

一、开始最基本的使用

[Linux黑客基础-01-概述](#)

[Linux黑客基础-02-道德黑客和渗透测试](#)

[Linux黑客基础-03-Linux为什么成为黑客的首选](#)

[Linux黑客基础-04-Kali Linux的介绍](#)

[Linux黑客基础-05-Kali Linux的安装](#)

[Linux黑客基础-06-使用虚拟机文件部署Kali Linux](#)

[Linux黑客基础-07-常用的术语和概念](#)

[Linux黑客基础-08-Kali 之旅](#)

[Linux黑客基础-09-基本的Linux命令](#)

[Linux黑客基础-10-如何在Linux系统下查找文件](#)

[Linux黑客基础-11-快速理解通配符](#)

[Linux黑客基础-12-使用grep实现过滤](#)

[Linux黑客基础-13-systemctl服务控制工具基本使用](#)

[Linux黑客基础-14-文件的基本操作01-使用cat创建文件](#)

[Linux黑客基础-15-文件的基本操作02-在脚本中使用cat创建一个文件](#)

[Linux黑客基础-16-文件的基本操作03-创建.删除.移动.复制.重命名](#)

[Linux黑客基础-17-文件的基本操作04-小测试](#)

Linux黑客基础-01-概述

PTES Technical Guidelines PTES 技术指南

http://www.pentest-standard.org/index.php/PTES_Technical_Guidelines



主页
PTES 技术指南
在媒体上
常见问题

工具
这里有什么链接
相关更改
特殊页面
可打印版本
永久链接
页面信息

PTES 技术指南

本节旨在成为PTES技术指南，帮助定义渗透测试期间要遵循的某些程序。需要注意的是，这些只是行业中已使用的基线方法。它们需要由社区以及您自己的标准不断更新和更改。指南只是，在某些情况下可以推动您朝着某个方向前进并提供帮助，但并不是有关如何执行渗透测试的一套包罗万象的说明。跳出框框思考。



Technical Guidelines

查看源代码 查看历史记录 搜索渗透测试执行标准

<https://github.com/kali-docs-cn> kali中文文档

Linux黑客基础-02-道德黑客和渗透测试

What Is Ethical Hacking? 道德黑客

有道德的黑客是那些懂得如何攻击基础设施，并在漏洞被坏人利用之前发现它们的人

Penetration Testing 渗透测试

随着组织越来越注重安全性并且安全漏洞的成本呈指数级增长，许多大型组织开始将安全服务外包出去。其中一项关键安全服务是渗透测试。渗透测试本质上是一个合法的、受委托的黑客行为，以证明公司的网络和系统的脆弱性。

通常，组织首先进行漏洞评估，以发现其网络、操作系统和服务中的潜在漏洞。我强调是潜在的，因为此漏洞扫描包含大量误报（确定为漏洞的事实并非如此）。渗透测试人员的角色是试图破解或渗透这些漏洞。只有这样，组织才能知道漏洞是否真实，并决定投入时间和金钱来弥补漏洞。

渗透测试是一种有目的性的、针对目标机构计算机（网络）系统安全的**检测评估方法（手段）**。

渗透测试可以**发现系统的漏洞**和安全机制方面的隐患，并以此进行渗透攻击来取得目标计算机的控制权

正如“罗马不是一天就能建成的”，我们也不可能一天就成为专家级的渗透测试工程师。**从一个“菜鸟”转变成渗透高手需要大量的实践工作，熟悉工作环境，具备对危急情况的处理经验，而最为重要的是，需要在反复的渗透测试工作中不断加深自己对该技能的领悟。**

Military and Espionage 军事间谍活动

现在，世界上几乎每个国家都在进行网络间谍活动和网络战争。人们只需要浏览一下头条新闻，就会发现网络活动是监视和攻击军事和工业系统的选择方法。

黑客在这些军事和情报收集活动中起着至关重要的作用，随着时间的推移，这种情况才会更加真实。想象一下未来的战争，黑客可以获得对手的作战计划，并摧毁他们的电网、炼油厂和供水系统。这些活动现在每天都在进行。因此，黑客成为国家防御的关键组成部分。

渗透测试分类

黑盒测试

Black-box Testing 又称为外部测试，完全模拟真实网络环境中的外部攻击者。

白盒测试

White-box Testing 可以让渗透测试者以最小的代价发现和验证系统中弱点。

灰盒测试

Grey-box Testing 两者的结合

Linux黑客基础-03-Linux为什么成为黑客的首选

为什么黑客使用 LINUX

那么为什么黑客使用 Linux 而不是其他操作系统呢？主要是因为 Linux 通过几种不同的方法提供了更高级别的控制。

Linux 是开源的

与 Windows 不同，Linux 是开源的，这意味着您可以使用操作系统的源代码。因此，您可以随意更改和操作它。如果您试图使系统以它不希望的方式运行，那么能够操作源代码是必不可少的。

Linux 是透明的

要有效地进行黑客攻击，您必须了解并理解您的操作系统，并在很大程度上了解您正在攻击的操作系统。Linux 完全透明，这意味着我们可以查看和操作其所有工作部分。

Windows 则不然。微软努力让它变得越来越难以了解其操作系统的内部工作方式，所以你永远不会真正知道“幕后”会发生什么，而在 Linux 中，你会直接看到操作系统的每一个组件。这使得使用 Linux 更加有效。

Linux 提供粒度控制

Linux 是细粒度的。这意味着您对系统几乎拥有无限的控制权。在 Windows 中，您只能控制 Microsoft 允许您控制的内容。在 Linux 中，一切都可以由终端控制，无论在最微小的级别或最宏观的级

别。此外，Linux使任何脚本语言的脚本编写变得简单有效。

多数黑客工具都是为 Linux 编写的

超过 90%的黑客工具都是为 Linux 编写的。当然也有例外，如 Cain and Abel 以及 Wikto，这些例外证明这一规则。即使在为 Windows 移植诸如 Metasploit 或 nmap 等黑客工具时，并非所有功能都能从 Linux 移植。

未来属于 Linux/Unix

这似乎是一个激进的陈述，但我坚信信息技术的未来属于 Linux 和 Unix 系统。微软在 20 世纪 80 年代和 90 年代曾风光一时，但它的增长正在放缓并停滞。

自互联网诞生以来，由于其稳定性、可靠性和稳健性，Linux / Unix 一直是 Web 服务器的首选操作系统。即便在今天，Linux / Unix 仍在三分之二的 Web 服务器中使用并占据市场主导地位。路由器，交换机和其他设备中的嵌入式系统几乎总是使用 Linux 内核，而虚拟化世界则由 Linux 主导，VMware 和 Citrix 都基于 Linux内核构建。

超过 80%的移动设备运行 Unix 或 Linux（iOS 是 Unix，Android 是 Linux），所以如果您认为计算机的未来在于平板电脑和手机等移动设备（否则很难争辩），那么未来就是 Unix / Linux。Microsoft Windows 仅占移动设备市场的 7%^①。那是你想要的情形吗？

① 2019 年 1 月 10 日微软通过自己的官方博客宣布放弃 Win10 Mobile，于 2019 年 12 月终止支持。

Linux黑客基础-04-Kali Linux的介绍

Linux和Kali Linux的介绍

Linux 最初由 Linus Torvalds 于 1991 年开发，作为 Unix 的开源替代品。由于它是开源的，志愿者开发人员对内核、实用程序和应用程序进行提交编码。这意味着没有凌驾于其之上的公司实体来监督发展，因此，通常缺乏公约和标准化。

Kali Linux 由 Offensive Security 开发，是一个基于 Linux 发行版的黑客操作系统。Linux 有很多发行版，而 Debian 是最好的发行版之一

Kali Linux is the new generation of the industry-leading BackTrack Linux penetration testing (渗透测试) and security auditing (安全审计) Linux distribution. Kali Linux is a complete rebuild of BackTrack from the ground up, adhering completely to Debian development standards.

