

# TÌM HIỂU VÀ TRIỂN KHAI KAMAILIO

## Nhóm L52

### 1. Danh sách thành viên và công việc

Họ và tên	Mã SV	Công việc	Tiến độ
Nguyễn Thế Vinh	175A071562	- Cài đặt Kamailio  - Hướng dẫn cài đặt - Hướng dẫn quản trị Kamailio - Triển khai Kamailio với Siremis	- Cài đặt hoàn thành - Hoàn thành - Hoàn thành  -20%
Nguyễn Khắc Kỷ	175A071312	- Cài đặt Kamailio  - Tìm hiểu tài liệu về Kamailio : khái niệm, chức năng, hoạt động. - Phân tích ưu, nhược điểm - Triển khai Kamailio với Siremis	- Cài đặt hoàn thành - Hoàn thành  - Hoàn thành  -20%

### 2. Nội dung nghiên cứu

#### a. Kamailio là gì?

- Kamailio còn được biết đến là 1 OpenSER, là một máy chủ SIP ( Session Initiation Protocol) (tạm hiểu là 1 giao thức truyền tín hiệu) và được nhận giấy phép công cộng từ GNU  
Nó có thể được thiết lập thành 1 nhà đăng ký SIP, hay là một máy chủ proxy, hay được dùng để điều hướng server.

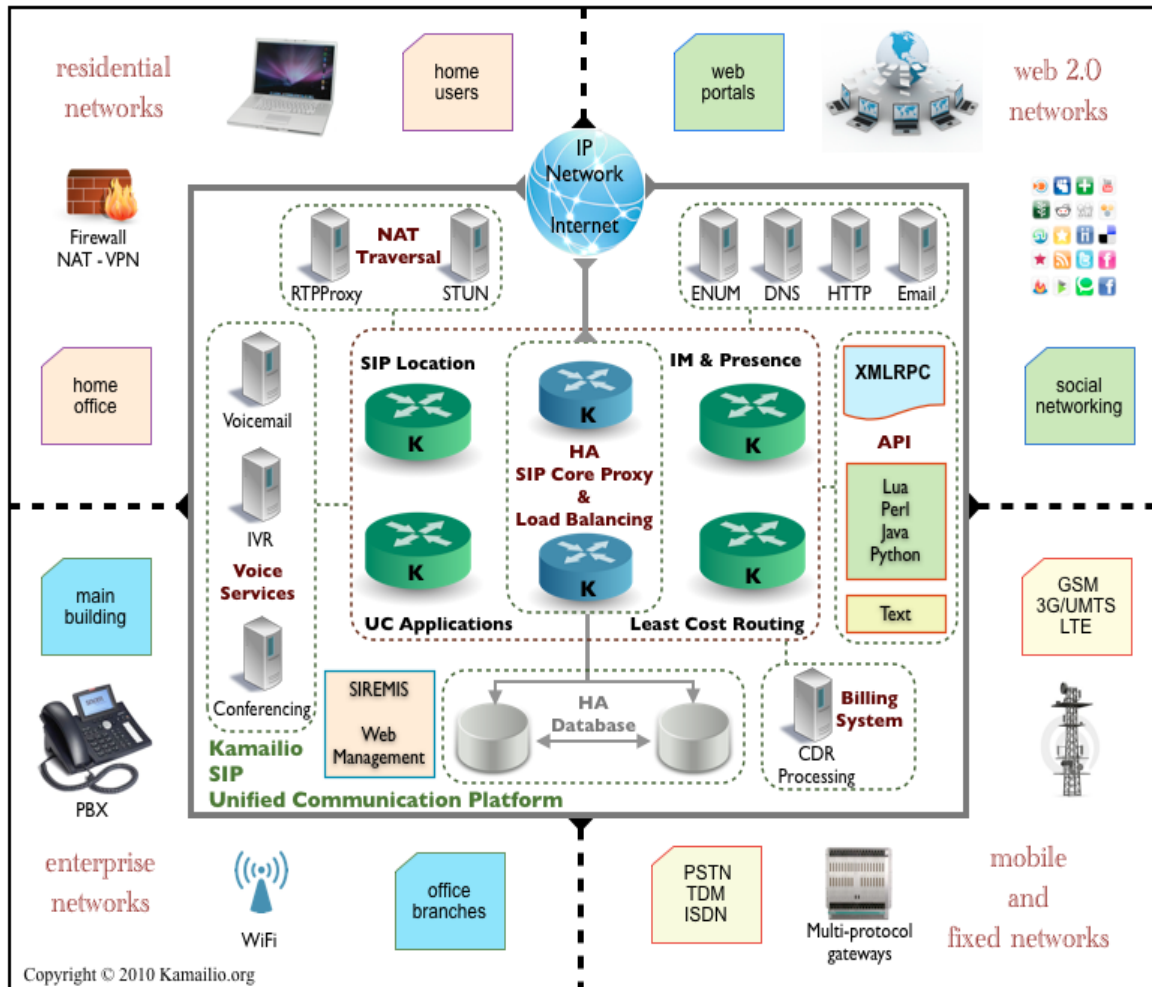
- Kamailio là một từ của người Hawaii nó có nghĩa là nói chuyện hay là giao tiếp’.
- Kamailio được viết thuần bởi C trong cấu trúc tổ chức của nó, nó có thể được thiết lập phù hợp với rất nhiều kịch bản cho những văn phòng nhỏ, hay một tổ chức.
- Kamailio được sử dụng rộng rãi trong ISP (nhà cung cấp dịch vụ mạng) để cung cấp dịch vụ gọi điện. Nhưng ngoài ra nó còn có thể làm: Bảo mật tường lửa (SIP), công cụ điều hướng giá rẻ, nhắn tin tức thì và video call...

