Моделът Netfilter. IP tables

проф. д-р инж. Венета Алексиева

Основни моменти

- Същност на Netfilter
- Таблици
- Конекции
- Вериги
- Правила
- Цели
- Пример

Netfilter

- Представлява общ програмен модел на филтриране на пакети.
- Осигурява ефективен контрол на трафика под Linux.
- Софтуерната реализация е базирана на пакета *iptables*.

iptables

- Изградени са от модули
- Основен модул е *ip_tables*
- Използват се таблици, всяка от които се състои от последователност от правила (вериги - *chains*), по които преминават пакетите.
- Правилата дефинират:
  - критериите, които трябва да удовлетворят преминаващите през машината пакети
  - действията, които ще се предприемат при тяхното удовлетворяване.
- Веригите са:
  - Стандартни
  - Дефинирани от потребителя

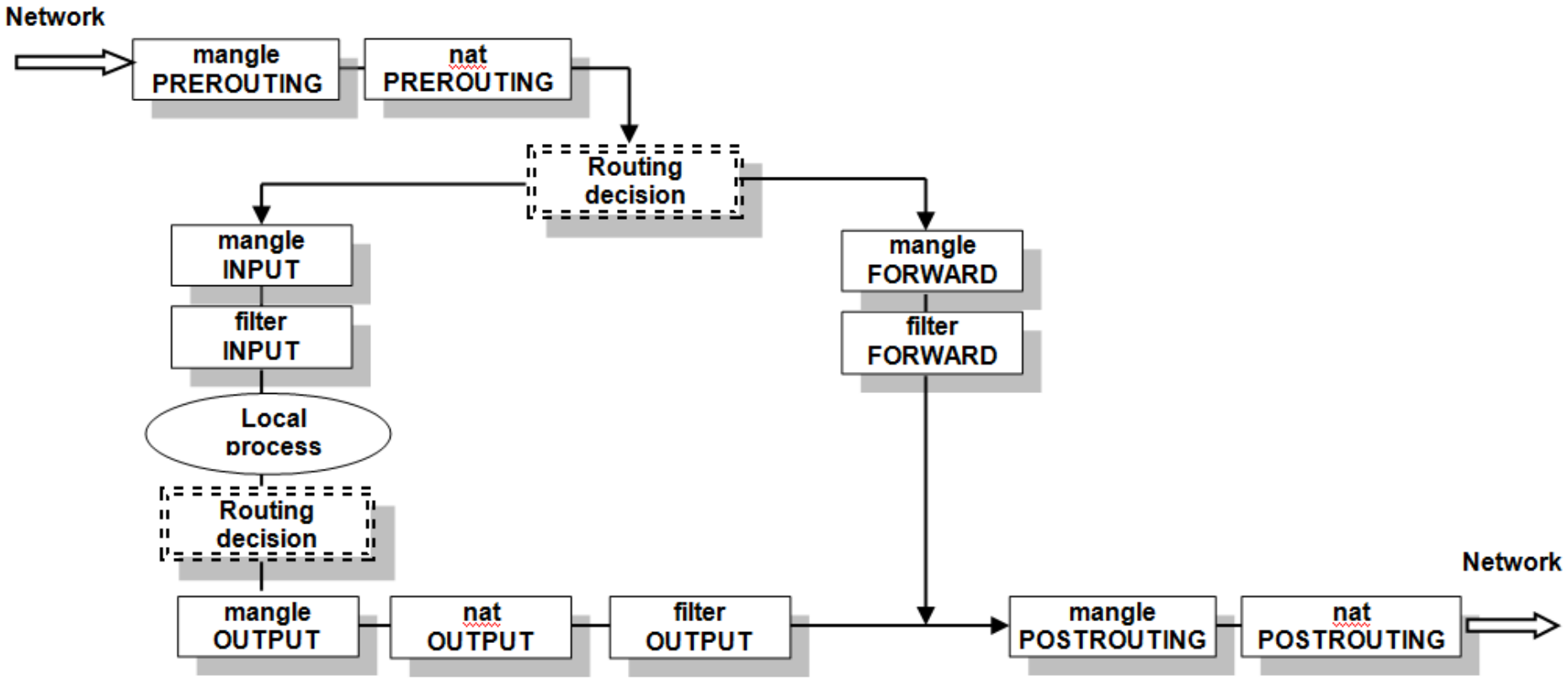
Стандартни таблици

- **mangle** – използва се за модифициране на пакетите
  - промяна на полето *Type of Service* - как пакетът да бъде маршрутизиран.
  - модул *iptable_mangle*;
- **nat** –за транслиране на адресите
  - DNAT – промяна на адреса на получателя;
  - SNAT – промяна на адреса на източника с фиксиран IP адрес;
  - MASQUARADE – промяна на адреса на източника с нов, избран от адресен пул.
  - модула *iptable_nat*;
- **filter** –за филтриране на пакетите
  - Всички стандартни цели са приложими за тази таблица.
  - модул *iptable_filter*.

