Laboratorium 5

Wireshark cz. 2

W sprawozdaniu zamieść zrzuty ekranu z istotnych etapów oraz odpowiedzi na pytania.

Taki symbol oznacza, że trzeba w sprawozdaniu dodać zrzut ekranu (najczęściej 1) z wyniku działania polecenie.

Taki symbol oznacza, że należy dodać opis (najczęściej 1 zdanie) z wyniku działania polecenia.

Możliwości deszyfrowania ruchu SSL

1. Uruchom program Wireshark.
2. Dodaj kolumny Stream oraz Host, tak jak na poprzednich zajęciach z Wiresharka.
3. Rozpocznij przechwytywanie na odpowiednim interfejsie sieciowym.
4. Uruchom przeglądarkę internetową Mozilla Firefox.
5. Wejdź na stronę *sekurak.pl*.
6. Wróć do programu Wireshark i wyszukaj pakiet `Client Hello` związany z połączeniem do strony *sekurak.pl*.

Jak było omawiane na wykładzie, każda „szanująca się” strona w Internecie powinna być zabezpieczona protokołem SSL/TLS.

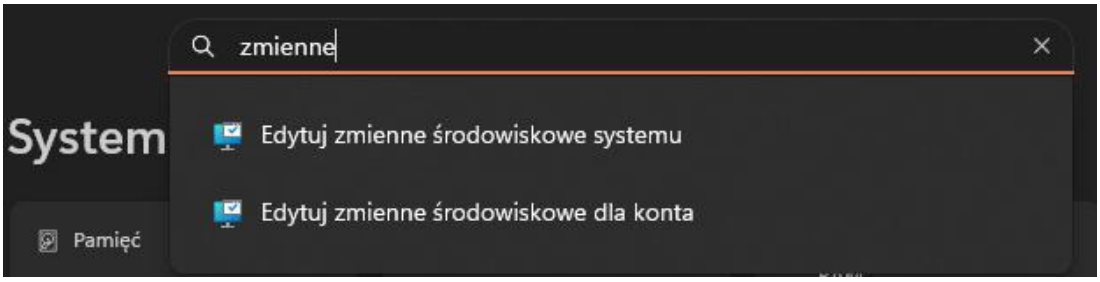
W czasie nawiązywania połączenia HTTPS (np. z `https://...`) przeglądarka i serwer:

- uzgadniają wspólny klucz symetryczny sesji (session key)
- a następnie cały ruch jest szyfrowany tym kluczem.

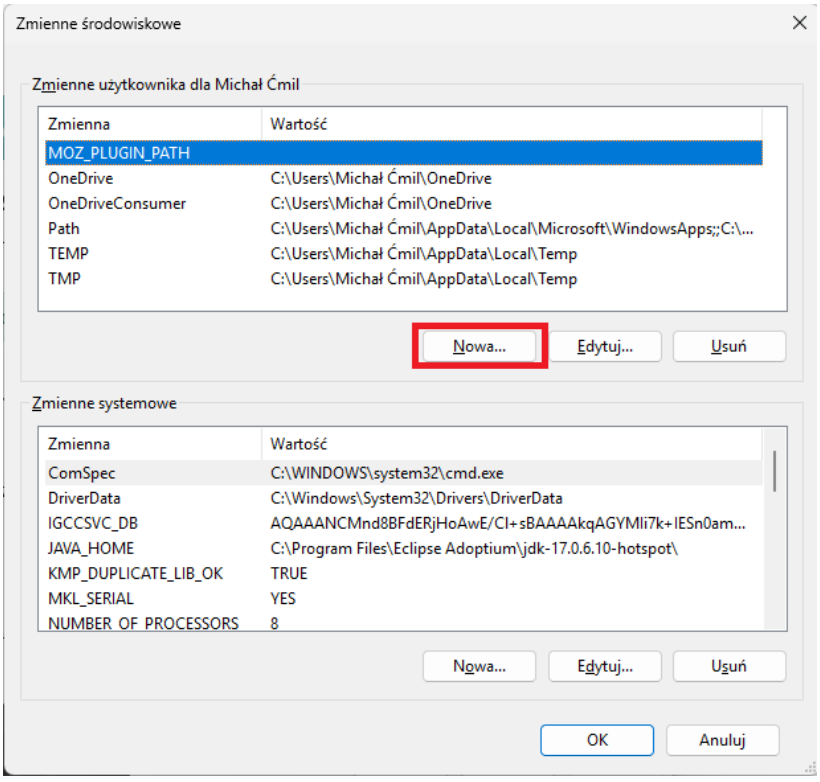
Jeśli posiadasz ten klucz, możesz w Wiresharku „podglądać” odszyfrowany ruch – czyli widzieć, co zostało wysłane, a nie tylko zaszyfrowane pakiety. Na tym etapie, podobnie jak na wcześniejszym zrzucie ekranu, wszystkie informacje są nadal zaszyfrowane. Zauważ także, że kolumny `Host` i `Stream` są puste.

