Załącznik D

Iproute2 i Wireless Tools

1 Pakiet iproute2

W skład pakietu *iproute2* wchodzą programy: *tc* oraz *ip*. Pierwszy udostępnia szereg opcji do klasyfikowania, określania priorytetów, współdzielenia oraz limitowania przyznawanego pasma zarówno dla ruchu wychodzącego, jak i przychodzącego. Natomiast program *ip* łączy w sobie cechy narzędzi *ifconfig*, *route* oraz *arp*, używanych dotychczas do konfiguracji Linuksa w roli routera. Program *ip* składa się z kilku modułów. Aby zapewnić poprawne działanie *iproute2*, jądro wykorzystywanego Linuksa musi być co najmniej w wersji 2.2.

1.1 Moduł link

Moduł *link* służy do ustawiania parametrów karty sieciowej. Po wpisaniu z linii komend *ip link*, należy podać jedną z poniższych opcji:

- *set DEV up/down* – aktywuje/wyłącza dany interfejs (np. eth0),
- *set DEV arp on/off* - włącza/wyłącza używanie protokołu ARP (Address Resolution Protocol). Jego wyłączenie powoduje, że host nie odpowiada na zapytania ARP, czyli praktycznie nie jest widoczny w sieci, gdyż nie można poznać adresu fizycznego jego (tj. hosta) interfejsu,
- *set DEV txqueuelen <LICZBA>* - definiuje kolejkę pakietów oczekujących na wysłanie na określoną liczbę pozycji;
- *set DEV multicast on/off* - zezwala/ zabrania interfejsowi sieciowemu na odbieranie i wysyłanie pakietów multicastowych;
- *set DEV name <NAZWA>* - umożliwia zastąpienie nazwy interfejsu (np. eth0) dowolną inną;
- *set DEV address <adres_MAC>* - zmiana adresu MAC interfejsu,
- *show [DEV]* - wyświetla informacje o urządzeniach sieciowych podłączonych do systemu lub w przypadku podania nazwy urządzenia (np. ip link show ppp0) wyświetla tylko jego charakterystykę. Użycie dodatkowo parametru –s pozwala na podgląd statystyk interfejsu (*ip -s link show DEV*).

Wyświetlane, przez polecenie *ip link show [DEV]*, pole *link* oznacza adres warstwy łącza danych (*link layer address*), a pole *ether* – typ sieci (*Ethernet*).
Znaczenie flag *PROMISC* (interfejs odbiera cały ruch na łączu, zastosowanie: monitorowanie sieci) i *ALLMULTI* (urządzenie odbiera wszystkie ramki multicastowe, zastosowanie: routery

multicastowe) wyświetlanych przez polecenie *ip* i *ifconfig* jest odmienne. Polecenie *ip link ls* pokazuje rzeczywisty stan urządzeń, gdy tymczasem *ifconfig*, pokazuje stan, jaki był obecny podczas definowania ich poleceniem *ifconfig*.

Przykłady:

ip link set eth1 up – podniesienie interfejsu eth1
ip link set eth1 address 00:00:00:00:00:01 – zmiana adesu MAC interfejsu eth1
ip link set eth1 name wireless – zmiana nazwy interfejsu na "wireless"
ip link show eth0 – podgląd atrybutów urządzenia
ip –s link show eth0 – podgląd atrybutów urządzenia z opcją *statistic*
ip –s -s link show eth0 – podgląd atrybutów urządzenia z wielokrotną opcją *statistic*

1.2 Moduł addr

Moduł *addr* umożliwia nadawanie adresów sieciowych danym interfejsom. Po wpisaniu z linii komend *ip addr*, należy podać jedną z poniższych opcji:

- *add ADRES [OPCJE] dev DEV* - przypisuje adres IP urządzeniu. Dostępne [OPCJE] to:

  - *broadcast <ADRES>* - ustawia adres broadcastowy,
  - *anycast <ADRES>* - ustawia adres anycast (dostępne tylko w IPv6),
  - *label <NAZWA>* - umożliwia przypisanie nazwy adresowi IP,
  - *peer* | *remote* - dotyczy interfejsów PPP, parametr ADRES uzupełniamy lokalnym numerem IP, a adres drugiego końca łącza PPP podajemy po opcji peer lub remote;
  - *scope host* | *link* | *global* – służy do wyboru zasięg adresu.