Стандартни вериги

- **INPUT** – входна верига за предназначените за локалната машина пакети
- **OUTPUT** – изходна верига за пакетите, генерирани от локалната машина
- **FORWARD** – верига за пакетите, предназначени за маршрутизиране към други машини;
- **PREROUTING** – верига, по която преминават пакетите преди определяне на получателя им (локално или за друг хост);
- **POSTROUTING** – верига, по която преминават пакетите след определяне на техния получател.

Преминаване на пакетите през таблиците на Netfilter



Състояние на конекциите

- **NEW** – указва първия наблюдаван пакет за специфична конекция.
  - За TCP конекции това е пакетът с вдигнат единствено флаг SYN.
  - При UDP и ICMP обмен това са първите пакети, предадени между източника и получателя;
- **ESTABLISHED** – всеки трафик в двете посоки удовлетворява това състояние.
  - При TCP конекции това са пакетите с установен флаг ACK.
  - При UDP обмен всички пакети между източника и получателя след първия предаден се третират в това състояние.
  - ICMP съобщенията за грешки също се разглеждат в това състояние, ако преди това е бил изпратен пакет, който предизвиква генерирането на съответното съобщение за грешка;
- **RELATED** – една конекция е в това състояние, ако тя се явява относителна спрямо друга вече създадена (ESTABLISHED).
  - Например това са FTP данновите конекции, които се създават допълнително спрямо създадените вече FTP контролни конекции;
- **INVALID** – в това състояние попадат пакетите, които не могат да се идентифицират или не са в нито едно от описаните състояния.
  - препълване на паметта
  - ICMP съобщения за грешки, които не са в отговор на нито една известна конекция. Препоръчва се пакети в това състояние да се отхвърлят.

Създаване на правила

iptables [-t *table*] *command* [*match*] [*target*/*jump*]

- Критериите (*match*)
- Съответните действия (*target*) - отхвърляне / пропускане.
- Преход (*jump*) към друга верига при удовлетворяването на критериите
- Таблица (*table*) - по подразбиране е таблица **filter**
- Командата (*command*) указва действието
  - например, включване на ново правило или премахване на правило.

Изпълнение на правилата

- Изпълняват се последователно от началото на списъка.
- При удовлетворяване на критериите за дадено правило се изпълняват съответните действия и се преустановява по-нататъшната обработка на списъка.
- Ако е указан преход към нова верига, то обработката продължава аналогично със списъка от правила в новата верига.
- В случай, че нито едно правило от новата верига не бъде удовлетворено, обработката продължава със следващото по ред правило от старата верига, следващо позицията от която е бил направен преди това преходът.
- Ако няма удовлетворяване на критериите на нито едно правило, то върху пакета се прилага политиката по подразбиране за даденатa верига.

Добавяне на правило

-A, --append

iptables –A INPUT ...

- добавя ново правило в края на указаната верига

Изтриване на правило

-D, --delete

iptables –D INPUT –s 192.168.0.1 –j DROP

iptables –D INPUT 2

- Премахва правило от указаната верига:
- чрез задаване на цялото правило (което напълно трябва да съответства на премахваното)
- чрез задаване на неговия пореден номер.
  - Правилата се номерират от началото на веригата, започвайки с номер 1.

Замяна на правило

-R, --replace

iptables –R INPUT 2 –s 192.168.0.1 –j DROP

- Заменя правило по зададен номер от указаната верига с ново.

Вмъкване на правило

I, --insert

iptables –I INPUT 3 –s 192.168.0.1 –j DROP

- Вмъква правило на зададената позиция в указаната верига.

Изчистване на верига

-F, --flush

iptables –F INPUT

- Изчиства всички правила от указаната верига.

Създаване на нова верига

-N, --new-chain

iptables –N tcp_allowed

- Създава се нова верига със зададеното име в съответната таблица.

Изтриване на верига

-X, --delete-chain

iptables –X tcp_allowed

- Изтрива указаната верига от съответната таблица. Необходимо е изтриваната верига да не съдържа правила.

Задаване на политика по подразбиране

-P, --policy

iptables –P INPUT DROP

- Установява зададената цел (политика) към указаната верига.
- Всички пакети, които не удовлетворят нито едно правило ще попаднат под тази политика за веригата.
- Валидни цели са **ACCEPT** и **DROP**.
- При изчистването на правилата от дадена верига зададената преди това политика по подразбиране остава валидна.

Преименоване на верига

-E, --rename-chain

iptables –E tcp_allowed tcp_disallowed

- Сменя името на указаната верига с ново.

