

PODSTAWY SIECI KOMPUTEROWYCH

SPRAWOZDANIE

z realizacji zadania

Protokół HTTP

Autor: posk\_HTTP\_Michał\_Gebel.odt

Adres interfejsu występujący w zadaniu: 192.168.0.5

UWAGA Zadanie wykorzystuje fakt, że maszyny wirtualne w laboratorium mają skonfigurowany pierwszy interfejs Ethernet w trybie połączonym z interfejsem gospodarza (maszyny fizycznej), w sieci fizycznej jest dostępna usługa automatycznej konfiguracji hostów i przydzielana konfiguracja zapewnia "dostęp do Internetu". Aby móc zrealizować zadanie poza laboratorium, należy spełnić we własnym środowisku te warunki. Pojęcie "interfejs" w zadaniu odnosi się do tegoż pierwszego interfejsu Ethernet maszyny wirtualnej.

UWAGA Pojęcie „strona testowa” odnosi się do zbioru stron udostępnionych na potrzeby zadania. Odnosi do tych stron znajdują się na stronie: <http://skl.it.p.lodz.pl/~mak/posk/> .

UWAGA Przeglądarka Firefox w maszynach wirtualnych używanych w ramach laboratorium została skonfigurowana tak, aby nie zachowywała żadnych danych (historia, cache stron). W celu realizacji niektórych punktów zadania wymagana jest korekta tych ustawień (szczegóły w treści zadania).

1. Na obu stacjach włącz monitorowanie ruchu na pierwszym interfejsie Ethernet (wireshark) ograniczając prezentowane wyniki do protokołu HTTP. **Ponadto przedstaw (wpisz na stronie tytułowej) adres pierwszego interfejsu.**

```
[root@localhost lsk]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:2d:d3:bc brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.5/24 brd 192.168.0.255 scope global dynamic enp0s3
        valid_lft 86262sec preferred_lft 86262sec
    inet6 fe80::4c:1c2b:a87d:4928/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:96:66:3a brd ff:ff:ff:ff:ff:ff
4: enp0s9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:5e:2f:3e brd ff:ff:ff:ff:ff:ff
```

2. Otwórz stronę oznaczoną „Punkt 2” za pomocą przeglądarki. **Zaprezentuj w historii komunikacji wszystkie wygenerowane w ten sposób żądania HTTP.** Następnie **otwórz ten sam URL za pomocą programu curl.** Sprawdź w historii komunikacji, ile żądań zostało w ten sposób wygenerowanych. Czym różni się działanie klienta HTTP, jakim jest *curl*, od działania przeglądarki?

68	30.04255310	192.168.0.5	212.51.220.3	HTTP	385 GET /~mak/posk/ssbd_tutor.jpg HTTP/1.1
69	30.07243852	212.51.220.3	192.168.0.5	TCP	66 80 → 34256 [ACK] Seq=1 Ack=320 Win=6912 Len=0 TSval=2652649469 TSecr=2072380723
70	30.07246837	212.51.220.3	192.168.0.5	TCP	4386 [TCP segment of a reassembled PDU]
71	30.07248082	192.168.0.5	212.51.220.3	TCP	66 34256 → 80 [ACK] Seq=320 Ack=4321 Win=37888 Len=0 TSval=2072380753 TSecr=2652649469
72	30.10328891	212.51.220.3	192.168.0.5	TCP	1506 [TCP segment of a reassembled PDU]
73	30.10332315	192.168.0.5	212.51.220.3	TCP	66 34256 → 80 [ACK] Seq=320 Ack=5761 Win=40768 Len=0 TSval=2072380784 TSecr=2652649500
74	30.10392002	212.51.220.3	192.168.0.5	TCP	4386 [TCP segment of a reassembled PDU]
75	30.10395493	192.168.0.5	212.51.220.3	TCP	66 34256 → 80 [ACK] Seq=320 Ack=10081 Win=49408 Len=0 TSval=2072380784 TSecr=2652649500
76	30.13124163	212.51.220.3	192.168.0.5	TCP	1506 [TCP segment of a reassembled PDU]
77	30.13127167	192.168.0.5	212.51.220.3	TCP	66 34256 → 80 [ACK] Seq=320 Ack=11521 Win=52288 Len=0 TSval=2072380812 TSecr=2652649530
78	30.13299474	212.51.220.3	192.168.0.5	TCP	1506 [TCP segment of a reassembled PDU]
79	30.13304458	192.168.0.5	212.51.220.3	TCP	66 34256 → 80 [ACK] Seq=320 Ack=12961 Win=55168 Len=0 TSval=2072380813 TSecr=2652649530
80	30.13432085	212.51.220.3	192.168.0.5	TCP	5826 [TCP segment of a reassembled PDU]
81	30.13439291	192.168.0.5	212.51.220.3	TCP	66 34256 → 80 [ACK] Seq=320 Ack=18721 Win=66688 Len=0 TSval=2072380815 TSecr=2652649530
82	30.13488387	212.51.220.3	192.168.0.5	HTTP	2115 HTTP/1.1 200 OK (JPEG JFIF image)

```
[root@localhost lsk]# curl http://skl.it.p.lodz.pl/~mak/posk/
<html>
  <head>
    <title>Strony testowe PoSK - strona główna</title>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  </head>
  <body>
    <h1>Strony testowe PoSK - strona główna</h1>
    <ul>
      <li><a href="img.html">Punkt 2</a></li>
      <li><a href="form-get.html">Punkt 3a</a></li>
      <li><a href="form-post.html">Punkt 3b</a></li>
      <li><a href="auth/">Punkt 4</a></li>
    </ul>
  </body>
</html>
```

