nhanh chóng chỉ trong vòng vài giây đầu. Teamviewer có thể chạy sau tường lửa và tự động khi gặp phải những máy tính có cấu hình proxy
* Độ bảo mật cao của phần mềm. Với phương thức kết nối cần phải xác nhận từ hai chiều của thiết bị thong qua ID vs Password cùng với việc giới hạn số luồng truy cập cố định được kết nối với nhau, nói cách khác, chi khi bạn cho phép người khác kết nối vào máy tính của mình việc cung cấp ID vs Passvord thì người đó mới có thể thao tác trên máy tính của bạn được. Rủi ro chỉ xảy ra khi bạn bị người khác lừa, có nghĩa là không phải rủi ro về phần mềm
* Teamviewer là phần mềm miễn phí dành cho người dùng cá nhân sử dụng một các thoải mái
* Kết nối đa chiều giữa các thiết bị với nhau giữa máy tính với máy tính hoặc smartphone
* Hỗ trợ hơn 30 ngôn ngữ khác nhau trên toàn thế giới, trong đó có cả tiếng việt

Nhước điểm của Teamvewer

* Miễn phí đối với người dùng cá nhân nhưng đối với đối tượng người dùng doanh nghiệp và thương mại thì chi phí tương đối cao.
* TeamViewer hoạt động có ổn định hay không còn phụ thuộc vào chất lượng đường truyền Internet ở hai đầu thiết bị.
* Việc chia sẻ các tệp dữ liệu lớn qua TeamViewer có thể không ổn định, mất nhiều thời gian hoặc thậm chí không thể thực hiện thành công.

Hướng dẫn cách sử dụng TeamViewer

Bước 1: Sau khi cài đặt TeamViewer, hãy mở ứng dụng lên từ máy tính của bạn.

Bước 2: Giao diện TeamViewer hiện ra. Ở chính giữa có hai phần chính:

**Allow Remote Control**: Phần này sẽ hiển thị thông tin kết nối của bạn. Nếu bạn muốn người khác kết nối vào máy tính của mình thì hãy cung cấp thông tin này cho đối phương, bao gồm:

**Your ID**: mã định danh máy tính của bạn

**Password**: mật khẩu kết nối

**Control Remote Computer**: Phần này sẽ giúp bạn kết nối đến máy tính của một người khác bao gồm các tùy chọn sau: **Remote control** (điều khiển từ xa) và **File transfer** (chia sẻ tệp tin).

Ô **Enter Partner ID** là phần bạn sẽ nhập ID từ máy tính bạn muốn kết nối.

Bước 3: Để kết nối đến một máy tính khác, bạn nhập ID vào **Enter Partner ID.**

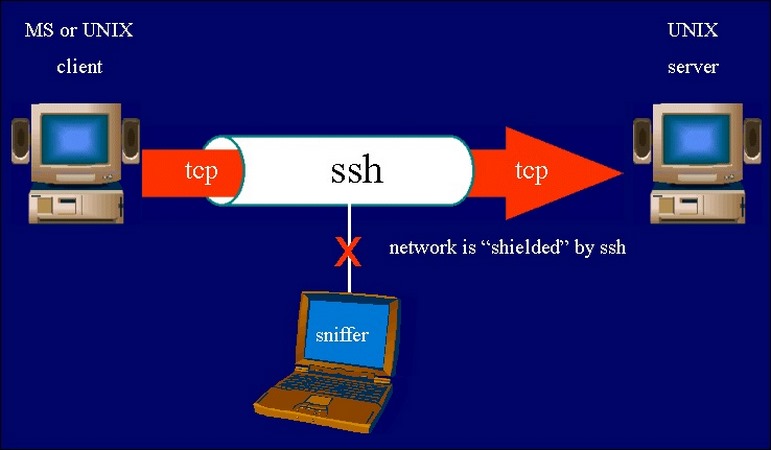
Bước 4: Nhấn **Connect**.

Bước 5: Hộp thoại **TeamViewer Authentication** hiện ra. Nhập mật khẩu kết nối vào ô **Password**. Sau đó nhấn **Log On**. Như vậy là bạn đã có thể thành công kết nối và có thể điều khiển máy tính đó.

