**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**CƠ SỞ TẠI THÀNH PHỐ HỒ CHÍ MINH**

**KHOA VIỄN THÔNG II**

**--------oOo--------**

**ĐỒ ÁN**

**TỐT NGHIỆP ĐẠI HỌC**

**CHUYÊN NGÀNH: ĐIỆN TỬ - VIỄN THÔNG**

**HỆ ĐÀO TẠO: ĐẠI HỌC CHÍNH QUY**

*Đề tài:*

**NGHIÊN CỨU OPENSIPS**

**Mã số đề tài : 11406160077**

**Giáo viên hướng dẫn : ThS NGUYỄN XUÂN KHÁNH**

**Sinh viên thực hiện : LÊ NGUYỄN ANH TRUNG MSSV : 406160077**

**Lớp : Đ06VTA1**

**TP HỒ CHÍ MINH – 2011**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**CƠ SỞ TẠI THÀNH PHỐ HỒ CHÍ MINH**

**KHOA VIỄN THÔNG II**

**--------oOo--------**

**ĐỒ ÁN**

**TỐT NGHIỆP ĐẠI HỌC**

**CHUYÊN NGÀNH: ĐIỆN TỬ - VIỄN THÔNG**

**HỆ ĐÀO TẠO: ĐẠI HỌC CHÍNH QUY**

*Đề tài:*

**NGHIÊN CỨU OPENSIPS**

**Mã số đề tài: 11406160077**

Nội dung:

* Chương I: Tìm hiểu kĩ thuật VoIP
* Chương II: Tìm hiểu dự án OPENSIPS
* Chương III:Phân tích xử lí cuộc gọi trong OPENSIPS
* Chương IV:Xây dựng mạng VoIP dùng OPENSIPS

**Giáo viên hướng dẫn : ThS NGUYỄN XUÂN KHÁNH**

**Sinh viên thực hiện : LÊ NGUYỄN ANH TRUNG MSSV : 406160077**

**Lớp : Đ06VTA1**

**TP HỒ CHÍ MINH – 2011**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**

**CƠ SỞ THÀNH PHỐ HỒ CHÍ MINH Độc Lập - Tự Do - Hạnh Phúc ---------oOo-------- ---o0o---**

*Tp. Hồ Chí Minh, ngày……tháng……năm ………*

**PHIẾU NHẬN XÉT ĐỒ ÁN TỐT NGHIỆP HỆ ĐẠI HỌC**

(Dành cho người hướng dẫn – Biểu 2)

1. Tên đề tài tốt nghiệp : Nghiên cứu Opensips

Mã đề tài : 11406160077

1. Họ tên sinh viên thực hiện : Lê Nguyễn Anh Trung

MSSV : 406160077 Lớp : Đ06VTA1

1. Những ưu điểm chính của đồ án tốt nghiệp :
2. Nội dung thực hiện : ………………………………………………………………………...............................

……………………………………………………………………………………………

……………………………………………………………………………………………

1. Kết quả sản phẩm : ………………………………………………………………………...............................

……………………………………………………………………………………………

……………………………………………………………………………………………

1. Khả năng áp dụng : ………………………………………………………………………..............................

……………………………………………………………………………………………

……………………………………………………………………………………………

1. Hình thức trình bày : ……………………………………………………………………….............................

……………………………………………………………………………………………

……………………………………………………………………………………………

1. Những thiếu sót chính của đồ án tốt nghiệp :

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

1. Đề nghị : Được bảo vệ Bổ sung thêm để bảo vệ Không được bảo vệ

1. Đánh giá chung : Xuất sắc Giỏi Khá Trung bình Yếu , Điểm ……/10.

**CHỦ TỊCH HỘI ĐỒNG GIÁO VIÊN HƯỚNG DẪN**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**

**CƠ SỞ THÀNH PHỐ HỒ CHÍ MINH Độc Lập - Tự Do - Hạnh Phúc**

**---------oOo-------- ---o0o---**

*Tp. Hồ Chí Minh, ngày……tháng……năm ……*

**PHIẾU NHẬN XÉT ĐỒ ÁN TỐT NGHIỆP HỆ ĐẠI HỌC**

(Dành cho người đọc duyệt - Biểu 3)

1. Tên đề tài tốt nghiệp : Nghiên cứu OPENSIPS

Mã đề tài : 11406160077

1. Họ tên sinh viên thực hiện : Lê Nguyễn Anh Trung

MSSV : 406160077 Lớp : Đ06VTA1

1. Những ưu điểm chính của đồ án tốt nghiệp :
2. Nội dung thực hiện : …………………………………………………………………………………...............

……………………………………………………………………………………………

1. Kết quả sản phẩm : …………………………………………………………………………………...............

……………………………………………………………………………………………

1. Khả năng áp dụng : …………………………………………………………………………………...............

……………………………………………………………………………………………

1. Hình thức trình bày : …………………………………………………………………………………...............

……………………………………………………………………………………………

1. Những thiếu sót chính của đồ án tốt nghiệp :

……………………………………………………………………………………………

……………………………………………………………………………………………

1. Đề nghị : Được bảo vệ Bổ sung thêm để bảo vệ Không được bảo vệ
2. 3 câu hỏi sinh viên phải trả lời trước Hội đồng :
3. ………………………………………………………………………………………

………………………………………………………………………………………

1. ………………………………………………………………………………………

………………………………………………………………………………………

c) ………………………………………………………………………………………

………………………………………………………………………………………

1. Đánh giá chung : Xuất sắc Giỏi Khá Trung bình Yếu , Điểm ……/10.

(Ghi chú : Trong trường hợp thay đổi điểm chấm giáo viên phải ký tên xác nhận).

**CHỦ TỊCH HỘI ĐỒNG GIÁO VIÊN ĐỌC DUYỆT**

**LỜI CÁM ƠN**

****

*Em xin bày tỏ lòng biết ơn chân thành đến thầy Nguyễn Xuân Khánh, người đã tận tình hướng dẫn cho em hoàn thành luận văn này.*

*Xin cám ơn các thầy cô của Học viện công nghệ bưu chính viễn thông cơ sở TP HCM đã cung cấp kiến thức cho em trong suốt những học kỳ vừa qua, giúp em có nền tảng vững chắc để thực hiện luận văn này.*

*Cuối cùng em không quên gửi lời cảm ơn đến gia đình, bạn bè, những người đã động viên, giúp đỡ em rất nhiều trong quá trình hoàn thành luận văn.*

*Cảm ơn tất cả mọi người.*

TP HCM 12/2010

Lê Nguyễn Anh Trung

**mỤC LỤC**

[**CHƯƠNG I: TÌM HIỂU KỸ THUẬT VOIP** 1](#_Toc282462001)

[1.1 Giới thiệu về voip 1](#_Toc282462002)

[1.1.1 VoIP là gì 1](#_Toc282462003)

[1.1.2 Phương thức hoạt động 1](#_Toc282462004)

[1.1.3 Các kiểu kết nối sử dụng VoIP 1](#_Toc282462005)

[1.1.4 Các thành phần trong mạng VoIP 2](#_Toc282462006)

[1.1.5 Các giao thức báo hiệu phổ biến trong VoIP 2](#_Toc282462007)

[1.2 Đặc tính của voip 3](#_Toc282462008)

[1.2.1 Ưu điểm 3](#_Toc282462009)

[1.2.2 Nhược điểm 3](#_Toc282462010)

[1.2.3 Yêu cầu chất lượng đối với VoIP 4](#_Toc282462011)

[1.3 Tổng quan về giao thức sip 4](#_Toc282462012)

[1.3.1 Tổng quan về giao thức khởi tạo phiên SIP 4](#_Toc282462013)

[1.3.2 Cấu trúc của SIP 5](#_Toc282462014)

[1.3.3 Hoạt động và các bản tin của SIP 6](#_Toc282462015)

[1.3.3.a Địa chỉ của SIP 6](#_Toc282462016)

[1.3.3.b Định vị server SIP 7](#_Toc282462017)

[1.3.3.c Định vị người dùng 7](#_Toc282462018)

[1.3.3.d Thay đổi một phiên đang tồn tại 7](#_Toc282462019)

[1.3.3.e Các bản tin của SIP 7](#_Toc282462020)

[1.3.3.f Tiêu đề bản tin 8](#_Toc282462021)

[1.3.3.g Bản tin yêu cầu 10](#_Toc282462022)

[1.3.3.h Bản tin đáp ứng 11](#_Toc282462023)

[1.3.4 Thiết lập và hủy cuộc gọi SIP 14](#_Toc282462024)

[1.4 Tính năng của sip 16](#_Toc282462025)

[**CHƯƠNG II: TÌM HIỂU DỰ ÁN OPENSIPS** 18](#_Toc282462026)

[2.1 Giới thiệu opensips 18](#_Toc282462027)

[2.2 Đặc điểm opensips 18](#_Toc282462028)

[2.2.1 Giao diện module PLUG and PLAY 18](#_Toc282462029)

[2.2.2 Hỗ trợ ENUM 19](#_Toc282462030)

[2.2.3 Hỗ trợ thoại 19](#_Toc282462031)

[2.2.4 Chức năng Load-Balancer 19](#_Toc282462032)

[2.2.5 NAT traversal 20](#_Toc282462033)

[2.2.6 Định tuyến với chi phí thấp nhất (Least cost routing) 20](#_Toc282462034)

[2.2.7 Hỗ trợ SRV và NAPTR DNS 20](#_Toc282462035)

[2.2.8 Call Processing Language (CPL) 21](#_Toc282462036)

[2.2.9 XCAP hỗ trợ cho các Presence Agent 21](#_Toc282462037)

[2.2.10 Giao diện quản lí và cơ sở dữ liệu 22](#_Toc282462038)

[2.2.11 Linh hoạt và mạnh mẽ về ngôn ngữ lập trình 22](#_Toc282462039)

[2.2.12 XMPP gateway 22](#_Toc282462040)

[2.2.13 Gateway to SMS 22](#_Toc282462041)

[2.2.14 IP blacklist 23](#_Toc282462042)

[2.2.15 Xác nhận, ủy quyền,thống kê 23](#_Toc282462043)

[2.2.16 Các giao thức vận chuyển 23](#_Toc282462044)

[2.2.17 Khả năng nâng cấp OpenSIPS 23](#_Toc282462045)

[2.3 Ứng dụng của opensips 24](#_Toc282462046)

[2.3.1 Ứng dụng trong dịch vụ VoIP 24](#_Toc282462047)

[2.3.2 Ứng dụng trong các doanh nghiệp 25](#_Toc282462048)

[2.3.3 SIP trunking 25](#_Toc282462049)

[**CHƯƠNG III: PHÂN TÍCH XỬ LÍ CUỘC GỌI TRONG OPENSIPS** 26](#_Toc282462050)

[3.1 Core and modules 26](#_Toc282462051)

[3.2 Các thành phần trong tệp tin opensips.cfg 26](#_Toc282462052)

[3.3 Quá trình xử lí bản tin trong tệp tin opensips.cfg 27](#_Toc282462053)

[3.3.1 SIP proxy 28](#_Toc282462054)

[3.3.2 Hoạt động của Stateful 28](#_Toc282462055)

[3.3.3 Scripting OpenSIPs : 29](#_Toc282462056)

[3.3.4 Listen interfaces: 29](#_Toc282462057)

[3.3.5 Logging 29](#_Toc282462058)

[3.3.6 Số lượng process 30](#_Toc282462059)

[3.3.7 Các thông số khác 31](#_Toc282462060)

[3.3.8 Modules và các thông số của chúng. 31](#_Toc282462061)

[3.3.9 Các script cơ bản. 34](#_Toc282462062)

[3.3.10 Các hàm của lõi. 34](#_Toc282462063)

[3.3.11 Các giá trị của lõi. 34](#_Toc282462064)

[3.3.12 Các biến giả. 35](#_Toc282462065)

[3.3.13 Các biến script 35](#_Toc282462066)

[3.3.14 Tổng quan Attribute-Value Pair (AVP) 36](#_Toc282462067)

[3.3.15 Flag 36](#_Toc282462068)

[3.4 Cơ bản định tuyến 38](#_Toc282462069)

[3.4.1 Định tuyến bản tin yêu cầu và phản hồi. 38](#_Toc282462070)

[3.4.2 Các bản tin yêu cầu đầu tiên và sau đó. 40](#_Toc282462071)

[3.4.3 Các đoạn Script định tuyến 41](#_Toc282462072)

[**CHƯƠNG IV: XÂY DỰNG MẠNG VOIP DÙNG OPENSIPS** 47](#_Toc282462073)

[4.1 Cài đặt opensips 47](#_Toc282462074)

[4.1.1 Hỗ trợ hệ điều hành và các gói phụ thuộc 47](#_Toc282462075)

[4.1.2 Các bước cài đặt OpenSIPS trên Linux Ubuntu 47](#_Toc282462076)

[4.2 Thiết lập cuôc gọi từ pc – pc thông qua sip server 51](#_Toc282462077)

[4.2.1 Mô hình 52](#_Toc282462078)

[4.3 Phân tích cuộc gọi: 55](#_Toc282462079)

[KẾT LUẬN VÀ HẠN CHẾ, HƯỚNG MỞ CỦA ĐỀ TÀI 65](#_Toc282462080)

[TÀI LIỆU THAM KHẢO 66](#_Toc282462081)

**MỤC LỤC HÌNH VẼ**

[Hình 1.1 :Vị trí của SIP trong mô hình OSI. 5](#_Toc282461952)

[Hình 1.2 :Cấu trúc của SIP 5](#_Toc282461953)

[Hình 1.3 :Hoạt động Proxy Server 14](#_Toc282461954)

[Hình 1.4 :Hoạt động của Redirect Server. 15](#_Toc282461955)

[Hình 1.5 :Thiết lập và hủy cuộc gọi SIP. 16](#_Toc282461956)

[Hình 2.1 :OpenSIPS trong VoIP 24](#_Toc282461957)

[Hình 2.2 :OpenSIPS Load Balancer 25](#_Toc282461958)

[Hình 3.1 :Transaction và Dialog 27](#_Toc282461959)

[Hình 3.2 :Thông tin bản tin INVITE 39](#_Toc282461960)

[Hình 3.3 :Thông tin record-route 40](#_Toc282461961)

[Hình 3.4 :Nội dung tiêu đề của bản tin ACK 41](#_Toc282461962)

[Hình 4.1 :Mô hình mạng VoIP dùng OpenSIPS 51](#_Toc282461963)

[Hình 4.2 :Hình phần mềm điện thoại Xlite 4 52](#_Toc282461964)

[Hình 4.3 :Hình phần mềm điện thoại Ekiga 53](#_Toc282461965)

[Hình 4.4 :Hình đường đi các bản tin SIP 54](#_Toc282461966)

**MỤC LỤC BẢNG**

[Bảng 1.1 : Các tiêu đề bản tin. 9](#_Toc282461967)

[Bảng 1.2 : Giải thích một số tiêu đề chính của SIP 10](#_Toc282461968)

[Bảng 1.3 : Các bản tin yêu cầu của SIP. 11](#_Toc282461969)

[Bảng 1.4 : Các loại bản tin đáp ứng của SIP. 11](#_Toc282461970)

[Bảng 1.5 : Một số mã trạng thái được định nghĩa trong SIP. 13](#_Toc282461971)

[Bảng 2.1 : Một số module của OpenSIPS 19](#_Toc282461972)

[Bảng 3.1 :Các loại Flag 37](#_Toc282461973)

[Bảng 4.1 :Bảng lưu thông tin người dùng 55](#_Toc282461974)

[Bảng 4.2 :Thông tin người dùng trong bảng location 57](#_Toc282461975)

**TỪ VIẾT TẮT**

AAA Authentication, Authorization, Accounting

CPAN Comprehensive Perl Archive Network

CPL Call Processing Language

DNS Domain Name System

ENUM Electronic Numbering

HTTP Hypertext Transfer Protocol

IEFT Internet Engineering Task Force

ISDN Integrated Service Digital Network

ISUP Integrated Services Digital Network User Part

LCR Least cost routing

LDAP Lightweight Directory Access Protocol

MIME Multipurpose Internet Mail Extension

MGCP Media Gateway Control Protocol

MTU Maximum Transmission Unit

NAPTR Name Authority Pointer

NAT Network address translation

PSTN Public Switched Telephone Network

RADIUS Remote Authentication Dial In User Service

RSVP Resource Reservation Protocol

RTP Real-time transport Protocol)

RTSP Real Time Streaming Protocol)

SAP Session Advertisement Protocol

SCTP Stream Transmission Control Protocol

SDP Session Description Protocol

SIP Session Initiation Protocol

SMS Short Message Service

SRV Service Location

TCP Transmission Control Protocol

TLS Transport Layer Security

UDP User Datagram Protocol

UAC User Agent Client

UAS User Agent Server

URI Universal Resource Identifier

URL Universal Resource Locator

VoIP Voice over Internet Protocol

XCAP XML Configuration Access Protocol

XML Extensible Markup Language

XMPP The Extensible Messaging and Presence Protocol

**LỜI MỞ ĐẦU**

Trong thập kỉ qua, mạng chuyển mạch gói phát triển một cách nhanh chóng và tương lai sẽ thay thế mạng chuyển mạch kênh. Với những lí do như chi phí, chia sẽ đường truyền, cung cấp nhiều dịch vụ đa dạng, …mà nó dần chiếm lĩnh thị trường.

Một trong những dịch vụ nổi bật mà chuyển mạch gói đem lại đó là VoIP. Nó đang xâm nhập vào những doanh nghiệp nơi mà nhu cầu thoại diễn ra hằng ngày và với chi phí thoại lớn. Và với một tổng đài nội bộ như Asterisk là có thể đáp ứng nhu cầu của doanh nghiệp đó, nhưng với các nhà cung cấp dịch vụ VoIP thì lưu lượng báo hiệu thoại là một vấn đề đáng quan tâm. Nhiều thiết bị đã ra đời để xử lí lưu lượng này, trong đó đáng chú ý là OpenSIPS , một mã nguồn mở được xây dựng thực hiện các công việc của một SIP server có thể xử lí hàng ngàn cuộc gọi với độ tin cậy cao. Các bản tin SIP được xử lí một cách nhanh chóng và tùy chỉnh theo mong muốn của người dùng với sự tích hợp các module.

Mục tiêu của đề tài là nghiên cứu cách xử lí các bản tin của OpenSIPS gồm 4 chương:

*Chương I*: Tìm hiểu kĩ thuật VoIP và các định nghĩa cơ bản của giao thức SIP.

*Chương II*:Tìm hiểu dự án OpenSIPS với các chức năng và khả năng xử lí của nó.

*Chương III*: Phân tích xử lí cuộc gọi trong OpenSIPS qua các module và các script.

*Chương IV*: Xây dựng mạng VoIP với Các người dùng thực hiện cuộc gọi qua OpenSIPS.

Do thời gian có hạn cũng như giải pháp đang trong giai đoạn nghiên cứu nên luận văn không tránh khỏi những sai sót và hạn chế, rất mong sự chỉ bảo bổ sung của quý thầy cô và các bạn.

Sinh viên thực hiện

**Lê Nguyễn Anh Trung**

# CHƯƠNG I: TÌM HIỂU KỸ THUẬT VOIP

1. Giới thiệu về VoIP
   1. VoIP là gì

VoIP: Voice over Internet Protocol (VoIP) là một công nghệ cho phép truyền thoại sử dụng giao thức mạng IP, trên cơ sở hạ tầng sẵn có của mạng Internet. VoIP là một trong những công nghệ viễn thông đang được quan tâm nhất hiện nay không chỉ đối với các nhà khai thác, các nhà sản xuất mà còn cả với người sử dụng dịch vụ. VoIP có thể vừa thực hiện cuộc gọi thoại như trên mạng điện thoại kênh truyền thống (PSTN) đồng thời truyền dữ liệu trên cơ sở mạng truyền dữ liệu. Như vậy, nó đã tận dụng được sức mạnh và sự phát triển vượt bậc của mạng IP vốn chỉ được sử dụng để truyền dữ liệu thông thường.