Và phần mềm nhóm em đang nghiên cứu hướng đến để cung cấp một dịch vụ gọi điện giữa 2 người.

#### **b. Kamailio có chức năng cụ thể là gì ?**

- Như đã nói sơ qua ở phần trên, Kamailio có những chức năng chủ yếu là registrar server, sip proxy và load balancer để phân chia tải đến các PBX (có thể là Asterisk hoặc Freeswitch). Với server Kamailio 4GB RAM có thể phục vụ tới 300.000 thuê bao, là một con số không nhỏ với một open source telecom.
- Các bạn có thể xem thêm những tính năng khác mà Kamailio hỗ trợ:
  - + Hỗ trợ giao tiếp thông qua UDP, TCP , TLS và SCTP
  - + Ipv4 và Ipv6
  - + Xử lý tiến trình không đồng bộ TCP
  - + Bảo mật giao tiếp IP và Network Authentication...
- Link: Kamailio offer:
 

<http://www.kamailio.org/w/features/?fbclid=IwAR3cUI7-IJ2VGXnLenu1kPVLPIXIpebvXoZmqDCF894qsQQt3CJiKYOHIffk>



### c. Hoạt động của Kamilio hoặc So sánh Ưu/Nhược điểm

- Ưu/Nhược điểm :

+. Như đã nói ở phần trên Kamilio Server dựa trên SIP (giao thức khởi tạo phiên) về cơ bản giao thức này là một giao thức gửi tín hiệu để khởi tạo duy trì, ngắt phiên trong thế giới thực những thứ như :giọng nói ,video, và các ứng dụng tin nhắn.

+. SIP có thể được sử dụng để khởi tạo phiên cho nhóm-2(unicast) hay nhóm-nhiều (multicast). Nó cũng cho phép tùy chỉnh những cuộc gọi đang tồn tại. Sự điều chỉnh này có thể liên quan đến địa chỉ hoặc cổng nguồn, mời nhiều người tham dự , hay thêm hoặc xóa media streams.

+. Nếu chọn Kamailio bạn sẽ có một cộng đồng người dùng hoạt động rộng rãi, phát triển nhanh chóng, những nhà phát triển thân thiện và sẵn sàng giúp

- So sánh với một vài server SIP khác như Opensips thì Kamailio có kho modules nhiều hơn hẳn.

#### **d. Hướng dẫn cài đặt Kamailio trên Ubuntu**

Bước 1 : Cài đặt máy chủ Apache2 HTTP

- Cài đặt Apache2 HTTP

```
#sudo apt update
#sudo apt install apache2
```

- Sau khi cài đặt Apache2 thì khởi động 3 lệnh dừng , khởi động và cho phép dịch vụ Apache2 luôn khởi động với máy chủ khởi động.

```
#sudo systemctl stop apache2.service
#sudo systemctl start apache2.service
#sudo systemctl enable apache2.service
```

Bước 2 : Cài đặt máy chủ cơ sở dữ liệu MariaDB

- Cài đặt MariaDB Database Server

```
#sudo apt install mariadb-server mariadb-client
```

- Sau khi cài đặt MariaDB thì khởi động 3 lệnh dừng , khởi động và cho phép dịch vụ MariaDB luôn khởi động khi máy chủ khởi động

```
#sudo systemctl stop mariadb.service
#sudo systemctl start mariadb.service
#sudo systemctl enable mariadb.service
```

- Sau đó, chạy các lệnh bên dưới để bảo mật máy chủ MariaDB bằng cách tạo mật khẩu gốc và không cho phép truy cập root từ xa

```
#sudo mysql_secure_installation
```