Kali 专为渗透测试人员和黑客而设计，并配有大量黑客工具。

您可能最熟悉 Ubuntu 作为 Linux 的流行桌面发行版。Ubuntu 也是基于 Debian 构建的。其他发行版包括 Red Hat, CentOS, Mint, Arch 和 SUSE。虽然它们都共享相同的 Linux 内核（控制 CPU, RAM 等操作系统的核心），但每个内核都有自己的实用程序、应用程序和用于不同目的的图形界面选择（GNOME, KDE 和其他）。因此，Linux 的这些发行版中的每一个外观和感觉都略有不同。

常用的渗透测试平台

Parrot Security OS <https://www.parrotsec.org/>



PentestBox---由印度人开发，运行在windows下的渗透测试环境，PentestBox是一款Windows平台下预配置的便携式开源渗透测试环境<https://pentestbox.org/zh/#download>



切换主题/颜色

提交/编辑工具

CTFTools

首页

工具合集

WEB工具

渗透环境

隐写工具

逆向工具

漏洞扫描

SQL注入

学习教程

暴力破解

链接失效请加入网安交流群 (点击加入) : 595176019@群主, 同时欢迎入群提供相关软件或者建议。

本工具库将持续更新:-)

作者博客

Dr3@m's Blog

项目地址

https://github.com/dr34-m/ctftools

上一次更新于: 2022-09-02, 欢迎前往Github提交PR, 顺便点个star我会高兴坏的

在线工具

MD5在线加密解密

CMD5

SOMD5

查MD5

常用编码

Base64编码

Unicode编码

Uri编码

网站相关

子域名爆破

端口扫描

Whois(万网)

在线运行

在线运行C(++)/Python/Java/PHP/Go/VB...

程序员工具箱

友情链接

Kali介绍

(1) 历史

Kali Linux (以前称为BackTrack Linux) 是一个开源的, 基于Debian的Linux发行版, 旨在进行高级渗透测试和安全审计。它通过提供通用工具、配置和自动化来实现这一点, 允许用户专注于需要完成的任务, 而不是周围的活动。

Kali Linux包含行业特定的修改以及针对各种信息安全任务的数百种工具, 例如渗透测试, 安全研究, 计算机取证, 逆向工程, 漏洞管理和红队测试。

Kali Linux是一个多平台解决方案, 信息安全专业人员和业余爱好者可以访问和免费获得。

(2) Kali Linux Features (特性)

- **More than 600 penetration testing tools included:** After reviewing every tool that was included in BackTrack, we eliminated a great number of tools that either simply did not work or which duplicated other tools that provided the same or similar functionality. Details on what's included are on the [Kali Tools](#) site.
- **Free (as in beer) and always will be:** Kali Linux, like BackTrack, is [completely free](#) of charge and always will be. You will never, ever have to pay for Kali Linux.
- **Open source Git tree:** We are committed to the open source development model and our [development tree](#) is available for all to see. All of the source code which goes into Kali Linux is available for anyone who wants to tweak or rebuild [packages](#) to suit their specific needs.
- **FHS compliant:** Kali adheres to the [Filesystem Hierarchy Standard](#), allowing Linux users to easily locate binaries, support files, libraries, etc.
- **Wide-ranging wireless device support:** A regular sticking point with Linux distributions has been support for wireless interfaces. We have built Kali Linux to support as many wireless devices as we possibly can, allowing it to run properly on a wide variety of hardware and making it compatible with numerous USB and other wireless devices.
- **Custom kernel, patched for injection:** As penetration testers, the development team often

needs to do wireless assessments, so our kernel has the latest injection patches included.

- **Developed in a secure environment:** The [Kali Linux team](#) is made up of a small group of individuals who are the only ones trusted to commit packages and interact with the repositories, all of which is done using multiple secure protocols.
- **GPG signed packages and repositories:** Every package in Kali Linux is signed by each individual developer who built and committed it, and the repositories subsequently sign the packages as well.
- **Multi-language support:** Although penetration tools tend to be written in English, we have ensured that Kali includes true multilingual support, allowing more users to operate in their native language and locate the tools they need for the job.
- **Completely customizable:** We thoroughly understand that not everyone will agree with our design decisions, so we have made it as easy as possible for our more adventurous users to [customize Kali Linux](#) to their liking, all the way down to the kernel.
- **ARMEL and ARMHF support:** Since ARM-based single-board systems like the [Raspberry Pi](#) and [BeagleBone Black](#), among others, are becoming more and more prevalent and inexpensive, we knew that [Kali's ARM support](#) would need to be as robust as we could manage, with fully working installations for both [ARMEL and ARMHF](#) systems. Kali Linux is available on [a wide range of ARM devices](#) and has ARM repositories integrated with the mainline distribution so tools for ARM are updated in conjunction with the rest of the distribution.

(1) More than 600 penetration testing tools included:

600多个渗透测试工具

<https://tools.kali.org/>

<https://tools.kali.org/tools-listing>

<https://github.com/enaqx/awesome-pentest>

工具的分类:

Information Gathering --信息搜集

Vulnerability Analysis --漏洞（脆弱性）分析

Wireless Attacks ---无线攻击

Web Applications ----Web 应用

Exploitation Tools--漏洞利用工具

Forensics Tools--电子取证

Stress Testing---压力测试

Sniffing & Spoofing---嗅探 & 欺骗

Password Attacks---密码攻击

Maintaining Access--维持访问

Reverse Engineering--逆向工程

Hardware Hacking--硬件破解

Reporting Tools--报告工具

(2) Free (as in beer) and always will be:

永远免费

(3) Open source Git tree

源代码在Git仓库中可以获取

Git 是 Linus Torvalds 为了帮助管理 Linux内核开发而开发的一个开放源码的版本控制软件。

<https://github.com/>

(4) FHS compliant

与FHS（文件系统层次标准）兼容

(5) Wide-ranging wireless device support:

广泛的无线设备（网卡）的支持

(6) Custom kernel, patched for injection

内核的定制，补丁的注入

(7) Developed in a secure environment

在一个安全的环境中开发

(8) GPG signed packages and repositories

软件包和软件仓库都是经过GPG签名

(9) Multi-language support:

支持多种语言

(10) Completely customizable:

支持完全定制

(11) ARMEL and ARMHF support

ARM处理器的支持

典型的设备如 Raspberry Pi (树莓派)

