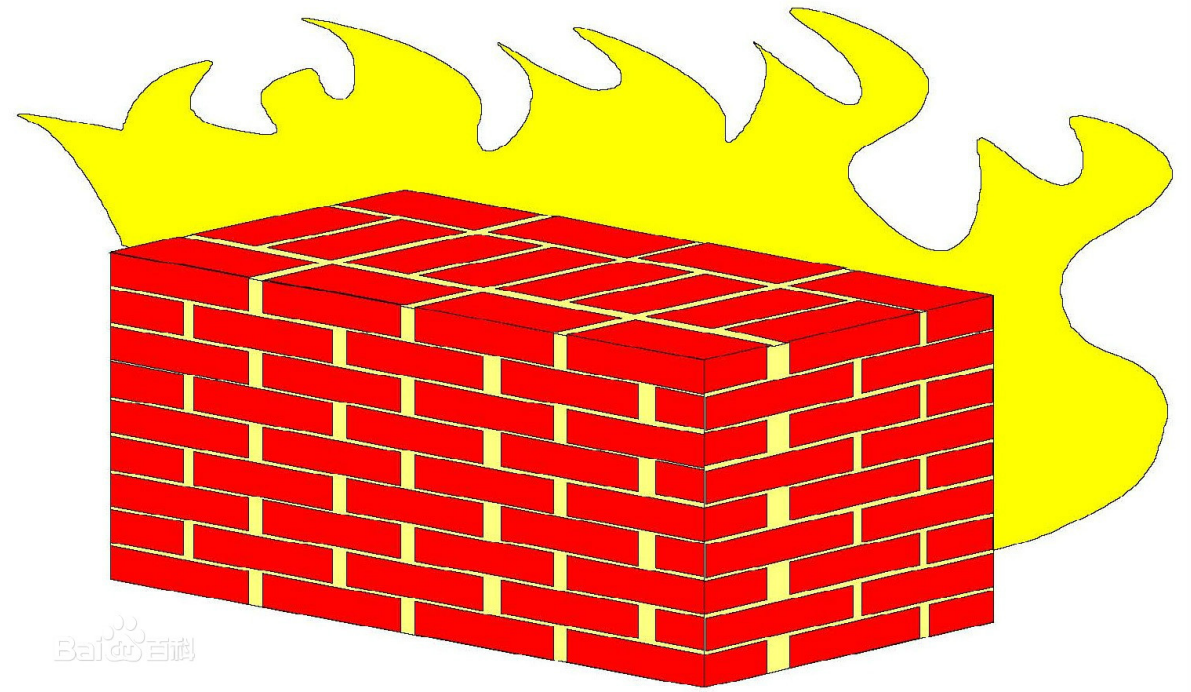
什么是防火墙?

