# 访问控制列表 ACL

**访问控制列表概述：**

**访问控制列表（ACL）是应用在路由器接口的指令列表（即规则）**

**其基本原理：ACL使用包过滤技术，在路由器上读取OSI七层参考模型的第三层和第四层包头中的信息。根据预先定义好的规则，对包进行过滤从而达到访问控制的目的。**

**ACL通过在路由器接口处控制数据包是转发还是丢弃。**

**路由器根据ACL中指定的条件来检测通过路由器的数据包**

**标准访问控制列表：检查数据包的源IP地址。其结果基于IP地址，来决定允许还是拒绝转发数据包。它使用1~99之间的数字作为表号。**

**扩展访问控制列表：对数据包的源IP地址和目的IP地址、指定协议、端口和标志来允许或拒绝数据包。使用100~199之间的数字作为表号。**

**命名访问控制列表：允许在标准和扩展中使用名称代替表号（更友好）。**

**访问控制列表的工作原理：ACL是一组规则的集合。应用在路由器的某个接口上，对路由器而言有两个方向：**

**出：已经过路由器的处理，正离开路由器接口的数据包**

**入：已到达路由器接口的数据包，将被路由器处理**

**如果对接口应用了ACL，那么路由器将对数据包进行该组规则的顺序进行检查。**

**如果匹配某一条规则路由器将决定该数据包允许或拒绝通过**

**如果最后没有任何一条规则匹配，则路由器根据默认的规则将丢弃该数据包。要注意是逐条匹配，按顺序来的**

**标准访问控制列表的配置：**

**R1(config)#access-list access-list-number ｛permit丨deny｝ source ｛source wildcard｝**

**access-list-number：访问控制列表表号**

**permit丨deny：满足测试条件，则拒绝/允许通过流量**

**Source：数据包的源地址，可以是主机地址或网络地址**

**｛source wildcard｝：通配符掩码，也叫做反码。在用二进制数0和1表示时，如果为1表明这一位不需要匹配，如果为0表明这一位需要严格匹配**

**扩展访问控制列表的配置：**

**创建ACL命令语法如下：**

**R1(config)#access-list access-list-number ｛permit丨deny｝protocol ｛source source-wildcard destination destination-wildcard｝｛operator operan｝**

