SPOŁECZNA WYŻSZA SZKOŁA
PRZEDSIĘBIORCZOŚCI I ZARZĄDZANIA W ŁODZI

KIERUNEK STUDIÓW: INFORMATYKA

MARIUSZ JAMA
Numer albumu: 18425

UKRYWANIE DANYCH/ STEGANOGRAFIA

Praca dyplomowa/inżynierska napisana
w Instytucie Technologii Informatycznych
pod kierunkiem
dr inż. Piotr Goetzen

Łódź 2011

Wstęp.

W ostatnich latach rozwój globalnej sieci jest bardzo intensywny, przesyłanych jest mnóstwo danych i informacji, które nie zawsze są bezpieczne. Wiele firm opracowujących jakąś nową technologię komunikacyjną pracuje równocześnie nad zapewnieniem jej maksymalnego bezpieczeństwa. Tworzy więc dodatkowe mechanizmy dzięki którym możliwa będzie kontrola przesyłanych danych. Wielu użytkowników chcąc chronić swoje dane używa metod steganograficznych w celu ich ukrycia. Dane są przesyłane poprzez dodatkowy kanał, który umieszczony jest w oficjalnym i ogólnie dostępnym strumieniu danych. Współcześnie możliwe jest wykorzystanie wielu metod steganograficznych, które mogą zostać wykorzystane w różnych mediach przesyłu danych takich jak: dźwięk, obraz, video czy też protokół TCP/IP. Takie ukrywanie dodatkowych informacji może być wykorzystywane również przez osoby niepowołane, np: w złośliwym oprogramowaniu, więc należy być ostrożnym przy korzystaniu z technik steganograficznych. W pracy tej skupimy się na zagadnieniu steganografii w protokole sieciowym TCP/IP.

Niniejsza praca ma przedstawić metody i techniki ukrywania informacji w protokole TCP/IP, jednak aby to pokazać musimy poznać kilka niezbędnych informacji z dziedziny sieci komputerowych. W rozdziale I opisano potrzebne nam zagadnienia sieciowe czyli opis modelu warstwowego OSI z wytłumaczeniem wszystkich poszczególnych warstw, zasad jego funkcjonowania. W kolejnym podpunkcie tego rozdziału przedstawiono opis interesującego nas najbardziej protokołu TCP/IP, zasadę jego funkcjonowania, adresowania. Wytłumaczenie zasad funkcjonowania tego protokołu rozbite zostało na poszczególne protokoły składające się na cały ten protokół czyli osobno opisano protokół IP i TCP. W podpunkcie 2 przedstawione zostały szczegółowe informacje dotyczące nagłówków powyżej wymienionych protokół, informacje te potrzebne będą nam przy korzystaniu z metod steganograficznych. Kolejny 3 punkt rozdziału I przedstawia zależności wynikające pomiędzy średniowarstwowym modelem OSI, a modelem uproszczonym TCP/IP (czterowarstwowym). W rozdziale II opisano w następującej kolejności podstawowe dane dotyczące szyfrowania i kryptografii, opis klucza symetrycznego i asymetrycznego. Zasadę funkcjonowania tych kluczy oraz najbardziej popularne algorytmy szyfrujące, wszystko zostało zobrazowane odpowiednimi rysunkami. Rozdział ten ma nam przybliżyć wszelkie aspekty przekazywania tajnych informacji. Kolejny podpunkt tego rozdziału przedstawia krótką historię powstania steganografii wraz z opisem, zaczynając od czasów starożytnych, aż do współczesnych. Dodatkowo opisane zostały grupy technik zabezpieczających wykorzystywanych w steganografii. Ostatnia część tego podpunktu tłumaczy zasadę funkcjonowania komunikacji

steganograficznej pokazanej na podstawowym przykładzie, zobrazowanym rysunkiem. Kolejny podpunkt przedstawia kilka kontenerów wykorzystywanych przy stosowaniu technik steganograficznych. Tymi kontenerami są tekst, obraz, dźwięk i pliki wideo. Ostatni z podpunktów podaje definicję stegoanalizy. Rozdział III przedstawia analizę nagłówków protokołów IP i TCP, analiza pokazuje miejsca, które mogą zostać użyte w ukrywaniu kanałów informacyjnych. Kolejny z podpunktów opisuje ukryte kanały informacyjne, ich tworzenie i możliwości. Na samym końcu przedstawione zostaną wnioski i przemyślenia dotyczące steganografii w protokole TCP/IP.