### **2.5.3. Giới thiệu về SSH**

**Khái niệm về SSH:**

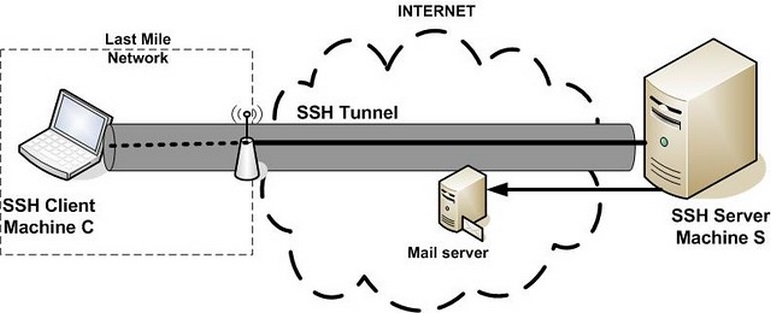
 SSH viết đầy đủ là Secure Shell, đây là một giao thức hỗ trợ các nhà quản trị mạng truy cập vào máy chủ từ xa thông qua mạng internet không bảo mật. Ngoài ra, SSH còn cung cấp các bộ tiện ích phục vụ phát triển chính giao thức SSH.



SSH tạo ra cơ chế xác thực qua mật khẩu mạnh, hình thành mối liên kết giao tiếp dữ liệu mã hóa ra giữa hai máy qua môi trường internet. Ngày nay giao thức SSH được giấy quản trị mạng sử dụng phổ biến trong quá trình quản lý, điều chỉnh ứng dụng từ xa. Nó cho phép vị tự đăng nhập vào mạng máy tính và thực hiện một số tác vụ cơ bản như dịch chuyển file.

-Cơ chế hoạt động của SSH:

Bạn sẽ chưa thể hiểu rõ SSH là gì nếu chưa nắm rõ cơ chế hoạt động của giao thức này. SSH gồm cả giao thức mạng và bộ tiện ích cơ bản để triển khai chính giao thức đó. Cụ thể, SSH ứng dụng ảnh mô hình client-server, kết nối với vùng hiển thị Session và vùng Session chạy.



Khi triển khai SSH hệ thống sẽ hỗ trợ cả giao thức ứng dụng, sử dụng cho trình giả lập Terminal hoặc truyền file. Trong thực tế, người ta còn sử dụng SSH để phát triển tunnel bảo mật cho các giao thức ứng dụng.

Mục đích SSH được tạo ra là để thay thế cho trình giả lập Terminal, cơ chế đăng nhập không an toàn (Telnet, Rlogin). Giao thức SSH hỗ trợ tính năng đăng nhập, khởi chạy Terminal Session thông qua hệ thống điều khiển từ xa.

Chức năng cơ bản nhất của giao thức SSH là liên kết với một host từ xa, ứng với một phiên Terminal bằng dòng lệnh "ssh server.example.org". Dòng lệnh này có thể liên kết Client với một máy chủ server.example.com thông qua ID người dùng UserName.

-Chức năng chính của SSH:

Giao thức đảm nhiệm khá nhiều chức năng trong hệ thống điều khiển, liên kết máy chủ. Các chức năng cơ bản phải kể đến như:

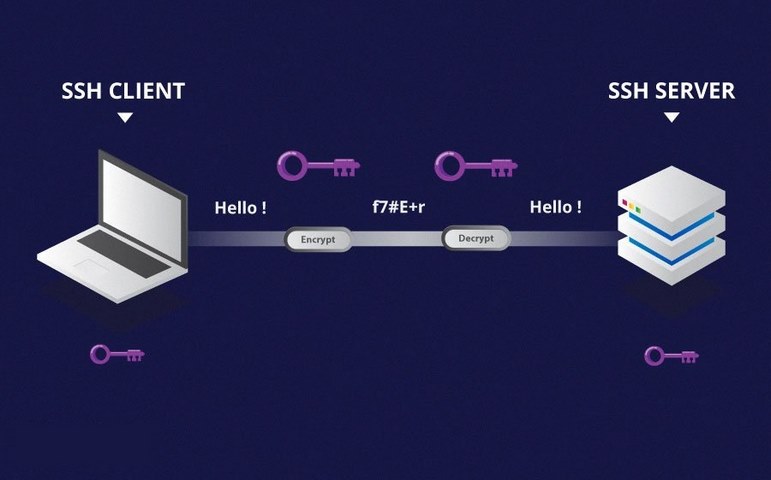
* Hỗ trợ truy cập từ xa vào những hệ thống, thiết bị ứng dụng giao thức SSH.
* Cho phép dịch chuyển file an toàn.
* Thực thi lệnh bảo mật, an toàn trên hệ thống điều khiển từ xa.
* Quản lý an toàn và hiệu quả thành phần hạ tầng mạng.