Общи критерии за удовлетворяване

- **Протокол -p, --protocol**
- Указва проверка за специфичен протокол като стандартно дефинираните TCP, UDP и ICMP или стойност, описана във файла */etc/protocols*.
- Могат да се изброят няколко имена на протоколи, разделени със запетая. За указване на всичките тези три протокола се използва ключовата дума ALL.

iptables –A INPUT –p tcp –j DROP

iptables –A OUTPUT –p ALL –j ACCEPT

Общи критерии за удовлетворяване

- **Източник на дейтаграмата -s, --src, --source**
- Указва съвпадение с пакети, имащи зададения IP адрес на източника.
- Може да се указва както конкретен IP адрес, така и адрес на мрежа с добавяне на мрежова маска.

iptables –A INPUT –s 192.168.0.1

iptables –A INPUT –s 192.168.0.0/24

Общи критерии за удовлетворяване

- **Получател на дейтаграмата -d, -dst, --destination**
- Указва съвпадение с пакети, имащи зададения IP адрес на получателя. Функционирането е аналогично както при указване на източник.

iptables –A FORWARD –d 192.168.0.1

iptables –A FORWARD –d 192.168.0.0/24

Общи критерии за удовлетворяване

- **Входен интерфейс –i, --in-interface**
- Указва съвпадение с интерфейса, по който пакетът се е получил. Тази опция е валидна единствено за веригите INPUT, FORWARD и PREROUTING.
- Указване на произволна последователност от букви и цифри се задава със символа +.
- Например, всички Ethernet интерфейси могат да се укажат като **eth+**.

iptables –A INPUT –i eth0

Общи критерии за удовлетворяване

- **Изходен интерфейс –o, --out-interface**
- Аналогично за пакети, които напускат по указания интерфейс.
- Опцията е валидна единствено за веригите OUTPUT, FORWARD и POSTROUTING.

iptables –A FORWARD –o eth1

Общи критерии за удовлетворяване

- **Фрагментирани пакети -f, --fragment**
- Използва се за указване на съвпадение с вторите и следващите части на фрагментирани пакети.

iptables –A INPUT -f

Задаване на критерии, различни от указаните се извършва чрез използване на знака за инвертиране ‘**!**’.

Неявни критерии за удовлетворяване

TCP порт на източника --sport, --source-port

- Указва съвпадение по зададен порт на източника.
- Без зададен порт се подразбират всички портове.
- Ако се използва име на услуга, то трябва да бъде дефинирано във файла */etc/services*.
- Могат да се задават диапазон от портове във вида <*начален_порт*>:<*краен_порт*>.
- Ако не се зададе *начален_порт*, се подразбира порт 0.
- Ако не се зададе *краен_порт* се подразбира порт 65535.

iptables –A FORWARD –p tcp –-sport 80

Неявни критерии за удовлетворяване

TCP порт на получателя --dport, --destination-port

- Аналогично, по отношение на порт на получателя.

iptables –A FORWARD –p tcp –-dport 22

Неявни критерии за удовлетворяване

TCP флагове --tcp-flags

- Указва съвпадение с TCP флагове в пакета.
- Задава се списък от флагове за проверка (маска) и втори списък от флагове, които трябва да бъдат установени в 1. Двата списъка се разделят със запетаи.
- Съвпадението е по отношение на флаговете SYN, ACK, FIN, RST, URG и PSH, като всичките могат да се укажат с ALL, а нито един – с NONE.

iptables –A FORWARD –p tcp –-tcp-flags SYN,ACK,FIN SYN

Неявни критерии за удовлетворяване

UDP порт на източника --sport, --source-port

- Функционира аналогично както при TCP протокола.

iptables –A FORWARD –p udp –-sport 53

Неявни критерии за удовлетворяване

UDP порт на получателя --dport, --destination-port

- Функционира аналогично както при TCP протокола.

iptables –A FORWARD –p udp –—dport 53

Неявни критерии за удовлетворяване

ICMP тип --icmp-type

- Указва съвпадение по ICMP тип съгласно RFC792.
- За получаване на пълния списък от типовете ICMP съобщения, може да се изпълни командата **iptables –p icmp --help**.

iptables –A FORWARD –p icmp —icmp-type 8

Явни критерии за удовлетворяване

Явните критерии трябва да се заредят изрично с опцията **–m**, **--match *module***. Някои от тях са специфични за даден протокол, други са въведени с цел тестване и експерименти с iptables. Най-често използваните са следните:

Съвпадение по MAC адрес --mac-source

- Използва се за установяване на пакети по Ethernet MAC адреса на източника.
- Опцията е валидна единствено за веригите PREROUTING, FORWARD и INPUT.

