Internet – wybrane aspekty bezpieczeństwa

Mariusz Piwiński
Instytut Fizyki
Wydział Fizyki, Astronomii i Informatyki Stosowanej
Uniwersytet Mikołaja Kopernika,
ul. Grudziądzka 5/7, 87-100 Toruń
e-mail:Mariusz.Piwinski@fizyka.umk.pl

Abstract: In the last decade we can observe the rapid grow of usage computer networks in all aspects of our life. The Internet has became the part of our life, and the web browser became one of the most important computer program. We use computers for shopping, working, keep in touch with friends, spending spare time, but usually we don't think about our safety in the virtual world. We have to be aware that the Internet is very convenient but also complicated and potentially dangerous technology.

1. Wstęp

Na przestrzeni ostatniej dekady mamy do czynienia z gwałtownym wzrostem wykorzystania komputerów we wszystkich sferach życia. Obecnie większość ludzi na świecie nie wyobraża sobie funkcjonowania bez komputera podłączonego do Internetu. Sytuacja ta jest o tyle zaskakująca, iż jeszcze 30 lat temu słowo komputer kojarzyło się wyłącznie z urządzeniem, którego obsługą mogą zajmować się tylko wyspecjalizowani użytkownicy. Faktycznie początkowo urządzenia te były wykorzystywane

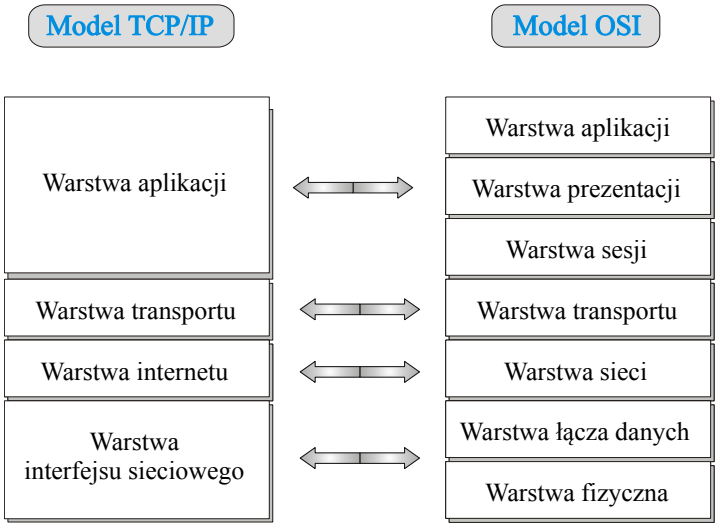
głównie do ściśle określonych, specjalistycznych celów i nikt nie zakładał, iż w przeciągu kilkudziesięciu następnych lat staną się one nieodłączną częścią naszego życia. Zatem warto zadać pytanie: Co wpłynęło na tak dynamiczny rozwój oraz upowszechnienie tej technologii? Jako jedną z głównych przyczyn masowej popularyzacji komputerów wskazuje się możliwość podłączenia ich do globalnej sieci. Od tego momentu urządzenia te zaczęły być wykorzystywane do komunikacji z otaczającym światem. Istnienie różnego rodzaju komunikatorów, portali społecznościowych, sklepów internetowych, serwisów informacyjnych oraz innych usług internetowych otworzyło przed użytkownikami zupełnie nowe możliwości. Gama oferowanych usług jest tak szeroka, iż zaczęto wręcz mówić o „wirtualnym świecie", w którym użytkownik może poczuć się zupełnie inną osobą. Co więcej, ogólne wrażenie anonimowości w globalnej sieci skłania często do opinii, iż w tym „równoległym świecie" nie do końca obowiązują zasady ustalone w świecie realnym. Oczywiście odczucie to jest bardzo złudne i w powiązaniu z brakiem podstawowej wiedzy dotyczącej sposobu działania sieci komputerowych, może okazać się potencjalnie niebezpiecznym. W większości przypadków użytkownicy nie mają świadomości w jaki sposób działają wykorzystywane przez nich technologie, traktując tę wiedzę jako zupełnie zbyteczną. Co więcej do takiego stanu rzeczy przyczyniają się również producenci sprzętu oraz oprogramowania, którzy tworząc „zautomatyzowane rozwiązania" coraz bardziej eliminują potrzebę podejmowania decyzji przez użytkownika. Obecnie coraz częściej w masmediach pojawiają się doniesienia o różnych atakach na komputery podłączone do sieci oraz próbach wykradzenia poufnych danych. Powszechnie utrzymuje się pogląd, iż problemy te dotyczą tylko wielkich firm, korporacji czy instytucji. Niestety należy mieć świadomość, iż podłączenie komputera do sieci powoduje narażenie go na potencjalne niebezpieczeństwa związane z sieciami komputerowymi.

2. Globalna sieć - Internet

Za początki Internetu uważa się lata 60-te ubiegłego stulecia kiedy amerykańska agencja rządowa DARPA (*ang. Defense Advanced Research Projects Agency*) zajmująca się rozwojem technologii wojskowych rozpoczęła prace nad projektem ARPANET mającym na celu zbudowanie rozproszonej sieci komputerowej. W swoich założeniach sieć ta nie miała mieć wyróżnionego jednego punktu centralnego, co oznaczało, iż mogłaby ona również działać w przypadku jej częściowego zniszczenia. Założenia te bardzo istotne dla projektów realizowanych w okresie tzw. „zimnej wojny", okazały się również kluczowe dla sposobu działania globalnej sieci. Pierwsze przesłanie danych za pomocą sieci ARPANET udało się zrealizować we wrześniu 1969 roku pomiędzy UCLA *(*University of California, Los Angeles) a SRI (Stanford Research Institute, Menlo Park, California). Wydarzenie to powszechnie uznaje się za moment narodzin Internetu.[1] Jednocześnie pojawiła się potrzeba stworzenia oprogramowania umożliwiającego komunikację pomiędzy wieloma urządzeniami. Ponadto tak stworzona sieć do swojej obsługi wymagała również zastosowania urządzeń pośredniczących, które analizując zaadresowane dane miały podejmować decyzję w którym kierunku należy je przesyłać. Ze względu na duży stopień skomplikowania całego zagadnienia, do jego opisu zdecydowano się na zastosowanie modelu warstwowego, w którym każdy z poziomów jest odpowiedzialny za realizację określonych funkcji. W takim rozwiązaniu protokół pracujący w warstwie nadrzędnej może wykorzystywać funkcjonalności protokołu pracującego w warstwie podrzędnej. Jednocześnie pełni on funkcję usługodawczą wobec protokołu pracującego w warstwie nadrzędnej. Należy przy tym pamiętać, iż dane wysłane przez urządzenie źródłowe przez protokół działający w warstwie X, po dotarciu do urządzenia docelowego muszą być analizowane przez ten sam protokół działający w tej samej warstwie X. Ponadto takie podejście jest bardzo istotne ze względu na sieciowe urządzenia pośredniczące,

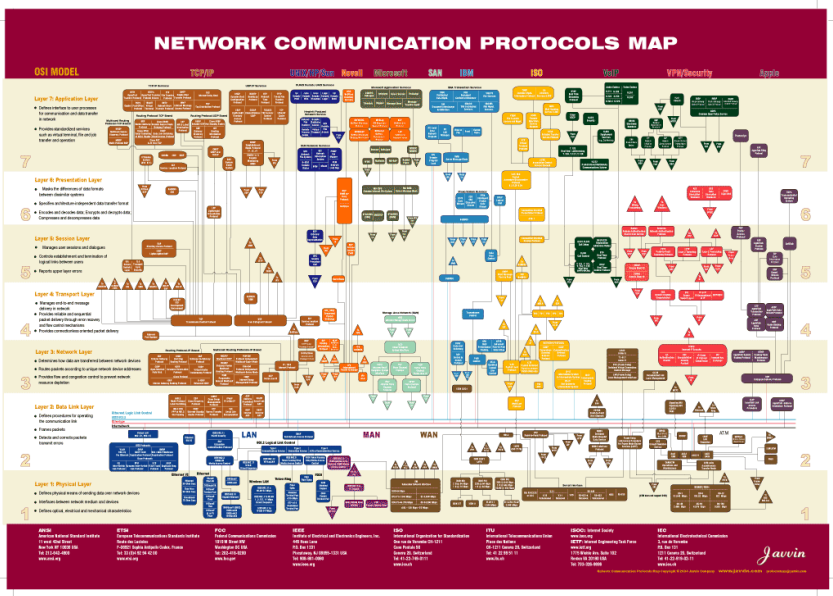
które nie zawsze działają we wszystkich warstwach takiego modelu, gdyż ich zadaniem jest tylko przesyłanie danych, a nie ich pełna analiza.

