Контрол на достъпа до услугите в Linux. Супер сървър inetd

проф. д-р инж. Венета Алексиева

Основни моменти

- Контрол на достъпа до услугите в Linux
  - Система със самостоятелно стартирани демони
  - Система с централен демон
- Стартиране на услуги
- Супер сървър inetd – конфигуриране
- Контролиране на достъпа до услугите
- Пример

Стартиране на сървърни услуги

Познати услуги:

- DHCP
- DNS
- Web сървър
- Mail сървър

Наблюдение на мрежата

Какво трябва да се наблюдава:

- Отворени портове
- Трансфер на съобщения
- Активни протоколи
- Стартирани услуги

Средства:

- tcpdump
- netstat
- wireshark

Контрол на достъпа до услугите

- Контролът на достъпа е техника за ограничаване на достъпа до съответни услуги.
- Ограничаването може да бъде по:
  - IP адрес на хост/мрежа
  - Име на хост, изискващ дадена услуга
  - Име на домейн
  - и др.
- Осъществява се на базата на контролен списък за достъп:
  - Ако в списъка е позволено на хоста да използва услугата, заявката се разрешава.
  - В противен случай тя се отхвърля.

Стартиране на услуги в Linux

- 2 начина за стартиране на услуги:
  - Система със самостоятелно стартирани демони
  - Система с централен демон

- Поддържането на услугите влияе пряко върху производителността на сървърните машини.

Да поразсъждаваме....

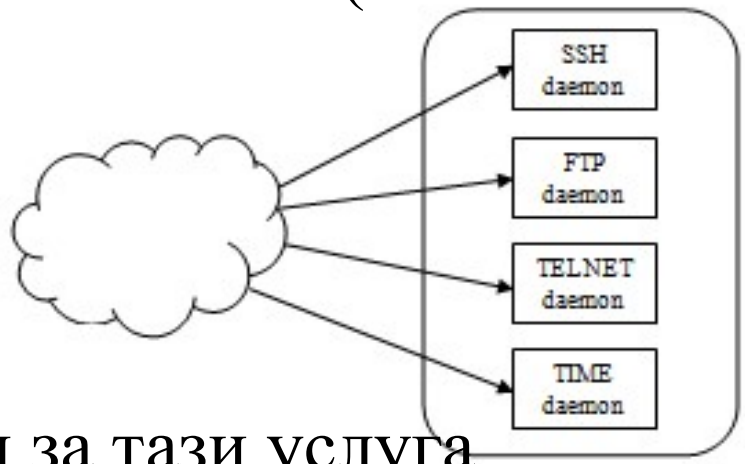
- Ако само върху една сървърна машина са инсталирани необходимите услуги:
  - синхронизиране на времето,
  - web сървър,
  - пощенски сървър,
  - разрешаване на отдалечен достъп,
  - свързване към бази данни и др.
- Услугите се реализират от изпълнявани във фонов режим програми – демони.
- Ефективно ли е и защо?

Система със самостоятелно стартирани демони

- Стартирани са десетки или стотици демони
- Всяка услуга се обслужва от свой демон:
  - след като обслужи процеса, заспива
  - в по-голямата част от времето очакват (а не изпълняват) заявки

  Недостатък:
  - заемат ресурс памет

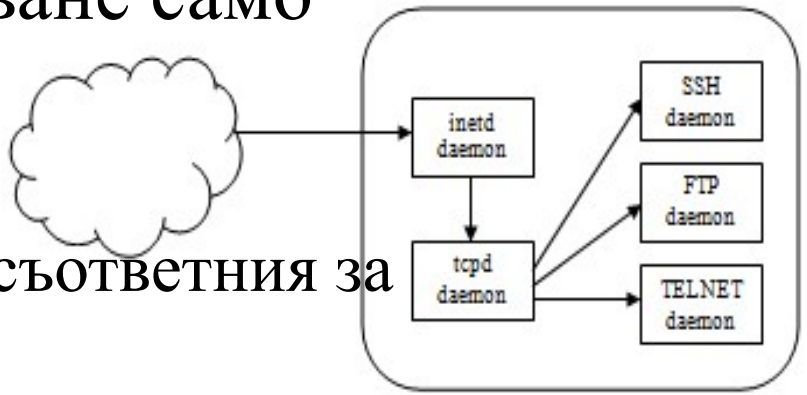
  Предимство:
  - сървърите са оптимизирани за тази услуга