SSH có thể kết hợp với Terminal Session thay thế cho những chương trình Telnet có tính bảo mật thấp.

**-Kĩ thuật mã hoá trong SSH:**

**Mã hóa Symmetric Encryption**

Symmetric Encryption chính là một phương thức mã hóa ứng dụng Secret Key theo hai chiều, giải mã tin cho Host và Client. Như vậy, bất kỳ ai sở hữu mã khóa đều có khả năng giải mã tin nhắn trong quá trình truyền tin.



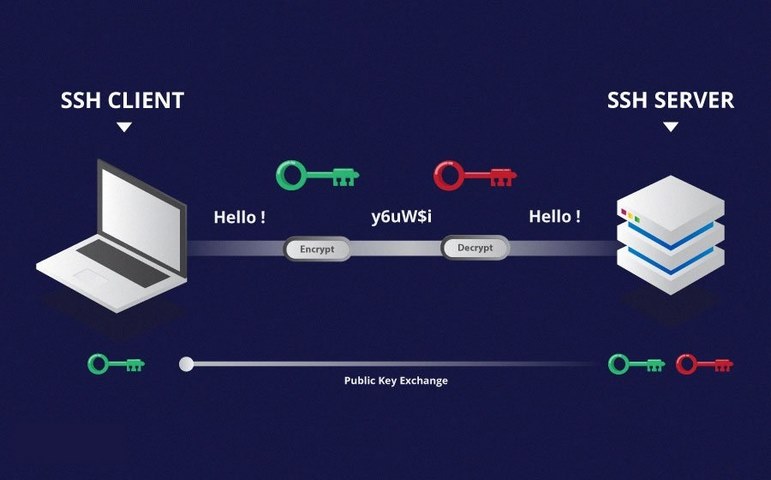
Symmetric Key được ứng dụng để mã hóa hoàn toàn phiên giao dịch diễn ra trong giao thức SSH. Trong đó, Host và Client có nhiệm vụ tạo Key bí mật, tuyệt đối không để lộ cho bên thứ ba.

Chính bởi Key không truyền tải giữa Client và Host nên thuật toán rất bảo mật. Cả hai máy tính có thể chia sẻ thông tin chung, ứng dụng chúng xác định mã Key bí mật. Bất kỳ máy tính khác có thể nắm bắt thông tin hay không, chúng cũng không dò được mã khóa bí mật.

Tuy nhiên cũng cần lưu ý rằng, Secret Token chỉ có thời hạn sử dụng trong một phiên SSH, nó hình thành từ chứng thực Client. Khi tạo mới Key, toàn bộ Packets giữa hai máy cần trải qua mã hóa bởi Private Key. Quá trình này gồm cả bước cung cấp mật khẩu bởi người dùng.

**Mã hóa Asymmetric Encryption:**

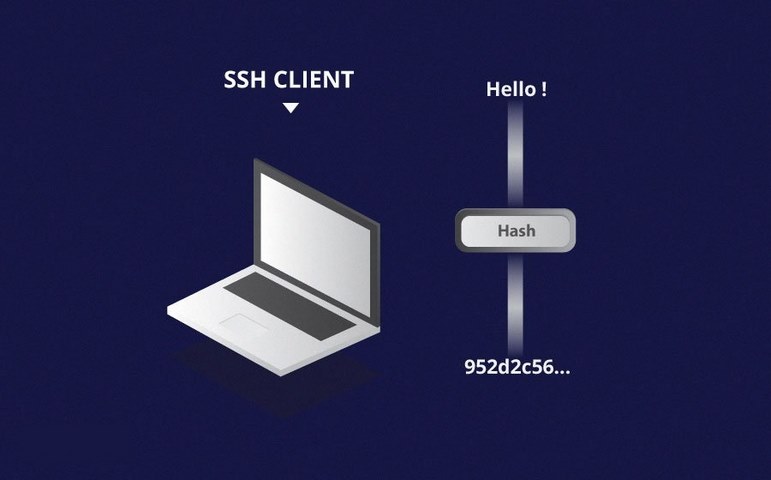
Khác với Symmetric Encryption, phương thức Asymmetric Encryption lại dùng 2 khóa riêng biệt để phục vụ mã hóa và giải mã. Bao gồm khóa công khai Public Key và khóa riêng tư Private Key, hình thành cặp khóa Public-private key pair.