* 1. Phương thức hoạt động

Với VoIP, tín hiệu thoại được số hóa, nén và đóng gói IP, sau đó truyền dẫn qua mạng IP. VoIP sử dụng phương pháp mã hóa làm tiết kiệm băng thông và tăng hiệu quả đường truyền, tăng lưu lượng phục vụ của mạng. Giao thức báo hiệu được sử dụng để thiết lập và kết thúc cuộc gọi, mang thông tin định vị user và thỏa thuận lưu lượng.

VoIP cho phép thực hiện cuộc gọi dùng máy tính qua mạng dữ liệu như Internet. VoIP chuyển đổi tín hiệu thoại từ điện thoại tương tự analog vào tín hiệu số (digital) trước khi truyền qua Internet, sau đó chuyển đổi ngược lại ở đấu nhận. Khi tạo một cuộc gọi VoIP dùng điện thoại với một bộ điều hợp, chúng ta sẽ nghe âm mời gọi, quay số sẽ xảy ra sau tiến trình này. VoIP có thể cũng sẽ cho phép tạo một cuộc gọi trực tiếp từ máy tính dùng loại điện thoại tương ứng hay dùng microphone.

VoIP cho phép tạo cuộc gọi đường dài qua mạng dữ liệu IP có sẵn thay vì phải được truyền qua mạng PSTN ( public switched telephone network). Ngày nay nhiều công ty đã thực hiện giải pháp VoIP của họ để giảm chi phí cho những cuộc gọi đường dài giữa nhiều chi nhánh xa nhau.

Mặc dù những khái niệm về VoIP là đơn giản, Tuy nhiên để thực hiện và ứng dụng VoIP là phức tạp. Để gửi voice, thông tin phải được tách biệt thành những gói (packet) giống như dữ liệu. Gói là những phần thông tin được chia nhỏ để dễ dàng cho việc gửi gói, cũng có thể dùng kĩ thuật nén gói để tiết kiệm băng thông, thông qua những tiến trình codec (compressor/de-compressor).

* 1. Các kiểu kết nối sử dụng VoIP

Computer to Computer: Với kênh truyền Internet có sẵn, là một dịch vụ miễn phí được sử dụng rộng rãi khắp nơi trên thế giới. Chỉ cần người gọi (caller) và người nhận ( receiver) sử dụng chung một VoIP service (Skype, MSN, Yahoo Messenger,…), 2 headphone + microphone, sound card . Cuộc hội thoại là không giới hạn.

Computer to phone: Là một dịch vụ có phí. Bạn phải trả tiền để có 1 account + software (VDC,Evoiz,Netnam,…). Với dịch vụ này một máy PC có kết nối tới một máy điện thoại thông thường ở bất cứ đâu ( tuỳ thuộc phạm vi cho phép trong danh sách các quốc gia mà nhà cung cấp cho phép). Người gọi sẽ bị tính phí trên lưu lượng cuộc gọi và khấu trừ vào tài khoản hiện có.

Phone to Phone: Là một dịch vụ có phí. Bạn không cần một kết nối Internet mà chỉ cần một VoIP adapter kết nối với máy điện thoại. Lúc này máy điện thoại trở thành một IP phone.

* 1. Các thành phần trong mạng VoIP

Các thành phần cốt lõi của một mạng VoIP bao gồm: Gateway, VoIP Server, IP network, End User Equipments

Gateway: là thành phần giúp chuyển đổi tín hiệu analog sang tín hiệu số (và ngược lại)

+ VoIP gateway : là các gateway có chức năng làm cầu nối giữa mạng điện thoại thường ( PSTN ) và mạng VoIP.

+ VoIP GSM Gateway: là các gateway có chức năng làm cầu nối cho các mạng IP, GSM và cả mạng analog.

VoIP server : là các máy chủ trung tâm có chức năng định tuyến và bảo mật cho các cuộc gọi VoIP .

Trong mạng H.323 chúng được gọi là gatekeeper. Trong mạng SIP các server được gọi là SIP server.

Thiết bị đầu cuối (End user equipments ) :

+ Softphone và máy tính cá nhân (PC) : bao gồm một headphone, một phần mềm và một kết nối Internet. Các phần mềm miễn phí phổ biến như Skype, Ekiga, GnomeMeeting, Microsoft Netmeeting, SIPSet, Windowns Messenger,..

+ Điện thoại truyền thông với IP adapter: để sử dụng dịch vụ VoIP thì máy điện thoại thông dụng phải gắn với một IP adapter để có thể kết nối với VoIP server. Adapter là một thiết bị có ít nhất 1 cổng RJ11 (để gắn với điện thoại) , RJ45 (để gắn với đường truyền Internet hay PSTN) và 1 cổng cắm nguồn.

+ IP phone : là các điện thoại dùng riêng cho mạng VoIP. Các IP phone không cần VoIP Adapter bởi chúng đã được tích hợp sẵn bên trong để có thể kết nối trực tiếp với các VoIP server.

* 1. Các giao thức báo hiệu phổ biến trong VoIP

Có rất nhiều loại giao thức dùng thực hiện dịch vụ VoIP, những giao thức báo hiệu (signaling) VoIP phổ biến là SIP và H323. Cả SIP và H323 đều cho phép người dùng thực hiện cùng công việc: để thiết lập giao tiếp cho những ứng dụng đa phương tiện (multimedia) như audio, video, những giao tiếp dữ liệu khác. H323 là một giao thức tương đối cũ, cấu trúc thì quá phức tạp, hỗ trợ các chức năng phần lớn là không cần thiết cho VoIP, do đó đòi hỏi chi phí cao và không hiệu quả và hiện đang được thay thế bởi giao thức [SIP](http://www.3cx.vn/voip-sip/sip.php). SIP đơn giản hơn, mềm dẻo linh hoạt hơn và hoạt động ở chế độ mở hơn so với H.323. Một trong những điểm ưu việt của SIP là nó ít phức tạp hơn rất nhiều và tương tự như giao thức HTTP / SMTP.

Vì vậy, hầu hết các thiết bị VoIP hiện có ngày nay đều theo chuẩn SIP. Chỉ có những thiết bị VoIP cũ theo chuẩn H323.

1. Đặc tính của VoIP
   1. Ưu điểm

Giảm chi phí: Đây là ưu điểm nổi bật của VoIP so với điện thoại đường dài thông thường. Chi phí cuộc gọi đường dài chỉ bằng chi phí cho truy nhập Internet. Một giá cước chung sẽ được thực hiện với mạng Internet và do đó tiết kiệm đáng kể các dịch vụ thoại và fax. Sự chia sẻ chi phí thiết bị và thao tác giữa những người sử dụng thoại và dữ liệu cũng tăng cường hiệu quả sử dụng mạng. Đồng thời kỹ thuật nén thoại tiên tiến làm giảm tốc độ bit từ 64Kbps xuống dưới 8Kbps, tức là một kênh 64Kbps lúc này có thể phục vụ đồng thời 8 kênh thoại độc lập. Như vậy, lý dó lớn nhất giúp cho chi phí thực hiện cuộc gọi VoIP thấp chính là việc sử dụng tối ưu băng thông.

Tích hợp nhiều dịch vụ: Do việc thiết kế cơ sở hạ tầng tích hợp nên có khả năng hỗ trợ tất cả các hình thức thông tin cho phép chuẩn hoá tốt hơn và giảm thiểu số thiết bị. Các tín hiệu báo hiệu, thoại và cả số liệu đều chia sẻ cùng mạng IP. Tích hợp đa dịch vụ sẽ tiết kiệm chi phí đầu tư nhân lực, chi phí xây dựng các mạng riêng rẽ.

Vấn đề quản lý băng thông: Trong PSTN, băng thông cung cấp cho một cuộc gọi là cố định. Trong VoIP, băng thông được cung cấp một cách linh hoạt và mềm dẻo hơn nhiều. Chất lượng của VOIP phụ thuộc vào nhiều yếu tố, quan trọng nhất là băng thông. Do đó không có sự bắt buộc nào về mặt thông lượng giữa các thiết bị đầu cuối mà chỉ có các chuẩn tuỳ vào băng thông có thể của mình, bản thân các đầu cuối có thể tự điều chỉnh hệ số nén và do đó điều chỉnh được chất lượng cuộc gọi.

Nâng cao ứng dụng và khả năng mở rộng: Thoại và fax chỉ là các ứng dụng khởi đầu cho VoIP, các lợi ích trong thời gian dài hơn được mong đợi từ các ứng dụng đa phương tiện (multimedia) và đa dịch vụ. Tính linh hoạt của mạng IP cho phép tạo ra nhiều tinh năng mới trong dịch vụ thoại. Đồng thời tính mềm dẻo còn tạo khả năng mở rộng mạng và các dịch vụ.

Tính bảo mật cao: VoIP được xây dựng trên nền tảng Internet vốn không an toàn, do đó sẽ dẫn đến khả năng các thông tin có thể bị đánh cắp khi các gói tin bị thu lượm hoặc định tuyến sai địa chỉ một cách cố ý khi chúng truyền trên mạng. Các giao thức SIP (Session Ineitiation Protocol – giao thức khởi đầu phiên) có thể thành mật mã và xác nhận các thông điệp báo hiệu đầu cuối. RTP (Real Time Protocol) hỗ trợ mã thành mật mã của phương thức truyền thông trên toàn tuyến được mã hoá thành mật mã đảm bảo truyền thông an toàn.

* 1. Nhược điểm

Chất lượng dịch vụ chưa cao: Các mạng số liệu vốn dĩ không phải xây dựng với mục đích truyền thoại thời gian thực, vì vậy khi truyền thoại qua mạng số liệu cho chất lượng cuộc gọi không được đảm bảo trong trường hợp mạng xảy ra tắc nghẽn hoặc có độ trễ lớn. Tính thời gian thực của tín hiệu thoại đòi hỏi chất lượng truyền dữ liệu cao và ổn định. Một yếu tố làm giảm chất lượng thoại nữa là kỹ thuật nén để tiết kiệm đường truyền. Nếu nén xuống dung lượng càng thấp thì kỹ thuật nén càng phức tạp, cho chất lượng không cao và đặc biệt là thời gian xử lý sẽ lâu, gây trễ.

Kỹ thuật phức tạp: Truyền tín hiệu theo thời gian thực trên mạng chuyển mạch gói là rất khó thực hiện do mất gói trong mạng là không thể tránh được và độ trễ không cố định của các gói thông tin khi truyền trên mạng. Để có được một dịch vụ thoại chấp nhận được, cần thiết phải có một kỹ thuật nén tín hiệu đạt được những yêu cầu khắt khe: tỉ số nén lớn (để giảm được tốc độ bit xuống), có khả năng suy đoán và tạo lại thông tin của các gói bị thất lạc... Tốc độ xử lý của các bộ Codec (Coder and Decoder) phải đủ nhanh để không làm cuộc đàm thoại bị gián đoạn. Đồng thời cơ sở hạ tầng của mạng cũng cần được nâng cấp lên các công nghệ mới như Frame Relay, ATM,... để có tốc độ cao hơn hoặc phải có một cơ chế thực hiện chức năng QoS (Quality of Service). Tất cả các điều này làm cho kỹ thuật thực hiện điện thoại IP trở nên phức tạp và không thể thực hiện được trong những năm trước đây.

* 1. Yêu cầu chất lượng đối với VoIP

Từ những nhược điểm chính của mạng chuyển mạch gói đã đặt ra những yêu cầu cho VoIP như sau:

Chất lượng thoại phải ổn định, độ trễ chấp nhận được.

Mạng IP cơ bản phải đáp ứng được những tiêu chí hoạt động khắt khe gồm giảm thiểu việc không chấp nhận cuộc gọi, mất mát gói và mất liên lạc. Điều này đòi hỏi ngay cả trong trường hợp mạng bị nghẽn hoặc khi nhiều người sử dụng chung tài nguyên của mạng cùng một lúc.

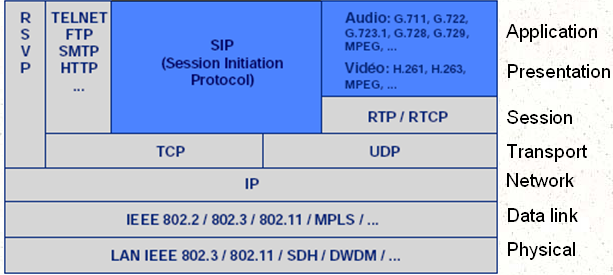
Việc báo hiệu có thể tương tác được với báo hiệu của mạng PSTN.

Quản lý hệ thống an toàn, địa chỉ hoá và thanh toán phải được cung cấp, tốt nhất là được hợp nhất với các hệ thống hỗ trợ hoạt động PSTN.

1. TỔNG QUAN VỀ GIAO THỨC SIP
   1. Tổng quan về giao thức khởi tạo phiên SIP

SIP ( Session Initiation Protocol) là giao thức điều khiển báo hiệu thuộc lớp ứng dụng trong mô hình OSI( hình II.1),được phát triển như là một chuẩn mở RFC 2543 của IEFT. Khác với H.323, nó dựa trên nguồn gốc Web (HTTP) và có thiết kế kiểu module, đơn giản và dễ dàng mở rộng với các ứng dụng thoại SIP. SIP là một giao thức báo hiệu để thiết lập, duy trì và kết thúc các phiên đa phương tiện như : thoại IP, hội nghị và các ứng dụng tương tự khác liên quan đến việc truyền thông tin đa phương tiện.

SIP được sử dụng kết hợp với các chuẩn giao thức IETF khác như là SAP, SDP và MGCP (MEGACO) để cung cấp một lĩnh vực rộng hơn cho các dịch vụ VoIP. Cấu trúc của SIP tương tự với cấu trúc của HTTP (giao thức client-server). Nó bao gồm các yêu cầu được gởi đến từ người sử dụng SIP client đến SIP server. Server xử lý các yêu cầu và đáp ứng đến client. Một thông điệp yêu cầu, cùng với các thông điệp đáp ứng tạo nên sự thực thi SIP.



1. : Vị trí của SIP trong mô hình OSI.

Nói chung, SIP hỗ trợ các hoạt động chính sau:

Định vị trí của người dùng.

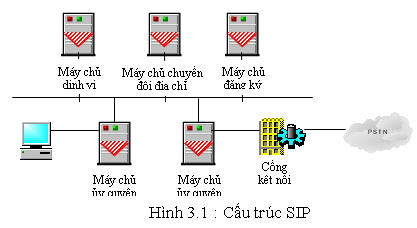
Định media cho phiên làm việc.

Định sự sẵn sàng của người dùng để tham gia vào một phiên làm việc.

Thiết lập cuộc gọi, chuyển cuộc gọi và kết thúc.

* 1. Cấu trúc của SIP

Các thành phần của SIP.



1. : Cấu trúc của SIP

Hệ thống SIP gồm 5 thành phần chính :

**User Agent (UA)** : đóng vai trò của thiết bị đầu cuối trong báo hiệu SIP , có thể là một máy điện thoại SIP, có thể là máy tính chạy phần mềm đầu cuối SIP... UA bao gồm hai loại User Agent Client (UAC) và User Agent Server (UAS). UAC khởi tạo cuộc gọi và UAS trả lời cuộc gọi. Điều này cho phép thực hiện cuộc gọi ngang hàng thông qua mô hình client-server.

**Máy chủ ủy quyền (Proxy server)** : là một chương trình trung gian, hoạt động như là một server và một client cho mục đích tạo các yêu cầu thay mặt cho các client khác. Các yêu cầu được phục vụ bên trong hoặc truyền chúng đến các server khác. Một proxy có thể dịch và nếu cần thiết có thể tạo lại các bản tin yêu cầu SIP trước khi chuyển chúng đến server khác hoặc một UA. Trong trường hợp này trường Via trong bản tin đáp ứng, yêu cầu chỉ ra các proxy trung gian tham gia vào tiến trình xử lý yêu cầu .

**Máy chủ định vị (Location Server)** : là phần mềm định vị thuê bao, cung cấp thông tin về những vị trí có thể của thuê bao bị gọi cho các phần mềm máy chủ ủy quyền và máy chủ chuyển đổi địa chỉ.

**Máy chủ chuyển đổi địa chỉ (Redirect Server**) : là phần mềm nhận yêu cầu SIP và chuyển đổi địa chỉ SIP sang một số địa chỉ khác và gửi lại cho đầu cuối. Không giống như máy chủ ủy quyền, máy chủ chuyển đổi địa chỉ không bao giờ hoạt động như một đầu cuối, tức là không gửi đi bất cứ yêu cầu nào. Máy chủ chuyển đổi địa chỉ cũng không nhận hoặc huỷ cuộc gọi.

**Máy chủ đăng ký (Register Server)** : là phần mềm nhận các yêu cầu đăng ký. Trong nhiều trường hợp máy chủ đăng ký đảm nhiệm luôn một số chức năng an ninh như xác nhận người sử dụng. Thông thường máy chủ đăng ký được cài đặt cùng với máy chủ ủy quyền và máy chủ hay địa chỉ hoặc cung cấp dịch vụ định vị thuê bao. Mỗi lần đầu cuối được bật lên ( ví dụ máy điện thoại hoặc phần mềm SIP) thì đầu cuối lại đăng ký với máy chủ. Nếu đầu cuối cần thông báo cho máy chủ về địa điểm của mình thì bản tin REGISTER cũng được gửi đi. Nói chung các đầu cuối đều thực hiện việc đăng ký lại một cách định kỳ.

* 1. Hoạt động và các bản tin của SIP
     1. Địa chỉ của SIP

Địa chỉ của SIP còn được gọi là bộ định vị tài nguyên chung URL (Universal Resource Locator), tồn tại dưới dạng user@host. Phần user trong phần địa chỉ có thể là tên người sự dụng hoặc số điện thoại. Phần host có thể là tên miền hoặc địa chỉ mạng.

Ví dụ địa chỉ SIP :

SIP: [ptithcm@ptithcm.edu.cn](mailto:ptithcm@ptithcm.edu.cn)

SIP: [406160077@203.162.17.8](mailto:406160077@203.162.17.8)

* + 1. Định vị server SIP

Khi client muốn gửi một yêu cầu, client gửi nó đến một proxy server SIP đã được cấu hình hoặc gửi yêu cầu đến địa chỉ IP và số cổng tương ứng với URL SIP. Gửi yêu cầu trực tiếp đến proxy server thì dễ dàng nếu ứng dụng cuối đã biết proxy server. Gửi yêu cầu theo cách thứ hai thì phức tạp hơn. Client phải cố gắng tiếp xúc với server ở số cổng được liệt kê trong bộ định vị tài nguyên đồng nhất URL SIP. Nếu số hiệu cổng không có trong URL SIP thì client sử dụng số cổng 5060. Nếu URL SIP chỉ định một giao thức (UDP hoặc TCP) thì client tiếp xúc với server sử dụng giao thức đó. Nếu không có giao thức nào được chỉ định hoặc nếu client không hỗ trợ UDP nhưng có hỗ trợ TCP thì nó cố gắng dùng TCP. Client cố gắng tìm một hoặc nhiều địa chỉ server SIP bằng cách truy vấn DNS (Domain Name System).

Tiến trình như sau:

Nếu phần host của URL SIP là địa chỉ IP, client tiếp xúc với server ở địa chỉ cho trước. Ngược lại nó xử lý bước kế tiếp.

Client truy vấn server DNS cho địa chỉ phần host của URL SIP. Nếu server DNS không trả về địa chỉ của URL SIP, client sẽ ngưng vì nó không thể định vị được server.

* + 1. Định vị người dùng

Người bị gọi có thể di chuyển giữa nhiều hệ thống đầu cuối theo thời gian.

Các vị trí này có thể đăng kí động với server SIP. Một server vị trí có thể trả về nhiều vị trí bởi vì người dùng đăng nhập ở nhiều trạm một cách đồng thời hoặc server vị trí có thông tin không chính xác. Server SIP kết hợp các kết quả để cung cấp một danh sách các vị trí hoặc không có vị trí nào.

Hoạt động nhận danh sách các vị trí thay đổi tùy thuộc vào SIP server. Một Redirect server trả về một danh sách hoàn chỉnh các vị trí và cho phép các client định vị người dùng chính xác. Một Proxy server cũng cố gắng định địa chỉ cho đến khi cuộc gọi thành công hoặc người bị gọi từ chối cuộc gọi.