4	0.032203308	192.168.0.5	212.51.220.3	HTTP	156	GET /~mak/posk/ HTTP/1.1
6	0.062309062	212.51.220.3	192.168.0.5	HTTP	729	HTTP/1.1 200 OK (text/html)

W odróżnieniu od curl, przeglądarka pobiera zawartość strony, natomiast curl tylko pobiera zawartość strony w postaci źródłowej (html).

3. Za pomocą przeglądarki otwórz stronę oznaczoną jako „Punkt 3a”. Wypełnij formularz używając *nieistotnych danych (POD ŻADNYM POZOREM nie podawaj żadnych prawdziwych danych uwierzytelniających)*. Wyślij formularz. W historii komunikatów odnajdź wygenerowane w ten sposób żądanie i użyj na nim funkcji „Follow HTTP Stream” lub „Follow TCP Stream” programu Wireshark, w **sprawozdaniu zademonstruj jej wynik**. (menu kontekstowe; UWAGA – wykonanie zmienia filtr prezentowanych pakietów, po realizacji prawdopodobnie będzie potrzebne manualne przywrócenie poprzedniego filtra). W jaki sposób zostały zawarte w żądaniu parametry formularza? Czy pola formularza typu „password” oraz „hidden” (możesz podejrzec źródło strony z formularzem) gwarantują poufność przesyłanych informacji? **Eksperyment powtórz dla strony oznaczonej jako „Punkt 3b”**. Jaka jest różnica między wynikami tych eksperymentów?

3.a

4	0.035716000	192.168.0.5	212.51.220.3	HTTP	408	GET /~mak/posk/ HTTP/1.1
6	0.069179524	212.51.220.3	192.168.0.5	HTTP	729	HTTP/1.1 200 OK (text/html)
15	4.741358712	192.168.0.5	212.51.220.3	HTTP	466	GET /~mak/posk/form-get.html HTTP/1.1
17	4.771320619	212.51.220.3	192.168.0.5	HTTP	958	HTTP/1.1 200 OK (text/html)
26	14.298269264	192.168.0.5	212.51.220.3	HTTP	530	GET /~mak/posk/index.html?login=michael&password=corleone&ukryte=Niby+ukryte%21 HTTP/1.1
28	14.327162355	212.51.220.3	192.168.0.5	HTTP	729	HTTP/1.1 200 OK (text/html)

```
GET /~mak/posk/index.html?login=michael&password=corleone&ukryte=Niby+ukryte%21
HTTP/1.1
```

```
Host: skl.it.p.lodz.pl
User-Agent: Mozilla/5.0 (X11; Fedora; Linux x86_64; rv:49.0) Gecko/20100101
Firefox/49.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://skl.it.p.lodz.pl/~mak/posk/form-get.html
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
```

```
HTTP/1.1 200 OK
Date: Sat, 19 Dec 2020 19:52:06 GMT
Server: Apache
Last-Modified: Tue, 11 Apr 2017 17:50:37 GMT
ETag: "214252-1a8-54ce7bab2f540"
```

Accept-Ranges: bytes  
Content-Length: 424  
Connection: close  
Content-Type: text/html

```
<html>
  <head>
    <title>Strony testowe PoSK - strona g....wna</title>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  </head>
  <body>
    <h1>Strony testowe PoSK - strona g....wna</h1>
  <ul>
    <li><a href="img.html">Punkt 2</a></li>
    <li><a href="form-get.html">Punkt 3a</a></li>
    <li><a href="form-post.html">Punkt 3b</a></li>
    <li><a href="auth/">Punkt 4</a></li>
  </ul>
</body>
</html>
```