**对命令参数的详细说明如下：**

**access-list-number：访问控制列表的表号，对于扩展ACL是100~199**

**｛permit丨deny｝：当满足条件，则允许或拒绝该流量通过**

**Protocol：用来指定协议类型，如IP、TCP、UDP、ICMP等**

**source source-wildcard：源地址和源反掩码**

**destination destination-wildcard：目的地址和目的反掩码**

**｛operator operan｝：ls（小于）、gt（大于）、eq（等于）或neq（不等于）一个端口号**

**命名访问控制列表的配置：**

**创建ACL的语法命令如下：**

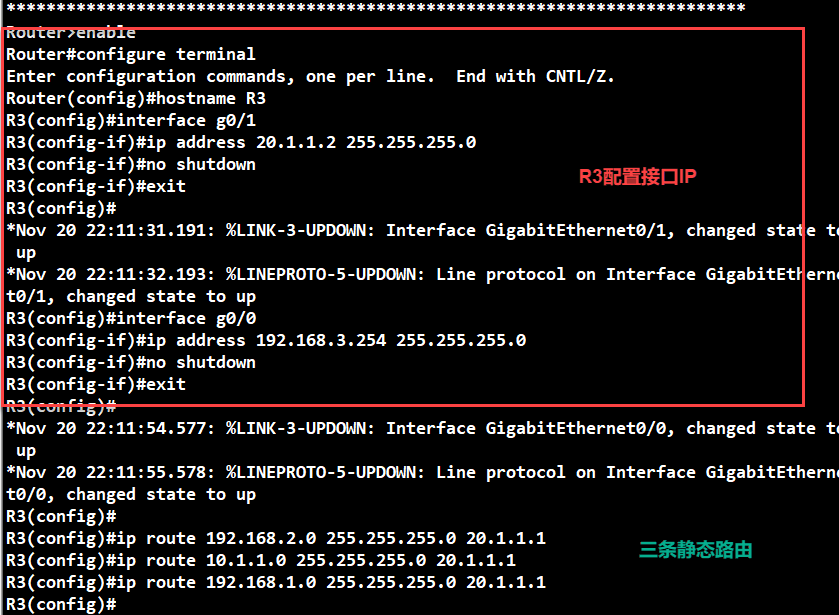
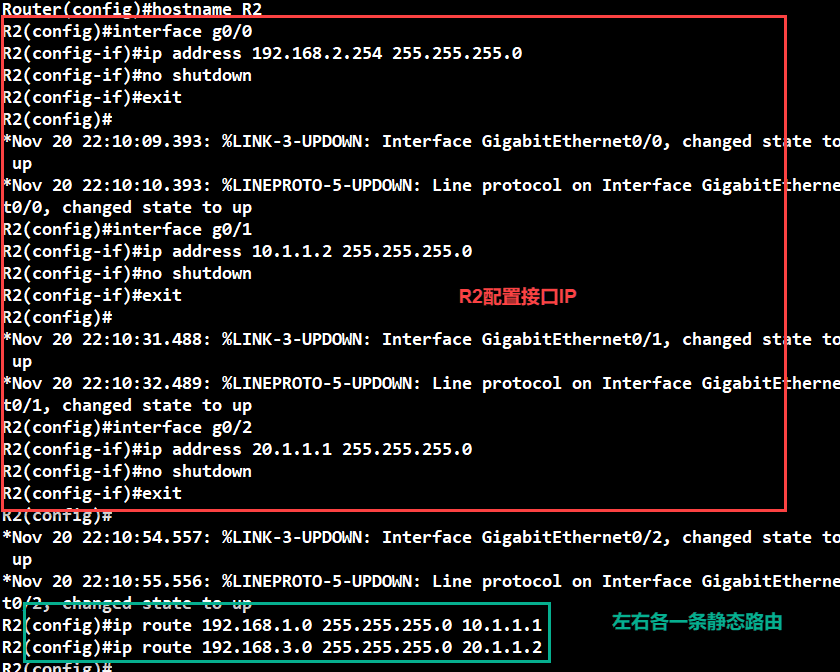
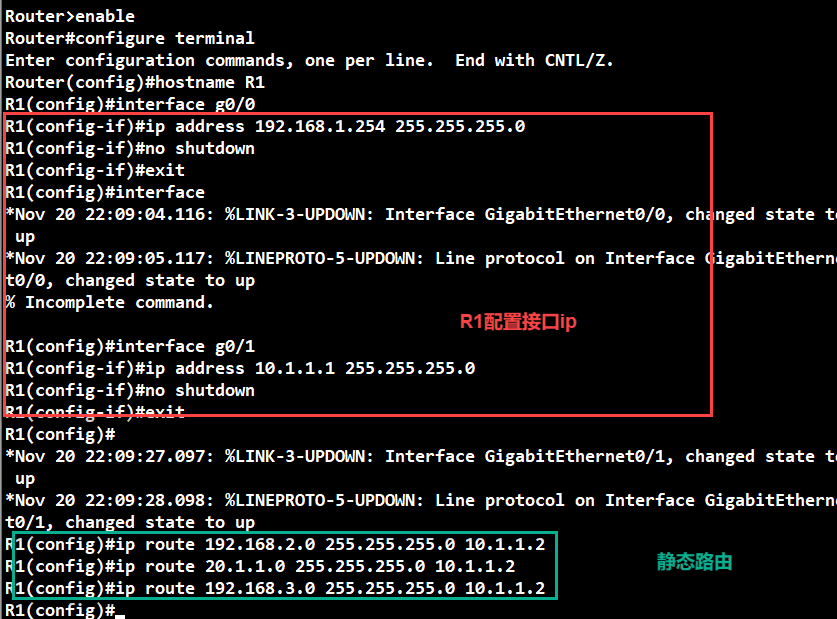
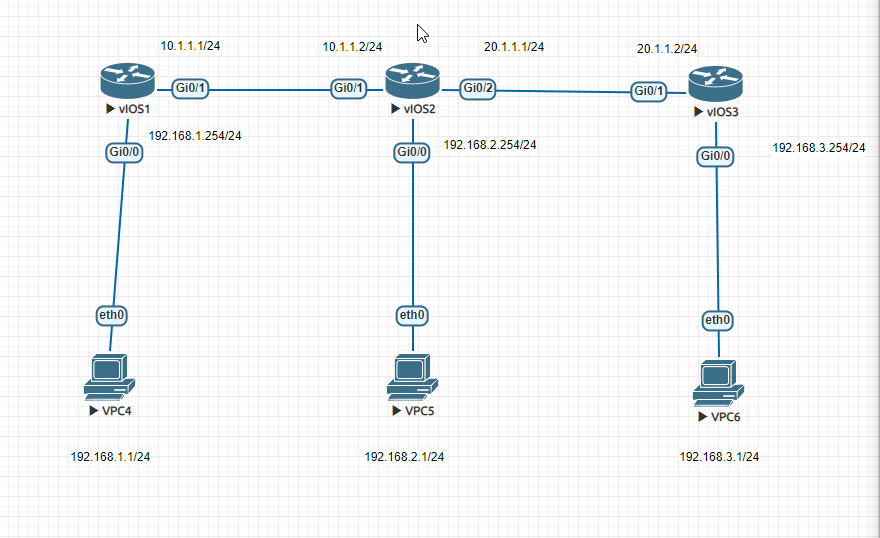
**R1(config)#ip access-list ｛standard丨extended｝access-list-name**