iptables –A INPUT –m mac –-mac-source 2b:00:01:1a:2a:05

Явни критерии за удовлетворяване

Съвпадение с множество портове -source-port, --destination-port, --port

- Използва се за указване на съвпадение по множество портове на източника на пакета, на получателя или и на двата.
- Тази опция елиминира необходимостта от задаване на множество еднотипни правила за съвпадение по различни портове.
- Използва се единствено с опции **–p tcp** и **–p udp** и максимален брой от 15 порта.

iptables –A INPUT –p tcp –m multiport -–source-port 23,53

iptables –A INPUT –p tcp –m multiport -–destination-port 23,53

iptables –A INPUT –p tcp –m multiport –port 22,53

Явни критерии за удовлетворяване

Съвпадение по състояние на конекциите --state

- Позволява откриване на конекция в определено състояние.
- Тази възможност изисква изрично зареждане с опцията **–m state**.
- За използването на състояния е необходимо предварително зареждане на модула *ipt_state*.

iptables –A INPUT –m state –-state RELATED,ESTABLISHED

Изпълнение на действия (target/jump) при удовлетворяване на критериите

- Действията (целите) указват на дадено правило как да се манипулира с удовлетворилия критериите пакет.
- Преходът е частен случай на действията, при който се извършва преминаване към предварително създадена от потребителя верига от правила в същата таблица.
- Ако пакетът достигне края на новата верига, той продължава своето преминаване по старата верига от правилото, преди което е бил изпълнен преходът.

iptables –N tcp_allowed

iptables –A INPUT –p tcp –j tcp_allowed

iptables –A INPUT –p tcp –j DROP

Цел ACCEPT

- Прекратява се преминаването на пакета през текущата верига или други вериги в същата таблица и той се приема за последваща обработка.

iptables –A INPUT –p tcp –j ACCEPT

Цел DNAT

- Използва се за преобразуване на адреса на получателя.
- Всеки пакет, удовлетворил критериите, и всички следващи го в същия поток данни ще бъдат преобразувани и препратени към съответните получатели.
- Типично приложение е когато е налице web сървър във вътрешната мрежа и необходимост от достъп до него на потребителите от Интернет.
- В този случай, firewall ще препраща всички получени на неговия собствен HTTP порт пакети към реалния web сървър във вътрешната мрежа. Тази цел е валидна единствено за веригите PREROUTING и OUTPUT на **nat** таблицата.

iptables –t nat –A PREROUTING –p tcp –d 200.100.50.10 –-dport 80 –j DNAT –-to-destination 192.168.0.1:80

Цел DROP

- Отхвърля пакетите от по-нататъшна обработка.

Цел LOG

- Дава възможност за съхраняване на информация за преминаването на пакетите чрез програмата *syslog*.
- Двете най-често използвани опции за задаване на нивото на приоритет на съобщенията (debug, info, notice, warning, error, alert, emerg, panic) – **log-level** и на допълнителния префикс до 29 символа към тези съобщения – **log-prefix**.
- За използването на тази цел е необходимо предварително зареждане на модула *ipt_LOG*.

iptables –A FORWARD –p tcp –j LOG -–log-level warning

iptables –A INPUT –p tcp –j LOG –-log-prefix “INPUT packets”

Цел MASQUERADE

- Използва се за преобразуване на адреса на източника основно при dial-up или DHCP конекции, при които се използва динамично конфигуриране на IP адресите.
- Ако се използват статични IP адреси се препоръчва използването на целта SNAT.
- Като опция при TCP и UDP протоколи могат да се задават порт на източника или диапазон от портове, които ще се използват от изходните пакети.
- Тази цел може да се използва единствено за веригата POSTROUTING на **nat** таблицата.

iptables –t nat –A POSTROUTING –p tcp –j MASQUERADE

iptables –t nat –A POSTROUTING –p tcp –j MASQUERADE –– to-ports 1024-3000

Цел REDIRECT

- Използва за пренасочване на пакетите и потокът данни към самата машина.
- Адресът на получателя се преобразува в собствения loopback IP адрес.
- Тази цел е ефективно използвана в случаите на реализиране на прозрачна proxy услуга (*transparent proxy*), при която на машините от локалната мрежа не им известно наличието на proxy сървър.
- Целта е валидна единствено за веригите PREROUTING и OUTPUT на **nat** таблицата.
- За използването на тази цел е необходимо предварително зареждане на модула *ipt_redirect*.

iptables –t nat –A PREROUTING –p tcp –—dport 80 –j REDIRECT –-to-ports 8080

Цел REJECT

- Тази цел функционира аналогично на DROP, с разликата, че се изпраща обратно на източника ICMP съобщение за грешка: icmp-port-unreachable, icmp-proto-unreachable, icmp-net-unreachable, icmp-host-unreachable, icmp-net-prohibited или icmp-host-prohibited.
- Допълнителната опция **tcp-reset** за TCP протокола позволява затваряне на отворена TCP конекция чрез изпращане на RST сегмент към източника.
- Целта е валидна единствено за веригите INPUT, FORWARD и OUTPUT.

iptables –A FORWARD –p tcp -–dport 22 –j REJECT –-reject-with icmp-host-unreachable

Цел RETURN

- Преустановява преминаването на пакет през текущата верига.
- Ако това е подверига, пакетът ще продължи преминаването си през веригата от по-горното ниво.