### 3.b

4	0.032044886	192.168.0.5	212.51.220.3	HTTP	531 GET /~mak/posk/form-post.html HTTP/1.1
6	0.062096512	212.51.220.3	192.168.0.5	HTTP	961 HTTP/1.1 200 OK (text/html)
15	5.472059671	192.168.0.5	212.51.220.3	HTTP	600 POST /~mak/posk/index.html HTTP/1.1 (application/x-www-form-urlencoded)
17	5.501327991	212.51.220.3	192.168.0.5	HTTP	729 HTTP/1.1 200 OK (text/html)

#### **POST /~mak/posk/index.html HTTP/1.1**

Host: skl.it.p.lodz.pl  
User-Agent: Mozilla/5.0 (X11; Fedora; Linux x86\_64; rv:49.0) Gecko/20100101 Firefox/49.0  
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8  
Accept-Language: en-US,en;q=0.5  
Accept-Encoding: gzip, deflate  
Referer: http://skl.it.p.lodz.pl/~mak/posk/form-post.html  
DNT: 1  
Connection: keep-alive  
Upgrade-Insecure-Requests: 1  
Content-Type: application/x-www-form-urlencoded  
Content-Length: 53

#### **login=michael&password=corleone&ukryte=Niby+ukryte%21HTTP/1.1 200 OK**

Date: Sat, 19 Dec 2020 19:53:29 GMT  
Server: Apache  
Last-Modified: Tue, 11 Apr 2017 17:50:37 GMT  
ETag: "214252-1a8-54ce7bab2f540"  
Accept-Ranges: bytes  
Content-Length: 424  
Connection: close  
Content-Type: text/html

```
<html>
  <head>
    <title>Strony testowe PoSK - strona g....wna</title>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  </head>
  <body>
    <h1>Strony testowe PoSK - strona g....wna</h1>
  <ul>
    <li><a href="img.html">Punkt 2</a></li>
    <li><a href="form-get.html">Punkt 3a</a></li>
    <li><a href="form-post.html">Punkt 3b</a></li>
    <li><a href="auth/">Punkt 4</a></li>
  </ul>
</body>
</html>
```

W jaki sposób zostały zawarte w żądaniu parametry formularza?

Odp. Na samym końcu adresu (login=<login>&password=<hasło>)

Czy pola formularza typu „password” oraz „hidden” gwarantują poufność przesyłanych informacji?

Odp. Nie gwarantują tego. Typ password zmienia widoczność na hidden (podczas wpisywania zamiast hasła widzimy czarne kropki)

Jaka jest różnica między wynikami eksperymentów?

Odp. Żądanie GET gwarantuje łatwy dostęp do danych, które są dołączone do adresu. Aby zwiększyć troszkę poziom zabezpieczeń należy stosować żądanie POST, dzięki któremu nie widać informacji w adresie oraz historii przeglądarki. Są one jednak dostępne w odpowiednim programie nasłuchującym np. wireshark.

4. Za pomocą przeglądarki otwórz stronę oznaczoną jako „Punkt 4”. Wypełnij i zatwierdź okno logowania (*POD ŻADNYM POZOREM nie podawaj żadnych prawdziwych, własnych danych uwierzytelniających! Jeżeli odgadniesz poprawne dane opublikuj je na forum przedmiotu i zgłoś się do prowadzącego po gratulacje*). W historii komunikatów odnajdź wygenerowane po podaniu danych uwierzytelniających żądanie i **zademonstruj zawartość żądania oraz co najmniej kod odpowiedzi w sposób analogiczny jak w zadaniu 3**. Czy przesłane dane są zabezpieczone przed ujawnieniem?

4	0.031395533	192.168.0.5	212.51.220.3	HTTP	468 GET /~mak/posk/auth/ HTTP/1.1
6	0.061331700	212.51.220.3	192.168.0.5	HTTP	760 HTTP/1.1 401 Authorization Required (text/html)
15	13.287783063	192.168.0.5	212.51.220.3	HTTP	515 GET /~mak/posk/auth/ HTTP/1.1
17	13.321763101	212.51.220.3	192.168.0.5	HTTP	760 HTTP/1.1 401 Authorization Required (text/html)

**GET /~mak/posk/auth/ HTTP/1.1**

Host: skl.it.p.lodz.pl

User-Agent: Mozilla/5.0 (X11; Fedora; Linux x86\_64; rv:49.0) Gecko/20100101 Firefox/49.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8

Accept-Language: en-US,en;q=0.5

Accept-Encoding: gzip, deflate

Referer: http://skl.it.p.lodz.pl/~mak/posk/index.html

DNT: 1

Connection: keep-alive

Upgrade-Insecure-Requests: 1

Authorization: Basic **bWljagF1bDpjb3JsZW9uZQ==**

HTTP/1.1 401 Authorization Required

Date: Sat, 19 Dec 2020 20:05:55 GMT

Server: Apache

WWW-Authenticate: Basic realm="Restricted Files"

Content-Length: 467

Connection: close

Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">

<html><head>

<title>401 Authorization Required</title>

```

</head><body>
<h1>Authorization Required</h1>
<p>This server could not verify that you
are authorized to access the document
requested. Either you supplied the wrong
credentials (e.g., bad password), or your
browser doesn't understand how to supply
the credentials required.</p>
<hr>
<address>Apache Server at skl.it.p.lodz.pl Port 80</address>
</body></html>

```

Dane są zabezpieczone, gdyż nie widać ich w powyższym komunikacie.

