**北京科技大学实验报告**

学院：计算机与通信工程学院 专业： 信息安全 班级： 信安192

姓名： 霍炟 学号：41824147 实验日期： 2022年 4月 20日

**实验名称：配置Kali Linux**

## 实验目的

实验内容：

准备内核头文件 自动评测

安装Nessus 自动评测

配置ProxyChains 自动评测

目录加密(TrueCrypt 已经不安全，TC 项目已经停止开发。)

本次实验会向你介绍Kali的定制，便于你更好地利用它。我们会涉及到后面章节所需的额外工具。我们会以ProxyChains的安装和数字信息的加密来结束这一次实验。

## 实验仪器

希冀平台

Linux kali 5.4.0-kali3-amd54

Nessus-8.11.0-debian6\_amd64.deb

## 实验原理

上一次实验介绍了Kali Linux的概念及安装。本章将介绍Kali的配置，以便于用户能充分利用它。本次实验主要介绍如何配置内核头文件、配置额外安全工具和设置ProxyChains等。本实验主要知识点如下：

准备内核头文件；

应用更新并配置额外的安全工具；

设置ProxyChains；

目录加密。

## 实验内容与步骤

打开桌面Virtual Machine，选择kali-linux虚拟机进入实验环境。

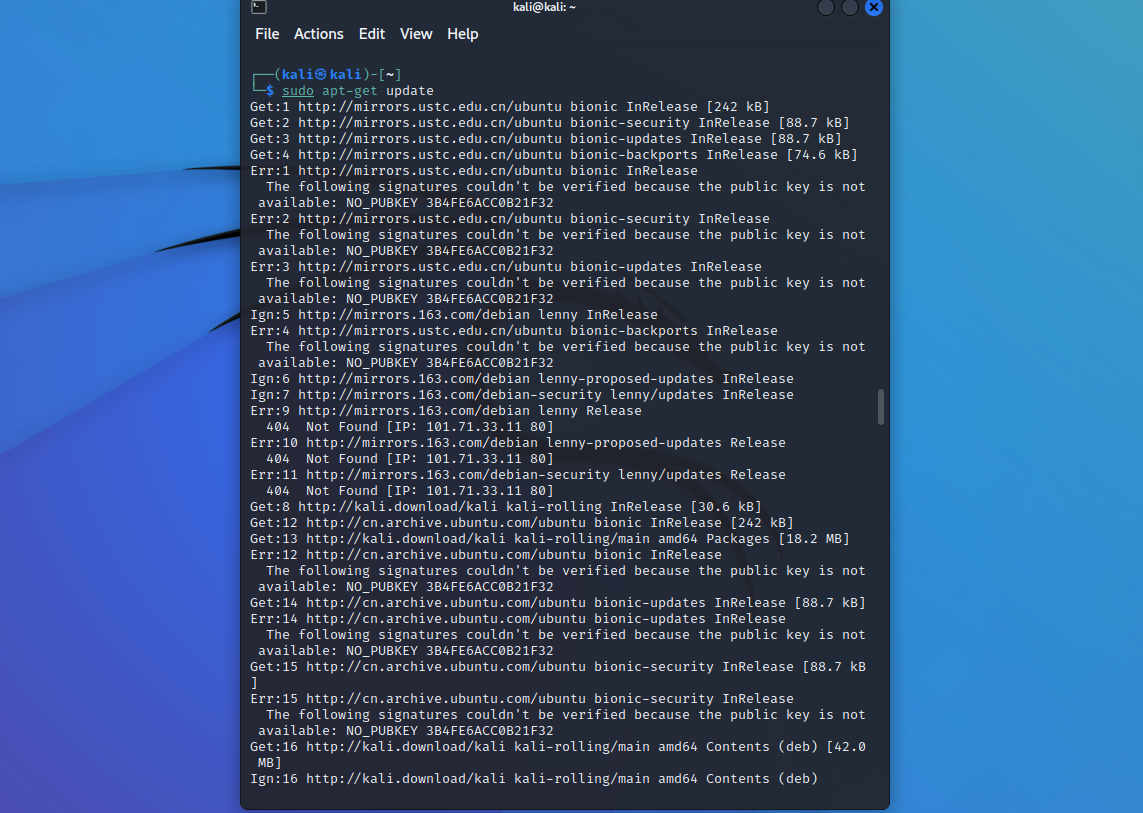
内核头文件是Linux内核的源代码。有时候，用户需要编译内核头文件代码，为以后使用内核头文件做准备，本节将介绍编译内核头文件的详细步骤。

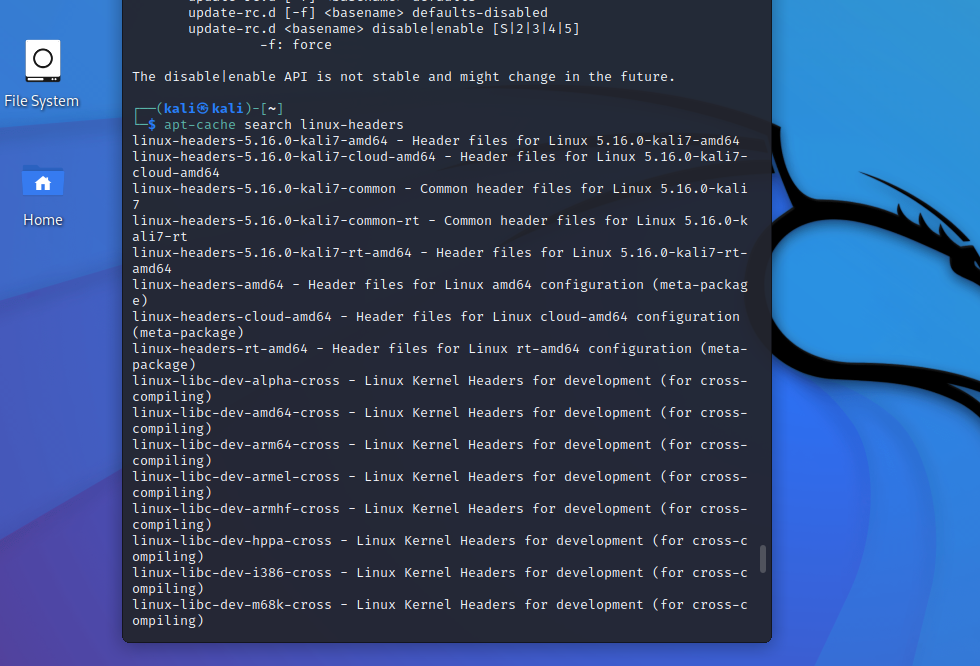
准备内核头文件的具体操作步骤如下所示。

1）更新软件包列表。执行命令如下所示：

通过以下命令列出可用的linux-headers和linux-image apt-cache search：

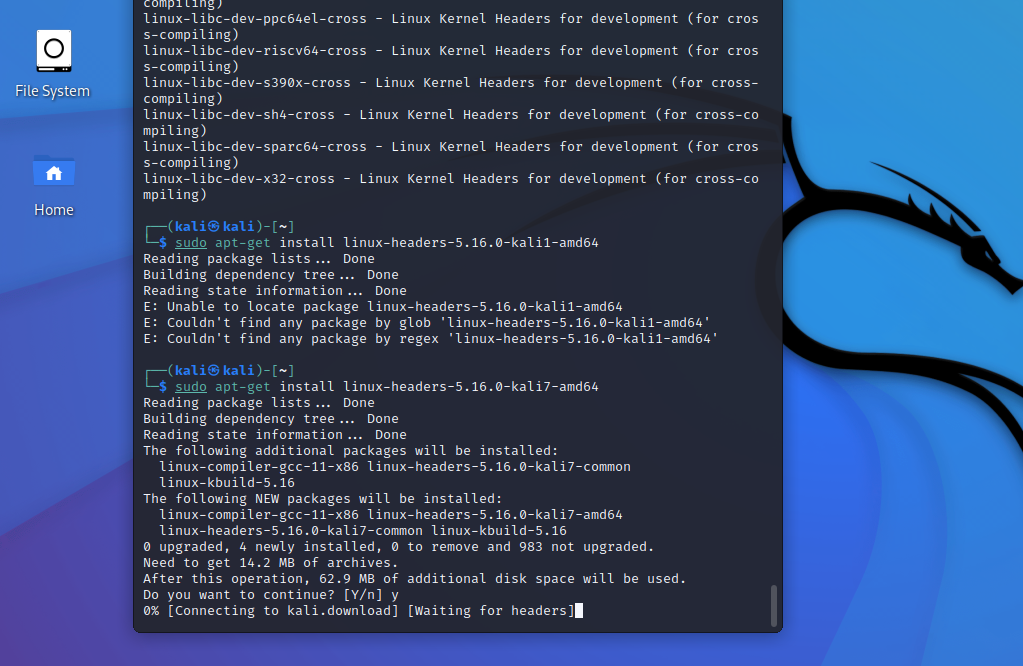
kali@Kali:~# apt-get update





1. 然后安装正确的软件包（这是一个示例，取决于先前的输出）.

sudo apt-get install linux-headers-5.16.0-kali7-amd64



输出的信息显示了linux-headers相关软件包的一个信息。提示将会安装哪些软件包及软件包的大小等信息。此时输入y，继续安装。安装完后，将退出程序。

3）复制generated下的所有内容。

$ sudo apt-get install linux-headers-5.16.0-kali7-amd64

Reading package lists... Done

Building dependency tree... Done

Reading state information... Done

The following additional packages will be installed:

linux-compiler-gcc-11-x86 linux-headers-5.16.0-kali7-common

linux-kbuild-5.16

The following NEW packages will be installed:

linux-compiler-gcc-11-x86 linux-headers-5.16.0-kali7-amd64

linux-headers-5.16.0-kali7-common linux-kbuild-5.16

0 upgraded, 4 newly installed, 0 to remove and 983 not upgraded.

