实验十三：使用 Firewall 保护系统

一、实验目的

1、理解防火墙服务；

2、掌握防火墙规则的设计与配置；

3、掌握通过防火墙对业务进行安全防护。

二、实验学时

2 学时

三、实验类型

设计性

四、实验需求

1、硬件

每人配备计算机 1 台。

2、软件

Windows 操作系统，安装 Oracle VM VirtualBox 软件，安装 Mobaxerm 软件。

3、网络

本地主机与虚拟机能够访问互联网，不使用 DHCP 服务。

4、工具

无。

五、实验任务

1、完成防火墙服务的管理；

2、完成防火墙日志的配置；

3、完成使用防火墙提升服务的安全性。

六、实验环境

1、本实验需要 VM 1 台；

2、本实验 VM 配置信息如下表所示；

虚拟机配置	操作系统配置
虚拟机名称：VM-Lab-13-Task-01-172.20.1.26 内存：1GB CPU：1 颗，1 核心	主机名：Lab-13-Task-01 IP 地址：172.20.1.26 子网掩码：255.255.255.0

虚拟磁盘：20GB 网卡：1 块，桥接	网关：172.20.1.1 DNS：8.8.8.8

3、本实验拓扑图。

无

4、本实验操作演示视频。

无

七、实验内容及步骤

1、管理防火墙服务

对防火墙服务的管理包括查看防火墙 Firewalld 服务状态、开启、关闭、重启、重新载入防火墙策略等。

```
# 查看防火墙 Firewalld 服务状态
[root@Lab-13-Task-01 ~]# systemctl status firewalld

# 关闭防火墙服务
[root@Lab-13-Task-01 ~]# systemctl stop firewalld

# 开启防火墙服务
[root@Lab-13-Task-01 ~]# systemctl start firewalld

# 重启防火墙服务
[root@Lab-13-Task-01 ~]# systemctl restart firewalld

# 设置防火墙为开机不自启
[root@Lab-13-Task-01 ~]# systemctl disable firewalld

# 设置防火墙为开机自启动
[root@Lab-13-Task-01 ~]# systemctl enable firewalld

# 重新载入防火墙规则
[root@Lab-13-Task-01 ~]# firewall-cmd --reload
```

2、配置防火墙日志策略

对防火墙日志的配置有全局日志配置和规则日志配置两部分。全局日志配置是对防火墙日志规则进行配置，防火墙日志服务由系统 rsyslog 服务进行管理，日志默认存放在/var/log/firewalld 日志文件中，日志文件基于日期时间自动归档。规则日志配置是设置防火墙触发特定防火墙规则时记录日志的方式。

2.1 全局日志配置

本步骤通过修改防火墙与 rsyslog 服务的配置文件，对防火墙日志字段、日志文件存放路径、日志文件分割方法等进行自定义配置，完成对防火墙全局日志的配置，实现以下 3 个目标：

（1）实现防火墙对单播网络通信的日志记录。

（2）防火墙日志存放目录变更为/var/log/firewalldlog。

（3）防火墙日志记录等级调整为所有等级的日志均记录。

```
# 使用 vi 命令编辑/etc/firewalld/firewalld.conf 文件
[root@Lab-13-Task-01 ~]# vi /etc/firewalld/firewalld.conf
# ----------/etc/firewalld/firewalld.conf 文件----------
# firewalld.conf 配置文件内容较多，本部分仅显示与防火墙日志配置有关的内容
# 将 LogDenied=off 改为 LogDenied=unicast，实现对单播网络通信的日志记录
LogDenied=unicast
# ----------/etc/firewalld/firewalld.conf 文件----------

# 使用 vi 命令编辑/etc/rsyslog.conf 文件
[root@Lab-13-Task-01 ~]# vi /etc/rsyslog.conf
# ----------/etc/rsyslog.conf 文件----------
# 在配置文件中增加以下内容，kern.*表示记录所有等级的系统内核产生的日志信息
kern.*                                                  /var/log/firewalldlog/loginfo
# ----------/etc/rsyslog.conf 文件----------

# 创建防火墙日志存放目录
[root@Lab-13-Task-01 ~]# mkdir /var/log/firewalldlog
# 重新载入配置文件
[root@Lab-13-Task-01 ~]# systemctl reload firewalld
# 重启日志相关服务
[root@Lab-13-Task-01 ~]# systemctl restart rsyslog
```

2.2 规则日志配置

在配置防火墙规则时，可定义由该规则产生的日志的记录方式。本步骤新增一条防火墙规则并实现下述 3 个目标：

（1）允许本地主机（172.20.1.134）通过 httpd 服务访问服务器。

（2）实现触发规则的通信的日志记录。

（3）设置日志记录的频率为每秒最多 3 条。

```
# 根据防火墙规则要求配置
[root@Lab-13-Task-01 ~]# firewall-cmd --permanent --add-rich-rule='rule family=ip
v4 source address=172.20.1.134 service name="http" log level=notice prefix="HTT
P" limit value="3/s" accept'

# 重新载入防火墙配置使其生效
[root@Lab-13-Task-01 ~]# systemctl reload firewalld
```

