

**TẬP ĐOÀN VIỄN THÔNG QUÂN ĐỘI**

**TRUNG TÂM PHẦN MỀM QUẢN TRỊ**

**TÀI LIỆU HƯỚNG DẪN CÀI ĐẶT**

**HỆ THỐNG SSO**

**Mã hiệu dự án:** CNTT\_ESS\_18009\_ERP

**Mã hiệu tài liệu:**

**Hà Nội, 2019**

**BẢNG GHI NHẬN THAY ĐỔI**

\*A – Tạo mới, M – Sửa đổi, D – Xóa bỏ

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Ngày**  **thay đổi** | **Vị trí**  **thay đổi** | **A\***  **M, D** | **Nguồn gốc** | **Phiên**  **bản cũ** | **Mô tả thay đổi** | **Phiên**  **bản mới** |
|  |  | A |  | N/A |  | v1.0 |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

**TRANG KÝ**

Người lập: Lê Mạnh Hùng...........<Ngày>...................................................

Kỹ sư phát triển

Người xem xét:.Lê Văn Việt.........<Ngày>...................................................

Trưởng BU

Người xem xét:.Nguyễn Thị Thoa…....<Ngày>...................................................

Kỹ sư kiểm thử

Người phê duyệt: Lê Văn Việt...............<Ngày>...................................................

Trưởng BU

MỤC LỤC

[1. GIỚI THIỆU 5](#_Toc32939906)

[***1.1.*** ***Mục đích, ý nghĩa của tài liệu*** 5](#_Toc32939907)

[***1.2.*** ***Phạm vi tài liệu*** 5](#_Toc32939908)

[***1.3.*** ***Các thuật ngữ và từ viết tắt*** 5](#_Toc32939909)

[***1.4.*** ***Các thành phần của hệ thống*** 5](#_Toc32939910)

[***1.5.*** ***Tài liệu hướng dẫn cài đặt bao gồm các phần sau:*** 8](#_Toc32939911)

[2. YÊU CẦU VỀ MÔI TRƯỜNG CÀI ĐẶT 9](#_Toc32939912)

[***2.1.*** ***Yêu cầu môi trường*** 9](#_Toc32939913)

[***2.2.*** ***Yêu cầu về cơ sở dữ liệu*** 9](#_Toc32939914)

[3. QUY TRÌNH CÀI ĐẶT 9](#_Toc32939915)

[***3.1.*** ***Cài đặt cơ sở dữ liệu*** 9](#_Toc32939916)

[***3.2.*** ***Cài đặt máy chủ*** 10](#_Toc32939917)

[3.2.1. Cài đặt JDK 10](#_Toc32939918)

[3.2.2. Cài đặt Tomcat 11](#_Toc32939919)

[***3.3.*** ***Cài đặt và kiểm tra dịch vụ*** 15](#_Toc32939920)

[3.3.1. Triển khai webapp 15](#_Toc32939921)

[3.3.2. Triển khai CMS 16](#_Toc32939922)

[3.3.3. Triển khai process 17](#_Toc32939923)

[3.3.4. Cài đặt các phần mềm khác: 17](#_Toc32939924)

[3.3.5. Logstash 19](#_Toc32939925)

[3.3.6. Elastic Search 22](#_Toc32939926)

[3.3.7. Hướng dẫn cài đặt Kibana 22](#_Toc32939927)

[4. BẢO MẬT, BACKUP VÀ SAO LƯU HỆ THỐNG 23](#_Toc32939928)

[***4.1.*** ***Bảo mật hệ thống và phòng chống Virus*** 23](#_Toc32939929)

[***4.2.*** ***Backup database và lập lịch sao lưu hệ thống*** 23](#_Toc32939930)

[5. CÁC LỖI CÀI ĐẶT VÀ CÁCH KHẮC PHỤC 24](#_Toc32939931)

[***5.1.*** ***Lỗi sau khi cấu hình port*** 24](#_Toc32939932)

[***5.2.*** ***Lỗi “OutOfMemoryError: PermGen space”*** 24](#_Toc32939933)

[6. PHỤ LỤC 24](#_Toc32939934)

# **GIỚI THIỆU**

## ***Mục đích, ý nghĩa của tài liệu***

Tài liệu này được xây dựng nhằm hướng dẫn:

* Giúp cán bộ triển khai hiểu rõ các yêu cầu về môi trường cài đặt hệ thống và thực hiện được chính xác, đầy đủ các thao tác cần thiết để thực hiện việc cài đặt hệ thống một cách nhanh nhất, hiệu quả nhất.
* Thống nhất được môi trường cài đặt (cách cấu hình các phần mềm hệ thống, ổ đĩa cài đặt, cách đặt tên thư mục…) để tạo thuận lợi cho việc vận hành hệ thống và công tác bảo trì hệ thống sau này.
* Cung cấp các mẫu biên bản và hướng dẫn cách thức kiểm tra kết quả cài đặt cũng như cách ghi nội dung các biên bản để xác nhận công việc triển khai
* Giúp quản trị hệ thống hiểu rõ các yêu cầu về môi trường cài đặt hệ thống và thực hiện được chính xác, đầy đủ các thao tác cần thiết để thực hiện việc cài đặt hệ thống một cách nhanh nhất, hiệu quả nhất.
* Đồng thời mô tả cách khắc phục một số sự cố gặp phải khi sử dụng và cài đặt hệ thống

Đối tượng sử dụng tài liệu bao gồm:

* Quản trị hệ thống: Sử dụng tài liệu này để cài đặt, quản lý cấu hình và duy trì hoạt động của hệ thống.
* Cán bộ phát triển: Sử dụng tài liệu này để hiểu hơn cơ chế hoạt động của hệ thống, cán bộ phát triển có thể tự dựng hệ thống ban đầu để tiêp tục nâng cấp phát triển hệ thống.

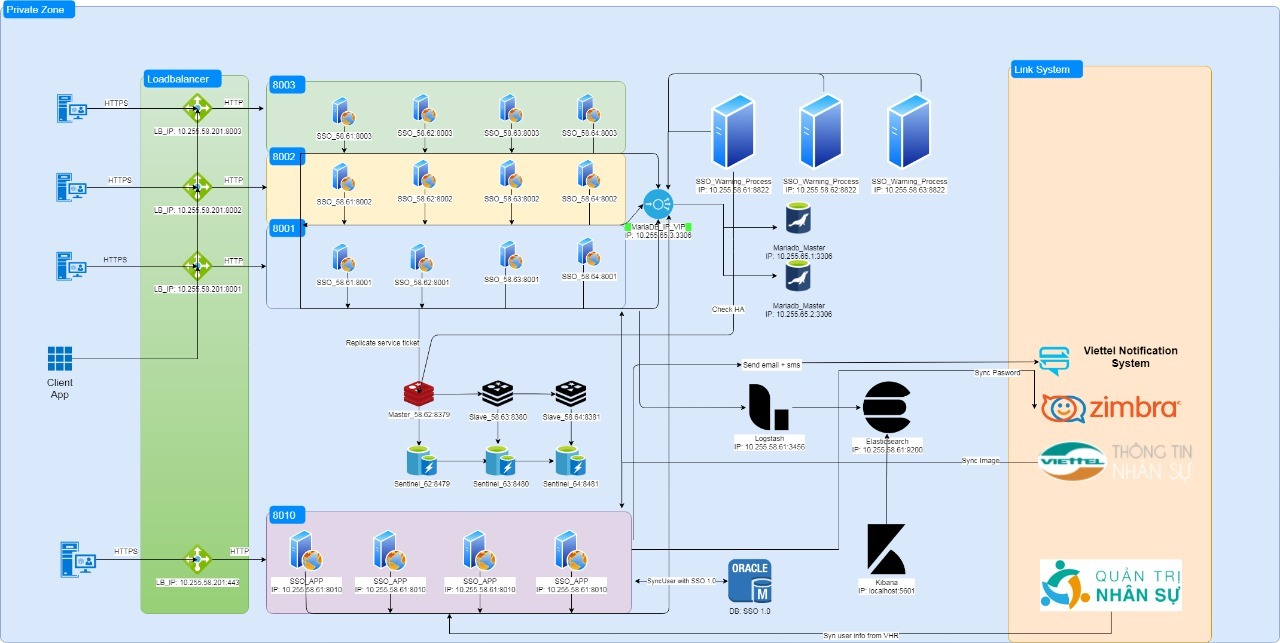
## ***Phạm vi tài liệu***

## ***Các thuật ngữ và từ viết tắt***

| **Thuật ngữ** | **Định nghĩa** | **Ghi chú** |
| --- | --- | --- |
| SSO | Hệ thống xác thực tập trung |  |
|  |  |  |
|  |  |  |
|  |  |  |

## ***Các thành phần của hệ thống***

Hệ thống bao gồm các thành phần như sau:



Thông tin chi tiết của các thành phần được mô tả chi tiết như sau:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| STT | Tên phần mềm | Loại | Phiên bản | Link source code/ link tham khảo | Vai trò | Đường dẫn tham số cấu hình | Chú thích |
| 1 | Load Balance | Open source |  |  | Sử dụng làm cân bằng tải của hệ thống, cân bằng các request vào hệ thống |  | LB do VTVET vận hành |
| 2 | WEB SSO | Tự code | 2.0 – 5.3.7 | Link source code các chức năng tự phát triển:  http://10.240.203.2:8180/it-dept/bu-core/sso  Link source code các chức năng sử dụng open source: https://github.com/apereo/cas.git | Sevlet Web bao gồm chức năng login của hệ thống | /u01/sso/sso\_app/  tomcat\_sso\_8001/  webapps/sso/WEB-INF/classes  /u01/sso/sso\_app/  tomcat\_sso\_8001/  webapps/sso/WEB-INF/classes  /u01/sso/sso\_app/  tomcat\_sso\_8001/  webapps/sso/WEB-INF/classes | Các chức năng tự phát triển được source code trên gitlab version 2.0, các chức năng sử dụng open source sử dụng code trên github version 5.3.7 |
| 3 | WEB\_ADMIN\_CMS | Tự code | 1.0 | http://10.240.203.2:8180/it-dept/bu-core/sso | Sevlet Web bao gồm các chức năng nghiệp vụ của hệ thống, bao gồm: quên mật khẩu, đổi mật khẩu, quản lý người dùng, … | /u01/sso/app/config |  |
| 4 | PROCESS | Tự code | 1.0 | http://10.240.203.2:8180/it-dept/bu-core/sso | Tiến trình gửi tin nhắn,email đến người dùng. | /u01/sso/process/SSOProcess/run/Warning/conf |  |
| 5 | MariaDB | Open source | 10.6 | https://mariadb.org/ | Cơ sở dữ liệu quan hệ của hệ thốn, lưu trữ các thông tin về người dùng, đơn vị, phòng ban, … |  |  |
| 6 | Redis | Open source | 5.3.1 | https://redis.io/ | Cơ sở dữ liệu NoSQL, lưu trữ thông tin về thông tin đăng nhập của người dùng | /u01/sso/environment/redis-5.0.5 |  |
| 7 | LOGSTASH | Open source | 7.10 | https://www.elastic.co/uk/logstash | Ghi lại log tác động của hệ thống và người dùng hệ thống | /u01/sso/environment/logstash-7.1.1/config |  |
| 8 | Elastic Search | Open source | 7.10 | https://www.elastic.co/uk/ | Cơ sở dữ liệu full-text, lưu lại toàn bộ tác động của người dùng và hệ thống. |  | VTNET vận hành |
| 9 | Kibana | Open source | 7.10 | https://www.elastic.co/uk/kibana | Data visualization dữ liệu trên elastic search phục vụ truy cứu dữ liệu, tạo báo cáo. |  | VTNET vận hành |