Khóa Public Key công khai trên tất cả các thành phần liên quan. Tuy nhiên, nó cũng liên hệ trực tiếp với khóa riêng tư Private Key. Chính sự phụ thuộc này nên Public Key gần như không thể tự mã hóa thư, giải mã bất cứ thứ gì đã mã hóa bởi Private Key.

**Mã hóa Hashing:**

Hashing một chiều là phương thức mã hóa ứng dụng phổ biến trong Secure Shell Connection. Khác với Symmetric Encryption và Asymmetric Encryption, Hashing không sử dụng vào mục đích giải mã. Chúng hình thành sau mỗi lần nhập liệu, không thể khai thác. Như vậy, Hashing sẽ không thể quay lại để giải mã.

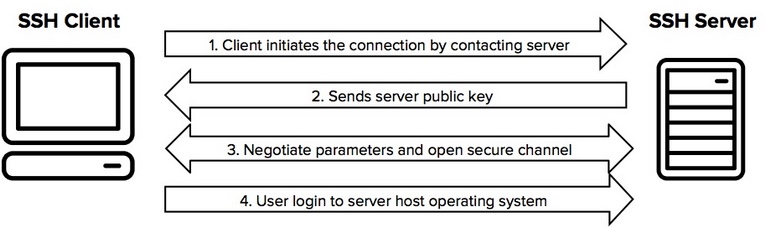


**Cách thức xử lý các dạng mã hóa trong giao thức SSH**

Trong phần tiếp theo của chuyên mục giải đáp SSH là gì, FPT Cloud sẽ giúp bạn giải thích sâu hơn về cách thức xử lý các dạng mã hóa. Như đã đề cập, SSH hoạt động theo mô hình Client - Server để chứng thực hai máy từ xa.

**Giai đoạn Session Encryption Negotiation:**

Để liên kết Client với Server thông qua TCP, Server cần xuất trình Encrytpion Protocal và các phiên bản hỗ trợ. Trường hợp Client có Protocol tương thích, phiên bản cũng phải đúng như vậy. Khi đó một thỏa thuận cũng đồng thời được khởi tạo, tiếp nhận Protocol. Mỗi Server còn sử dụng Symmetric Public Key để xác thực.



Mô tả hình ảnh trên:

* Client và Server phải đồng ý với nguyên tắc sử dụng seed value.
* Cả hai phía ra cần thiết được lưu ý thách thức mã hóa triển khai bởi seed value thông qua thuật đặc biệt. Đó là cơ chế tạo mã hóa, tác động lớn đến seed value.
* Cả hai bên tạo ra Private Key.
* Private Key này chung thuật toán với mã hóa AES ứng dụng để tạo ra Public Key, phân phối đến máy còn lại.
* Cả hai bên có quyền sử dụng Private Key. Trong khi đó, Public Key của máy còn lại sẽ kết hợp với Public Key để tạo ra một Key chung.
* Khi cả hai đều có sẵn Shared Key, quá trình mã hóa Symmetric cho phiên SSH chính thức được khởi động.

**Giai đoạn chứng thực người dùng:**

Đây là bước cuối cùng để người dùng có thể truy cập vào server, thực hiện xác nhận thông tin trên chính hệ thống đăng nhập. Người dùng trước tiên cần nhập tên đăng nhập, sau đó là mật khẩu. Thông tin người dùng cung cấp tiếp tục được truyền đến một hệ thống bảo mật Symmetric, không cho bất kỳ bên thứ ba nào thu thập.

Mật khẩu mặc dù đã trải qua mạng mã hóa nhưng nếu người dùng không cẩn thận vẫn có thể bị lộ. Khi một bên nào đó thu thập thành công mật khẩu, tài khoản của người dùng dễ bị chiếm đoạt. Theo nhiều chuyên gia bảo mật, người dùng nên sử dụng SSH Key Pair, bộ khóa Asymmetric có chức năng xác thực thành viên không yêu cầu nhập mật khẩu.