Rozdział I

Zagadnienia sieciowe.

1.1 Model warstwowy OSI.

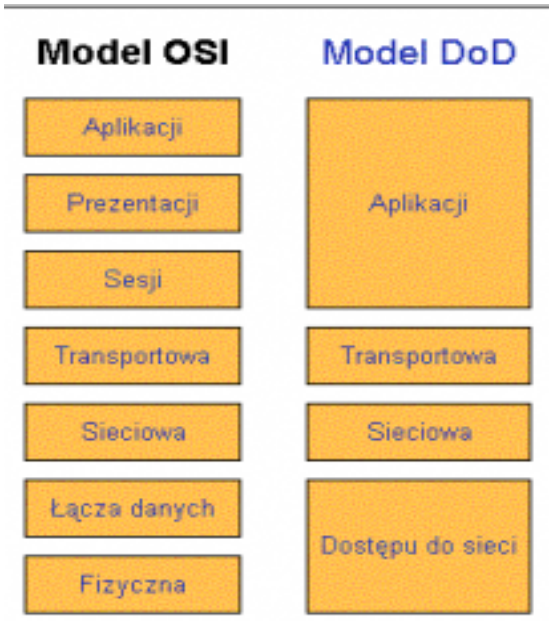
Aby ułatwić prace nad projektowaniem sieci komputerowych oraz oprogramowaniem systemów sieciowych stworzono architekturę sieci komputerowych. Architektura ta miała budowę warstwową. Wyodrębniono w niej siedem poziomów ustawionych hierarchicznie. Każdej warstwie przypisano własne usługi i funkcje oraz protokoły komunikacyjne i jednostki danych przez nie używane. Do 1978 roku nie istniał żaden ogólnoświatowy standard, który by definiował ilość warstw w architekturze sieciowej oraz jakie zadania i usługi miały one udostępniać. Wiele firm wcześniej opracowało swoje zamknięte architektury, możemy do nich zaliczyć m.in.:

a) architektura SNA (*ang. System Network Architecture*) opracowaną przez firmę IBM. Architektura ta umożliwia łączenie pomiędzy produktami tej firmy. Chodzi tu głównie o duże średnie systemy komputerowe czyli tzw. mainframe i midrange. W tej architekturze określono zbiór protokołów oraz zdefiniowano sposoby komunikacji pomiędzy maszynami.
b) architektura DNA (*ang. Digital Network Architecture*) opracowaną w firmie DEC. W ramach tej architektury zdefiniowany został zbiór protokołów, formy i sposoby wymiany komunikatów sterujących w sieci. Model ten posiada strukturę średniowarstwową.

Wiele powstających rozwiązań wprowadzało chaos w funkcjonowaniu różnorodnych systemów sieciowych w związku z tym w roku 1978 zdecydowano o opracowaniu w ramach międzynarodowej organizacji standaryzacyjnej ISO jednolitego modelu, który umożliwiałby komunikację we wszystkich systemach komputerowych. Model OSI został przedstawiony w formie normy ISO 7498.

OSI (*ang. Open System Interconnection*) lub Model OSI jest zdefiniowanym standardem, który opisuje strukturę komunikacyjną w sieci. Został stworzony przez organizacje ISO i ITU-T. Model OSI jest odniesieniem dla protokołów komunikacyjnych jako standard powiązań. Najpopularniejszym typem jest model OSI-RM (*ang. OSI Reference*

Model). Do podstawowych zadań modelu należy podział systemów sieciowych na siedem warstw (*ang. Layers*), które są od siebie niezależne. W internecie funkcjonuje uproszczony Model DoD, który posiada tylko cztery warstwy[1]. Model OSI-RM i model DoD możemy zobrazować przy pomocy poniższego rysunku.



Rysunek 1. Prezentacja graficzna modeli OSI i DoD.