## ***Tài liệu hướng dẫn cài đặt bao gồm các phần sau:***

* + - 1. Giới thiệu
      2. Yêu cầu về môi trường cài đặt
      3. Quy trình cài đặt
      4. Bảo mật, backup và sao lưu hệ thống
      5. Các lỗi cài đặt, khắc phục
      6. Phụ lục

# **YÊU CẦU VỀ MÔI TRƯỜNG CÀI ĐẶT**

## ***Yêu cầu môi trường***

* *Số lượng máy*: Tối thiểu 04 máy chủ cung cấp các dịch vụ
* *Cấu hình tối thiểu*:

|  |
| --- |
| * Processor: E5-2650 v4 hoặc tương đương * Memory: 8x32 GB RAM DDR4   - Hard disk: 2x600GB 12G SAS 15K RPM |

* *Phần mềm cài đặt*

|  |
| --- |
| Hệ điều hành Linux (CentOS 7.0) |

## ***Yêu cầu về cơ sở dữ liệu***

* *Số lượng máy****:*** Tối thiểu 03 máy chủ cung cấp các dịch vụ
* *Cấu hình tối thiểu:*

|  |
| --- |
| * Processor: E5-2650 v4 hoặc tương đương * Memory: 8x32 GB RAM DDR4   - Hard disk: 2x600GB 12G SAS 15K RPM |

* *Phần mềm cài đặt:*

|  |
| --- |
| * Hệ điều hành Linux (CentOS 7.0) * Cài đặt máy chủ * Database Maria DB |

# **QUY TRÌNH CÀI ĐẶT**

Việc cài đặt hệ thống bao gồm cài đặt các thành phần sau:

* Cài đặt cơ sở dữ liệu
* Cài đặt máy chủ
* Cài đặt và kiểm tra dịch vụ

Phần phía sau sẽ đi sâu vào hướng dẫn cài đặt chi tiết cho từng thành phần của hệ thống

## ***Cài đặt cơ sở dữ liệu***

* Môi trường cài đặt trên môi trường Linux. ( CentOS 7.0 )
* Chi tiết cài đặt như sau:

- Bước 1: Cài đặt hệ quản trị CSDL MariaDB

MariaDB là một nhánh của MySQL (một trong những cơ sở dữ liệu phổ biển nhất thế giới), là máy chủ cơ sở dữ liệu kế thừa các chức năng cốt lõi của MySQL, ngoài ra MariaDB cung cấp thêm nhiều các tính năng cải tiến về cơ chế lưu trữ, tối ưu máy chủ.

Khác với MySQL, MariaDB là phần mềm mã nguồn mở, còn MySQL thì không

Thông tin port: 3306

Yêu cầu hệ thống:

* Server có cài hệ điều hành CentOS 7
* Tài khoản có thể đăng nhập vào hệ thống và có quyền sudo, hướng dẫn cài đặt ở dưới đây sử dụng tài khoản root

Hướng dẫn cài đặt MariaDB

* Bước 1: Tải gói cài đặt MariaDB 10.4.7
* Bước 2: Thêm kho lưu trữ YUM vào máy chủ CentOS 7.0

|  |
| --- |
| tee /etc/yum.repos.d/MariaDB.repo<<EOF  [mariadb]  ...  disable\_unix\_socket  name = MariaDB  baseurl = http://yum.mariadb.org/10.4/centos7-amd64  gpgkey = https: //yum.mariadb.org/RPM-GPG-KEY-MariaDB  gpgcheck = 1  EOF |

Trong đó:

|  |
| --- |
| name – tên của kho lưu trữ  baseurl – đường dẫn tới kho lưu trữ  gpg – là viết tắt của GNU Private Guard, là một loại chữ kí số giúp bảo vệ quyền riêng tư. Nếu đặt giá trị gpgcheck =1 thì nó sẽ xác thực gói bằng cách kiểm tra chữ ký GPG. Đặt gpgcheck=0 để bỏ qua xác thực hoặc trong trường hợp gói không được đăng ký.  gpgcheck – cài đặt bảo mật(1=có,0=không)  enable – trạng thái của kho lưu trữ(1=bật,0=tắt)  gpgkey – đường dẫn của file key(nếu có) |

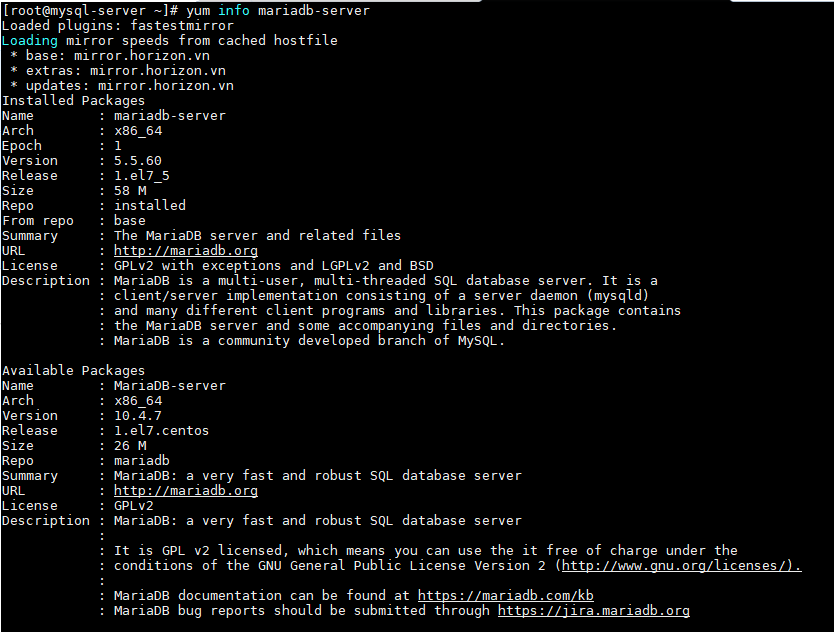
* Bước 3: Cập nhật bộ đệm:

|  |
| --- |
| yum makecache fast |

* Bước 4: Kiểm tra phiên bản MariaDB-Server từ repo trước khi cài đặt

|  |
| --- |
| yum info mariadb-server |

Kết quả



* Bước 5: Gỡ cài đặt phiên bản MariaDB trước và các gói liên quan:

|  |
| --- |
| yum erase mysql  rm -rf / var / lib / mysql  rm /etc/my.cnf |

* Bước 6: Cài đặt MariaDB 10.4.7

|  |
| --- |
| yum -y install MariaDB-server MariaDB-client |

* Bước 7: Khởi động dịch vụ MariaDB

|  |
| --- |
| systemctl start mariadb |

* Bước 8: Cài đặt để MariaDB khởi động cùng hệ điều hành

|  |
| --- |
| systemctl enable mariadb |

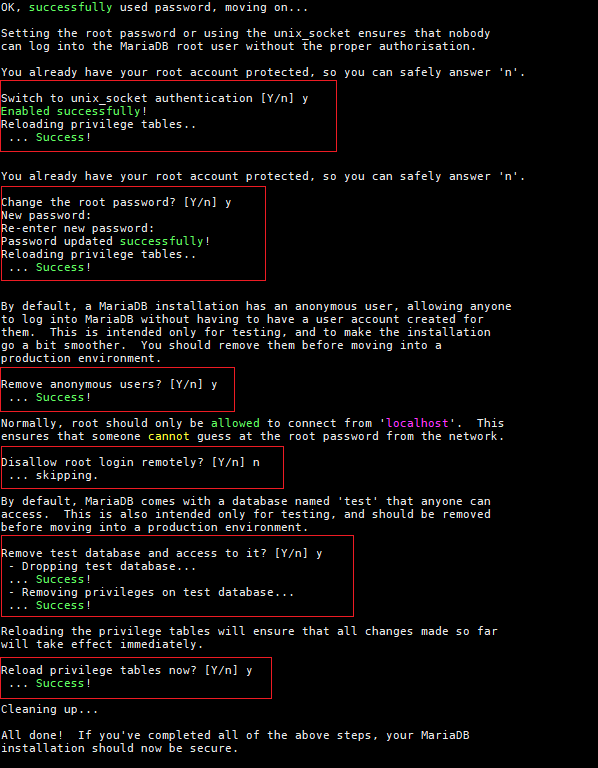
* Bước 9: Kiểm tra trạng thái dịch vụ

|  |
| --- |
| systemctl status mariadb |

* Bước 10: Thiết lập cấu hình bảo mật cơ bản cho dịch vụ MariaDB

Sau khi cài đặt, tài khoản root của MariaDB chưa có mật khẩu. Câu lệnh cài đặt mật khẩu và thiết lập thêm một số tùy chọn khác như sau:

|  |
| --- |
| mysql\_secure\_installation |



Trong đó:

|  |
| --- |
| Switch to unix\_socket authentication: xác thực bằng unix\_socket, chọn Y để bật, n để tắt.  Enter current password for root (enter for none): nếu đã từng đặt mật khẩu thì bạn nhập mật khẩu hiện tại đang sử dụng, nếu là lần đầu tiên thì bạn có thể để trống. Enter để tới bước tiếp theo.  Remove anonymous users: Xóa tài khoản vô danh, Bạn chon Y (Yes).  Disallow root login remotely: Tắt truy cập từ xa. Tùy bạn lựa chọn.  Remove test database and access to it: Xóa database có tên là “test”. Nó chỉ dùng để test lúc ban đầu, bạn có thể xóa nó. Chon Y (Yes)  Reload privilege tables now: reload lại các bảng liên quan đến quyền hạn. |

Tắt plugin unix\_socket trong file /etc/my.cnf.d/server.cnf

|  |
| --- |
| [mariadb]  ...  unix\_socket = OFF |

Hoặc

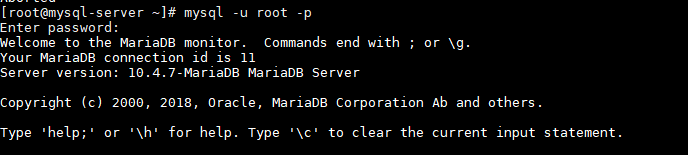
|  |
| --- |
| [mariadb]  ...  disable\_unix\_socket |

Khởi động lại dịch vụ MariaDB

|  |
| --- |
| systemctl restart mariadb |

Đăng nhập và kiểm tra lại phiên bản MariaDB

|  |
| --- |
| Mysql –u root -p |



Hoặc sử dụng

|  |
| --- |
| Mysql -v |

Kết quả:

|  |
| --- |
| Welcome to the MariaDB monitor. Commands end with; or \ g.  Your MariaDB connection id is 10  Server version: 10.4.7-MariaDB MariaDB Server |

- Bước 2: Tạo CSDL phục vụ cho hệ thống vận hành

* Bước 1: Tạo database : db\_sso
* Bước 2: Import dữ liệu DB backup vào schema vừa tạo

## ***Cài đặt máy chủ***

1. *Quy trình tổng quát*

|  |
| --- |
| **Bước 1** – Cài đặt JDK  **Bước 2** –Cài đặt Tomcat |

1. *Quy trình chi tiết*

### *3.2.1. Cài đặt JDK*

Tài khoản cài đặt **:** root

Mặc định CentOS đã được cài đặt trước OpenJDK. Trong trường hợp bạn muốn sử dụng Oracle JDK để chạy dịch vụ thì làm theo hướng dẫn sau.