* + 1. Thay đổi một phiên đang tồn tại

Trong một số trường hợp, người ta mong muốn thay đổi các thông số của một phiên đang tồn tại. Điều này được thưc hiện bằng cách phát lại bản tin INVITE, sử dụng cùng Call-ID, nhưng nội dung mới hoặc các trường tiêu đề mang thông tin mới. Chẳng hạn, hai đối tác đang trò chuyện và muốn thêm và một người thứ ba. Một trong hai mời người thứ ba với địa chỉ multicast mới và đồng thời gửi bản tin INVITE đến đối tác thứ hai với sự mô tả phiên mulicast mới, ngoại trừ số nhận dạng cũ.

* + 1. Các bản tin của SIP

Có hai loại bản tin SIP: bản tin yêu cầu được khởi tạo từ client và bản tin đáp ứng được trả lại từ server. Mỗi bản tin chứa một tiêu đề mô tả chi tiết về sự truyền thông.

SIP có thể sử dụng UDP. Khi được gửi trên UDP hoặc TCP, nhiều sự giao dịch SIP có thể được mang trên một kết nối TCP đơn lẻ hoặc gói dữ liệu UDP. Gói dữ liệu UDP (bao gồm tất cả các tiêu đề) thì không vượt quá đơn vị truyền dẫn lớn nhất MTU (Maximum Transmission Unit) nếu MTU được định nghĩa, hoặc không quá 1500 byte nếu MTU không được định nghĩa.

Một bản tin SIP cơ bản bao gồm: dòng bắt đầu (start-line), một hoặc nhiều trường tiêu đề, một dòng trống (CRLF) dùng để kết thúc các trường tiêu đề và một nội dung bản tin tùy chọn.

|  |  |
| --- | --- |
| Bản tin chung = | Dòng bắt đầu |
|  | Tiêu đề bản tin |
|  | CRLF |
|  | [ nội dung bản tin] |

* + 1. Tiêu đề bản tin

Tiêu đề bản tin dùng để chỉ ra người gọi, người bị gọi, đường định tuyến và loại bản tin của cuộc gọi. Có 4 nhóm tiêu đề bản tin như sau:

Tiêu đề chung: áp dụng cho các yêu cầu và các đáp ứng.

Tiêu đề thực thể: định nghĩa thông tin về loại bản tin và chiều dài.

Tiêu đề yêu cầu: cho phép client thêm vào các thông tin yêu cầu.

Tiêu đề đáp ứng: cho phép server thêm vào các thông tin đáp ứng.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tiêu đề chung** | **Tiêu đề thực thể** | **Tiêu đề yêu cầu** | **Tiêu đề đáp ứng** |
| Accept | Content-Encoding | Authorization | Allow |
| Accept-Encoding | Content-Length | Contact | Proxy-Authenticate |
| Accept-Language | Content-Type | Hide | Retry-After |
| Call-ID |  | Max-Forwards | Server |
| Contact |  | Organization | Unsupported |
| CSeq |  | Priority | Warning |
| Date |  | Proxy-Authorization | www-Authenticate |
| Encryption |  | Proxy-Require |  |
| Expires |  | Route |  |
| From |  | Require |  |
| Record-Route |  | Response-Key |  |
| Timestamp |  | Subject |  |
| To |  | User-Agent |  |
| Via |  |  |  |

1. : Các tiêu đề bản tin.

Giải thích một số tiêu đề chính của SIP:

|  |  |
| --- | --- |
| **Tiêu đề** | **Giải thích** |
| Call-ID | So khớp các yêu cầu với các đáp ứng tương ứng, nhận dạng duy  nhất lời mời hoặc sự đăng kí của client. |
| Cseq | Trong một cuộc gọi, Cseq tăng lên khi một yêu cầu mới được gửi đi và bắt đầu ở một giá trị ngẫu nhiên. Tuy nhiên, đối với yêu cầu ACK và Cancel thì Cseq không tăng. |
| To | Có mặt trong tất cả các yêu cầu và đáp ứng để chỉ ra nơi nhận  yêu cầu. |
| From | Có mặt trong tất cả yêu cầu và đáp ứng chứa tên và địa chỉ của  nơi khởi tạo yêu cầu. |
| Via | Ghi lại đường đi của yêu cầu để cho phép các server SIP trung  gian chuyển các câu trả lời trở lại cùng đường đi. |
| Encryption | Chỉ định nội dung và một số tiêu đề bản tin đã được mã hóa như  thế nào. |
| Content-Length | Chỉ ra kích thước của nội dung bản tin (tính bằng octet). |
| Content-type | Chỉ ra loại media của nội dung bản tin (văn bản/html,…). |
| Expires | Nhận dạng ngày và thời gian khi bản tin hết hạn . |
| Accept | Chỉ ra loại media nào được chấp nhận trong bản tin đáp ứng. |
| Subject | Cho thông tin về bản chất của cuộc gọi. |

1. : Giải thích một số tiêu đề chính của SIP
   * 1. Bản tin yêu cầu

Các yêu cầu cũng có thể được xem như các phương pháp cho phép User Agent và server mạng định vị, mời và quản lý các cuộc gọi. Bản tin yêu cầu SIP có dạng sau:

|  |  |
| --- | --- |
| Yêu cầu = | Dòng yêu cầu (Yêu cầu-line) |
|  | Tiêu đề chung/tiêu đề yêu cầu/tiêu đề thưc thể. |
|  | CRLF |
|  | [Nội dung bản tin] |

Dòng yêu cầu bắt đầu với mã phương pháp, bộ nhận dạng tài nguyên đồng nhất yêu cầu, phiên bản giao thức SIP và kết thúc với CRLF. Các thành phần được phân cách bởi kí tự SP.

Dòng yêu cầu = Method SP Request-URI SP SIP-Version CRLF.

Có 6 loại bản tin yêu cầu SIP:

INVITE, ACK, OPTIONS, BYE, CANCEL và REGISTER.

|  |  |
| --- | --- |
| **Bản tin** | **Ý nghĩa** |
| INVITE | Khởi tạo một phiên |
| ACK | Khẳng định rằng client đã nhận được bản tin đáp ứng cho bản tin INVITE |
| BYE | Yêu cầu kết thúc phiên |
| CANCEL | Huỷ yêu cầu đang nằm trong hàng đợi |
| REGISTER | Đầu cuối SIP đăng ký với registrar server |
| OPTIONS | Sử dụng để xác định năng lực của server |
| INFO | Sử dụng để tải các thông tin |

1. : Các bản tin yêu cầu của SIP.
   * 1. Bản tin đáp ứng

Các bản tin đáp ứng có dạng như sau:

|  |  |
| --- | --- |
| Đáp ứng = | Dòng trạng thái |
|  | Tiêu đề chung/tiêu đề đáp ứng/tiêu đề |
|  | CRLF |
|  | [nội dung bản tin] |

Dòng trạng thái bao gồm phiên bản của giao thức, mã trạng thái (số), lý do và CRLF. Các thành phần được cách nhau bằng hai kí tự SP.

Dòng trạng thái = SIP-version SP Status-Code SP Reason-Phrase CRLF

Mã trạng thái có 3 chữ số chỉ ra kết quả của việc đáp ứng yêu cầu. Lý do (Reason-Phrase) là sự mô tả ngắn gọn về mã trạng thái.

Chữ số đầu tiên của mã trạng thái định nghĩa lớp đáp ứng. SIP phiên bản 2.0 định nghĩa 6 giá trị cho lớp đáp ứng :

|  |  |
| --- | --- |
| **Bản tin** | **Ý nghĩa** |
| 1xx | Các bản tin chung |
| 2xx | Thành công |
| 3xx | Chuyển địa chỉ |
| 4xx | Yêu cầu không được đáp ứng |
| 5xx | Sự cố của server |
| 6xx | Sự cố toàn mạng |

1. : Các loại bản tin đáp ứng của SIP.

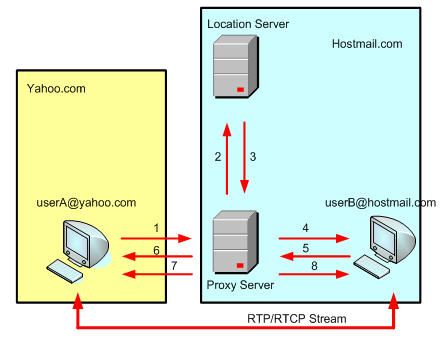
Một số mã trạng thái được định nghĩa trong SIP được trình bày trong bảng sau:

|  |  |  |
| --- | --- | --- |
| **Lớp đáp ứng** | **Mã trạng thái** | **Giải thích** |
| Thông tin | 100 | Đang cố gắng |
|  | 180 | Rung chuông |
|  | 181 | Cuộc gọi được chuyển |
|  | 182 | Được xếp hàng đợi |
| Thành công | 200 | OK |
| Đổi hướng | 300 | Nhiều chọn lựa |
|  | 301 | Được di chuyển thường xuyên |
|  | 302 | Được di chuyển tạm thời |
|  | 380 | Dịch vụ thay đổi |
| Lỗi client | 400 | Yêu cầu lỗi |
|  | 401 | Không nhận thực được |
|  | 402 | Yêu cầu trả tiền (payment required) |
|  | 403 | Cấm |
|  | 404 | Không tìm thấy |
|  | 405 | Bản tin không cho phép |
|  | 406 | Không chấp nhận |
|  | 407 | Yêu cầu nhận thực proxy |
|  | 408 | Yêu cầu timeout |
|  | 409 | Xung đột |
|  | 410 | Tiếp tục (gone) |
|  | 411 | Yêu cầu chiều dài |
|  | 413 | Thực thể yêu cầu quá lớn |
|  | 414 | URL yêu cầu quá lớn |
|  | 415 | Không hỗ trợ loại media |
|  | 420 | Mở rộng sai |
|  | 480 | Không sẵn có |
|  | 481 | Cuộc gọi hoặc sự trao đổi |
|  | 482 | Vòng lặp được phát hiện |
|  | 483 | Quá nhiều hop |
|  | 484 | Địa chỉ không hoàn thành |
|  | 485 | Mơ hồ |
|  | 486 | Đang bận |
| Lỗi server | 500 | Lỗi server bên trong |
|  | 501 | Không thực thi |
|  | 502 | Gateway lỗi |
|  | 503 | Dịch vụ không có sẵn |
|  | 504 | Gateway timeout |
|  | 505 | Phiên bản SIP không hỗ trợ |
| Lỗi toàn cầu | 600 | Bận ở mọi nơi |
|  | 603 | Từ chối |
|  | 604 | Không tồn tại ở mọi nơi |
|  | 606 | Không chấp nhận |

1. : Một số mã trạng thái được định nghĩa trong SIP.
   1. Thiết lập và hủy cuộc gọi SIP

Trước tiên ta tìm hiểu hoạt động của máy chủ ủy quyền và máy chủ chuyển đổi :

a. Hoạt động của máy chủ ủy quyền (Proxy Server)



1. Hình: Hoạt động Proxy Server

Client SIP [userA@yahoo.com](mailto:userA@yahoo.com) gửi bản tin INVITE cho [userB@hotmail.com](mailto:userB@hotmail.com) để mời tham gia cuộc gọi.

Các bước như sau:

Bước 1: [userA@yahoo.com](mailto:userA@yahoo.com) gửi bản tin INVITE cho userB ở miền hotmail.com, bản tin này đến proxy server SIP của miền hotmail.com (bản tin INVITE có thể đi từ Proxy server SIP của miền yahoo.com và được Proxy này chuyển đến Proxy server của miền hotmail.com).

Bước 2: Proxy server của miền hotmail.com sẽ tham khảo server định vị (Location server) để quyết định vị trí hiện tại của UserB.

Bước 3: Server định vị trả lại vị trí hiện tại của UserB (giả sử là [UserB@hotmail.com](mailto:UserB@hotmail.com)).

Bước 4: Proxy server gửi bản tin INVITE tới [userB@hotmail.com](mailto:userB@hotmail.com). Proxy server thêm địa chỉ của nó trong một trường của bản tin INVITE.

Bước 5: UAS của UserB đáp ứng cho server Proxy với bản tin 200 OK.

Bước 6: Proxy server gửi đáp ứng 200 OK trở về [userA@yahoo.com](mailto:userA@yahoo.com).

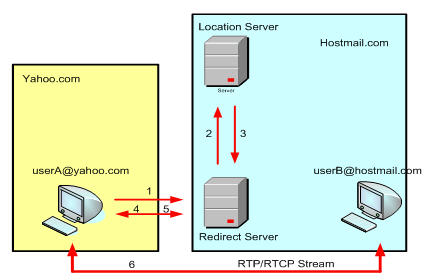
Bước 7: [userA@yahoo.com](mailto:userA@yahoo.com) gửi bản tin ACK cho UserB thông qua proxy server.

Bước 8: Proxy server chuyển bản tin ACK cho [userB@hostmail.com](mailto:userB@hostmail.com).

Bước 9: Sau khi cả hai bên đồng ý tham dự cuộc gọi, một kênh RTP/RTCP được mở giữa hai điểm cuối để truyền tín hiệu thoại.

Bước 10: Sau khi quá trình truyền dẫn hoàn tất, phiên làm việc bị xóa bằng cách sử dụng bản tin BYE và ACK giữa hai điểm cuối.

b. Hoạt động của máy chủ chuyển đổi địa chỉ (Redirect Server):



1. : Hoạt động của Redirect Server.

Các bước như sau:

Bước 1: Redirect server nhận được yêu cầu INVITE từ người gọi (Yêu cầu này có thể đi từ một proxy server khác).

Bước 2: Redirect server truy vấn server định vị địa chỉ của B.

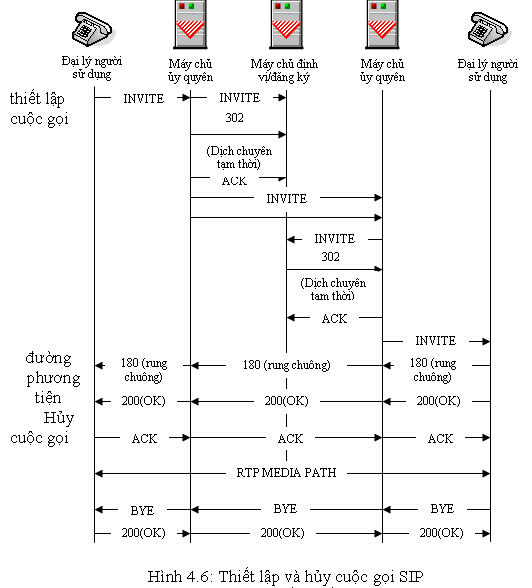
Bước 3: Server định vị trả lại địa chỉ của B cho Redirect server.

Bước 4: Redirect server trả lại địa chỉ của B đến người gọi A. Nó không phát yêu cầu INVITE như proxy server.

Bước 5: User Agent bên A gửi lại bản tin ACK đến Redirect server để xác nhận sự trao đổi thành công.

Bước 6: Người gọi A gửi yêu cầu INVITE trực tiếp đến địa chỉ được trả lại bởi Redirect server (đến B). Người bị gọi B đáp ứng với chỉ thị thành công (200 OK), và người gọi đáp trả bản tin ACK xác nhận, cuộc gọi được thiết lập. Ngoài ra SIP còn có các mô hình hoạt động liên mạng với SS7 (đếnPSTN) hoặc là liên mạng với chồng giao thức H.323.

**Tổng quát lại trong mạng SIP quá trình thiết lập và hủy một phiên kết nối:**



1. : Thiết lập và hủy cuộc gọi SIP.

Bước 1: Đăng ký, khởi tạo và định vị đầu cuối.

Bước 2: Xác định phương tiện của cuộc gọi, tức là mô tả phiên mà đầu cuối được mời tham gia.

Bước 3: Xác định mong muốn của đầu cuối bị gọi, trả lời hay không. Phía bị gọi phải gửi bản tin xác nhận chấp thuận cuộc gọi hay từ chối.

Bước 4: Thiết lập cuộc gọi.

Bước 5: Thay đổi hay điều khiển cuộc gọi (ví dụ như chuyển cuộc gọi).

Bước 6: Hủy cuộc gọi.

1. Tính năng của SIP

Giao thức SIP được thiết kế với những chỉ tiêu sau:

Tích hợp với các giao thức đã có của IETF: Các giao thức khác của IETF có thể xây dựng để xây dựng những ứng dụng SIP. SIP có thể hoạt động cùng với nhiều giao thức như :

RSVP (Resource Reservation Protocol) : Giao thức giành trước tài nguyên mạng.

RTP (Real-time transport Protocol) : Giao thức truyền tải thời gian thực.

RTSP (Real Time Streaming Protocol) : Giao thức tạo luồng thời gian thực.

SAP (Session Advertisement Protocol) : Giao thức thông báo trong phiên kết nối.

SDP (Session Description Protocol) : Giao thức mô tả phiên kết nối đa phương tiện.

MIME (Multipurpose Internet Mail Extension - Mở rộng thư tín Internet đa mục đích) : Giao thức thư điện tử.

HTTP (Hypertext Transfer Protocol) : Giao thức truyền siêu văn bản.

COPS (Common Open Policy Service) : Dịch vụ chính sách mở chung.

OSP (Open Settlement Protocol) : Giao thức thỏa thuận mở.

Đơn giản và có khả năng mở rộng SIP có rất ít bản tin, không có các chức năng thừa nhưng SIP có thể sử dụng để thiết lập những phiên kết nối phức tạp như hội nghị. Đơn giản, gọn nhẹ, dựa trên khuôn dạng văn bản, SIP là giao thức ra đời sau và đã khắc phục được điểm yếu của nhiều giao thức trước đây. Các phần mềm của máy chủ ủy quyền, máy chủ đăng kí, máy chủ chuyển đổi địa chỉ, máy chủ định vị… có thể chạy trên các máy chủ khác nhau và việc cài đặt thêm máy chủ hoàn toàn không ảnh hưởng đến các máy chủ đã có. Chính vì thế hệ thống chuyển mạch SIP có thể dễ dàng nâng cấp.

Hỗ trợ tối đa sự di động của đầu cuối. Do có máy chủ ủy quyền, máy chủ đăng ký và máy chủ chuyển đổi địa chỉ hệ thống luôn nắm được địa điểm chính xác của thuê bao. Ví dụ thuê bao với địa chỉ [ptit@vnpt.com.vn](mailto:ptit@vnpt.com.vn) có thể nhận được cuộc gọi thoại hay thông điệp ở bất cứ địa điểm nào qua bất cứ đầu cuối nào như máy tính để bàn, máy xách tay, điện thoại SIP… Với SIP rất nhiều dịch vụ di động mới được hỗ trợ.

Dễ dàng tạo tính năng mới cho dịch vụ và dịch vụ mới. Là giao thức khởi tạo phiên trong mạng chuyển mạch gói SIP cho phép tạo ra những tính năng mới hay dịch vụ mới một cách nhanh chóng. Ngôn ngữ xử lý cuộc gọi (Call Processing Language) và Giao diện cổng kết nối chung (Common Gateway Interface) là một số công cụ để thực hiện điều này. SIP hỗ trợ các dịch vụ thoại như chờ cuộc gọi, chuyển tiếp cuộc gọi, khóa cuộc gọi… (call waiting, call forwarding, call blocking…), hỗ trợ thông điệp thống nhất…

# CHƯƠNG II: TÌM HIỂU DỰ ÁN OPENSIPS

1. Giới thiệu OpenSIPS

OpenSIPS là một phiên bản mã nguồn mở hoàn chỉnh của một SIP server. OpenSIPS hơn một SIP proxy/ router vì nó bao gồm tính năng của lớp ứng dụng. OpenSIPS như một SIP server, là thành phần cốt lõi của mọi giải pháp VoIP dựa trên nền SIP. Là một thiết bị định tuyến có khả năng mềm dẻo và tùy biến, OpenSIPS tích hợp voice, video, IM (Instant Message) và các dịch vụ Presence với một phương pháp có hiệu quả cao, nhờ được thiết kế kiểu module nên dễ dàng nâng cấp. OpenSIPS mang lại các đặc tính đáng tin cậy và hiệu suất cao. OpenSIPS là một trong những SIP server nhanh nhất, là một giải pháp thông suốt cho các tổ chức kinh doanh hoặc lớp vận chuyển.

