Sieci komputerowe – diagnostyka

Teoria

Istnieje wiele programów służących do diagnostyki sieci komputerowych oraz pobierających informacje z baz danych i pozwalających sprawdzić, kto odpowiada za adresy bądź nazwy w Internecie. Sama diagnostyka sieci jest jednak problemem dość złożonym i z tego powodu trudno o oprogramowanie, które w ogólnym przypadku byłoby w stanie bez ingerencji użytkownika zdiagnozować i wskazać konkretny problem i jego przyczynę.

Poniżej znajduje się opis poleceń (programów konsolowych) dostępnych w systemie Linux, które pozwalają zweryfikować ustawienia sieci oraz przeprowadzić podstawową diagnostykę. Wiele z tradycyjnych poleceń sieciowych (jak *ifconfig*, *route*, *arp* czy *netstat*), wchodzących w skład przestarzałego już pakietu *net-tools*, zostało zastąpionych w nowoczesnych dystrybucjach Linuksa przez potężny pakiet *iproute2* (polecenia *ip* oraz *ss*).

***ip addr* (skrót: *ip a*)**

Polecenie to służy do wyświetlania oraz modyfikacji konfiguracji interfejsów sieciowych. Podstawowe parametry konfiguracyjne, jakie można uzyskać przy jego pomocy, to:

- adres IP oraz maska podsieci (zapisane w formacie CIDR, np. /24),
- adres rozgłoszeniowy (brd – broadcast),
- adres sprzętowy karty sieciowej (MAC – oznaczony jako link/ether),
- stan interfejsu (np. UP, DOWN).

***ip route* (skrót: *ip r*)**

Polecenie to służy do wyświetlania oraz modyfikacji tablicy trasowania (rutingu). W przypadku, gdy nie korzystamy z wirtualnych interfejsów (VPN, Docker itp.), tablica rutingu zazwyczaj składa się z dwóch elementów:

- adres bramy domyślnej, czyli wyjście z sieci lokalnej do Internetu lub innych sieci (np. `via 192.168.1.1`);
- adres sieci, który określa trasę do sieci lokalnej (np. `192.168.1.0/24 dev enp0s3 ... proto kernel ... src 192.168.1.100`).

***ip neighbor* (skrót: *ip n*)**

Polecenie to służy do wyświetlania oraz modyfikacji tablicy sąsiadów. Tablica ta przechowuje adresy IP oraz mapowane na nie adresy sprzętowe (MAC) urządzeń w tej samej podsieci, z którymi nasz komputer w ostatnim czasie prowadził komunikację. Stany wpisów w tablicy mogą przyjmować wartości takie jak REACHABLE (urządzenie osiągalne i potwierdzone), STALE (wpis starszy, wymaga potwierdzenia przy kolejnej próbie wysyłki) czy FAILED (nie udało się ustalić adresu MAC).

Inne zastosowania polecenia *ip*

Za pomocą polecenia *ip* można również w prosty sposób konfigurować interfejsy z poziomu wiersza poleceń (z uprawnieniami administratora), np.:

```
sudo ip addr add 192.168.1.100/24 dev enp0s3
sudo ip link set enp0s3 up
```

W ten sposób ustawiamy adres IP na 192.168.1.100 oraz maskę podsieci na 255.255.255.0 (24 bity o wartości 1) dla interfejsu sieciowego o nazwie enp0s3 (pierwsze polecenie) oraz włączamy ten interfejs (drugie polecenie).

ping

Polecenie ping pozwala sprawdzić, czy możliwa jest komunikacja z określonym adresem IP. Program wysyła komunikaty *ICMP Echo Request* pod zadany adres. System operacyjny otrzymujący taki komunikat odpowiada pakietem *ICMP Echo Reply*. Program *ping* liczy i wyświetla czas odpowiedzi oraz wartość TTL (Time to Live).

Brak komunikacji w sieci może być spowodowany różnymi przyczynami. Jeśli powodem jest błędna konfiguracja lub awaria sprawiająca, że jeden z ruterów w sieci nie ma informacji o tym, jak dostarczyć pakiet pod zadany adres IP, odsyła on adekwatny do tej sytuacji komunikat ICMP. Informację o odebraniu takiego komunikatu wyświetla program ping, pozwalając zorientować się co do miejsca i przyczyny awarii. Brak odpowiedzi na komunikat może być również spowodowany tym, że system operacyjny docelowego komputera (lub pośrednicząca zapora sieciowa/firewall) został poinstruowany, aby takiej odpowiedzi nie udzielać.