Słowo Internet oficjalnie funkcjonuje od 1974, kiedy to w dokumencie RFC 675 opublikowanym pod tytułem „Specification of Internet Transmission Control Program" definiującym sposób działania protokołu TCP użyto go jako nazwy rozproszonej sieci. Podczas prac nad projektem ARPANET stworzono różne protokoły komunikacyjne, które ostatecznie doprowadziły do standaryzacji zestawu protokołów TCP/IP. Zdefiniowano tutaj cztery warstwy, które określały w jaki sposób dane powinny być sformatowane, zaadresowane i przesyłane tak, aby wysłane przez nadawcę dotarły do urządzenia docelowego. W 1984 roku na bazie doświadczeń modelu TCP/IP organizacja ISO (*ang. International Organization for Standardization*) stworzyła model OSI składający się aż z siedmiu warstw. Nie oznacza to jednakże, że w obu modelach działają zupełnie inne protokoły. Jest wręcz przeciwnie, a co więcej można wręcz mówić nawet o równoważności poszczególnych warstw w obu modelach. Siedem warstw w modelu OSI jest zatem przejawem zwiększającego się zaawansowania stosowanych technologii, a co za tym idzie również pojawiających się nowych protokołów. Zestawienie obu modeli przedstawia Rysunek 1.



Rysunek 1 Porównanie modeli TCP/IP oraz OSI [2]

Jak zatem na tle tych modeli wyglądają zależności pomiędzy protokołami? Odpowiedź na to pytanie przedstawia mapa protokołów sieciowych (Rysunek 2).[3]



Rysunek 2 Mapa protokołów sieciowych

Należy zwrócić uwagę, iż protokoły te zostały przedstawione na tle modelu OSI z uwzględnieniem innych modeli. Jak widać ilość stosowanych rozwiązań jest tak szeroka, iż przedstawienie tego zestawienia w czytelnej formie bez zastosowania minimalnego formatu A2 jest praktycznie niemożliwa. Dlatego też, wnikliwych czytelników zachęcam do odwiedzenia strony *http://www.sharewareconnection.com/network-protocols-map-poster.htm*, skąd poster ten został zaczerpnięty.

Podsumowując rozważania dotyczące Internetu należy stwierdzić, iż w przypadku komunikacji pakietowej mamy do czynienia z wykorzystaniem wielu protokołów sieciowych oraz sieciowych urządzeń pośredniczących. Złożoność ta wpływa bezpośrednio na problemy związane z zapewnieniem bezpieczeństwa dla całej transmisji danych.

3. Strony WWW czyli komunikacja klient serwer

Obecnie strony WWW stanowią nieocenioną bazę informacji. Dzięki dostępowi do odpowiednich serwisów WWW możemy korzystać z banków, realizować zakupy czy też rozliczać swoje zobowiązania podatkowe. Jednakże czy sposób, w który korzystamy z tych usług jest bezpieczny? Postarajmy się prześledzić ten proces przez pryzmat poznanych już modeli sieciowych. Przeglądarka sieciowa, którą obecnie wykorzystuje każdy użytkownik Internetu jest typowym oprogramowaniem klienckim umożliwiającym komunikowanie się z wybranym serwerem WWW. A zatem komunikacja ta jest typowym przykładem realizacji połączenia typu klient serwer. Oznacza to, iż klient wysyła żądania do serwera, w wyniku czego serwer odsyła zwrotnie do klienta odpowiednie dokumenty, które następnie są interpretowane przez przeglądarkę. Twórcą oprogramowania realizującego takie funkcjonalności był sir Tim Berners-Lee, który pracując w CERN starał się rozwiązać problem dotyczący dostępu wielu użytkowników do wspólnej bazy danych. Prace te skłoniły go do budowy systemu

hipertekstowego World Wide Web (WWW), którego głównym celem było zbieranie zasobów wiedzy ludzkiej i dzielenie się nią z innymi użytkownikami. Jednocześnie zastosowany w dokumentach system hiperłączy umożliwiał użytkownikowi przenoszenie się pomiędzy udostępnionymi dokumentami. W trakcie realizacji tego projektu stworzony został pierwszy serwer oraz przeglądarka WWW. Prawdziwym przełomem w wykorzystaniu serwisów WWW było połączenie tej technologii Internetem. W tym celu został opracowany system ogólnodostępnych, unikalnych identyfikatorów zasobów sieci UDI (ang. The Universal Document Identifier), język służący projektowaniu stron HTML (ang. HyperText Markup Language) oraz protokół służący do przesyłania dokumentów tekstowych HTTP (ang. Hypertext Transfer Protocol). World Wide Web w przeciwieństwie do innych podobnych projektów takich jak protokół Gopher, nie została skomercjalizowana, co umożliwiło dalszy rozwój serwerów WWW oraz publikowanych na nich stron. [4]

W tym miejscu warto zwrócić uwagę, iż bardzo często pojęcia Internet oraz World Wide Web w sposób mylny są traktowane jako swoje synonimy. Należy podkreślić, iż Internet, to globalna sieć łącząca ze sobą komputery na całym świecie, a WWW jest jedną z usług, które ta sieć może zapewniać.

Zajmijmy się przez moment samym protokołem HTTP, z którego korzystają wszystkie przeglądarki sieciowe. Biorąc pod uwagę model warstwowy można powiedzieć, iż pracuje on w warstwie aplikacji modelu OSI. Jak wskazuje nazwa protokołu, został on stworzony do przesyłania dokumentów hipertekstowych, ale wprowadzenie w 1993 przeglądarki Mosaic pracującej w trybie graficznym szybko wymusiło pojawianie się rozwiązania umożliwiającego przesyłanie za jego pomocą również dokumentów w innych formatach. Protokół ten jest cały czas rozwijany przez konsorcjum W3C, a ostatnia jego wersja (HTTP/1.1) została w sposób szczegółowy opisana w dokumentach RFC 1945 i RFC 2616. W sposób ogólny można go scharakteryzować w następujący sposób:

- służy do przesyłania dokumentów udostępnianych przez serwery WWW,

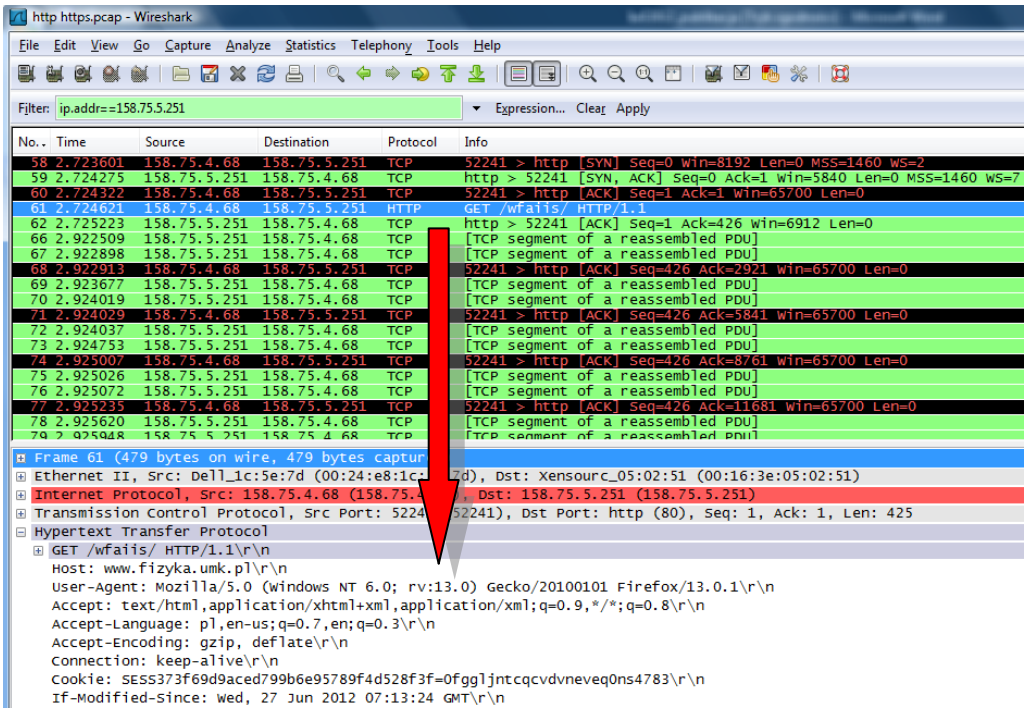
- pozwala na przesłanie dokumentów tekstowych i binarnych,
- oparty na protokole TCP (port 80, alternatywnie 8080),
- oparty na modelu żądanie-odpowiedź,
- wykorzystuje znakowe komendy i komunikaty,
- po dostarczeniu dokumentu może podtrzymać lub zamknąć połączenie,
- jest protokołem bezstanowym i bezsesyjnym,
- wszystkie dane przesyłane są otwartym tekstem.