OpenSIPS bao gồm sự đa dạng hợp nhất, sự đa dạng xuất phát từ nhiều người tham gia vào việc phát triển dự án và do tính phức tạp và các tính năng phong phú của OpenSIPS (nó tích hợp và kết hợp chặt chẽ rất nhiều chức năng và tính năng, đứng trong Top thiết bị định tuyến SIP). Để đạt được độ ổn định và tin cậy cao, thì cần phải hợp nhất tính đa dạng này.

Dự án OpenSIPS là sự tiếp tục của dự án OpenSER. Kế thừa OpenSER tính mở rộng với cộng đồng và phát triển lên ( liên tục phát triển tăng cường và mở rộng code) . OpenSIPS tiếp tục và nâng cấp OpenSER lên.

Dựa trên những kinh nghiệm được cung cấp bởi dự án OpenSER, nên quá trình hợp nhất là quan trọng để đảm bảo kết quả tốt nhất.

1. Đặc điểm OpenSIPS
   1. Giao diện module PLUG and PLAY

OpenSIPS bao gồm hơn 70 module, mỗi module có một chức năng riêng, với giao diện module PLUG and PLAY thì OpenSIPS dễ dàng mở rộng và nâng cấp các tính năng của mà không ảnh hưởng tới phần lõi và càng làm tăng sự ổn định của hệ thống lõi. Số lượng và tính năng của các module được thay đổi theo từng phiên bản để tăng sự ổn định và tính mền dẻo trong OpenSIPS.

|  |  |
| --- | --- |
| Tên Module | Mô tả |
| CPL-C | Module Call Processing Language |
| ENUM | Module tra cứu ENUM |
| LCR | Module định tuyến chi phí ít nhất |
| LOAD\_BALANCER | Module cân bằng tải (cho các cuộc gọi) |
| MI\_FIFO | Hỗ trợ FIFO cho giao diện quản lý |
| NAT\_TRAVERSAL | Module thực hiện chức năng Nat traversal |
| PERL | Thực hiện của chức năng Perl |
| XCAP\_CLIENT | Hỗ trợ XCAP cho client |
| XMPP | Cho phép trao đổi tin nhắn tức thời giữa SIP client và XMPP(Jabber) client |

1. : Một số module của OpenSIPS
   1. Hỗ trợ ENUM

OpenSIPS hỗ trợ ENUM để chuyển đổi số điện thoại thường thành điạ chỉ SIP nhờ sử dụng DNS. Nhờ đó người sử dụng SIP vẫn có thể sử dụng số điện thoại thường để liên lạc để liên lạc với nhau.

Tính năng này sẽ có thể tự động tra cứu số điện thoại để xác định xem chúng có trùng khớp với địa chỉ SIP đã biết, cho phép cuộc gọi hoàn tất trên Internet (thay vì chuyển giao nó ra PSTN) . Vì lưu lượng truy cập không đặt trên PSTN, ENUM cung cấp một biện pháp bổ xung thêm để tiết kiệm chi phí cho các doanh nghiệp mà liên lạc với các công ty khác sử dụng SIP.

Ví dụ: Tại Tp.HCM số 1234567 sẽ trở thành +84-08-1234567, trong đó 84 đại diện cho mã quốc gia Việt Nam. Dấu cộng “+” biểu thị ký tự đầu tiên của một số E.164 dạng chuẩn đầy đủ. Khi đó tên miền của số điện thoại 1234567 đầy đủ sẽ là “7.6.5.4.3.2.1.3.8.0.84.e164.arpa”.

* 1. Hỗ trợ thoại

Đặc điểm này là chức năng của module Dialog trong OpenSIPS, module cung cấp nhận biết cuộc thoại cho OpenSIPS Proxy. Tính năng của nó là phân loại và lưu trữ giám sát các cuộc thoại hiện hành, để đưa ra thông tin về chúng (ví dụ có bao nhiêu Dialog đang hoạt động). Ngoài ra , còn đưa ra chức năng như thuộc tính cho mỗi cuộc hội thoại, hồ sơ hội thoại (người gọi, điểm đến, kiểu cuộc gọi v..v) và kết thúc cuộc gọi (khi timeout hoặc được kích hoạt từ bên ngoài) .

* 1. Chức năng Load-Balancer

Được thực hiện bởi module Load-Balancer cung cấp lưu lượng định tuyến dựa trên tải. Một thời gian ngắn, khi OpenSIPS định tuyến các cuộc gọi đến một tập các điểm đích, có thể giữ nguyên tình trạng của tải (như số lượng các cuộc gọi liên tục / cuộc gọi đi đâu thì nó chuyển theo đường đó) của mỗi điểm đích và lựa chọn để định tuyến đến các điểm đích ít tải hơn (tại thời điểm đó) . OpenSIP biết rõ khả năng mỗi điểm đích, nó được cấu hình trước với tải trọng tối đa được chấp nhận bởi những điểm đích.

Để chính xác hơn, khi định tuyến OpenSIPS sẽ xem xét các điểm đích ít tải hơn, đó không phải là điểm đích với số lượng các cuộc gọi đang diễn ra nhỏ nhất, mà là điểm đích có số slot trống lớn nhất. Ngoài ra, module Load-Banlancing có thể nhận được những phản hồi từ những điểm đích (nếu chúng có khả năng). Cơ chế này được sử dụng để thông báo cho OpenSIPS khi dung lượng tối đa của điểm đích có sự thay đổi .

Chuyên ngành thuật toán băm trong OpenSIPS có thể cấu hình để cân bằng tải bằng user name, ruri, callid, và các đặc tính khác.

* 1. NAT traversal

NAT là một chức năng của router mạng, nó ánh xạ giữa private IP và public IP. OpenSIPS cung cấp hỗ trợ xử lý từ xa NAT traversal cho tín hiệu SIP và chức năng phát hiện ra các User Agent đằng sau NAT, để sửa đổi các header SIP để cho phép User Agent trong suốt đằng sau NAT và gửi bản tin keepalive tới các User Agent đằng sau NAT để duy trì khả năng nhìn thấy chúng trong mạng. Có khả năng phân phối lưu lượng RTP trên nhiều bộ chuyển tiếp lưu lượng chạy trên nhiều máy chủ.

OpenSIPS cho phép gửi voice ( hoặc luồng video) từ IP Telephone hoặc phần mền VoIP khách hàng trực tiếp đến nhà cung cấp dịch vụ điện thoại VoIP sử dụng trong hầu hết các trường hợp (non-symmetric NAT). OpenSIPS cho phép trực tiếp điều khiển giao tiếp SIP trong các trường hợp đặc biệt, chẳng hạn như khi có hai máy điện thoại VoIP phía sau một NAT router, và muốn trao đổi dữ liệu thông tin trực tiếp với nhau.

* 1. Định tuyến với chi phí thấp nhất (Least cost routing)

Chức năng này được được hỗ trợ bởi module Least Cost Routing (LCR) trong OpenSIPS. Giúp cho khả năng chuyển tiếp một yêu cầu đến đích với một chi phí ngắn nhất thông qua các gateway, các gateway lựa chọn dựa From URI và user part của Request-URI ( số điện thoại ) để tìm ra gateway phù hợp chịu trách nhiệm xử lý yêu cầu đó. Một gateway trùng khớp với một yêu cầu nếu user part của Request-URI và “ From” URI trùng với “prefix” và “From pattern” của gateway.

* 1. Hỗ trợ SRV và NAPTR DNS

Bản ghi NAPTR được sử dụng phổ biến với SIP kết hợp với các bản ghi SVR để tìm ra loại dịch vụ nào đang hiện hữu (như SIP, email hoặc web), cái tên nào sử dụng tra cứu SRV và port nào sử dụng bản ghi SVR.

Bản ghi NAPTR là bản ghi đặc biệt trong hệ thống tên miền, chứa đựng thông tin về các nguồn, những dịch vụ và ứng dụng nào sẽ được kết hợp với một số điện thoại xác định. Những dịch vụ này được xác định và lựa chọn bởi khách hàng.

Ví dụ về khai báo bản ghi NAPTR:

<tên miền> IN NAPTR <thứ tự> <mức ưu tiên> <cờ> <dịch vụ> <biểu thức> <thay thế>

Bản ghi *SRV* cung cấp cơ chế định vị dịch vụ, Active Directory sử dụng Resource Record này để xác định bộ điều khiển miền, Global catalog servers, Lightweight Directory Access Protocol (LDAP) servers.

Các field trong SVR:

Service: dịch vụ mà máy chủ cung cấp, với DC thì service là \_ldap

Protocol: giao thức sử dụng. Thường là TCP và UDP

Domain\_name: tên miền mà bản ghi trỏ tới.

TTL: giá trị Time to live của bản ghi

Class: phân lớp bản ghi, thường là IN

Type: kiểu bản ghi, ở đây là SRV

Priority: độ ưu tiên cho máy chủ được định vị bởi bản ghi này. Trong trường hợp bạn có nhiều máy chủ cùng cung cấp một dịch vụ, máy chủ nào có độ ưu tiên thấp nhất sẽ được lựa chọn trước. Priority có giá trị từ 0 – 65535

Weight: một kỹ thuật cân bằng tải. Khi các bản ghi có Priority giống nhau, chúng sẽ được lựa chọn ngẫu nhiên với xác suất bằng với trường Weight.

Port: cổng trên đó dịch vụ hoạt động.

Target: tên máy chủ cung cấp dịch vụ trong bản ghi SRV.

* 1. Call Processing Language (CPL)

CPL là một ngôn ngữ có thể được dùng để mô tả và điều khiển dịch vụ điện thoại Internet. Nó được xác định sẽ sử dụng với cả hai SIP server va H323 server. CPL đủ mạnh có thể mô tả rất nhiều dịch vụ và tính năng, nhưng nó lại bị giới hạn sao cho nó có thể hoạt động một cách an toàn trong dịch vụ điện thoại qua dịch vụ Internet. Mục đích của ngôn ngữ CPL làm cho người dùng không thể thực hiện bất cứ điều gì phức tạp và nguy hiểm khi mô tả dịch vụ điên thoai internet. CPL được thiết kế đơn giản, dễ dàng tạo và chỉnh sửa bằng công cụ đồ họa. Và độc lập với hoạt động hệ điều hành hoặc giao thức truyền tín hiệu. Điều này phù hợp cho chạy trên một máy chủ mà nơi người dùng có thể không được phép thực hiện các chương trình tùy ý, vì CPL không có biến,vòng hoặc khả năng chạy các chương trình bên ngoài.

CPL có hỗ trợ uploading/downloading/removing các script thông qua các phương thức SIP REGISTER. CPL XML là một ngôn ngữ script mạnh mẽ và linh hoạt cho phép user dùng để phát triển việc định tuyến cuộc gọi. Một số tính năng phổ biến như chuyển hướng, chuyển tiếp, quay số nhanh, call/blacklist… có thể thực hiện thông qua CPL.

* 1. XCAP hỗ trợ cho các Presence Agent

Giao thức XCAP cho phép một client đọc, ghi và sửa đổi dữ liệu cấu hình ứng dụng lưu trữ trong định dạng XML trên một máy chủ. XCAP ánh xạ tài liệu sub-trees XML và các phần tử thuộc HTTP URIs, để các thành phần này có thể truy nhập trực tiếp bởi các client sử dụng giao thức HTTP. Máy chủ XCAP được sử dụng bởi các client XCAP để lưu dữ liệu như danh sách buddy và kết hợp với máy chủ SIP Presence hỗ trợ PUBLISH, SUBSCRIBE và phương pháp NOTIFY để cung cấp giải pháp hoàn chỉnh cho máy chủ SIP SIMPLE. Một XCAP client phải sử dụng sip : user@domain để xác định các thuê bao. Giả sử rằng chỉ có một phần username để yêu cầu sẽ làm cho client không thể vận hành trong môi trường đa miền. Và phải bao gồm một User-Agent header trong bản tin yêu cầu XCAP, nó sẽ giúp xác định nguyên nhân, khắc phục sự cố và tìm ra các yêu cầu trong log(nhật kí) của server dễ dàng hơn.

* 1. Giao diện quản lí và cơ sở dữ liệu

Giao diện quản lí (cho tích hợp ngoài) thông qua FIFO file, XMLRPC hoặc Datagram (UDP hoặc Unixsockets).

OpenSIPS đa cơ sở dữ liệu backends - MySQL, PostgreSQL, Oracle, Berkeley, flat files và các loại database khác có thể xử lý unixodbc.

* 1. Linh hoạt và mạnh mẽ về ngôn ngữ lập trình

OpenSIPS được viết bằng ngôn ngữ lập trình C và cho phép tùy chỉnh các ứng dụng của SIP trong giao diện lập trình PERL và Java . Giao diện lập trình PERL giúp cho người dùng có thể nhúng phần mở rộng viết trong Perl.Với tính năng này có thể dễ dàng thực hiện OpenSIPS mở rộng trong Perl . Điều này cho phép truy cập đơn giản đến các module của CPAN-Comprehensive Perl Archive Network. Với giao diện lập trình Perl, có một số chức năng cho phép đọc và sửa đổi các tin nhắn SIP hiện hành, như cờ Ruri hoặc tin nhắn. SIP URI có thể được viết lại dựa trên các biểu thức thông thường, truy nhập tùy ý vào dữ liệu backends một cách đơn giản. Với Java SIP Servlet Application Interface , thì sử dụng Java SIP servlet để mở rộng các dịch vụ VoIP và tích hợp với dịch vụ web.

* 1. XMPP gateway

XMPP (The Extensible Messaging and Presence Protocol) là một chuẩn mở rộng sử dụng XML cho việc vận chuyển message, trình bày nội dung, dịch vụ yêu cầu và đáp ứng (resquest -response services). Module XMPP của OpenSIPS, nó là một gateway giữa OpenSIPS và máy chủ Jabber. XMPP cho phép trao đổi qua lại tin nhắn tức thời giữa SIP client và XMPP(Jabber) client. Không có yêu cầu ánh xạ địa chỉ SIP tới địa chỉ XMPP thông qua database trên bộ nhớ của nhau, giản đồ địa chỉ cho phép dịch chuyển dễ dàng. Chỉ cần cài đặt SIP-to-XMPP gateway và tất cả user SIP trở thành có sẵn trong mạng XMPP, và user có thể trò chuyện với bất kì ai trong mạng XMPP.

* 1. Gateway to SMS

OpenSIPS cung cấp cách giao tiếp giữa mạng SIP ( thông qua SIP MESSAGE) và mạng GSM ( qua Short Message Service ). Có thể giao tiếp từ SIP tới SMS và ngược lại. Và cung cấp các phương tiện như xác nhận SMS – là gateway có thể xác nhận người sử dụng SIP, nếu tin nhắn thực sự liên lạc được đến đích của nó như tin nhắn SMS, hoặc tin nhắn đa phần, và nếu một bản tin SIP dài quá, nó sẽ được tách ra và gửi dưới dạng đa tin nhắn SMS.

* 1. IP blacklist

OpenSIPS xác định danh sách các địa chỉ IP ( IP tĩnh và IP động ) bị chặn khi chuyển tiếp, danh sách các địa chỉ IP đó được gọi là IP blacklists. Phụ thuộc vào điểm đích của user xác định qua script, OpenSIPS áp dụng IP blacklists khác nhau để chắc chắn rằng yêu cầu không được chuyển tiếp đến một địa chỉ IP được bảo vệ (qua DNS).

Ví dụ :

OpenSIPS định nghĩa một blacklist với địa chỉ IP của gateway PSNT.

Nếu tiếp cận dựa trên định vị người dùng , OpenSIPS kích hoạt danh sách đen để ngăn chăn DNS dựa vào định tuyến IP của các Gateway.

* 1. Xác nhận, ủy quyền,thống kê

OpenSIPS thực hiện xác nhận, ủy quyền,thống kê (authentication, authorization, accounting - AAA) thông qua các cơ sở dữ liệu (MySQL,Postgress,text files), RADIUS và DIAMETER. Nó cung cấp việc xác thực (authentication) người dùng nhằm bảo đảm có thể nhận dạng đúng người dùng. Một khi đã nhận dạng người dùng, OpenSIPS có thể giới hạn thẩm quyền (authorization) mà người dùng có thể làm. Cũng như tập hợp được thông tin như thời gian bắt đầu hay kết thúc của người dùng (accounting). Khi người dùng sử dụng mạng, nó cũng có thể giám sát tất cả những gì mà họ làm.

* 1. Các giao thức vận chuyển

OpenSIPS sử dụng các giao thức UDP/TCP/TLS/SCTP để vận chuyển các bản tin SIP.

UDP (User Datagram Protocol ) : Giao thức Datagram người dùng.

TCP([Transmission Control Protocol](http://translate.googleusercontent.com/translate_c?hl=vi&sl=en&tl=vi&u=http://en.wikipedia.org/wiki/Transmission_Control_Protocol&prev=hp&rurl=translate.google.com.vn&usg=ALkJrhjLDn9IHOwsgaP7TQUP_H1x2NauyQ)): Giao thức điều khiển truyền vận.

TLS (Transport Layer Security ): Bảo mật lớp vận chuyển.

SCTP (Stream Transmission Control Protocol) : Giao thức truyền điều khiển luồng.

* 1. Khả năng nâng cấp OpenSIPS

OpenSIPS có thể chạy trên hệ thống nhúng, hiệu suất có thể lên đến hàng trăm thiết lập cuộc gọi / giây.

Với hệ thống có bộ nhớ 4GB, OpenSIP có thể phục vụ hơn 300000 thuê bao trực tuyến.

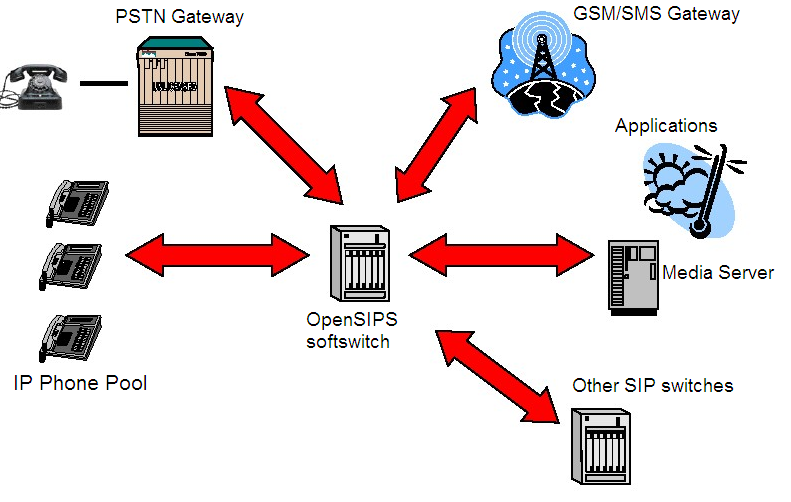
Hệ thống có thể dễ dàng mở rộng bằng cách thêm máy chủ OpenSIPS.

OpenSIPS hỗ trợ IPv4 và IPv6 và có thể phục vụ được multiple domain.