**其中access-list-name是ACL的名字，应具有实际意义**

**｛standard丨extended｝选择标准ACL或可扩展的ACL**

-----------------------

# 实验1:标准ACL



**//创建一个标准访问控制列表1，允许192.168.2.0网段访问**

**R3(config)#access-list 1 permit 192.168.2.0 0.0.0.255**

**//创建一个标准访问控制列表1,拒绝任何IP访问**

**R3(config)#access-list 1 deny any**

**//进入g0/1接口**

**R3(config)#interface g0/1**

**//将标准访问控制列表1应用在入口**

**R3(config-if)#ip access-group 1 in**

**R3(config-if)#exit**

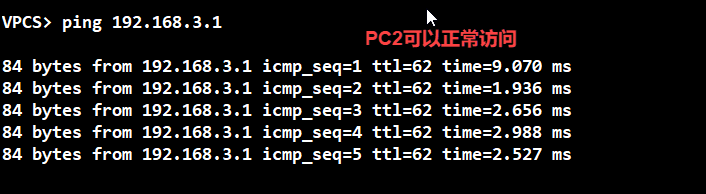
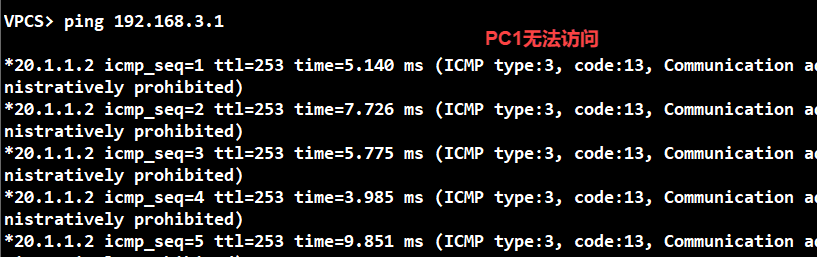
**//查看访问控制列表**