Z powyższej charakterystyki widać, iż z założenia stosunkowo prosty protokół HTTP standardowo nie posiada w sobie mechanizmów, które zapewniałyby bezpieczeństwo przesyłanych danych. Brak kryptowania informacji oznacza, iż wszystkie dane przesyłane zarówno w żądaniach klienta jak i odpowiedziach serwera przesyłane są otwartym tekstem, to oznacza, iż każda osoba, która je przechwyci będzie mogła je swobodnie odczytać. Należy zwrócić uwagę, iż dotyczy to zarówno przesyłanych dokumentów jak i danych wysyłanych za pomocą formularzy. W celu rozwiązania tego problemu stworzono protokół HTTPS, który funkcjonalnie jak protokół HTTP, ale dodatkowo jest on wspierany przez protokół TLS (*ang. Transport Layer Security*). Protokół TLS stanowi rozwinięcie protokołu SSL (*ang. Secure Socket Layer*), który pierwotnie został stworzony w 1994 roku przez firmę Netscape do bezpiecznego przesyłania danych. W 1999 roku grupa robocza Transport Layer Security powołana przez organizację Internet Engineering Task Force opublikowała pierwszą wersję protokołu TLS. Protokół ten działa w warstwie prezentacji modelu OSI, dzięki czemu może zabezpieczać protokoły warstwy wyższej nie tylko HTTP (HTTPS), ale również FTP, SMTP, POP, czy IMAP. Wykorzystując szyfrowanie symetryczne, asymetryczne oraz certyfikaty X.509 pozwala on na uwierzytelnianie serwera, klienta oraz zapewnienie poufności przesyłanych danych. W 2009 w tym, jak się wydawało bezpiecznym protokole, zauważono lukę umożliwiającą atak w trakcie procesu renegocjacji sesji. Ze

względu na fakt, iż błąd ten dotyczył sposobu działania protokołu, jedynym sposobem obrony było wyłączenie możliwości renegocjacji sesji. W późniejszym okresie zaproponowano rozszerzenie protokołu eliminujące tą lukę w zabezpieczeniach.[5,6,7]

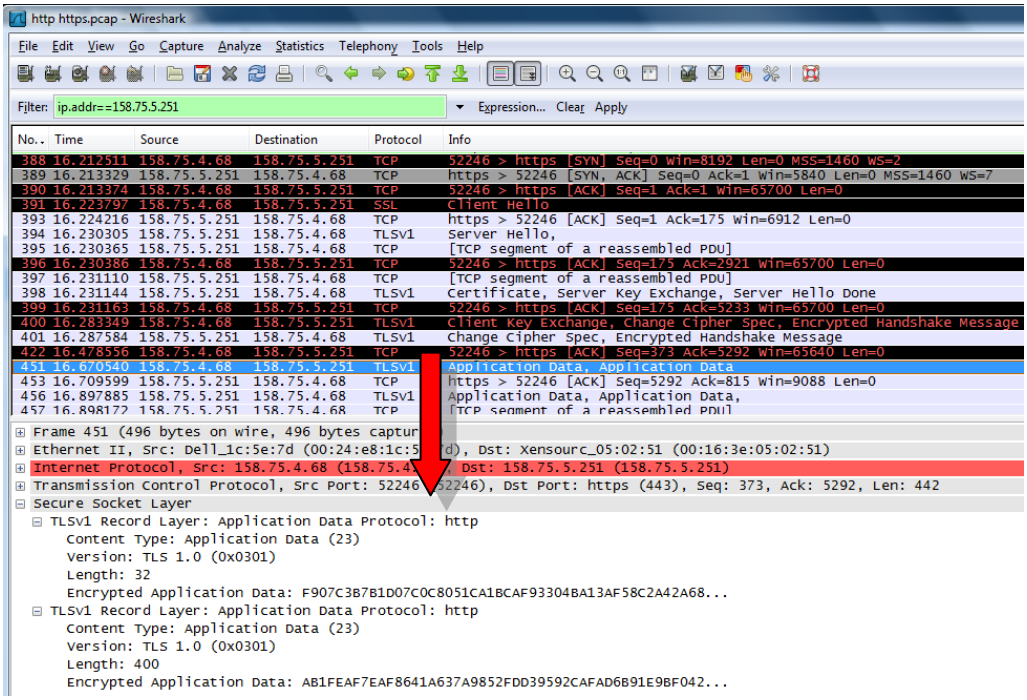
W celu obserwacji różnic między sposobem działania protokołu HTTP oraz HTTPS wystarczy wykorzystując program Wireshark przechwycić transmisję realizowaną z wykorzystaniem tych protokołów. Takie przykładowe transmisje przedstawiono na Rysunkach 3 i 4.

Osoby nie posiadające doświadczenia w pracy z programem Wireshark, mogą jako wstęp do analizy przechwytywanych danych wykorzystać dostępne na stronie www.wireshark.org liczne samouczki lub też opracowanie „Praktyczna analiza sieci komputerowych z wykorzystaniem programu Wireshark" mojego autorstwa.[8]



Rysunek 3 Przechwycona transmisja realizowana za pomocą protokołu HTTP

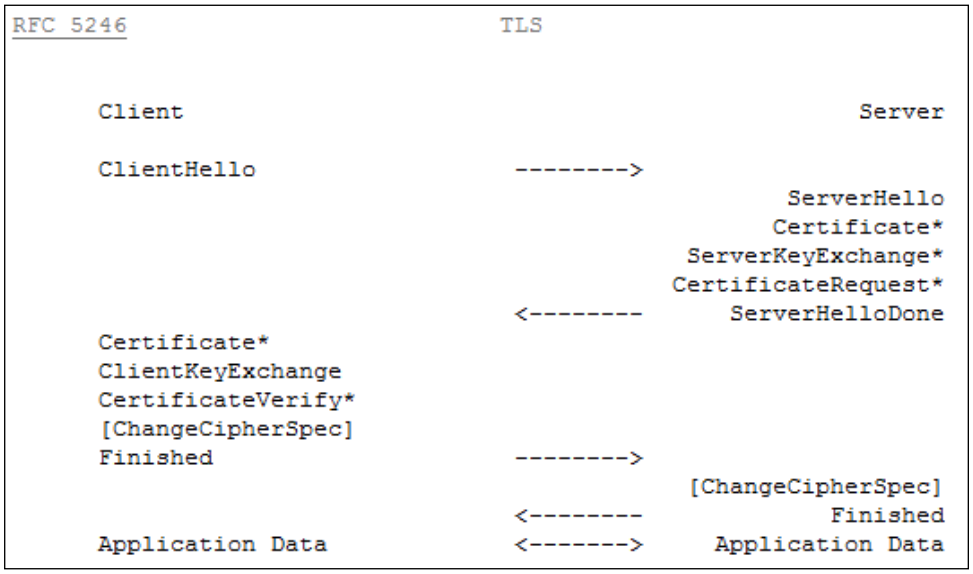
W przypadku protokołu HTTP wyraźnie widać postać zapytania GET wysyłanego przez klienta do serwera WWW. Z łatwością można sprawdzić, że przechwycone zapytanie dotyczyło strony www.fizyka.umk.pl i zostało wysłane z przeglądarki Firefox 13.0.1. Warto również zwrócić uwagę, iż protokół HTTP nie zapewnia również poufności zmiennym Cookie, które jak widać na zamieszczonym rysunku również przesyłane są w jawny sposób.



Rysunek 4 Przechwycona transmisja realizowana za pomocą protokołu HTTPS

Wywołanie tej samej strony przy użyciu protokołu HTTPS daje zupełnie odmienny efekt. Na pierwszy rzut oka zamiast pakietu protokołu HTTP widać pakiety związane z protokołem TLSv1. Ich wymiana pomiędzy klientem a serwerem ma za zadanie zweryfikowanie certyfikatu serwera oraz ustanowienie kryptowanego połączenia słu-

żącego do przesyłania danych. Schemat wymiany informacji pomiędzy klientem a serwerem opisany w dokumencie RFC 5246 został przedstawiony na Rysunku 5.