Od wersji 1.6+ Wireshark umożliwia podanie klucza symetrycznego sesji SSL/TLS bezpośrednio do programu. Klucz sesji można uzyskać m.in. przez ustawienie zmiennej środowiskowej w przeglądarce (np. Firefox), co wymusza zapisywanie danych szyfrowania do pliku.

7. Jeśli pracujesz na koncie administratora, przejdź do: `Panel sterowania-> System` i w górnym panelu wybierz `Edytuj zmienne środowiskowe systemu`. Jeżeli pracujesz na koncie lokalnym, wybierz opcję `Edytuj zmienne środowiskowe dla konta`.

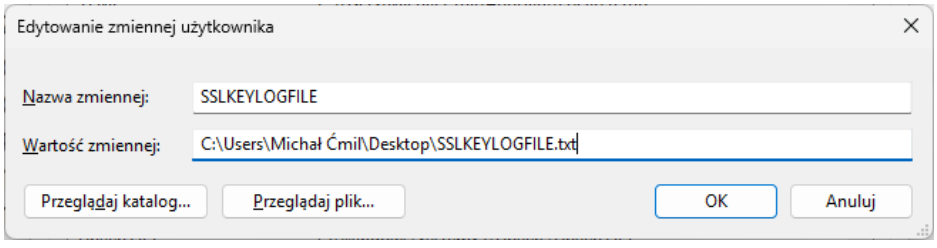


8. W oknie `Zmienne środowiskowe` w sekcji `Zmienne użytkownika dla <nazwa_użytkownika>` kliknij przycisk `Nowa...`.



9. Jako Nazwa zmiennej wpisz: `SSLKEYLOGFILE`.

10. Kliknij `Przeglądaj katalog...` i ustaw ścieżkę na `Pulpit`. Na końcu ścieżki dopisz nazwę pliku, np.: `SSLKEYLOGFILE.txt`.



11. Zamknij wszystkie okna przyciskiem `OK`.

12. Uruchom ponownie przeglądarkę Firefox i spróbuj ponownie wejść na stronę *sekurak.pl*.

13. Na pulpicie powinien pojawić się plik `SSLKEYLOGFILE.txt` (jeżeli go nie widać, odśwież pulpit).

14. Otwórz ten plik i upewnij się, że zawiera wpisy z kluczami.

15. Wróć do Wiresharka i przejdź do `Edycja -> Preferencje -> Protocols -> TLS`.

16. W polu `(Pre)-Master-Secret log filename` wskaż plik `SSLKEYLOGFILE.txt` znajdujący się na pulpicie.

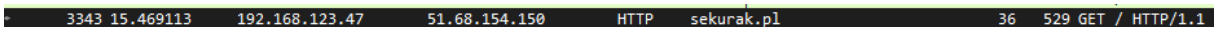
17. Zatwierdź zmiany przyciskiem `OK`.

18. Wróć do Wiresharka i ustaw filtr:

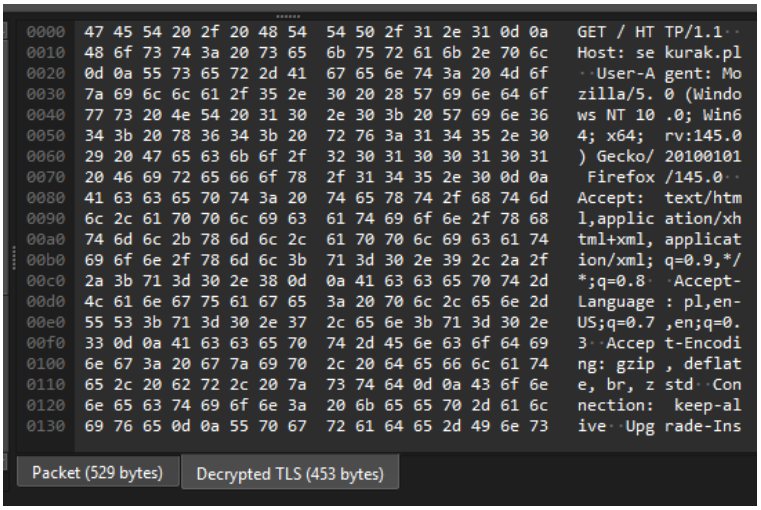
```
http.request.method == GET
```

19. Uruchom ponownie przeglądarkę Firefox i przejdź na stronę *sekurak.pl*.

20. Wróć do Wiresharka i zlokalizuj pakiet GET kierowany do *sekurak.pl*.



21. Zwróć uwagę, że w dolnym panelu pojawia się informacja Decrypted TLS – oznacza to, że Wireshark potrafi odczytać nagłówki HTTP z odszyfrowanego ruchu.