1. Ứng dụng của OpenSIPS
   1. Ứng dụng trong dịch vụ VoIP

OpenSIPS là một Carrier Class SIP Proxy sử dụng chủ yếu bởi các nhà cung cấp VoIP. OpenSIPS là một SIP server đa chức năng, đa mục đích : router, switch, registrar, application server, redirect server, gateway, load-balancer …

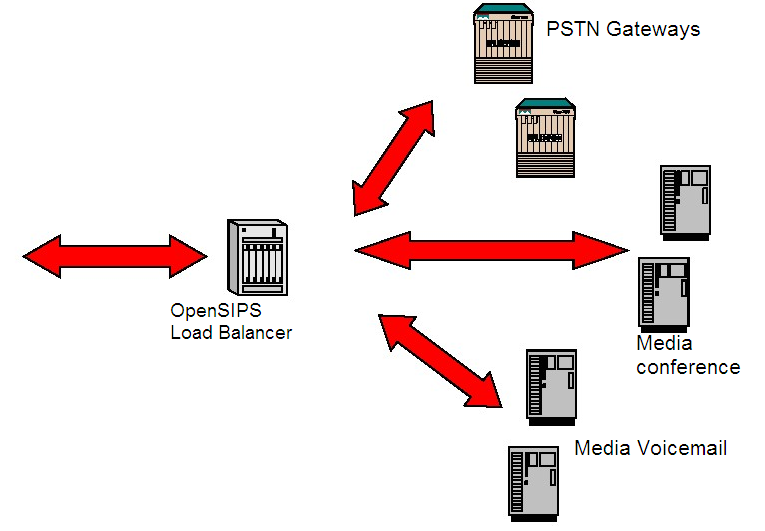
Nó được thiết kế để xử lý lượng lớn các cuộc gọi, cân bằng tải, SIP, giao tiếp, giải quyết vấn đề NAT cao cấp và để xử lý báo hiệu SIP như các báo hiệu khác trong mạng VoIP. OpenSIPS có thể sử dụng phân bố vùng trong nền VoIP.



1. :OpenSIPS trong VoIP

OpenSIPS luôn luôn cần một SIP gateway kết nối với PSTN. Thông thường, sẽ lắp đặt Asterisk như PSTN gateway cho OpenSIPS. OpenSIPS là một SIP proxy chỉ xử lý tín hiệu, và không có khả năng của bất kì phương tiện liên quan đến những dịch vụ (phần nhỏ voice của cuộc gọi điện thoại). Nó không phải là nền tảng mà sử sụng để tạo ra hệ thống như IVR, thư thoại, TTS và nhận dạng giọng nói... Tuy nhiên, một hệ thống OpenSIPS có thể tích hợp bất kì dịch vụ nào bằng cách sử dụng một máy chủ riêng biệt như Asterisk.

* 1. Ứng dụng trong các doanh nghiệp



1. : OpenSIPS Load Balancer

Sức mạnh của OpenSIPS là khả năng cân bằng tải cho cuộc gọi VoIP. Chuyên ngành thuật toán băm trong OpenSIPS có thể cấu hình để cân bằng tải bằng “user name”,”ruri”,”callid”, và các đặc tính khác. OpenSIPS có thể là phần bổ xung thực sự cho giải pháp Asterisk đối với các doanh nghiệp.

OpenSIPS thường được sử dụng khi các doanh nghiệp có nhu cầu đặc biệt, chẳng hạn như định tuyến và cân bằng tải hoặc khi các doanh nghiệp đang quản lý khối lượng lớn như hơn một nghìn người đăng kí sử dụng.

* 1. SIP trunking

SIP trunk như một kết nối giữa IP PBX và một Service Provider's application servers cho phép trao đổi lưu lượng thoại qua IP giữa hai bên. Khi cuộc gọi từ một điện thoại nội bộ đến một số bên ngoài, tổng đài PBX sẽ gửi các thông tin cần thiết đến nhà cung cấp SIP Trunk, nhà cung cấp thiết lập cuộc gọi đến dialed numbers và hoạt động như một trung gian cho các cuộc gọi. Tất cả các báo hiệu và lưu lượng thoại giữa các PBX và các nhà cung cấp được trao đổi bằng cách sử dụng giao thức SIP và RTP qua mạng IP.

SIP Trunking cho phép khách hàng doanh nghiệp triển khai IP-PBX xây dựng các giải pháp hội tụ thoại và dữ liệu vào các kết nối IP một cách dễ dàng và tiết kiệm được. SIP Trunking sử dụng OpenSIPS để thực hiện các chức năng định tuyến, báo hiệu, thiết

# CHƯƠNG III: PHÂN TÍCH XỬ LÍ CUỘC GỌI TRONG OPENSIPS

1. Core and modules

Opensips được xây dựng chủ yếu từ core, chịu trách nhiệm việc xử lí bản tin SIP và các tính năng cơ bản. Các module chịu trách nhiệm chính về các chức năng OpenSIPs. Các module thực hiện các chức năng của chúng trong các script OpenSIPs với các lệnh và tham số mới chứa trong script đó.

OpenSIPs được cấu hình trong một tệp tin opensips.cfg. Tệp tin cấu hình này điều khiển các thứ mà các module tải vào và các tham số tương ứng. Tất cả các lưu lượng SIP cũng được điều khiển trong nhiều khối định tuyến được định nghĩa trong tệp tin này. Tệp tin opensips.cfg là tệp tin cấu hình chính.

1. Các thành phần trong tệp tin opensips.cfg

Các định nghĩa global: Phần này chứa nhiều tham số làm việc cho Opensips, bao gồm IP và port cho dịch vụ SIP và cấp bậc debug. Những tham số global này tác động đến core và tất cả các module một cách tổng thể.

Các module: chứa một loạt các danh sách thư viện cần thiết trong việc thêm các tính năng không có trong core. Các module được đưa vào với lệnh loadmudule.

Cấu hình module: các module có nhiều tham số và cần được thiết lập chính xác. Các tham số này được cấu hình bằng lệnh modparam(modulename, parametername,parametervalue).

Khối định tuyến chính: Là nơi quá trình xử lí các bản tin SIP yêu cầu bắt đầu . Nó điều khiển viêc xử lí các bản tin yêu cầu nhận được.

Khối định tuyến phụ: người quản trị có thể định nghĩa các khối định tuyến mới sử dụng lện route(). Các khồi định tuyến này làm việc như các chương trình con trong scritpt của Opensips.

Các khối định tuyến phản hồi: các khối định tuyến phản hồi được sử dụng để xử lý các bản tin phản hồi ( các bản tin phản hồi thành công hay là các bản tin phản hồi từ chối) thường là bản tin 200 Ok.

Các khối định tuyến Failure: các khối định tuyến failure được sử dụng để xử lí các điều kiện failure như là busy hoặc timeout

Các khối định tuyến nhánh: chứa các biểu thức logic để thi hành cho mỗi nhánh của bản tin yêu cầu, trước khi nó chuyển bản tin đi.

Các khối định tuyến cục bộ: các khối định tuyến cục bộ được thực hiện khi Opensips phát ra một yêu cầu ( với vai trò UAS) sử dụng Transaction module (TM).

Khối định tuyến Error : khối này được thi hành khi một lỗi của bản tin yêu cầu phát hiện.

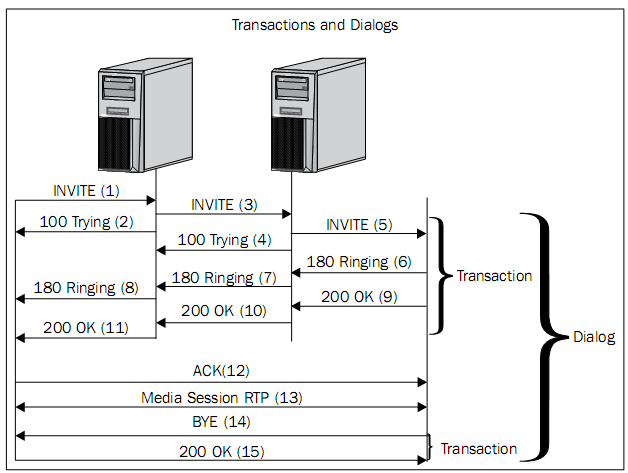
Sessions, dialogs và transactions.

Các định nghĩa được sử dụng trong OpenSIPS:

SIP transaction: là một yêu cầu được gửi đi và nhận lại các bản tin trả lời ( như là: register và 200 OK).

SIP dialog: một kết nối tồn tại trong một khoảng thời igan giữa hai thực thể SIP ( như là một dialog được thiết lập giữa hai UAC từ bản tin INVITE tới bản tin BYE)

SIP session: một luồng dữ liệu (audio/video/text) giữa hai thực thể SIP.



1. :Transaction và Dialog
2. Quá trình xử lí bản tin trong tệp tin opensips.cfg

Tệp tin opensips.cfg la một script được thi hành khi nhận được một bản tin SIP. Ví dụ, nếu người dùng A muốn nói chuyện vơi người dung B, thì nó gửi một bản tin INVITE. Bản tin này được xử lí trong khối định tuyến chính. Quá trình xử lí này chứa một t\_relay() (chuyển bản tin) hoặc một t\_reply/s1\_send\_reply( gửi một hồi âm phủ định), hoặc quá trình này có thể hủy bản tin đó tại cuổi khối định tuyến đó sử dụng lệnh exit().

* 1. SIP proxy

Mỗi proxy sẽ có các quyết định định tuyến khác nhau, chỉnh các bản tin yêu cầu trước khi gửi chúng đi tới thành phần tiếp theo. Các bản tin response se được định tuyến qua cùng các proxy mà các bản tin yêu cầu đã đi qua theo chiều ngược lại.

Một SIP proxy có thể hoạt động ở chế độ statless hoặc là stateful. Khi một SIP proxy làm việc như là một công cụ gửi các gói tin SIP đơn giản, thì nó sẽ chuyển các bản tin đó tới đích được xác định bởi bản tin yêu cầu. Chế độ stateless sẽ hủy bỏ bất kì thông tin nội nào về bản tin đó, sau khi bản tin đã được chuyển đi. Đặc điểm này giới hạn về việc tính cước và cư xử với trường hợp cuộc gọi thất bại.

Khi OpenSIP biết rằng bản tin 200 OK tương ứng với bản tin INVITE trước đó, thi nó đang làm việc ở chế độ stateful. Việc này quản lí các bản tin response trong một khối on\_reply\_route(). Với quá trình xử lí stateless, mỗi bản tin được xử lí mà không đi kèm với một ngữ cảnh nào.

Khi yêu cầu các công việc phức tạp hơn như là tính cước, chuyển cuộc gọi, và voicemail, thì sử dụng chế độ stateful. Mỗi transaction sẽ được lưu lại trong bộ nhớ và các trường hợp thất bại, các bản tin response và việc truyền lại các bản tin sẽ được gắn liền với transaction này. Stateful transaction được xử lí bởi modun TM và thường sử dụng lệnh t\_relay().

* 1. Hoạt động của Stateful

Khi hoạt động ở chế độ stateful, một proxy đơn giản là một bộ xử lí SIP transaction.

Các bước xử lí :

1. Kiểm tra sự hợp lệ của bản tin yêu cầu
2. Tiền xử lí thông tin đinh tuyến.
3. Xác định đích của bản tin yêu cầu.
4. Chuyển bản tin yêu cầu đến đích.
5. Xử lí tất cả các bản tin response.

Một stateful proxy tạo ra một server transaction mới khi nhận được mỗi bản tin yêu cầu mới. Bất kì sự truyền lại một bản tin yêu cầu nào sẽ được xử lí bởi server trancsaction đó.

Ví dụ: Mỗi bản tin yêu cầu đi ngang qua một SIP proxy, thi nó sẽ làm các bước sau:

Bước 1: Kiểm tra sự hợp lệ của bản tin yêu cầu.

* Kiểm tra kích thước bản tin để tránh tràn bộ đệm
* Kiểm tra Max-forwards của header để tránh các vòng lặp

Bước 2: Tiền xử lí các thông tin định tuyến.

* Nều một thông tin định tuyến trong header tồn tại thì xử lí nó.

Bước 3: Xác định đích của bản tin yêu cầu đó.

* Có phải nằm trong cơ sở dữ liệu vị trí hay không ( các thuê bao đã đăng kí)?
* Có một đường đi đến đích đó hay không ( các đích đến cổng)
* Có phải nó được gửi đến một domain ngoài hay không?

Bước 4: chuyển các bản tin yêu cầu.

* Gọi hàm t\_relay và OpenSIPs sẽ làm tất cả công việc.

Bước 5: Xử lí các bản tin response

* 1. Scripting OpenSIPs :

Có một tệp tin cấu hình cơ bản nhất cho OpenSIPs. Tệp tin này có tên là opensips.cfg. Nó chứa nhiều phần; như là “Global Parameters”, “Load Modules”, “Module Parameters”, và “Routing Script”. Để tạo tệp tin này phải có nhiều lệnh và các hàm lấy từ lõi và các module.

Global parameters

Nó điều khiển hành vi của **daemon.**

Và sau đây là một vài tham số:

* 1. Listen interfaces:

Tham số này chỉ ra cổng IP nào sẽ sử dụng cho OpenSIPs để nhận các lưu lượng, và tham số này chỉ thiết lập cho các giao diện nào đang được sử dụng. Nếu không thay đổi thông số đó, nó sẽ tự động tìm tất cả các giao diện có thể và nó sẽ tiêu thụ nhiều tài nguyên hơn cần thiết. Tham số cổng định nghĩa cổng nào của máy chủ SIP sẽ lắng nghe. Mặc định là 5060.

listen=udp:192.168.152.148:5060

listen=tcp:192.168.152.148:5061

listen=tls:192.168.152.148:5062

port=5060

* 1. Logging

Có một vài tham số điều khiển hệ thống log.Và quan trọng nhất là tham số debug. Sử dụng debug=3 thì nó chỉ in ra các bản tin lỗi quan trọng trong khi đó debug=9 sẽ in tất cả các bản tin. Để thay đổi các mức của debug ta sử dụng một lệnh như là opensipsctl fifo debug 1. Mặc định là 2. Các mức cao hơn thì các thông tin được ghi trong log nhiều hơn. Sau đây là mô tả các mức của log:

* L\_ALERT (-3) mức này được sử dụng nếu lỗi cần hành động tức thì.
* L\_CRIT (-2) mức này được sử dụng nếu lỗi là nghiêm trọng.
* L\_ERR (-1) mức này nên được sử dụng để báo cáo các lỗi trong suốt quá trình xử lí dữ liệu mà không gây cho hệ thống trục trặc.
* L\_WARN (1) mức này được sử dụng để viết các bản tin cảnh báo.
* L\_NOTICE (2) mức này được sử dụng để báo cáo các tình huống không bình thường.
* L\_INFO (3) mức này được sử dụng để viết các bản tin thông tin.
* L\_DBG (4) mức này được sử dụng để viết các bản tin cho việc sửa lỗi.
  1. Số lượng process

Fork trực tiếp bảo quá trình OpenSIPs thực thi foreground hay là background. Để hoạt động background thì fork= yes. Thỉnh thoảng ta có thể thấy hữu ích bắt đầu trong foreground để định các lỗi của script. Nếu fork =no, OpenSIPs sẽ không thể lắng nghe nhiều hơn một gia diện và hỗ trợ TCP/TLS sẽ tự động vô hiệu hóa. Một chế độ xử lí đơn nhất, chỉ một giao diện UDP được chấp nhận. Children trực tiếp xác nhận bao nhiêu quá trình xử lí “con” mỗi giao diện được tạo ra để xử lí các bản tin yêu cầu hướng vào. Bốn quá trình là khá tốt cho hầu hết các hệ thống. Thông số này chỉ áp dụng trên giao diện UDP. Nó không tác động đến các quá trình của TCP.

fork = yes

children = 4

tcp\_children=6

disable\_tcp=no

disable\_tls=no

Các tùy chọn Daemon

Các tùy chọn deamon có thể được sử dụng để thiết lập người dùng và nhóm và chúng rất hữu ích cho việc tránh chạy OpenSIPs với quyển root của hệ thống.

gid/group=sip # unix group

uid/user=sip # unix user

wdir="/" # working directory

chroot="/usr/local/opensips-1.6"

Đặc tính bản tin SIP

Đặc tính bản tin SIP hữu ích cho việc thiết lập các tham số chuẩn được sử dụng bên trong các bản tin yêu cầu và response. Có thể sử dụng nó để dấu đặc tính phần mềm đang sử dụng.

server\_header="Server: My openSIPS

#default is "openSIPS (<version> (<arch>/<os>))"

server\_signature = yes

user\_agent\_header="User-Agent: My openSIpS

* 1. Các thông số khác

Tham số alias rất quan trọng; nó định nghĩa domain đang được phục vụ. Có thể kiểm tra bản tin yêu cầu đang đến hoặc đi tới một domain bằng cách sử dụng giá trị của core myself. Sử dụng auto\_aliases=yes sẽ làm cho hệ thống tìm ra các aliase đang sử dụng phân giả tên miền DNS. Chuyển DNS được sử dụng để chuyển các đích đến đang sử dụng hệ thống tên miền đó sang domain khác khi bị lỗi.

alias="mydomain.sip" # thiết lập tên cho máy chủ

auto\_aliases=no # tìm aliase thông qua phân giải tiên miền DNS

disable\_dns\_failover = yes

sip\_warning=yes #thêm một tiêu đề gỡ lỗi trong các gói trả về.

* 1. Modules và các thông số của chúng.

Các module có thể đưa vào sử dụng bằng lệnh loadmodule. Tham số mpath thiết lập đường dẫn cho module.

mpath="/usr/lib/opensips/modules/"

loadmodule "tm.so"

Để cấu hình các tham số module thì sử dụng lệnh modparam. Lệnh này có ba tham số- tên module, tham số module và giá trị tham số:

modparam("tm", "fr\_inv\_timer", 20)

Nó cũng có khả năng thiết lập cùng tham số cho nhiều module:

modparam("usrloc|auth\_db","db\_url", "mysql:opensips@localhost/opensips")

Cấu hình chuẩn cho các module và các tham số.

Bên dưới là các module chính được sử dụng trong script mặc định:

loadmodule "sl.so"

loadmodule "tm.so"

loadmodule "rr.so"

loadmodule "maxfwd.so"

loadmodule "usrloc.so"

loadmodule "registrar.so"

loadmodule "textops.so"

loadmodule "mi\_fifo.so"

loadmodule "uri.so"

loadmodule "xlog.so"

loadmodule "acc.so"

loadmodule "signaling.so"

Chức năng của các module:

**Module SL**

Module SL cho phép OpenSIPS đóng vai trò như một stateless UA server và gửi nhưng bản tin đấp ứng cho các bản tin yêu cầu mà không giữ trạng thái của chúng.

Và hàm thường sử dụng trong module này là sl\_send\_reply.

sl\_send\_reply(code, reason)

* code – mã trả về.
* reason – đoạn thông tin .

Với một yêu cầu nhận được thì một bản tin đáp ứng được gửi ngược trở lại với một mã số và thông báo dưới dạng chữ.

Ví dụ:

sl\_send\_reply(“404”’, “Not found”);

**Module TM**

Module này xử lí các bản tin SIP ở chế độ stateful. Khi được sử dụng thì nó sẽ lưu các các bản tin SIP nhận được vào bộ nhớ.

Và hàm chính trong module này là hàm t\_relay().

Chức năng của hàm là chuyển các bản tin SIP đến đích dựa vào địa chỉ URI.

**Module RR**

Hai hàm chính trong module này là loose\_route() và record\_route().

Record\_route(): thêm tiêu đề record\_route vào bản tin SIP.

Loose\_route(): thực hiện việc định tuyến dựa vào tiêu đề route trong bản tin SIP

**Module MAXFWD**

Module này có chức năng thực hiện kiểm tra các toán tử trong tiêu đề Max-forward như là thêm, giảm và kiểm tra giá trị đang tồn tại.

Hàm hay sử dụng trong OpenSIPS :

### mf\_process\_maxfwd\_header(max\_value)

Khi bản kiểm tra bản tin nếu không có tiêu đề max-forward thì bản tin sẽ được thêm vào với giá trị là max\_value. Nếu đã có rồi thì nó sẽ giảm một đơn vị nếu nó khác 0.

**Module USRLOC**

Module này quản lí bảng location, cung cấp sự truy cập vào bảng cho các module khác. Không có hàm trích xuất nào được sử dụng và nó có thể được sử dụng trực tiếp trong các đoạn mã.

**Module REGISTRA**

Module này có nhiệm vụ xử lí các bản tin REGISTER.

Để sử dụng được module này thì các module usrloc và module signaling phải được tải trước.

Một vài hàm trích xuất của module :

Save()

Hàm này xử lí một bản tin REGISTER. Nó có thể thêm, xóa hoặc chỉnh sửa các thông tin lưu trong usrloc phụ thuộc vào tiêu đề Contact và Expire. Khi thành công nó sẽ gửi bản tin 200 OK liêt kê tất cả các contact hiện tại trong usrloc.