Informacje na temat uzyskanej odpowiedzi wyświetlane przez program ping mogą posłużyć do oceny liczby przeskoków dzielących komputery oraz oceny jakości i stanu łącza. Pierwsza informacja może być odczytana z wartości TTL, która jest zmniejszana o 1 przez każdy pośredniczący ruter. Czasy uzyskania odpowiedzi (tzw. *round-trip time*) zależą od liczby przeskoków i długości połączeń sieciowych. W przypadku łącz typu WAN ich wartość powinna być rzędu kilkudziesięciu milisekund, a w przypadku sieci LAN – kilku milisekund lub mniej. Ważny jest również rozrzut tych czasów (stabilność łącza).

Warto wiedzieć, że polecenie ping występuje również w innych popularnych systemach operacyjnych: Windows i macOS.

traceroute

Polecenie to służy do wyznaczania trasy w sieci prowadzącej do stacji z określonym adresem. Trasa opisana jest poprzez listę adresów (symbolicznych bądź IP) ruterów, przez które wykonywane są kolejne przeskoki do określonego celu. Poza kolejnymi adresami program wypisuje też czasy odpowiedzi uzyskane od kolejnych węzłów – dla każdego z nich pomiar wykonywany jest trzykrotnie.

traceroute pozwala zorientować się, w którym miejscu sieci występuje problem (np. przy którym ruterze urwało się połączenie) w przypadku awarii lub błędnej konfiguracji. Czasy podawane przez program pozwalają też zgrubnie ocenić obciążenie poszczególnych segmentów sieci. Należy pamiętać, że trasy w Internecie wyznaczane są dynamicznie i mogą zmieniać się w czasie. Mogą też występować sytuacje (tzw. ruting asymetryczny), w których trasa w jednym kierunku prowadzi inną drogą (poprzez inny zestaw ruterów) niż w kierunku przeciwnym.

mtr

Polecenie *mtr* (*My Traceroute*) jest nowoczesną, interaktywną alternatywą łączącą funkcjonalność programów ping i *traceroute*. Za jej pomocą możemy uzyskać w czasie rzeczywistym informacje o procentowej ilości zgubionych pakietów dla każdego węzła trasy (kolumna *Loss*) oraz wartości statystyczne, takie jak średnia (*Avg*), odchylenie standardowe (*StDev*) czasów odpowiedzi, a także

najlepsze (*Best*) i najgorsze (*Worst*) zarejestrowane czasy. Program po uruchomieniu działa w pętli do momentu jego przerwania przez użytkownika.

***ss* (Socket Statistics)**

Polecenie *ss* jest narzędziem służącym do badania gniazd sieciowych, które składają się z adresu IP i numeru portu. Pozwala zorientować się, z jakimi gniazdami uruchomione programy nawiązały połączenia oraz jakie porty są otwarte na nasłuchiwanie (listening). Warto wypróbować następujący zestaw opcji:

- `-t` – wyświetla połączenia TCP;
- `-u` – wyświetla połączenia UDP;
- `-l` – wyświetla gniazda nasłuchujące (waiting for connections);
- `-a` – wyświetla wszystkie gniazda (zarówno nasłuchujące, jak i nawiązane);
- `-p` – wyświetla nazwę i numer identyfikacyjny (PID) procesu, który korzysta z danego połączenia (wymaga uprawnień administratora – sudo).

Opcje można łączyć w jednym wywołaniu, np. `ss -tulpn` (opcja `-n` zapobiega zamianie numerów portów i adresów IP na nazwy symboliczne, co znacznie przyspiesza działanie polecenia).

host

Dzięki temu poleceniu, podając jako parametr adres symboliczny (domenę), uzyskamy odpowiadający mu adres IP i odwrotnie. Plecenie host odpytuje serwery DNS w celu uzyskania wyniku. Odpytywanie innych, konkretnych rekordów DNS możliwe jest dzięki opcji `-t` określającej typ poszukiwanego rekordu. Np. `host -t MX prz.edu.pl` zwróci rekordy MX dla podanej domeny, czyli informację o serwerach pocztowych obsługujących jej pocztę przychodzącą.

hostname

Polecenie *hostname* umożliwia odczytanie lub (przy użyciu uprawnień root/sudo) ustawienie nazwy hosta danego urządzenia w systemie Linux.