* Bước 1: Gỡ bỏ phiên bản JDK cũ

|  |
| --- |
| **Tìm kiếm phiên bản JDK cũ**  # rpm -qa | grep -E '^open[jre|jdk]|j[re|dk]'  **Kết quả xuất ra:**  java-1.6.0-openjdk-1.6.0.0-1.66.1.13.0.el6.i686  eject-2.1.5-17.el6.i686  perl-Object-Accessor-0.34-136.el6.i686  java-1.7.0-openjdk-1.7.0.45-2.4.3.4.el6\_5.i686  openjpeg-libs-1.3-10.el6\_5.i686  **Java 1.6 và Java 1.7 đã được cài đặt trên hệ thống, chúng ta cần gở bỏ 2 phiên bản này.**  # rpm -e java-1.6.0-openjdk or yum remove java-1.6.0-openjdk  # rpm -e java-1.7.0-openjdk or yum remove java-1.7.0-openjdk |

* Bước 2: Cài đặt phiên bản JDK mới
  + Bạn vào địa chỉ <http://www.oracle.com/technetwork/java/javase/downloads/index.html> để download JDK bản mới nhất (Ít nhất từ 1.8 trở lên) để thực hiện cài đặt.
  + Thực hiện các lệnh sau để cài đặt

|  |
| --- |
| # rpm -ivh jdk-8uversion-linux- x86\_64.rpm  Ex: # rpm -ivh jdk-8u5-linux- x86\_64.rpm |

* + Tham khảo link sau để chi tiết hơn các cách cài đặt <https://docs.oracle.com/javase/8/docs/technotes/guides/install/linux_jdk.html>

### *3.2.2. Cài đặt Tomcat*

**-** Tài khoản cài đặt **:** os\_sso

+ Bạn vào trang <http://tomcat.apache.org/> download bản Tomcat mới về để cài đặt. Bạn download từ bản 8 trở lên và download bản không cần cài đặt, chỉ cần giải nén là chạy được.

+ Cấu hình ATTT cho Tomcat: file đính kèm: [HD thiet lap ATTT Webserver Tomcat 2017 - 2018](file:///D:\Users\tudd5.VIETTELGROUP.000\AppData\Roaming\Microsoft\Word\HD%20thiet%20lap%20ATTT%20Webserver%20Tomcat%202017%20-%202018.docx)

+ Cấu hình Tomcat:

* Bước 1 : Copy bản tomcat chuẩn đã fix ATTT vào thư mục **TOMCAT\_DIR**
* Bước 2: Cấp quyền thực thi cho các tệp trong thư mục **TOMCAT\_DIR**/tomcat\_8001/bin
* Bước 3: Cấu hình RAM
* Mở file server.xml tại đường dẫn CATALINA\_HOME/bin/catalina.sh
* Chỉnh sửa cấu hình Xms và Xmx cho phù hợp

|  |
| --- |
| JAVA\_OPTS**=**"-server -Djava.awt.headless=true -Xms128m -Xmx128m -XX:+CMSClassUnloadingEnabled -XX:SurvivorRatio=8 -Dsun.rmi.dgc.client.gcInterval=3600000 -Dsun.rmi.dgc.server.gcInterval=3600000 -XX:+UseParNewGC -XX:+CMSParallelRemarkEnabled -XX:+UseConcMarkSweepGC" |

* Trường hợp hệ thống lớn nhiều người dùng chú ý bổ sung thêm tham số PermSize và MaxPermSize

|  |
| --- |
| JAVA\_OPTS**=**"-server -Djava.awt.headless=true -Xms4G -Xmx8G -XX:PermSize=512M -XX:MaxPermSize=1G -XX:+CMSClassUnloadingEnabled -XX:SurvivorRatio=8 -Dsun.rmi.dgc.client.gcInterval=3600000 -Dsun.rmi.dgc.server.gcInterval=3600000 -XX:+UseParNewGC -XX:+CMSParallelRemarkEnabled -XX:+UseConcMarkSweepGC" |

* Bước 4: Cấu hình port HTTP
  + Mở file server.xml tại đường dẫn CATALINA\_HOME/conf/server.xml
  + Chỉnh sửa cấu hình cho phù hợp

|  |
| --- |
| <Connector port="8001"  protocol="HTTP/1.1"  connectionTimeout="20000"  URIEncoding="UTF-8"  useBodyEncodingForURI="true"  maxThreads="3000"  acceptCount="200"  minSpareThreads="500"  maxSpareThreads="2000"  maxHttpHeaderSize="8192"  disableUploadTimeout="true"  compression="on"  compressionMinSize="2048"  noCompressionUserAgents="gozilla, traviata"  compressableMimeType="text/html,text/xml,text/plain,text/css,text/javascript,application/json,application/x-javascript,application/javascript"  enableLookups="false"  redirectPort="8443"  /> |

* Bước 5: Cấu hình port HTTPS ( sử dụng cho Web Server, Mobile Server)
  + Mở file server.xml tại đường dẫn CATALINA\_HOME/conf/server.xml
  + Mở đoạn code chứa HTTPS và chỉnh sửa cấu hình cho phù hợp ( chú ý chỉnh redirectPort cua HTTP khớp với port cua HTTPS)

|  |
| --- |
| <Connector port="8443"  protocol="HTTP/1.1"  SSLEnabled="true"  sslEnabledProtocols="TLSv1, TLSv1.1, TLSv1.2"  scheme="https" secure="true"  clientAuth="false" sslProtocol="SSL"  keystoreFile="/u01/PRS\_release/apache-tomcat-7/apache-tomcat-7-8175/conf/keystore "  keystorePass="changeit"  ciphers="TLS\_ECDHE\_RSA\_WITH\_AES\_128\_CBC\_SHA256, TLS\_ECDHE\_RSA\_WITH\_AES\_128\_CBC\_SHA, TLS\_ECDHE\_RSA\_WITH\_AES\_256\_CBC\_SHA384, TLS\_ECDHE\_RSA\_WITH\_AES\_256\_CBC\_SHA, TLS\_ECDHE\_RSA\_WITH\_RC4\_128\_SHA, TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA256, TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA, TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA256, TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA, SSL\_RSA\_WITH\_RC4\_128\_SHA"  URIEncoding="UTF-8"  useBodyEncodingForURI="true"  connectionTimeout="20000"  maxThreads="3000"  acceptCount="200"  minSpareThreads="500"  maxSpareThreads="2000"  maxHttpHeaderSize="8192"  disableUploadTimeout="true"  compression="on"  compressionMinSize="2048"  noCompressionUserAgents="gozilla, traviata"  compressableMimeType="text/html,text/xml,text/plain,text/css,text/javascript,application/json,application/x-javascript,application/javascript"  enableLookups="false"  redirectPort="8443"  /> |

* Bước 6 : Cấu hình ConnectTimeOut:
  + ConnectionTimeout là thời gian chờ để thực hiện một yêu cầu tối đa . Mặc định default là 20s = 20\*1000ms nó sẽ báo lỗi nếu không đáp ứng được yêu cầu từ client.
  + Cấu hình phù hợp cho các ứng dụng
* WEB,CORE,PASSPORT,VSA : 20000 ( 20s) ( Giúp giảm tải RAM của ứng dụng web)

MOBILE\_SERVER : 600000ms ( 10m)

* Bước 7: Cấu hình share\_session giữa các TomCat (Sử dụng cho Mobile Server và Web):
* Mở đoạn code chứa share\_session và chỉnh sửa cấu hình cho phù hợp

|  |
| --- |
| <?xml version='1.0' encoding='utf-8'?>  <Context>  <!-- Chu y : Su dung share session => source code phia duoi  1. memcachedNodes : IP memcached  Ex : memcachedNodes="n1:10.30.174.211:11211,n2:10.30.174.212:11211"  -->  <Manager className="de.javakaffee.web.msm.MemcachedBackupSessionManager"  memcachedNodes="n1:10.65.65.45:11211,n2:10.65.65.46:11211"  sticky="false"  operationTimeout="10000"  sessionBackupAsync="false"  sessionBackupTimeout="10000"  lockingMode="uriPattern:/path1|/path2"  requestUriIgnorePattern=".\*\.(ico|png|gif|jpg|css|js)$"  transcoderFactoryClass="de.javakaffee.web.msm.serializer.kryo.KryoTranscoderFactory"  />  <!-- Default set of monitored resources -->  <WatchedResource>WEB-INF/web.xml</WatchedResource>  </Context> |

* Bước 8: Cấu hình JMX ( kiểm tra vùng heap hệ thống)
* Mở file setenv.sh tại đường dẫn CATALINA\_HOME/bin/ setenv.sh
* Mở đoạn code chứa cấu hình JMX và chỉnh sửa cho phù hợp.
* 18080 : ( tiền tố 1 + Port Tomcat)
* 28080 : ( tiền tố 2 + Port Tomcat)

|  |
| --- |
| CATALINA\_OPTS**=**"-Dcom.sun.management.jmxremote -Dcom.sun.management.jmxremote.rmi.port=18080 -Dcom.sun.management.jmxremote.port=28080 -Dcom.sun.management.jmxremote.ssl=false -Dcom.sun.management.jmxremote.authenticate=false" |

## ***Cài đặt và kiểm tra dịch vụ***

1. *Quy trình tổng quát*

|  |
| --- |
| Bước 1 - Triển khai dịch vụ SSO lên server  Bước 2 - Cấu hình hệ thống SSO  Bước 3 – Bật/tắt hệ thống SSO |