Linux黑客基础-05-Kali Linux的安装

<https://www.kali.org/docs/installation/hard-disk-install/> 安装文档

注：在bios中开启虚拟化功能

准备好安装镜像

建议大家创建一个专用的实验文件夹（英文命名）

推荐配置（最小）：

RAM：2GB



Hard disk：60GB

CPU：2个核心

网络类型：NAT

虚拟机上网问题（IP地址自动获取）：打开服务管理工具：services.msc，确保两个服务开启：

VMware DHCP Service, VMware NAT Service

 VMware Authorization Se...	正在运行	自动	本地系统
 VMware DHCP Service	正在运行	自动	本地系统

基本命令：

Ctrl+ALT+F2--切换到控制台（另一个终端）

Ctrl+ALT+F5--回到图形窗口

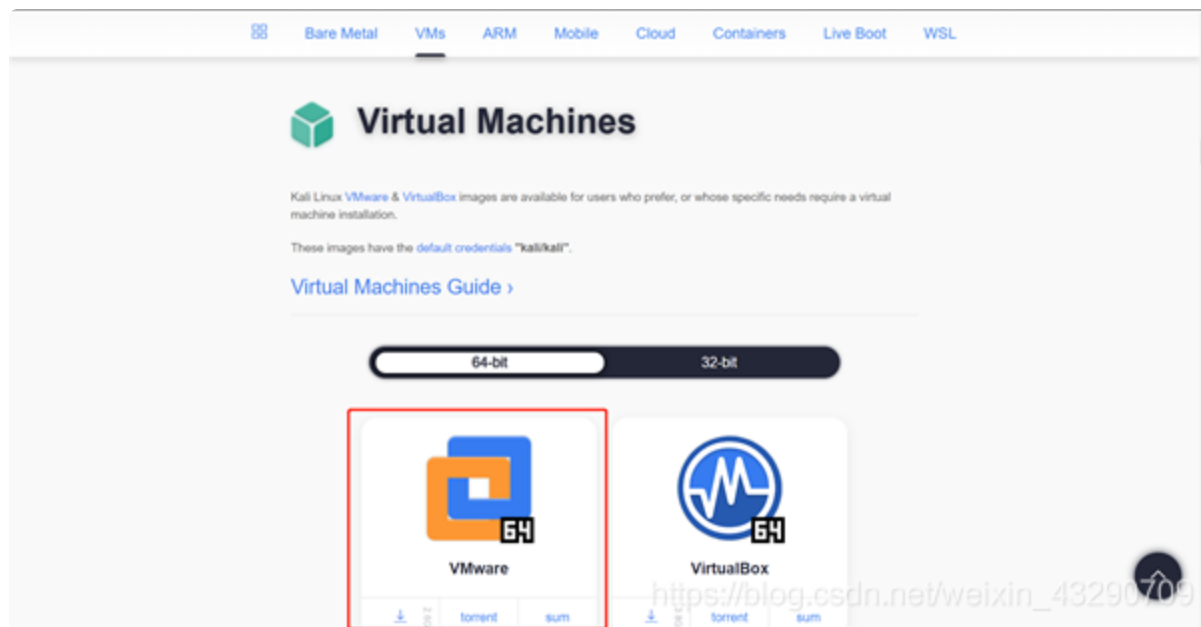
ip a 查看IP地址

ip r 查看路由

cat /etc/resolv.conf 查看DNS

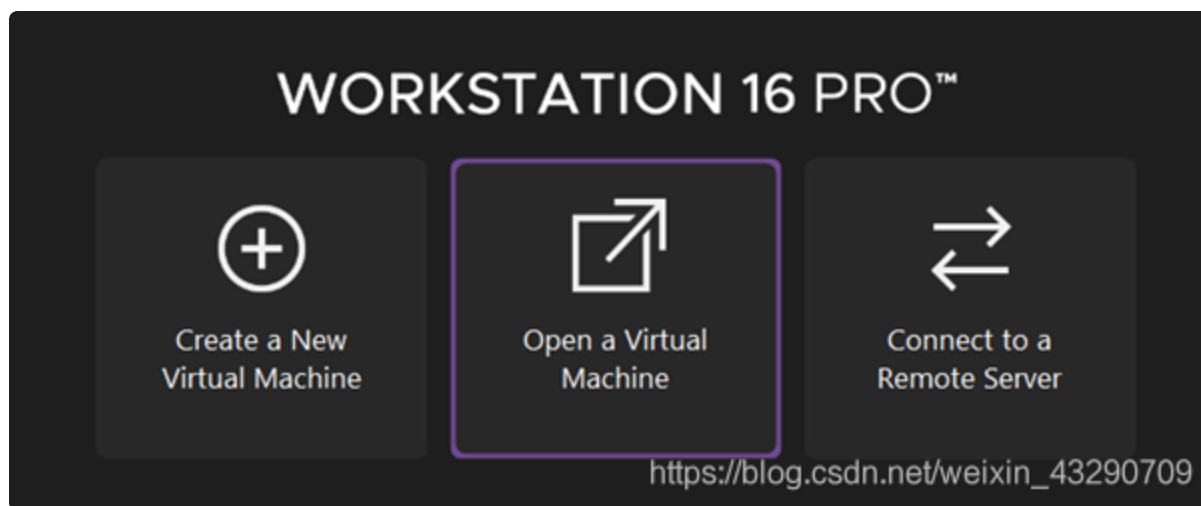
Linux黑客基础-06-使用虚拟机文件部署Kali Linux

点击下载VMware版本的kali Linux，下载后解压



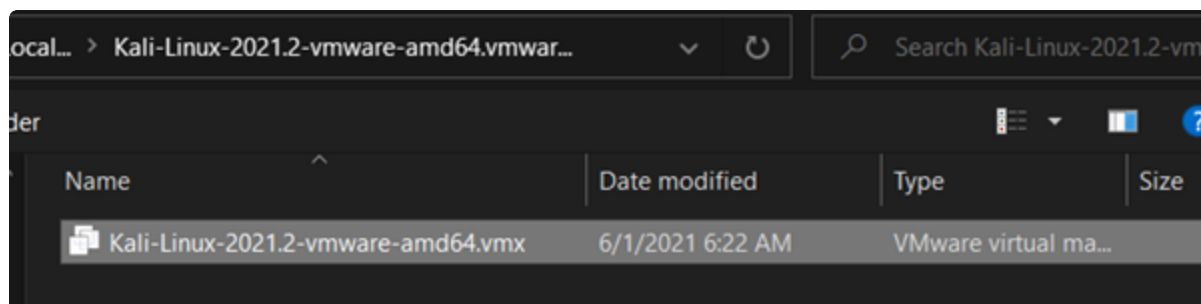
STEP 2 \textcolor{FF8F80}{STEP 2}STEP2

在VMware主页选择打开一个虚拟机



STEP 3 \textcolor{FF8F80}{STEP 3}STEP3

选择刚刚下载并解压后的文件



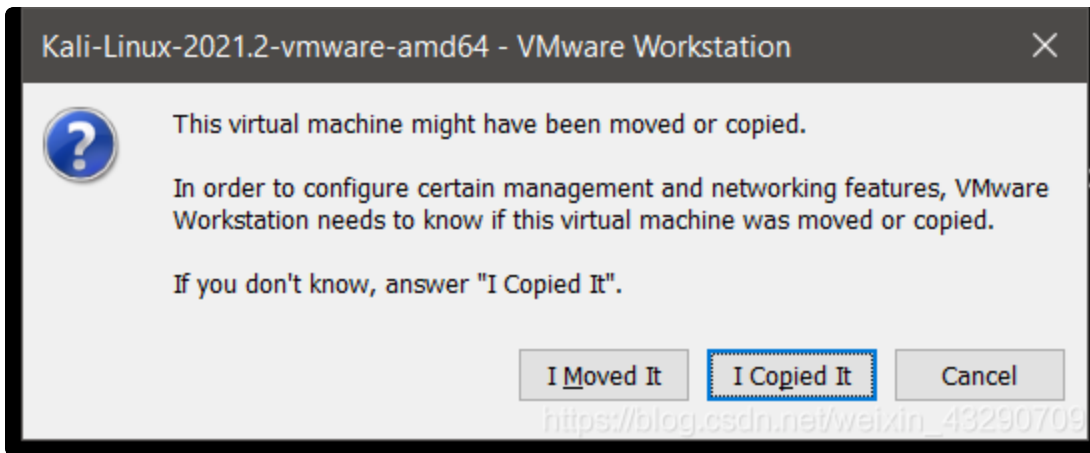
STEP 4 \textcolor{FF8F80}{STEP 4}STEP4

这样就可以直接打开虚拟机使用啦，可以编辑虚拟机属性进行个性化配置



STEP 5 \textcolor{FF8F80}{STEP 5}STEP5

启动虚拟机时，若弹出这个界面，选择“I Copied It ”即可



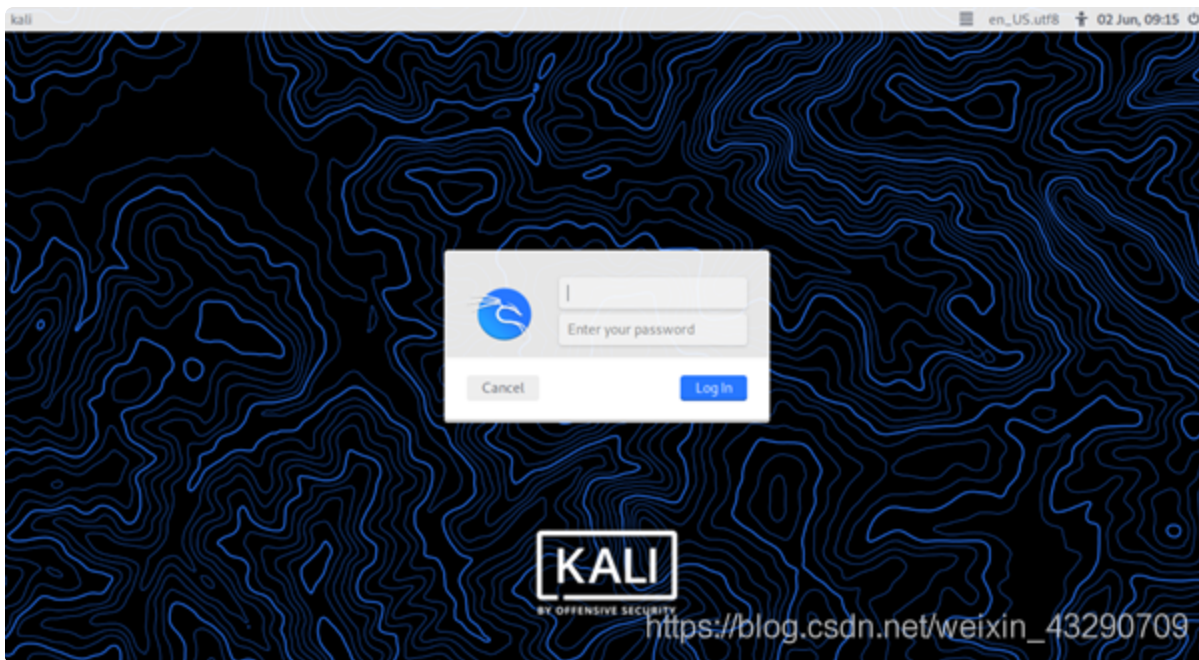
STEP 6 \textcolor{FF8F80}{STEP 6}STEP6

选择第一项启动



STEP 7 \textcolor{FF8F80}{STEP 7}STEP7

默认的初始用户及密码都是Kali，关于如何更改用户名及密码，点击此处（近期会更新）



P.s 可以调整屏幕分辨率或拉伸屏幕，使之符合主机的屏幕显示

Linux黑客基础-07-常用的术语和概念

Introductory Terms and Concepts 常用的术语的概念

(1) binaries(二进制)

二进制文件，可执行，类似于windows中的.exe文件

位置： /usr/bin-----普通程序，如： cat (查看)

 /usr/sbin-----管理程序，如halt(关机)

(2) Case sensitivity (区分大小写)

Unlike Windows, Linux is case sensitive (不像windows, linux是区分大小写的)

(3) Directory (目录)

This is the same as a folder in Windows. A directory provides a way of organizing files, usually in a hierarchical manner (目录这与Windows 中的文件夹相同。目录提供了一种组织文件的方式，通常是采用分层方法。)