防火墙，顾名思义，就是可以防火的墙，主要用于建筑领域，由不燃烧体构成，当一个区域失火时，可以一定程度上阻止火势蔓延。



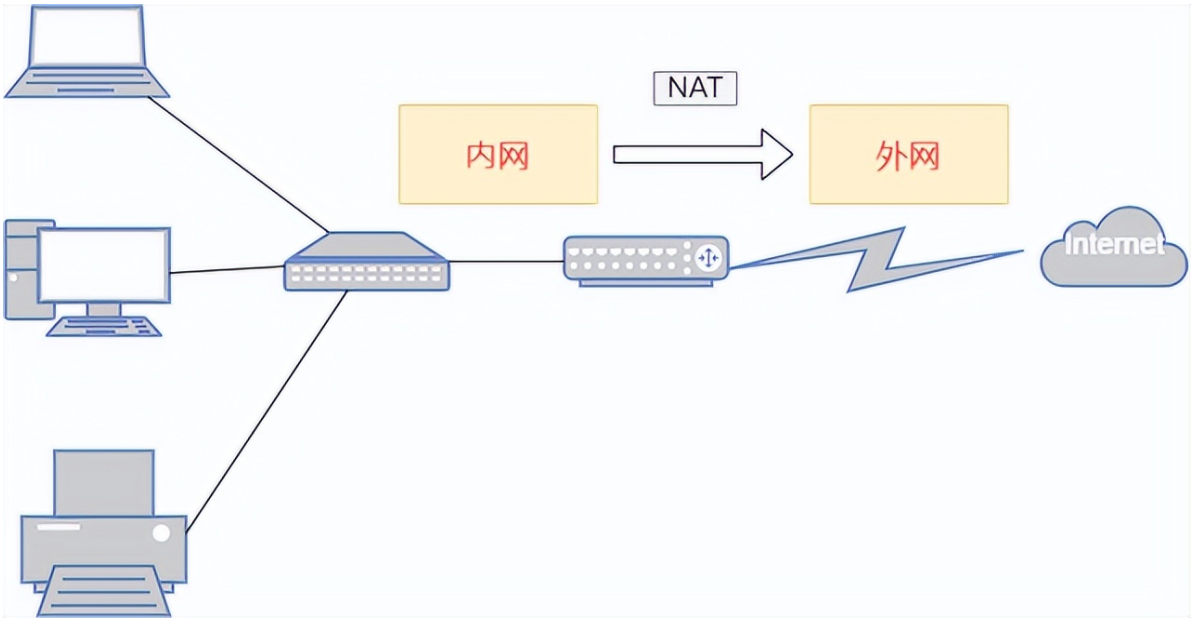
在计算机网络当中，**防火墙被赋予了新的意义，是一种位于内网与外网之间的网络安全系统，起到隔离作用，通过监测、限制、更改跨越防火墙的数据流，尽可能地对外部屏蔽网络内部的信息、结构和运行状况，以此来实现网络的安全保护。**

公众号：黑猫编程
网址：https://noi.hioier.com

比如，早期的网民还是比较纯粹的，上网就是为了分享和获取信息。**随着网络技术发展，黑客群体自然开始出现，黑客通过技术手段窃取私密信息，更改计算机数据，对网络安全造成巨大威胁。或者直接攻击目标主机，使其无法正常工作，比如著名的洪水攻击，即时向目标主机发送信息建立无意义的连接，使目标主机一直处于连接状态，无法处理其他任务。**

什么是内网和外网？

学习IP地址时，IP地址有A、B、C、D、E分类，且了解到IPv4地址最多43亿个，数量少并不充裕。因此，我们平时查询自己电脑IP地址一般都是192开头的，这是私有IP，经常变化，而且不在一个局域网内的用户IP可以重复，只有当我们连接外网，比如访问百度网站的时候，才会经过NAT等转换机制以公有IP进行访问。



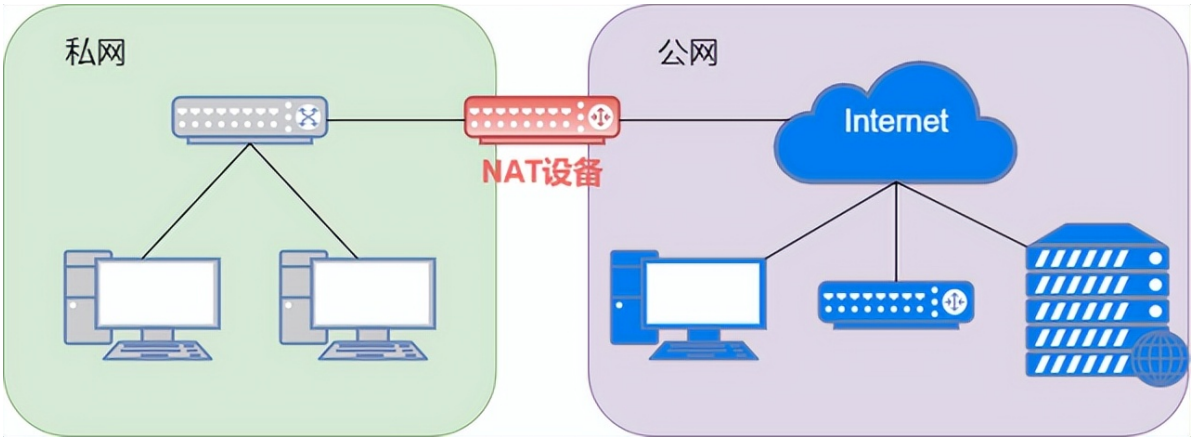
通过cmd的ipconfig命令，可以查询自己的IP：

```
无线局域网适配器 WLAN:

   连接特定的 DNS 后缀 . . . . . . . :
   IPv6 地址 . . . . . . . . . . . . : 240e:388:a2b0:7400:8013:bc0c:3792:b9f
   临时 IPv6 地址. . . . . . . . . . : 240e:388:a2b0:7400:21c5:c32c:219b:c828
   本地链接 IPv6 地址. . . . . . . . : fe80::8013:bc0c:3792:b9f%6
   IPv4 地址 . . . . . . . . . . . . : 192.168.1.2
   子网掩码 . . . . . . . . . . . . : 255.255.255.0
   默认网关. . . . . . . . . . . . . : fe80::1%6
                                       192.168.1.1
```