Need to get 14.2 MB of archives.

After this operation, 62.9 MB of additional disk space will be used.

Do you want to continue? [Y/n] y

Get:1 http://kali.download/kali kali-rolling/main amd64 linux-compiler-gcc-11-x86 amd64 5.16.18-1kali1 [1,152 kB]

Get:2 http://kali.download/kali kali-rolling/main amd64 linux-headers-5.16.0-kali7-common all 5.16.18-1kali1 [9,988 kB]

Get:3 http://kali.download/kali kali-rolling/main amd64 linux-kbuild-5.16 amd64 5.16.18-1kali1 [1,398 kB]

Get:4 http://kali.download/kali kali-rolling/main amd64 linux-headers-5.16.0-kali7-amd64 amd64 5.16.18-1kali1 [1,671 kB]

Fetched 14.2 MB in 12s (1,144 kB/s)

Selecting previously unselected package linux-compiler-gcc-11-x86.

(Reading database ... 289376 files and directories currently installed.)

Preparing to unpack .../linux-compiler-gcc-11-x86\_5.16.18-1kali1\_amd64.deb ...

Unpacking linux-compiler-gcc-11-x86 (5.16.18-1kali1) ...

Selecting previously unselected package linux-headers-5.16.0-kali7-common.

Preparing to unpack .../linux-headers-5.16.0-kali7-common\_5.16.18-1kali1\_all.deb ...

Unpacking linux-headers-5.16.0-kali7-common (5.16.18-1kali1) ...

Selecting previously unselected package linux-kbuild-5.16.

Preparing to unpack .../linux-kbuild-5.16\_5.16.18-1kali1\_amd64.deb ...

Unpacking linux-kbuild-5.16 (5.16.18-1kali1) ...

Selecting previously unselected package linux-headers-5.16.0-kali7-amd64.

Preparing to unpack .../linux-headers-5.16.0-kali7-amd64\_5.16.18-1kali1\_amd64.deb ...

Unpacking linux-headers-5.16.0-kali7-amd64 (5.16.18-1kali1) ...

Setting up linux-compiler-gcc-11-x86 (5.16.18-1kali1) ...

Setting up linux-kbuild-5.16 (5.16.18-1kali1) ...

Setting up linux-headers-5.16.0-kali7-common (5.16.18-1kali1) ...

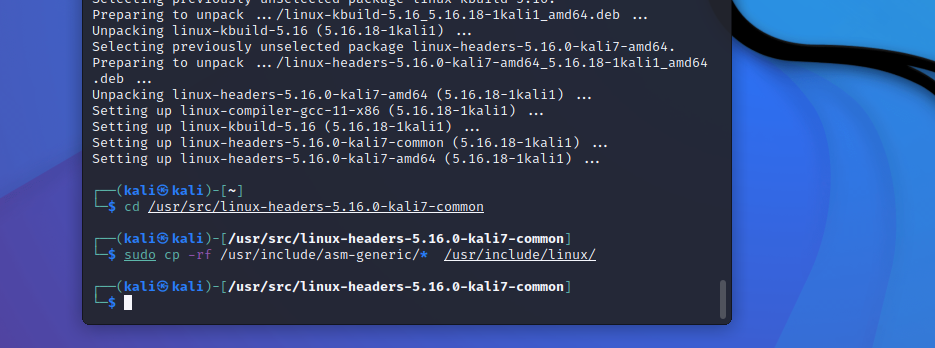
Setting up linux-headers-5.16.0-kali7-amd64 (5.16.18-1kali1) ...

执行命令如下所示：

kali@Kali:~# cd /usr/src/linux-headers-5.16.0-kali7-common/

kali@Kali:/usr/src/linux-headers-5.16.0-kali7-common# sudo cp -rf /usr/include/asm-generic/\* /usr/include/linux/