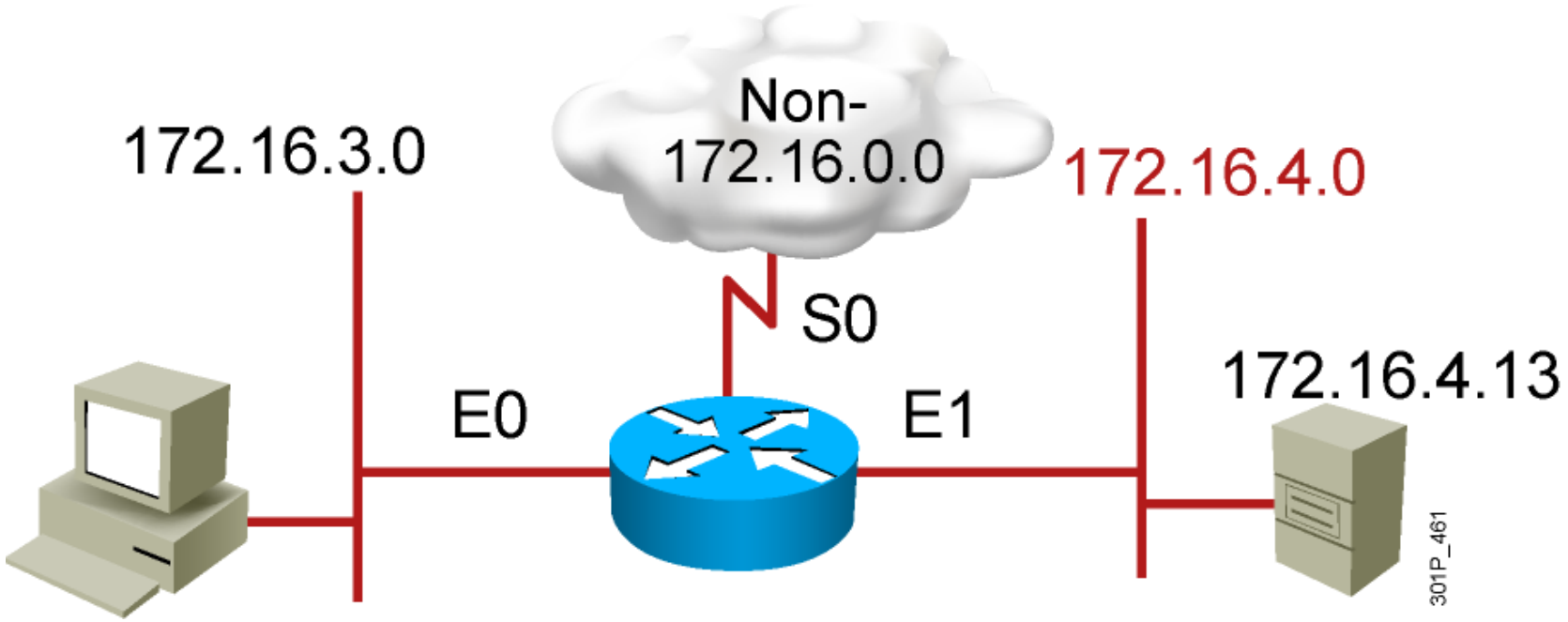
Цел SNAT

- Използва се за преобразуване на адреса на източника.
- Основното използване е в случаите, когато е необходим достъп от вътрешна мрежа с private IP адреси към Интернет.
- Целта променя IP адреса на източника от локалната мрежа с реалния IP адрес на firewall.
- Възможно е задаването на диапазон от IP адреси на източника, по този начин различните потоци данни ще използват отделни IP адреси на източниците.
- Като допълнение, за TCP и UDP протоколи може да се зададе диапазон от портове, към които ще се транслират портовете на източника.
- Единственото допустимо използване на целта е с веригата POSTROUTING на **nat** таблицата.

iptables –t nat –A POSTROUTING –p tcp –o eth0 –j SNAT –-to-source 200.100.50.1

iptables –t nat –A POSTROUTING –p tcp –o eth0 –j SNAT –-to-source 200.100.50.1-200.100.50.50:1024-30000