Należy tu zwrócić uwagę na notację zapisywania numerów IP wraz z przypisaną długością maski podsieci. Program *ifconfig* wymagał podawania po adresie IP maski podsieci w formacie 255.255.0.0. Natomiast w *ip* jej długość zapisywana jest jako suma jedynek z zapisu maski w formacie binarnym, czyli powyższej masce odpowiada długość /16, a np. 255.255.255.0 odpowiada /24.
Podczas przypisywania adresu do adaptera sieciowego dodawane są automatycznie zapisy do tabeli routowania, określające trasy do hosta oraz do sieci, do której należy.

- *del ADRES [OPCJE] dev DEV* - to samo, co wyżej, tylko usuwa numer IP z podanego urządzenia sieciowego. Wymaga podania wszystkich parametrów, które zostały wpisane przy dodawaniu adresu.
- *show* [*DEV]* - wyświetla informacje o dostępnych w systemie urządzeniach sieciowych wraz z ich parametrami, flagami, adresami.

Przykłady:

ip addr show eth0 – przeglądanie adresów interfejsu eth0
ip addr add 127.0.0.1/8 dev lo – dodawanie adresu IP (loopback) wraz z maską podsieci
ip addr add 10.0.0.1/24 brd + dev eth0 label eth0 – przypisanie adresu 10.0.0.1 z prefiksem 24 bitów oraz adresu broadcastowego urządzeniu eth0. Utworzenie aliasu o nazwie eth0
ip addr del 127.0.0.1/8 dev lo – usuwanie adresu. Nazwa urządzenia jest wymagana, pozostałe parametry są opcjonalne. Jeżeli nie podano argumentów, usuwany jest pierwszy z adresów.
ip –s –s a flush to 10/8 – wyczyszczenie adresów z prywatnej podsieci 10.0.0.0/8
ip -4 addr flush label eth* - wyczyszczenie adresów IPv4 we wszystkich interfejsach Ethernet

1.3 *Moduł* neigh

Obsługuje tzw. *tablicę sąsiadów*. Hosty komunikujące się ze sobą w sieci lokalnej muszą znać swoje adresy IP, a także unikalne adresy sprzętowe kart sieciowych. Do ich wykrywania służy protokół ARP, który powoduje, że gdy host chce skomunikować się z innym, wysyła pakiet rozgłoszeniowy do wszystkich hostów w sieci, z umieszczonym w nim adresem IP docelowego hosta. Ten, gdy go odbierze, wysyła do komputera, który zadał pytanie pakiet z odpowiedzią zawierającą adres sprzętowy. Znając adresy fizyczne swoich kart sieciowych, oba hosty mogą się teraz wzajemnie komunikować. Jądra każdego z nich przechowują tablice sąsiednich hostów, które są po prostu bazą danych adresów IP komputerów w sieci wraz z odpowiadającymi im adresami sprzętowymi adapterów sieciowych.

Do wykrywania urządzeń w sieci służy także algorytm wykrywania dostępności hosta NUD (ang. *Neighbor Unreachability Discovery*). Metoda ta jest wykonywana cyklicznie co jakiś czas i aktualizuje stan tabeli o sąsiednich hostach. Oprócz adresów tabela zawiera informacje

o liczbie odwołań oraz czasie ostatniego wywołania rekordu. Spotykane stany rekordów to: *permanent* (rekord dodany ręcznie przez administratora), *incomplete* (żaden z hostów nie odpowiedział na zapytanie o dany adres), *reachable* (adres osiągalny), *stale* (adres osiągalny, ale rekord wymaga uaktualnienia), *noarp* (rekord nieuaktualniany przez ARP). Opcje dla parametru *neigh*:

- *add ADRES [OPCJE] dev DEV* - przypisanie adresowi IP adresu sprzętowego danego interfejsu. Dostępne opcje to:

  - *lladdr LLADDR* - umożliwia ręczne wpisanie adresu fizycznego. W przypadku kart Ethernet zapis *lladdr* możemy pominąć, gdyż każda z nich ma już taki adres - unikalny w skali światowej i ustalany przez producenta karty,
  - *nud permanent|noarp| stale|reachable* - nadanie rekordowi jeden ze stanów opisanych powyżej.
  - *proxy PROXY* - określenie adresu hosta, dla którego nasz interfejs będzie pośredniczył w wymianie zapytań ARP.

- *del ADRES [OPCJE] dev DEV* - usuwa dany rekord z tablicy sąsiadów. Wymaga wszystkich parametrów podawanych przy dodawaniu rekordu.
- *show* - wyświetla tablicę hostów sąsiadujących.