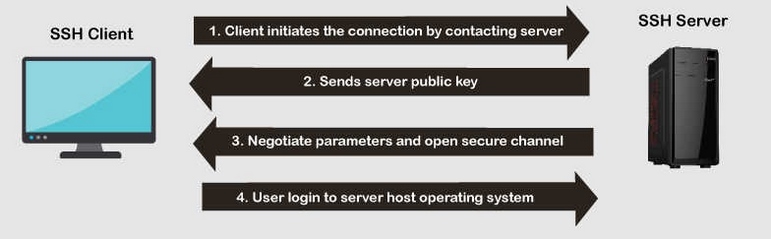
**Một vài vấn đề bảo mật của SSH**

Được đánh giá cao về tính an toàn nhưng SSH vẫn tồn tại một số nó học bảo mật máy tính. Cụ thể là tình trạng key lưu trữ trên Client dễ bị tích tụ, yêu cầu phải có nhân viên IT riêng để thực hiện khâu quản lý, dọn dẹp.

Mặt khác, dữ liệu lưu trong file SSH có thể bị lợi dụng chiếm đoạt quyền truy cập xác thực vào hệ thống từ xa. Vì thế mỗi doanh nghiệp cần phần xây dựng một quy trình lưu trữ, kiểm soát file cụ thể.

Phía nhà phát triển nên cẩn trọng khi kết hợp các lệnh hoặc hàm SSH trong Script hay những chương trình khác. Tuy rằng không khẩu để đưa lệnh SSH chứa ID người dùng cùng với password nhưng quá trình này dễ tạo lỗ hổng, tạo điều kiện cho hacker tấn công.

**SSH nên sử dụng khi nào?**



Trong mô hình TCP / IP, giao thức SSH sẽ hoạt động tại tầng thứ 4. Đối tượng nhiệm vụ tương tác duy trì kết nối giữa máy chủ và máy khách. Thông qua cơ chế mã hóa chuyên biệt, giao thức này có thể đảm bảo dữ liệu truyền đi an toàn. So với Telnet, Rlogin đã khắc phục yếu điểm tốt về mặt bảo mật.

**Sử dụng trong mọi datacenter:**

Giao thức SSH có mặt trong hầu hết datacenter và luôn đi kèm phần lớn Server Unix, Mac và Linux. Kết nối SSH hiệu ứng dụng phổ biến để xây dựng giao thức bảo mật cho hệ thống mấy cục bộ, Host từ xa.

Chẳng hạn như ứng dụng xây dựng quyền truy cập an toàn từ xa vào hệ thống tài nguyên, cập nhật phần mềm,.. Ngoài chức năng tạo đường dẫn an toàn cho máy cục bộ và Host từ xa, SSH cũng hỗ trợ quản lý Router, phần cứng máy chủ, một số nền tảng ảo hóa.

**Kết nối hệ thống server:**

SSH có phải là kết nối hệ thống server. Từ đó thực hiện các thay đổi, nâng cấp thông qua công cụ hoặc Terminal. SSH Key thử sử dụng để truy cập tập vào server trong cách tự động, ứng dụng chủ yếu trong Script, backup, công cụ quản lý cấu hình.

**Ứng dụng cho hệ thống đăng nhập một lần:**

Giao thức SSH hỗ trợ hoàn hảo để ứng dụng vào hệ thống đăng nhập một lần SSO. Theo đó, người dùng sẽ đăng nhập nhanh chóng, di chuyển qua lại giữa tài khoản mà không cần tốn thời gian nhập mật khẩu.

**Mã hóa dữ liệu:**

Không chỉ hỗ trợ xác thực kết nối mã hóa mà SSH traffic còn tham gia trực tiếp vào quá trình lão hóa. Khi người dùng chuyển file, duyệt web, nhập lệnh hay những tác vụ khác đều diễn ra an toàn.

**Xác thực thông tin:**

SSH kết hợp với ID người dùng và mật khẩu khi xác thực thông tin. Tuy vậy hiện nay, giao thức này chủ yếu hỗ trợ xác thực giữa hệ thống Host với nhau. Như vậy người dùng cá nhân vẫn phải sử dụng ID user và mật khẩu khi cần kết nối với Host từ xa.

Quá trình kết nối để thực hiện thông qua việc tạo một Public Key Pair ứng với từng Host. Trong đó, mỗi Session lại yêu cầu hai Public Key Pair. Một Key phục vụ xác thực máy cục bộ từ xa, Key còn lại làm nhiệm vụ xác thực máy cục bộ trực tiếp.