Пример

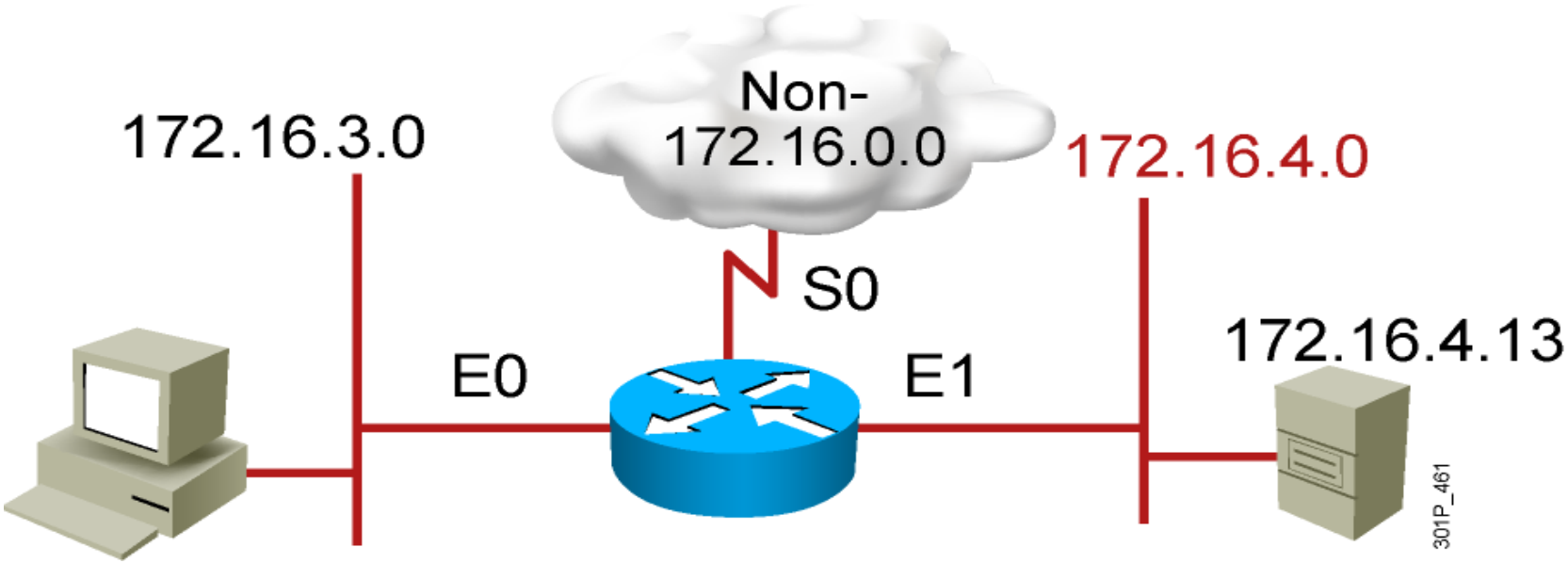


- Да се спира целият трафик от 172.16.3.0/24 до 172.16.4.0/24.
- Да се позволява целият друг трафик.

Решение

- Къде да се постави забраната на маршрутизатор или на крайно устройство?
- По-близо до сорса или по-близо до дестинацията?
- Кой е сорсът? Коя е дестинацията?
- Кой е протоколът, който пренася този трафик?
- Да се изтрият правилата във веригите.
- Да се зададат политиките по подразбиране за веригите.
- Да се опишат конкретните правила.

