

# Warsztaty z Sieci komputerowych

## Lista 7

### Przed zajęciami

Uruchom trzy maszyny wirtualne *Virbian1*, *Virbian2* i *Virbian3*, każda z jedną kartą sieciową podłączoną do wirtualnej sieci `local0`. Maszyna *Virbian1* będzie serwerem pocztowym. Wygodnie jest myśleć, że maszyna *Virbian2* należy do użytkownika `student2`, zaś maszyna *Virbian3* do użytkownika `student3`.

### Tutorial #1

W tej części przyjrzymy się działaniu protokołu pocztowego SMTP.

- ▶ Zmień nazwę interfejsu sieciowego na wszystkich maszynach na `enp0`. Na maszynie *Virbiani* (dla  $i \in \{1, 2, 3\}$ ) przypisz temu interfejsowi adres `10.0.0.i/8`. Na wszystkich maszynach dodaj wpis

```
10.0.0.1    mail.example.com
```

do pliku `/etc/hosts`.

- ▶ Na maszynie *Virbian1* uruchom serwery SMTP i POP3 poleceniami

```
V1#> systemctl start postfix
V1#> systemctl start dovecot
```

- ▶ Na maszynie *Virbian2* skonfiguruj program *Thunderbird* do korzystania z adresu `student2@mail.example.com`. W tym celu w Thunderbirdzie (w kreatorze tworzenia konta pocztowego) wpisz swoje imię i nazwisko w polu *Your name*, w polu *Email address* wpisz `student2@mail.example.com`, zaś w polu *Password* wpisz `student2`. Następnie kliknij *Configure manually* i uzupełnij pozostałe pola w następujący sposób.
  - ▷ W części *Incoming* powinien być wybrany protokół POP3, serwer `mail.example.com`, port 110, *wyłączone* szyfrowanie, w polu *Authentication* wybrana opcja *Normal password*, a jako użytkownik `student2` (bez `@mail.example.com`).
  - ▷ W części *Outgoing* powinien być wybrany protokół SMTP, serwer `mail.example.com`, port 25, *wyłączone* szyfrowanie, a w polu *Authentication* wybrana opcja *No authentication*.

Po kliknięciu przycisku *Done* należy przeczytać i następnie zignorować ostrzeżenie o używaniu nieszyfrowanych protokołów. Jeśli pojawi się okno konfiguracji modułu Enigmail, to należy je zamknąć przyciskiem *Cancel*.

- ▶ Wykonaj powyższy punkt z odpowiednimi zmianami, tak żeby skonfigurować program *Thunderbird* na maszynie *Virbian3* do korzystania z adresu `student3@mail.example.com`. (Jego hasło to `student3`, pozostałe opcje należy wybrać analogicznie).
- ▶ Włącz Wiresharka na maszynie *Virbian2*. W *Thunderbirdzie* na maszynie *Virbian2* kliknij przycisk *Write*, napisz i wyślij email do `student3@mail.example.com`). Odbierz ten mail w *Thunderbirdzie* uruchomionym na maszynie *Virbian3*.
- ▶ Obejrzyj przesłane pakiety w Wiresharku: znajdź jeden z przesyłanych segmentów TCP i wybierając z kontekstowego menu opcję *Follow | TCP Stream* sprawdź, jakie komunikaty zostały wymienione między maszyną *Virbian2* a serwerem SMTP uruchomionym na maszynie *Virbian1*. Zapisz je do pliku.

- ▶ Poleceniem

```
V2$> telnet mail.example.com 25
```

połącz się z portem SMTP i wykorzystaj zdobyte w Wiresharku i zapisane do pliku dane do wysłania wiadomości do adresu `student3@mail.example.com`. Zauważ, że treść maila (po poleceniu *DATA*) musi być zakończona pojedynczą kropką. Możesz pominąć pola nagłówka lub wpisać tylko niektóre. Na maszynie *Virbian3* sprawdź w *Thunderbirdzie*, czy mail dotarł.

