SIP

# I.Khái Niệm

1. SIP

SIP (**Session Initiation Protocol**) là một giao thức được sử dụng trong việc truyền thông đa phương tiện thông qua mạng IP.

SIP là viết tắt của cụm từ “Session Initiation Protocal”, là một giao thức truyền tín hiệu ở tầng  ứng dụng  (application-level) và được định nghĩa bởi IETF. SIP được sử dụng để tạo và quản lý các session thông qua mạng IP. Thuật ngữ “Session” là một phiên làm việc trong quá trình trao đổi media (voice, video,….).

2.Mục Đích

SIP là một giao thức điều khiển application-layer có thể **thiết lập**,**Sửa đổi**, và **chấm dứt** các session đa phương tiện (hội nghị) như Các cuộc gọi điện thoại Internet.

SIP cũng có thể mời người tham gia Các session đã tồn tại. Media có thể Được thêm vào (và loại bỏ khỏi) một session hiện có.

SIP hỗ trợ năm khía cạnh của thiết lập và chấm dứt giao tiếp đa phương tiện:

**User location**: xác định hệ thống kết thúc sẽ được sử dụng cho giao tiếp;

**User availability**: xác định sự sẵn sàng của người được gọi

**User capabilities**: xác định phương tiện và thông số truyền thông được sử dụng;

**Session setup**: thiết lập các tham số session tại Cả hai bên gọi và bên được gọi;

**Session management**:bao gồm chuyển và chấm dứt session, sửa đổi các tham số session và gọi Dịch vụ.

**3**. **SIP Address**

Được gọi là SIP URI

Có dạng như sau:

<Sip:username:password@host:port>

Trong đó:

* đầu tiên bắt buộc phải có bổ ngữ sip.
* (không bắt buộc) gồm có :  username và password.
* host (bắt buộc)  phải khai báo để có thể trao đổi media session.

Ví dụ: sip:tinh.34632@112.167.4.9:2008.

# II. Các Thực Thể Trong SIP

1. Proxy, Proxy Server:

Một thực thể trung gian hoạt động như một server và client với mục đích đưa ra yêu cầu Thay mặt cho các client khác. Một server proxy chủ yếu đóng vai trò Vai trò định tuyến, có nghĩa là công việc của nó là đảm bảo rằng một request được gửi đến một thực thể khác "gần hơn" với mục tiêu cuối là người dùng.

Proxy Server trong SIP là một thực thể trung gian, vai trò chính của proxy là định tuyến, nghĩa là bảo đảm một request sẽ được gởi đến người dùng. Proxy Server có thể được dùng  để kiểm tra, thống kê, tìm đường truyền kế tiếp, hoặc một chính sách bắt buộc nào đó,….

2. Redirect Server:

là một user agent server Tạo ra phản hồi 3xx cho các request nhận được, hướng dẫn client liên hệ với một bộ URI thay thế.

3. Registrar:

là một máy chủ chấp nhận request REGISTER Và đặt thông tin mà nó nhận được trong những request đó vào Dịch vụ vị trí cho tên miền mà nó xử lý.

Registrar là một server mà chấp nhận các registration request từ các UA. Mỗi UA phải được đăng ký trước khi nó có thể gởi và nhận media .

4. User Agent (UA):

Một thực thể hợp lý có thể hoạt động như một user. User agent client và user agent server

5. User Agent Client (UAC):

Một UAC là một thực thể logic Tạo ra request và gửi nó cho UAS, nhận các response từ UAS.

6. User Agent Server (UAS):

Một UAS à một thực thể logic Tạo ra response cho một SIP request. response Chấp nhận, từ chối, hoặc chuyển hướng request.

7. Back-to-Back User Agent:

Back-to-Back User Agent (B2BUA) là một Thực thể logic nhận được yêu cầu và xử lý nó như một user agent server (UAS). Để xác định yêu cầu Cần được trả lời, nó hoạt động như một user agent client (UAC) và Tạo ra các yêu cầu. Không giống như một server proxy, nó duy dialog và phải tham gia vào tất cả các request được gửi trên dialog Nó đã được thành lập. Vì nó là một điểm nối của một UAC và UAS, //không có định nghĩa rõ ràng là cần thiết cho hành vi của nó.

8. Address-of-Record, Dialog

9. Forking :

Forking của SIP request có nghĩa là nhiều session media có thể được thực hiện từ 1 request duy nhất. Đây là 1 tính năng mạnh mẽ của SIP. Với SIP Forking, bạn có thể đăng ký 1 Address of Record với nhiều Contact Adress trong location service. Khi có 1 request gởi đến proxy để thực hiện cuộc gọi, nếu proxy tìm thấy nhiều địa chỉ contact  khác nhau thì nó sẽ đưa ra 1 thuật toán để cố gắng liên lạc với các địa chỉ contact khác nhau. Có 2 thuật toán được sử dụng :

* Sequential Forking: cố gắng gửi request lần lượt từng contact, contact này xong đến contact khác.
* Parallel Forking: sẽ gửi request tất cả các contact cùng lúc.

10.**Location Service:**

Location Service là một cơ sở dữ liệu mà chứa một tập danh sách được ánh xạ giữa Adresses of Record (AORs – địa chỉ công khai của UA) và Contact Addresses (địa chỉ IP của UA).

Khi Registrar nhận một registration request từ một UA, registrar sẽ đưa thông tin nó nhận được vào Location Service. Location service liên hệ với Proxy Servcers chỉ ra domain chính xác để lấy về thông tin địa chỉ mà có thể được thực hiện cuộc gọi đến người dùng.