Rysunek 5 Przepływ informacji realizowany w celu ustanowienia sesji TLS [9]

Analizując przechwycone dane związane z ustanowieniem szyfrowanego kanału komunikacyjnego można zaobserwować poszczególne elementy tej komunikacji przedstawione na Rysunku 5. Jako pierwszy widoczny jest tutaj pakiet *Client Hello*, w którym klient wysyła do serwera między innymi informacje o obsługiwanej wersji protokołu SSL, dozwolonych sposobach szyfrowania, kompresji danych, identyfikator sesji oraz losową liczbę wykorzystywaną w późniejszym procesie generowania klucza. W odpowiedzi serwer przesyła wiadomość *Server Hello* zawierający podobne informacje. Kolejny krok polega na przesłaniu przez serwer swojego certyfikatu, klucza publicznego oraz informacji o zakończeniu tego etapu nawiązywania połączenia (wiadomość *Certificate, Server Key Exchange, Server Hello Done*). Po otrzymaniu tych informacji oprogramowanie klienckie weryfikuje przesłany certyfikat, autentykując tym samym serwer. W przypadku gdy autentykacja ta nie jest możliwa, użytkownik zostaje powiadomiony o problemie oraz poinformowany, iż kryptowane połącze-

nie może zostać ustanowione. Poprawna identyfikacja serwera pozwala na przejście do następnego kroku, w którym klient na podstawie uzyskanych danych tworzy wstępny klucz sesji i wysyła go serwerowi (wiadomość *Client Key Exchange*). Dla zapewnienia bezpieczeństwa jest on zaszyfrowany przy użyciu przesłanego przez serwer klucza publicznego. Następnie klient i serwer posługując się dwoma liczbami losowymi przesłanymi uprzednio przez klienta i serwer oraz wstępnym kluczem sesji generują klucz sesji, który jest wykorzystywany do kryptowania przesyłanych danych. Szyfrowanie to jest symetryczne (najczęściej jest to algorytm DES), co oznacza, iż ten sam klucz jest wykorzystywany zarówno do kodowania jak i dekodowania informacji. Następnie klient zawiadamia, że dokonał wszystkich niezbędnych operacji i od tej pory będzie przesyłał dane wykorzystując szyfrowany kanał (wiadomość *Change Cipher Spec*). Jednocześnie w pakiecie tym przesyłana jest zakryptowana informacja powiadamiająca serwer o tym, iż proces ustanowienia szyfrowanego połączenia dobiegł końca (wiadomość *Encrypted Hadshake Message*). Analogiczne dane (wiadomość *Change Cipher Spec, Encrypted Hadshake Message*) przesyłane są z serwera do klienta. Ten ostatni proces pozwala ostatecznie utworzyć kanał kryptograficzny i zweryfikować poprawność jego działania. Jeżeli oba urządzenia w sposób poprawny odczytają zakryptowaną informację proces ustanowienia szyfrowanego połączenia dobiegnie końca. Od tego momentu przesyłane pakiety będą już szyfrowane. W naszym przypadku szyfrowanie było związane z protokołem HTTPS, co należy rozumieć jako protokół HTTP szyfrowany za pomocą TLS. Analizując kolejne przechwycone dane możemy zauważyć, iż pakiety zawierają nagłówki protokołu TLS, z których możemy dowiedzieć się, że dotyczą protokołu HTTP, ale treść ich jest już w pełni zaszyfrowana. Tak realizowane nawiązywanie szyfrowanego kanału komunikacyjnego zapewnia zgodność kluczy sesji generowanych po stronie klienta i serwera przy zachowaniu ich poufności. Ostatecznie bezpieczeństwo takiego połączenia bardzo mocno zależy od długości generowanego klucza sesji. Pierwotnie ze względu na ograniczenia prawne obowiązujące w Stanach Zjednoczonych dotyczące eksportu technik kryptograficznych większość implementacji SSL nie mogła wykorzystywać kluczy dłuższych niż

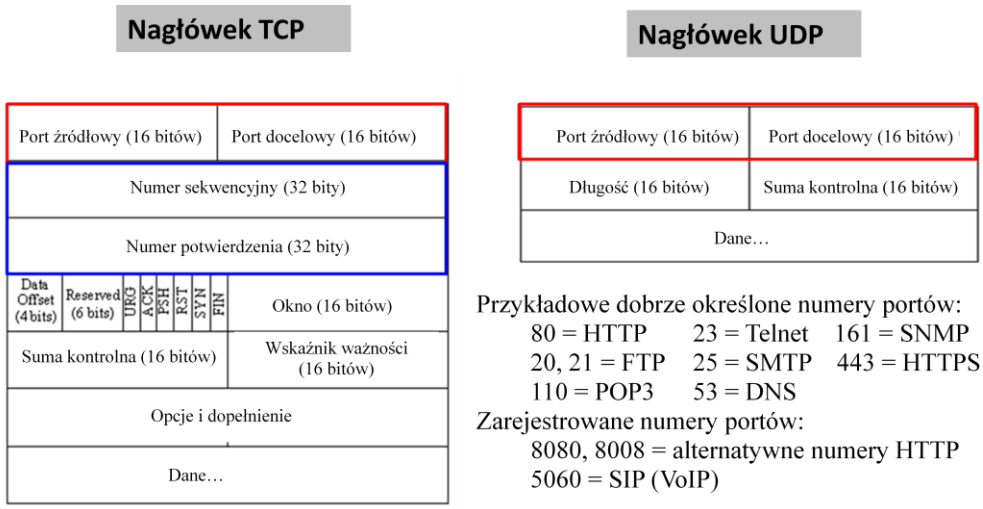
40 bitów. Wymóg ten umożliwiał agencjom bezpieczeństwa (i nie tylko) dysponującymi dużymi mocami obliczeniowymi złamanie tego szyfru przy użyciu podejścia „*brute-force*", co w skrócie oznacza dopasowanie rozwiązania poprzez sprawdzenie wszystkich możliwości. Obecnie po usunięciu tych ograniczeń klucze 40-bitowe zostały zastąpione kluczami posiadającymi minimum 128-bitów.

Podsumowując należy stwierdzić, iż protokół HTTPS, a ściślej mówiąc wspierający go protokół TLS uważany jest powszechnie za bezpieczny sposób przesyłania danych. Główne problemy związane z atakami na taką transmisję danych dotyczą najczęściej błędów implementacji protokołu w oprogramowaniu. Ze względu na warstwę, w której działa HTTPS typowymi atakami, z którymi tutaj możemy mieć do czynienia są ataki typu „*Man in the middle*". Atak ten polega na podsłuchu i modyfikacji przesyłanych kryptowanych informacji w celu ich deszyfracji. Może on dotyczyć np. próby uzyskania lub modyfikacji przesyłanego klucza, który będzie wykorzystywany do zabezpieczenia kanału komunikacyjnego. Ze względu na rozległość wykorzystywanych sieci oraz różne lokalizacje, z którymi realizowana jest komunikacja, źródło takiego ataku może znajdować się w dowolnym węźle sieci przez, który przesyłane są dane. Należy zaznaczyć, iż protokół ten jest cały czas udoskonalany, a w kolejnych jego wersjach publikowanych w dokumentach RFC twórcy rozwiązują pojawiające się problemy dotyczące bezpieczeństwa. W związku z powyższym w celu zapewnienia maksymalnego bezpieczeństwa trzeba dbać, aby w szyfrowanej komunikacji wykorzystywana była najnowsza wersja tego protokołu.

4. Protokoły warstwy transportowej

Mówiąc o protokołach HTTP oraz HTTPS należy pamiętać, iż wykorzystują one protokół TCP (*ang. Transmission Control Protocol*) pracujący w warstwie transportowej. Jego zadanie polega na dzieleniu strumienia danych na segmenty o wielkości

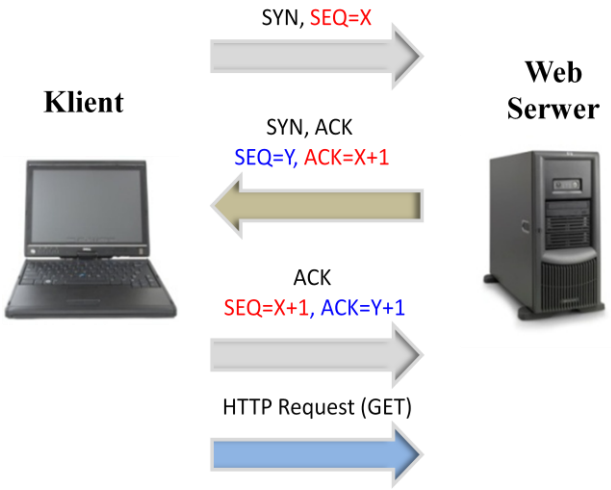
dopasowanej do tworzonych przez niższą warstwę pakietów. Ponadto sprawuje on kontrolę nad przesyłanymi danymi. W odróżnieniu od protokołu UDP (*ang. User Datagram Protocol)* wyposażony jest on w szereg mechanizmów mających na celu zapewnienie jak najszybszego oraz skutecznego dostarczenia przesyłanych danych do odbiorcy. Niestety przejawia się to w ilości pól, które zostały wykorzystane w jego datagramie. Oznacza to, iż protokół ten w przeciwieństwie do protokołu UDP zapewnia niezawodność dostarczenia danych, ale przez to jest od niego wolniejszy. Zestawienie datagramów obu protokołów przedstawia Rysunek 6.