Przykłady:

ip neigh add 10.0.0.3 lladdr 0:0:0:0:0:1 dev eth0 nud perm – przypisanie adresu WŁD do tablicy ARP dla obiektu o adresie IPv4 10.0.0.3 zdefinowanego jako urządzenie eth0. Wpis o stan sąsiedztwa (perm, permanent) jest zawsze ważny i może być zmieniany przez reguły administracyjne.
ip neigh chg 10.0.0.3 dev eth0 nud reachable – zmiana stanu na reachable (wpis o sąsiedztwie jest ważny do czas uwygaśniecia czasu reachability.
ip neigh del 10.0.0.3 dev eth0 – usunięcie wpisu z tablicy ARP dla obiektu o adresie IPv4 10.0.0.3, zdefiniowanego jako urządzenie eth0.
ip neigh show – przeglądanie wpisów o sąsiedztwie
ip –s n show 193.233.7.254 – przeglądanie wpisów o sąsiedztwie z opcją – statistics
ip –s –s n f 193.233.7.254 – kasowanie wpisu o sąsiedztwie

1.4 Moduł route

Służy do manipulacji tablicami routingu. Jądra Linuksa o wersjach 2.0.x utrzymywały w pamięci tylko jedną tablicę trasowania. W nowszych wersjach istnieje możliwość korzystania z max. 250 różnych tablic routowania, przy czym zawsze dostępne są trzy z nich:

- *default* (253) - domyślna tablica, początkowo pusta;
- *main* (254) - odpowiednik starej tablicy z jądra 2.0, umieszczane są w niej trasy dodawane przez administratora, jeśli nie zadeklaruje innej tablicy. Znajdują się tu także trasy dodawane automatycznie przez jądro podczas aktywacji interfejsu.
- *local* (255) - zawiera trasy dodawane przez jądro, np. trasy broadcastowe czy trasy do lokalnych interfejsów hosta.

Do manipulacji tablicami służy szereg opcji podawanych po poleceniu *ip route* (w skrócie ip ro):

- *add <ADRES> <JAK> src <IP>* - dodaje trasę do tablicy routowania. W najprostszym przypadku *ADRES* zawiera adres sieci lub hosta docelowego, *JAK* określa sposób trasowania (najczęściej poprzez parametr "via"), a opcja *src IP* powoduje, że pakiety wychodzące tą trasą będą miały adres źródłowy podany przez nas w parametrze IP - należy tu podać jeden z adresów, który należy do naszego hosta.
- *del <ADRES> <JAK>* - usuwa routing na *<ADRES>*. Wymagane jest podanie wszystkich opcji użytych przy dodawaniu trasy.
- *show* - wyświetla aktualną tablicę routingu;
- *monitor* - wyświetla na ekranie wszelkie zmiany zaistniałe w tablicach routingu; przydatne przy testowaniu dynamicznego routingu.

Przykłady:

ip route show – przeglądanie tablicy routingu,
ip ro chg 10.0.0/24 dev dummy – zmiana routingu z powyższego przykładu poprzez interfejs o nazwie „dummy”,
ip route add 172.16.0.0/16 via 192.168.1.254 - dodaje do głównej tablicy routingu trasę do sieci 172.16.0.0/16 prowadzącą przez router 192.168.1.254,
ip route add 0/0 via 192.168.1.254 - pakiety o adresie przeznaczenia nieodpowiadające żad-

nemu rekordowi w tablicy routowania kierowane są do bramki 192.168.1.254 (tzw. trasa domyślna),

ip route add 192.168.0.0/24 dev eth1 table 3 - dopisuje do tabeli routowania 3 regułkę przesyłania pakietów do sieci 192.168/24 poprzez interfejs eth1,

ip route add unreachable 192.168.1.0/24 - pakiety skierowane do sieci 192.168.1.0/24 zostaną odrzucone, a nadawca otrzyma komunikat ICMP "Host unreachable",

ip route add nat 192.168. 2.0/24 via 194.204.56.0/24 table local - uaktywnia translację adresów docelowych. Każdy adres z sieci prywatnej 192.168.2.0 zostanie odwzorowany na adres publiczny z sieci 194.204.56.0 (maski podsieci muszą być takie same).

ip route add nat 192.168.0.2 via 195.113.148.34 - spowoduje, że pojedynczy adres 192.168.0.2 będzie widoczny "na zewnątrz" jako 195.113.148.34.

ip -4 –s –s ro flush cache – usuwanie wszystkich duplikatów routingu dla IPv4 z podwójną opcją statistic.

