

Programowanie sieciowe - Laboratorium

LAB01

Termin „komputer PC” oznacza komputer z systemem Linux – może być to maszyna wirtualna.

Do wykonania ćwiczeń wymagane są komputery z systemem operacyjnym LINUX.

Proszę uruchomić komputer PC z system LINUX.

Zadanie 1. Sprawdzić konfigurację sieci na komputerze PC. Za pomocą komend 'ifconfig', 'netstat', 'route' i 'ip' znaleźć (za pomocą każdej komendy z osobna, jeśli to możliwe):

- jakiesą adresy IP i IPv6 (L3) komputera i maski adresów?(np.: ifconfig -a, ip addr show, netstat -i)
- jakie są adresy MAC (L2) komputera?(np.: ifconfig -a, ip link show, netstat -eia)
- jaki jest domyślny ruter (domyślna brama)?(np. route, ip ro, netstat -r)
- ile interfejsów sieciowych posiada komputer?
- które z komend umożliwiają wyświetlenie powyższych informacji i za pomocą jakich opcji?

Wypełnij tabelę - do wypełnienia tabeli mogą być pomocne doświadczenia z pozostałych zadań.

	ifconfig	netstat	ip	route
Odczyt adresu L3 (IP)				
Ustawienie adresu L3 (IP)				
Odczyt adresu L2 (MAC)				
Ustawienie adresu L2 (MAC)				
Odczyt tablicy routingu				
Zmiana tablicy routingu				
Liczba i nazwy interfejsów sieciowych				
Stany gniazd (ss, lsof)				
Liczba i nazwy interfejsów fizycznych	lspci, lsusb			

Zadanie 2. Proszę uruchomić dwa komputery PC podłączone do jednej sieci LAN:

2a. Za pomocą komendy **'ip'** dodać dodatkowy adres IPv4 10.2.10.x z maską 24 bitową, gdzie x jest najmniej znaczącym bajtem z istniejącego adresu IPv4. Przetestować osiągalność komputerów za pomocą programu ping.

2b. Za pomocą komendy **'ip'** dodać dodatkowy adres IPv6 `fc00:1:1:1::xx` z maską 64 bitową, gdzie xx są najmniej znaczącymi bajtami z istniejącego adresu. Przetestować osiągalność komputerów za pomocą programu ping6. Adresy z zakresu `fc00::/7` służą do użycia w sieciach prywatnych.

Zadanie 3. Skopiować programy `daytimetcpcliv6.c` i `daytimetcpsrvv6.c` do katalogu domowego użytkownika student na komputerze PC, a następnie skompilować poleceniem `gcc` lub `make`. (przykłady znajdują się na platformie UPEL lub w następującej lokalizacji: pluton.kt.agh.edu.pl/~gozdecki/PS_2020/LAB01)

Przykłady z serwera pluton na PC można skopiować poleceniem `scp` jeśli komputer ma dostęp do sieci Internet:

```
scp -r pluton.kt.agh.edu.pl/~gozdecki/PS_2020/LAB01 .
```

Uwaga: Punkty od a do c wykonać w konfiguracji:

1. komputer PC - komputer PC
2. na komputerze PC za pomocą interfejsu *loopback*
 - a) Uruchomić serwer (uwaga na prawa dostępu). Co należałoby zrobić, aby dało się uruchomić serwer z konta student na komputerze PC?
 - b) Uruchomić klienta z adresem IPv6 serwera jako parametrem - zaobserwować działanie programu.
 - c) Programem `netstat` i `ss` znaleźć gniazda utworzone w procesie serwera i klienta.

Zadanie 4. (Dla pierwszych 3 osób dodatkowy 1 pkt za aktywność w czasie trwania laboratorium)
Przerobić programy na IPv4 (zmiana stałych `AF_INET6` na `AF_INET`, struktur adresowych `sockaddr_in6` na `sockaddr_in`, `servaddr.sin_addr.s_addr = htonl(INADDR_ANY);`) i przetestować działanie.