(źródło: http://www.sieci.komputerowe.krakow.pl/files/osi_m.gif)

W modelu Osi zdefiniowane mamy jakie zadania i jakie informacje przekazywane są pomiędzy poszczególnymi warstwami. Wyróżnić możemy trzy warstwy, które są na samej górze. Warstwami tymi są warstwa aplikacji, prezentacji i sesji. Ich główną funkcją jest praca z oprogramowaniem które realizuje zadania zlecone przez użytkowników systemu komputerowego. Te trzy warstwy budują pewien typ interfejsu za pomocą którego możliwa jest komunikacja z warstwami niższymi. Te same warstwy realizują dokładnie odwrotne zadanie zależne od kierunku przepływu informacji. Komunikacja pomiędzy komputerami realizowana jest na poziomie równoważnych sobie warstw, Dla każdej warstwy powinien zostać stworzony jednoznaczny protokół komunikacyjny. Przy rzeczywistej pracy sieci komputerowej komunikacja działa tylko i wyłącznie w warstwie fizycznej. Informacje każdorazowo przekazywane są do sąsiedniej niższej warstwy, aż w końcu dotrą do warstwy fizycznej. Więc wywnioskować z tego można, że komunikacja „de facto" jest wirtualna, a wszystko zależne jest od warstwy fizycznej.

1 http://m6.mech.pk.edu.pl/~habel/dydaktyka/Kk/511e/t2/index.html

Warstwa aplikacji

Warstwa aplikacji jest najwyższą z warstw. Użytkownik, który chce skorzystać z oprogramowania i przesłać dane przez urządzenia sieciowe, to wykonywane to jest w warstwie aplikacji. W przypadku przesyłania informacji w górę, to właśnie warstwa aplikacji umożliwia użytkownikowi na odbiór tej informacji. Tabela przedstawiona poniżej pokazuje jakie protokoły są przypisane do tej warstwy.

Tabela 1. Protokoły w warstwie aplikacji.

PROTOKÓŁ WARSTWY APLIKACJI	
TELNET	Zdalne zarządzanie hostami
FTP	Protokół przesyłania plików
HTTP	Protokół przesyłania dokumentów tekstowych
DHCP	Protokół do dynamicznego przyznawania adresów
DNS	Protokół do przesyłu informacji o IP na podstawi nazwy hosta
SMTP, POP, IMAP	Protokoły do przesyłania wiadomości elektronicznych
SNMP	Protokół do zarządzania i monitorowania urządzeń sieciowych
NFS	Protokół umożliwiający dostęp do zasobów między hostami
DAP, LDAP	Protokół umożliwia do usług katalogowych X.500

Warstwa prezentacji

Podstawowym zadaniem warstwy prezentacyjnej jest przetwarzanie danych przekazywanych przez aplikacje do postaci kanonicznej (*ang. canonical representation*) czyli takiej, która jest zgodna ze specyfikacją OSI-RM. Dzięki takiemu działaniu do niższych warstw przekazywane są dane w jednolitym formacie. W przypadku odwrotnym (informacje przepływają do góry) warstwa ta przetwarza otrzymywane dane na format zgodny z wewnętrzną reprezentacją systemu docelowego. Odpowiedni format zależy od zastosowania konkretnego systemu komputerowego, które mogą być w dużym stopniu zróżnicowane systemów komputerowych oraz interpretacja otrzymywanych danych jest różna.

Warstwa sesji

Warstwa sesji przyjmuje od aplikacji dane i informacje, które muszą zostać zsynchronizowane. W warstwie sieci zaimplementowano informacje, które pozwalają wiedzą która aplikacja łączy się z którą i dzięki temu zapewniony jest prawidłowy kierunek przepływu informacji. We współczesnych systemach sieciowych możliwe jest jednoczesne działanie kilkudziesięciu aplikacji. Dlatego tak ważna jest wcześniej przestawiona synchronizacja pomiędzy przesyłem danych, inaczej nastąpiłoby ich wymieszanie. Przy przesyle danych w górę, ważna jest właściwa kolejność przesyłanych danych do warstwy prezentacyjnej. Warstwy umiejscowione na dole modelu OSI często doprowadzają do fragmentacji oraz przemieszania tych danych, które nie są wysyłane po kolei. Warstwa sesji działa jako mechanizm synchronizujący aplikacje pracujące w warstwie najwyższej na różnych maszynach. Warstwa sesji posiada pewien zbiór danych, które zostały nazwane punktem synchronizacji (*ang. synchronization point*) na ich podstawie warstwa sesji potrafi stwierdzić, czy jedna aplikacja dostarczyła już dane potrzebne dla drugiej. W tabeli 2 przedstawiono protokoły tej warstwy.