4）我们现在已准备好编译需要内核头文件的代码。



这次实验步骤主要讲解如何安装Nessus。

**操作步骤**

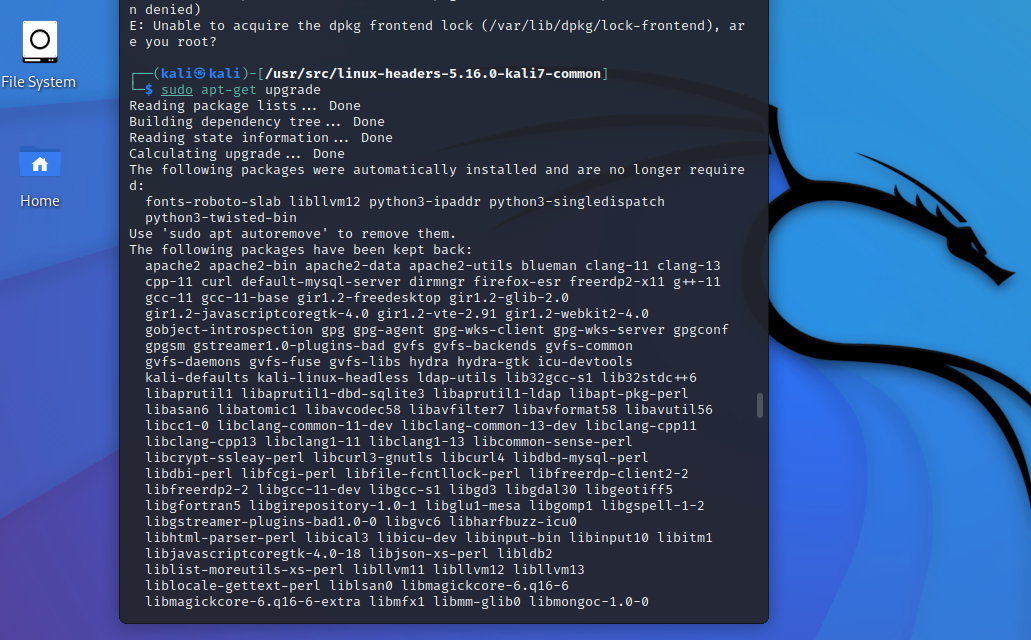
让我们开始进行升级，以及配置额外的安全工具。

1. 使用仓库中最新的修改来更新本地的包索引：

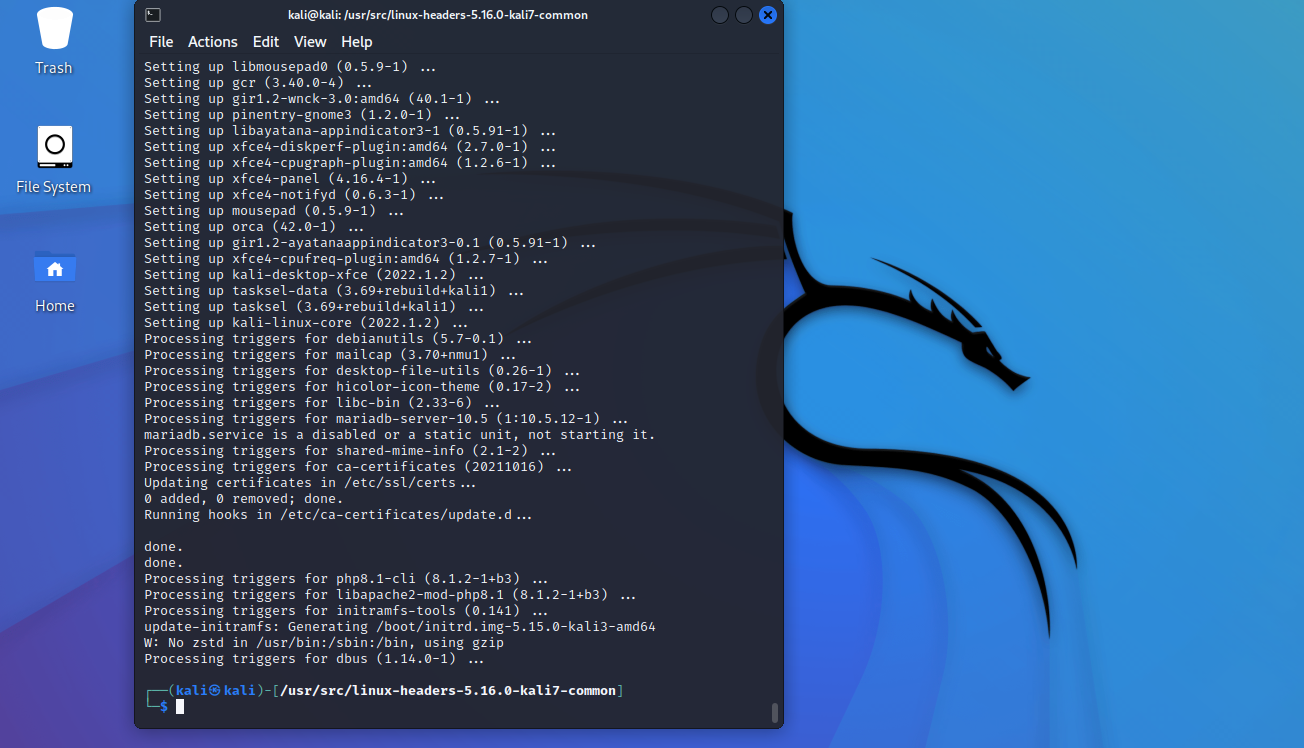
apt-get update

1. 升级现有的包：

apt-get upgrade



中间好大一段



1. 安装Nessus

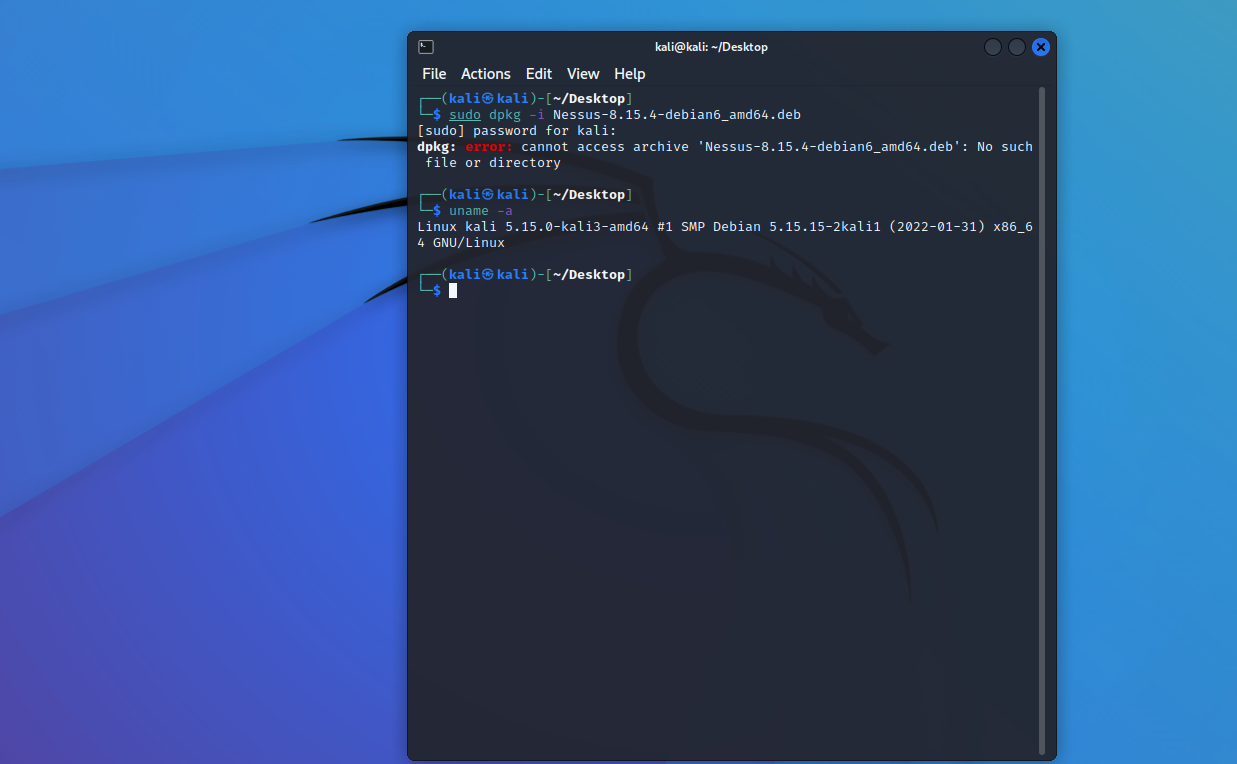
升级到最新版本（如果可用的话）：

apt-get dist-upgrade

[在这里去下载合适的版本](https://www.tenable.com/downloads/nessus?loginAttempted=true)

<https://www.tenable.com/downloads/nessus?loginAttempted=true>

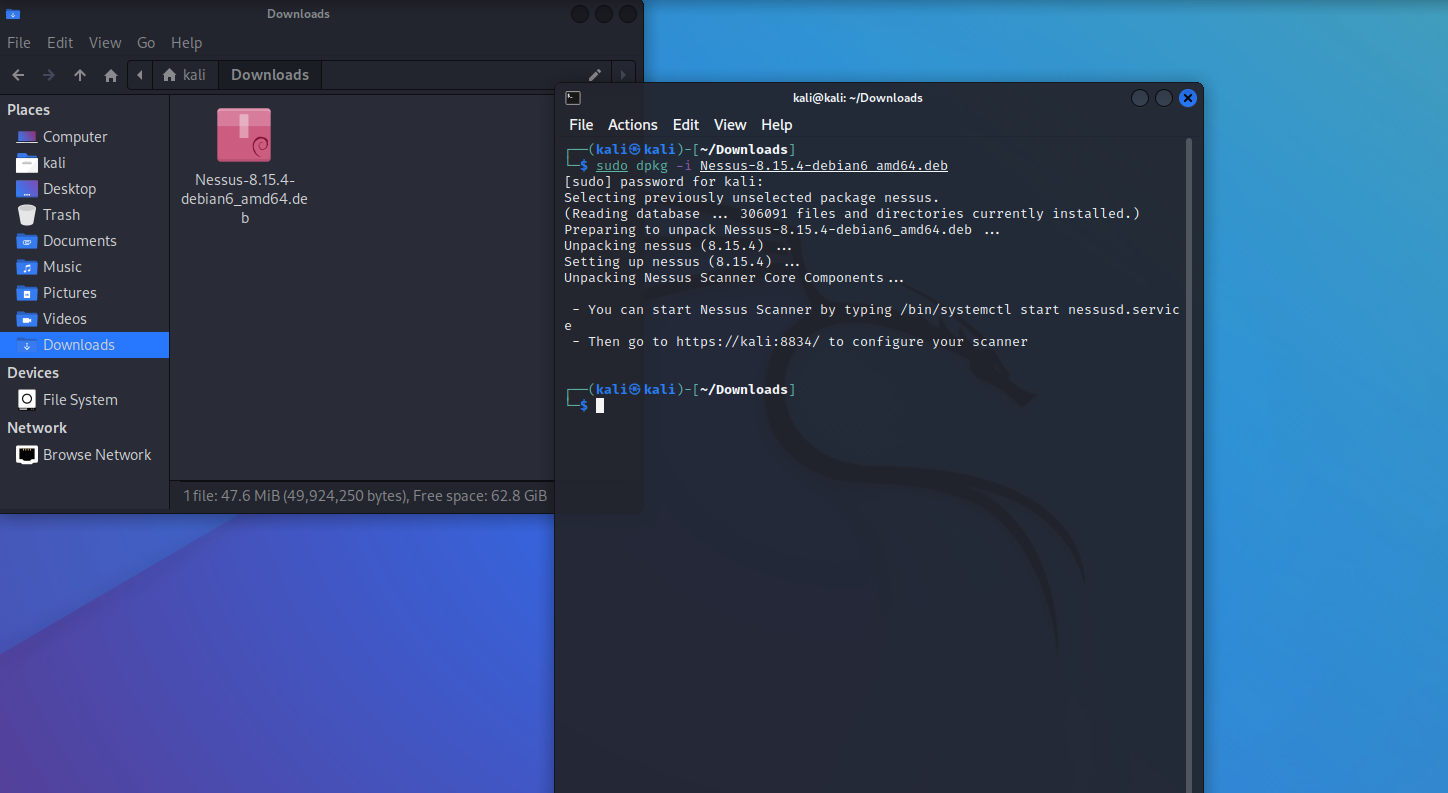
首先使用命令行： uname -a 查看系统信息。



选择对应的版本下载：

然后使用dpkg命令安装：

sudo dpkg -i Nessus-8.15.4-debian6\_amd64.deb



1. 获得Nessus的激活码，通过在[这里](http://www.nessus.org/products/nessus/nessus-plugins/obtain-an-activation-code" \t "_blank)注册。