- ▶ Włącz teraz szyfrowanie SSL protokołu SMTP w *Thunderbirdzie* na maszynie *Virbian2*. W tym celu w lewym panelu okna programu kliknij prawym przyciskiem myszy nazwę konta `student2@mail.example.com` i z menu kontekstowego wybierz opcję *Settings*. W oknie konfiguracji z menu po lewej stronie wybierz opcję *Outgoing Server (SMTP)*, kliknij przycisk *Edit*, a następnie w części *Connection security* wybierz opcję *STARTTLS* i zatwierdź zmiany przyciskiem *OK*.
- ▶ Wyślij ponownie email testowy do `student3@mail.example.com` i zaobserwuj przesyłane za pomocą protokołu SMTP dane w Wiresharku (*Follow | TCP Stream*). Poza samym początkiem komunikacji późniejsze dane powinny być zaszyfrowane i nie powinno się ich dać odczytać. Obejrzyj przesyłane przez protokół SSL komunikaty; w szczególności znajdź pakiet, w którym serwer SMTP wysłał certyfikat SSL i obejrzyj go.<sup>1</sup>
- ▶ Wyślemy teraz mail wykorzystując szyfrowane połączenie. Wykonaj polecenie:

```
V2$> openssl s_client -quiet -connect mail.example.com:25 -starttls smtp
```

i wyślij maila posługując się poleceniami protokołu SMTP (*MAIL FROM*, *RCPT TO* i *DATA*). Możesz skorzystać z transmisji uprzednio zapisanej w pliku. Obejrzyj przesyłane dane w Wiresharku i sprawdź w *Thunderbirdzie*, że email został dostarczony.

---

<sup>1</sup>Możliwe jest, że przy kolejnych połączeniach serwer nie będzie wysyłał certyfikatu, lecz będzie wykorzystywał ustalony wcześniej klucz sesji. W takim przypadku, aby zobaczyć certyfikat, zazwyczaj wystarczy ponownie uruchomić *Thunderbirda* wysłać kolejny email.

## Tutorial #2

W tej części zapoznamy się z programem `gpg` będącym implementacją standardu OpenPGP.

- ▶ W ustawieniach Virtualboksa dla maszyny *Virbian2* przełącz kartę sieciowa w tryb NAT i w maszynie *Virbian2* skonfiguruj połączenie z Internetem za pomocą DHCP.

- ▶ Na maszynie *Virbian2* utwórz parę kluczy PGP (publiczny i prywatny) poleceniem

```
V2$> gpg --gen-key
```

Jako nazwę użytkownika wybierz `student2`, a jako adres email wpisz `student2@mail.example.com`. Utwórz i zapamiętaj hasło chroniące klucz prywatny.

- ▶ Posiadane klucze (odpowiednio prywatne i publiczne) można wyświetlić poleceniami

```
V2$> gpg --list-secret-keys
```

```
V2$> gpg --list-keys
```

Na razie będą tam widoczne tylko klucze użytkownika `student2`.

- ▶ Wejdź na stronę <https://www.veracrypt.fr/en/Downloads.html> i pobierz ten program (w dowolnej wersji) razem z odpowiadającym podpisem PGP (link *PGP Signature*). Zamiast programu Veracrypt możesz wybrać dowolny inny program podpisany kluczem PGP jego autora/autorów. Zapisz program w pliku `veracrypt.deb` a jego podpis w pliku `veracrypt.deb.sig`.

- ▶ Poleceniem

```
V2$> gpg --verify veracrypt.deb.sig veracrypt.deb
```

sprawdź, czy podpis jest poprawny. Otrzymasz komunikat o braku odpowiedniego klucza publicznego o identyfikatorze `5069A233D55A0EEB174A5FC3821ACD02680D16DE`.

- ▶ Pobierz ten klucz publiczny z ogólnodostępnego repozytorium kluczy poleceniem

```
V2$> gpg --recv-keys identyfikator_klucza
```

i wyświetl posiadane klucze publiczne poleceniem

```
V2$> gpg --list-keys
```

Zauważ, że przy Twoim kluczu publicznym jest napis `ultimate`, zaś przy kluczu publicznym opisanym jako *Veracrypt* jest napis `unknown`. Obie te wartości oznaczają poziom zaufania do tego, czy dany klucz należy do konkretnej osoby/instytucji.