1.5 Moduł rule

Moduł *rule* służy do manipulacji tabelami reguł, które odnoszą się do różnych tabel routowania. Reguły zezwalają na wybieranie tras na podstawie adresu docelowego, adresu źródłowego, rodzaju usługi (ang. *Type of Service*) lub rodzaju interfejsu, na którym odbieramy pakiet. Wymaga skompilowania jądra z uaktywnionymi opcjami:

- "IP: advanced router"
- "IP: policy routing"
- "IP: fast network address translation"

Parametry polecenia rule:

- *add SELEKTOR AKCJA* - dodaje regułkę, zdefiniowaną na podstawie SELEKTORA i podejmuje odpowiednią AKCJĘ. Dostępne selektory:

  - *from* - adres źródłowy pakietu,
  - *to* - adres docelowy pakietu,
  - *tos* - typ usługi (Type of Service, wymaga zaznaczenia przed kompilacją jądra opcji

"IP: use TOS value as routing key"),
- *iif* - interfejs, na który przychodzi pakiet;
- *fwmark* - oznakowanie nadane przez firewall (wymaga kompilacji kernela z opcją "IP: use netfilter MARK value as routing key").

Selektory te można łączyć w różne kombinacje, co daje spore możliwości routowania. AKCJA to sposób potraktowania pakietu pasującego do danej regułki. Najczęściej jest to skierowanie pakietu do odpowiedniej tablicy routowania, gdzie zostanie podjęta decyzja, którą drogą go wysłać. Możliwe też są inne akcje, np. odrzucanie pakietów na różne sposoby z poinformowaniem nadawcy lub nie, albo tzw. translacja adresów źródłowych.

Przykłady:

ip rule add iif eth0 table 32 - dla pakietów przychodzących na interfejs eth0 zostanie użyta tabela routowania nr 32,
ip rule add from 1.2.3.4 table 6 - dla pakietów wysłanych z hosta 1.2.3.4 użyta będzie tabela trasująca nr 6,
ip rule add from 119.231. 45.0/24 nat 192.168.1.0/24 - adresy źródłowe pakietów z sieci 119.231.45.0/24 są tłumaczone na adresy sieci 192.168.1.0/24.

Istnieje również użyteczna funkcja znakowania pakietów przez firewall w celu kierowania oznakowanego ruchu odpowiednią trasą:

iptables -A PREROUTING -i eth0 -t mangle -p tcp --dport 25 -j MARK --set-mark 1 - cała wychodząca poczta jest oznakowana przez firewall,
ip rule add fwmark 1 table mail.out - reguła, która kieruje pakiety zaznaczone przez firewall do przetwarzania przez tablicę routowania o nazwie mail.out,
ip route add default via 191.56.38.23 dev ppp0 table mail.out - poczta wychodzi przez urządzenie ppp0 na bramkę o adresie 191.56.38.23.

- *del* - kasuje regułkę z tabeli regułek,
- *list* - wyświetla listę wszystkich reguł (domyślnych i zdefiniowanych przez administratora)

1.6 Moduł tunnel

Moduł *tunnel* umożliwia stawianie tunelu między sieciami. Przypuśćmy, że mamy dwie sieci, z których pierwsza to 192. 168.1.0/24, o wewnętrznym adresie routera 192.168.1.1 i zewnętrznym 1.1.1.1, druga zaś to 192.168.2.0/24, wewnętrzny adres routera to 192.168.2.1, a zewnętrzny - 2.2.2.2. Aby zestawić tunel między nimi, poprzez Internet w routerze pierwszej należy wpisać:

ip tunnel add netB mode gre remote 2.2.2.2 local 1.1.1.1 ttl 255
ip addr add 192.168.1.1 dev netB
ip route add 192.168.2.0/24 dev netB

W pierwszej linii utworzony został interfejs tunelujący o nazwie *netB* oraz adresy internetowe routerów na końcach tunelu, które prowadzą do sieci wewnętrznych. Druga linia nadaje adres nowo powstałemu interfejsowi *netB*, trzecia natomiast dodaje routing do sieci drugiej przez interfejs tunelujący *netB*.

Konfiguracja routera sieci drugiej wygląda prawie identycznie:

ip tunnel add netA mode gre remote 1.1.1.1 local 2.2.2.2 ttl 255
ip addr add 192.168.2.1 dev netA
ip route add 192.168.1.0/24 dev netA

Używanie *ip tunnel* wymaga wcześniejszego załadowania modułu *ip_gre.o*. W podobny sposób można zestawiać tunele w sieciach IPv6.

Więcej informacji:

- man pages,
- http://www.linuxgrill.com/iproute2-toc.html

2 Pakiet Wireless Tools

Pakiet *Wireless Tools(WT)* to zestaw narzędzi do konfiguracji parametrów radiowych karty bezprzewodowej w trybie tekstowym.