<http://www.nessus.org/products/nessus/nessus-plugins/obtain-an-activation-code>

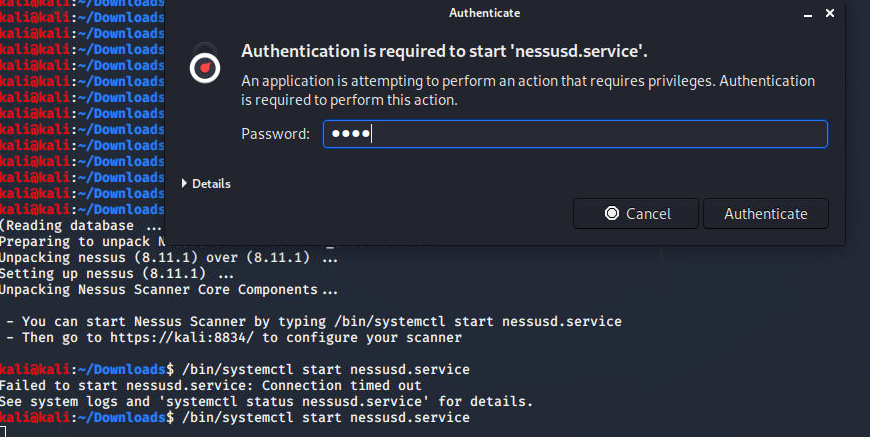
注册：



邮箱获取激活码

1. 使用命令行启动Nessus服务

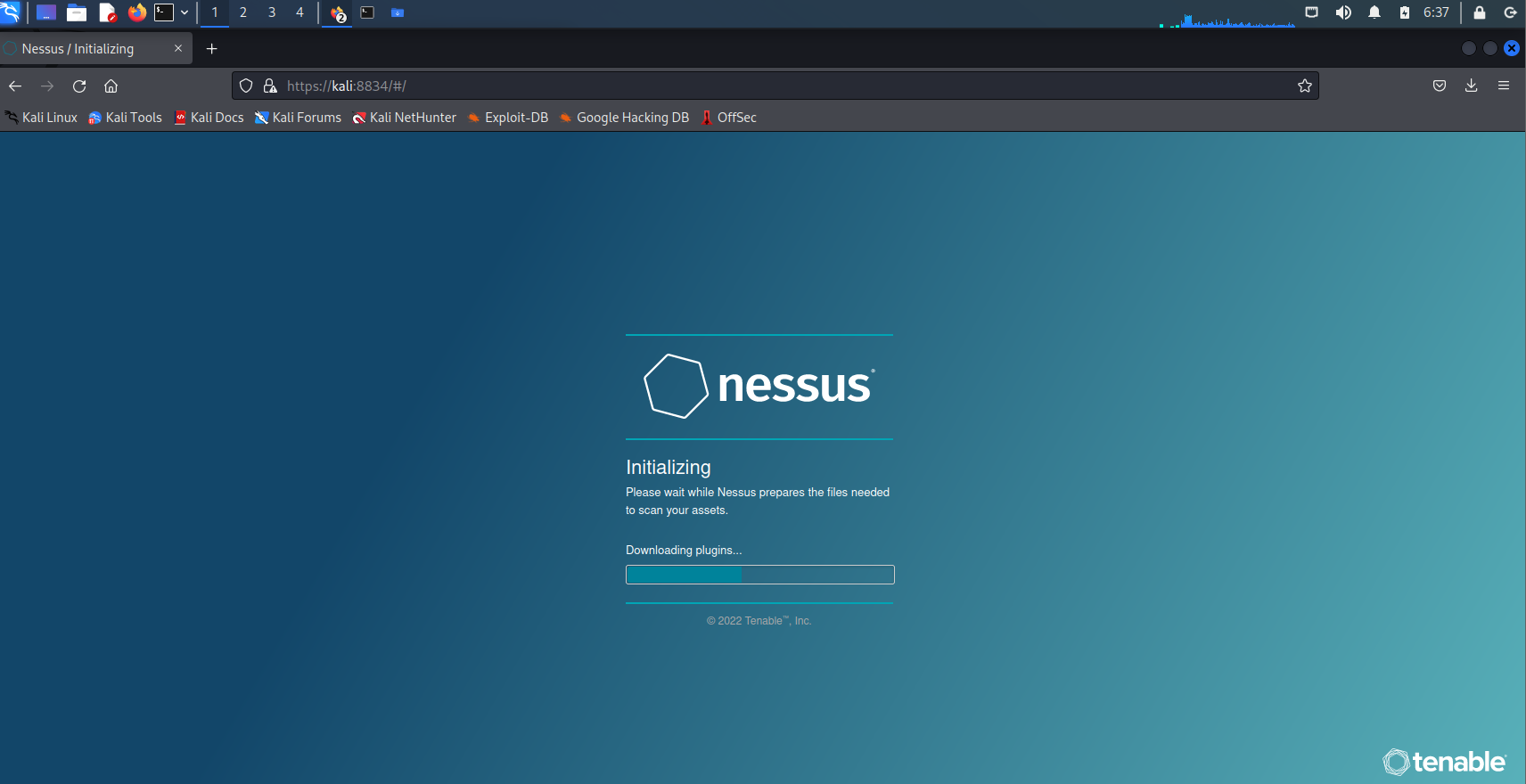
/bin/systemctl start nessusd.service



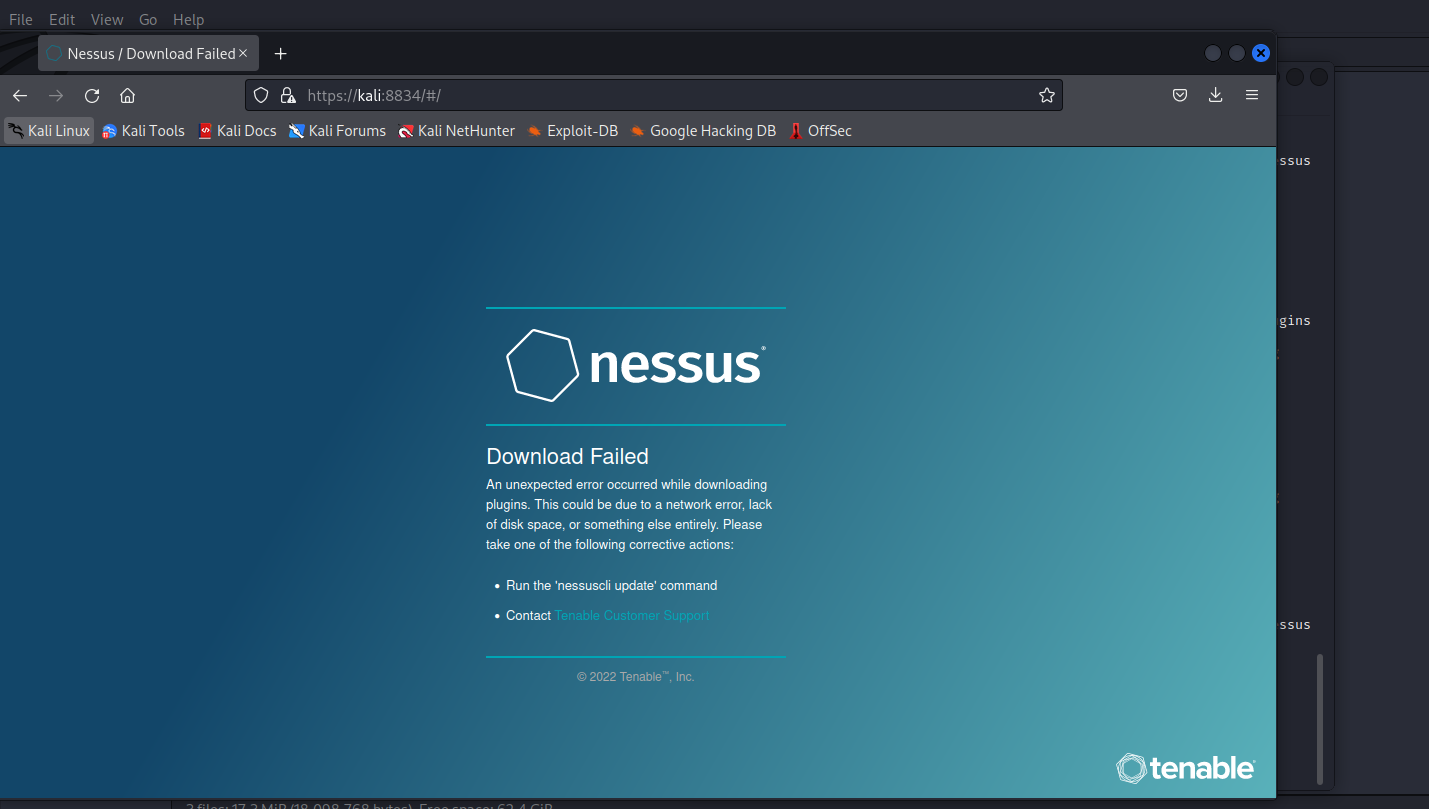
1. 运行Nessus

在浏览器当中输入：https://kali:8834/ 激活Nessus，完善配置。

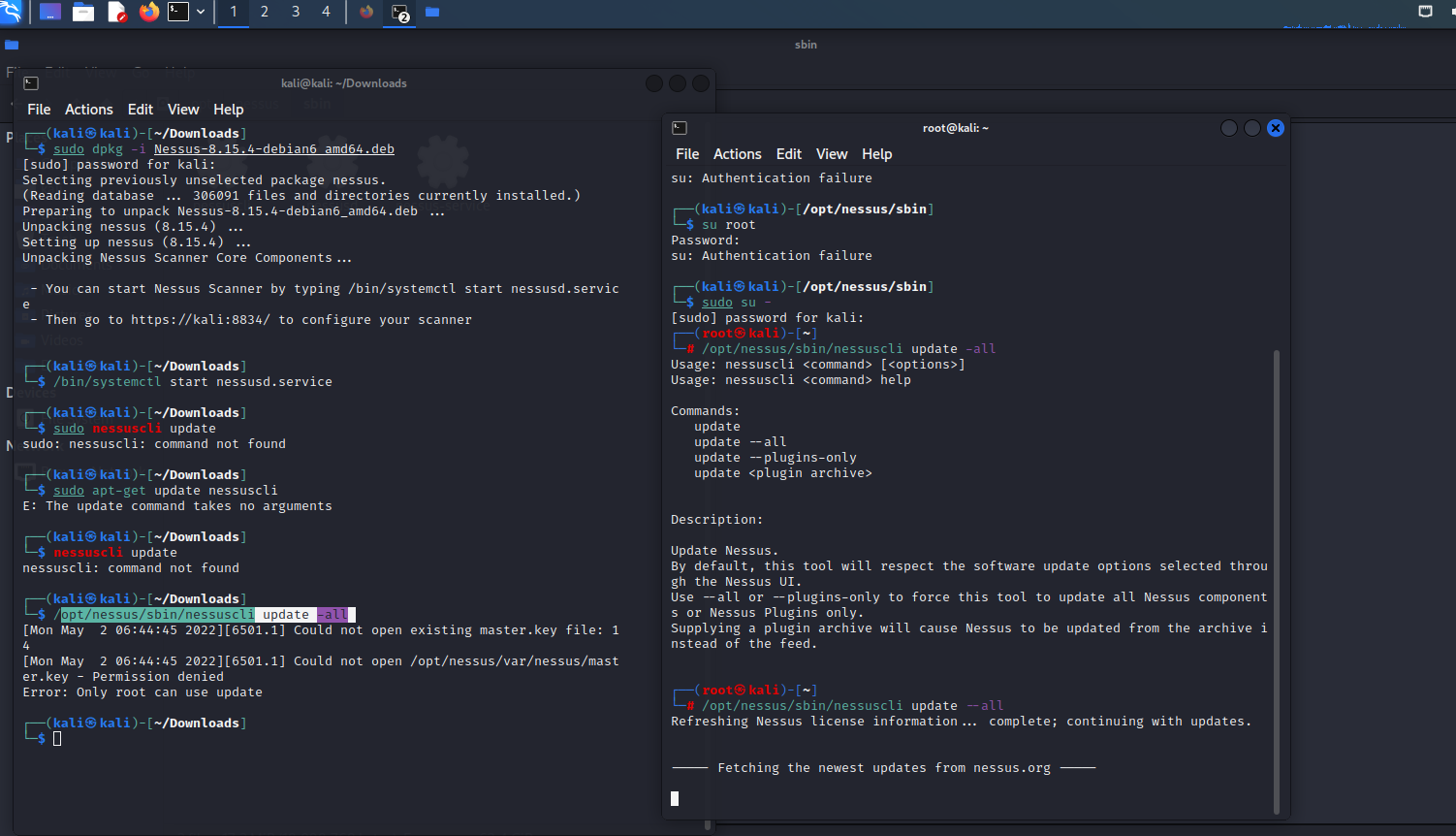
然后根据提示填写激活码和设置账号密码即可。 最后就是下载插件，注意该过程耗时较长。