Ponów próbę weryfikacji podpisu. Tym razem okaże się, że podpis jest poprawny, ale nie mamy żadnej gwarancji, że właśnie pobrany przez nas klucz publiczny faktycznie należy do autorów oprogramowania.

- ▶ Aby to naprawić, wejdź w tryb edycji tego klucza poleceniem

```
V2$> gpg --edit-key Veracrypt
```

Po znaku zachęty wpisz polecenie

```
gpg> fpr
```

wyświetlające skrót klucza publicznego. Teraz powinniśmy poprosić autorów oprogramowania o podanie nam zaufanym kanałem obliczonego po ich stronie skrótu klucza. Zamiast tego zadowolimy się porównaniem wyświetlanej funkcji skrótu z funkcją skrótu dostępną na ich stronie [www](#).<sup>2</sup> Załóż, że posiadany klucz faktycznie należy do autorów oprogramowania i podpisz go poleceniem

```
gpg> sign
```

a następnie opuść tryb edycji poleceniem

```
gpg> quit
```

Zauważ, że jeśli teraz wyświetlisz dostępne klucze publiczne, to przy kluczu *Veracrypt* będzie informacja o pełnym (*full*) zaufaniu do tego klucza.

- ▶ Wykonaj kolejną próbę weryfikacji podpisu. Tym razem powinna ona zakończyć się powodzeniem.
- ▶ W ustawieniach VirtualBoksa dla maszyny *Virbian2* przełącz kartę sieciową z powrotem w tryb *Internal Network* (tak, żeby karta była połączona z wirtualną siecią *local0*). Przywróć tej maszynie ustawienia sieciowe z poprzedniego zadania (przypisz adres *10.0.0.2/8*).

## Wyzwanie #1

- ▶ Zapisz klucz publiczny użytkownika *student2* z maszyny *Virbian2* w czytelnej postaci do pliku *student2-gpg-key* poleceniem

```
V2$> gpg -a --export student2 > student2-gpg-key
```

Wyślij ten klucz mailem do użytkownika *student3*.

- ▶ Na maszynie *Virbian3* wygeneruj klucz prywatny i publiczny PGP, jako użytkownika podając *student3*, a jako adres email *student3@mail.example.com*. Wyeksportuj klucz publiczny do pliku *student3-gpg-key* i wyślij użytkownikowi *student2*.
- ▶ Na maszynie *Virbian2* zaimportuj klucz publiczny użytkownika *student3* za pomocą polecenia

```
V2$> gpg --import < student3-gpg-key
```

---

<sup>2</sup>Od pewnego czasu skrót klucza publicznego jest zarazem jego identyfikatorem, więc wyświetlanym skrótem jest *5069 A233 D55A 0EEB 174A 5FC3 821A CD02 680D 16DE*.

Wejdź w tryb edycji tego klucza, upewnij się, że jego funkcja skrótu jest odpowiednia i podpisz go kluczem prywatnym użytkownika `student2`

- ▶ Wykonaj powyższy punkt, ale na maszynie *Virbian3* zamieniając role `student2` i `student3`.
- ▶ Na maszynie *Virbian2* utwórz plik `message` i umieść w nim jakąś treść. W celu podpisania wiadomości prywatnym kluczem użytkownika `student2` i zaszyfrowania jej kluczem publicznym użytkownika `student3` wydaj polecenie

```
V2$> gpg -a -r student3 -se message
```

Szyfrogram zostanie zapisany do pliku `message.asc`, który można wysłać mailem do użytkownika `student3`.

- ▶ Na maszynie *Virbian3* otrzymany plik `message.asc` należy odszyfrować kluczem prywatnym użytkownika `student3` i zweryfikować prawdziwość podpisu poleceniem

```
V3$> gpg -d message.asc > deciphered.message
```

- ▶ Wyłącz serwery SMTP i POP3 poleceniami

```
V1#> systemctl stop postfix  
V1#> systemctl stop dovecot
```

Materiały do kursu znajdują się w systemie Canvas: <https://canvas.ii.uni.wroc.pl/>.

*Marcin Bieńkowski*