whois

Polecenie *whois* służy do odpytywania publicznych baz danych WHOIS, zawierających informacje o rejestratorach, abonentach lub osobach odpowiedzialnych za daną domenę internetową albo pulę adresów IP. Uruchamiając narzędzie, należy podać jako parametr nazwę domeny lub konkretny adres IP, o którym chcemy uzyskać szczegółowe informacje.

wget

Za pomocą polecenia *wget* możemy pobrać dowolny plik z adresu internetowego podanego jako parametr, bezpośrednio z wiersza poleceń, bez użycia przeglądarki graficznej. Domyślnie pobierany plik zapisywany jest w bieżącym katalogu roboczym.

tcpdump

Polecenie *tcpdump* jest potężnym snifferem sieciowym pozwalającym analizować ruch w sieci, tzn. przechwytywać i wyświetlać informacje o pakietach przesyłanych do i ze wszystkich urządzeń połączonych z interfejsem, na którym prowadzimy nasłuch. Dostępne są informacje o pakietach, które „widzi" wybrana karta sieciowa. *tcpdump* jest nieocenionym narzędziem administratora sieci przy rozwiązywaniu złożonych problemów z komunikacją.

Domyślnie program wypisuje na ekranie informacje o nagłówkach przechwyconych pakietów, jednak w sieciach o dużym natężeniu ruchu ilość danych może być zbyt duża do analizy w czasie rzeczywistym.

Z tego względu program pozwala na stosowanie rozbudowanych filtrów (np. ograniczających nasłuch do konkretnego portu, protokołu lub adresu IP) oraz zapis przechwyconych pakietów do pliku (np. `-w plik.pcap`) w celu późniejszej, szczegółowej analizy. Informacje o pakietach uzyskane za pomocą polecenia *tcpdump* to m.in.:

- czas przechwycenia pakietu,
- nazwa protokołu sieciowego,
- adresy IP źródłowy i docelowy wraz z numerami portów,
- flagi protokołu TCP (np. [S] - SYN, [P] - PUSH, [F] - FIN, [.] - ACK),
- numery sekwencyjne i numery potwierdzeń,
- rozmiar okna w protokole TCP,
- dodatkowe opcje i długość danych payload.

Zadania do wykonania w systemie Linux

Wykonaj zadania i zapisz ich rezultaty (polecenia i wyniki) w sprawozdaniu, np. w LibreOffice Writer. Przed zakończeniem zajęć przekaż sprawozdanie prowadzącemu.

Uwaga: Niektóre polecenia używane w poniższych zadaniach (np. *ping*, *mtr*, *tcpdump*) działają w trybie ciągłym przez nieograniczony czas. Można je w dowolnym momencie natychmiastowo zakończyć skrótem klawiszowym Ctrl + C.

1. Sprawdzenie podstawowej konfiguracji interfejsów

Wykonać polecenie:

```
ip addr
```

(lub w skrócie `ip a`).

Sprawdzić nazwy interfejsów w systemie (np. lo, enp0s3, eth0, wlan0), przydzielony interfejsowi sieciowemu adres IP oraz maskę podsieci (zapisane razem, np. 192.168.1.15/24), adres rozgłoszeniowy (oznaczony jako *brd*, np. 192.168.1.255) oraz adres sprzętowy MAC (oznaczony po słowie *link/ether*). Porównać z informacjami uzyskanymi przez innych studentów na swoich stacjach roboczych. Które parametry (np. prefiks sieci, adres MAC, końcówka adresu IP) są takie same, a które są różne i dlaczego?

2. Sprawdzenie zewnętrznego adresu IP

Serwisy internetowe takie jak WhatIsMyIP lub ifconfig.me pozwalają sprawdzić adres IP, z jakim nasz komputer widoczny jest w publicznym Internecie. Otworzyć w przeglądarce stronę https://www.whatismyip.com. Jaki adres IP wyświetlono? Czy jest to ten sam adres, który został wyświetlony w poprzednim zadaniu poleceniem `ip addr`? Wyjaśnić przyczynę ewentualnych różnic (wskazówka: pojęcie prywatnych adresów IP i mechanizmu NAT na ruterze brzegowym). Porównać swój adres zewnętrzny z wynikami innych studentów w sali.