5. **Za pomocą programu *curl* zgłoś żądanie HEAD (zob. *curl --help*) dla URL <http://www.ics.p.lodz.pl>. Co oznacza ten kod odpowiedzi? Otwórz ten sam URL za pomocą przeglądarki i **zademonstruj historię komunikatów** (wystarczy lista pakietów w programie Wireshark, o ile będą na niej widoczne dwa pierwsze żądania i kody odpowiedzi).**

```

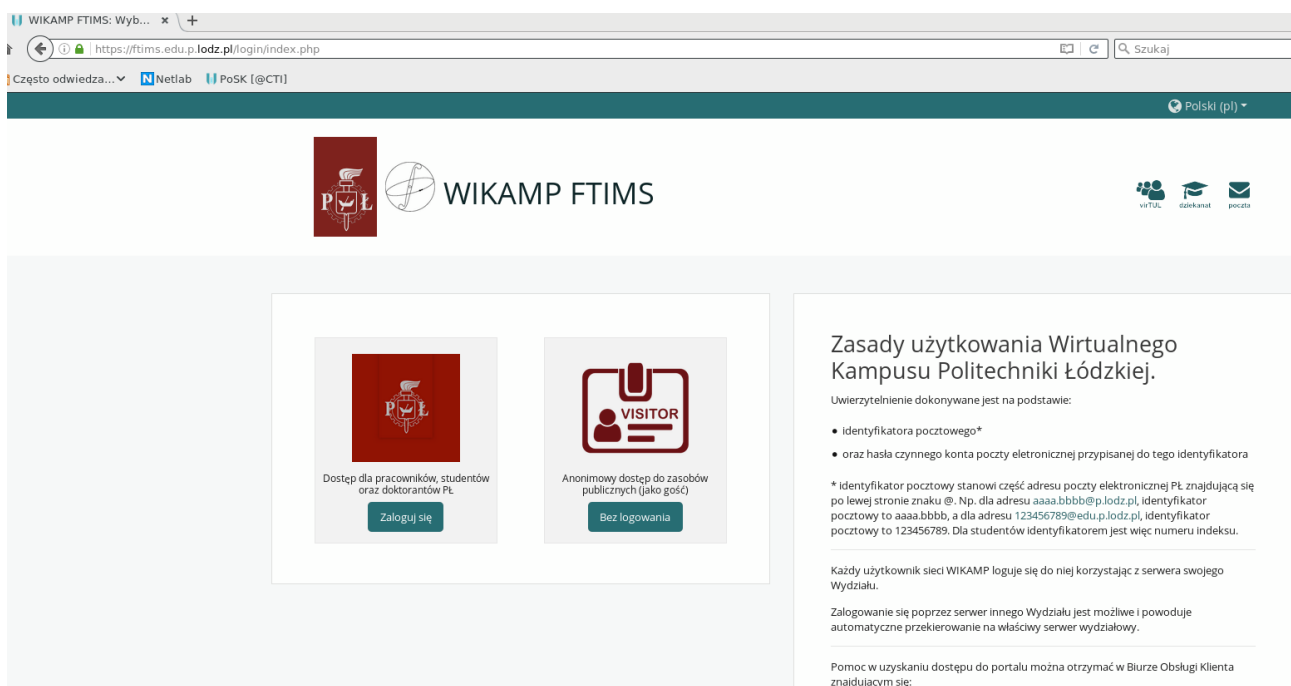
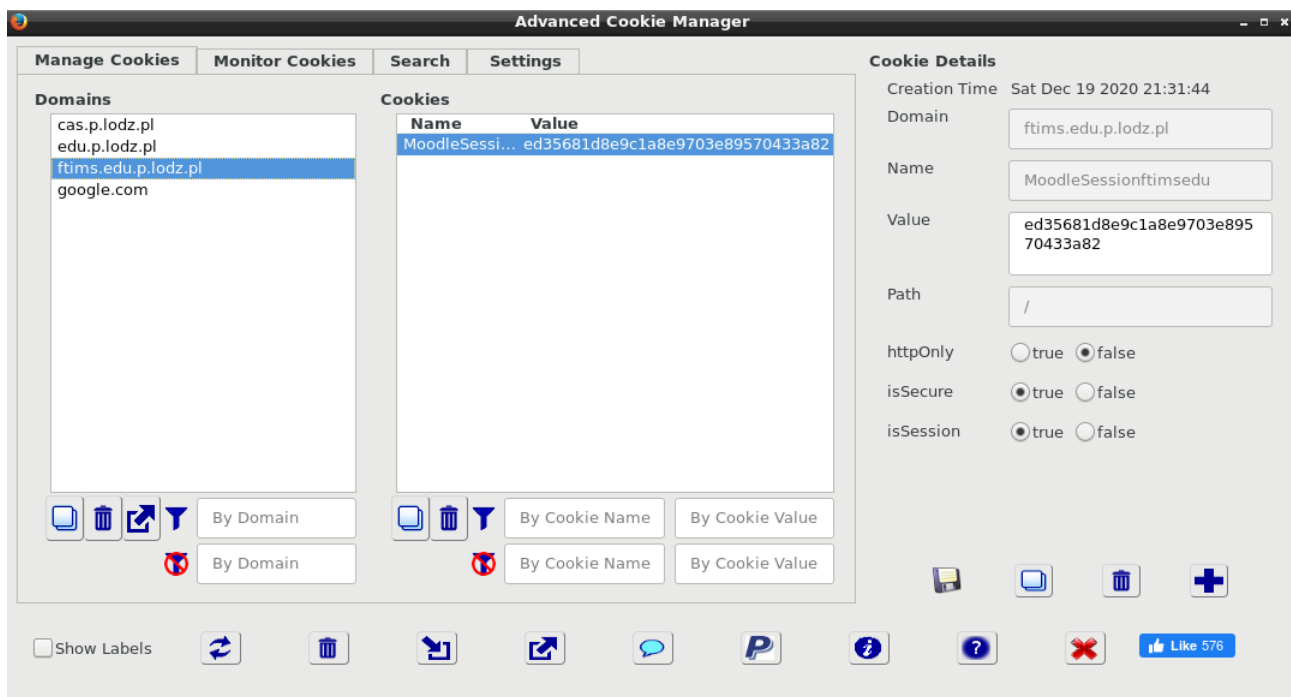
[root@localhost lsk]# curl -I http://www.ics.p.lodz.pl
HTTP/1.1 302 Found
Date: Sat, 19 Dec 2020 20:12:36 GMT
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/5.4.16
Location: http://it.p.lodz.pl/
Content-Type: text/html; charset=iso-8859-1