(4) Home (家目录) ~表示家

Each user has their own /home directory, and this is generally where files you create will be saved by default. 每个用户在自己的目录下都有一个家目录

(5) root

linux下的管理员 (超级用户)

/root ---root用户的家目录

└───(root👤kali)-[~]

└───# sudo su - 切换到管理员，相当于sudo su -root(相当于提权)

su 切换用户身份，如果没有指定用户，默认就是root用户

命令： su - 用户 -表示切换到用户的工作环境

(6) script (脚本)

脚本，就是一段可执行的代码，许多黑客工具都是一段简单的脚本(Many hacking tools are simply scripts)

scripting language interpreters, such as Python, Perl, or Ruby.
(常用的脚本语言：shell、python、perl、ruby)

Python is currently the most popular interpreter among hackers. (python是目前黑客最喜欢用的脚本语言)

(7) shell：这是一个在linux中运行命令的环境和解释器

常用的shell是Bash，而在kali中默认的shell是zsh(/usr/bin/zsh)

A terminal window screenshot with a dark background. The prompt is `(root👁kali)-[~]` in red and white. Below it, the command `# echo $SHELL` is entered in green. The output `/usr/bin/zsh` is shown in white.

kali当中zsh的优势

1. 更加用户友好的命令提示符：zsh支持自定义命令提示符，可以将当前目录、用户名、主机名等信息显示在提示符中，方便用户了解当前所处环境。
2. 更强大的自动补全功能：zsh支持更多的自动补全选项，可以根据用户输入的内容智能推断出可能的选项，并提供给用户选择。
3. 更加丰富的插件支持：zsh拥有丰富的插件支持，可以通过安装插件来增强其功能，例如语法高亮、命令历史记录等。
4. 更加灵活的别名和函数定义：zsh支持更加灵活的别名和函数定义，可以方便地自定义命令和函数，提高用户的工作效率。
5. 更加强大的参数扩展功能：zsh支持更加强大的参数扩展功能，可以方便地操作和处理命令行参数，提高用户的工作效率。

(8) Terminal ：终端

终端就是一个命令行接口

PS1：在kali中打开终端(ctrl+alt+t)

PS2: sudo---授权管理工具，默认就是以root用户身份执行，相当于sudo -u root命令

sudo-l: 在执行sudo命令时，默认会对当前用户身份进行验证

查看当前sudo权限(用户可以以所有用户身份在当前主机上运行所有命令(具备完全管理权限))

```
(rootkali)-[/usr/sbin]
# sudo -l
匹配 %2$s 上 %1$s 的默认条目 :
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/
bin\:/sbin\:/bin
用户 root 可以在 kali 上运行以下命令 :
    (ALL : ALL) ALL
    用户 角色
```