- Не е ефективно – излишно разходване на памет

Система с централен демон

- Стартиран е един централен демон
- Под Linux това е супер-сървър – ***inetd***.
- Той очаква заявки по всички портове и стартира за тяхното обслужване само необходимите услуги:
  - Постъпва заявка;
  - Създава дъщерен процес със съответния за услугата сървър;
  - Предава му за обработка тази конекция;
  - След като обработи напълно заявката, завършва.



- По-ефективно – само 1 сървър е в стадий на обработка и не се дублира памет.

Знаем вече, че...

/etc/rc.d/rc.inet1.conf

- Конфигурира машината, за да бъде тя в мрежа:

```
# Configuration of eth0:
IPADDR[0]=""
NETMASK[0]=""
USE_DHCP[0]=""
DHCP_HOSTNAME[0]=""
GATEWAY=“
```

- Но къде се описват стартираните услуги?

/etc/rc.d/rc.inet2

- Стартира демоните, описани в него
- Могат да се стартират самостоятелни сървъри при инициализация на системата
- Пример:
  - inetd
  - sshd
  - bind
  - nfs
- Демони като httpd, mysql, samba се стартират от rc.M

Конфигуриране на демона inetd

/etc/inetd.conf

- кои услуги да бъдат приемани
- кои програми да бъдат стартирани за тяхното обслужване
- на кои сокети /портове слуша

/etc/services

- съответствия между имена на услуги и съответните им номера на портове

/etc/protocols

- използваните Интернет протоколи

Как работи *inetd*

- Създава сокети, зададени в конфигурационния файл
- Слуша едновременно на всички портове
- При заявка на някой порт стартира процес, който изпълнява услуга
- Изпълнява и вътрешни услуги като discard, daytime, time ...
- След изпълнението на услугата, завършва процеса

/etc/inetd.conf

service type protocol wait user server cmdline

- ***Service*** – Задава името на услугата (преобразува се в номер на порт, съгласно */etc/services)*
- ***Type*** –типа на използвания сокет (*stream* за TCP и *dgram* за UDP)
- ***Protocol*** –името на транспортния протокол (да е дефинирано в */etc/protocols*)
- ***Wait*** –само за dgram сокети:
  - *wait* - ***inetd*** стартира само един сървър за съответния порт, като изчаква неговото завършване за повторно обслужване на заявка
  - *Nowait* - ***inetd*** незабавно продължава да очаква заявки по същия порт след стартирането на услугата.

  При моногонишков сървър и *stream* сокет трябва да се задава *nowait*.
- ***User*** –името на потребителя, който ще бъде собственик на създадения дъщерен процес - *root*, *nobody*…
  - не се препоръчва задаването му, ако услугата изрично не го изисква.
- ***Server*** –пълния път до изпълнимата сървърна програма
- ***Cmdline*** –командният ред, който ще бъде предаден като параметър на стартирания сървър. - името на сървъра и аргументите за предаване.