```

1	0.000000000	192.168.0.5	212.51.220.230	TCP	74	48954 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1400357559 TSecr=0 WS=64
2	0.032020665	212.51.220.230	192.168.0.5	TCP	74	80 → 48954 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1452 SACK_PERM=1 TSval=2152773540 TSecr=1400357559 WS=128
3	0.032055944	192.168.0.5	212.51.220.230	TCP	66	48954 → 80 [ACK] Seq=1 Ack=1 Win=29248 Len=0 TSval=1400357591 TSecr=2152773540
4	0.032111989	192.168.0.5	212.51.220.230	HTTP	148	HEAD / HTTP/1.1
5	0.063705327	212.51.220.230	192.168.0.5	TCP	66	80 → 48954 [ACK] Seq=1 Ack=83 Win=29056 Len=0 TSval=2152773572 TSecr=1400357591
6	0.063730616	212.51.220.230	192.168.0.5	HTTP	264	HTTP/1.1 302 Found
7	0.063740801	192.168.0.5	212.51.220.230	TCP	66	48954 → 80 [ACK] Seq=83 Ack=199 Win=30272 Len=0 TSval=1400357623 TSecr=2152773572
8	0.063808311	192.168.0.5	212.51.220.230	TCP	66	48954 → 80 [FIN, ACK] Seq=83 Ack=199 Win=30272 Len=0 TSval=1400357623 TSecr=2152773572
9	0.093507934	212.51.220.230	192.168.0.5	TCP	66	80 → 48954 [FIN, ACK] Seq=199 Ack=84 Win=29056 Len=0 TSval=2152773602 TSecr=1400357623
10	0.093538518	192.168.0.5	212.51.220.230	TCP	66	48954 → 80 [ACK] Seq=84 Ack=200 Win=30272 Len=0 TSval=1400357653 TSecr=2152773602

Kod 302 = znaleziono. Żądany zasób jest chwilowo dostępny pod innym adresem, a przyszłe odwołania do zasobu powinny być kierowane pod adres pierwotny