PS3: 清屏快捷命令ctrl+l

PS4: cd -: 返回到上次工作目录

cd ~: 返回到家目录

cd ~hy: 回到hy的家目录

Linux黑客基础-08-Kali 之旅

passwd 更改密码

-S 查看用户密码的状态

-l, --lock 锁定指定的帐户

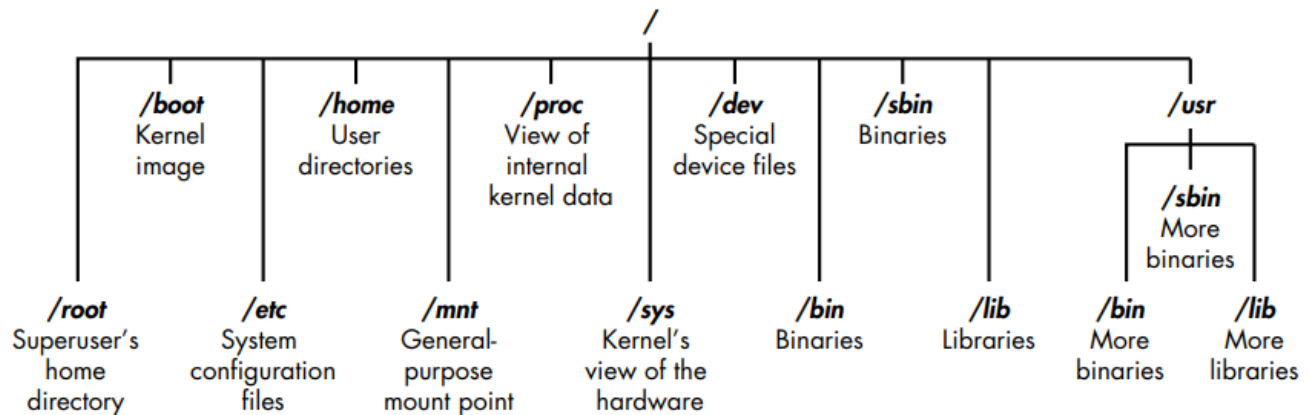
-u, --unlock 解锁被指定帐户

用户密码文件 /etc/shadow

用户账户文件 /etc/passwd

linux文件系统

linux只有一个根；根是整个文件系统的入口(起始位置)；从根开始描述的路径称为绝对路径



/root The home directory of the all-powerful root user

/etc Generally contains the Linux configuration files—files that control when and how programs start up (系统的配置文件)

/home The user's home directory

/mnt Where other filesystems are attached or mounted to the filesystem (挂载点) (mount point)

1. 对于设备的访问需要进行挂载才可以访问
2. 挂载 (mount) 就是把一个设备 (文件系统) 和一个目录关联
3. mount 设备 (文件系统) 目录-----挂载光盘

mount /dev/cdrom /mnt

umount 挂载点-----卸载设备

/media Where CDs and USB devices are usually attached or mounted to the filesystem (CD 和 USB 设备通常连接或安装到文件系统的位置)

/bin Where application binaries (the equivalent of executables in Microsoft Windows) reside (其中包含应用程序二进制文件 (相当于 Microsoft

Windows 中的可执行文件))

/lib Where you'll find libraries (shared programs that are similar to Windows DLLs) (库文件 (与 Windows DLL 类似的共享程序))

/boot 引导文件和内核文件

```
(root👤kali)-[/boot]
# ls
config-5.10.0-kali3-amd64      System.map-5.10.0-kali3-amd64
config-5.14.0-kali2-amd64      System.map-5.14.0-kali2-amd64
grub                          vmlinuz-5.10.0-kali3-amd64
initrd.img-5.10.0-kali3-amd64  vmlinuz-5.14.0-kali2-amd64
initrd.img-5.14.0-kali2-amd64

                                     内核文件

(root👤kali)-[/boot]
# uname -r
5.14.0-kali2-amd64  内核文件版本
```

/home 用户的家目录

/dev 设备文件目录

如: /dev/cdrom 光盘

/dev/sda 第一块scsi接口的硬盘

/dev/sda1 第一块scsi接口的硬盘第一个分区

/proc 虚拟文件系统, 反映的是内核内部文件的数据信息

/sys 内核关于硬件的数据信息

/usr 面向用户级

/usr/bin、/usr/sbin、/usr/lib

PS1:df (查看文件系统)

-T 文件系统类型

-h 以我们适合的容量单位显示

```
(root👾kali)-[/mnt]
# df -T -h
文件系统      类型      容量  已用  可用  已用% 挂载点
udev          devtmpfs   1.9G    0    1.9G    0% /dev
tmpfs         tmpfs      390M   1.2M   389M    1% /run
/dev/sda1     ext4       195G   30G   156G   17% /
tmpfs         tmpfs      2.0G   4.0K   2.0G    1% /dev/shm
tmpfs         tmpfs      5.0M    0    5.0M    0% /run/lock
tmpfs         tmpfs      390M   64K   390M    1% /run/user/0
```

Linux黑客基础-09-基本的Linux命令

1、创建一个自己的工作目录

└──(root👾kali)-[~/桌面]

└──# mkdir -pv work/{doc,app,bak,script}

mkdir: 已创建目录 'work'

mkdir: 已创建目录 'work/doc'-----文档

mkdir: 已创建目录 'work/app'-----应用

mkdir: 已创建目录 'work/bak'-----备份

mkdir: 已创建目录 'work/script'-----脚本

mkdir: 已创建目录 'work/exam'-----练习目录

mkdir 创建目录

用法: mkdir [选项]... 目录...

`-p, --parents` 需要时创建目标目录的上层目录，但即使这些目录已存在也不当作错误处理

`mkdir -p test/{1..100}`-----test目录下创建100个目录

`mkdir {a..d}`-----创建a-d四个目录

`ls -ld work` 查看目录本身的属性

`ls -l work` 查看目录下对象的属性

`rm -rf work` 删除目录

2、Finding Yourself with pwd

查看当前你所在的目录 打印当前工作目录

3、Checking Your Login with whoami

`whoami` 查看当前用户（我是谁）

└──(root👤kali)-[~/桌面/work/exam]

└──# whoami

root

4、Navigating the Linux Filesystem

linux文件系统导航

5、Changing Directories with cd

切换目录

`cd ..`-----返回上一级目录

`cd .`-----当前目录

`cd ./work-----`返回当前目录的work目录

`cd ../..-----`返回上一级目录的上一级目录

6、 Listing the Contents of a Directory with ls

列出目录的内容

`-l` 使用较长格式列出信息

`-a, --all` 不隐藏任何以 `.` 开始的项目

`ls -la` 显示隐藏文件

7、 Getting Help 获取帮助

```
└──(root🐼kali)-[/]
```

```
└─# aircrack-ng --help    无线破解工具
```

```
└──(root🐼kali)-[/]
```

```
└─# nmap -h    著名的网络扫描工具
```

`manual (man)` 查看联机手册

```
└──(root🐼kali)-[/]
```

```
└─# man nmap
```

PS1:相对路径（从当前工作目录的描述的路径）

```
cd work/app
```

PS2:

shell中常用的快捷键

`ctrl+w` 删除光标左侧的单词

`ctrl+A` 回到行首

ctrl+e 回到行尾

Linux黑客基础-10-如何在Linux系统下查找文件

1、Searching with locate 使用 locate 搜索

根据关键字在整个linux文件系统中查找（整个）相关文件

Usage: plocate [OPTION]... PATTERN...