Rysunek 6 Zestawienie datagramów protokołów TCP i UDP

Jak łatwo zaobserwować oba protokoły posiadają pola „Port źródłowy" oraz „Port docelowy", które pozwalają na wskazanie których aplikacji i związanych z nimi protokołów warstwy wyższej ma dotyczyć obsługiwana komunikacja. Podstawę działania protokołu TCP stanowi możliwość nawiązania sesji pomiędzy nadawcą a odbiorcą. Oznacza to, iż przed przesłaniem konkretnych danych urządzenie inicjujące połączenie rozpoczyna proces trójstronnego nawiązania sesji (*ang. three-way handshake*). W przypadku transmisji związanej ze stronami WWW urządzeniem tym jest urządzenie klienckie z przeglądarką internetową. Jak już zostało to wspomniane, protokół TCP

może być wykorzystywany do obsługi wielu różnych aplikacji i protokołów działających w warstwie aplikacji. W związku z powyższym musi on zawierać numer portu identyfikujący daną usługę/aplikację, której ma dotyczyć konkretna komunikacja. W przypadku protokołu HTTP numer portu, z którym stowarzyszona jest usługa serwera WWW ma wartość 80, dla HTTPS 443, dla SMTP 25, a w przypadku protokołu TELNET wynosi on 23. Zatem podczas komunikacji dotyczącej protokołu HTTP urządzenie klienckie przed wysłaniem do serwera żądania strony (HTTP GET) wcześniej musi nawiązać z nim sesję TCP (Rysunek 7).



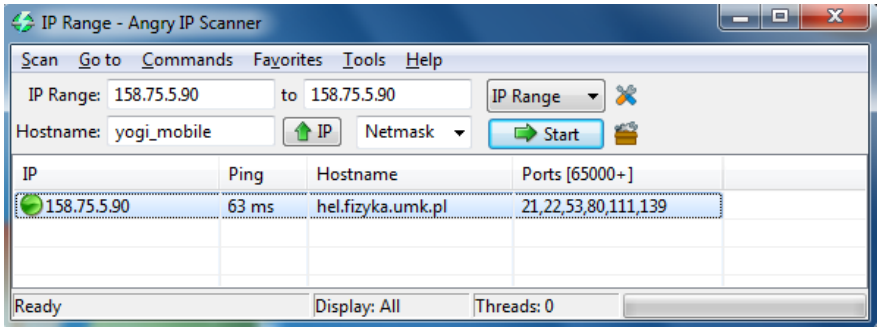
Rysunek 7 Nawiązanie trójstronnego połączenia TCP wraz wysłaniem pakietu dotyczącego żądania strony WWW.

W tym celu w pierwszym pakiecie klient zgłasza swoją chęć nawiązania połączenia z usługą serwera skojarzoną z portem o numerze 80 (ustawiona flaga *SYN*). Jeżeli na serwerze zainstalowano odpowiednie oprogramowanie nasłuchujące żądań dotyczących tego portu, to potwierdzi on swoją gotowość do udzielania odpowiedzi na takie żądania. W tym celu zwrotnie prześle on swoją odpowiedź do klienta (ustawione flagi *SYN* i *ACK*). Następnie klient potwierdza ustanowienie sesji poprzez przesłanie do serwera pakietu z ostatecznym potwierdzeniem (flaga *ACK*). Po takiej wymianie informacji pomiędzy klientem a serwerem zostanie zestawione połączenie. Oznacza

to, iż serwer od tego momentu będzie oczekiwał od klienta na pakiety dotyczące jego usługi stowarzyszonej z portem 80. Klient zaś będzie oczekiwał na odpowiedzi serwera, które będą kierowane do portu związanego z uruchomioną przeglądarką internetową. Taki mechanizm umożliwia realizację kontroli niezawodności połączenia, gdyż każda ze stron tego połączenia wysyłając dane numeruje je (wartość *SEQ*) a następnie oczekuje od urządzenia docelowego potwierdzenia ich otrzymania (wartość *ACK*). W przypadku braku potwierdzenia urządzenie wysyłające dane ponowi ostatnią transmisję. Mechanizm ten nosi nazwę pozytywnej retransmisji danych. Opisywany proces nawiązania sesji TCP przedstawiony ideowo na Rysunku 7 można z łatwością zaobserwować w przypadku protokołu HTTP na Rysunku 3. Podobnie sprawa wygląda w przypadku protokołu HTTPS z tą różnicą, iż po nawiązaniu sesji do serwera nie jest wysyłane żądanie strony, a uruchamiany jest protokół TLS w celu utworzenia szyfrowanego połączenia, co zostało omówione w poprzednim rozdziale.

Warstwa transportowa standardowo nie jest analizowana przez urządzenia obsługujące ruch sieciowy. Oznacza to, iż sesja TCP nawiązywana jest tylko pomiędzy urządzeniami końcowymi i protokół ten nie jest w stanie komunikować się z urządzeniami pośredniczącymi. Z drugiej strony warstwa ta jest kluczową z punktu widzenia bezpieczeństwa sieciowego. Ze względu na fakt, iż numery portów powiązane są z konkretnymi usługami, wysyłając do konkretnego hosta zapytania TCP/UDP z kolejnymi numerami portów i badając uzyskane odpowiedzi można w dosyć łatwy sposób określić jakie usługi są na nim uruchomione. Informacja ta z kolei może posłużyć do dalszego badania możliwych luk w systemie zabezpieczeń związanych z implementacjami różnych protokołów warstwy aplikacji. Ponadto część z tych usług może okazać się niezabezpieczona lub nieskonfigurowana, co potencjalnie stanowi zagrożenie dla bezpieczeństwa systemu. W związku z powyższym większość ścian ogniowych (*ang. Firewall*) stosowanych w systemach operacyjnych filtruje ruch pod kątem wykorzystywanych portów. Jednocześnie administrator sieci LAN w celu ograniczenia możliwości atakowania komputerów z sieci zewnętrznej, może na granicy administrowanej

sieci zastosować urządzenie filtrujące. Jego zadaniem jest analiza przesyłanych danych pod kątem wykorzystywanych numerów portów. W przypadku, gdy jakaś usługa nie powinna być dostępna dla urządzeń znajdujących się w sieci zewnętrznej, urządzenie filtrujące będzie odrzucać wszystkie dane wykorzystujące związany z nią numer portu. Jako typowy przykład można podać tutaj eliminowanie potencjalnie niebezpiecznego protokołu TELNET, poprzez filtrowanie danych wykorzystujących port 23. Protokół ten nie powinien być wykorzystywany w sieciach rozległych ze względu na fakt, iż standardowo wszystkie dane przesyła on otwartym tekstem. W przypadku gdybyśmy chcieli skorzystać z protokołu TELNET wspartego szyfrowaniem SSL/TLS powinniśmy korzystać z portu 992. Bardzo restrykcyjne sieci często stosują filtrowanie, które z zewnątrz dopuszcza ruch związany tylko z portem 80 (HTTP), 443 (HTTPS) oraz 25 (SMTP). W tym miejscu warto stwierdzić, iż istnieje cały szereg programów umożliwiających automatyczne sprawdzenie na jakich numerach portów odpowiada konkretny system. Proces ten nazywa się skanowaniem portów i jest uznawany za atak na system docelowy. Jako przykład takiej aplikacji można podać program Angry IP Scanner, którego działanie przedstawiono na Rysunku 8.