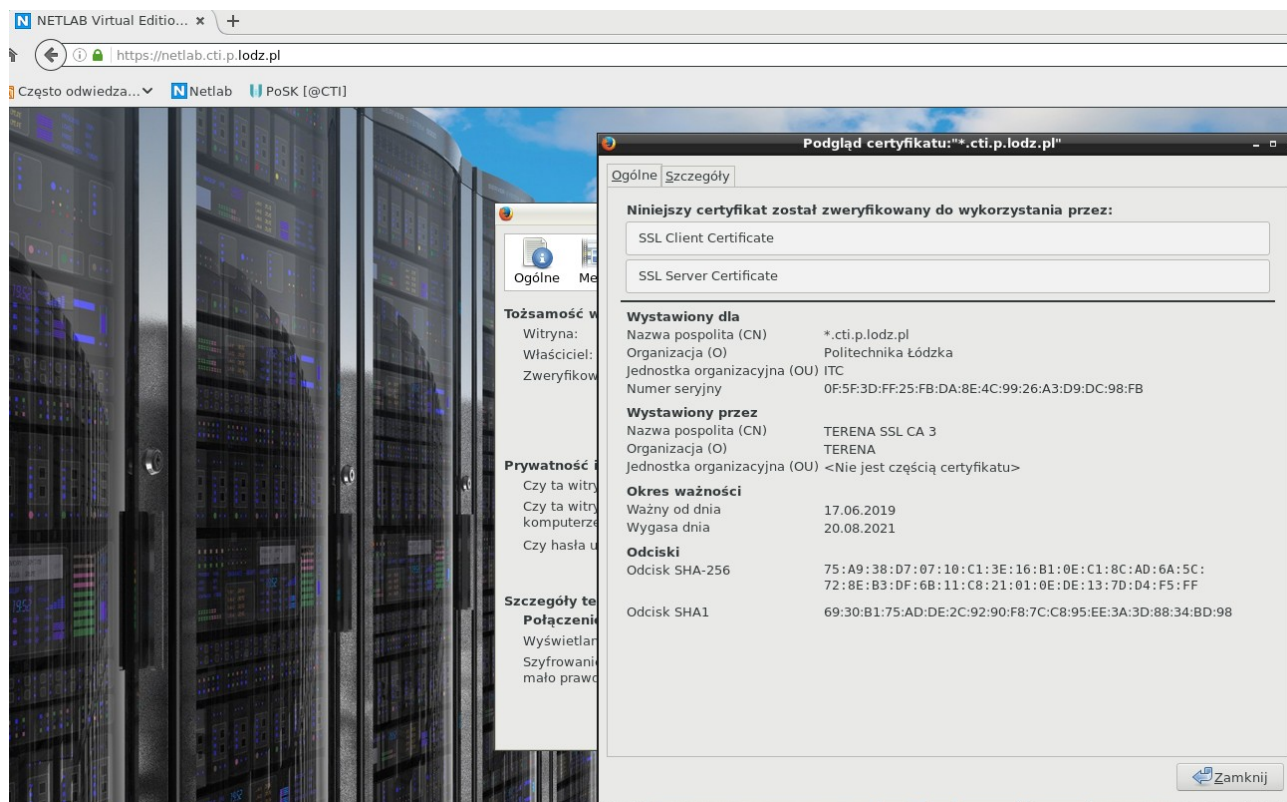
6. Zmień konfigurację przeglądarki Firefox tak, aby przechowywała ona historię (Preferencje -> Prywatność -> Historia), *bez tej zmiany nie otrzymasz poprawnych wyników*. Zaloguj się na platformę Wikamp i używając dodatku przeglądarki Cookie Manager (ikona CM na pasku przeglądarki) **zaprezentuj parametry ciasteczka, którego wartość identyfikuje Twoją sesję**. Następnie usuń to ciasteczko i odśwież stronę platformy Wikamp, **wynik odświeżenia (wyświetloną stronę) zaprezentuj w sprawozdaniu**. Jaki jest skutek usunięcia ciasteczka?



Poprzez usunięcie ciasteczka z przeglądarki usunęliśmy informację o danej sesji. Jednak informacje o sesji zostają przechowane w systemie operacyjnym dzięki czemu, po ponownym kliknięciu „zaloguj” nie istniała potrzeba podania loginu i hasła. Strona automatycznie zalogowała się na moje konto.