1. *Quy trình cài đặt chi tiết*

### *3.3.1. Triển khai webapp*

* **Bước 1**: Kiểm tra các kết nối đến database, kết nối tới các hệ thống khác (notification platform, thông tin nhân sự, …) - check các kết nối theo phục lục xxx

#telnet DB\_IP DB\_PORT

telnet 10.255.63.3 3306

* **Bước 2**: Triển khai sso lên server: Để triển khai SSO lên server, thực hiện các bước sau:
* Copy tệp “tomcat\_8001” lên server vào đường dẫn /u01/sso/sso\_app. Giải nén file tới thư mục này gọi là **BUIL\_DIR.**
* Sau khi đẩy các tệp cần thiết lên server, thực hiện các bước sau để cấu hình hệ thống
* Cấu hình module WEB: Trong thư mục ứng dụng web BUILD\_DIR, sửa lại các file sau cho phù hợp cấu hình của hệ thống:

+ File application.properties lưu trữ các cấu hình của hệ thống

+ File log4j2.xml lưu trữ cấu hình ghi log

* + Bật/tắt hệ thống tomcat: Sau khi đã hoàn thành các bước trên, cd tới thư mục BUILD\_DIR/tomcat\_8001/bin và sử dụng các lệnh sau:

+ Bật : ./startup.sh

+ Tắt: ./shutdown.sh

* + Kiểm tra hệ thống có đang chạy

|  |
| --- |
| Ps –ef|grep 8001 |

Được kết quả ví dụ như sau:

|  |
| --- |
| pmns\_app 37277 1 0 Nov07 ? 00:22:36 /u01/pmns\_app/jdk1.8.0\_101/bin/java -Djava.util.logging.config.file=/u01/pmns\_app/vEAS/tomcat\_eas\_8050/conf/logging.properties -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -Xms300m -Xmx6246m -XX:NewSize=512m -XX:MaxNewSize=512m -XX:PermSize=1024m -XX:MaxPermSize=1504m -Xms650M -Xmx1520M -Djava.endorsed.dirs=/u  01/pmns\_app/vEAS/tomcat\_eas\_8050/endorsed -classpath /u01/pmns\_app/vEAS/tomcat\_eas\_8050/bin/bootstrap.jar:/u01/pmns\_app/vEAS/tomcat\_eas\_8050/bin/tomcat-juli.jar -Dcatalina.base=/u01/pmns\_app/  vEAS/tomcat\_eas\_8050 -Dcatalina.home=/u01/pmns\_app/vEAS/tomcat\_eas\_8050 -Djava.io.tmpdir=/u01/pmns\_app/vEAS/tomcat\_eas\_8050/temp org.apache.catalina.startup.Bootstrap start  pmns\_app 119470 119153 0 15:36 pts/0 00:00:00 grep --color=auto 8050 |

* Gõ lệnh kill -9 37277 hoặc lệnh ./shutdown.sh để tắt ứng dụng

|  |
| --- |
| Kill -9 37277  Hoặc  ./shutdown.sh |

* Gõ lệnh ./startup.sh để triển khai ứng dụng

|  |
| --- |
| ./startup.sh |

### *3.3.2. Triển khai CMS*

* Bước 1: Copy toàn bộ tệp webapp.zip của ứng dụng và đẩy lên server /u01/sso/app. Giải nén file zip sẽ được thư mục webapp, đường dẫn tới thư mục này gọi là BUILD\_APP.
* Bước 2: Cấu hình hệ thống CMS: Trong thư mục config BUILD\_APP, sửa lại file cấu hình sau cho phù hợp với cấu hình của hệ thống triển khai
* File “*application.yml*”: cấu hình toàn bộ các cấu hình quan trọng của hệ thống
* Bước 3: Bật/tắt ứng dụng:
  + Sau khi hoàn thành các bước trên, cd tới thư mục BUILD\_APP
  + Bật ứng dụng: ./start.sh
  + Tắt ứng dụng: ./stop.sh
  + Kiểm tra hệ thống có đang chạy:

|  |
| --- |
| Ps –ef|grep 8010 |

Kết quả trả về như sau (ví dụ):

|  |
| --- |
| erp 24543 1 0 Jan20 ? 00:45:50 java -Dfile.encoding=UTF-8 -Dspring.config.location=/erp/erp/cas-client/config/ -jar /erp/erp/cas-client/webapp/  WEB-INF/pro-vt-sso.jar |

Gõ lệnh kill -9 24543 hoặc ./stop.sh để tắt ứng dụng

|  |
| --- |
| Kill -9 24543  ./stop.sh |

Gõ lệnh ./start.sh để khởi động ứng dụng

|  |
| --- |
| ./start.sh |

### *Triển khai process*

* Bước 1: Đẩy ứng dụng lên server u01/process
* Bước 2: Cấu hình:
  + SendWarningProcess.cfg: cấu hình gửi sms
  + SyncIbmDataprocess.conf: cấu hình kết nối tới IBM
* Bước 3: Bật/tắt tiến trình
  + Vào thư mục u01/process/SSOProcess/run/Warning/bin, gõ lệnh ./startup.sh để bật tiến trình
  + Vào thư mục u01/process/SSOProcess/run/Warning/bin, gõ lệnh ./shutdown.sh để tắt tiến trình.
* Bước 4: Kiểm tra hệ thống có chạy không bằng cách gõ lệnh ps –ef|grep 8822

### *Redis*

Download Redis bản 5.0.3 tại địa chỉ:  <http://download.redis.io/redis-stable.tar.gz>

Thực hiện cài đặt Redis 5.0.3 bằng cách như sau:

wget <http://download.redis.io/releases/redis-5.0.3.tar.gz>

tar xzf redis-5.0.3.tar.gz

cd redis-5.0.3

make

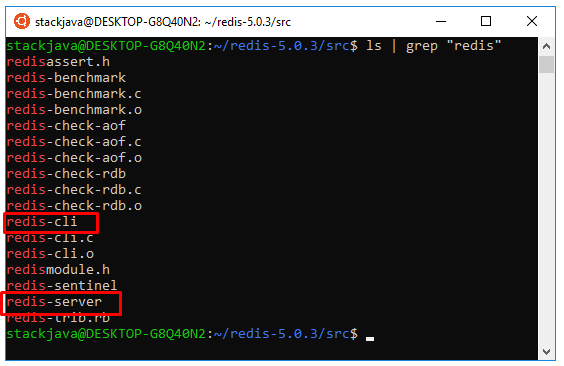
Trong đó: Lệnh thứ nhất để download file cài đặt redis-server

Lệnh thứ hai để giải nén

Lệnh thứ 3 để di chuyển vào thư mục vừa giải nén

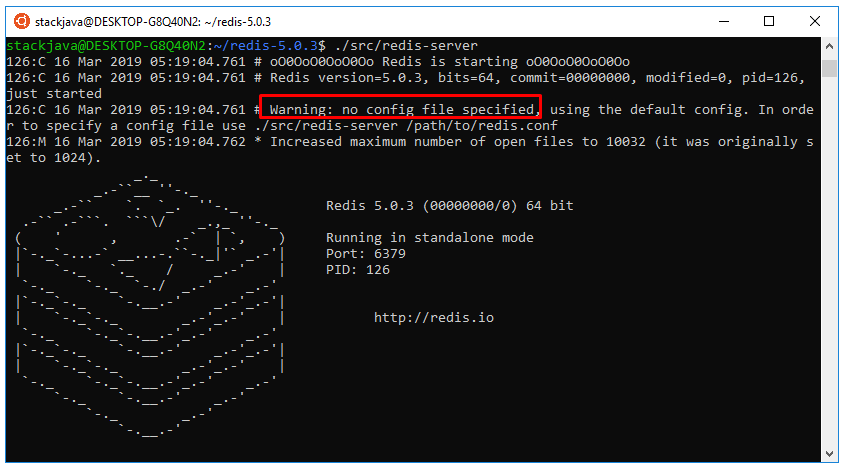
Lệnh thứ 4 thực hiện complite các file lại với nhau

Sau khi thực hiện xong các bước trên, trong folder src sẽ xuất hiện các file binary redis-server và redis-cli



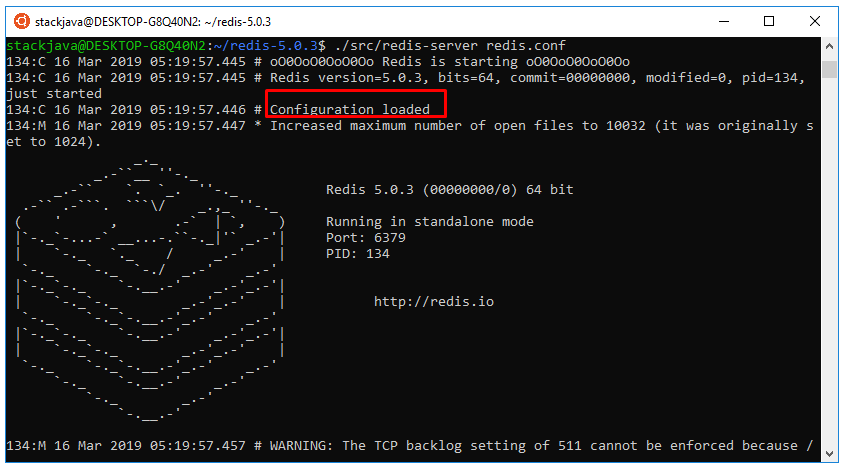
Để start redis server, chỉ cần chạy file redis-server.

Ở đây, đang ở trong folder redis-5.0.3, để chạy file redis-server trong folder src, sử dụng lệnh: *./src/redis-server*



Mặc định redis sẽ không có file cấu hình redis.conf nào được sử dụng. Để sử dụng cấu hình redis, khi khởi động, cần chỉ rõ xem file config ở đâu.

Ở đây dùng file redis.conf trong thư mục redis 5.0.3, vì vậy, chạy redis như sau: ./src/redis-server redis.conf



Như vậy là đã cài đặt thành công redis cho hệ thống.

Các cấu hình quan của redis được cấu hình như sau:

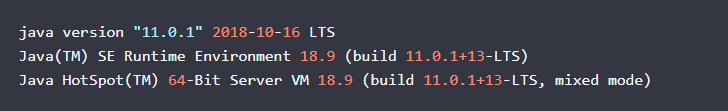
### *Logstash*

Logstash yêu cầu Java 8 hoặc Java 11. Vì vậy, để cài đặt logstash, cần kiểm tra version hiện tại của Java.

Để kiểm tra version hiện tại của Java, chạy lệnh command line:

Java –version

Nếu Java đã được cài đặt trong hệ thống, đoạn command line phía trên sẽ có output tương tự như sau:



Download phiên bản 7.1.1 của Logstash ở địa chỉ [https://www.elastic.co/downloads](https://www.elastic.co/downloads/logstash). Download file cài đặt phù hợp với môi trường – TAR.GZ, DEB, ZIP hoặc RPM.