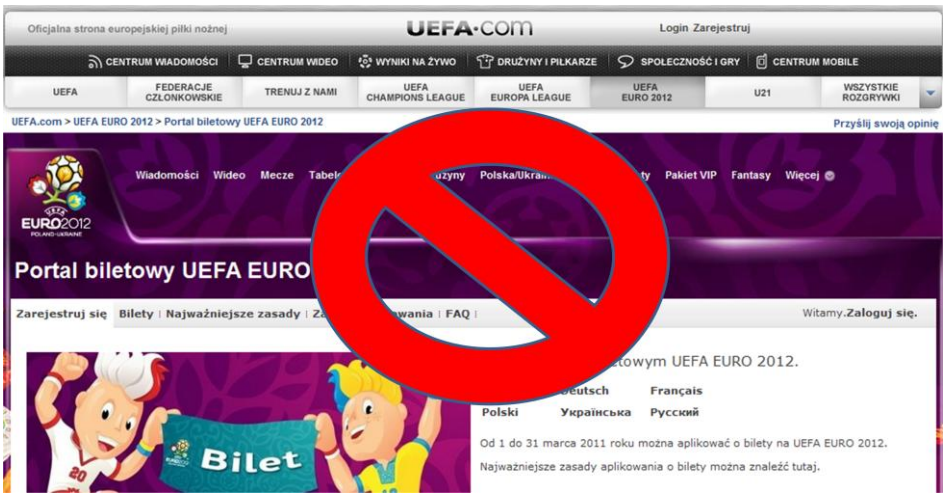
Rysunek 8 Działanie programu Angry IP Scanner.

Jak widać nieodłączną częścią wszystkich serwisów WWW jest obsługa protokołu TCP. Jak się okazuje ten bardzo przydatny protokół może stać się potencjalnym zagrożeniem dla samego serwera. Należy zauważyć, iż każde odwiedziny strony WWW związane są z ustanowieniem sesji TCP pomiędzy serwerem a klientem. Po pobraniu

wszystkich żądanych dokumentów, lub określonym czasie sesja zostaje zamknięta. Podczas nawiązywania sesji serwer przydziela do danej komunikacji pewną określoną część swojej pamięci. W przypadku normalnego wykorzystania serwera nie stanowi to żadnego problemu. Jednakże co stanie się, gdy ilość inicjowanych połączeń będzie na tyle duża, że serwer nie zdoła ich obsłużyć? Z punktu widzenia użytkowników serwis nie będzie dostępny, a sytuacja sama powróci do normy, gdy użytkownicy zrezygnują z odwiedzin tej strony. Sytuacje takie zdarzają się, gdy twórcy serwisów uruchamiają je na serwerach, których zasoby są zbyt małe aby obsłużyć wszystkie żądania. Nie jest to typowy atak, a raczej wynik bardzo dużej popularności. Jako przykład takiej sytuacji można podać dzień 1.03.2011 roku, kiedy na serwerze UEFA uruchomiono możliwość zapisów na bilety dotyczące rozgrywek EURO 2012 (Rysunek 9). Pomimo informacji, iż kolejność zgłoszeń nie ma najmniejszego wpływu na wynik losowania, tego dnia ilość osób pragnących zarejestrować się w tym systemie była tak duża, iż serwer praktycznie przestał działać.

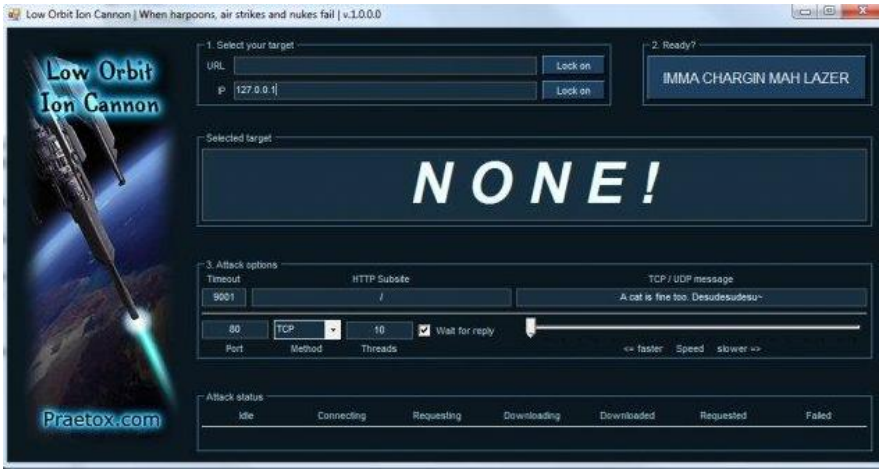
Oprócz takich przypadkowych sytuacji niestety coraz częściej pojawiają się celowe ataki na serwery WWW mające na celu zablokowanie stron WWW. Ataki te nazywane są atakami DoS (*ang. Denial of Service*), w przypadku gdy atak ten przeprowadzany jest z wielu miejsc jednocześnie nazywany jest atakiem DDoS (*ang. Distributed Denial of Service*). Na czy może polegać ten atak? Z punktu widzenia serwisu nie różni się on niczym od sytuacji, w której bardzo wielu użytkowników w tym samym czasie wysyła żądanie dotyczące nawiązania sesji. Jak się jednak okazuje źródłem takich pakietów jest jedno lub klika urządzeń wyposażonych w oprogramowanie, które z maksymalną prędkością wysyła do serwera żądania nawiązania połączenia TCP, za każdym razem używając innego numeru portu źródłowego. Ostatecznie prowadzi to do przeciążenia serwera i zablokowania zapewnianej przez niego usługi. Ataki takie stały się powszechne ze względu na fakt, iż istnieje szereg aplikacji, które pozwalają na wykonanie takiego ataku. Co więcej takie oprogramowanie nie wymaga od użyt-

kownika posiadania praktycznie żadnej zaawansowanej wiedzy dotyczącej sposobu działania protokołów.



Rysunek 9 Strona WWW UEFA.

Przykładem takiego oprogramowania może być aplikacja Low Orbit Ion Cannon (Rysunek 10), która została wykorzystana przez wielu użytkowników do unieruchomienia stron rządowych w styczniu 2012 roku. Należy zaznaczyć, iż atak ten nie ma nic wspólnego z włamaniem na serwer, a wykorzystuje tylko standardowe właściwości protokołu TCP. W związku z powyższym obrona przed tego typu atakami jest dosyć skomplikowana.

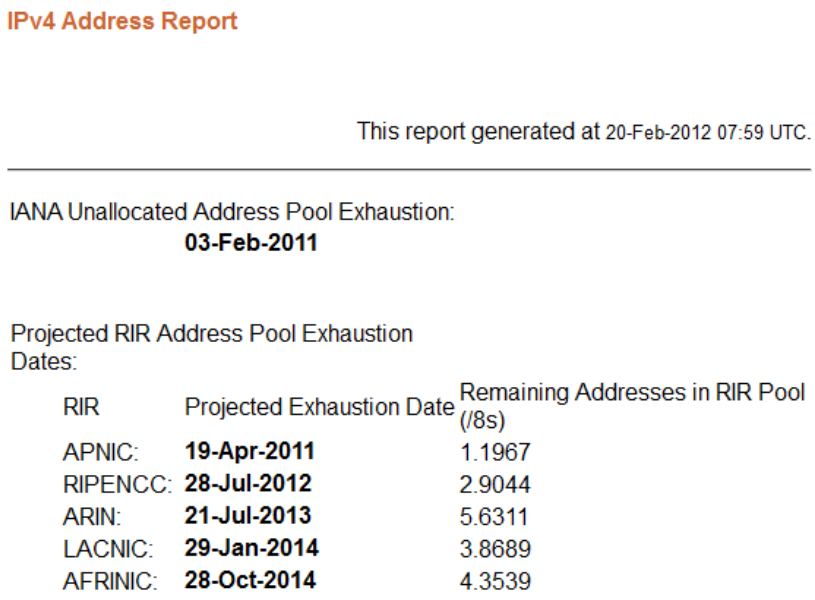


Rysunek 10 Aplikacja Low Orbit Ion Cannon

5. Internet Protocol

Bardzo ważnym aspektem w sieci globalnej było zapewnienie jednoznacznej niepowtarzalnej adresacji dla każdego hosta. Ponadto hierarchiczność przydzielanych adresów miała zapewnić, aby urządzenia znajdujące się na określonym obszarze geograficznym należąc do jednej sieci miały podobne adresy. Takie rozwiązanie umożliwiło określanie w globalnej sieci tras do konkretnych sieci zawierających odpowiednie hosty. Obecnie obowiązującym protokołem pracującym w warstwie sieci jest protokół IP w wersji 4. Ostatecznie został on opisany w dokumencie RFC 791 w 1981 roku.[10] Do adresowania hostów wykorzystuje on 32 bitowy adres, który jednocześnie określa adres sieci (pole sieci) jak i adres samego hosta (pole hosta). Jak widać protokół ten umożliwia równoczesne zaadresowanie około 4 miliardów hostów. Hierarchiczność adresów IP oznacza, iż wszystkie urządzenia działające w tej samej sieci otrzymują adres rozpoczynający się w polu sieci taką samą sekwencją znaków. W momencie tworzenia protokołu IP pula dostępnych adresów wydawała się być praktycznie nieograniczona. W praktyce pomimo rezygnacji z klasowości adresów, uru-

chomienia techniki CIDR, VLSM oraz NAT w 2011 roku nastąpił koniec wolnych adresów IP przydzielanych z puli dostępnej dla organizacji IANA (*ang. Internet Assigned Numbers Authority*) (Rysunek 11). Co prawda dostawcy usług internetowych posiadają jeszcze pewne ograniczone pule adresowe, ale i one wkrótce zostaną wykorzystane, co mogłoby oznaczać brak możliwości podłączenia nowego urządzenia do sieci.