Tabela 2. Protokoły warstwy sesji.

PROTOKÓŁ WARSTWY SESJI	
RPC	Protokół zarządzający połączeniami pomiędzy aplikacjami z różnych hostów
ASP	Protokół do obsługi sesji Apple Talk
SQL	Protokół obsługujący zapytania języka SQL
NFS	Protokół obsługi sieciowego systemu plików

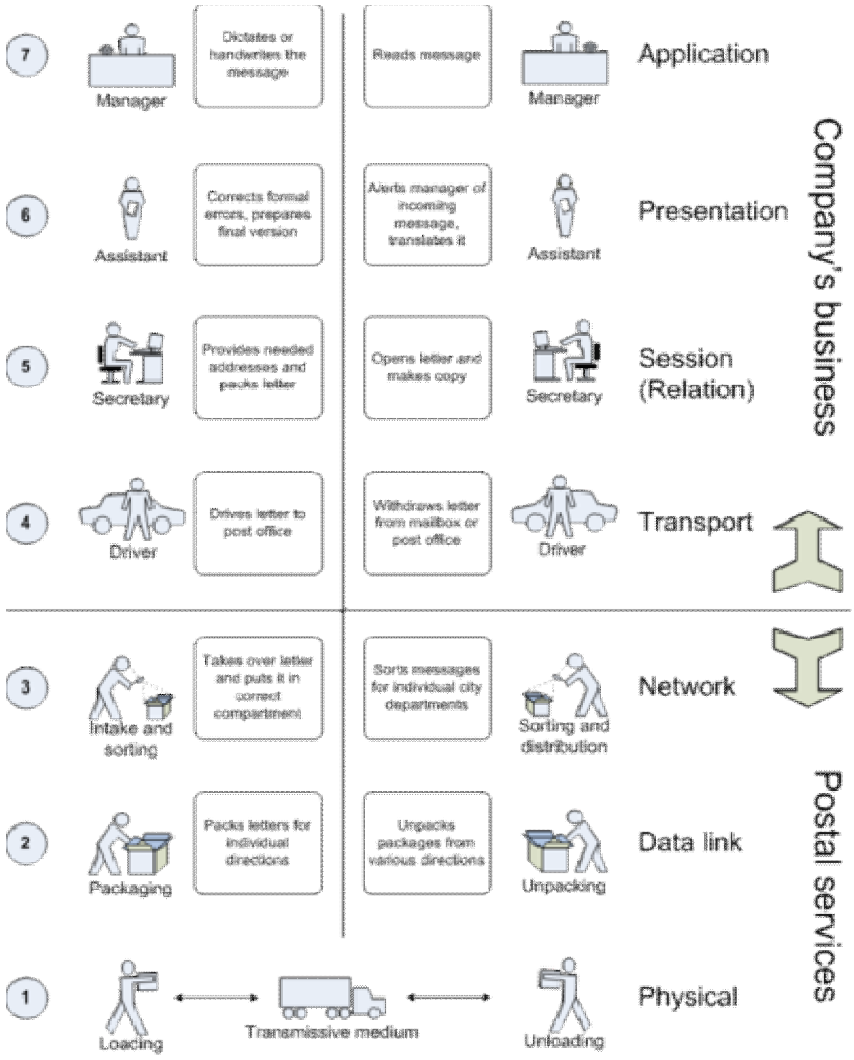
Warstwa transportowa

Zadaniem warstwy transportowej kontrola nad poprawnością przesyłanych danych. W warstwie paczka danych oznaczana jest jako TPDU (*ang. Transport Protocol Data Unit*). Dosyć często zdarza się, że informacje (dane) aby mogły zostać przesłane do niższych warstw muszą zostać podzielone na mniejsze części. W przypadku gdy za pierwszym razem nie da się przesłać poprawnie informacji, warstwa transportowa ponawia próbę, aż do momentu wyczerpania limitu przekazów. Ważnym zadaniem tej warstwy jest szeregowanie przesyłanych informacji według ustalonych priorytetów, a co z tego wynika przydzielane są

im określone wielkości pasma transmisyjnego. Wtedy kiedy niższe warstwy nie dają rady obsłużyć przesyłanych danych ostateczności, a kolejkowanie nie poprawia sytuacji warstwa transportowa zwraca do góry komunikaty o ich zapełnieniu i usuwa nadmiarowe dane. Warstwa transportowa rejestruje również komunikaty o przerwaniu połączenia i pozwala na bezpieczne zakończenie komunikacji. Na rysunku 2 pokazany jest system komunikacyjny na podstawie działania przykładowej firmy.