Giải né file. Lưu ý không giải nén Logstash dưới đường dẫn tuyệt đối có chứa kí tự “:”.

Sau khi giải nén, có thể thực hiện chạy logstash bằng cách

cd bin

./logstash

Như vậy, logstash đã được khởi động, tuy nhiên, logstash chưa được cấu hình. Tạo file logstash-sso.conf trong thư mục config. Chi tiết như sau:

|  |  |  |
| --- | --- | --- |
| STT | Cấu hình | Tham số |
| 1 | input | udp {  port => 3456  } |
| 2 | output | if [@metadata][typeLog] == "History" {  stdout { codec => rubydebug }  elasticsearch {  hosts => "http://10.58.71.137:8200/"  index=> "userhistory"  }    }  if [@metadata][typeLog] == "NormalLog" {  stdout { codec => rubydebug }  elasticsearch {  hosts => "http://10.58.71.137:8200/"  index=> "logstash-%{+yyyy.MM.dd}"  }  } |
| 3 | filter | grok {  match => { "message" => "%{GREEDYDATA:actionName};%{GREEDYDATA:applicationCode};%{GREEDYDATA:duration};%{GREEDYDATA:service};%{GREEDYDATA:sessionId};%{GREEDYDATA:principal};%{GREEDYDATA:startTime};%{GREEDYDATA:transactionStatus};%{GREEDYDATA:errorCode};%{GREEDYDATA:errorDescription};%{GREEDYDATA:endTime} ;%{URI:originalUrl},%{IP:clientIp},%{WORD:devicefingerprint},%{GREEDYDATA:username},%{NUMBER:longitude},%{NUMBER:latitude},%{WORD:browser},%{WORD:deviceType},%{NUMBER:timestamp},%{WORD:devicename}"}  }  if "\_grokparsefailure" not in [tags] {  mutate {  add\_field => { "[@metadata][typeLog]" => "History" }  remove\_field => [ "message","@version","host","port"]  }  }  if "\_grokparsefailure" in [tags] {  grok {  match => { "message" => "%{GREEDYDATA:actionName};%{GREEDYDATA:applicationCode};%{GREEDYDATA:duration};%{GREEDYDATA:service};%{GREEDYDATA:sessionId};%{GREEDYDATA:principal};%{GREEDYDATA:startTime};%{GREEDYDATA:transactionStatus};%{GREEDYDATA:errorCode};%{GREEDYDATA:errorDescription};%{GREEDYDATA:endTime} ;%{GREEDYDATA:sso\_message}"}  }  mutate {  remove\_field => [ "message"]  add\_field => { "[@metadata][typeLog]" => "NormalLog" }  }  } |

Sau đó, để chạy logstash cùng cấu hình, thực hiện chạy lệnh như sau:

cd bin

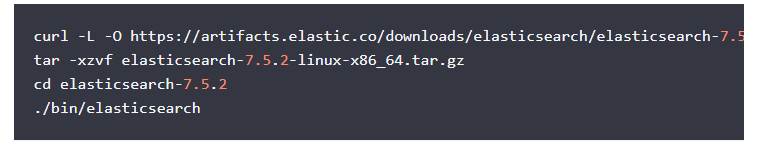
./logstash –f /config/logstash-sso.conf

Như vậy, logstash đã được khởi động thành công.

* + - Chỗ này có cách nào kiểm tra thế nào là đã chạy ko?

### *Elastic Search*

Elastic Search là một công cụ lưu trữ, tìm kiếm, và phân tích dữ liệu phân tán và real-time. Để download và cài đặt Elastic Search, sử dụng command line như ở dưới đây:



Để kiểm tra xem Elastic Search đã được khởi động và chạy thành công chưa, gửi một request HTTP GET vào port 9200.

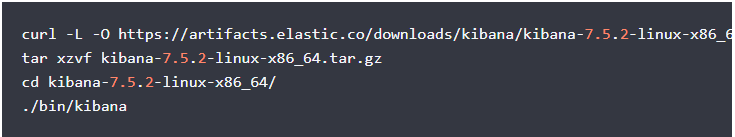


Nếu elastic search đã được cài đặt thành công, dữ liệu trả về sẽ tương tự như sau:



### *Kibana*

Để download và cài đặt Kibana, sử dụng command line ở dưới đây:



Tuy nhiên, để Kibana đọc được dữ liệu trong Elastic Search, cần phải cấu hình trong file config của Kiabana như sau:

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Cấu hình | Giá trị | Tên file |
| 1 | Elasticsearch.hosts | <http://10.58.71.137:9200> | /config/kibana.yml |

Sau đó khởi động lại Kiabana để sử dụng dữ liệu trong Elastic Search.

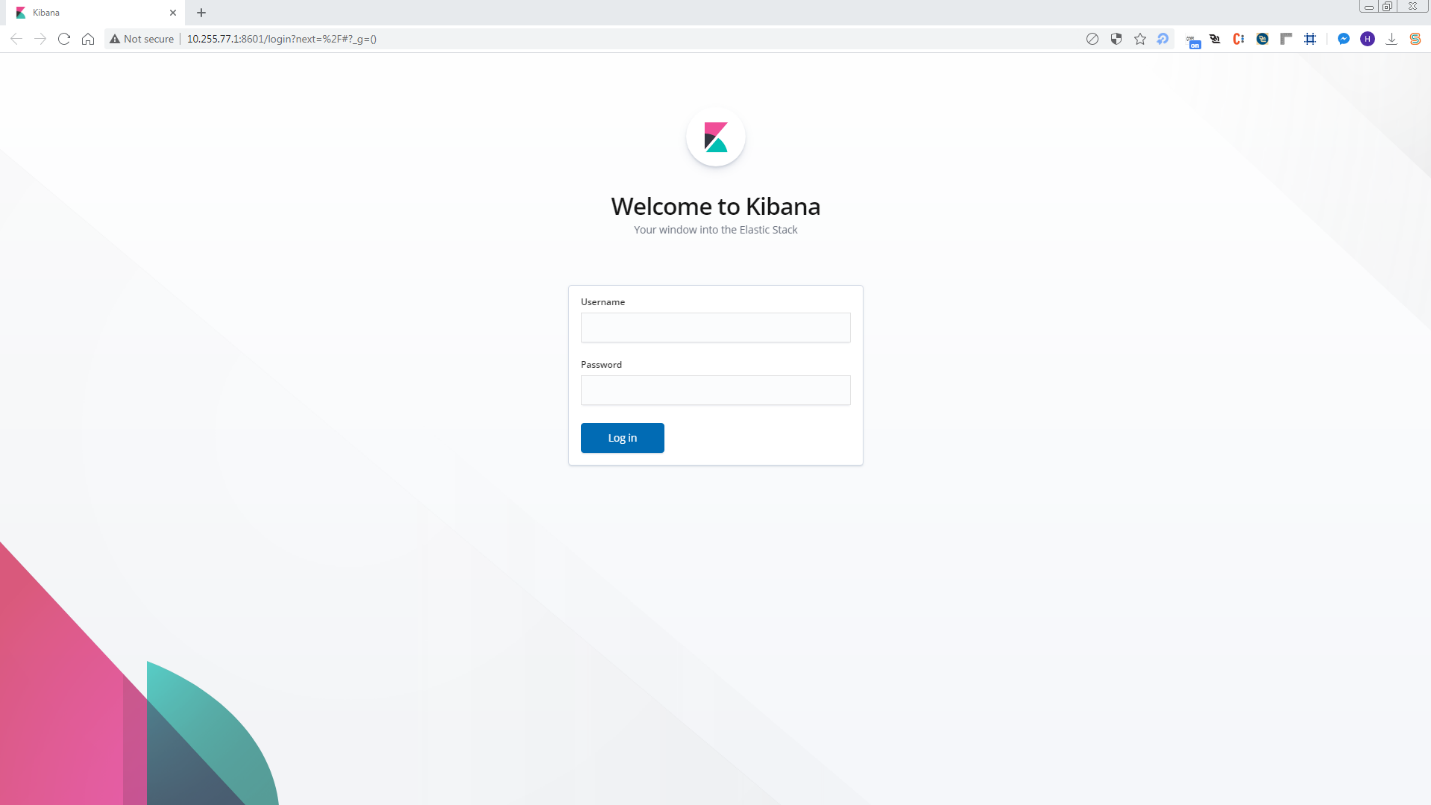
### *Hướng dẫn tra cứu lỗi trên Kibana*

Kibana của hệ thống SSO bao gồm 2 thành phần:

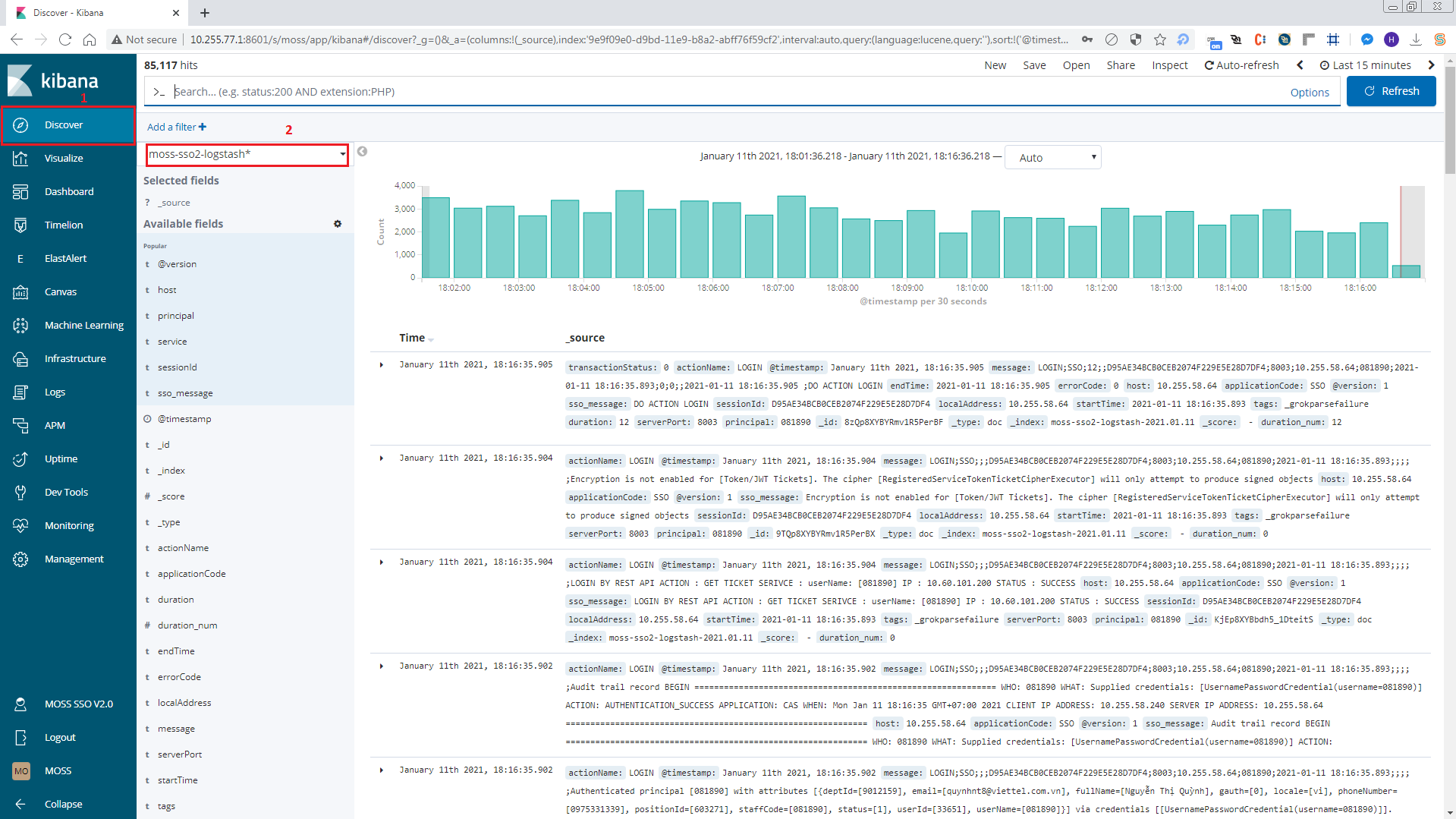
* Công cụ tra cứu lỗi
* Dashboard hiển thị thông tin của hệ thống.