Решение



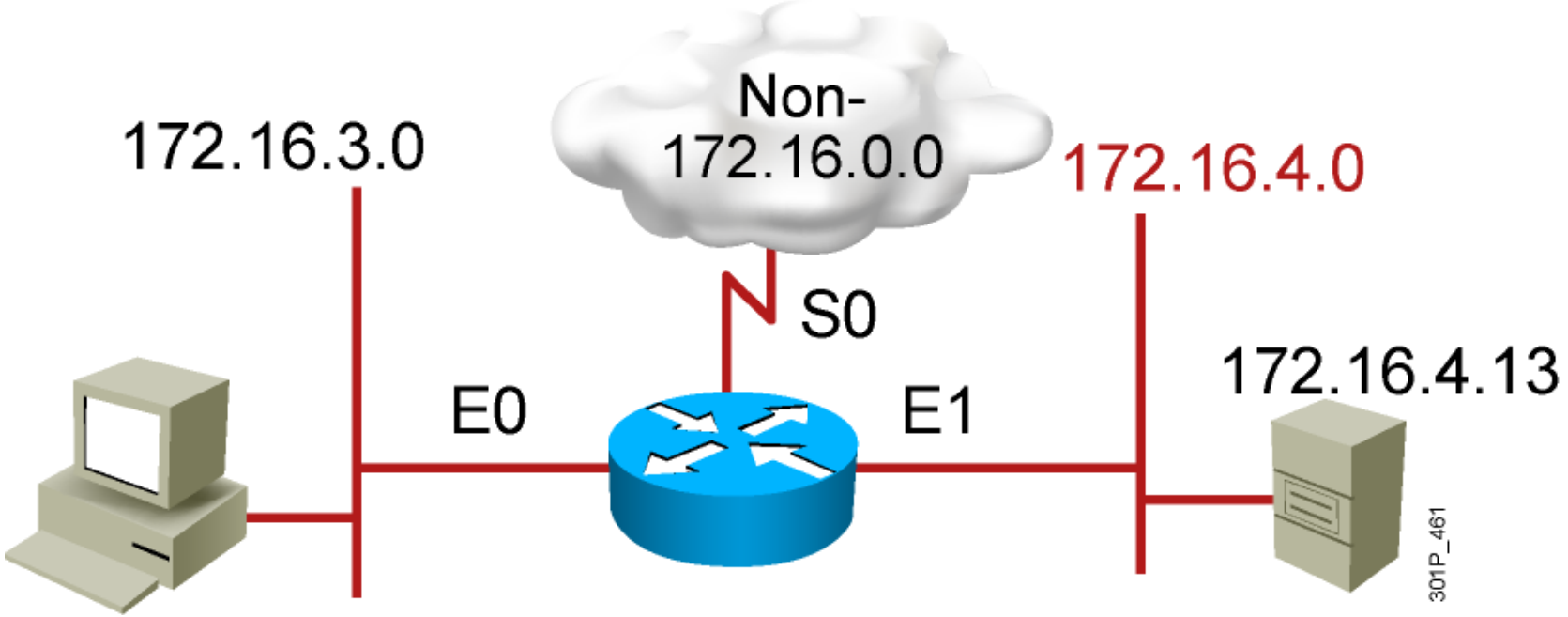
На маршрутизатора:

iptables –F FORWARD

iptables –P FORWARD –j ACCEPT

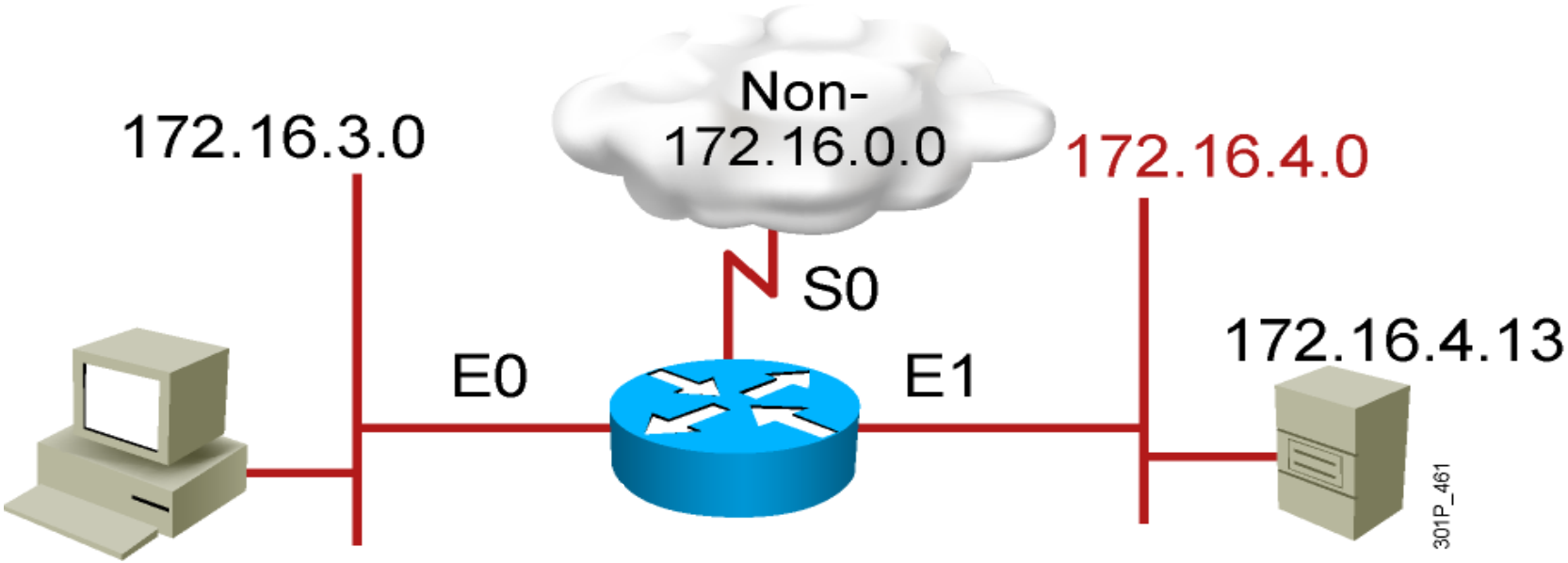
iptables –A FORWARD –s 172.16.3.0/24 –d \
172.16.4.0/24 –j DROP

Пример



- Да се позволява само FTP трафик от 172.16.3.0/24 до 172.16.4.13 , но не и друг трафик до тази машина.
- Да се позволява целият друг трафик.