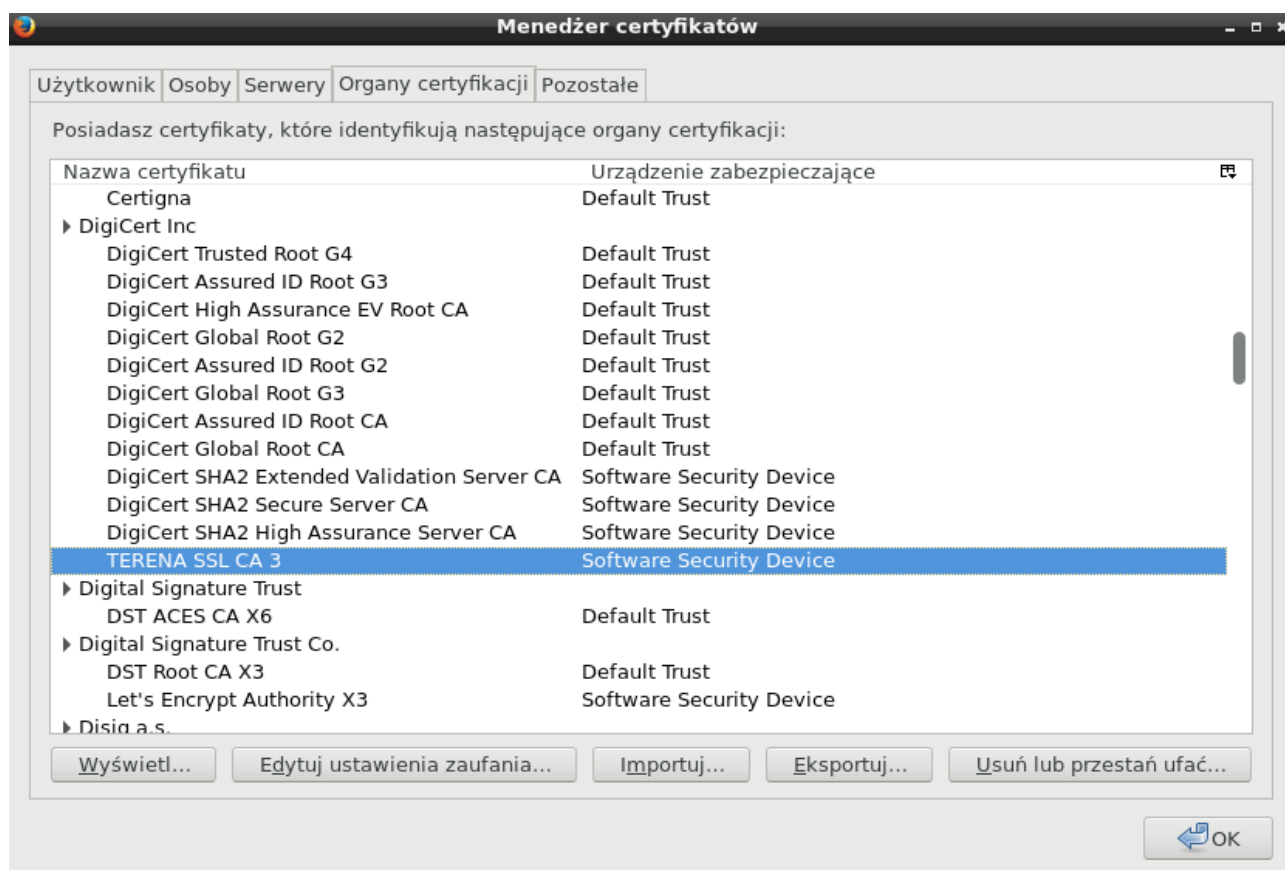


7. Otwórz w przeglądarce witrynę systemu Netlab (<https://netlab.cti.p.lodz.pl>). Zaprezentuj informacje o certyfikacie w taki sposób, aby widoczna była informacja o podmiocie, dla którego został wydany certyfikat. Czy nazwa podmiotu jest tożsama z nazwą witryny? Zaprezentuj także ścieżkę zaufania certyfikatu, oraz zaprezentuj listę certyfikatów zaufanych przeglądarki (Preferencje -> Zaawansowane) w taki sposób, aby wykazać obecność na niej certyfikatu nadrzędnego w ścieżce zaufania.