- (1) 基于自己的数据库对文件进行查找
- (2) 会把匹配的文件全部给出
- (3) 使用locate之前可以使用updatedb更新数据库

(/var/lib/mlocate/mlocate.db)

```
└──(root🐼kali)-[~/桌面]
```

```
└─# locate aircrack-ng
```

```
└──(root🐼kali)-[~/桌面]
```

```
└─# locate password | more
```

```
└──(root🐼kali)-[~/桌面/work/exam]
```

```
└─# updatedb
```

但是，locate 命令并不完美。有时，定位的结果可能是压倒性的，给你太多的信息。此外，locate 使用的数据库通常每天只更新一次，因此如果您刚刚在几分钟或几小时前创建了一个文件，它可能会在第二 28 天才显示在此列表中。了解这些基本命令的缺点，以便您可以更好地决定何时最好使用每个命令。

2、Finding Binaries with whereis 用 whereis 查找二进制文件

```
└──(root🐼kali)-[~/桌面/work/exam]
```

```
└─# whereis sqlmap
```

```
sqlmap: /usr/bin/sqlmap /etc/sqlmap /usr/share/sqlmap  
/usr/share/man/man1/sqlmap.1.gz
```

3、 Finding Binaries in the PATH Variable with which 用 which 在 PATH 变量中查找二进制文件

```
└──(root🐼kali)-[~/桌面/work/exam]  
└─# which sqlmap  
  
/usr/bin/sqlmap
```

在这里，它能够在 PATH 变量中列出的目录中找到单个二进制文件。至少，这些目录通常包含 /usr/bin， 但可能包括 /usr/sbin 以及其他一些目录。

PATH变量：

1、 路径搜索变量

2、 查看变量： echo \$变量名

```
└──(root🐼kali)-[~/桌面/work/exam]  
└─# echo $PATH
```

```
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/local/games:  
/usr/games
```

3、 路径之间用:分割

4、 作用： 当执行一个（外部） 命令程序时会在path变量中查找有无这个程序

type： 判定命令程序是内部命令还是外部命令

4、 Performing More Powerful Searches with find 使用 find 执行更强大的搜索

查找的条件： 文件的类型、名称、属组、大小、创建（或修改）时间、权限等

基本语法： find directory options expression

find 路径 选项 表达式

例: find /^❶ -type f^❷ -name apache2^❸

-type 文件的类型, f表示普通文件, d表示目录

-name 文件的名称, 可以支持通配符

```
└──(root@kali)~[~/桌面/work/exam]
```

```
└─# find / -type f -name apache2
```

First I state the directory in which to start the search, in this case / .

Then I specify which type of file to search for, in this case f for an ordinary file . Last, I give the name of the file I'm searching for, in this case apache2 .

首先, ❶是说明了开始搜索的目录, 在这种情况下是 / 。然后❷我指定要搜索的文件类型, 在本例中 为 f 表示普通文件。最后, ❸我给出了我正在搜索的文件的名称, 在本例中为 apache2。

```
└──(root@kali)~[~/桌面/work/exam]
```

```
└─# find /etc -type f -name apache2
```

The find command started at the top of the filesystem (/), went through every directory looking for apache2 in the filename, and then listed all instances found.

find 命令从文件系统 (/) 的顶部开始, 遍历每个目录, 在文件名中查找 apache2, 然后列出找到的所有 实例。 └──(root@kali)~[~/桌面/work/exam]

```
└─# find /etc -type f -name apache2.*
```

```
1 x
```

```
/etc/apache2/apache2.conf
```

This much quicker search only found occurrences of apache2 in the /etc directory and its subdirectories. It's also important to note that unlike some

other search commands, find displays only exact name matches. If the 12 Chapter 1 file apache2 has an extension, such as apache2.conf, the search will not find a match. We can remedy this limitation by using wildcards, which enable us to match multiple characters. Wildcards come in a few different forms: *, ., ? and [].

这个更快的搜索只在/etc 目录及其子目录中发现了 apache2 的出现。同样重要的是，要注意与其他一些搜索命令不同，find 只显示确切的名称匹配。如果文件 apache2 有扩展名，例如 apache2.conf，则搜索找不到匹配项。我们可以通过使用通配符来解决此限制，这使我们能够匹配多个字符。通配符有几种不同的形式：*, ., ? 和 []。

Wildcards come in a few different forms: *, ., ? and []. 常用的通配符

*: 任意 (0个或多个) 字符 ? : 单个字符 []:列表

练习:

```
└──(root@kali)─[~/桌面/work/exam]
```

```
└─# touch a1 a2 a3 a4 a5 a100
```

创建文件夹

```
└──(root@kali)─[~/桌面/work/exam]
```

```
└─# ls
```

```
a a1 a100 a2 a3 a4 a5 b c d hh tes
```

查看

```
└──(root@kali)─[~/桌面/work/exam]
```

```
└─# find . -name "a?"
```


./a5

./a4

./a2

./a1

./a3

寻找a1-a5

```
└──(root👤kali)-[~/桌面/work/exam]
```

```
└─# find . -name "a[1,3,5]"
```

./a5

./a1

./a3

只寻找a1,a3,a5

Linux黑客基础-11-快速理解通配符

A Quick Look at Wildcard s(快速理解通配符)

常用的通配符

*: 任意 (0个或多个) 字符

? : 单个字符

[:列表

```
└──(root👤kali)-[~/桌面/work/exam]
```

```
└─# mkdir day
```

```
└──(root👤kali)-[~/桌面/work/exam]
```

```
└─# cd day
```

```
└─(root👁kali)-[~/桌面/work/exam/day]
```

```
└─# touch cat hat what bat
```

```
└─(root👁kali)-[~/桌面/work/exam/day]
```

```
└─# ls
```

```
bat cat hat what
```

```
└─(root👁kali)-[~/桌面/work/exam/day]
```

```
└─# find . -name "?at"
```

```
./hat
```

```
./cat
```

```
./bat
```

```
└─(root👁kali)-[~/桌面/work/exam/day]
```

```
└─# ls ?at
```

```
bat cat hat
```

```
└─(root👁kali)-[~/桌面/work/exam/day]
```

```
└─# find . -name "[bc]at"
```

```
./cat
```

```
./bat
```

```
└─(root👁kali)-[~/桌面/work/exam/day]
```

```
└─# find . -name "[b-c]at"
```

```
./cat
```

```
./bat
```

```
└──(root👤kali)-[~/桌面/work/exam/day]
```

```
└─# find . -name "[b,c]at"
```

```
./cat
```

```
./bat
```

```
└──(root👤kali)-[~/桌面/work/exam/day]
```

```
└─# find . -name "[b,h]at"
```

```
./hat
```

```
./bat
```

which matches any character(s) of any length, from none to an unlimited number of characters. 最常用的通配符是星号 (*), 它匹配任何长度的任何字符, 从无字符到无限数量的字符

```
└──(root👤kali)-[~/桌面/work/exam/day]
```

```
└─# find . -name "*at"
```

```
./hat
```

```
./what
```

```
./cat
```

```
./bat
```

Linux黑客基础-12-使用grep实现过滤

在每个<文件>中查找给定<模式>。也可以对来自另一个命令的输出做过滤

例如: `grep -i 'hello world' menu.h main.c`

<模式>可以包括多个模式字符串, 使用换行符进行分隔。

(1) 常与管道符 (|) 结合在一起

```
(root@kali)~[~/桌面/work/exam/day]
```

```
# ls /usr/bin | grep zip
```

(2) 常用的命令选项

用法: `grep [选项]... 模式 [文件]...`

`-r, --recursive` 等同于 `--directories=recurse` (递归)

`-R, --dereference-recursive` 同上, 但遍历所有符号链接

```
ls -R /usr | grep zip
```

(3) `-i` 忽略大小写

```
(root@kali)~[~/桌面/work/exam/day]
```

```
# grep -r -i hy 01
```