Решение



На маршрутизатора:

iptables –F FORWARD

iptables –P FORWARD –j ACCEPT

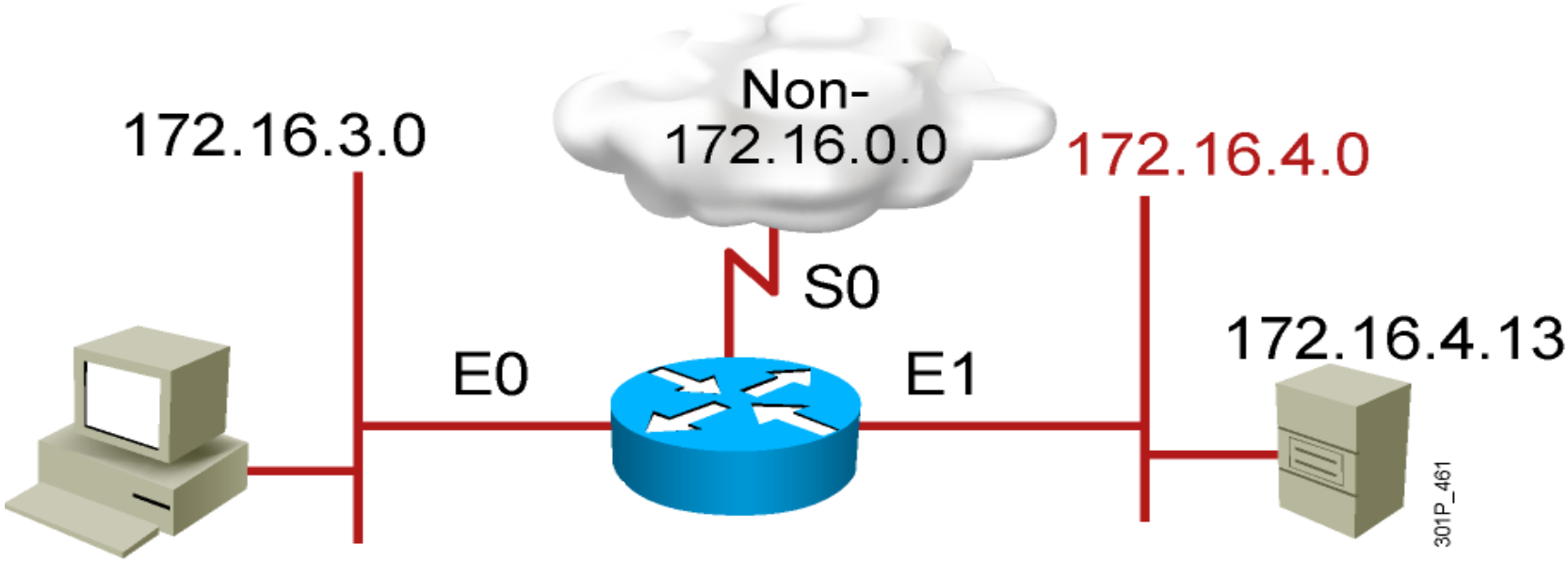
iptables –A FORWARD –s 172.16.3.0/24 –d 172.16.4.13 !--dport 20:21 -j DROP

На сървъра:

iptables –F INPUT

iptables –P INPUT –j ACCEPT

Решение



На сървъра:

iptables –F INPUT

iptables –P INPUT –j DROP

iptables –A INPUT –s 172.16.3.0/24 –d 172.16.4.13 --dport 20:21 –j ACCEPT

На маршрутизатора:

iptables –F FORWARD

iptables –P FORWARD –j ACCEPT

И други примери:

- **Защита от SYN-flood :**
  iptables -A INPUT -p tcp --syn -m limit --limit 1/s -j ACCEPT
  iptables -A INPUT -p tcp --syn -j DROP
- **Защита от Ping of death:**
  iptables -A INPUT -p icmp --icmp-type echo-request -m limit \
--limit 1/s -j ACCEPT
  iptables -A INPUT -p icmp --icmp-type echo-request -j DROP
- **Защита от сканиране на портове:**
  iptables -A INPUT -p tcp --tcp-flags SYN,ACK,FIN,RST RST \
-m limit --limit 1/s -j ACCEPT
  iptables -A INPUT -p tcp --tcp-flags SYN,ACK,FIN,RST RST \
-j DROP
- **Защита от IP- spoofing:**
  iptables -A INPUT -i eth1 -s 192.168.0.0/24 -j DROP
  iptables -A INPUT -i eth1 -s 10.0.0.0/8 -j DROP

Въпроси ?

Благодаря за вниманието !