3. Sprawdzenie łączności sieciowej

Poleceniem:

```
ping www.google.com
```

Sprawdzić łączność ze zdalnym serwerem. Zwrócić uwagę na czasy odpowiedzi (*time=* w milisekundach) i wartość *ttl=*. Przerwać działanie programu kombinacją Ctrl + C i porównać statystyki (np. średni czas) z wynikami innych studentów. Następnie wykonać polecenie ping na nieistniejący lub nieosiągalny

adres IP w sieci lokalnej (np. 192.168.1.254 lub 10.1.1.5). Jaki jest wynik polecenia po kilku sekundach prób? Jakie komunikaty o błędach są wyświetlane?

4. Sprawdzanie trasy w sieci

Poleceniem:

```
traceroute google.com
```

Wyświetlić trasę do serwera *google.com*. Sprawdzić, jakie są czasy poszczególnych przeskoków na kolejnych ruterach i ile jest wszystkich przeskoków do celu. Powtórzyć polecenie kilkakrotnie. Czy za każdym razem trasa i numery IP węzłów są dokładnie takie same? Czy występują wpisy zawierające jedynie znaki *? Takie wpisy oznaczają, że dany ruter „ukrywa się" (firewall) lub ignoruje *traceroute* (ograniczony priorytet odpowiadania na *traceroute*).

Poleceniem:

```
mtr -b onet.pl
```

Wyświetlić interaktywną trasę do serwera www.onet.pl. Sprawdzić, czy są jakieś węzły, przy których pakiet został zgubiony (kolumna *Loss%*). Sprawdzić maksymalne (*Wrst*), minimalne (*Best*) i średnie (*Avg*) czasy odpowiedzi oraz ich odchylenie standardowe (*StDev*). Opcja `-b` powoduje, że oprócz nazw symbolicznych węzłów wyświetlają się w nawiasach również ich adresy IP. Wyjść z programu klawiszem q lub skrótem Ctrl + C.

5. Sprawdzenie konfiguracji rutingu

Poleceniem:

```
ip route
```

(lub w skrócie: `ip r`*)* Wyświetlić lokalną tablicę rutingu. Zidentyfikować wiersz zaczynający się od słowa default – jaki jest adres IP bramy domyślnej (po słowie via) oraz przez który interfejs (po słowie dev) wysyłane są pakiety w świat?

6. Uzyskiwanie adresu IP na podstawie adresu symbolicznego i na odwrót

Za pomocą polecenia `host <adres>`:

1. Uzyskać adresy IP, na które rozwija się adres *onet.pl*.
2. Wykonać to samo dla adresu *google.com*.
3. Sprawdzić, na jaki adres symboliczny rozwija się adres IP 212.77.98.9.
4. Sprawdzić, na jaki adres symboliczny rozwija się adres IP 185.31.24.68.

7. Sprawdzanie nazwy hosta

Za pomocą polecenia:

```
hostname
```

Sprawdzić skonfigurowaną w systemie nazwę hosta swojego komputera.

8. Uzyskiwanie informacji z bazy WHOIS

Jeśli program *whois* nie jest domyślnie zainstalowany w systemie, należy go zainstalować za pomocą polecenia (wymagane będzie hasło do konta root/sudo, które poda prowadzący):

```
sudo apt update && sudo apt install whois
```

Na początku wpisujemy polecenie *sudo* oznaczające wykonanie polecenia z uprawnieniami administratora. Kolejna część polecenia aktualizuje polecenie *apt* służące do instalowania i zarządzania programami. Ostatnia część polecenia instaluje brakujący program *whois*.

Sprawdzić za pomocą polecenia:

```
whois wp.pl
```

informacje o domenie:

1. Kto jest rejestratorem domeny wp.pl i kiedy domena została utworzona?
2. Kiedy utworzono wpis i kiedy dokonano ostatniej jego modyfikacji?

Następnie odpytać bazę o jeden z adresów IP należących do Wirtualnej Polski (najpierw sprawdź adres IP poleceniem `host wp.pl`):

```
whois <adres_ip_domeny_wp.pl>
```

3. Do jakiej sieci (o jakiej nazwie *netname* i o jakim zakresie adresów *inetnum*) należy ten adres IP?
4. Kto jest właścicielem/dysponentem tej podsieci (pola *descr* / *organization*) i z jakiego jest kraju (*country*)?
5. Z jakim adresem e-mailowym należy się kontaktować w przypadku zgłaszania nadużyć (pole *abuse-mailbox*) z tego zakresu adresowego?