Tabela 3. Protokoły warstwy transportowej.

PROTOKÓŁ WARSTWY TRANSPORTOWEJ	
TCP	Strumieniowy protokół transportowy
UDP	Protokół do transportu pakietów bezpołączeniowy
SPX	Protokół transportowy środowiska NetWare



RM – OSI and letter communication parallel

Rysunek 2. Zobrazowanie komunikacji w warstwach OSI.

(źródło: http://m6.mech.pk.edu.pl/~habel/dydaktyka/Kk/511e/t2/400px-Rm-osi_parallel.png)

Warstwa sieciowa

Warstwa sieciowa jako jedyna z wszystkich istniejących warstw posiada informacje o fizycznej topologii sieci. Identyfikuje ona drogi łączące poszczególne komputery w sieci tzw. *Routing*. Drugim jej zadaniem jest podział przesyłanych informacji na poszczególne połączenia. W przypadku kiedy jest za dużo danych do przesłania, to warstwa sieciowa je pomija. To samo się dzieje wtedy gdy podczas transmisji wystąpią błędy. Standardowa paczka danych w tej warstwie oznaczana jest jako NPDU (*ang. Network Protocol Data Unit*). Jedynym zadaniem tej paczki jest zapewnienie sprawnej łączności między rożnymi odległymi punktami w sieci. Warstwa sieciowa przy przekazywaniu danych w dół umieszcza dane

wewnątrz pakietów, które są zrozumie przez warstw niższe jest to enkapsulacja. Podstawowe protokoły dla tej warstwy przedstawiono w tabeli 4.

Tabela 4. Protokoły warstwy sieciowej.

PROTOKÓŁ WARSTWY SIECIOWEJ	
IPv4	Protokół do transmisji pakietów w sieci
IPv6	Protokół do transmisji pakietów w sieci
IPX	Protokół do transmisji pakietów w sieci dla NetWare
ICMP	Protokół sprawdzania poprawnego przesyłu danych przez IP
ARP	Protokół wyznaczania adresów MAC na podstawie IP hosta

Warstwa łącza danych

Warstwa jest też nazywana warstwą liniową. Główną funkcją warstwy łącza danych jest nadzór nad jakością przekazywanych informacji. Nadzór ten dotyczy wyłącznie warstwy niższej. W warstwie łącza danych mamy możliwość zmian parametrów dotyczących pracy warstwy fizycznej. Zmiana parametrów pracy pozwala na obniżenie pojawiających się błędów podczas przekazu. Kolejnym zadaniem warstwy jest pakowanie danych w ramki i przekaz do warstwy fizycznej. W warstwie łącza danych zaimplementowano również możliwość rozpoznawana błędów związanych z niedostarczeniem pakietu lub uszkodzeniem ramki oraz ich naprawą. W przypadku przekazywania informacji do niższych warstw następuje enkapsulacja pakietów z warstwy sieciowej tak, aby wynikowa ramka była zgodna ze standardem. Są one oznaczane jako LPDU (*ang. data Link Protocol Data Unit*).

Tabela 5. Protokoły warstwy łącza danych.