Bước 3 : Cài đặt PHP các Modules liên quan

- Thêm kho lưu trữ bên thứ ba để nâng cấp lên PHP 7.2

```
#sudo apt-get install software-properties-common
#sudo add-apt-repository ppa:ondrej/php
```

- Cập nhật và nâng cấp lên PHP 7.2

```
#sudo apt update
```

- Cài đặt PHP 7.2 và các modules liên quan

```
#sudo apt install php7.2 libapache2-mod-php7.2 php7.2-common
php7.2-curl php7.2-intl php7.2-mbstring php7.2-xmlrpc php7.2-mysql
php7.2-gd php7.2-imagick php-pear php7.2-xml
Php7.2-cli php7.2-zip php7.2-sqlite
```

- Mở tệp cấu hình mặc định PHP cho Apache2

```
#sudo nano /etc/php/7.2/apache2/php.ini
```

- Thay đổi các giá trị như bảng rồi lưu

- file\_uploads = Bật
- allow\_url\_fopen = Bật
- short\_open\_tag = Bật
- bộ nhớ\_limit = 256M
- upload\_max\_filesize = 100M
- max\_execut\_time = 360
- max\_input\_vars = 1500
- date.timezone = Mỹ / Chicago

- Khởi động lại Apache2

```
#sudo systemctl restart apache2.service
```

- Kiểm tra cài đặt PHP 7.2 với Apache2, hãy tạo tệp phpinfo.php trong thư mục gốc Apache2

```
#sudo nano /var/www/html/phpinfo.php
```

- Nhập nội dung vào tệp và lưu

```
<php phpinfo( ) ; ?>
```

- Sau đó duyệt đến máy chủ của bạn : <http://localhost/phpinfo.php>

#### Bước 4 : Tải Kamailio

- Tải xuống và thêm khóa GPG cho kho lưu trữ của nó

```
#wget -O- http://deb.kamailio.org/kamailiodebkey.gpg | sudo apt-
key add -
```

- Tạo tệp lưu trữ

#sudo nano /etc/apt/sources.list.d/kamailio.list
- Chép, dán và lưu vào tệp
#deb http://deb.kamailio.org/kamailio52 bionic main #deb-src http://deb.kamailio.org/kamailio52 bionic main
- Cập nhật các gói Ubuntu và cài đặt Kamailio
#sudo apt update #sudo apt install kamailio kamailio-mysql-modules kamailio-websocket-modules
- Mở tệp cấu hình Kamailio
#sudo nano /etc/kamailio/kamctlrc
- Chỉnh sửa trong tệp và lưu
SIP_DOMAIN = kamailio.example.com DBENGINE = MYSQL
- Tạo cơ sở dữ liệu, người dùng và các bảng cần thiết cho Kamailio
#kamdbctl create
- Nếu quyền truy cập bị từ chối bởi <b>root@localhost</b> thì :
<ul style="list-style-type: none"> <li>- Đăng nhập vào máy chủ MariaDB chạy lệnh : #sudo mysql -u root</li> <li>- Tắt xác thực plugin cho người dùng root use mysql; update user set plugin="" where User='root'; flush privileges; exit</li> <li>- Khởi động lại #sudo systemctl restart mariadb.service</li> </ul>
- Mở /etc/kamailio/kamailio.cfg bằng cách chạy lệnh
#sudo nano /etc/kamailio/kamailio.cfg
- Thêm và lưu vào tệp
#!define WITH_MYSQL #!define WITH_AUTH #!define WITH_USRLOCDB #!define WITH_ACCDB
- Áp dụng các lệnh thay đổi
#sudo systemctl restart kamailio

Bước 5 : Cài đặt và cấu hình Siremis

- Cài đặt XML\_RPC

```
#sudo pear install XML_RPC2
```

- Thay đổi thư mục thành root web Apache, cài đặt git và tải xuống các gói Siremis.

```
#cd / var / www /  
#sudo apt install  
#sudo git clone https://github.com/asipto/siremis
```

- Vào thư mục Siremis và chuẩn bị môi trường

```
#cd /var/www/siremis  
#sudo make prepare24  
#sudo make chown
```

- Định cấu hình tệp cấu hình trang Apache2 cho Siremis

```
#sudo nano /etc/apache2/sites-available/siremis.conf
```