下载失败



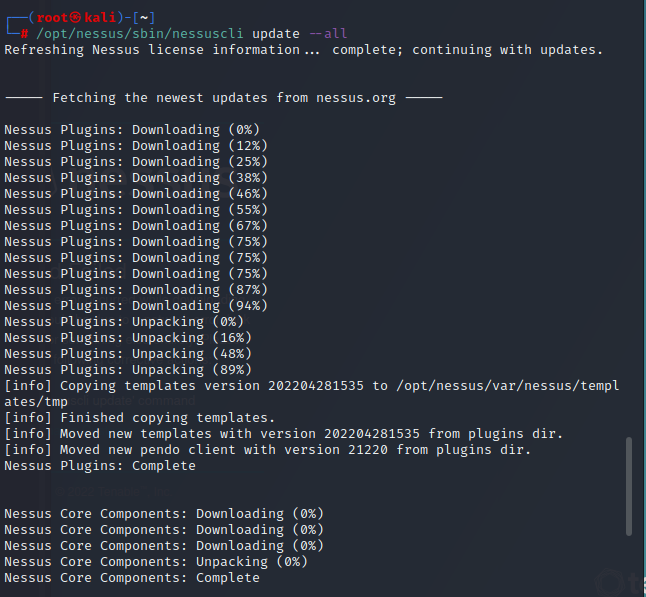
命令行更新，首先要获取root权限



先停止服务

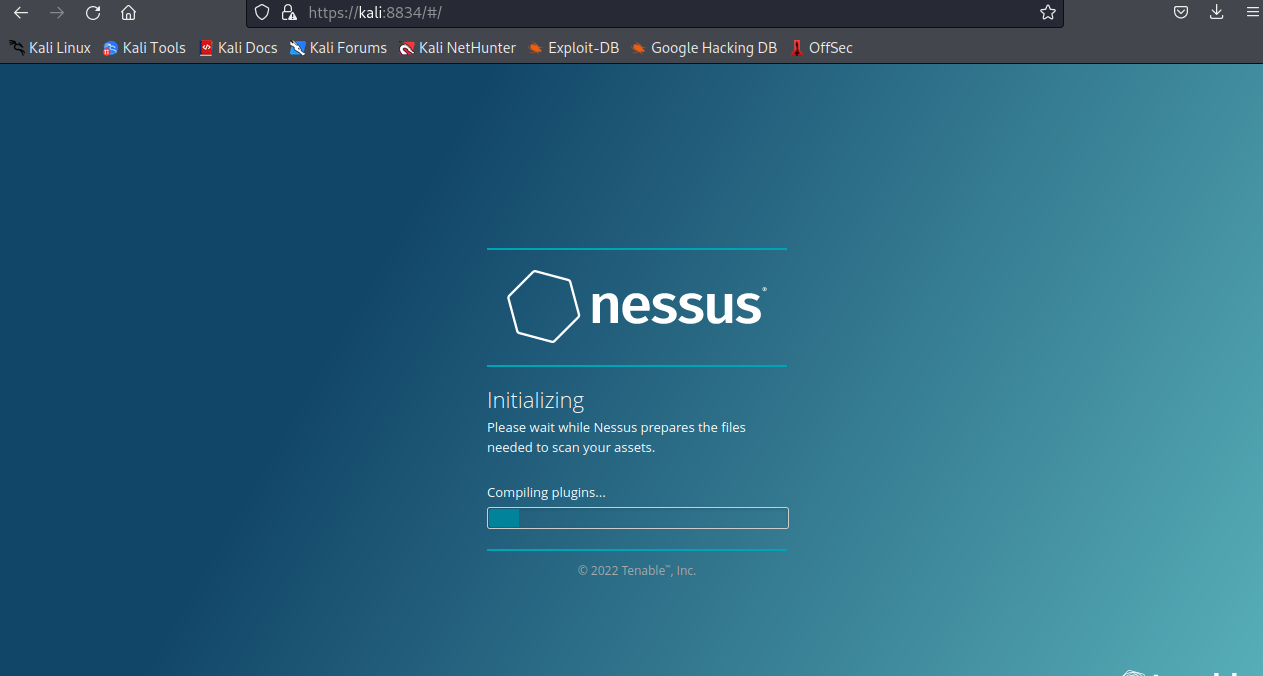


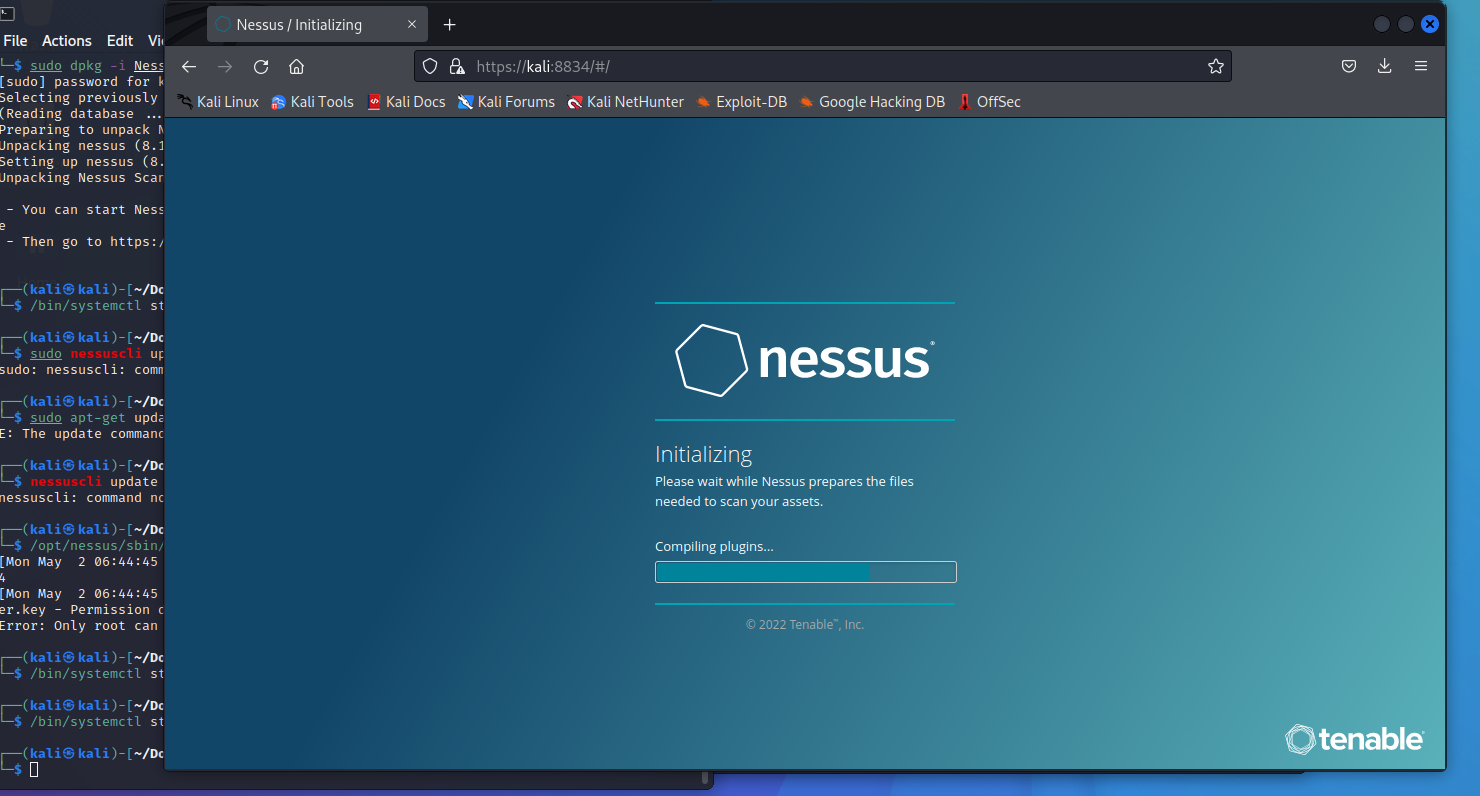
再进行更新

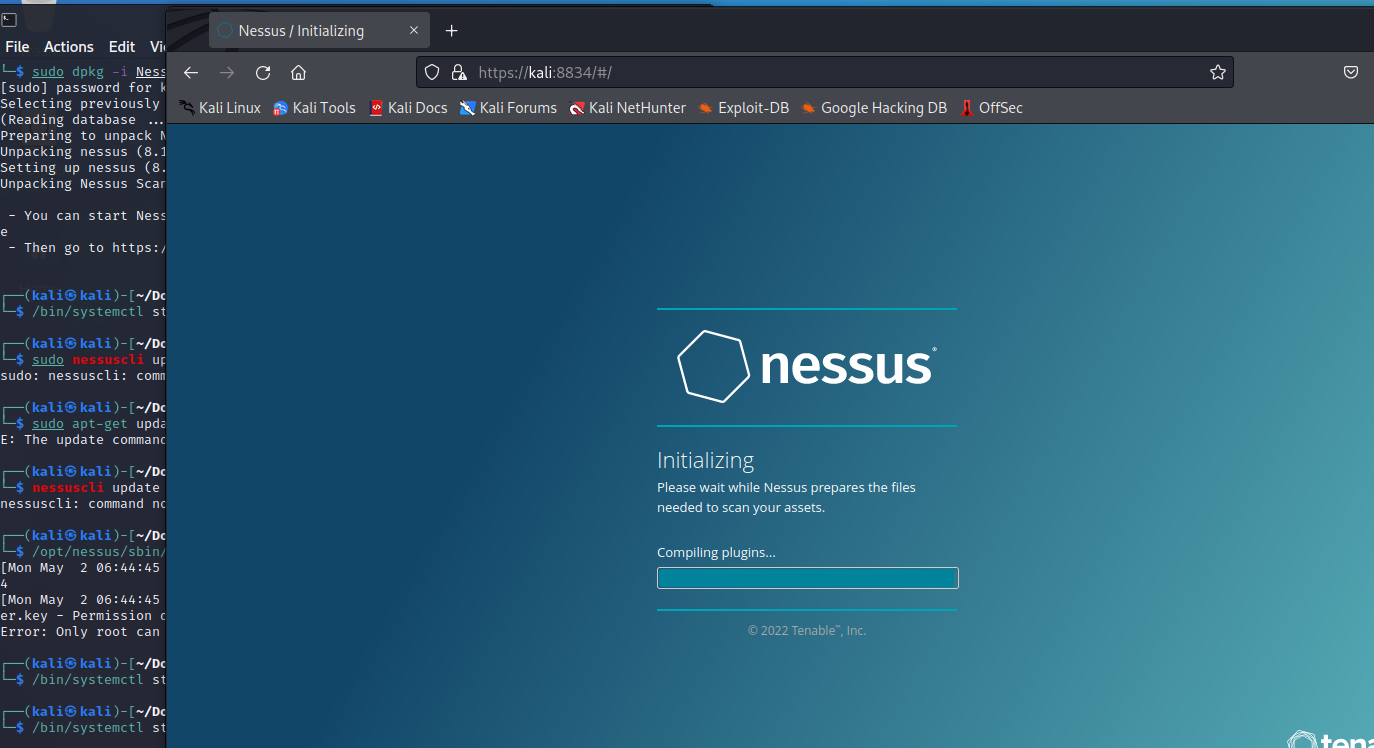


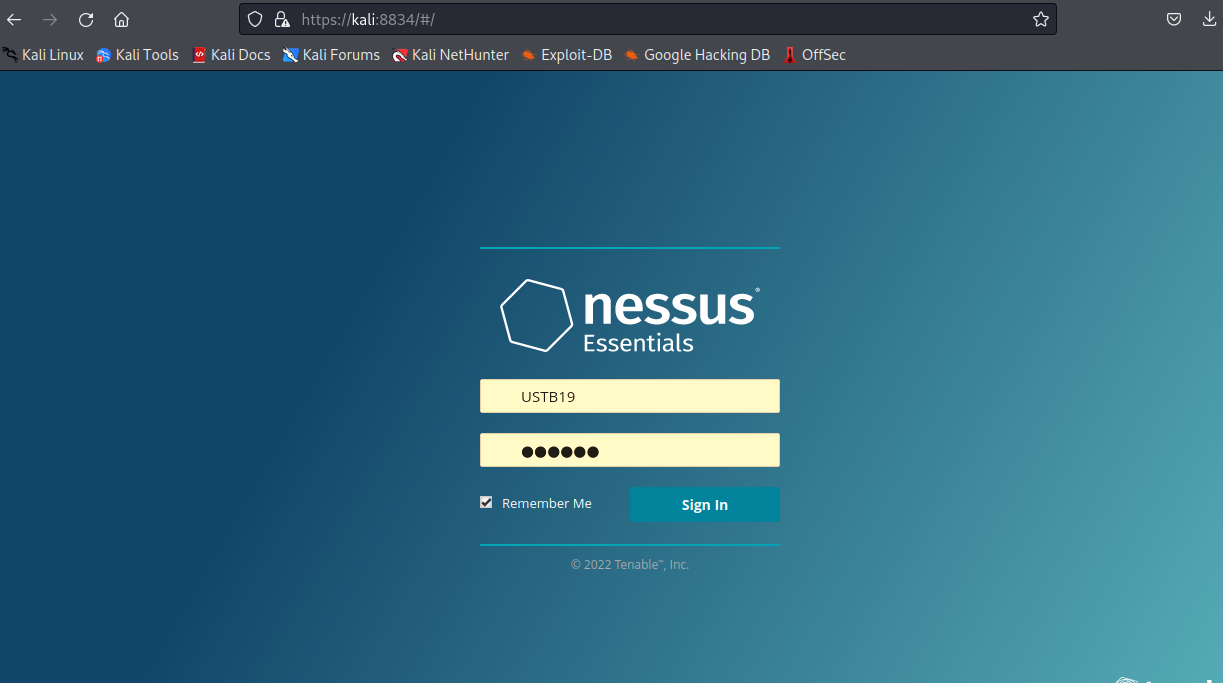


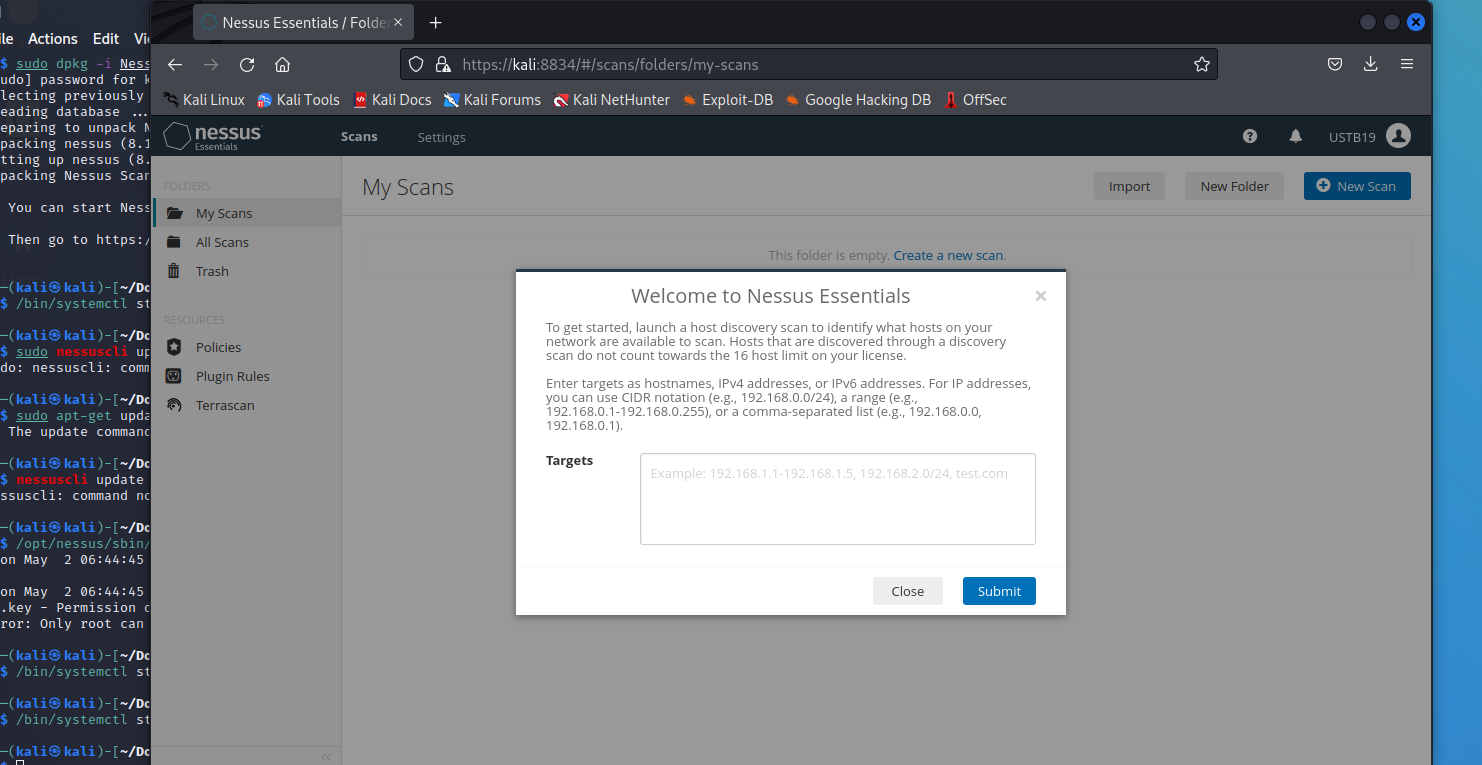
开启nessus后进入<https://kali:8834/>











1. Squid：

**Squid 是什么？**

Squid是一种用来缓冲Internet数据的软件。它是这样实现其功能的，接受来自人们需要下载的目标（object）的请求并适当地处理这些请求。也就是说，如果一个人想下载一web页面，他请求Squid为他取得这个页面。Squid随之连接到远程服务器（比如：[http://squid.nlanr.net/）](http://squid.nlanr.net/%EF%BC%89" \t "_blank) 并向这个页面发出请求。然后，Squid显式地聚集数据到客户端机器，而且同时复制一份。当下一次有人需要同一页面时，Squid可以简单地从磁盘中读到它，那样数据迅即就会传输到客户机上。当前的Squid可以处理HTTP，FTP，GOPHER，SSL和WAIS等协议。但它不能处理如POP，NNTP，RealAudio以及其它类型的东西。