9. Wyświetlanie tablicy sąsiadów (zastępuje dawne ARP)

Poleceniem:

```
ip neighbor
```

(lub w skrócie: `ip n`) Wyświetlić adresy IP oraz sprzętowe (MAC - po słowie lladdr) urządzeń w sieci lokalnej, z którymi nasz komputer ma w pamięci podręcznej nawiązaną relację. Następnie poprosić kolegę ze stanowiska obok, aby wykonał z terminala ping na nasz adres IP (lub wykonać ping na adres IP komputera kolegi) i wywołać polecenie `ip neighbor` ponownie. Jeśli komunikacja w sieci lokalnej nastąpiła, adres IP oraz adres MAC komputera kolegi powinny pojawić się na liście ze stanem REACHABLE lub STALE.

10. Wyświetlanie aktywnych połączeń sieciowych i gniazd

Poleceniem:

```
ss -tuln
```

Wyświetlić listę wszystkich gniazd w systemie nasłuchujących na połączenia (`-l`) dla protokołów TCP (`-t`) i UDP (`-u`) bez zamiany numerów portów na nazwy usług (`-n`). Zwrócić uwagę na kolumny *Local Address:Port*.

Następnie wykonać polecenie z uprawnieniami administratora, aby wyświetlić aktywne (nawiązane) połączenia i programy, które je obsługują:

```
sudo ss -tap
```

Wyświetli to listę połączeń TCP (`-t`), włączając gniazda nawiązane i nasłuchujące (`-a`), wraz z informacją o numerze PID i nazwie programu (`-p`), który wywołał dane połączenie. Zwrócić uwagę na różnice między stanami połączeń (np. ESTAB – nawiązane połączenie, LISTEN – oczekiwanie na połączenie).

11. Pobieranie pliku z wiersza poleceń

Skopiować adres internetowy (URL), z którego została pobrana lub pod wierszem którego udostępniona jest niniejsza instrukcja. Następnie pobrać ją przy użyciu polecenia:

```
wget <adres_url>
```

Wskazówka: Jeśli pliku nie uda się pobrać z powodu błędu weryfikacji certyfikatu SSL/TLS, można użyć opcji pomijającej sprawdzanie certyfikatu:

```
wget --no-check-certificate <adres_url>
```

Należy używać tej opcji wyłącznie wtedy, gdy mamy pewność, jaki plik znajduje się pod podanym adresem.

Plik pobierze się do katalogu bieżącego, którego pełną ścieżkę możemy sprawdzić poleceniem `pwd`. Aby upewnić się, że plik został poprawnie pobrany i sprawdzić jego rozmiar, należy skorzystać z polecenia `ls -lh`. Na koniec usuń pobrany plik za pomocą polecenia `rm <nazwa_pliku>`.

12. Analiza ruchu w sieci przy użyciu sniffera *tcpdump*

Sprawdzić działanie polecenia do przechwytywania pakietów:

```
sudo tcpdump -i any -c 20
```

Parametr `-i any` oznacza nasłuch na wszystkich dostępnych interfejsach w systemie, a opcja `-c 20` spowoduje, że program zakończy działanie automatycznie po przechwyceniu dokładnie 20 pakietów.

Może się zdarzyć, że pierwsze wyniki pojawią się dopiero po kilkudziesięciu sekundach, więc trzeba cierpliwie poczekać.

Należy przeanalizować uzyskane linie logu. Na początku są zawsze znaczniki czasu przechwycenia pakietu. Następnie po spacji jest typ protokołu (np. IP). Kolejnym elementem są adresy źródłowy i docelowy oddzielone znakiem >. Na końcu, po dwukropku wypisane są flagi TCP i/lub dodatkowe informacje (więcej w opisie teoretycznym w pierwszej części instrukcji).

Aby przetestować filtrowanie ruchu, uruchomić nasłuch ograniczony wyłącznie do komunikacji ICMP (np. pakiety programu ping):

```
sudo tcpdump -i any icmp
```

W drugim oknie terminala wywołać polecenie `ping 8.8.8.8` (adres serwera systemu nazw domenowych DNS od Google) i zaobserwować w oknie z działającym *tcpdump* przechwytywane na żywo pakiety *echo request* oraz *echo reply*. Przerwać nasłuch kombinacją klawiszy Ctrl + C.