Пример

```
service          type     protocol wait      user      server     cmdline

# discard        dgram    udp      wait      root      internal
# daytime        stream   tcp      nowait    root      internal
# daytime        dgram    udp      wait      root      internal
# chargen        stream   tcp      nowait    root      internal
# chargen        dgram    udp      wait      root      internal
# time           stream   tcp      nowait    root      internal
# time           dgram    udp      wait      root      internal
#.......
#
# ftp      stream   tcp      nowait root      /usr/sbin/tcpd      wu.ftpd -l -i -a
ftp        stream   tcp      nowait root      /usr/sbin/tcpd      proftpd
telnet     stream   tcp      nowait root      /usr/sbin/tcpd      in.telnetd
#
# telnet stream     tcp      nowait root      /usr/sbin/tcpd \
#   /usr/sbin/in.telnetd
#
```

/etc/services

service-name port-number/protocol-name [aliases]

Услуга номер на порт / трансп. протокол [2-ро име на услугата]

Пример:

```
service-name   port-number/protocol-name    [aliases]
ftp-data      20/tcp        #File Transfer [Default Data]
ftp-data      20/udp        #File Transfer [Default Data]
ftp           21/tcp         #File Transfer [Control]
ftp           21/udp         #File Transfer [Control]
#
ssh           22/tcp         #Secure Shell Login
ssh           22/udp         #Secure Shell Login
#
telnet        23/tcp
telnet        23/udp
#
smtp          25/tcp   mail  #Simple Mail Transfer
smtp          25/udp   mail  #Simple Mail Transfer
```

/etc/protocols

protocol-name number [aliases]

Име номер друго име

Пример:

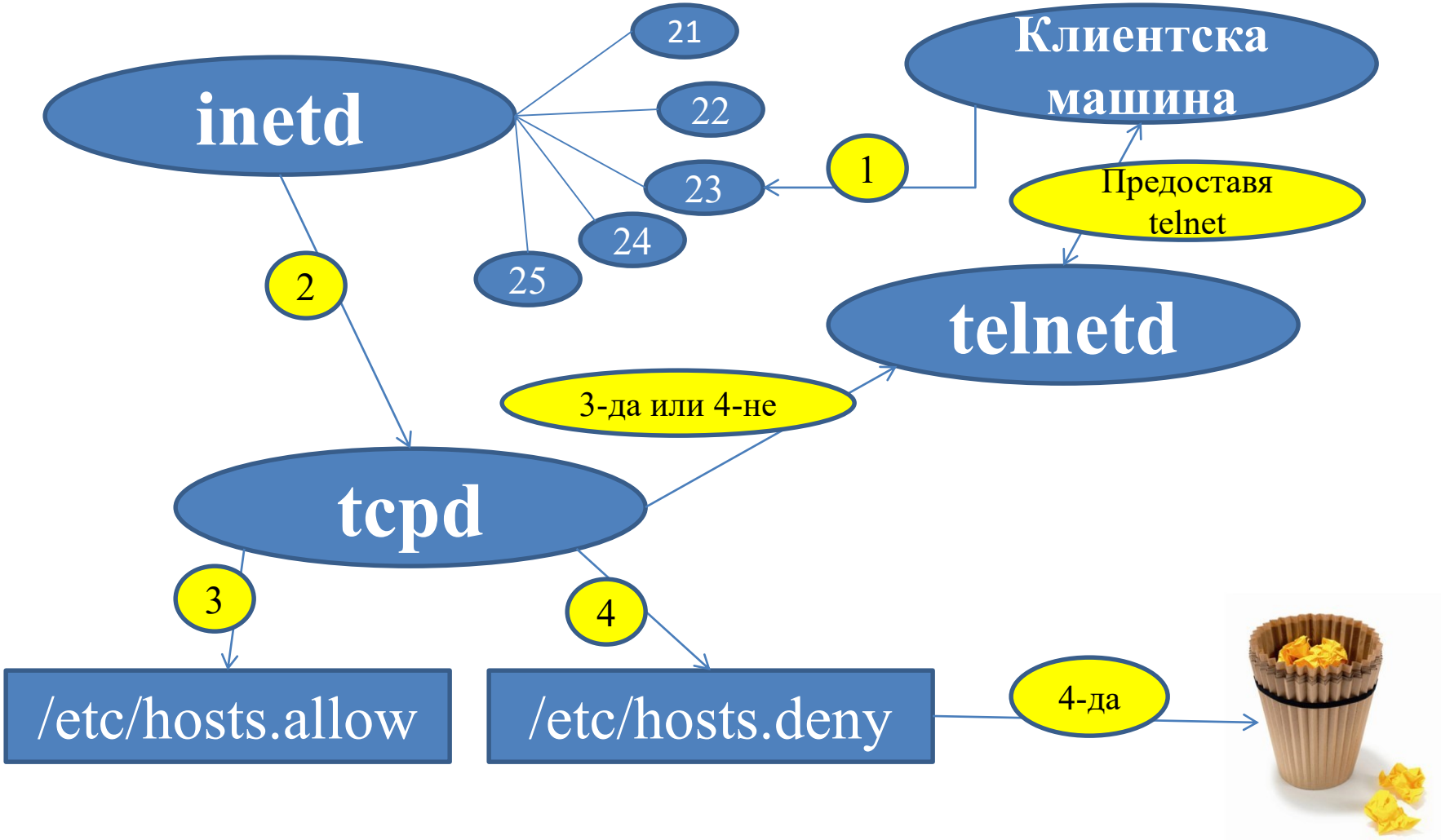
```
protocol-name  number  [aliases]
    ip         0       IP          # internet protocol
    icmp       1       ICMP        # internet control message protocol
    igmp       2       IGMP        # internet group management protocol
    ggp        3       GGP         # gateway-gateway protocol
    ipencap    4       IP-ENCAP    # IP encapsulated in IP
    st2        5       ST2         # ST2 datagram mode (RFC 1819)
    tcp        6       TCP         # transmission control protocol
```