- Chèn vào tập tin và lưu. Thay thế tên miền (ServerName) và vị trí thư mục gốc (ServerAlias)

```
- <VirtualHost *:80>  
-   ServerAdmin admin@example.com  
-   DocumentRoot /var/www/siremis/siremis  
-   Alias /siremis "/var/www/siremis/siremis"  
-   ServerName kamilio.example.com  
-   ServerAlias www.kamilio.example.com  
-  
-   <Directory "/var/www/siremis/siremis">  
-       Options Indexes FollowSymLinks MultiViews  
-       AllowOverride All  
-       Require all granted  
-  
-       <FilesMatch "\.xml$">  
-           Require all denied  
-       </FilesMatch>  
-       <FilesMatch "\.inc$">  
-           Require all denied  
-       </FilesMatch>  
-   </Directory>  
-  
-   ErrorLog ${APACHE_LOG_DIR}/error.log  
-   CustomLog ${APACHE_LOG_DIR}/access.log combined  
-
```

```
- </VirtualHost>
```

#### Bước 6 : Kích hoạt Modules và viết

- Sau khi định cấu hình Virtualhost ở trên, bật nó :

```
#sudo a2ensite siremis.conf  
#sudo a2enmod viết lại  
#sudo systemctl khởi động lại apache2.service
```

- Cấp quyền truy cập cho người dùng Siremis vào cơ sở dữ liệu của nó

```
#sudo mysql -u root -p
```

- Chạy lệnh SQL

```
- GRANT ALL PRIVILEGES ON siremis.*  
  TO siremis@localhost IDENTIFIED BY 'password';  
- FLUSH PRIVILEGES;  
- EXIT;
```

#### **d. Hướng dẫn quản trị Kamailio**

Thiết lập Module Call\_Cotrol của Kamailio để nói chuyện giữa hai người.

- Module này cho phép một người giới hạn thời lượng cuộc gọi và tự động kết thúc chúng khi vượt quá giới hạn áp đặt. Trường hợp sử dụng chính của nó là để thực hiện một hệ thống trả trước, nhưng nó cũng có thể được sử dụng để áp đặt giới hạn toàn cầu cho tất cả các cuộc gọi được xử lý bởi proxy.
- Một ứng dụng bên ngoài được gọi là call\_control theo dõi các cuộc gọi có giới hạn thời gian và tự động kết thúc chúng khi chúng vượt quá nó. Ứng dụng này nhận được yêu cầu từ Kamailio và đưa ra yêu cầu cho công cụ xếp hạng để tìm hiểu xem cuộc gọi có cần được giới hạn hay không. Khi một cuộc gọi kết thúc (hoặc kết thúc), nó cũng sẽ hướng dẫn công cụ xếp hạng ghi nợ số dư cho người gọi với số tiền đã tiêu thụ. Ứng dụng callcontrol có sẵn từ <http://callcontrol.ag-projects.com/>
- Công cụ xếp hạng được sử dụng để tính thời hạn dựa trên tín dụng của người gọi và giá đích và để ghi nợ số dư của người gọi sau khi cuộc gọi kết thúc.



Điều này có sẵn như là một phần của CDRTool từ <http://cdrtool.ag-projects.com/>

- Ứng dụng callcontrol chạy trên cùng một máy với Kamailio và chúng giao tiếp qua ổ cắm hệ thống tập tin, trong khi công cụ đánh giá có thể chạy trên một máy chủ khác và giao tiếp với ứng dụng callcontrol bằng kết nối TCP.
- Callcontrol được gọi bằng cách gọi hàm `call_control ()` cho HÓA ĐƠN ban đầu của mỗi cuộc gọi mà chúng tôi muốn áp dụng giới hạn. Điều này sẽ kết thúc như một yêu cầu cho ứng dụng điều khiển cuộc gọi, ứng dụng này sẽ kết hợp công cụ đánh giá trong một giới hạn thời gian cho người gọi và đích đã cho. Công cụ xếp hạng sẽ xác định xem điểm đến có bất kỳ chi phí liên quan nào không và liệu người gọi có bất kỳ giới hạn tín dụng nào không và nếu có sẽ trả lại thời gian anh ta được phép gọi điểm đến đó. Nếu không, nó sẽ chỉ ra rằng không có giới hạn liên quan đến cuộc gọi. Nếu có giới hạn, ứng dụng điều khiển cuộc gọi sẽ giữ lại phiên và gắn bộ hẹn giờ với nó sẽ hết hạn sau thời gian nhất định khiến nó gọi lại cho Kamailio với yêu cầu kết thúc hợp thoại. Nếu công cụ đánh giá trả về rằng không có giới hạn cho cuộc gọi, phiên sẽ bị loại bỏ bởi ứng dụng điều khiển cuộc gọi và nó sẽ cho phép nó tiến hành bất kỳ giới hạn nào. Một phản hồi thích hợp được trả về mô-đun `call_control` sau đó được trả về bởi lệnh gọi hàm `call_control ()` và cho phép tập lệnh đưa ra quyết định dựa trên câu trả lời.