**R3#show access-list**

**Standard IP access list 1**

**10 permit 192.168.2.0, wildcard bits 0.0.0.255**

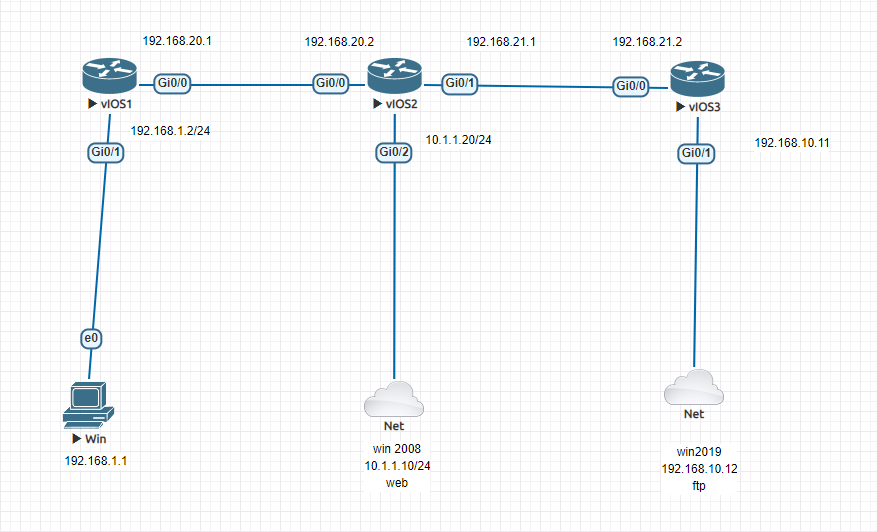
**20 deny any**



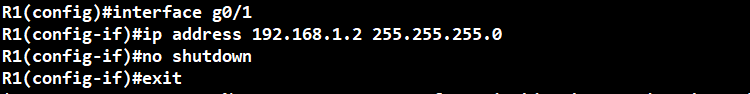
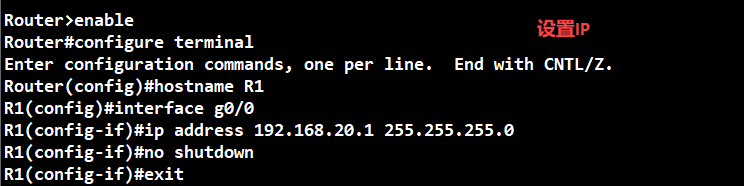
# 实验2：扩展访问控制列表

**基于上下文的访问控制列表CBAC能够处理那些不能使用自反ACL处理的多端口协议**

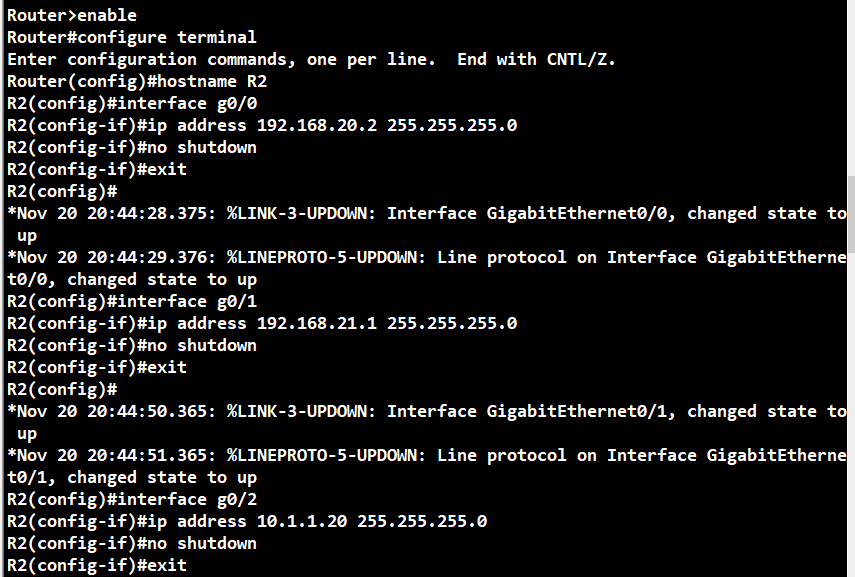
**CBAC的限制：(1)CBAC不能为所有协议提供智能过滤,他只能为你所指定的协议进行过滤；(2)CBAC只能提供一定类型的入侵保护.只能提供大部分常规的入侵保护；(3)CBAC只能检查TCP和UDP连接，不能检测ICMP、路由协议等；(4)对第三者路由器的加密的信道也无法进行检测，如IPSec信道。**