3、使用 firewall-cmd 工具管理防火墙策略

案例 1：指定端口和协议允许访问

需求描述：

（1）打开 443/TCP 端口。

（2）永久打开 3690/TCP 端口。

（3）永久打开 100-500/TCP 端口（指定范围内端口全部打开）。

配置方法：

```
# 打开 443/TCP 端口
[root@Lab-13-Task-01 ~]# firewall-cmd --add-port=443/tcp

# 永久打开 3690/TCP 端口
[root@Lab-13-Task-01 ~]# firewall-cmd --add-port=3690/tcp --permanent

# 永久打开 100-500/TCP 端口（指定范围内端口全部打开）
[root@Lab-13-Task-01 ~]# firewall-cmd --remove-port=100-500/tcp --permanent

# 重新载入防火墙配置
[root@Lab-13-Task-01 ~]# firewall-cmd --reload
```

案例 2：指定服务允许/禁止访问

需求描述：

（1）允许访问本机的 http、https 服务。

（2）允许访问本机的 zabbix-server 服务。

（3）禁止访问本机的 cockpit、dhcpv6-client 服务。

（4）启用 SYN、ICMP 洪泛攻击保护。

配置方法：

```
# 允许访问本机的 http、https 服务
[root@Lab-13-Task-01 ~]# firewall-cmd --permanent --add-service=http
[root@Lab-13-Task-01 ~]# firewall-cmd --permanent --add-service=https

# 允许访问本机的 zabbix-server 服务
[root@Lab-13-Task-01 ~]# firewall-cmd --permanent --add-service=zabbix-server

# 禁止访问本机的 cockpit、dhcpv6-client 服务
[root@Lab-13-Task-01 ~]# firewall-cmd --permanent --remove-service=cockpit
[root@Lab-13-Task-01 ~]# firewall-cmd --permanent --remove-service=dhcpv6-clie
nt

# 启用 SYN、ICMP 洪泛攻击保护
[root@Lab-13-Task-01 ~]# firewall-cmd --permanent --add-service=syn-flood
[root@Lab-13-Task-01 ~]# firewall-cmd --permanent --add-service=icmp-flood

# 重新载入防火墙配置
[root@Lab-13-Task-01 ~]# firewall-cmd --reload
```

案例 3：指定 IP 地址允许/禁止访问

需求描述：

（1）允许来自 IP 地址为 172.20.1.134 的主机的流量通过防火墙。

（2）禁止来自 IP 地址为 172.20.1.135 的主机的流量通过防火墙。

配置方法：

```
# 允许来自 IP 地址为 172.20.1.134 的主机的流量通过防火墙
[root@Lab-13-Task-01 ~]# firewall-cmd --add-source=172.20.1.134 --permanent

# 禁止来自 IP 地址为 172.20.1.135 的主机的流量通过防火墙
[root@Lab-13-Task-01 ~]# firewall-cmd --remove-source=172.20.1.135 --permanent

# 重新载入防火墙配置
[root@Lab-13-Task-01 ~]# firewall-cmd --reload
```

案例 4：远程管理服务安全性提升

需求描述：

（1）允许地址范围 172.20.1.134/24 内的客户端远程连接服务器，进行远程管理维护。

（2）客户端远程连接服务器时，每分钟最多允许 5 次远程连接，禁止频繁请求。

配置方法：

```
# 使用 firewall-cmd 命令删除默认 ssh 服务规则
[root@Lab-13-Task-01 ~]# firewall-cmd --permanent --remove-service=ssh

# 使用 firewall-cmd 命令添加指定地址能够远程访问的规则
[root@Lab-13-Task-01 ~]# firewall-cmd --permanent --add-rich-rule='rule family=ip
v4 source address=172.20.1.134/24 service name="ssh" limit value="5/s" accept'

# 重新载入防火墙配置
[root@Lab-13-Task-01 ~]# firewall-cmd --reload
```

案例 5：数据库服务安全性提升

需求描述：

（1）本地客户端（172.20.1.134）能够使用 MySQL WorkBench 连接 MariaDB 数据库。

（2）本地客户端（172.20.1.134）能够通过浏览器访问 phpMyAdmin 管理界面，进行数据库管理。

配置方法：

```
# 使用 firewall-cmd 命令添加本地客户端允许远程连接数据库
[root@Lab-13-Task-01 ~]# firewall-cmd --permanent --add-rich-rule='rule family=ip
v4 source address=172.20.1.134 port port=3306 protocol=tcp accept'

# 使用 firewall-cmd 命令添加本地客户端允许访问 phpMyAdmin
[root@Lab-13-Task-01 ~]# firewall-cmd --permanent --add-rich-rule='rule family=ip
v4 source address=172.20.1.134 port port=80 protocol=tcp accept'

# 重新载入防火墙配置
[root@Lab-13-Task-01 ~]# firewall-cmd --reload
```

案例 6：文件传输服务安全性提升

需求描述：

（1）允许地址范围 172.20.1.134/24 内的客户端通过主动与被动模式访问 FTP 服务器。

配置方法：

```
# 使用 firewall-cmd 命令添加通过主动模式访问 FTP 服务器
[root@Lab-13-Task-01 ~]# firewall-cmd --permanent --add-rich-rule='rule family=ipv4 source address=172.20.1.134/24 port port=20-21 protocol=tcp limit value="10/m" accept'

# 使用 firewall-cmd 命令添加本地客户端允许访问 phpMyAdmin
[root@Lab-13-Task-01 ~]# firewall-cmd --permanent --add-rich-rule='rule family=ipv4 source address=172.20.1.134/24 port port=9000-9020 protocol=tcp limit value="10/m" accept'

# 重新载入防火墙配置
[root@Lab-13-Task-01 ~]# firewall-cmd --reload
```