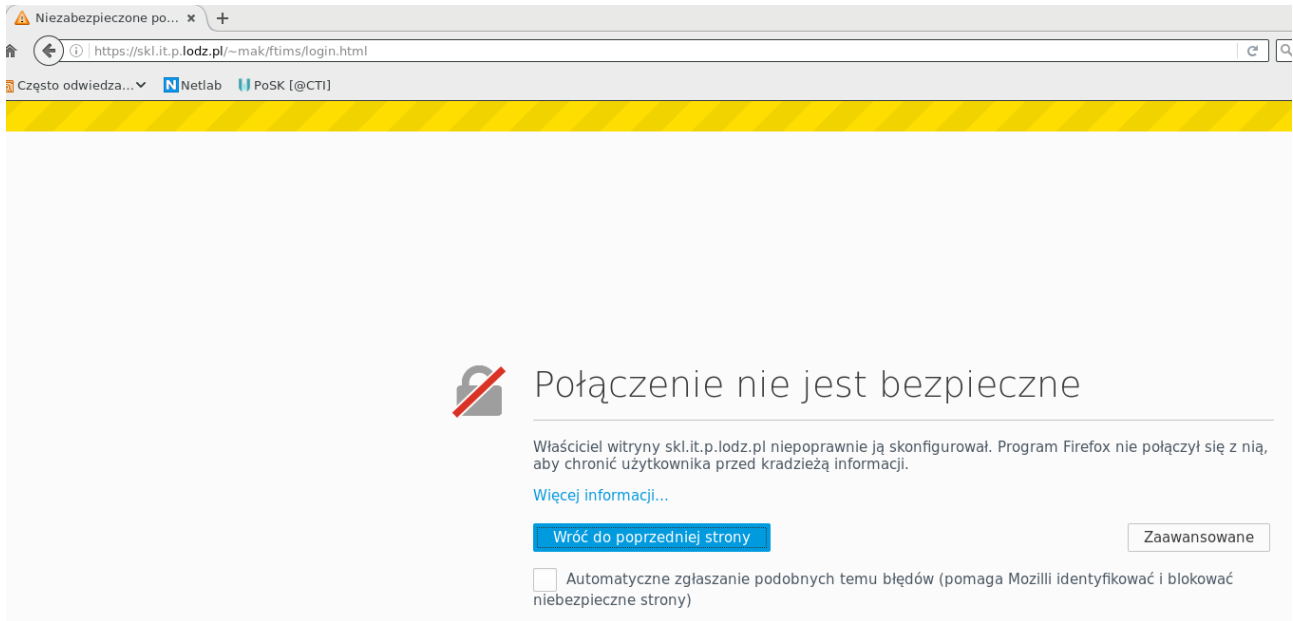
Nazwa podmiotu jest tożsama z nazwą witryny.





8. Otwórz fałszywą stronę logowania <http://skl.it.p.lodz.pl/~mak/ftims/login.html>. Następnie spróbuj otworzyć ją ponownie zmieniając protokół na HTTPS. **Zaprezentuj ekran przeglądarki z komunikatem błędu. W opcjach zaawansowanych zademonstruj, dlaczego przeglądarka nie uznała certyfikatu dla tej witryny za wiarygodny.** Dodaj na trwałe wyjątek dla tej witryny (wymaga to włączenia zachowywania historii przez przeglądarkę) i **wykaż dodanie certyfikatu do listy zaufanych certyfikatów serwerów.**

## Komunikat błędu:



## Powód:

skl.it.p.lodz.pl uses an invalid security certificate.

The certificate is only valid for [zskl.zsk.p.lodz.pl](https://zskl.zsk.p.lodz.pl)

Error code: [SSL\\_ERROR\\_BAD\\_CERT\\_DOMAIN](#)


[Dodaj wyjątek...](#)

Po dodaniu:

virTUL - Politechnika Łódź... x +

https://skl.it.p.lodz.pl/~mak/ftims/login.html

Centralny System Uwierzytelniania

 **virTUL**  
virtual Lodz University of Technology

Login:

Lognem dla studentów jest numer albumu, natomiast dla pracowników identyfikator użytkownika z adresu e-mail - zwykle: "[imię.nazwisko]".


W pozostałych przypadkach należy podać pełny adres skrzynki pocztowej: "[nazwa-uzytkownika]@[domena-poczty-PL]".

Hasło:

W celu zachowania bezpieczeństwa, po zakończeniu korzystania z usług należy wylogować się i zamknąć przeglądarkę!

**ZALOGUJ**

English Français Deutsch Polski

 Politechnika Łódźka

virTUL | PL  
Ochrona Danych Osobowych

© Centrum Komputerowe PL

Dodanie certyfikatu:


**Menedżer certyfikatów**

Użytkownik Osoby **Serwery** Organy certyfikacji Pozostałe

Posiadasz certyfikaty, które identyfikują następujące serwery:

Nazwa certyfikatu	Serwer	Czas życia	Wygasa dnia
▶ DigiNotar			
DigiNotar Root CA	*	Permanent	31.03.2025
▶ DigiNotar B.V.			
DigiNotar PKIoverheid CA ...	*	Permanent	23.03.2020
▶ TERENA			
zskl.zsk.p.lodz.pl	skl.it.p.lodz.pl:443	Permanent	12.11.2021

Wyświetl... Eksportuj... Usuń... Dodaj wyjątek...

 OK