Squid是一个高性能的代理缓存服务器，Squid支持FTP、gopher、HTTPS和HTTP协议。和一般的代理缓存软件不同，Squid用一个单独的、非模块化的、I/O驱动的进程来处理所有的客户端请求。

**Internet缓冲的一些概念**

你可能会想到一些问题：缓冲有多大的用处？什么时候目标（object）应该或者不应该被缓冲？例如，缓冲信用卡号码是完全不适合的，脚本文件的执行结果在远程服务器端，站点经常更新（像www.cnn.com） 或者甚至站点不允许缓冲，这些情况也都是不适合缓冲的。Squid处理各种情况是不错的(当然,这需要远程站点按标准形式工作)。可执行的cgi-bin脚本文件不被缓冲，返回正确页眉的页面是在一段限制了的时间内被缓冲，而且你可以规定特殊的规则以确定什么是可以或不可以被缓冲的，还有缓冲的时间为多长。谈到缓冲的用处有多大，这要看Internet的容量大小，各有不同。对于小型的缓冲区（几转磁盘空间）来说，返回值非常高（达到25%）。这个空间缓冲经常访问的站点，如netscape，cnn和其它一些类似情况的站点。如果你增加一倍缓冲的磁盘空间，但你不会成倍增加你的命中率。这是因为你开始缓冲网络中剩余部分时，这些通常时很大的而且很少被访问。一个非常大的高速缓冲区，有20转左右，可能返回值仍小于50%，除非你对保存数据的时间长短经常改变（一般地你不要分配20转的磁盘空间，因为页面很快就会过时，应该被删除掉）。我们在这里说的目标（object）指的是可保存的web页面或其它类似的可下载页面（ftp文件或目录内容也称为目标（object））。

**Squid运行在什么系统上？**

Squid可运行在大多数Unix和OS/2版本的系统之上，已知的可工作的有：Windows,AIX，Digital Unix，FreeBSD，HP-UX，Irix，Linux，NetBSD，Nextstep，SCO，Solaris