Lookup()

Hàm này sẽ lấy username từ URI của bản tin yêu cầu và tìm tất cả các contact cho username này trong usrloc. Nếu có contact thì URI sẽ được thay thế bởi contact có giá trị q cao nhất.

**Module TEXTOPS**

Module này thực hiện các hoạt động dựa trên chữ qua các bản tin SIP được xử lí bởi OpenSIPS. SIP là một giao thức dựa trên chữ và module này cung cấp một bộ các hàm hữu ích để phân tích bản tin ở cấp độ chữ như là chèn vào một header mới.

**Module MI\_FIFO**

Module này cung cấp một lớp vận chuyển FIFO được thi hành ở giao diện quản lí.

Không có hàm trích xuất cho module này.

Module URI

### has\_totag()Kiểm tra trong tiêu đề to có trường tag hay không.

### is\_user(username)Kiểm tra username trong lúc đăng kí có giống với username đã được đưa ra hay không.

### Module XLOG

### Module này cung cấp khả năng in ra log để có thể sữa được lỗi từ OpenSIPS.

### Và đề xuất ra log thì sử dụng hàm xlog().

### Module ACC

### Tính toán các thông tin của các transaction

**Module SIGNALLING**.

Module này như là một sự kết hợp giũa hai module sl và tm cung cấp một hàm để được gọi bởi các module muốn gửi một bản tin đáp ứng

### Hàm được sử dụng send\_reply()

### Module DB\_MYSQL

### Module này cung cấp sự kết nối giữa OpenSIPS và mySQL.

* 1. Các script cơ bản.

OpenSIPs sử dụng một ngôn ngữ riêng biệt giống như C, nhưng tập trung vào nhiệm vụ định tuyến các bản tin SIP yêu cầu và xử lí các bản tin SIP phản hồi.

Một vài hàm và giá trị được cung cấp bởi lõi, trong khi có một vài được cung cấp bởi các module. Và điều quan trọng hiểu được cái gì là có thể từ lõi và từ các module. Thường thì ta thử khởi động OpenSIPs và nhận một bản tin lỗi là hàm xác nhận là không tồn tại. Hầu hết các trường hợp là chỉ do các module đó chưa được tải vào.

* 1. Các hàm của lõi.

Các hàm của lõi là các hàm có khả thể dùng mà không cần bất cứ module nào. Có một vài là quan trọng cho hành vi của script đó. Các hàm lõi không có giới hạn về số tham số mà chúng có thể chấp nhận. Các hàm module có thể có tối đa sáu tham số. Sử dụng các hàm lõi có thể quyết định cái gì làm với script đó.

* Forward(): Định tuyến bản tin yêu cầu “stateless” ( dựa vào R-URI)
* Drop() : Dừng lại việc thi hành script cấu hình và thay đổi hành động ẩn sẽ được thi hành về sau.
* Exit (): kết thúc quá trình xử lí script và gửi đi.

Các hàm quan trọng khác là seturi(),setflag(),isflagset(),strip(),prefix(), và rewritehostprort().

* 1. Các giá trị của lõi.

Vài giá trị trong script là được định nghĩa trước và có thể rất hữu ích. Một vài ví dụ là:

* INET/INET 6-thiết lập nếu giao thức là IPv4 hoạc IPv6
* TCP/TLS/UDP- thiết lập dựa trên giao thức được sử dụng.
* Myself là một tham chiếu tới danh sách địa chỉ IP nội bộ , hostnames, và các aliase.

Các từ khóa lõi.

Các từ khóa lõi được sử dụng để nhận dạng các giá trị trong bản tin SIP. Ví dụ:

* af- address family (INET/INET 6)
* proto-Protocol (TCP/TLS/UDP)
* dst\_ip- Địa chỉ IP của giao diện cục bộ mà bản tin SIP được nhận.
* method- biến này là một tham chiếu đến method của bản tin SIP.
* Status- nếu được sử dụng tỏng onreply\_route, biến này là một tham chiếu đến mã trạng thái của bản tin phản hồi.
* Retcode- Nó đại diện cho giá trị nhận được bởi việc thực thi hàm cuối cùng.
* Uri- biến này có thể được sử dụng để kiểm tra giá trị URI của bản tin yêu cầu.
* From\_uri- biến script này có thể được sử dụng là một tham chiếu đến URI của tiêu đề FROM .
* To\_uri- biến này có thể được sử dụng để kiểm tra giá trị của URI từ tiêu đề TO.

Một vài ví dụ sử dụng các từ khóa lõi và giá trị lõi :

if(af==INET6) {

log("Message received over Ipv6 link\n");

};

if(is\_method("INVITE") && from\_uri=~".\*@opensips.org")

{

log("the caller is from opensips.org\n");

};

* 1. Các biến giả.

Các biến giả là các biến hệ thống mà có thể sử dụng trong script để truy cập nhiều loại thông tin khác nhau từ các bản tin SIP ( như là tiêu đề, R-URI, và địa chỉ IP nguồn ) hoặc OpenSIPs ( như là thời gian và ID của quá trình). Những biến này có thể được sử dụng trực tiếp từ script đó hoặc có thể xem như là các tham số trong các hàm script. Các biến này được đánh giá dựa trên ngữ cảnh và bản tin SIP được xử lí. Một vài module có thể nhận biến này như là:

* ACC
* AVPOPS
* TEXTOPS
* UAC
* XLOG

Một biến giả luôn bắt đầu với $. Nếu sử dụng $ trong script, thì nó phải thoát nó với $$. Có một bộ các biến được định nghĩa trong OpenSIPS.

<http://www.opensips.org/Resources/DocsCoreVar16>

* 1. Các biến script

Các biến script ám chỉ các biến có thể được sử dụng trong script cấu hình. Các loại biến này nhanh hơn AVPs và chúng chỉ được đính kèm trong script đó. Các biến script tồn tại suốt quá trình thi hành script và một khi script kết thúc, chúng sẽ biến mất. một biết script có thể có giá trị là số hoặc là giá trị chuỗi.

$var(name)

Một vài ví dụ:

* $var(b)=1;
* $var(b)="1";
* $var(b)="$fu"+"$tu";
* $var(b)=1+2;

Các hàm toán tử :

* + :Plus
* -: Minus
* /: Divide
* \*: Multiply
* %: Modulo division
* |: Bitwise OR
* &: Bitwise AND
* ^: Bitwise XOR
* ~: Bitwise NOT

Các phép biến đổi chuỗi

* {s.int}

{s.len}

* {s.substr,offset,length}
* {s.select,index,separator}
* {uri.user}
* {uri.host}
* {tham số.giá trị,tên} – trả về giá trị “tên” của “tham số”

Ví dụ:

"a=1;b=2;c=3"{param.value,c} = "3"

* 1. Tổng quan Attribute-Value Pair (AVP)

Một AVP là một biến được đính kèm vào bản tin SIP của SIP transaction đó ( nếu nằm trong chế độ stateful) – vì vậy AVPs là các biến tồn tại trong transaction. AVP được cấp khi transaction đó bắt đầu và không được dùng khi nó hoàn thành. Tên của AVP có thể là một số hoặc một chuỗi và giá trị của AVP có thể cũng là một số hoặc một chuỗi.

* 1. Flag

Trong OpenSIPS thường thấy sử dụng các script flag. Các flag này được sử dụng để kích họat một vài tiến trình như tính cước, điều khiển cuộc gọi, xử lí NAT, và một vài thứ khác.

Có ba loại flag là : flag bản tin, script flag, và các flag nhánh.

|  |  |  |  |
| --- | --- | --- | --- |
| Loại | phần | Hàm | Mục đích |
| Flag bản tin trong | Transaction mức transaction | setflag(flag\_idx) | Làm hoạt động một vài hàm ở cấp độ transaction |
| Flag nhánh trong | Nhánh | setbflag(flag\_idx) | Làm hoạt động vài hàm ở mức độ nhánh. |
| Script flag khác | Định tuyến ở Mức đầu | setsflag(flag\_idx) | Được sử dụng để lưu các flag khác. chỉ hợp lệ cho script |

1. :các loại Flag

Có thể kiểm tra các giá trị thập phân của các flag thiết lập sử dụng biến giả - $mf (message flag),$bf(branch flags), và $sf(script flag).

Các Hàm:

**If-else**

if ( t\_check\_trans() ) {

t\_relay();

exit;

} else {

exit;

}

**Switch**

switch($retcode)

{

case -1:

log("process INVITE requests here\n");

break;

case 1:

log("process REGISTER requests here\n");

break;

case 2:

case 3:

log("process SUBSCRIBE and NoTIFY requests here\n");

break;

default:

log("process other requests here\n");

}

**Subroutes**

route[1]{

if(is\_method("INVITE"))

{

return(-1);

};

if(is\_method("REGISTER"))

return(1);

}

if(is\_method("SUBSCRIBE"))

return(2);

}

if(is\_method("NoTIFY"))

return(3);

}

return(-2);

}

1. Cơ bản định tuyến
   1. Định tuyến bản tin yêu cầu và phản hồi.

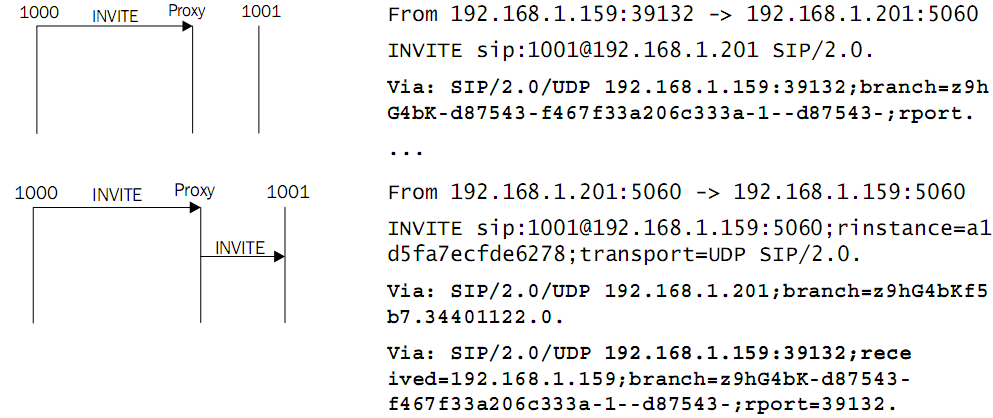
Các bản tin yêu cầu được định tuyến sử dụng một vài kĩ thuật trong các script của OpenSIPS, thường cho các cuộc gọi qua nhiều domain, sử dụng máy chủ DNS để tìm ra các địa chỉ của đích đến, trong khi các cuộc gọi trong cùng domain thường được định tuyến sử dụng bản vị trí người sử dụng. Các bản tin phản hồi được định tuyến ngược trở lại dựa vào tiêu đề VIA được chèn trong suốt tuyến của bản tin yêu cầu.

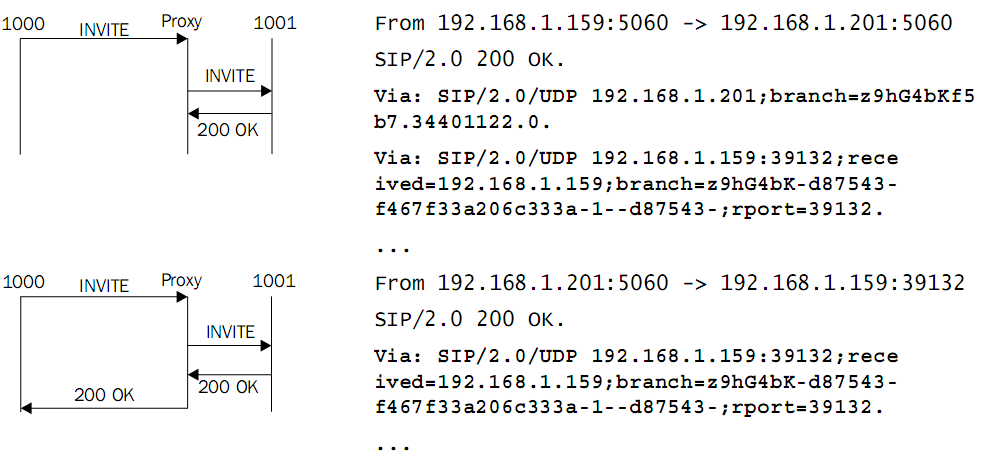
Với định tuyến stateful, transaction được so sánh dựa trên tham số nhánh trong trường VIA.

Ví dụ: địa chỉ của SIP proxy là 192.168.1.201:5060, địa chỉ IP của người dùng 1000 là 192.168.1.159:39132 và địa chỉ IP của người dùng 1001 là 192.168,1.149:5060. Tất cả các tiêu đề khác được bỏ đi để xét đơn giản hơn.

Ảnh

Tiêu đề



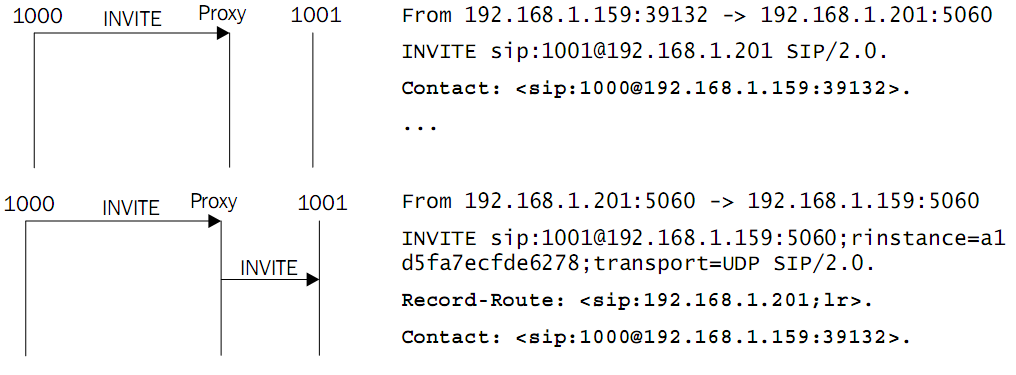


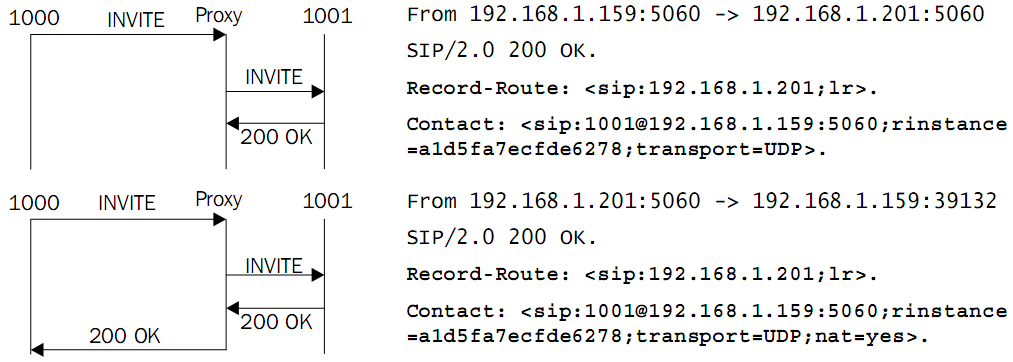
1. :Thông tin bản tin INVITE

Quan sát nội dung của tiêu đề VIA, có đầy đủ thông tin để định tuyến bản tin phản hồi ngược trở lại. Tham số quan trọng khác là branch – được sử dụng cho transaction trùng với chế độ stateful. Trường received và rport được sử dụng bởi RFC3581 cho SIP NAT.

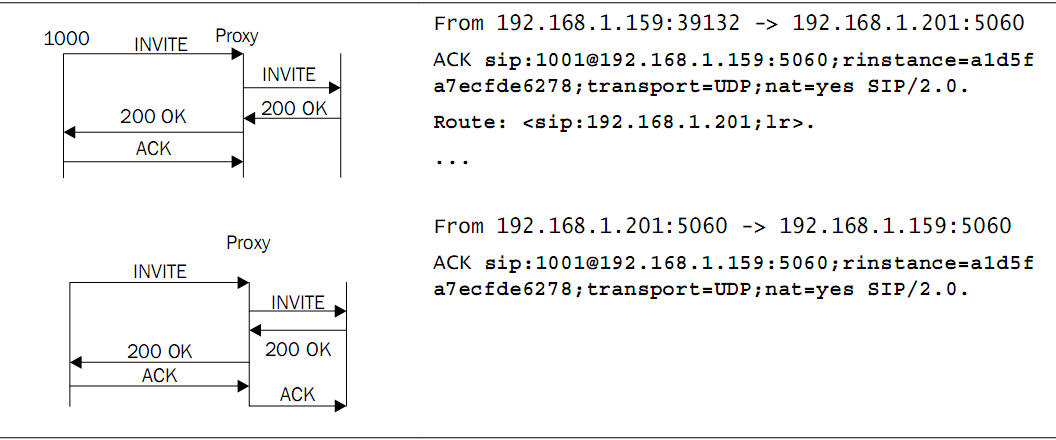
* 1. Các bản tin yêu cầu đầu tiên và sau đó.

Các bản tin ban đầu được định tuyến dựa trên kĩ thuật tìm đường, thường trong bảng vị trí hoặc DNS, để xác định đích. Bản tin yêu cầu ban đầu lưu lại các SIP proxy tương ứng trên đường đi, từ nguồn tới đích sử dụng kĩ thuật gọi là record routing. Ví dụ:



1. :Thông tin record-route

Các bản tin yêu cầu liên tiếp sau đó được định tuyến dựa trên thông tin thu thập được từ bản tin yêu cầu đầu tiên. Các thông tin định tuyến được thu thập lại như là một bộ định tuyến. Trong script đó, sử dụng hàm loose\_route() để định tuyến dựa trên bộ định tuyến. Các bản tin yêu cầu liên tiếp sau đó thường là ACK,BYE,và re-INVITE. Có thể phân biệt giữa các bản tin yêu cầu và đầu tiên và sau đó dựa vào tham số TAG trong tiêu đề TO. Các bản tin yêu cầu sau đó được định tuyến dựa trên tiêu đề Route và URI. Mặt khác, UAC nhận bộ định tuyến dựa vào tiêu đề Record-Route và Contact. Một khi UAC nhận được bộ định tuyến, nó sẽ làm một bản sao tiêu đề Contact trong URI của bản tin yêu cầu đó. Đối với proxy, sử dụng thông tin định tuyến chứa trong bản tin yêu cầu sử dụng hàm loose\_route hơn là tìm kiếm lại đích đến sử dụng bảng vị trí người dùng hoặc là sử dụng DNS.



1. :Nội dung tiêu đề của bản tin ACK
   1. Các đoạn Script định tuyến

Các script định tuyến trong tệp tin opensips.cfg có thể sử dụng được sau khi cài đặt. Các lệnh sẽ được giải thích trong dòng tiếp theo. Dưới đây là đoạn đầu của định tuyến của một bản tin SIP yêu cầu. Đoạn này bắt đầu với một dấu { . Các bản tin SIP yêu cầu sẽ được sử lí trong đoạn này.

if (!mf\_process\_maxfwd\_header("10")) {

sl\_send\_reply("483","Too Many Hops");

exit;

}

Khi một bản tin yêu cầu vào đoạn định tuyến chính này, thì có một vài sự kiểm tra được thực hiện. mf\_process\_maxfwd\_header kiểm tra về bảo mật và luôn nằm trong dòng đầu tiên của khối định tuyến chính. Hàm này nằm trong module Maxforward (maxfwd.so) và được sử dụng để đăng kí số lần một bản tin yêu cầu qua máy chủ SIP này. Nó được sử dụng để tránh vòng lặp.