IPv4 Address Report

This report generated at 20-Feb-2012 07:59 UTC.

IANA Unallocated Address Pool Exhaustion:
03-Feb-2011

Projected RIR Address Pool Exhaustion Dates:

RIR	Projected Exhaustion Date	Remaining Addresses in RIR Pool (/8s)
APNIC:	**19-Apr-2011**	1.1967
RIPENCC:	**28-Jul-2012**	2.9044
ARIN:	**21-Jul-2013**	5.6311
LACNIC:	**29-Jan-2014**	3.8689
AFRINIC:	**28-Oct-2014**	4.3539

Rysunek 11 Raport dotyczący dostępności adresów IPv4 dostępny na stronie http://www.potaroo.net/tools/ipv4/

W związku z powyższym od lat 90-tych ubiegłego stulecia rozpoczęto prace nad nową wersją tego protokołu czyli IPv6. Protokół ten wykorzystuje 128 bitową adresację, co powinno zupełnie rozwiązać problem ograniczoności adresów IP (około 10^{38} dostępnych adresów). Ponadto zawiera on w sobie szereg nowych mechanizmów takich jak wsparcie dla mobilności, QoS, optymalizacji routingu oraz możliwości rozszerzania jego funkcji poprzez kaskadowanie nagłówków. Jednocześnie duża liczba

adresów IP oraz nieco odmienna polityka ich przydzielania sprawia, iż przestają mieć sens ataki typu „*brute-force*", które w tym przypadku dotyczyły testowania kolejno wszystkich adresów IP z określonej sieci wraz z kolejnymi numerami portów. Jednocześnie model warstwowy zapewnia, iż wprowadzenie protokołu IPv6 nie będzie wymagało modyfikacji protokołów działających w wyższych warstwach. W tym momencie nasuwa się pytanie dlaczego nie wprowadzono tego protokołu dużo wcześniej. Protokoły warstwy sieciowej są analizowane przez routery i na podstawie zawartych w nich informacji urządzenie sieciowe podejmuje decyzję gdzie dalej należy przesłać dany pakiet. Oznacza to, iż każdy router znajdujący się na drodze danych od hosta źródłowego do docelowego w niezależny sposób musi analizować informacje zawarte w datagramie IP. Wprowadzenie nowej wersji tego protokołu oznaczałoby, iż wszystkie routery na całym świecie musiałyby go obsługiwać. Obecnie mamy do czynienia z sytuacją przejściową, gdzie w wielu miejscach można korzystać zarówno z protokołu IPv4 jak i IPv6.

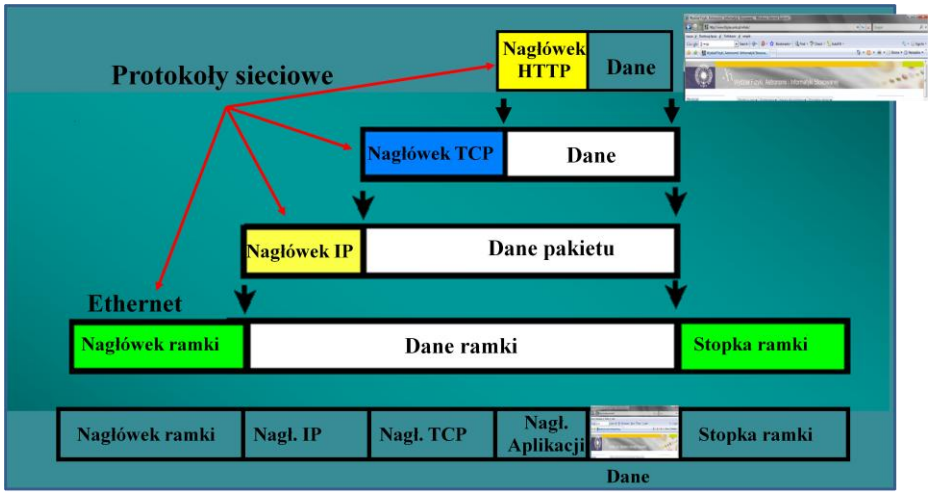
6. Ethernet

Obecnie najczęściej stosowaną technologią dostępową w sieciach LAN związaną z najniższą warstwą modelu TCP/IP jest Ethernet. Jej podwaliny zostały określone przez firmę XEROX PARC w 1973 roku. Do jej popularyzacji w sieciach LAN w znaczący sposób przyczyniło się konsorcjum DIX utworzone przez firmy Digital Equipment Corporation, Intel oraz Xerox. W 1980 określiły one standard komunikacji umożliwiający transmisję z prędkością 10 Mb/s. Opisywał on komunikację w sieciach rozgłoszeniowych, przy wykorzystaniu 48-bitowego adresu sieciowego (źródłowego i docelowego) oraz 16-bitowego pola określającego długość ramki. W tym samym roku IEEE (*ang. Institute of Electrical and Electronics Engineers*) rozpoczął prace nad standardem opisywanym jako 802 mającym na celu dokonanie standaryzacji sieci LAN. W 1982 roku konsorcjum DIX opublikowało poprawioną specyfikację swojego

standardu definiując ją jako Ethernet II. Równoległe prace prowadzone w ramach organizacji IEEE doprowadziły do stworzenia w 1985 roku standardu 802.3, z którego w głównej mierze korzystamy do dnia dzisiejszego. Wraz z rozwojem tej technologii zwiększono prędkość przesyłania danych, jednakże ogólna postać ramki pozostała niezmieniona. Oznacza to, iż do identyfikacji poszczególnych urządzeń w sieci LAN na poziomie tej warstwy wykorzystywany jest 48 bitowy adres fizyczny zapisany na karcie sieciowej. W założeniach niepowtarzalność tych adresów miała zapewniać organizacja IANA (*ang. Internet Assigned Numbers Authority*) przydzielając każdemu z producentów odpowiedni zakres numerów.[11] Niestety pojawienie się na rynku producentów kart sieciowych, którzy nie dbali o unikalność adresów MAC spowodowało, iż czasami w sieci LAN mogą pojawić się dwa urządzenia posiadające karty sieciowe z tym samym adresem MAC. Taka sytuacja prowadzi do wielu błędów w działaniu Ethernetu i powinna być jak najszybciej eliminowana. W celu połączenia ze sobą kliku komputerów w sieci Ethernetowej wykorzystuje się przełączniki wieloportowe. Ich zadaniem jest analizowanie adresów MAC zawartych w przesyłanych ramkach i na ich podstawie przesyłanie danych do odpowiednich portów. Jak widać pojawienie się urządzeń z tym samym adresem MAC powoduje zachwianie sposobu działania tego urządzenia. Przełączniki wieloportowe mogą również stać się przedmiotem ataków DoS. Jeden z nich polega na wysyłaniu przez hosta dużej ilości ramek, które za każdym razem są adresowane innym adresem MAC. Przełącznik odbierając je zakłada, iż każda ramka pochodzi z innego urządzenia, a jego adres MAC należy przypisać do portu na którym została odebrana ramka. Dane te przechowywane są w pamięci przełącznika, która w przypadku takiego ataku zostaje zapełniona w przeciągu kliku minut. Stan taki może doprowadzić do niewłaściwego działania przełącznika.