PS1:管道符 (`|`) piping

把前一个命令的输出当作后一个命令的输入, 常用于连接多个命令

PS2:查看端口

```
(root@kali)~[~/桌面/work/exam/day]
```

```
# netstat -tunlp
```

`-t:tcp` `-u:udp` `-n:数字形式显示` `-l:显示当前正在监听的端口` `-p:`

查看哪个进程在使用该端口

```
(root@kali)~[~/桌面/work/exam/day]
# netstat -tunlp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      769/sshd: /usr/sbin
tcp        0      0 127.0.0.1:45432         0.0.0.0:*               LISTEN      831/postgres
tcp        0      0 0.0.0.0:13443          0.0.0.0:*               LISTEN      851/opsrv
tcp6       0      0 :::22                  :::*                   LISTEN      769/sshd: /usr/sbin
tcp6       0      0 :::1:45432            :::*                   LISTEN      831/postgres
udp        0      0 0.0.0.0:68             0.0.0.0:*               560/dhclient
```

PS3: web服务程序---apache2

PS4:ps aux

```
(root@kali)~[~/桌面/work/exam/day]
```

└─# ps aux | grep apache2

```
root      4134  0.0  0.5 194652 20392 ?        Ss   16:18   0:00
/usr/sbin/apache2 -k start

www-data  4137  0.0  0.3 195236 12172 ?        S    16:18   0:00
/usr/sbin/apache2 -k start

www-data  4138  0.0  0.2 195188 10196 ?        S    16:18   0:00
/usr/sbin/apache2 -k start

www-data  4139  0.0  0.2 195172 10196 ?        S    16:18   0:00
/usr/sbin/apache2 -k start

www-data  4140  0.0  0.2 195172 10196 ?        S    16:18   0:00
/usr/sbin/apache2 -k start

www-data  4141  0.0  0.2 195172 10196 ?        S    16:18   0:00
/usr/sbin/apache2 -k start

www-data  4146  0.0  0.2 195172 10196 ?        S    16:19   0:00
/usr/sbin/apache2 -k start

root      4155  0.0  0.0   6412   2196 pts/0    S+   16:20   0:00 grep --
color=auto apache2
```

PS5: systemctl 系统服务控制工具

systemctl [OPTIONS...] COMMAND ...

systemctl 命令 服务名称

start UNIT... Start (activate) one or more units 启动

stop UNIT... Stop (deactivate) one or more units

停止

restart UNIT... Start or restart one or more units 重启

└─(root🐼kali)-[~/桌面/work/exam/day]

```
└─# systemctl start apache2
```

PS6: echo \$? 查看命令的执行状态

PS7: SSH服务（常用于远程连接（安全））

```
└─(root👤kali)-[~/桌面/work/exam/day]
```

```
└─# ps aux | grep ssh
```

```
1 x
```

```
root      769  0.0  0.1 13516  7116 ?        Ss   11:55   0:00 sshd:
/usr/sbin/sshd -D [listener] 0 of 10-100 startups
```

```
root      1258  0.0  0.0  5964   468 ?        Ss   11:57   0:00 /usr/bin/ssh-
agent /usr/bin/im-launch x-session-manager
```

```
root      4211  0.0  0.0  6544  2276 pts/0    S+   16:26   0:00 grep --
color=auto ssh
```

```
└─(root👤kali)-[~/桌面/work/exam/day]
```

```
└─# netstat -tunlp | grep 22
```

```
tcp        0      0 0.0.0.0:22          0.0.0.0:*           LISTEN
769/sshd: /usr/sbin
```

```
tcp6       0      0 :::22              :::*                LISTEN      769/sshd:
/usr/sbin
```

PS8: !n（调用最近一次以n打头的命令）

```
└─(root👤kali)-[~/桌面/work/exam/day]
```

```
└─# !n
```

PS9: pstree | grep apache 查看服务进程号

```
└─(root👤kali)-[~/桌面/work/exam/day]
```

```
└─# pstree | grep apache
```

|—apache2---6*[apache2]

练习：递归练习

|—(root👤kali)—[~/桌面/work/exam/day]

|—# mkdir -pv 01/001/0001/00001

mkdir: 已创建目录 '01'

mkdir: 已创建目录 '01/001'

mkdir: 已创建目录 '01/001/0001'

mkdir: 已创建目录 '01/001/0001/00001'

|—(root👤kali)—[~/桌面/work/exam/day]

|—# echo hyisgood >01test.txt

|—(root👤kali)—[~/桌面/work/exam/day]

|—# echo hyisgood >01test1.txt

|—(root👤kali)—[~/桌面/work/exam/day]

|—# echo hyisgood >01/001test01.txt

|—(root👤kali)—[~/桌面/work/exam/day]

|—# grep -r hy 01

01/001test01.txt:hyisgood

01/001/0001test0001.txt:hyisgood

Linux黑客基础-13-systemctl服务控制工具基本使用

systemctl 系统服务控制工具

systemctl [OPTIONS...] COMMAND ...

systemctl 命令 服务名称

start UNIT... Start (activate) one or more units 启动

stop UNIT... Stop (deactivate) one or more units

停止

restart UNIT... Start or restart one or more units 重启

enable [UNIT...|PATH...] Enable one or more unit files 开机自

启动

disable UNIT... Disable one or more unit files 开机不

启动

is-enabled UNIT... Check whether unit files are enabled

查看是否为开机自启动

status [PATTERN...|PID...] Show runtime status of one or

more units 查看服务状态

```
(root@kali) ~ | ~/桌面/work/exam/day |
# systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; disabled; vendor preset: disabled)
   Active: active (running) since Thu 2023-03-30 16:18:35 CST; 41min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 4123 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
    Main PID: 4134 (apache2)
      Tasks: 7 (limit: 4578)
     Memory: 18.4M
        CPU: 413ms
    CGroup: /system.slice/apache2.service
            └─4134 /usr/sbin/apache2 -k start
              4137 /usr/sbin/apache2 -k start
              4138 /usr/sbin/apache2 -k start
              4139 /usr/sbin/apache2 -k start
              4140 /usr/sbin/apache2 -k start
              4141 /usr/sbin/apache2 -k start
              4146 /usr/sbin/apache2 -k start

3月 30 16:18:34 kali systemd[1]: Starting The Apache HTTP Server...
3月 30 16:18:35 kali apachectl[4133]: AH00558: apache2: Could not reliably determine the server's fully qualif>
3月 30 16:18:35 kali systemd[1]: Started The Apache HTTP Server.
```


Linux黑客基础-14-文件的基本操作01-使用cat创建文件

1、Creating Files

(1) cat

concatenate（连接）的缩写，即combine pieces together

1、把碎片组合在一起）意味着可以使用cat创建一个小文件

2、显示文件

```
└──(root👤kali)-[~/桌面/work/exam]
```

```
└─# cat > hackingskill
```

```
1 🌀
```

what your name

my name is hy. ctrl+d退出

```
└──(root👤kali)-[~/桌面/work/exam]
```

```
└─# cat >> hackingskill      追加数据
```

```
└──(root👤kali)-[~/桌面/work/exam]
```

```
└─# cat > hackingskill
```

```
1 🌀
```

heyuanheyuan 覆盖原文件

2、File Creation with touch 使用touch创建文件

3、vi创建文件

PS1:重定向 >覆盖 >>追加

PS2: bash小技巧 esc+. 或!\$ 调用上一个命令的参数

PS3: 在脚本中使用cat创建文件

我的第一个脚本：#!/bin/bash 第一行

#!/:告诉系统脚本将会使用哪个解释器

bash：是常用的一种shell解释器

我的第一个脚本

#!/bin/bash

echo "hello,hacking"

