## 11. 使用sudo

| | | | |
|---|---|---|---|
| **笔记本:** | 优秀笔记 | | |
| **创建时间:** | 2018/12/24 16:07 | **更新时间:** | 2018/12/25 19:56 |
| **作者:** | 306798658@qq.com | | |

# 1：sudo：授权命令

**什么是sudo?**
sudo命令用来以其他身份来执行命令，预设的身份为root。

在/etc/sudoers中设置了可执行sudo指令的用户。若其它未经授权的用户企图使用sudo，则会发出警告的邮件给管理员。用户使用sudo时，必须先输入密码，之后有5分钟的有效期限，超过期限则必须重新输入密码。

**sudo配置文件**
/etc/sudoers文件为sudo的配置文件，可以设置用户能以什么身份执行哪些命令

**visudo命令:**
使用visudo命令来配置sudoers文件，保存时会自动检查是否有语法错误。

执行visudo命令后，找到:

```
root    ALL=(ALL)       ALL
```

然后在找到的内容下面添加以下内容:

```
user1   ALL=(ALL)   ALL
```

添加这一行的意思是user1用户可以在所有终端上以root权限运行所有命令。

添加后先按ESC键，然后输入 “ :wq ” 保存退出。再切换到user1用户，看看是否可以用sudo命令执行一些root用户才可以执行的操作。

```
[user1@long01 ~]$ sudo mkdir /etc/test2
[sudo] user1 的密码:                          #使用sudo命令需要输入当前用户的密码。
[user1@long01 ~]$ ls -dl /etc/test2           #查看刚才创建的test2目录。其属主和属组都是root。
drwxr-xr-x 2 root root 6 5月  18 17:51 /etc/test2
[user1@long01 ~]$ rm -r /etc/test2            #普通用户不能删除/etc/目录下的文件或目录。
rm: 是否删除有写保护的目录 "/etc/test2"? y
rm: 无法删除"/etc/test2": 权限不够
[user1@long01 ~]$ sudo rm -r /etc/test2       #使用sudo授权后就可以正常删除了。
[user1@long01 ~]$ ls -dl /etc/test2
ls: 无法访问/etc/test2: 没有那个文件或目录
```

**sudoers文件配置格式说明:**

配置格式：用户名　ALL=(ALL)　[NOPASSWD:] Command

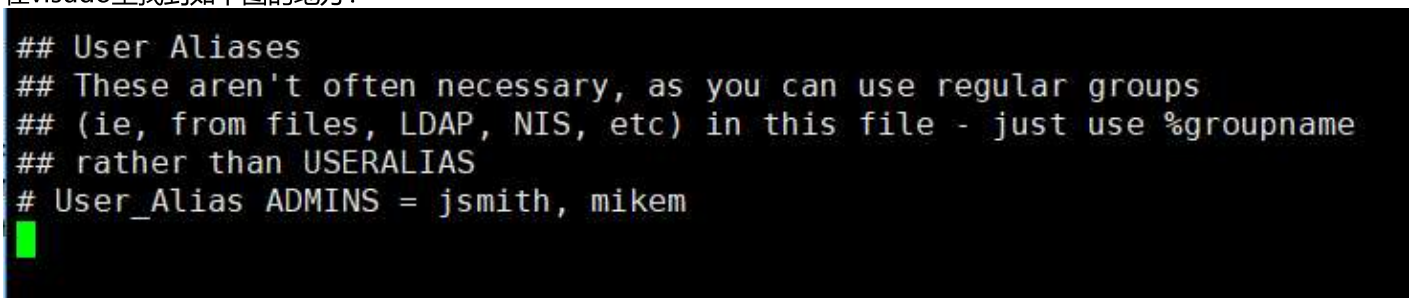
用户名：授予哪个用户或用户别名（多个用户建立一个别名，User_Alias）
ALL=(ALL): 其中等号左边的ALL表示主机IP或者主机名,一般都为ALL，等号右边的ALL为前面的用户名将授予那个用户的权限，ALL表示root或者说是所有用户。

NOPASSWD： 添加这个字符串后，用户使用sudo授权的时候不需输入用户密码。

Command： 表示什么命令可以使用sudo授权。命令必须以绝对路径表示。多个命令使用逗号隔开。当Command是ALL的时候，表示所有命令都可以使用sudo授权。还可以配置命令组别名。

**用户别名配置:**

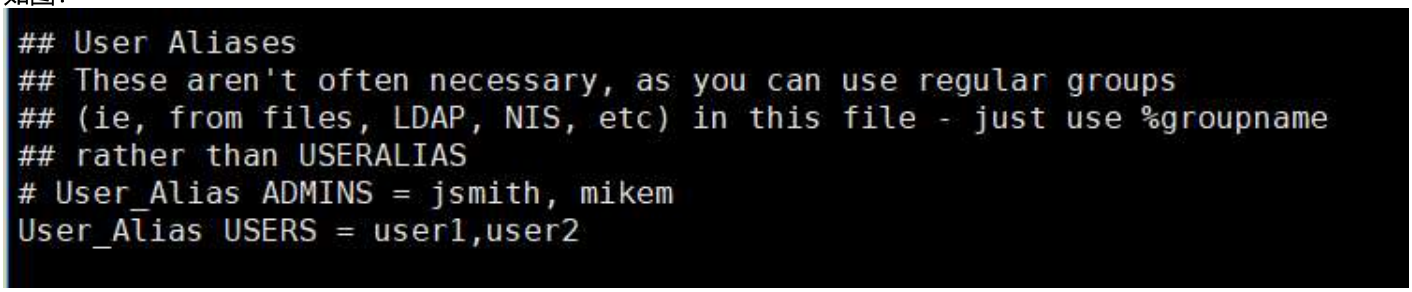
在visudo里找到如下图的地方：




配置格式：User_Alias 别名 = 用户名1，用户名2，用户名3... 在图中位置的下方添加以下内容：

```
User_Alias USERS = user1,user2
```