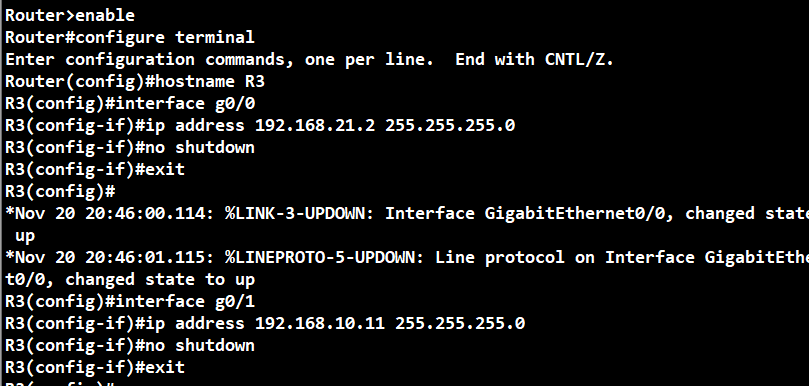
**R1路由器**

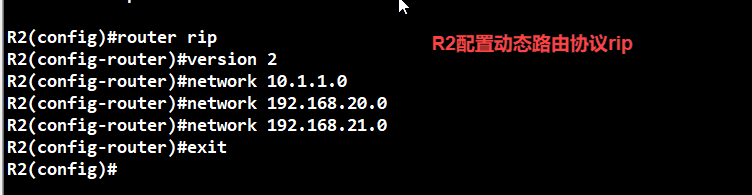
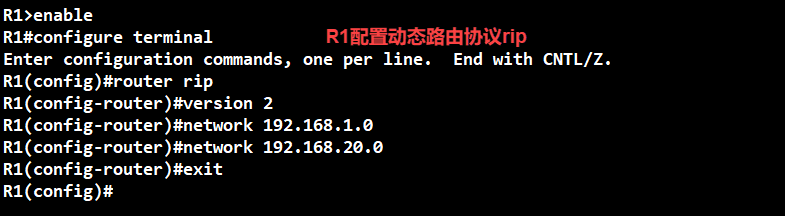