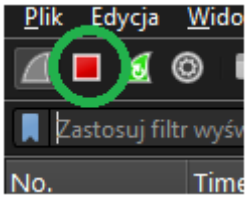


22. Aby sprawdzić, czy możemy odczytać więcej danych, kliknij prawym przyciskiem myszy na wybrany pakiet GET i wybierz: `Follow -> TLS Stream`.

Jak widzisz nagłówki da się odczytać, ale zawartości strony już nie – wynika to nie z szyfrowania, a z kompresji. Spójrz na nagłówki w górnej części okna: Ten Content-Encoding: gzip mówi, że treść strony jest spakowana gzipem. Dlatego pod nagłówkami widzisz „krzaki" – to nie jest HTML, tylko skompresowane dane.

23. Zamknij okno `Śledź strumień`.

24. Zatrzymaj przechwytywanie ruchu w Wiresharku.

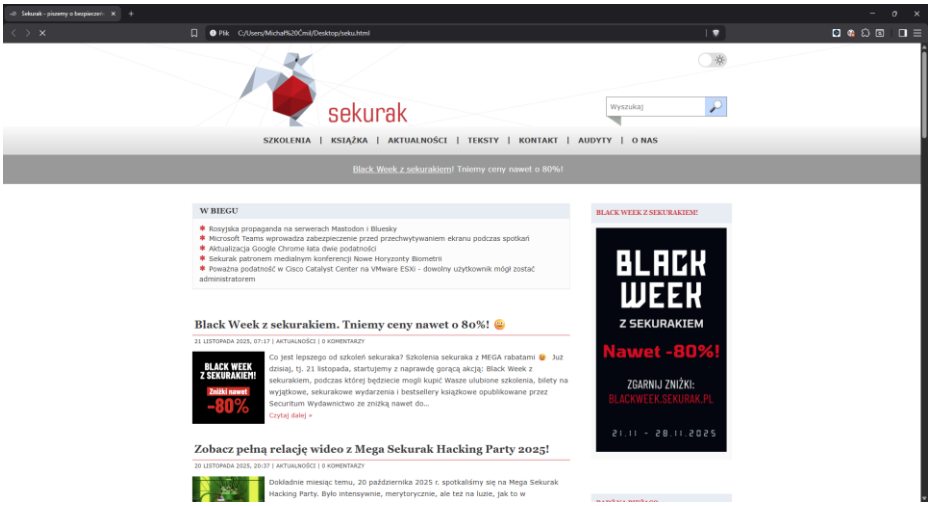


25. W głównym oknie Wiresharka wybierz: `File -> Export Objects -> HTTP…`.

26. W oknie, które się pojawi, znajdź na liście obiekt `text/html` z hostem `sekurak.pl`.

27. Zaznacz go, kliknij `Save` i zapisz jako np. `sekurak.html`.

28. Uruchom (otwórz w przeglądarce) zapisany plik.



Strona powinna wyglądać normalnie, mimo że adres URL będzie wskazywał na lokalny plik, np. `C:/Users/.../sekurak.html`.

Takie działanie można zastosować np. podczas analizy ruchu użytkownika, który padł ofiarą phishingu – po wpisaniu swoich danych na fałszywej stronie. Odtwarzając w ten sposób zawartość strony, analityk może w kolejnym kroku poddać ją dalszej analizie, np. za pomocą **BurpSuite**, narzędzia przeznaczonego do testów bezpieczeństwa aplikacji webowych.

Istnieją także inne metody, ale wszystkie pozwalają na efektywne deszyfrowanie ruchu SSL/TLS w Wiresharku, co jest szczególnie przydatne w diagnostyce i analizie bezpieczeństwa sieci.

29. Wróć do punktu 17 i wykonaj całą procedurę, ale tym razem dla strony zsz.prz.edu.pl. Na końcu wykonaj zrzut ekranu analogiczny do tego z punktu 26 (strona załadowana z lokalnego pliku HTML) i dołącz go do sprawozdania.