如图：




**命令组别名配置：**

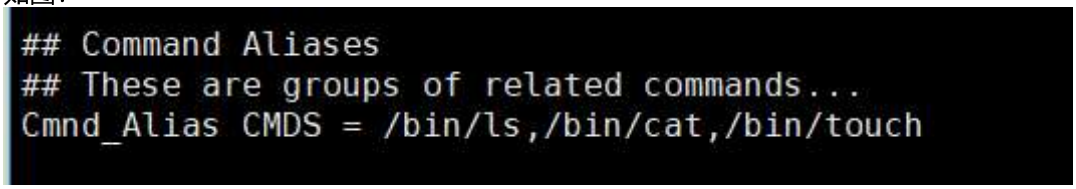
在visudo里找到如下图的地方：


配置格式：Cmnd_Alias 别名 = 命令1，命令2，命令3 #注：命令必须使用绝对路径。

在图中的位置下方添加以下内容：

```
Cmnd_Alias  CMDS = /bin/ls,/bin/cat,/bin/touch
```

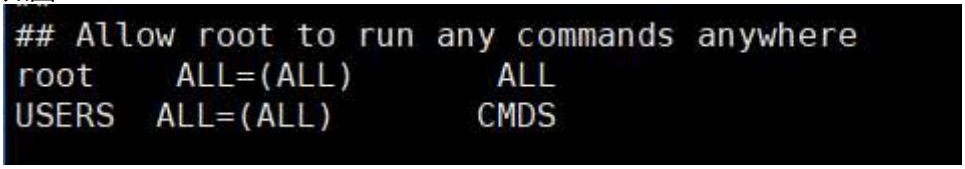
如图：




然后在visudo中## Allow root to run any commands anywhere下面添加以下内容：

```
USERS  ALL=(ALL)       CMDS
```

如图：




这样添加之后，先按ESC键，然后输入“ :wq ”保存退出。然后user1与user2使用sudo授权ls，cat，touch三个命令时，可以做一些ROOT才能执行的操作。

比如cat /etc/shadow

```
[user1@long01 ~]$ cat /etc/shadow
cat: /etc/shadow: 权限不够
[user1@long01 ~]$ sudo cat /etc/shadow
[sudo] user1 的密码：
root:$6$E1PL/I19EBApJlAI$ZzCD.bd0.Ia8ScHk/b64cIay9r2MFE
jBujF1::0:99999:7:::
bin:*:17110:0:99999:7:::
```

又比如ls /etc/sudoers.d

```
[root@long01 ~]# ls -ld /etc/sudoers.d/
drwxr-x---. 2 root root 6 8月    4 2017 /etc/sudoers.d/
[root@long01 ~]# su - user2
上一次登录：五 5月 18 23:20:58 CST 2018pts/0 上
[user2@long01 ~]$ ls -l /etc/sudoers.d
ls: 无法打开目录/etc/sudoers.d: 权限不够
[user2@long01 ~]$ sudo ls -la /etc/sudoers.d
总用量 12
drwxr-x---.  2 root root    6 8月    4 2017 .
drwxr-xr-x. 75 root root 8192 5月   18 23:20 ..
```

sodu详细用法参考：【https://blog.csdn.net/heli200482128/article/details/77833881】

## 2: 限制root远程登录

### 2.1：为什么要限制root远程登陆?

生产环境中，我们为了安全起见，是应该禁止root用户远程登录的。如果有需求要用到root权限，可以用sudo设置，授权给普通用户，

### 2.2：如何限制root远程登陆?

编辑/etc/ssh/sshd_config配置文件：

```
[user2@long01 ~]$ vi /etc/ssh/sshd_config
#找到：#PermitRootLogin yes
#改为：PermitRootLogin no     #注意去掉井号“#”，   改为yes即允许root用户远程登陆
```

#然后保存退出，重启sshd服务

```
[root@long01 ~]# systemctl restart sshd.service
```

这样配置后，当前的root用户退出之后就不能用远程登陆工具登陆了。

### 2.3：限制root远程登陆后怎么登陆root用户?

限制root远程登陆后，只有使用普通用户登陆后，再使用 su - 切换到root，前提是必须知道root用户的密码。还有一个办法就是通过sudo授权su命令，让su命令不需要密码就可以切换到root用户。在visudo里面添加以下内容

```
user1  ALL=(ALL)    NOPASSWD:/usr/bin/su -
```

添加配置后保存退出，之后user1用户就可以直接使用sudo su - 命令切换到root了。

```
[root@long01 ~]# su - user1
上一次登录：五 5月 18 23:07:13 CST 2018pts/1 上
[user1@long01 ~]$ sudo su -
上一次登录：五 5月 18 23:54:39 CST 2018从 10.1.1.169pts/1 上
```

这样做也不安全，最好的方法还是根据需求来设置sudo规则。