如果，我们启动一个服务器，在自己的手机或者其他电脑访问本机IP是可以访问的，因为都在局域网内，但是如果将IP和端口发给外地的朋友，是无法打开的，所以搭建网站等服务需要申请共有的不变IP，或者直接租用云服务器，也会有一个共有IP。

公众号：黑猫编程
网址：https://noi.hioier.com



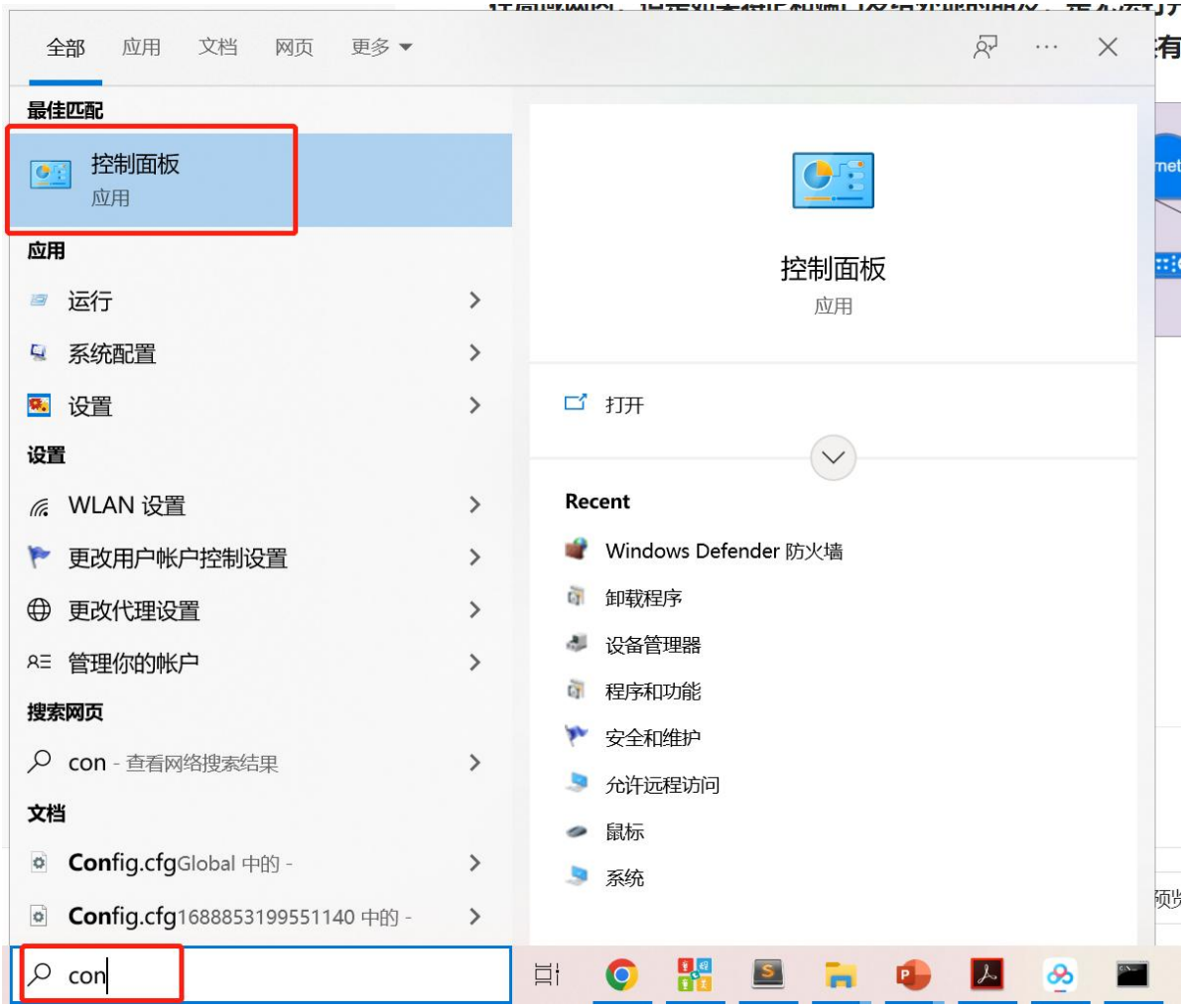
硬件防火墙和软件防火墙

硬件防火墙是有的硬件设备和软件结合的防火墙，而软件防火墙只是软件上的拦截规则。硬件防火墙效果更好，当遭遇密集攻击时比软件防火墙具备更佳的性能，但是需要购买，成本较高，价格差异也较大，一般是几千元，还有5万、10万的防火墙。

Windows系统防火墙设置

打开控制面板，可以在搜索框输入control：

公众号：黑猫编程
网址：https://noi.hioier.com



选择系统和安全：

点击防火墙：

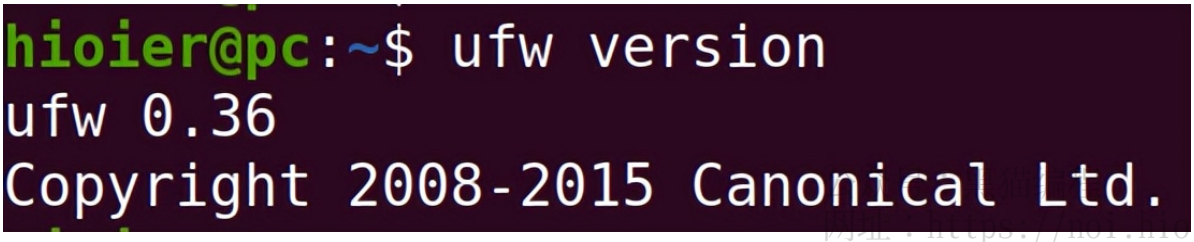
公众号：黑猫编程
网址：https://noi.hioier.com

点击启用或关闭设置是否打开防火墙，点击允许应用或程序通过防火墙可以设置防火墙拦截规则，一般都采用默认设置：

Ubuntu20.04防火墙设置

ubuntu系统使用ufw（简单防火墙Uncomplicated FireWall）配置防火墙。

查看防火墙版本：



博客：https://hoi.hioier.com

启动和关闭防火墙：

```
hioier@pc:~$ sudo ufw enable
Firewall is active and enabled on system startup
hioier@pc:~$ sudo ufw status
Status: active
hioier@pc:~$ sudo ufw disable
Firewall stopped and disabled on system startup
hioier@pc:~$ sudo ufw status
Status: inactive
```