Nếu trong tình huống này vòng lặp xảy ra, OpenSIPS sẽ báo cho bên SIP client có lỗi xảy ra. Hàm sl\_send\_reply chịu trách nhiệm trong việc này. Hàm này được lấy ra từ module stateless (sl.so) và công việc nó làm là gửi một bản tin yêu cầu stateless tới SIP client. Điều này có nghĩa OpenSIPS sẽ không chờ bản tin phản hồi của bản tin đó. Và chỉ dẫn exit sẽ nói OpenSIPS dừng việc xử lí cho các bản tin yêu cầu và thoát.

if (has\_totag()) {

if (loose\_route()) {

if (is\_method("BYE")) {

setflag(1); # thực hiện tính toán ...

setflag(3);

} else if (is\_method("INVITE")) {

record\_route();

}

# chuyển nó tới bất kì đích nào được thiết lập trong loose\_route()

# in $du (destination URI).

route(1);

} else {

/\* uncomment the following lines if you want to enable presence \*/

##if (is\_method("SUBSCRIBE") && $rd == "your.server.ip.address") {

## # in-dialog subscribe requests

## route(2);

## exit;

##}

if ( is\_method("ACK") ) {

if ( t\_check\_trans() ) {

# non loose-route, but stateful ACK; must be an ACK after

# a 487 or e.g. 404 from upstream server

t\_relay();

exit;

} else {

# ACK without matching transaction ->

# ignore and discard

exit;

}

}

sl\_send\_reply("404","Not here");

}

exit;

}

Phần trên sẽ xử lí các bản tin yêu cầu tiếp theo. Nếu một gói tin có phần tag trong tiêu đề “To:” , điều này nói lên rằng bản tin yêu cầu này không phải là bản tin yêu cầu đầu tiên mà là các bản tin yêu cầu tiếp theo. Các bản tin yêu cầu như bản tin BYE và CANCEL của các transaction đang tồn tài sẽ được chuyển đi. Các gói với tag trong tiêu đề To nhưng không có ;lr sẽ bị hủy với một bản tin lỗi.

#initial requests

# CANCEL processing

if (is\_method("CANCEL"))

{

if (t\_check\_trans()) t\_relay();

exit;

}

Đây là phần xử lý của bản tin yêu cầu CANCEL. Nếu nó thuộc một transaction với bản tin INVITE đang tồn tại thì nó sẽ được chuyển đi tới đích mà gói INVITE đã được định tuyến tới.

Hàm t\_check\_trans() được sử dụng để xác định một bản tin yêu cầu có thuộc một transaction hay không. Trong đoạn trên hàm này được sử dụng để dừng đoạn script lại nếu bản tin yêu cầu đó là bản tin truyền lại.

##if (!(method=="REGISTER") && from\_uri==myself) /\*no multidomain\*/

##if (!(method=="REGISTER") && is\_from\_local()) /\*multidomain\*/

##{

## if (!proxy\_authorize("", "subscriber")) {

## proxy\_challenge("", "0");

## exit;

## }

## if (!db\_check\_from()) {

## sl\_send\_reply("403","Forbidden auth ID");

## exit;

## }

##

## consume\_credentials();

## # caller authenticated

##}

Đây là đoạn chứng thực cho các bản tin yêu cầu không đăng kí.

if (loose\_route())

{

xlog("L\_ERR",

if (!is\_method("ACK"))

sl\_send\_reply("403","Preload Route denied");

exit;

}

Phần trên kiểm tra các bản tin yêu cầu không có tag trong tiêu đề To nhưng lại có tiêu đề Route. Nếu tìm thấy điều này thì chúng sẽ bị hủy ngoại các bản tin ACK.

# record routing

if (!is\_method("REGISTER|MESSAGE"))

record\_route();

Đoạn chương trình sẽ thêm trường record\_route vào bản tin yêu cầu nếu nó không phải là bản tin REGISTER hoặc MESSAGE.

if (is\_method("REGISTER"))

{

##if (!www\_authorize("", "subscriber"))

##{

## www\_challenge("", "0");

## exit;

##}

##

##if (!db\_check\_to())

##{

## sl\_send\_reply("403","Forbidden auth ID");

## exit;

##}

if (!save("location"))

sl\_reply\_error();

exit;

}

}

Nếu bản tin yêu cầu là REGISTER thì lưu địa thông tin liên hệ trong bảng location sử dụng save(“location”).

if ($rU==NULL) {

# Bản tin yêu cầu không có username

sl\_send\_reply("484","Address Incomplete");

exit;

}

Bản tin yêu cầu sẽ bị hủy khi không có username trong địa chỉ.

if (!lookup("location","m")) {

switch ($retcode) {

case -1:

case -3:

t\_newtran();

t\_reply("404", "Not Found");

exit;

case -2:

sl\_send\_reply("405", "Method Not Allowed");

exit;

}

}

# when routing via usrloc, log the missed calls also

setflag(2);

route(1);

Đoạn mã trên cho ta thấy, thông tin của user được tìm kiếm trong cơ sở dữ liệu location. Tham số m trong lệnh lookup(“location”,”m”) cho phép lọc các bản tin. Vì thế chỉ có những contact hỗ trợ các bản tin đang được yêu cầu từ các contact được tìm thấy sẽ được cho phép.

Giá trị trả về từ hàm lookup() như sau:

1. - contact được tìm thấy và được trả về

-1 - không có contact nào được tìm thấy.

-2 - contact được tìm thấy , nhưng method không được hỗ trợ.

-3 - có lỗi trong suốt quá trình xử lý.

Hàm lookup(‘location”,”m”) sẽ thử tìm AOR của R-URI. Nếu AOR được tìm thấy ( UA đã đăng được đăng kí) thì nó sẽ thay đổi R-URI bằng địa chỉ IP của UA đó. Nếu AOR không được tìm thấy, thì nó sẽ trả về một bản tin lỗi (“404”,”Not Found”). Nếu AOR được tìm thấy thì nó sẽ kết thúc với route(1).

# CHƯƠNG IV: XÂY DỰNG MẠNG VOIP DÙNG OPENSIPS

1. Cài đặt OpenSIPS
   1. Hỗ trợ hệ điều hành và các gói phụ thuộc

OpenSIPS hỗ trợ các hệ điều hành sau đây:

Linux/i386, Linux/armv4l.

FreeBSD/i386.

OpenBSD/i386.

Solaris/sparc64.

NetBSD/sparc64.

Nếu muốn hỗ trợ các chức năng của OpenSIPS theo ý muốn , lựa chọn các chức năng của module để cài đặt sao cho phù hợp :

Openssl: nếu muốn biên dịch hỗ trợ TLS.

Libsctp: nếu muốn biên dịch hỗ trợ SCTP.

Libmysqlclient & libz (zlib)-libs: nếu muốn hỗ trợ mysql DB ( module db\_mysql).

Libpq / postgresql -libs: nếu muốn hỗ trợ hỗ trợ postgres DB (module db\_postgres).

Unixodbc -libs: nếu muốn hỗ trợ unixodbc DB (module db\_unixodbc).

Libexpat: nếu muốn hỗ trợ jabber gateway (jabber module) hoặc hỗ trợ XMPP gateway.

Libxml2: nếu muốn sử dụng module cpl-c (Call Processing Language) hoặc module Presence (presence và pua\*).

Libradius-ng -libs: nếu muốn sử dụng chức năng với hỗ trợ RADIUS xác nhận và tính cước, hỗ trợ nhóm…

Unixodbc – libs: nếu muốn hỗ trợ UNIXODBC như lớp đệm DB.

Libxmlrpc-c3 – libs: nếu muốn có hỗ trợ XML-RPC cho giao diện quản lý (MI).

Libperl: nếu muốn bộ nối PERL , perl scripting hỗ trợ khi config file (module perl).

Libsnmp9 – libs: nếu muốn thực hiện chức năng SNMP client (SNMP AgentX subagent) cho OpenSIPS.

Libldap libs: nếu muốn hỗ trợ LDAP.

Libconfuse: nếu muốn biên dịch module carrierroute.

* 1. Các bước cài đặt OpenSIPS trên Linux Ubuntu

Bước 1: Cài đặt các gói phụ thuộc cần phải cài trước khi cài OpenSIPS :

sudo apt-get install tên gói

apt-get install gcc bison flex make openssl libmysqlclient-dev libradiusclient-ng2 libradiusclient-ng-dev mysql-server libxmlrpc-c3-dev

Bước 2:

Dowload phần mềm về máy:

Cd /usr/src

wget <http://opensips.org/pub/opensips/latest/src/opensips-1.6.2-notls_src.tar.gz>

tar -xvfz opensips-1.6.2-notls\_src.tar.gz để giải nén

vi opensips-1.6.2-notls /Makefile

Tìm trong file Makefile loại bỏ chuỗi tên module mà người cấu hình muốn có biên dịch module đó trong:

exclude\_modules?= jabber cpl-c mysql postgres osp unixodbc \

avp\_radius auth\_radius group\_radius uri\_radius xmpp \

presence presence\_xml presence\_mwi pua pua\_bla pua\_mi\

pua\_usrloc pua\_xmpp rls mi\_xmlrpc perl snmpstats perlvdb \

ldap carrierroute h350 xcap\_client db\_berkeley seas\

Sau đó biên dịch OpenSIPS

# make prefix=/ all

Sau khi biên dịch OpenSIPS tiến hành cài OpenSIPS với lệnh dưới đây:

#make prefix=/ install

#mkdir /var/run/opensips

Vào thư mục debian

#cd /usr/src/opensips-1.6.2-notls/packaging/debian

Sao chép những tệp tin opensips.default và opensips.init vào những thư mục bên dưới.

#cp opensips.default /etc/default/opensips

#cp opensips.init /etc/init.d/opensips

#cd /etc/init.d

#chmod 777 opensips

Sửa trong tệp tin /etc/default/opensips tham số memory 128 MB và RUN\_OPENSIPS thành yes

Bước 3: Các thư mục quan trọng cần biết sau khi cài OpenSIPS

1. /sbin/local/sbin : Đây là thư mục chứa các tập tin.  
   - opensips: OpenSIPS   
   - opensipsdbctl: script để tạo và quản lý Databases   
   - opensipsctl: script quản lý OpenSIPS  
   - opensipsunix: script quản lý OpenSIPS qua unix sockets
2. /lib/opensips/modules/ : Thư mục chứa tất cả các module của OpenSIPS
3. /etc/opensips/opensips.cfg  
   Đây là file cấu hình OpenSIPS , một file quan trọng nhất để cấu hình OpenSIPS.

Bước 4: Thực hiện các lệnh cấu hình sau nếu cần persistence và authentication thì cần phải cài

# /etc/default/opensips

Thay đổi RUN\_OPENSIPS=no

Thành RUN\_OPENSIPS=yes

#sudo gedit /etc/init.d/opensips

Thay đổi DAEMON=/usr/sbin/opensips và RUN\_OPENSIPS=no

Thành DAEMON=/sbin/opensips và RUN\_OPENSIPS=yes

1. Cấu hình OpenSIPS để sử dung SQL

# vi /etc/opensips/opensips.cfg

Bỏ dấu # ở các dòng trong tập tin cấu hình opensips.cfg có liên quan tới xác nhận:

#loadmodule“db\_mysql.so”

#loadmodule “auth.so”

#loadmodule “auth\_db.so”

#modparam(“usrloc”, “db\_mode”, 2)

#modparam(“usrloc”, “db\_url”

# “mysql://opensips:opensipsrw@localhost/opensips”)

#modparam(“auth\_db”, “calculate\_ha1″, yes)

#modparam(“auth\_db”, “password\_column”, “password”)

#modparam(“auth\_db”, “db\_url”,

# “mysql://opensips:opensipsrw@localhost/opensips”)

#if (!www\_authorize("sip.org", "subscriber"))

{

www\_challenge("sip.org", "0");

break;

};

Thêm # vào trước dòng :

Thay modparam(“usrloc”, “db\_mode”, 0)

Thành# modparam(“usrloc”, “db\_mode”, 0)

1. Có thể thiết lập các giá trị cho các biến opensipsctlrc như sau :

#sudo vi /etc/opensips/opensipsctlrc

Loại bỏ dấu # trước các dòng :

# SIP\_DOMAIN=opensips.org ( tên miền SIP của user )

# DBENGINE=MYSQL ( loại hình cơ sở dữ liệu)

# DBHOST=localhost ( cơ sở dữ liệu chính)

# DBNAME=opensips (tên cơ sở dữ liệu)

# DBRWUSER=opensips (loại thuê bao có cơ sở dữ liệu dạng đọc/viết)

# DBRWPW=“opensipsrw”

#DBROUSER=opensipsro (loại thuê bao có cơ sở dữ liệu dạng chỉ đọc)

# DBROPW=opensipsro

# DBROOTUSER=”root”

# USERCOL=”username”

# INSTALL\_EXTRA\_TABLES=ask

# INSTALL\_PRESENCE\_TABLES=ask

Bỏ dấu # và thay đổi dòng lệnh : từ # PID\_FILE=/var/run/opensips.pid thành PID\_FILE=/var/run/opensips/opensips.pid

Bước 5: Tạo username hay số điện thoại SIP bằng công cụ opensipsctl.

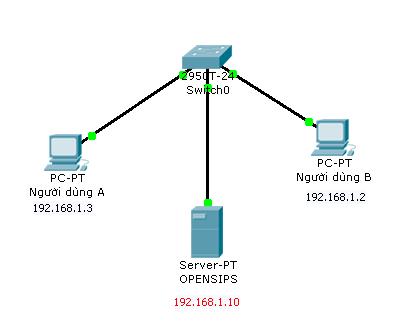
1. Trong /sbin/opensips/opensipsctlrc tìm đoạn:

#SIP\_DOMAIN=opensips.org Thay bằng SIP\_DOMAIN=IP của opensips server.

1. Tạo user với lệnh 'opensipsctl add <username><password>

# /sbin/opensipsctl add 1000 1000

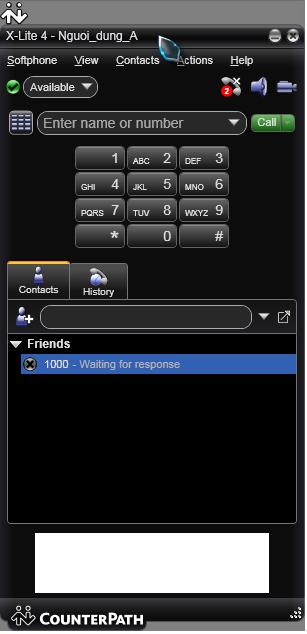
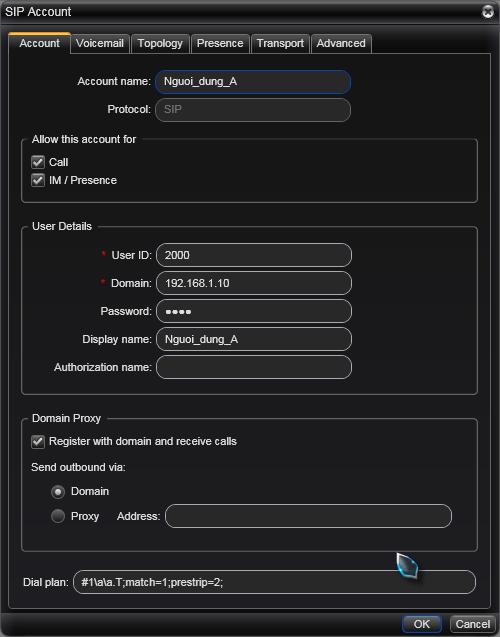
Bước 6: Tới đây xem như opensips đã chạy và công việc cài opensips kết thúc để test xem opensips đã chạy ổn định chưa gõ lệnh kiểm tra:

# /sbin/opensipsctl start hoặc /etc/init.d/opensips start

1. Thiết lập cuộc gọi từ PC – PC thông qua SIP server
2. Mô hình mạng VoIP dùng OpenSIPS
   1. Mô hình

Người dùng A sử dụng hệ điều hành winxp và phần mềm X-lite.

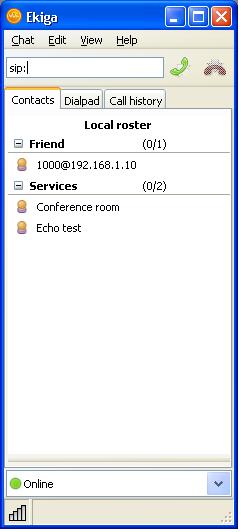
Với các thông tin đăng kí như trong hình:



1. Hình phần mềm điện thoại Xlite 4

Người dùng B sử dụng hệ điều hành winXP và phần mềm Ekiga

Thông tin đăng kí như hình dưới.



1. Hình phần mềm điện thoại Ekiga

Thực hiện cuộc gọi.

Người dùng A nhấn 1000 gọi cho người dùng B.

Hai bên nói chuyện với nhau.

Người dùng B gác máy.

Các bản tin SIP trong cuộc gọi được mô hình hóa như hình bên dưới:

**Media session RTP**

REGISTER

REGISTER

REGISTER

401 Unauthorized

401 Unauthorized

REGISTER

200 OK

INVITE

100 trying

INVITE

100 Trying

200 OK

200 OK

ACK

ACK

BYE

BYE

ACK

ACK

200 OK

1. Hình đường đi các bản tin SIP
2. Phân tích cuộc gọi:

Khi đăng kí người dùng trong OpenSIPS thì trong bảng subscriber sẽ lưu thông tin đăng kí trong đó.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID | username | Domain | password | Email\_address | Ha1 | Ha1b | rpid |
| 1 | 1000 | Opensips.org | 1000 |  | 99cee0333ad98b1c77413f76243b7675 | b22f5956d1eb95619bfe816a6b9832f0 | NULL |
| 2 | 2000 | Opensips.org | 2000 |  | d24f1cd07a238c83bfea55b4246d9525 | a5cf718029e9c4227571b35aae6c3381 | NULL |

1. :Bảng lưu thông tin người dùng

Khi Người dùng A lần đâu tiên sử dụng phần mềm để gọi, thì sau khi điền thông tin đăng kí bản tin REGISTER từ phần mềm của người dùng A sẽ gửi đến OpenSIPS để đăng kí thông tin người dùng.

*REGISTER sip:192.168.1.10 SIP/2.0*

*Via: SIP/2.0/UDP 192.168.1.3:30940;branch=z9hG4bK-d8754z-e141792baa96c5c8-1---d8754z-;rport*

*Max-Forwards: 70*

*Contact: <sip:2000@192.168.1.3:30940;rinstance=a452dff427f3d2f9>*

*To: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>*

*From: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=ca1f67ac*

*Call-ID: ZGY3MDc4MmQxYjdiM2ExNTJhMjE1OTU2YzBlMGFhMzM.*

*CSeq: 1 REGISTER*

*Expires: 3600*

*Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO*

*User-Agent: X-Lite 4 release 4.0 stamp 58832*

*Content-Length: 0*

Trường branch giúp phân biệt transaction này với transaction khác.

Tiêu đề Call-ID định nghĩa ra một dialog giữa các user agent.

Khi nhận được REGISTER, OpenSIPS sẽ xử lí bản tin thông qua các đoạn mã trong tệp tin opensips.conf và cụ thể là đoạn mã dưới đây :

if (is\_method("REGISTER"))

{

if (!www\_authorize("", "subscriber"))

{

www\_challenge("", "0");

exit;

}

if (!save("location"))

sl\_reply\_error();

exit;

}

Kiểm tra trong bản tin trên không có trường Authentication, vậy người dùng này chưa chứng thực. Nó sẽ gửi gói Unauthorized yêu cầu chứng thực với *realm* và *nonce*

***SIP/2.0 401 Unauthorized***

*Via: SIP/2.0/UDP 192.168.1.3:30940;branch=z9hG4bK-d8754z-e141792baa96c5c8-1---d8754z-;rport=30940*

*To: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=0ea86e9237480d326a1ba5873424d1c6.5053*

*From: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=ca1f67ac*

*Call-ID: ZGY3MDc4MmQxYjdiM2ExNTJhMjE1OTU2YzBlMGFhMzM.*

*CSeq: 1 REGISTER*

***WWW-Authenticate****: Digest realm="192.168.1.10", nonce="4d2422a300000002e0c249f05a557265b2373721e6612790"*

*Server: OpenSIPS (1.6.2-notls (i386/linux))*

*Content-Length: 0*

Bản tin này sẽ đính kèm thêm trường *WWW-Authenticate* với những thông tin cần cho việc chứng thực.

Nhận được bản tin này người dùng A gửi lại bản tin REGISTER với trường *Authorization* cùng với các thông tin cần thiết để chứng thực, và thuật toán dùng để chứng thực, trong trường hơp này la MD5.