PS4：vi基本操作

1、当使用vi编辑一个文件时，默认进入的模式是命令模式

2、由默认模式进入插入模式（输入、编辑等）

i (insert, 插入)

a () 也可以插入

3、由插入模式返回命令模式

esc

4、保存退出，进入到末行模式

： 进入底行模式；执行wq保存退出；q! 强制退出不保存

Linux黑客基础-15-文件的基本操作02-在脚本中使用cat创建一个文件

└──(root🐼kali)-[~/桌面/work/exam/work01]

└─# ./ex01.sh

1 🌀

hello,hacking 执行脚本的方法

└──(root🐼kali)-[~/桌面/work/exam/work01]

└─# vi cat_ex_01.sh

▼ 创建文件

Plain Text |

```
1  #!/bin/bash
2  echo "this is user cat create file example"
3  cat > catfile <<EOF
4  i am a hacker
5  hahahha
6  lisijia
7  hahahahaha
8  I love you
9  EOF
10 ls catfile
11 cat -n catfile
```

编写脚本

└─(root👤kali)-[~/桌面/work/exam/work01]

└─# chmod 777 cat_ex_01.sh 赋予权限

└─(root👤kali)-[~/桌面/work/exam/work01]

└─# ./cat_ex_01.sh

1 ⚙

this is user cat create file example

catfile

1 i am a hacker

2 hahahha

3 lisijia

4 hahahahaha

5 I love you 执行脚本

Linux黑客基础-16-文件的基本操作03-创建.删除.移动.复制.重命名

File Creation with touch 使用touch创建一个空文件

The second command for file creation is touch. This command was originally developed so a user could simply touch a file to change some of its details, such as the date it was created or modified. However, if the file doesn't already exist, this command creates that file by default. 文件创建的第二个命令是 touch。此命令最初开发是用来用户只需 touch 文件即可更改其某些详细信息，例如创建或修改日期。但是，如果该文件尚不存在，则此命令默认情况下会创建该文件。

Creating a Directory 创建目录

The command for creating a directory in Linux is mkdir, a contraction of make directory. To create a directory named newdirectory, enter the following command: 在 Linux 中创建目录的命令是 mkdir，它是创建目录 (make directory)的缩写。要创建名为 newdirectory 的目录，请输入以下命令：

Copying a File 复制文件

To copy files, we use the cp command. This creates a duplicate of the file in the new location and leaves the old one in place. Here, we'll create the file oldfile in the root directory with touch and copy it to /root/newdirectory, renaming it in the process and leaving the original oldfile in place: 要复制文件，我们使用 cp 命令。这会在新的目录位置创建文件的副本，并保留旧文件。在这里，我们将使用 touch 在根目录中创建一个文件 oldfile，并将其复制到/root/newdirectory，在进程中重命名并保留原始 oldfile：

Renaming the file is optional and is done simply by adding the name you want to give it to the end of the directory path. If you don't rename the file when you copy it, the file will retain the original name by default. 重命名文件

是可选的，只需将您想要的名称添加到目录路径的末尾即可。如果在复制文件时不重命名 该文件，则默认情况下该文件将保留原始名称。

Renaming a File 重命名文件

Unfortunately, Linux doesn't have a command intended solely for renaming a file, as Windows and some other operating systems do, but it does have the mv (move) command. 不幸的是，Linux 没有专门用于重命名文件的命令（如 Windows 和其他一些操作系统那样），但它确实 有 mv (move) 命令。

Removing a File 删除文件

To remove a file, you can simply use the rm command, like so:

Removing a Directory 删除目录

The command for removing a directory is similar to the rm command for removing files but with dir (for directory) appended, like so:

```
kali >rmkdir newdirectory
```

```
rmkdir:failed to remove 'newdirectory': Directory not empty
```

It's important to note that rmdir will not remove a directory that is not empty, but will give you a warning message that the “directory is not empty,”

as you can see in this example. You must first remove all the contents of the directory before removing it. This is to stop you from accidentally deleting objects you didn't intend to delete.

If you do want to remove a directory and its content all in one go, you can use the -r switch after rm, like so:

```
kali >rm -r newdirectory
```

Just a word of caution, though: be wary of using the -r option with rm,

at least at first, because it's very easy to remove valuable files and directories

by mistake. Using `rm -r` in your home directory, for instance, would delete every file and directory there—probably not what you were intending.

删除目录的命令类似于删除文件的 `rm` 命令，但附加了 `dir`（用于目录），如下所示：

```
kali > rmdir newdirectory
```

```
rmdir:failed to remove 'newdirectory': Directory not empty
```

重要的是要注意 `rmdir` 不会删除非空的目录，但会给你一条警告信息“目录不为空”(Directory not

empty)，如本例所示。在删除目录之前，必须先删除该目录下的所有内容。这是为了阻止您意外删除您不

想删除的对象文件。

如果你想一次性删除一个目录及其下的内容，你可以在 `rm` 之后使用 `-r` 参数，如下所示：

```
kali > rm -r newdirectory
```

34

但需要注意的是：要小心使用 `-r` 选项和 `rm`，至少在开始时，因为错误地删除有价值的文件和目录非常

容易。例如，在主目录（`/`）中使用 `rm -r` 会删除那里的每一个文件和目录 — 这不是你想要的致命性的动

作。

Linux黑客基础-17-文件的基本操作04-小测试

1. Use the ls command from the root (/) directory to explore the directory structure of Linux. Move to each of the directories with the cd command and run pwd to verify where you are in the directory structure.

```
└──(root👤kali)~[~/桌面/work/exam/work01]
```

```
└──# ls /
```

```
bin  dev  home      initrd.img.old  lib32  libx32  media  opt  root /sbin  sys  usr  vmlinuz
```

```
boot  etc  initrd.img  lib          lib64  lost+found  mnt   proc  run  srv  tmp  var  vmlinuz.old
```

2. Use the whoami command to verify which user you are logged in as.

```
└──(root👤kali)~[~/桌面/work/exam/work01]
```

```
└──# whoami
```

```
root
```

3. Use the locate command to find wordlists that can be used for password cracking.

```
└──(root👤kali)~[~/桌面/work/exam/work01]
```

```
└──# cd /usr/share/wordlists
```

```
└──(root👤kali)~[~/桌面/work/exam/work01]
```

```
└──# ls
```

```
dirb  dirbuster  fasttrack.txt  fern-wifi  metasploit  nmap.lst  rockyou.txt.gz  wfuzz
```

```
└──(root👤kali)~[~/桌面/work/exam/work01]
```

```
└──# locate wordlists
```

4. Use the cat command to create a new file and then append to that file. Keep in mind that > redirects input to a file and >> appends to a file.

```
└──(root👤kali)~[~/桌面/work/exam/work01]
```

```
└──# cat > f1
```

```
file cat
```

```
now ok
```

```
└──(root👤kali)~[~/桌面/work/exam/work01]
```

```
└──# cat >> f1
```

hello

```
└──(root👤kali)-[/]
```

```
└─# cat f1
```

file cat

now okhello

5. Create a new directory called hackerdirectory and create a new file in that directory named hackedfile. Now copy that file to your /root directory and rename it secretfile

```
└──(root👤kali)-[~/桌面/work/exam]
```

```
└─# mkdir hackerdirectory
```

```
└──(root👤kali)-[~/桌面/work/exam]
```

```
└─# cd hackerdirectory
```

```
└──(root👤kali)-[~/桌面/work/exam/hackerdirectory]
```

```
└─# touch hackedfile
```

```
└──(root👤kali)-[~/桌面/work/exam/hackerdirectory]
```

```
└─# cp hackedfile /root/secretfile
```

常用文件类型

l----软链接（link）文件（类似于windows中的快捷方式）

d---目录

-- 普通文件