Warsztaty z Sieci komputerowych
Lista 1

Tutorial 1 (0 pkt.)

Otwórz znajdujący się na stronie wykładu dokument *Maszyna wirtualna Virbian* i przeczytaj jego pierwszą sekcję. Wykorzystując obraz maszyny z katalogu `/opt`, utwórz maszynę wirtualną *Virbian0* z domyślną konfiguracją sieciową i uruchom ją.

Znak `Vi$>` oznacza wykonanie danego polecenia w konsoli maszyny *Virbiani* z uprawnieniami zwykłego użytkownika. Natomiast znak `Vi#>` oznacza konieczność wykonania polecenia z prawami administratora. W tym celu należy uprzednio zalogować się na konto użytkownika `root` albo poprzedzić takie polecenie komendą `sudo`. Poniższe zadanie należy wykonać w uruchomionej maszynie wirtualnej *Virbian0*.

- Poleceniem

  ```
  V0$> ip addr
  ```

  wyświetl wszystkie dostępne interfejsy sieciowe. Powinny być dostępne dwa interfejsy: `lo` i `enpxsy`, gdzie x i y są liczbami naturalnymi. Interfejs `enpxsy`, będziemy nazywać `enp0`; w wydawanych poleceniach należy oczywiście podstawić faktyczną nazwę interfejsu.

- Uzyskaj konfigurację sieciową dla maszyny wirtualnej poleceniem

  ```
  V0#> dhclient -v enp0
  ```

  Ponownie wykonaj polecenie

  ```
  V0$> ip addr
  ```

  i sprawdź, że wyświetlana informacja zmieniła się i karta maszyny wirtualnej ma teraz przypisany adres IP równy `10.0.2.15` (lub podobny).

- Uruchom przeglądarkę Firefox. Z menu wybierz polecenie *View* | *Sidebar* | *HTTP Header Live* wyświetlające w pasku bocznym przeglądarki wysyłane i odbierane nagłówki HTTP. Wejdź przeglądarką na stronę http://www.ii.uni.wroc.pl/~mbi/dyd/ i obejrzyj przesyłane nagłówki protokołu HTTP. Ile żądań HTTP jest wysyłanych? Do jakich serwerów są one skierowane?

- Sprawdź jaki jest adres IP związany z adresem `www.ii.uni.wroc.pl` poleceniem

  ```
  V0$> host -t a www.ii.uni.wroc.pl
  ```

Niech *w.x.y.z* będzie tym adresem IP. Uruchom program Wireshark i włącz w nim obserwację interfejsu `enp0` klikając dwukrotnie na jego nazwie. Aby odfiltrować wyświetlanie zbędnych pakietów w polu *Apply a display filter ...* wpisz `ip.addr == w.x.y.z` i kliknij przycisk *Apply* (niebieska strzałka po prawej stronie tego pola). W razie potrzeby możesz również kliknąć ikonę *Restart current capture* (jedna z pierwszych ikon od lewej na górze okna programu).

Odśwież przeglądarką oglądaną stronę naciskając `Shift` + `Ctrl` + `R`. W Wiresharku wśród wysyłanych pakietów znajdź ten zawierający żądanie HTTP pobierające stronę HTML. Obejrzyj w tym pakiecie nagłówki warstwy sieciowej (IP) i transportowej (TCP). Klikając poszczególne pola opisu, podświetlasz w widoku szesnastkowym pakietu (na dole okna) odpowiadające im bajty. Które części pakietu zawierają powyższe nagłówki? Jaki jest źródłowy i docelowy adres IP tego pakietu? Jaki jest jego źródłowy i docelowy port? W których nagłówkach znajdują się te informacje?

Powtórz powyższe operacje dla pakietu zawierającego odpowiedź HTTP (powinien zawierać kod odpowiedzi `200 OK` wraz ze stroną w HTML lub kod odpowiedzi `304 Not Modified`). Czy dane identyfikujące połączenie (źródłowy/docelowy adres/port) zmieniły się czy są takie same? Dlaczego?

► W rozszerzeniu HTTP Header Live obejrzyj jeszcze raz żądanie HTTP wysyłane w momencie pobierania strony http://www.ii.uni.wroc.pl/~mbi/dyd/. Po kliknięciu przycisku *File Save* zawartość okna zapisze się w pliku `HTTPHeaderLive.txt` (w katalogu `Downloads`). Zmień zawartość tego pliku, tak żeby zawierał tylko nagłówki żądania HTTP pobierającego stronę HTML i następujący po nich pusty wiersz. Następnie zmień w pierwszym wierszu napis

```
http://www.ii.uni.wroc.pl/~mbi/dyd/
```

na

```
GET /~mbi/dyd/ HTTP/1.1
```

Otrzymany plik powinien zawierać zapytanie HTTP podobne do:

```
GET /~mbi/dyd/ HTTP/1.1
Host: www.ii.uni.wroc.pl
User-Agent: ...
...
<wiersz-odstępu>
```