# III.SIP message

1 SIP message có thể là 1 request hoặc 1 respones: bao gồm 1 start-line, các header fields, 1 dòng rỗng, và một hoặc nhiều message-body

1. **Response**

Status-Line: (start line): SIP-Version Status-Code Reason-Phrase

+ Status-Code : bộ 3 số …

+ Reason-Phrase: mô tả ngắn gọn về mã response

Response có 1 start-line là 1 **status code** (mã trạng thái) có 3 chữ số.

+ Số đầu tiên định nghĩa loại response, số đầu tiên là 1 đó là **provisonal response**

+ các số khác được gọi chung là **final response**.

* 2xx là **Successful response**
* còn lại là **Negative response.**

Trong SIP, 1 request có nhiều response. Bất kỳ request nào cũng phải có ít nhất 1 final repsonse và có thể có nhiều provisional responses.

Có 6 nhóm trong SIP Response

1xx: provisional đã nhận được request xử lý tiếp tục.

2xx: Sucessful Request đã được xử lý thành công.

3xx: Redirection Session sẽ được chuyển hướng đến địa chỉ khác.

4xx: Client error Request lỗi cú pháp và không được server xử lý.

5xx: Server eror lỗi server không xử lý được request hợp lệ.

6xx: Global Failure Request không được bất kỳ server nào xử lý.

Detail response

1. **1xx Response (Early dialogs) different with 100 trying**

Có thể có 1 hoặc nhiều 1xx trả về trước khi nhận final response .

Một phản hồi tạm thời để thông báo rằng máy chủ đang có một số hành động khác và chưa có phản hồi dứt khoát.

Khi server gửi 1xx response nghĩa là server cần nhiều hơn 200 ms để trả về response cuối cùng.

1xx không phải là phản hồi đáng tin???

1. 100 Trying

Xác nhận request đã được nhận bởi next-hop server. Stop UAC gửi INVITE.

100 Trying khác với những 1xx khác : vì nó không bao giờ chuyển tiếp ngược dòng bởi một stateful proxy

1. 180 Ringing

UE Thông báo cho User rằng UA đã nhận được INVITE. Câu trả lời này CÓ THỂ được sử dụng để bắt đầu quay trở lại cục bộ.

1. 181 call Is Being Forwarded  
   Cuộc gọi đang được chuyển đến một destination khác
2. 182 Queued

Phía nhận cuộc gọi tạm thời không có, server đẩy cuộc gọi vào hàng đợi thay vì reject nó. Khi callee avilable nó sẽ trả về phản hợp lí nhất. Có thể dùng để update trạng thái của queued cuộc gọi

1. 183 Session Progress

Truyền tải thông tin về tiến trình cuộc gọi, Reason-Phrase, header fields, or message body có thể được sử dụng để truyền tải chi tiết về tiến độ cuộc gọi

1. **200 OK Request thành công**

Mỗi response được phân biệt bởi tag parameter tại To header field, với mỗi định danh riêng biệt.

UAC phải generate một ACK request cho mỗi 2xx nhận được. Header fields của ACK được xây dựng giống như bất kì request nào được gửi bên trong dialog, với ngoại lệ của CSeg và header field liên quan tới authentication.

Via, To, From, Call-ID, Cseq header field phải giống với INVITE đã được sent trước đó, thông tin cũng phải tương tự,

Nếu 2xx chứa một đề nghị A gửi ACK phải trả về một câu trả lời. Nếu để nghị A trong 2xx không được chấp nhận, UAC phải tạo ra một câu trả lời có giá trị trong ACK và gửi một BYE ngay sau đó

**2. SIP Request**

Một Sip Request có start-line là một request line có dạng:

Request-Line = Method SP Request-URI SP SIP-Version

SIP Request có các method: INVITE, BYE, ACK, CANCEL, REGISTER, OPTIONS, SUBSCRIBE, NOTIFY, MESSAGE, REFER, PRACK, UPDATE, INFO, PUBLISH. (chia ra: thiết lập session (INVITE, ACK, CANCEL), terminate session (BYE), truy vấn server (OPTION) )

Request-URI là 1 SIP URI chỉ điểm đến mong muốn của Request.

SIP-Version: SIP/2.0

**a. Register**

**b.**

**3. Message Header:**

+ Request URI: Điểm đến của Request

+ Header

+ To

+ From

+Contact lấy từ đâu:

+ Sequence: là số tự nhiên, sinh ra ngẫu nhiên trong dialog đó

+Proxy là gì: chuyển tiếp các node: là chốt chặn, trạm kiểm tra,

+Route quyết định con đường đi, xác định node tiếp theo để message đi tới node đó

+ Via: quyết định con đường về, theo dõi tất cả các proxy mà request đã đi qua, được sử dụng để định tuyến phản hồi qua cùng một bộ proxy.

+ Service Route: khi register vào ... Thông báo cho P-CSCF của S-CSCF URI để được sử dụng cho các yêu cầu orinating

+ Record Route: inside dialog - modify: ví dụ: thay đổi thông tin dialog, request tương tự, vào đúng cái đã đi trước đó (Được sử dụng bởi các proxy để chỉ ra rằng nó muốn duy trì đường dẫn của một hộp thoại. Được sử dụng để định tuyến các yêu cầu tiếp theo trong hộp thoại được thiết lập) check RFC3261?

+ Max forward: số lần hạn chế để loại bỏ…

+ Path: Thông báo cho S-CSCF của P-CSCF URI để được sử dụng để chấm dứt các request.

+