2.1 *Polecenie iwconfig*

Polecenie *iwconfig* to główne narzędzie pakietu WT. Umożliwia ustawienie następujących parametrów karty bezprzewodowej:

- tryb pracy karty (*mode*) - *ad-hoc* lub *managed*,
- identyfikator sieci (*ESSID*),
- kanał operacyjny (*channel*),
- prędkość transmisji (*rate*),
- czułość odbiornika (*Sensitivity*),
- klucz WEP (*key*),
- próg RTS (*rts*),
- próg fragmentacji (*frag*),
- adres MAC punktu dostępowego (*ap*),
- nazwę karty (*nick*).

Składnia polecenia – *iwconfig [DEV] <parametr> <wartość>*

Przykłady:

iwconfig wlan0 mode ad-hoc – ustawienie interfejsu *wlan0* do pracy w trybie *ad-hoc*
iwconfig wlan0 mode managed - interfejs *wlan0* pracuje w trybie *infrastructure*
iwconfig wlan0 essid wlab – ustawienie identyfikatora sieci na ‘wlab’
iwconfig wlan0 channel 6 – ustawienie numeru kanału na 6
iwconfig wlan0 sens -80 – ustawienie czułości karty na -80 dBm
iwconfig wlan0 rate 11M – ustawienie szybkości transmisji na 11Mbit/s
iwconfig wlan0 rts 250 – ustawienie progu RTS/CTS
iwconfig wlan0 frag 512 – ustawienie progu fragmentacji
iwconfig wlan0 enc 0123-4567-89 – ustawienie klucza WEP (hex)

2.2 Polecenie iwlist

Polecenie *iwlist* służy do skanowania sieci oraz do uzyskania informacji o dostępnych parametrach interfejsu bezprzewodowego. Dostępne parametry to:

- *freq* – wyświetla listę częstotliwości/kanałów, które karta obsługuje,
- *ap* – wyświetla listę punktów dostępowych w obrębie swojego zasięgu,
- *rate/bit* – wyświetla listę prędkości transmisji obsługiwanych przez kartę,
- *key/enc* – podaje jakie długości klucza WEP obsługiwane są przez kartę; wyświetla również aktualnie ustawione klucze WEP,
- *power* – podaje jakie tryby zarządzania mocą ma wbudowane karta,
- *txpower* – wyświetla listę dostępnych mocy nadawania karty,
- *retry* – wyświetla limity retransmisji.

Składnia polecenia – *iwlist [DEV] <parametr>*

2.3 Polecenie iwspy

Polecenie *iwspy* umożliwia utrzymywanie listy adresów IP oraz MAC urządzeń, o których chcemy mieć informację o jakości połączenia, siły sygnału oraz poziomu szumu do tych urządzeń.

Składnia polecenia – *iwspy [DEV] [+] adres_IP | adres MAC*

Przykłady:

iwlist wlan0 – wyświetla listę adresów IP oraz MAC monitorowanych urządzeń,
iwlist wlan0 + 192.168.1.10 | 00:20:30:40:50:60 – dodaje adres IP oraz MAC monitorowanego urządzenia,
iwlist wlan0 off – skasowanie listy monitorowanych urządzeń i wyłączenie funkcji spy

2.4 *Polecenie* iwpriv

Polecenie *iwpriv* umożliwia konfigurację dodatkowych, ukrytych parametrów interfejsu bezprzewodowego. Za pomocą narzędzia *iwpriv* możliwe jest również włączenie roamingu (o ile jest obsługiwany) oraz konfiguracja portów.

Składnia polecenia – *iwpriv [DEV] ukryta _komenda ukryty_parametr*
iwpriv [DEV] roam on/off
iwpriv [DEV] port ad-hoc/managed/N

Użycie polecenia *iwpriv* tylko z parametrem *[DEV]* wyświetla listę ukrytych komend oraz odpowiadających im parametrów.

2.5 Polecenie iwgetid

Polecenie *iwgetid* jest stosowane do wyświetlenia informacji o identyfikatorze ESSID oraz innych dostępnych informacji o aktualnie używanej sieci bezprzewodowej.

Składnia polecenia – *iwgetid [DEV]*
iwgetid [DEV] - -scheme

Parametr *sheme* jest używany gdy ma zostać wyświetlona informacja tylko o identyfikatorze – ESSID. Jest to bardzo przydatna funkcja przy pisaniu skryptów.

Więcej informacji:

- http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html
- **man pages** – iwconfig, iwlist, iwspy, iwpriv, iwgetid