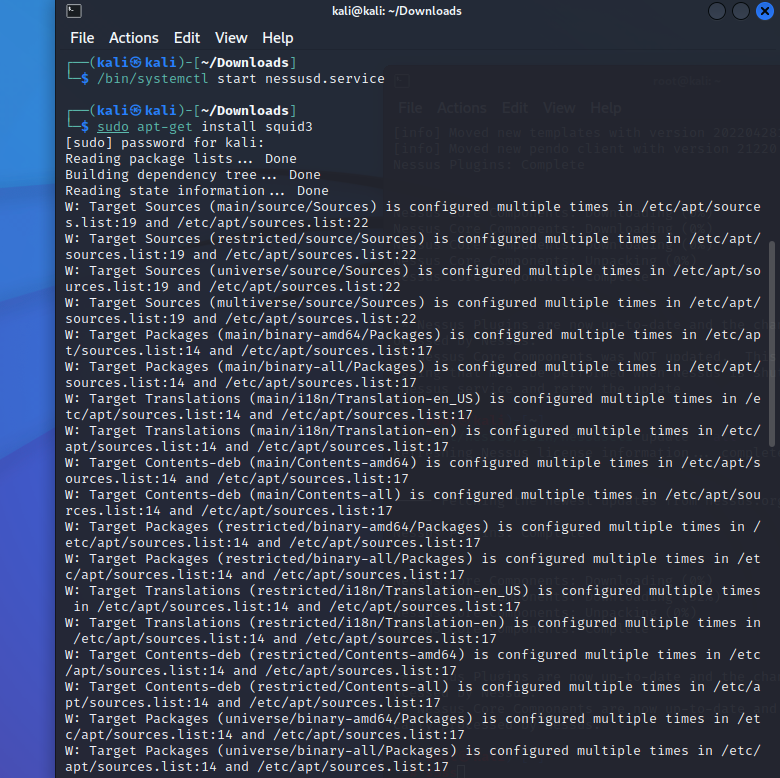
**Squidd的安装**

sudo apt-get install squid3

为了在仓库中找到特定的包，我们可以在apt-get update之后使用下列命令：

apt-cache search <keyword>

其中<keyword>是包名称或者正则表达式。



这个章节中，我们会强制指定应用的网络连接使用用户定义的代理列表，来打破接受者和发送者之间的直接连接。

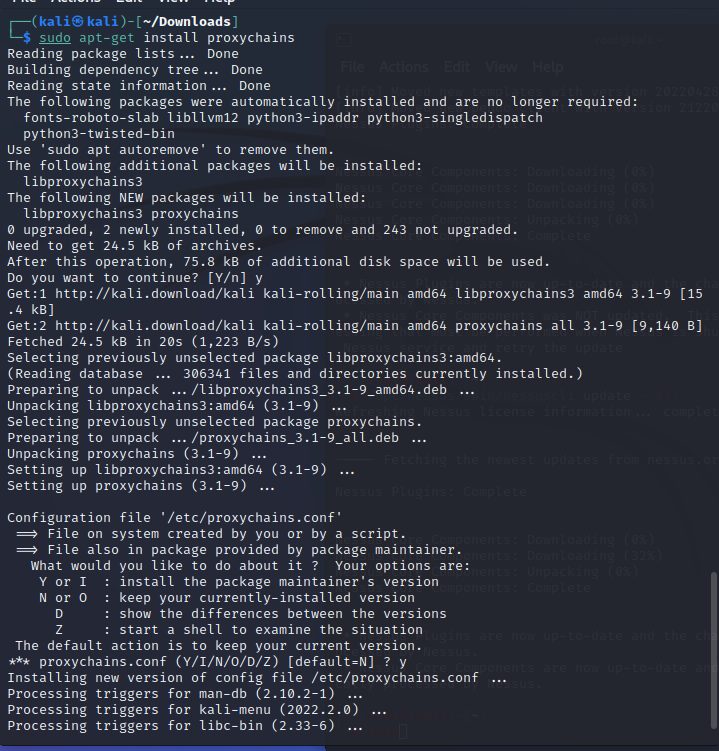
操作步骤

1. 打开ProxyChains的配置文件：

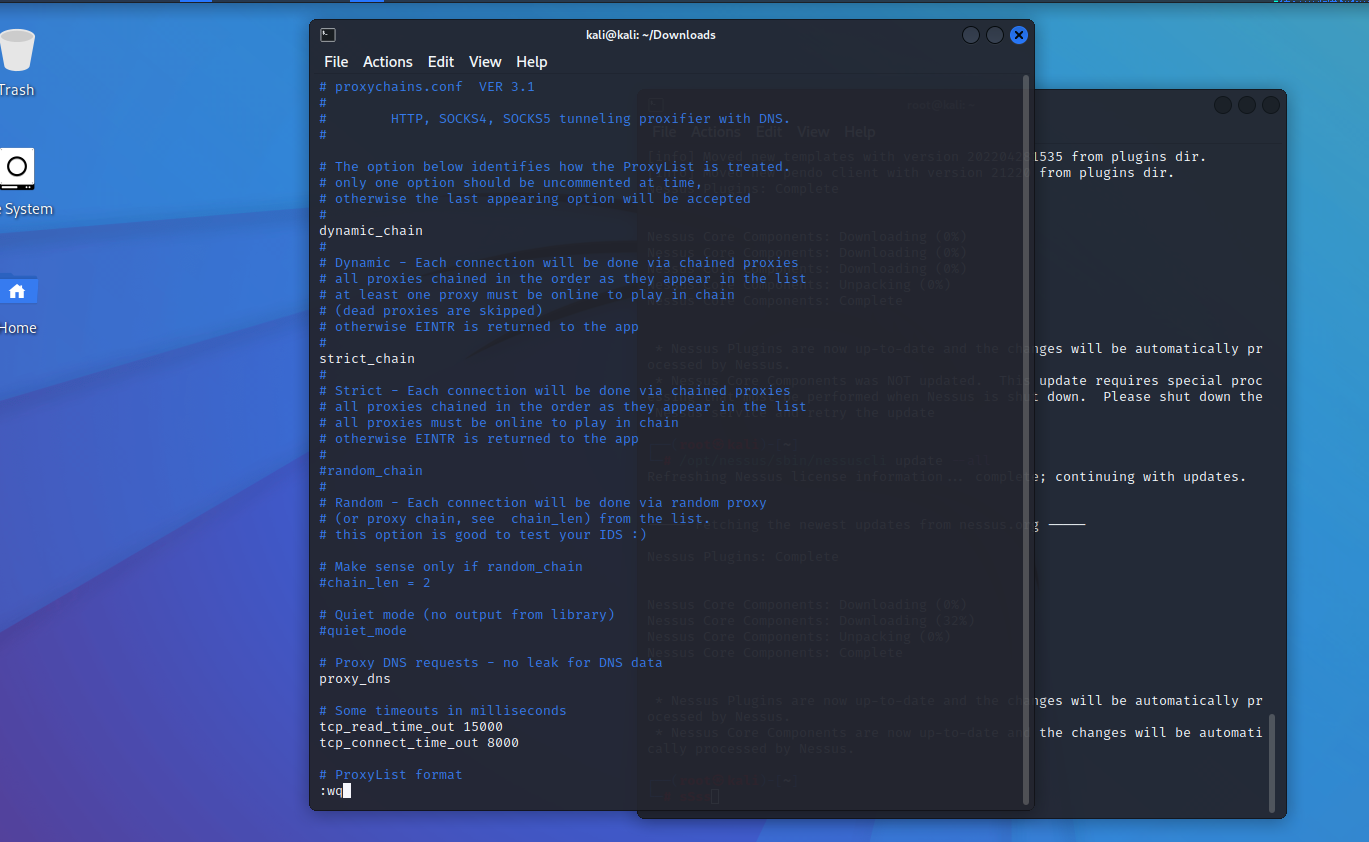
sudo vim /etc/proxychains.conf

因为我们使用的不是root用户，所以在使用命令行的时候，尽量加上sudo，避免权限的问题。

首先需要安装proxychains



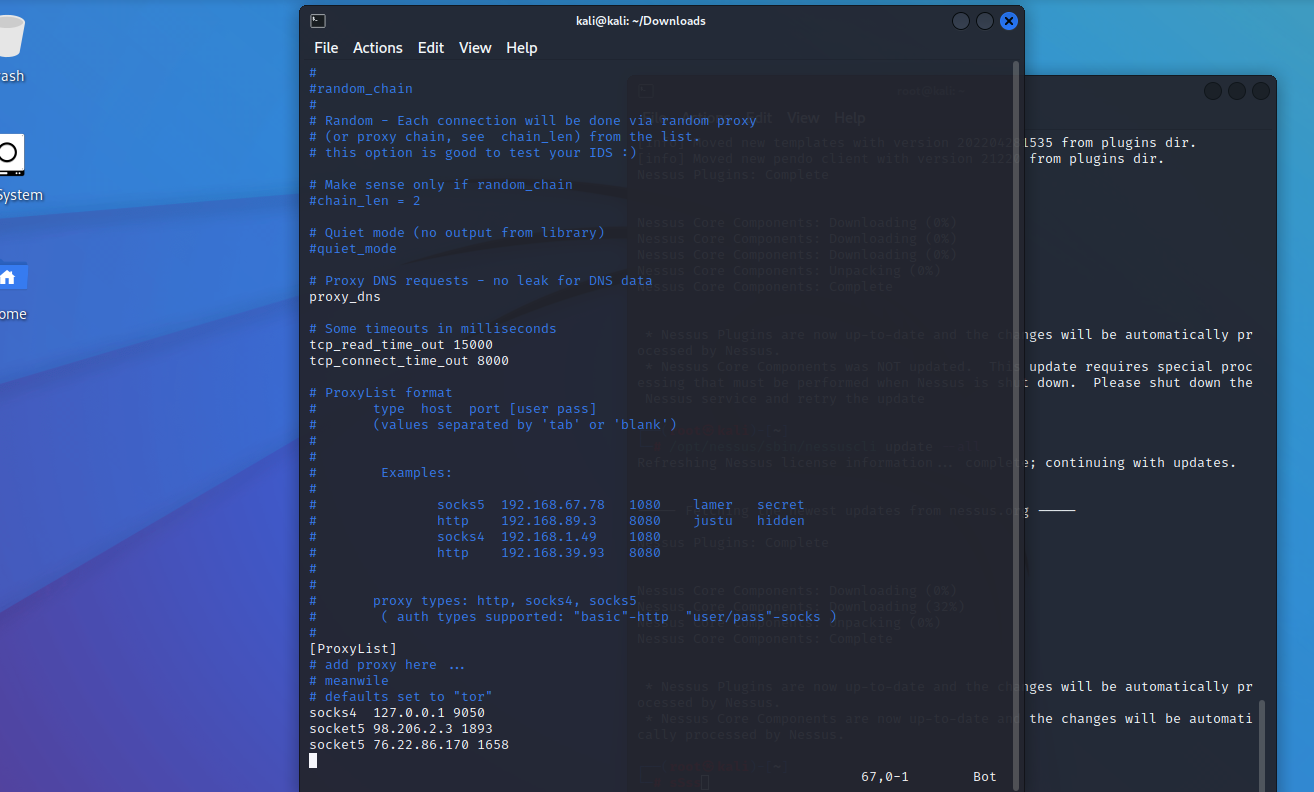
1. 解除我们打算使用的链接类型的注释，这里是dynamic\_chain：



1. 向列表中添加一些代理服务器：

socket5 98.206.2.3 1893

socket6 76.22.86.170 1658



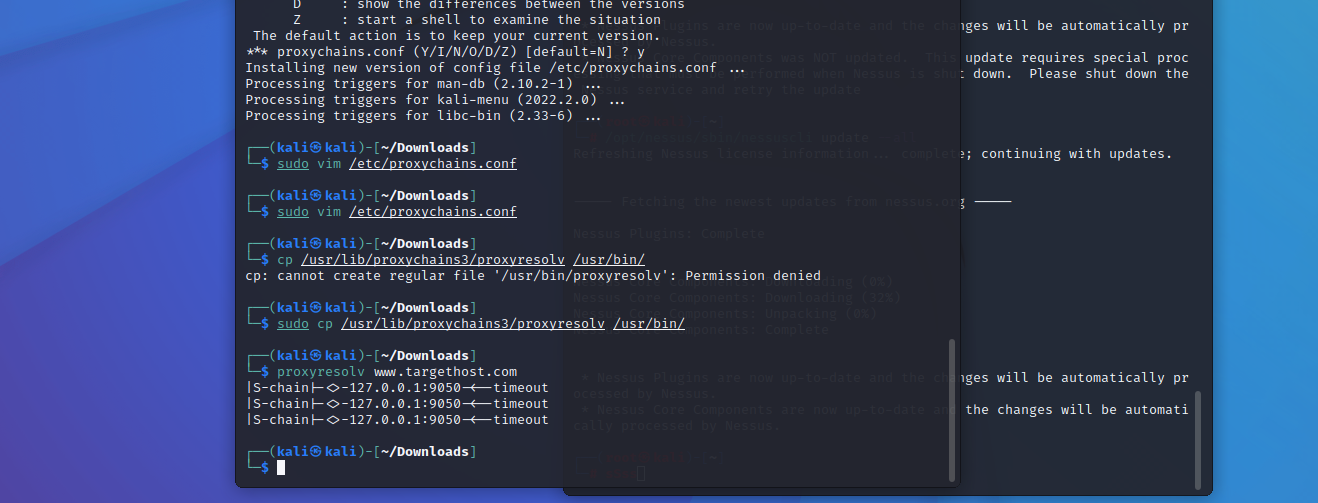
1. 使用我们的链式代理来解析目标主机：

proxyresolv www.targethost.com