Контрол на достъпа до услуги на ниво хост

- Демон, наречен „*TCP wrapper*”, реализиран от ***tcpd***:
  - Прилага механизъм за контрол на достъпа до услугите
  - Създава лог файл за опитите за свързване към услугите

1. Той се стартира при всяка заявка вместо изискваната сървърна програма.
2. Проверява дали на клиентския хост е разрешено да използва услугата.
   1. Ако да - стартира реалния сървър
   2. Ако не – не предприема действия
3. Същевременно, ***tcpd*** може да съхранява историята на заявките, изпращайки информацията към демона ***syslog***.
4. За реализиране на контрола на достъп от ***tcpd*** се използват два файла: */etc/hosts.allow* и */etc/hosts.deny*.

tcpd


inetd
21
22
23
24
25
Клиентска машина
1
Предоставя telnet
telnetd
2
3-да или 4-не
tcpd
3
4
/etc/hosts.allow
/etc/hosts.deny
4-да

/etc/hosts.allow и */etc/hosts.deny*

servicelist : host-list [: shellcmd]

- ***Servicelist*** –списък от имена на услуги, разделени със запетаи (съвпадат с имената от файла */etc/inetd.conf)*.
  - **ALL –** съвпада с всички
  - **ALL EXCEPT -** съвпадение с всички, но с изключение на посочените
- ***Host-list*** – Съдържа списък, разделен със запетаи:
  - имена на хостове (*www*)
  - имена на домейни (*.tu-varna.bg*),
  - IP адреси на хостове (*194.141.30.99*) или мрежи (*172.16.*).
  - **ALL-** съвпадение с всички имена на хостове и IP адреси
  - **ALL EXCEPT -**с изключение на определени
  - **LOCAL** –съвпадение с всички локални имена
  - **PARANOID** – с всички които са имена на хостове но няма DNS резолване
- ***Shellcmd*** –команда/скрипт от команди на *shell-a*. (Например информация към ***syslogd*** за опитите за конекция към дадена услуга)

Препоръчителна техника

Да се забрани достъпът до всички услуги от всички източници и само на позволените източници да се разреши.

 # /etc/hosts.deny
ALL: ALL

 # /etc/hosts.allow
proftpd: .tu-varna.bg
in.telnetd: 194.141.24.15, 194.141.25.

Въпроси ?

Благодаря за вниманието !