Zadanie 5. Testowanie programu ss. Zestawić sesję ssh z serwerem `pluton.kt.agh.edu.pl` i sesję ssh z pomiędzy komputerami PC używając adresu IPv6. Przetestować opcje `-6 -4 -n -t -u -p -a -r -l -m -i` programu `ss`. Za pomocą komendy `ss` wyświetlić na komputerze PC:

- a) gniazda w systemie dla protokołu IPv6 z portem źródłowym 22
- b) gniazda dla protokołu IPv4 z portem źródłowym lub docelowym 22
- c) gniazda dla połączeń z serwerem `pluton.kt.agh.edu.pl`

- d) gniazda programów sshd i ssh
- e) gniazda w stanie ESTABLISHED i TIME_WAIT

Zadanie 6. Testowanie programu netstat. Zestawić sesję ssh z serwerem pluton.kt.agh.edu.pl i sesję ssh pomiędzy komputerami PC używając adresu IPv6. Przetestować opcje -6 -4 -a -u -t -n -p -v -e -c 1 komendy netstat. Za pomocą komend netstat (i ew. grep) wyświetlić na komputerze PC (Uwaga: ze zwykłego użytkownika nie wszystkie informacje są wyświetlane):

- a) gniazda dla protokołu IPv6 (np. ss -6)
- b) gniazda dla protokołu IPv4 (np. ss -4)
- c) gniazda TCP dla protokołu IPv6 i PID procesów, które utworzyły te gniazda (ss -t6p)
- c) gniazda UDP dla protokołu IPv6 i PID procesów, które utworzyły te gniazda (ss -6up)
- d) gniazda programów sshd i ssh (np. lsof -c ssh -i -a)
- e) gniazda TCP i UDP w stanie LISTEN (np. ss -ut state listening, ss -utl)

Zadanie 7. Testowanie programu lsof. Zestawić sesję ssh z serwerem pluton.kt.agh.edu.pl i sesję ssh pomiędzy komputerami PC używając adresu IPv6. Przetestować opcje programu lsof z wykładu #1. Za pomocą komendy lsof wyświetlić na komputerze PC:

- a) gniazda w systemie dla protokołu IPv6
- b) gniazda dla protokołu IPv4
- c) gniazda w systemie dla protokołu IPv6, których właścicielem jest użytkownik 'root'
- d) gniazda TCP dla protokołu IPv6 i PID procesów, które utworzyły te gniazda
- e) gniazda programów sshd i ssh
- f) gniazda w stanie LISTEN

Zadanie 8. Testowanie programu tracepath/traceroute.

- a). Znaleźć parametr MTU (*Maximum Transfer Unit*) dla ścieżki od komputera PC i serwera pluton do węzłów o adresach: 212.191.224.69, ae3.mx1.fra.de.geant.net, 2001:798:23:10aa::9, time-d.nist.gov. Dla każdego adresu, jeśli to możliwe, użyć komendy dla protokołu IPv4 i IPv6.
- b). Znaleźć routery na ścieżce i opóźnienia (minimalne, maksymalne i średnie) do tych routerów od komputera PC (dodatkowo od serwera pluton) do węzłów o adresach: 212.191.224.69,

ae3.mx1.fra.de.geant.net, 2001:798:23:10aa::9, time-d.nist.gov
(time.nist.gov), www.onet.pl. Dla każdego adresu, jeśli to możliwe, użyć komendy dla protokołu IPv4 i IPv6. Dla IPv4, w przypadku, gdy routery na ścieżce nie odpowiadają użyć opcji 'icmp'

Zadanie 9. Za pomocą programu Wireshark i komendy "tcpdump -i eth0 -n" prześledzić wymianę pakietów pomiędzy serwerem i klientem (programy daytimetcpcliv6.c i daytimetcpsrvv6.c). Serwer uruchomić na jednym komputerze i łączyć się z drugiego komputera. Do odfiltrowania niepotrzebnych wyników dla komendy tcpdump użyć filtrów 'port', 'host', 'src' lub 'dst', np.:

```
tcpdump -i eth0 -n port 13
```

Do przygotowania na następne zajęcia (LAB02):

1. Wiadomości z LAB01!
2. Wiadomości z wykładów 1 i 2.
3. Funkcje getsockopt() i setsockopt() - do czego służą, jakie mają parametry wejściowe i wyjściowe?
4. Zapoznać się z opcjami gniazd:
 - SO_RCVBUF, SO_SNDBUF, SO_KEEPALIVE, SO_BINDTODEVICE, SO_REUSEADDR, SO_LINGER, SO_RCVLOWAT, SO_SNDLOWAT, SO_DEBUG, SO_DONTROUTE, SO_OOBINLINE,
 - IP_TOS, IP_TTL, IP_MTU
 - IPV6_UNICAST_HOPS, IPV6_MTU, IPV6_MTU_DISCOVER, IPV6_DONTFRAG
 - TCP_MAXSEG, TCP_NODELAY, TCP_KEEPALIVE