Để tra cứu lỗi trên hệ thống, thực hiện lần lượt các bước sau:

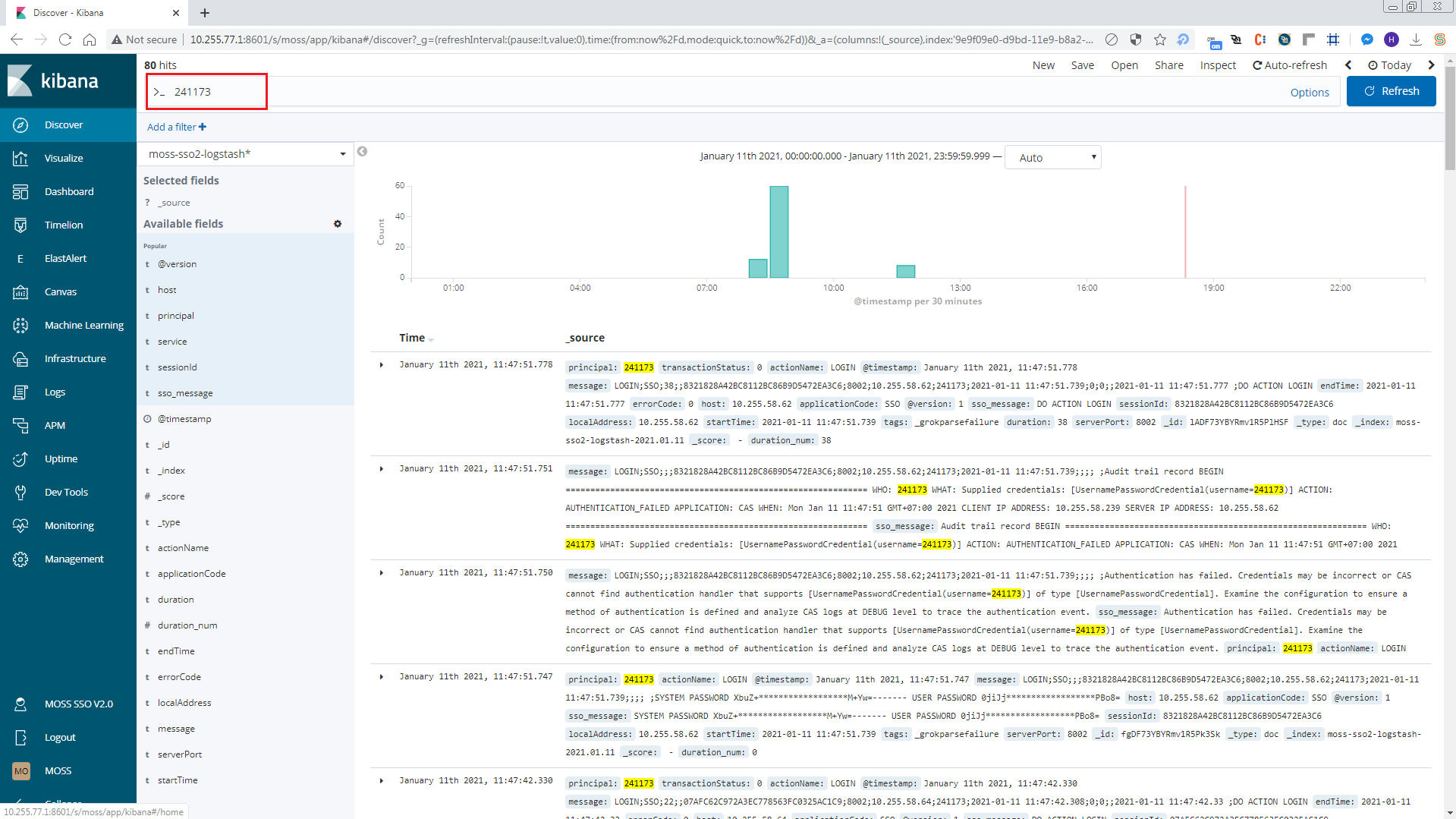
1. Truy cập vào địa chỉ Kibana, nhập username và password



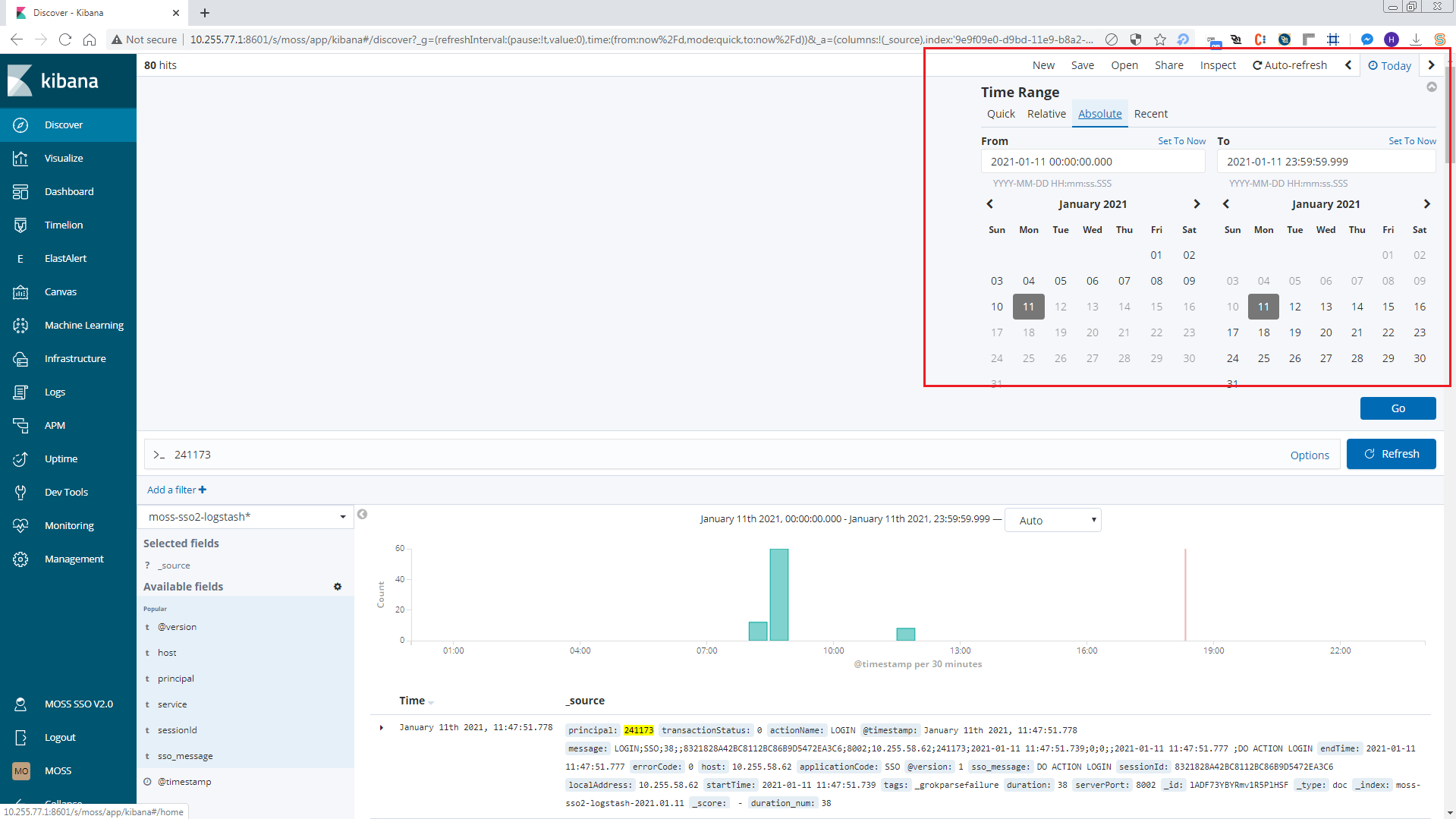
1. Chọn tab Discover, chọn index moss-sso2.0-logstash-\*