Sau khi tính toán xong người dùng A gửi về một giá trị responde để so sánh với giá trị mà OpenSIPs tính được.

***REGISTER*** *sip:192.168.1.10 SIP/2.0*

*Via: SIP/2.0/UDP 192.168.1.3:30940;branch=z9hG4bK-d8754z-1b4936947ca093bc-1---d8754z-;rport*

*Max-Forwards: 70*

*Contact: <sip:2000@192.168.1.3:30940;rinstance=a452dff427f3d2f9>*

*To: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>*

*From: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=ca1f67ac*

*Call-ID: ZGY3MDc4MmQxYjdiM2ExNTJhMjE1OTU2YzBlMGFhMzM.*

*CSeq: 2 REGISTER*

*Expires: 3600*

*Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO*

*User-Agent: X-Lite 4 release 4.0 stamp 58832*

***Authorization:*** *Digest username="2000",realm="192.168.1.10",nonce="4d2422a300000002e0c249f05a557265b2373721e6612790",uri="sip:192.168.1.10",response="bd4360e1b2ca3d47405504020474753a",algorithm=MD5*

*Content-Length: 0*

Sau khi nhận được bản tin REGISTER lần hai và được kiểm tra với hàm WWW-Authenticate, thì nó sẽ lưu thông tin người dùng vào table “location” bằng hàm save() (được lấy từ module REGISTRA).

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Username | Domain | Contact | Reveived | Path | Expires | Q |
| Callid | Cseq | Last\_modified | Flags | Cflags | User\_agent | Socket | method |
| 53 | 2000 | null | sip:2000@192.168.1.3:30940;rinstance=a452dff427f3d2f9 | nul | nul | 2011-01-05 15:49:25 | -1.00 |
| ZGY3MDc4MmQxYjdiM2ExNTJhMjE1OTU2YzBlMGFhMzM | 2 | 2011-01-05 14:49:25 | 0 | 0 | X-Lite 4 release 4.0 stamp 58832 | udp:192.168.1.10:5060 | 5951 |

1. :Thông tin người dùng trong bảng location

Với người dùng B khi đăng kí các bước cúng giống với người dùng A.

Và thông tin trong bảng location như sau:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Username | Domain | Contact | Reveived | Path | Expires | Q |
| Callid | Cseq | Last\_modified | Flags | Cflags | User\_agent | Socket | method |
| 56 | 1000 | null | <sip:1000@192.168.1.2> | nul | nul | 2011-01-05 16:57:36 | 1.00 |
| ed29244c-2503-1910-961a-000c296362a0@anhtrung-1b1397 | 50 | 2011-01-05 15:57:36 | 0 | 0 | Ekiga/3.2.7 | udp:192.168.1.10:5060 | 22335 |

Sau khi đăng kí thành công. Người dùng A gọi cho người dùng B với username: 2000.

Đầu tiên người dùng A gửi bản tin INVITE cho OpenSIPS.

***INVITE*** *sip:1000@192.168.1.10 SIP/2.0*

*Via: SIP/2.0/UDP 192.168.1.3:30940;branch=z9hG4bK-d8754z-be9de0b831435b25-1---d8754z-;rport*

*Max-Forwards: 70*

*Contact: <sip:2000@192.168.1.3:30940>*

*To: <sip:1000@192.168.1.10>*

*From: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=afc18557*

*Call-ID: Yjk4ODZlZjViNDE4NjMwMTIzMjY4M2VhMzY3N2MwYmQ.*

*CSeq: 1 INVITE*

*Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO*

*Content-Type: application/sdp*

*Supported: replaces*

*User-Agent: X-Lite 4 release 4.0 stamp 58832*

*Content-Length: 402*

Bảng tin INVITE này chứa trường branch giúp nhận dạng transaction giữa người dùng A và OpenSIPS, tiêu đề Call-ID sẽ xuất hiện trong suốt cuộc gọi vì nó là thông tin nhận dạng dialog này.

Khi nhận được bản tin INVITE thì dựa vào đoạn mã :

if (!is\_method("REGISTER|MESSAGE"))

record\_route();

Xác định đây không phải là bản tin REGISTER hoặc MESSAGE, lúc này với hàm record\_route() sẽ thêm trường record\_route vào bản tin INVITE.

Sau đó OpenSIPS gửi bản tin 100 trying về cho người dùng A để thông báo đang xử lí bản tin yêu cầu. Sau đó dựa vào thông tin lưu trong bảng location để biết được địa chỉ IP của người dùng B và sửa địa chỉ IP trong URI của bản tin INVITE để gửi bản tin INVITE tới đích.

***SIP/2.0 100 Giving a try***

*Via: SIP/2.0/UDP 192.168.1.3:30940;branch=z9hG4bK-d8754z-be9de0b831435b25-1---d8754z-;rport=30940*

*To: <sip:1000@192.168.1.10>*

*From: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=afc18557*

*Call-ID: Yjk4ODZlZjViNDE4NjMwMTIzMjY4M2VhMzY3N2MwYmQ.*

*CSeq: 1 INVITE*

*Server: OpenSIPS (1.6.2-notls (i386/linux))*

*Content-Length: 0*

***INVITE sip:1000@192.168.1.2 SIP/2.0***

*Record-Route: <sip:192.168.1.10;lr=on>*

*Via: SIP/2.0/UDP 192.168.1.10;branch=z9hG4bKef3f.5e50bb33.0*

*Via: SIP/2.0/UDP 192.168.1.3:30940;received=192.168.1.3;branch=z9hG4bK-d8754z-be9de0b831435b25-1---d8754z-;rport=30940*

*Max-Forwards: 69*

*Contact: <sip:2000@192.168.1.3:30940>*

*To: <sip:1000@192.168.1.10>*

*From: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=afc18557*

*Call-ID: Yjk4ODZlZjViNDE4NjMwMTIzMjY4M2VhMzY3N2MwYmQ.*

*CSeq: 1 INVITE*

*Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO*

*Content-Type: application/sdp*

*Supported: replaces*

*User-Agent: X-Lite 4 release 4.0 stamp 58832*

*Content-Length: 402*

Ta thấy tiêu đề record-route đã được thêm vào.

Và với branch mới *z9hG4bKef3f.5e50bb33.0* này thì một transaction mới giữa OpenSIPS và người dùng B đã được thiết lập

Sau khi nhận được bản tin INVITE người dùng B gửi bản tin 100 Trying tới OpenSIPS dựa vào thông tin trong trường VIA để thông báo là nó đã nhận được và đang xử lí bản tin INVITE.

*SIP/2.0 100 Trying*

*CSeq: 1 INVITE*

*Via: SIP/2.0/UDP 192.168.1.10;branch=z9hG4bKef3f.5e50bb33.0*

*Via: SIP/2.0/UDP 192.168.1.3:30940;received=192.168.1.3;branch=z9hG4bK-d8754z-be9de0b831435b25-1---d8754z-;rport=30940*

*From: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=afc18557*

*Call-ID: Yjk4ODZlZjViNDE4NjMwMTIzMjY4M2VhMzY3N2MwYmQ.*

*To: <sip:1000@192.168.1.10>*

*Contact: <sip:1000@192.168.1.2>*

*Content-Length: 0*

*Record-Route: <sip:192.168.1.10;lr=on>*

Sau đó nó gửi bản tin 180 Ringing cho người dùng B thông qua OpenSIPs để thông báo cho người dùng A bên người dùng B đang đỗ chuông.

***SIP/2.0 180 Ringing***

*CSeq: 1 INVITE*

*Via: SIP/2.0/UDP 192.168.1.10;branch=z9hG4bKef3f.5e50bb33.0*

*Via: SIP/2.0/UDP 192.168.1.3:30940;received=192.168.1.3;branch=z9hG4bK-d8754z-be9de0b831435b25-1---d8754z-;rport=30940*

*User-Agent: Ekiga/3.2.7*

*From: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=afc18557*

*Call-ID: Yjk4ODZlZjViNDE4NjMwMTIzMjY4M2VhMzY3N2MwYmQ.*

*To: <sip:1000@192.168.1.10>;tag=34bb254f-2503-1910-961f-000c296362a0*

*Allow: INVITE,ACK,OPTIONS,BYE,CANCEL,SUBSCRIBE,NOTIFY,REFER,MESSAGE,INFO,PING*

*Content-Length: 0*

*Record-Route:* [*sip:192.168.1.10;lr=on*](sip:192.168.1.10;lr=on)

Và OpenSIP sẽ dựa vào trường VIA và gửi cho người dùng A.

***SIP/2.0 180 Ringing***

*CSeq: 1 INVITE*

*Via: SIP/2.0/UDP 192.168.1.3:30940;received=192.168.1.3;branch=z9hG4bK-d8754z-be9de0b831435b25-1---d8754z-;rport=30940*

*User-Agent: Ekiga/3.2.7*

*From: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=afc18557*

*Call-ID: Yjk4ODZlZjViNDE4NjMwMTIzMjY4M2VhMzY3N2MwYmQ.*

*To: <sip:1000@192.168.1.10>;tag=34bb254f-2503-1910-961f-000c296362a0*

*Allow: INVITE,ACK,OPTIONS,BYE,CANCEL,SUBSCRIBE,NOTIFY,REFER,MESSAGE,INFO,PING*

*Content-Length: 0*

*Record-Route: <sip:192.168.1.10;lr=on>*

Khi người dùng B nhấc máy bản tin 200 Ok từ người dùng B sẽ gởi tới người dùng A thông báo bên B đã nhấc máy.

***SIP/2.0 200 OK***

*CSeq: 1 INVITE*

*Via: SIP/2.0/UDP 192.168.1.10;branch=z9hG4bKef3f.5e50bb33.0*

*Via: SIP/2.0/UDP 192.168.1.3:30940;received=192.168.1.3;branch=z9hG4bK-d8754z-be9de0b831435b25-1---d8754z-;rport=30940*

*User-Agent: Ekiga/3.2.7*

*From: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=afc18557*

*Call-ID: Yjk4ODZlZjViNDE4NjMwMTIzMjY4M2VhMzY3N2MwYmQ.*

*To: <sip:1000@192.168.1.10>;tag=34bb254f-2503-1910-961f-000c296362a0*

*Contact: <sip:1000@192.168.1.2>*

*Allow: INVITE,ACK,OPTIONS,BYE,CANCEL,SUBSCRIBE,NOTIFY,REFER,MESSAGE,INFO,PING*

*Content-Type: application/sdp*

*Content-Length: 213*

*Record-Route:* [*sip:192.168.1.10;lr=on*](sip:192.168.1.10;lr=on)

Nhận được bản tin OK OpenSIPS sẽ chuyển bản tin tới người dùng A nhờ vào tiêu đề VIA

***SIP/2.0 200 OK***

*CSeq: 1 INVITE*

*Via: SIP/2.0/UDP 192.168.1.3:30940;received=192.168.1.3;branch=z9hG4bK-d8754z-be9de0b831435b25-1---d8754z-;rport=30940*

*User-Agent: Ekiga/3.2.7*

*From: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=afc18557*

*Call-ID: Yjk4ODZlZjViNDE4NjMwMTIzMjY4M2VhMzY3N2MwYmQ.*

*To: <sip:1000@192.168.1.10>;tag=34bb254f-2503-1910-961f-000c296362a0*

*Contact: <sip:1000@192.168.1.2>*

*Allow: INVITE,ACK,OPTIONS,BYE,CANCEL,SUBSCRIBE,NOTIFY,REFER,MESSAGE,INFO,PING*

*Content-Type: application/sdp*

*Content-Length: 213*

*Record-Route: <sip:192.168.1.10;lr=on>*

Bên A nhận được bản tin 200 Ok nó sẽ gửi bản tin ACK về cho người dùng B tới OpenSIPS.

***ACK sip:1000@192.168.1.2 SIP/2.0***

*Via: SIP/2.0/UDP 192.168.1.3:30940;branch=z9hG4bK-d8754z-96fb927b789729e0-1---d8754z-;rport*

*Max-Forwards: 70*

*Route: <sip:192.168.1.10;lr>*

*Contact: <sip:2000@192.168.1.3:30940>*

*To: <sip:1000@192.168.1.10>;tag=34bb254f-2503-1910-961f-000c296362a0*

*From: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=afc18557*

*Call-ID: Yjk4ODZlZjViNDE4NjMwMTIzMjY4M2VhMzY3N2MwYmQ.*

*CSeq: 1 ACK*

*User-Agent: X-Lite 4 release 4.0 stamp 58832*

*Content-Length: 0*

Ở đây trong bản tin ACK có xuất hiện trường Route, tức là bản tin sẽ được định tuyến dựa vào thông tin trong này và lờ đi thông tin định tuyến URI của bản tin yêu cầu cụ thể là *1000@192.168.1.2 ,* do đó bản tin sẽ không được gửi đến trực tiếp người dùng B.

Nhận được bản tin ACK từ người dùng A, OpenSIPs sẽ chuyển bản tin đến người dùng B nhờ vào URI của bản tin yêu cầu với hàm t\_relay() trong module TM, lúc này tiêu đề Route đã mất.

***ACK sip:1000@192.168.1.2 SIP/2.0***

*Via: SIP/2.0/UDP 192.168.1.10;branch=z9hG4bKef3f.5e50bb33.2*

*Via: SIP/2.0/UDP 192.168.1.3:30940;received=192.168.1.3;branch=z9hG4bK-d8754z-96fb927b789729e0-1---d8754z-;rport=30940*

*Max-Forwards: 69*

*Contact: <sip:2000@192.168.1.3:30940>*

*To: <sip:1000@192.168.1.10>;tag=34bb254f-2503-1910-961f-000c296362a0*

*From: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=afc18557*

*Call-ID: Yjk4ODZlZjViNDE4NjMwMTIzMjY4M2VhMzY3N2MwYmQ.*

*CSeq: 1 ACK*

*User-Agent: X-Lite 4 release 4.0 stamp 58832*

*Content-Length: 0*

Sau khi nhận được bản tin ACK cuộc gọi đã diễn ra giữa người dùng A và người dùng B

Cuộc hội thoại kết thúc , người dùng B gác máy bản tin BYE sẽ gửi tới người dùng A thông báo người dùng B đã gác máy.

***BYE sip:2000@192.168.1.3:30940 SIP/2.0***

*Route: <sip:192.168.1.10;lr=on>*

*CSeq: 2 BYE*

*Via: SIP/2.0/UDP 192.168.1.2:5060;branch=z9hG4bK10313d4f-2503-1910-9620-000c296362a0;rport*

*From: <sip:1000@192.168.1.10>;tag=34bb254f-2503-1910-961f-000c296362a0*

*Call-ID: Yjk4ODZlZjViNDE4NjMwMTIzMjY4M2VhMzY3N2MwYmQ.*

*To: "Nguoi\_dung\_A" <sip:2000@192.168.1.10>;tag=afc18557*

*Contact: <sip:1000@192.168.1.2>*

*Content-Length: 0*

*Max-Forwards: 70*

Tương tự ở trên lúc này bản tin sẽ không được gửi trực tiếp tới người dùng A mà sẽ gửi tới OpenSIPs thông qua thông tin định tuyến trong tiêu đề Route.

Và khi nhận được bản tin này OpenSIPs sẽ gửi tới cho người dùng A dựa vào URI của bản tin BYE lúc này không có tiêu đề Route.

***BYE sip:2000@192.168.1.3:30940 SIP/2.0***

*CSeq: 2 BYE*

*Via: SIP/2.0/UDP 192.168.1.10;branch=z9hG4bKbf3f.ea00dad1.0*

*Via: SIP/2.0/UDP 192.168.1.2:5060;received=192.168.1.2;branch=z9hG4bK10313d4f-2503-1910-9620-000c296362a0;rport=5060*

*From: <sip:1000@192.168.1.10>;tag=34bb254f-2503-1910-961f-000c296362a0*

*Call-ID: Yjk4ODZlZjViNDE4NjMwMTIzMjY4M2VhMzY3N2MwYmQ.*

*To: "Nguoi\_dung\_A" <sip:2000@192.168.1.10>;tag=afc18557*

*Contact: <sip:1000@192.168.1.2>*

Bên A nhận được bản tin BYE biết được bên B đã gác máy thì nó gửi bản tin 200 OK về bên B là đồng ý và tự động gác máy.

***SIP/2.0 200 OK***

*Via: SIP/2.0/UDP 192.168.1.10;branch=z9hG4bKbf3f.ea00dad1.0*

*Via: SIP/2.0/UDP 192.168.1.2:5060;received=192.168.1.2;branch=z9hG4bK10313d4f-2503-1910-9620-000c296362a0;rport=5060*

*Contact: <sip:2000@192.168.1.3:30940>*

*To: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=afc18557*

*From: <sip:1000@192.168.1.10>;tag=34bb254f-2503-1910-961f-000c296362a0*

*Call-ID: Yjk4ODZlZjViNDE4NjMwMTIzMjY4M2VhMzY3N2MwYmQ.*

*CSeq: 2 BYE*

*User-Agent: X-Lite 4 release 4.0 stamp 58832*

*Content-Length: 0*

***SIP/2.0 200 OK***

*Via: SIP/2.0/UDP 192.168.1.2:5060;received=192.168.1.2;branch=z9hG4bK10313d4f-2503-1910-9620-000c296362a0;rport=5060*

*Contact: <sip:2000@192.168.1.3:30940>*

*To: "Nguoi\_dung\_A"<sip:2000@192.168.1.10>;tag=afc18557*

*From: <sip:1000@192.168.1.10>;tag=34bb254f-2503-1910-961f-000c296362a0*

*Call-ID: Yjk4ODZlZjViNDE4NjMwMTIzMjY4M2VhMzY3N2MwYmQ.*

*CSeq: 2 BYE*

*User-Agent: X-Lite 4 release 4.0 stamp 58832*

# KẾT LUẬN VÀ HẠN CHẾ, HƯỚNG MỞ CỦA ĐỀ TÀI

## Kết luận

SIP là giao thức báo hiệu linh hoạt, mềm dẻo và rất thích hợp khi sử dụng trong mạng IP. SIP ngày càng trở nên phổ biến và đã trở thành một giao thức báo hiệu chính trong mạng VoIP. SIP có phần mở rộng đồ sộ cho những tính năng, ứng dụng, dịch vụ và môi trường cụ thể. Vì vậy vấn đề quan trọng cần xác định trước tiên đối với nhà cung cấp dịch vụ là xác định phương hướng phát triển dịch vụ, trên cơ sở đó sẽ tiếp tục xây dựng tài liệu chuẩn cho các phần mở rộng liên quan.

OpenSIPS là SIP Proxy Server mã nguồn mở chủ đạo. OpenSIPS có khả năng định tuyến hàng tỉ phút và xử lý hàng triệu hoạt động các cuộc gọi điện thoại VoIP mỗi tháng. Nổi bật về tinh năng linh hoạt và sự ổn định, OpenSIPS đang liên tục phát triển các các tính năng đó trong Viễn thông.

Rất phổ biến trong các dịch vụ cung cấp VoIP/Internet Telephony, OpenSIPS cung cấp nền tảng truyền thông hoàn chỉnh : Vocie ,Video, Instant Messaging và Presence. Hơn thế nữa.Cấu hình OpenSIPS server không phải là dễ dàng ,nhưng đó là chìa khóa thành công và bảo mật cho dự án IP. Với đặc tính linh hoạt của OpenSIPS cho phép triển khai dich vụ tiên tiến và tiết kiệm được thời gian và chi phí.

## Hạn chế và hướng phát triển của đề tài

### Hạn chế của đề tài

Chưa phát triển mở rộng thêm các module trong OPENSIPS. Chưa đi sâu vào các đoạn mã bên trong module.

### Hướng phát triển của đề tài

Phát triển mở rộng các module, tạo ra nhiều tính năng hữu ích cho OPENSIPS dựa trên chuẩn RFC 3261.

# TÀI LIỆU THAM KHẢO

[1].Trang web <http://www.opensips.org/>

[2]. Building Telephony Systems with OpenSIPS 1.6 tác giả Flavio E.Goncalves

[3]. SIP - Understanding the Session Initiation Protocol tác giả Alan B.Johnston.

[4]. RFC 3261. SIP - Session Initiation Protocol