Wyślij to zapytanie do serwera WWW (tj. do portu `80` adresu IP związanego z nazwą `www.ii.uni.wroc.pl`) poleceniem

```
V0$> nc -q 3 www.ii.uni.wroc.pl 80 < HTTPHeaderLive.txt
```

Opcja `-q 3` czeka 3 sekundy przed zamknięciem połączenia. Na ekranie wyświetli się odpowiedź serwera WWW, ale będzie ona nieczytelna dla człowieka. Problematyczny okazuje się wiersz `Accept-Encoding: gzip, deflate` proszący serwer WWW o kompresję przesyłanych danych. Usuń ten wiersz z pliku `HTTPHeaderLive.txt` i spróbuj ponownie. Obejrzyj przesyłane pakiety w Wiresharku.

- Sprawdź, czy uzyskasz odpowiedź, jeśli w pliku `HTTPHeaderLive.txt` pozostawisz jedynie dwa pierwsze wiersze (zaczynające się od `GET` i `Host:`) i następujący po nich pusty wiersz. Ponownie obejrzyj pakiety w Wiresharku. Co stanie się, jeśli zostawisz tylko pierwszy wiersz i wiersz odstępu?

- Poleceniem

  ```
  V0$> telnet www.ii.uni.wroc.pl 80
  ```

  otwórz strumień danych do serwera WWW na komputerze `www.ii.uni.wroc.pl`. Wpisz tam zapytanie HTTP, czyli wiersze

  ```
  GET /~mbi/dyd/ HTTP/1.1
  Host: www.ii.uni.wroc.pl
  ```

  a następnie pusty wiersz. W odpowiedzi otrzymasz kolejny raz powyższą stronę WWW.

- Poleceniami

  ```
  V0$> netstat -l46n
  V0$> netstat -l46
  ```

  wyświetl uruchomione na Twoim komputerze usługi „przypięte" do konkretnych portów warstwy transportowej. Pierwsze polecenie wyświetla wartości numeryczne, drugie zaś stara się je interpretować wykorzystując plik `/etc/services` (obejrzyj ten plik).

  Uruchom serwer SSH poleceniem

  ```
  V0#> systemctl start ssh
  ```

  i ponownie wyświetl listę usług poleceniami `netstat`.

- Wybierz kilka lokalnych usług wykorzystujących protokół TCP, w tym usługę SSH (port 22), serwer echa (port 7) i serwer czasu (port 13). Za pomocą programu `telnet` połącz się z nimi w interaktywny sposób i wyślij do tych usług jakieś dane. Przykładowo z portem 7 połączysz się poleceniem

  ```
  V0$> telnet localhost 7
  ```

  Nazwa `localhost` zostanie zamieniona na adres IP maszyny wirtualnej, w której aktualnie pracujesz, tzn. powyższe polecenie utworzy połączenie z działającą lokalnie usługą (serwerem echa) „przypiętą" do portu 7. Aby rozłączyć się, naciśnij kombinację `Ctrl` + `]` i następnie wpisz polecenie `quit`.

Na końcu zamknij maszynę wirtualną *Virbian0*.

Tutorial 2 (0 pkt.)

To zadanie należy wykonywać w parach z osobą siedzącą przy komputerze obok. Przeczytaj drugą sekcję z dokumentu *Maszyny wirtualne*. Na tej podstawie utwórz maszynę wirtualną *Vir-*

bian1, której wirtualna karta sieciową będzie zmostkowana z interfejsem `remote2`. Po uruchomieniu otrzymasz wirtualną maszynę, której karta sieciowa jest podłączona wirtualnym kablem do karty sieciowej wirtualnej maszyny osoby siedzącej obok.

- Wyświetl dostępne interfejsy sieciowe poleceniami

  ```
  V1$> ip link
  V1$> ip addr
  ```

  Aktywne interfejsy oznaczone są napisem `UP`, nieaktywne — `DOWN`. Drugie z tych poleceń wyświetla dodatkowo przypisane do interfejsów adresy IP. Podobną informację można również uzyskać za pomocą starszego polecenia

  ```
  V1$> ifconfig -a
  ```

  Jak w poprzednim zadaniu interfejs `enp`x`s`y będziemy nazywać `enp0`. Jest on połączony (wirtualnie) z interfejsem `enp0` maszyny wirtualnej *Virbian1* uruchomionej na komputerze obok. Zauważ, że interfejs `enp0` nie ma przypisanego adresu IP.