启动一个服务，网页端返回字符串“Hello World!”：

```python
# 从flask模块导入Flask类
from flask import Flask

# 创建flask对象app
app = Flask(__name__)

# 设置访问路径
@app.route('/hello')
def hello():

    # 返回响应内容
    return 'Hello, World!'

if __name__ == '__main__':

    # 启动服务器程序
    app.run(host='0.0.0.0', port='8080')
~
```

```
Desktop  Documents  Downloads  index.py
hioier@pc:~$ python3 index.py
 * Serving Flask app 'index'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deploymen
. Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:8080
 * Running on http://192.168.92.134:8080
Press CTRL+C to quit
192.168.92.1 - - [18/Jan/2023 11:32:39] "GET /hello HTTP/1.1" 200 -
192.168.92.1 - - [18/Jan/2023 11:32:40] "GET /favicon.ico HTTP/1.1" 404 -
```

公众号：黑猫编程
网址：https://noi.hioier.com

当前服务使用8080端口，可以访问。服务器用Ubuntu20.04虚拟机，网址在Windows主机输入。

```
hioier@pc:~$ sudo ufw deny 8082
Skipping adding existing rule
Skipping adding existing rule (v6)
hioier@pc:~$
hioier@pc:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

To                         Action      From
--                         ------      ----
8080                       ALLOW IN    Anywhere
8082                       DENY IN     Anywhere
8080 (v6)                  ALLOW IN    Anywhere (v6)
8082 (v6)                  DENY IN     Anywhere (v6)

hioier@pc:~$ python3 index.py
 * Serving Flask app 'index'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a produc
. Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:8082
 * Running on http://192.168.92.134:8082
Press CTRL+C to quit
```

deny禁止8082端口，再次启动服务，在主机上就无法访问。

默认情况下，**UFW默认策略设置为阻止所有传入流量并允许所有传出流量**，你可以使用以下命令来设置自己的默认策略：

```
# ufw default allow outgoing
# ufw default deny incoming
```

设置SSH或其他规则：

```
允许传入SSH连接，可以使用以下命令：
# ufw allow ssh
```

公众号：黑猫编程
网址：https://noi.hioier.com

这将创建防火墙规则-允许端口22上的所有连接
也可以通过直接指定端口来创建等效规则

```
# ufw allow 22
```

可以基于TCP或UDP协议来过滤数据包，命令如下：

```
# ufw allow 80/tcp
# ufw allow 21/udp
```

```
hioier@pc:~$ sudo ufw allow 8082/tcp
Rule added
Rule added (v6)
hioier@pc:~$ sudo ufw deny 8083/udp
Rule added
Rule added (v6)
hioier@pc:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

To                         Action      From
--                         ------      ----
8080                       ALLOW IN    Anywhere
8082                       DENY IN     Anywhere
8082/tcp                   ALLOW IN    Anywhere
8083/udp                   DENY IN     Anywhere
8080 (v6)                  ALLOW IN    Anywhere (v6)
8082 (v6)                  DENY IN     Anywhere (v6)
8082/tcp (v6)              ALLOW IN    Anywhere (v6)
8083/udp (v6)              DENY IN     Anywhere (v6)
```

查看防火墙编号：

```
hioier@pc:~$ sudo ufw status numbered
Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 8080                       ALLOW IN    Anywhere
[ 2] 8082                       DENY IN     Anywhere
[ 3] 8082/tcp                   ALLOW IN    Anywhere
[ 4] 8083/udp                   DENY IN     Anywhere
[ 5] 8080 (v6)                  ALLOW IN    Anywhere (v6)
[ 6] 8082 (v6)                  DENY IN     Anywhere (v6)
[ 7] 8082/tcp (v6)              ALLOW IN    Anywhere (v6)
[ 8] 8083/udp (v6)              DENY IN     Anywhere (v6)
```

按照编号删除规则：

公众号：黑猫编程
网址：https://noi.hioier.com

```
hioier@pc:~$ sudo ufw delete 3
Deleting:
 allow 8082/tcp
Proceed with operation (y|n)? y
Rule deleted
```

公众号：黑猫编程
网址：https://noi.hioier.com