默认情况下，执行proxyresolv命令，可能看到该命令没找到错误信息。因为proxyresolv保存在/usr/lib/proxychains3/目录中，而不能被执行。proxyresolv会被proxychains调用，所以将这两个文件放在一个目录中，如/usr/bin。执行命令如下所示：

kali@kali:~# cp /usr/lib/proxychains3/proxyresolv /usr/bin/

执行完以上命令后，proxyresolv命令就可以执行了。



1. 现在可以在我们打算使用的应用上运行ProxyChains，例如msfconsole：

proxychains msfconsole



## 实验数据

见上图

## 实验数据处理

无

## 实验结果与分析

Squid是一个高性能的代理缓存服务器，Squid支持FTP、gopher、HTTPS和HTTP协议。和一般的代理缓存软件不同，Squid用一个单独的、非模块化的、I/O驱动的进程来处理所有的客户端请求。

Squid可运行在大多数Unix和OS/2版本的系统之上，已知的可工作的有：Windows,AIX，Digital Unix，FreeBSD，HP-UX，Irix，Linux，NetBSD，Nextstep，SCO，Solaris

squid在诸多unix like系统上都有软件库的提供，比如在ubuntu可以使用apt install squid进行安装，也可以到squid的官网直接下载二进制的编译好的软件包