- Poleceniem

  ```
  V1$> ethtool enp0
  ```

  sprawdź status warstwy fizycznej karty `enp0`. Zwróć uwagę na pola `Speed` i `Duplex`. Deklarowana szybkość połączenia powinna wynosić 1 Gbit/s.[1]

- Aktywuj interfejs `enp0` i nadaj mu odpowiedni adres IP poleceniami:

  ```
  V1#> ip link set up dev enp0
  V1#> ip addr add 192.168.0.x/24 dev enp0
  ```

  gdzie x jest numerem Twojego komputera. Wartość `/24` jest tzw. maską podsieci i jej znaczenie zostanie wyjaśnione na kolejnych zajęciach. Sprawdź, jak zmieniła się informacja wyświetlana przez polecenia `ip link` i `ip addr`. Jeśli przypadkowo nadasz karcie `enp0` błędny adres IP, możesz usunąć wszystkie przypisane do tej karty adresy poleceniem `ip addr flush dev enp0`.

- Polecenie `ping` służy do testowania warstwy sieciowej. W polu danych pakietów IP wysyłane są wtedy specjalne komunikaty protokołu ICMP. Wykonaj polecenia

  ```
  V1$> ping adres_IP_interfejsu_enp0_sąsiada
  ```

  Jaki jest wyświetlany RTT (*round trip time*)? Uruchom program Wireshark i włącz w nim obserwację wszystkich interfejsów (wybierając sztuczny interfejs *any*). Obejrzyj pakiety wysyłane i odbierane przez program `ping`. Czy znaczniki czasowe (pole *timestamp*) w wysyłanym zapytaniu i odpowiedzi różnią się, czy są takie same?

[1] Niestety jak się okaże później w przypadku kart wirtualnych informacje te nie są do końca prawdziwe. Dodatkowo pole `Link detected` w przypadku fizycznej karty określa, czy z drugiej strony jest aktywna karta sieciowa. Tutaj natomiast będzie równe `yes`, jeśli tylko aktywujemy interfejs `enp0` maszyny wirtualnej.

- Otwórz plik `/etc/hosts` i przeczytaj dokumentację poleceniem

  ```
  V1$> man hosts
  ```

  Zmodyfikuj ten plik związując adres IP karty `enp0` sąsiada z wymyśloną przez siebie nazwą. Uwaga: to przyporządkowanie działa tylko lokalnie, na jednym komputerze. Sprawdź, czy polecenie `ping` działa też z tymi nazwami.

- Na komputerze sąsiada uruchomcie polecenie

  ```
  V1$> iperf3 -s
  ```

  zaś na swoim komputerze polecenie

  ```
  V1$> iperf3 -c adres_IP_interfejsu_enp3s0_sąsiada
  ```

  Jaką ilość danych udaje Ci się przesłać przez jednostkę czasu? Z czego może wynikać różnica między tą wartością a deklarowaną przez `ethtool` przepustowością kanału (1 Gbit)?

- Na końcu usuń adres IP z interfejsu `enp0` i deaktywuj ten interfejs poleceniami

  ```
  V1#> ip addr flush dev enp0
  V1#> ip link set down dev enp0
  ```

Wyłącz maszynę wirtualną *Virbian1*.

Zadanie do zaprezentowania (5 pkt.)

Poniższe zadanie należy wykonywać samodzielnie na jednym komputerze.

- Utwórz dodatkową maszynę wirtualną *Virbian2*. Maszynom *Virbian1* i *Virbian2* zmostkuj ich karty sieciowe z wirtualną siecią `local0`. Spowoduje to, że po ich uruchomieniu (obu na jednym komputerze) będą one połączone wirtualną siecią `local0`. Uruchom obie wirtualne maszyny.

- Aktywuj karty sieciowe w obu urządzeniach poleceniem `ip` i sprawdź stan warstwy fizycznej kart poleceniem `ethtool`.

- Karcie sieciowej maszyny *Virbian1* przypisz adres IP równy `192.168.100.1`, zaś karcie maszyny *Virbian2* adres `192.168.100.2`. Pamiętaj o masce podsieci `/24`.

- Poleceniem `ping` sprawdź czy jedna maszyna jest osiągalna z drugiej. Jaki jest RTT? Obejrzyj przesyłane pakiety Wiresharkiem.

- Wykorzystaj program `iperf3`, żeby zbadać przepustowość połączenia między maszynami.

- Z maszyny *Virbian1* połącz się z serwerem echa maszyny *Virbian2*. Zaobserwuj przesyłane pakiety w Wiresharkach uruchomionych jednocześnie na obu maszynach.

- ▶ Zdekonfiguruj karty sieciowe obu maszyn i wyłącz wirtualne maszyny.

Lista i materiały znajdują się pod adresem http://www.ii.uni.wroc.pl/~mbi/dyd/.

Marcin Bieńkowski