**R2路由器**

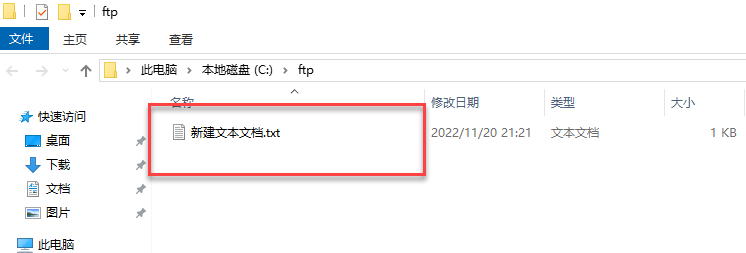
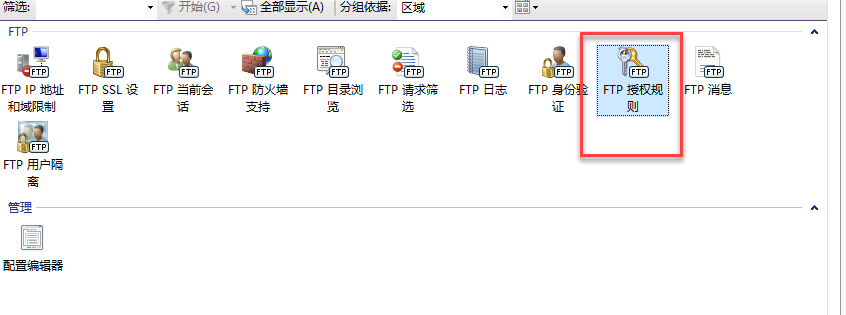
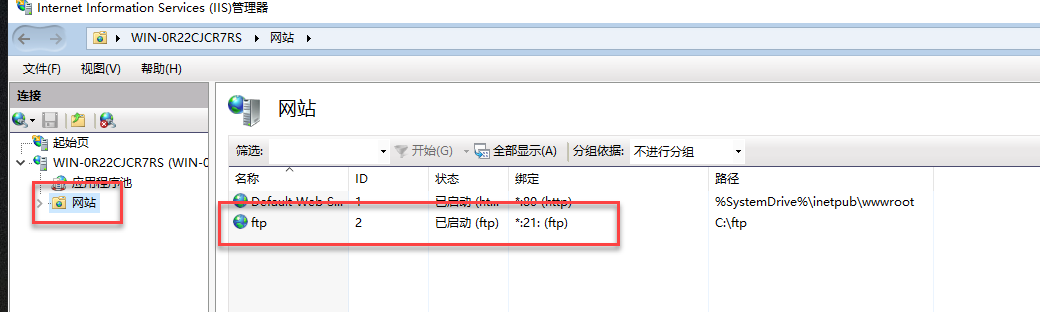
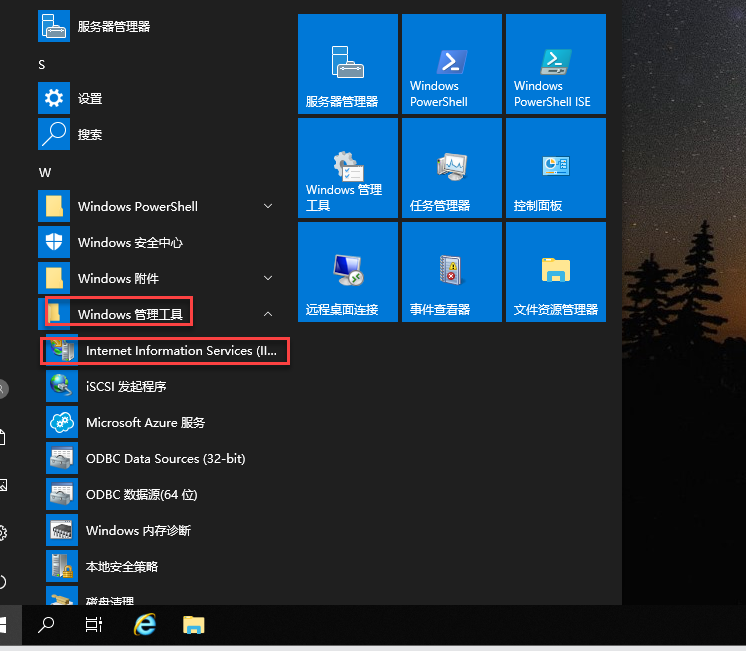
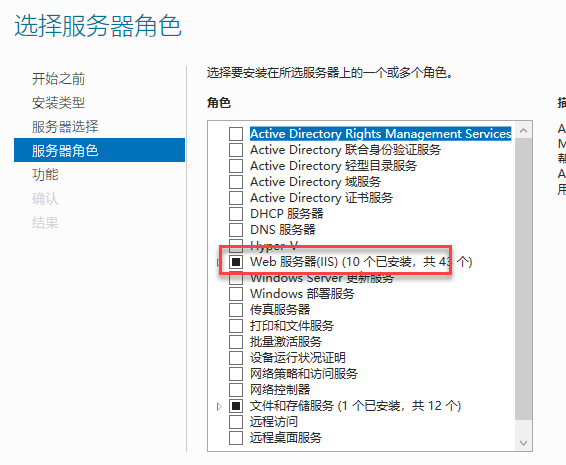
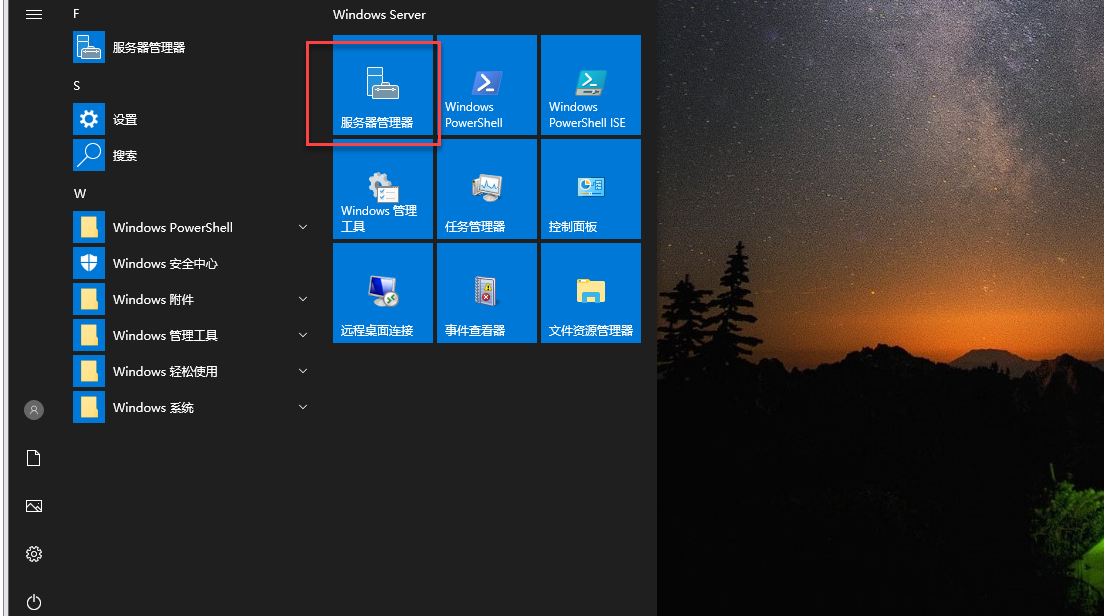