7. Stos TCP/IP

Dotychczasowe rozważania dotyczące funkcjonowania stosu protokołów TCP/IP zostały podsumowane na Rysunku 12. Jak widać podczas komunikacji z serwisami WWW przeglądarka wykorzystuje domyślnie protokół HTTP, który do przesyłanych danych dołącza swój nagłówek zawierający informacje dotyczące żądań lub odpowiedzi oraz sposobu w jaki dane mają być obsłużone przez odbiorcę. Bezpośrednio protokół ten test obsługiwany przez protokół TCP dbający o nawiązanie sesji, podział przesyłanych danych na segmenty oraz sprawujący kontrolę niezawodności ich dostarczenia do odbiorcy. Na tym etapie każdy segment wyposażony jest w nagłówek TCP, a następnie niezależnie jest wysłany do odbiorcy. W niższej warstwie korzystając z protokołu IP tworzone są pakiety, w których podawane są adresy IP nadawcy i odbiorcy. Tak stworzone pakiety zgodnie z wykorzystywaną technologią dostępową zamykane są w ramkach. W przypadku sieci LAN najczęściej jest to Ethernet, a zatem w tworzonej ramce zostają użyte adresy MAC nadawcy oraz urządzenia do którego bezpośrednio mają trafić przesyłane dane. Jeżeli serwis, z którym chcemy się połączyć znajduje się w tej samej sieci co nadawca, to jako adres docelowy wykorzystywany jest adres MAC karty sieciowej urządzenia docelowego. W przeciwnym przypadku jako adres docelowy używany jest adres MAC interfejsu sieciowego bramy. Przedstawiony proces nazywany opakowywaniem danych realizowany jest każdorazowo przez urządzenie komunikujące się z serwisami WWW. Oczywiście po otrzymaniu ramek przez hosta następuje proces odwrotny mający na celu uzyskanie kompletu danych, które zostaną zinterpretowane przez aplikację WWW (odpowiednio przeglądarkę lub serwer).



Rysunek 12 Opakowanie danych w stosie TCP/IP

Przedstawiony tutaj opis został bardzo mocno uproszczony i ograniczony do protokołu HTTP(S) , aby wskazać tylko podstawowe zależności między protokołami bez szczegółowego opisywania wszystkich ich funkcjonalności. Podejście takie pozwala na uświadomienie użytkownikom stopnia złożoności technologii wykorzystywanych w sieciach komputerowych.

8. Bezpieczeństwo w sieci

Jak widać z poprzednich rozdziałów złożoność wszystkich procesów realizowanych w sieciach komputerowych utrudnia zapewnienie bezpieczeństwa użytkownikom, którzy z nich korzystają. Jednocześnie trudno wymagać od każdego z użytkowników pełnej znajomości wszystkich aspektów działania sieci komputerowych. W związku z powyższym naturalną konsekwencją jest stworzenie dokumentów opisujących dozwolone i niedozwolone aspekty korzystania z komputerów i sieci komputerowych oraz związane z nimi procedury bezpieczeństwa. Podstawowym takim dokumentem, który powinien zostać przygotowany dla każdej sieci jest „polityka bezpie-

czeństwa". Podstawowe zasady dotyczące tworzenia takich specyfikacji zostały opisany w dokumentach RFC 2196 oraz RFC 3586.[12,13] Ponadto istnieje wiele firm, które w sposób profesjonalny zajmują się przygotowaniem tego typu dokumentacji w oparciu o przeprowadzone wcześniej badania.

Zapewnienie bezpieczeństwa sieci wymaga stałego jej monitoringu oraz reagowania na pojawiające się zdarzenia. Jak widać w przypadku Internetu nie można myśleć tutaj o lokalnym, ale globalnym podejściu. Ataki dotyczące sieci komputerowych mogą pochodzić z różnych lokalizacji geograficznych, a zatem ograniczenie nadzoru tylko i wyłącznie do konkretnych sieci jest niewystarczające. Starając się zapewnić bezpieczeństwo sieciom lokalnym administratorzy najczęściej stosują systemy zabezpieczeń na granicy sieci, konfigurując je na tak zwanych routerach brzegowych. Niestety jak pokazują statystyki około 80% ataków (w tym również działanie złośliwego oprogramowania) ma swoje źródło w sieci wewnętrznej. W związku z powyższym cały czas oprócz systemu zainstalowanego na granicy sieci wdraża się rozwiązania mające na celu śledzenie zachowań w sieci wewnętrznej, które często połączone są w jeden system z oprogramowaniem zabezpieczającym zainstalowanym na stacjach klienckich.

Coraz liczniej pojawiające się problemy związane z zapewnieniem bezpieczeństwa w Internecie spowodowały powołanie do życia organizacji zajmujących się analizą potencjalnych oraz występujących rzeczywistych zagrożeń w sieci. Publikowane przez nie informacje stanowią bardzo cenną pomoc dla osób dbających o bezpieczeństwo użytkowników. Do głównych organizacji tego typu należą organizacje CERT (*ang. Computer Emergency Response Team*). W Polsce od 1996 roku działa CERT Polska (www.cert.pl), która od roku 1997 jest członkiem FIRST (*ang. Forum of Incidents Response and Security Teams*). Organizacja CERT Polska finansowana jest przez NASK (Rysunek 13).

Rysunek 13 Witryna CERT POLSKA (www.cert.pl)

Ponadto w 2008 roku powołano również Rządowy Zespół Reagowania na Incydenty Komputerowe (cert.gov.pl). Jak można przeczytać na stronie tej organizacji „jej podstawowym zadaniem jest zapewnianie i rozwijanie zdolności jednostek organizacyjnych administracji publicznej Rzeczypospolitej Polskiej do ochrony przed cyberzagrożeniami, ze szczególnym uwzględnieniem ataków ukierunkowanych na infrastrukturę obejmującą systemy i sieci teleinformatyczne, których zniszczenie lub zakłócenie może stanowić zagrożenie dla życia, zdrowia ludzi, dziedzictwa narodowego oraz środowiska w znacznych rozmiarach, albo spowodować poważne straty materialne, a także zakłócić funkcjonowanie państwa". Zespół ten funkcjonuje w ramach Departamentu Bezpieczeństwa Teleinformatycznego ABW (Rysunek 14).

Rysunek 14 Witryna zespołu CERT.GOV.PL

9. Hakerzy i akty prawne w Polsce

Łatwość dostępu do informacji, poczucie anonimowości w sieci oraz szeroko dostępne oprogramowanie umożliwiające w prosty sposób realizowanie różnego rodzaju typowych ataków powoduje, znaczący wzrost zagrożenia dla osób korzystających z Internetu. Jednocześnie zwłaszcza wśród młodych ludzi istnieje przeświadczenie, iż z punktu widzenia prawodawstwo polskiego działania te nie podlegają karze. Pogląd ten jest błędny. Głównym aktem prawnym regulującym te aspekty jest Kodeks Karny, a ściślej mówiąc artykuły: 165, 267, 268, 269, 278, 291, 292 i 293. Aby zachęcić czytelnika do ich analizy, warto w tym miejscu zacytować Art. 269b

§ 1. Kto wytwarza, pozyskuje, zbywa lub udostępnia innym osobom urządzenia lub programy komputerowe przystosowane do popełnienia przestępstwa określonego w art. 165 § 1 pkt 4, art. 267 § 2, art. 268a § 1 albo § 2 w związku z § 1, art. 269 § 2 albo art. 269a, a także hasła komputerowe, kody dostępu lub inne dane umożliwiające dostęp do informacji przechowywanych w systemie komputerowym lub sieci teleinformatycznej, podlega karze pozbawienia wolności do lat 3.

Jak widać w sposób jasny wskazuje on, iż karze podlega nie tylko osoba realizująca konkretny atak, ale również osoba, która udostępnia oprogramowanie do jego przeprowadzenia.

Celem niniejszego opracowania nie było przedstawienie wszystkich zagrożeń związanych z Internetem, co z góry byłoby skazane na niepowodzenie, a jedynie miało ono posłużyć jako wstęp do dalszych rozważań oraz zachęcić użytkowników do głębszej, świadomej analizy zjawisk, które realizowane są w globalnej sieci, z której na co dzień korzystamy.

Literatura

1. "Roads and Crossroads of Internet History" by Gregory Gromov. 1995, http://www.netvalley.com/cgi-bin/intval/net_history.pl?chapter=1
2. M. Piwiński, „Uczniowie i komputery w sieci...", „Komputer w Szkole.", nr 5, (2003), s 38
3. http://www.sharewareconnection.com/network-protocols-map-poster.htm
4. http://pl.wikipedia.org/wiki/World_Wide_Web
5. http://www.phonefactor.com/sslgap/ssl-tls-authentication-patches
6. http://extendedsubset.com/Renegotiating_TLS.pdf
7. http://tools.ietf.org/html/rfc6176
8. M. Piwiński, „Praktyczna analiza sieci komputerowych z wykorzystaniem programu Wireshark", „Informatyka w Edukacji, V", A.B. Kwiatkowska, M. Sysło, (2008) 277 285
9. http://tools.ietf.org/html/rfc5246
10. http://tools.ietf.org/html/rfc791

11. http://www.iana.org/

12. http://www.ietf.org/rfc/rfc2196.txt

13. http://www.ietf.org/rfc/rfc3586.txt