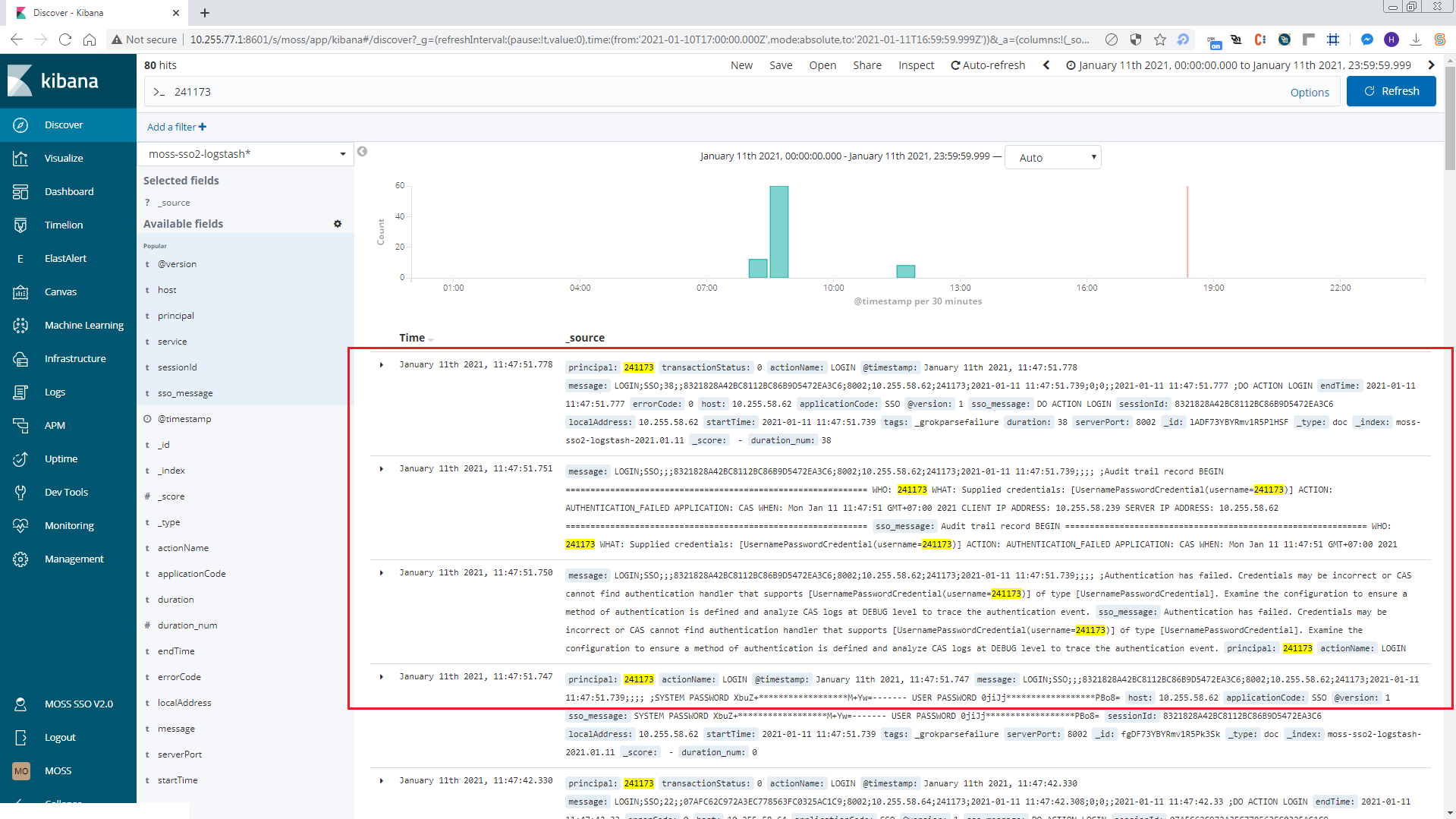
1. Nhập từ khóa cần tra cứu thông tin, từ khóa có thể là username, email, tên server, log lỗi,…



1. Chọn thời gian tra cứu

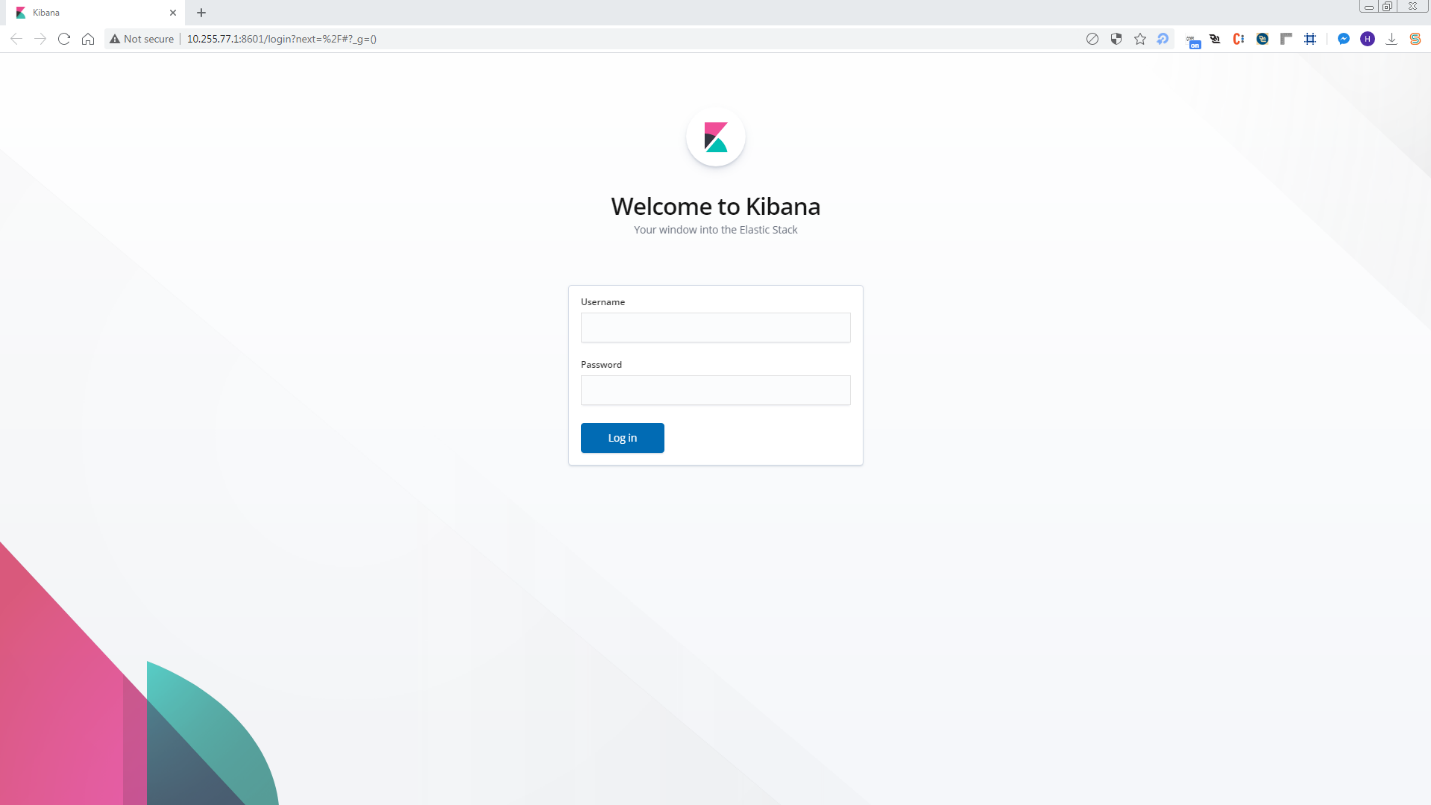


1. Thông tin logs của hệ thống sẽ hiển thị như sau

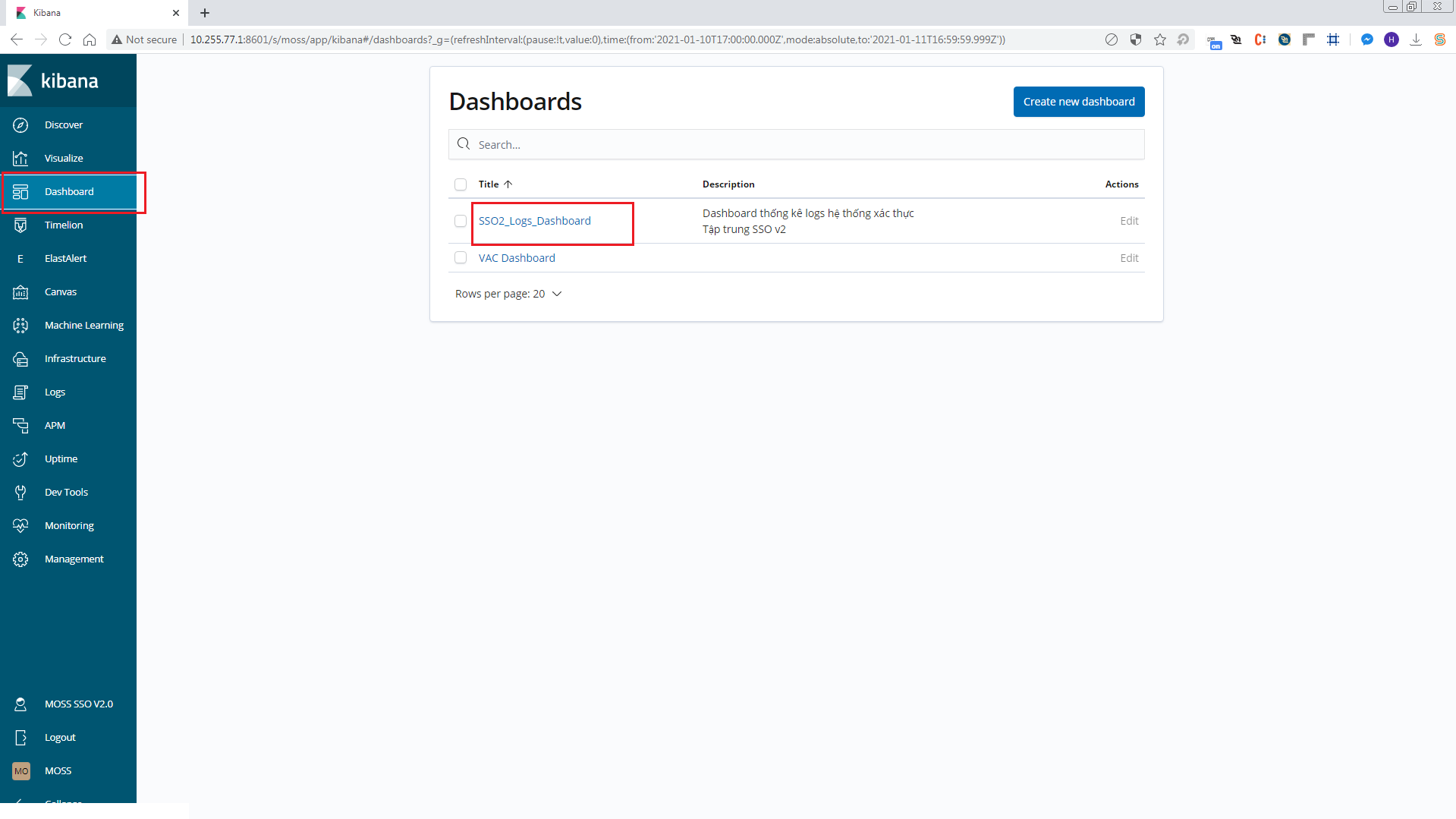


Dashboard của hệ thống bao gồm các dashboard nhỏ, mỗi dashboard chứa các thông tin khác nhau, cụ thể các bước để view dashboard của hệ thống như sau:

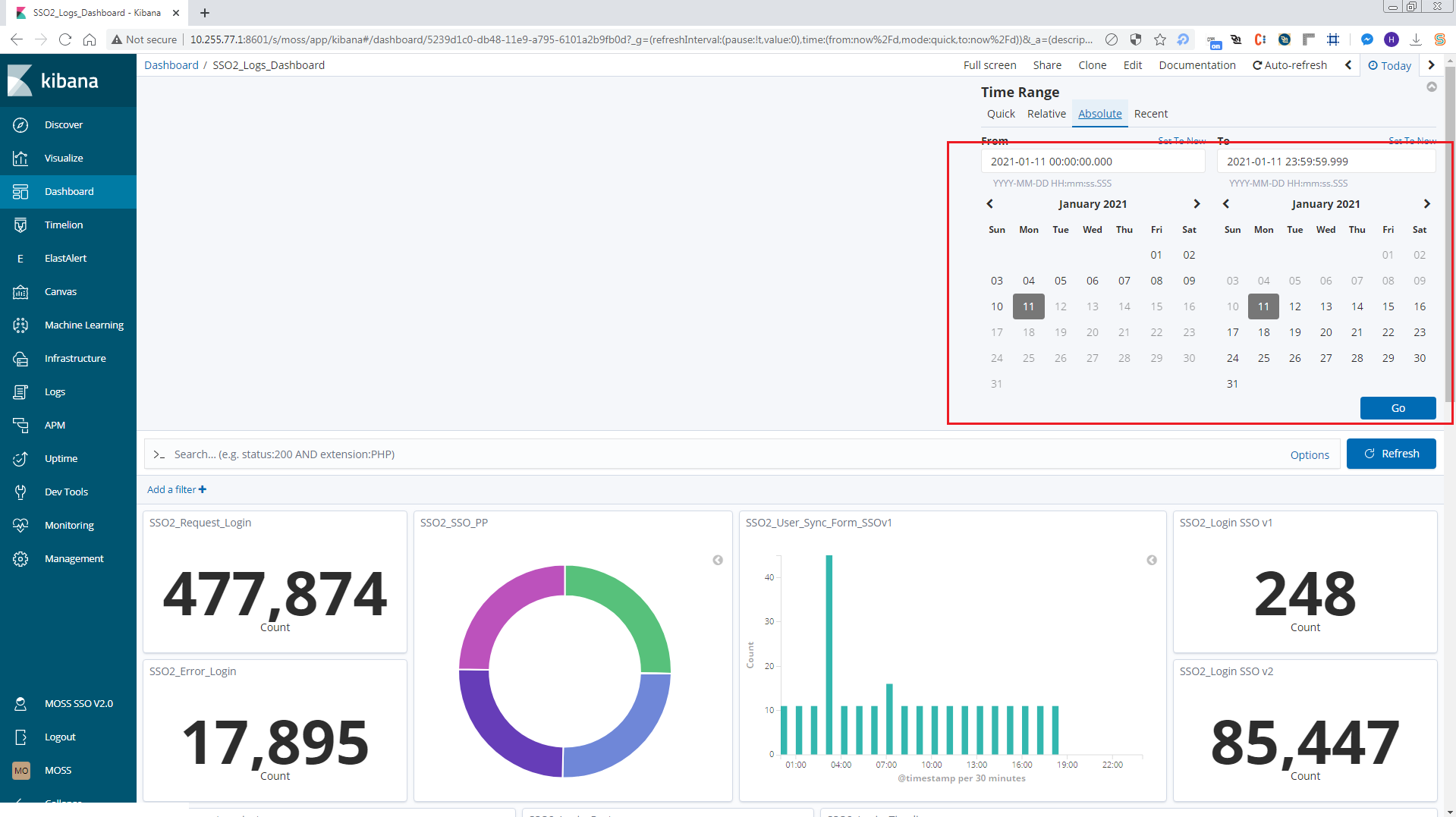
1. Đăng nhập vào hệ thống, nhập thông tin username và password



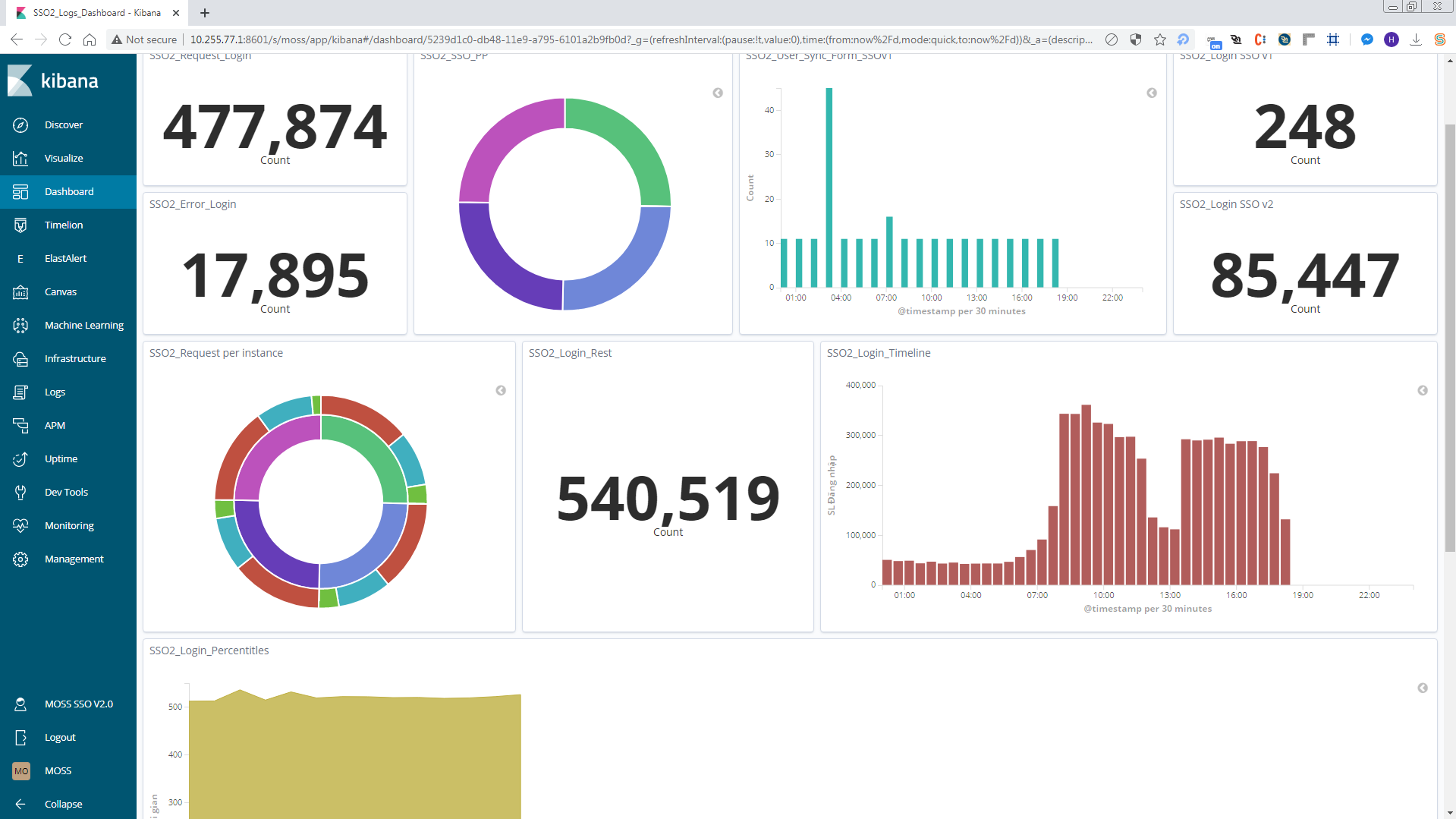
1. Chọn tab Dashboard, chọn dashboard SSO2\_Logs\_Dashboard



1. Chọn thời gian hiển thị dashboard



1. Dashboard của hệ thống hiển thị lên như sau:



# **BẢO MẬT, BACKUP VÀ SAO LƯU HỆ THỐNG**

## ***Bảo mật hệ thống và phòng chống Virus***

Để đảm tính bảo mật và an toàn cho hệ thống thì cần*:*

* Đảm bảo an toàn cho Database Server: Đối với khách hàng PRS.LITE thì CSDL về bán hàng là rất quan trọng, với giá trị rất lớn. Vì thế việc bảo mật đảm bảo an toàn cho Database Server là không thể bỏ qua.
* Đảm bảo an toàn và chống các truy cập không được phép trên Server: Server là nơi cung cấp toàn bộ dịch vụ cho nhân viên bán hàng hàng ngày, vì vậy khi Server có vấn đề toàn bộ dịch vụ sẽ bị tê liệt. Vì thế cần đảm bảo an toàn cho server bằng các cách sau:
  + Cấu hình server của hệ thống để đảm bảo ngăn chặn các truy cập không được phép.
  + Cài đặt các phần mềm AntiVirus và thường xuyên cập nhật. Cần cấu hình để đảm bảo chương trình AntiVirus này không quét và kiểm tra các dịch vụ mà hệ thống cung cấp.
  + Mã hóa toàn bộ thông tin truy cập CSDL từ ứng dụng (username, password, jdbc url ).

## ***Backup database và lập lịch sao lưu hệ thống***

Backup Database

Các database cần backup:

* SSO: database lưu trữ dữ liệu tài khoản của toàn bộ nhân viên.

Lập lịch sao lưu hệ thống:

* Định kỳ backup\_full 3 ngày gần nhất
* Cách backup database mariadb

# **CÁC LỖI CÀI ĐẶT VÀ CÁCH KHẮC PHỤC**

## ***Lỗi sau khi cấu hình port***

- Mô tả lỗi: Cấu hình port trên Apache Tomcat sau đó chạy ứng dụng vẫn bị lỗi. Vẫn vào được trang nhưng không hiển thị được các màn hình.

- Nguyên nhân: Thông thường lỗi này do mới cấu hình port trên Tomcat mà chưa cấu hình phía ứng dụng.

- Cách sửa: Làm theo đúng các bước như trong phần Cài đặt Tomcat: cấu hình port

## ***Lỗi “OutOfMemoryError: PermGen space”***

**-** Cách sửa: Cần tăng không gian nhớ của vùng **Heap**. Làm theo đúng các bước như trong phần [Cài đặt Tomcat : Cấu hình RAM](#_Bước_9_-).

# **PHỤ LỤC**

Một số lệnh thao tác trên hệ điều hành linux

* Kiểm tra thông tin bộ nhớ
  + cat /proc/meminfo
  + top
  + ps
* Kiểm tra CPU:
  + dmesg,
  + cat /proc/cpuinfo
* Kiểm tra HDD:
  + df -ah
  + df -k|sed 1d|awk '{total += $2} \ END {print total/1000}'
  + df -k|sed 1d| awk '{total += $4} \END {print total/1000}'
* Kiểm tra phiên bản jre:
  + java –version
* Kiểm tra Phiên bản và loại hệ điều hành
  + uname -a