**R3配置ip**

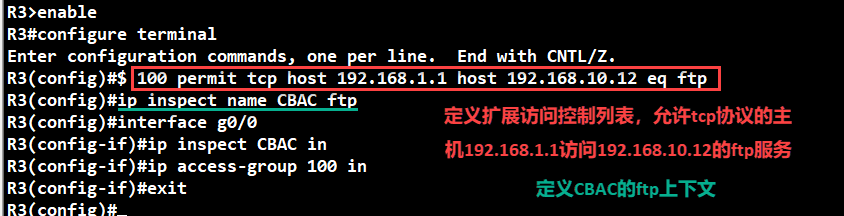
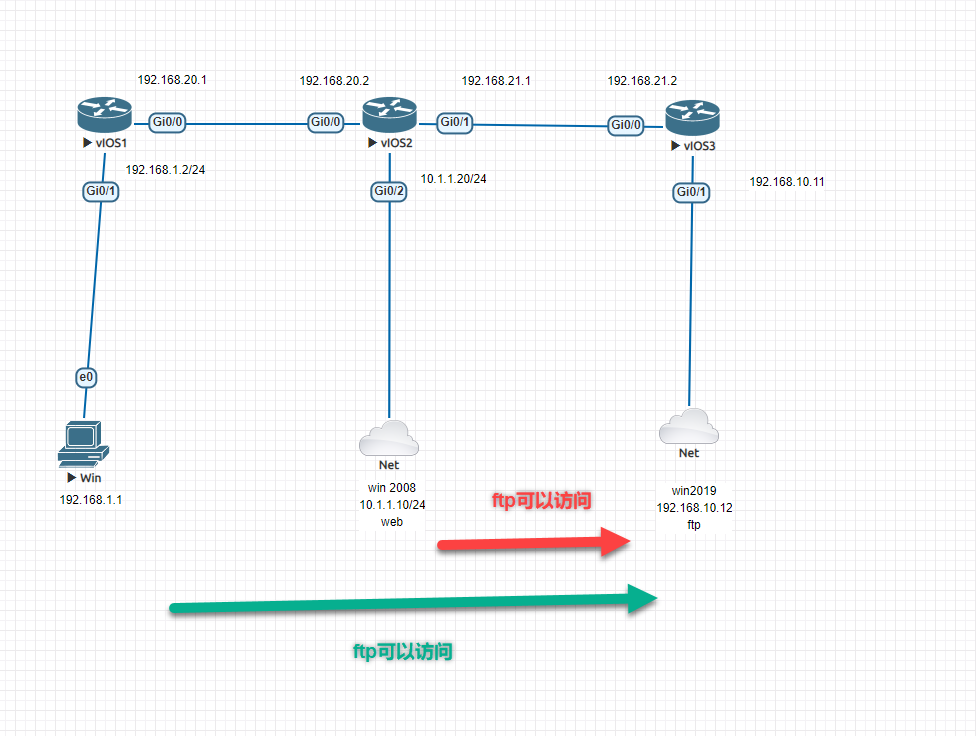
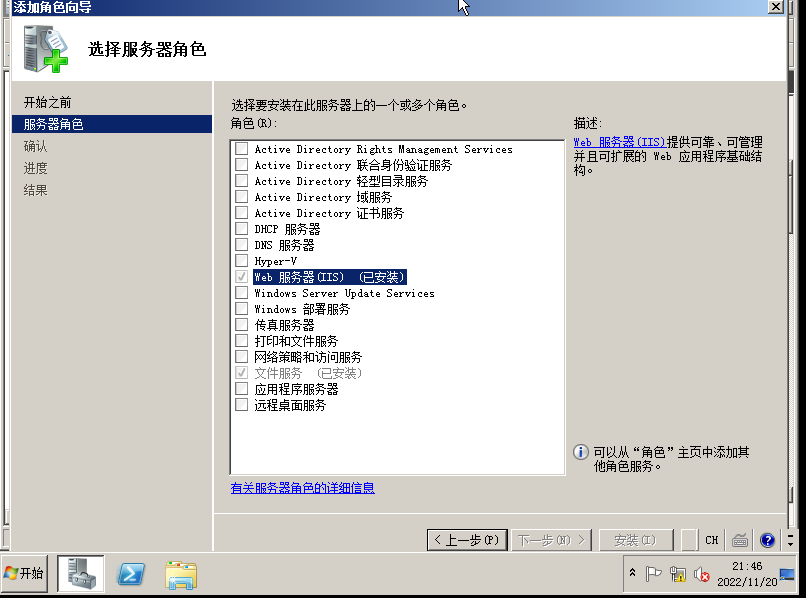




Win2019配置web和ftp服务



Win2008配置web和ftp：



**//定义扩展访问控制列表100，允许主机192.168.1.1的tcp访问主机10.1.1.10的web服务**

**R2(config)#access-list 100 permit tcp host 192.168.1.1 host 10.1.1.10 eq 80**

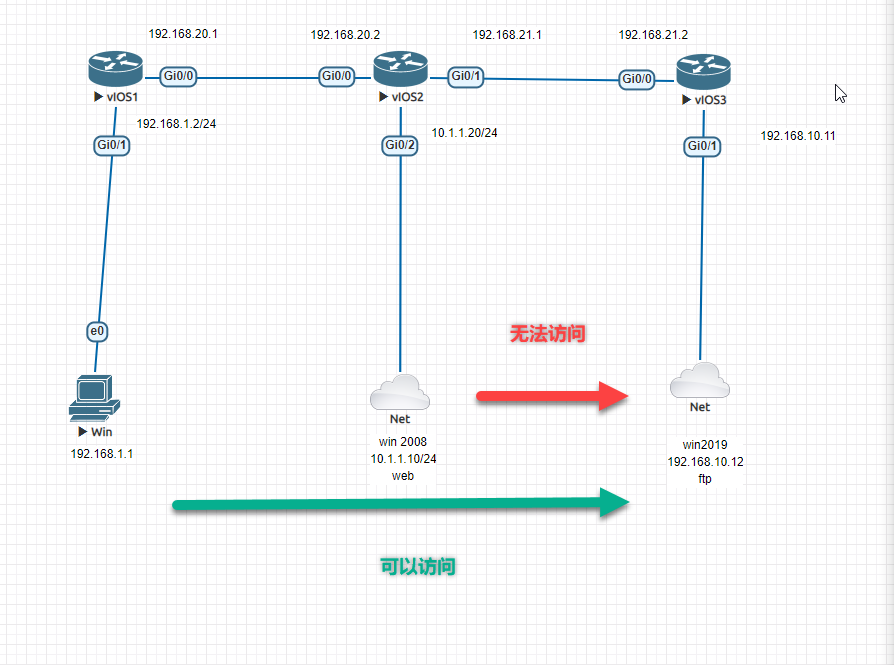
**R2(config)#interface g0/2**

**R2(config-if)#ip access-group 100 out //在g0/2出接口应用扩展访问控制列表100**

**//定义扩展访问控制列表100，允许主机192.168.1.1的tcp访问主机192.168.10.12的ftp服务**

**R3(config)#access-list 100 permit tcp host 192.168.1.1 host 192.168.10.12 eq ftp**

**R3(config)#ip inspect name CBAC ftp //定义ftp访问的CBAC上下文**



实验2：扩展访问控制列表

基于上下文的访问控制列表CBAC能够处理那些不能使用自反ACL处理的多端口协议

CBAC的限制：(1)CBAC不能为所有协议提供智能过滤,他只能为你所指定的协议进行过滤；(2)CBAC只能提供一定类型的入侵保护.只能提供大部分常规的入侵保护；(3)CBAC只能检查TCP和UDP连接，不能检测ICMP、路由协议等；(4)对第三者路由器的加密的信道也无法进行检测，如IPSec信道。

=========================================

Router(config)#router rip //使用rip协议

Router(config-router)#version 2 //使用版本2

Router(config-router)#network 192.168.2.0 //宣告网段192.168.2.0

Router(config-router)#network 192.168.10.0

Router(config-router)#network 192.168.11.0

Router(config-router)#exit

3(config)#interface g0/1 //在出接口应用扩展访问控制列表100策略

R3(config-if)#ip inspect CBAC out

R3(config-if)#ip access-group 100 out