PROTOKÓŁ WARSTWY ŁĄCZA DANYCH	
Ethernet	Protokół dostępu do medium transmisyjnego
FDDI	Protokół transmisyjny oparty na światłowodach
SLIP	Protokół używany przy połączeniach modemowych
PPP	Protokół typu Point-to-Point

X.25	Standard protokołu komunikacyjnego w sieciach WAN
Frame Relay	W przeszłości główny protokół transmisyjny
ATM	Asynchroniczny protokół przesyłu danych
HDLC	Protokół obsługujący połączenia dwu i więcej punktowe

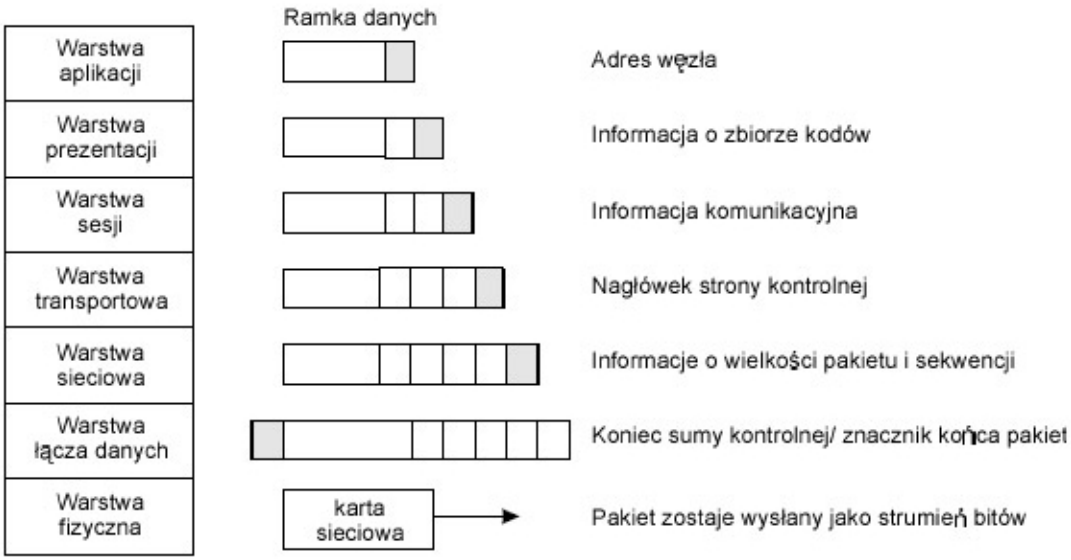
Warstwa fizyczna

Warstwa fizyczna ma postać konkretnego układu elektronicznego. Tworzy on kanał komunikacyjny związany ściśle z medium fizycznym przez które przesyłane są dane (kabel miedziany, światłowód, fale radiowe) oraz pozwalające na wymianę informacji pomiędzy urządzeniami sieciowymi. Warstwa fizyczna odbiera ramki od warstwy łącza danych i przesyła je do nośnika (i na odwrót), którego łącze stanowi jej granicę. Warstwa fizyczna ma w posiadaniu wyłącznie informacje o właściwościach fizycznych / optycznych bitów, które przesyła. Warstwa jest tak zbudowana, że znaczna większość przesyłanych danych dociera do odbiorców bez zniekształceń. Warstwa fizyczna może mieć własny system, który identyfikuje poszczególnych uczestników komunikacji. System ten jest niewidoczny dla warstw wyższych. Jako dodatkowe zastosowania wykorzystywane w warstwie fizycznej jest ochrona informacji przed zmianą lub podsłuchem przez niepowołane osoby.

1.2 Protokół TCP/IP

Protokół – jest to zbiór powiązań i połączeń elementów funkcjonalnych występujących w sieci komputerowej. Dzięki protokołom urządzenia sieciowe tworzące sieć mogą się komunikować. Do podstawowych zadań protokołu należy identyfikacja procesu, z którym chce się komunikować proces bazowy. Ze względu na ogromną ilość że komputerów pracujących w sieci koniecznością stało się opracowanie sposobu określania właściwego adresata, sposobu rozpoczynania i kończenia transmisji, a także sposobu przesyłania danych. Przekazywane dane mogą być dzielone, jednak określony protokół musi umieć odtworzyć dane w postaci początkowej. Drugą funkcją protokołu musi być wykrywanie i usuwanie błędów powstałych w wyniku niepoprawnego przesłania danych. Następnym zadaniem wykonywany przez protokół, a wynikającym z różnorodności urządzeń pracujących w sieci jest synchronizacja przesyłania danych poprzez zrealizowanie sprzężenia zwrotnego pomiędzy urządzeniami transmitującymi dane. Ponadto biorąc pod uwagę różne możliwości

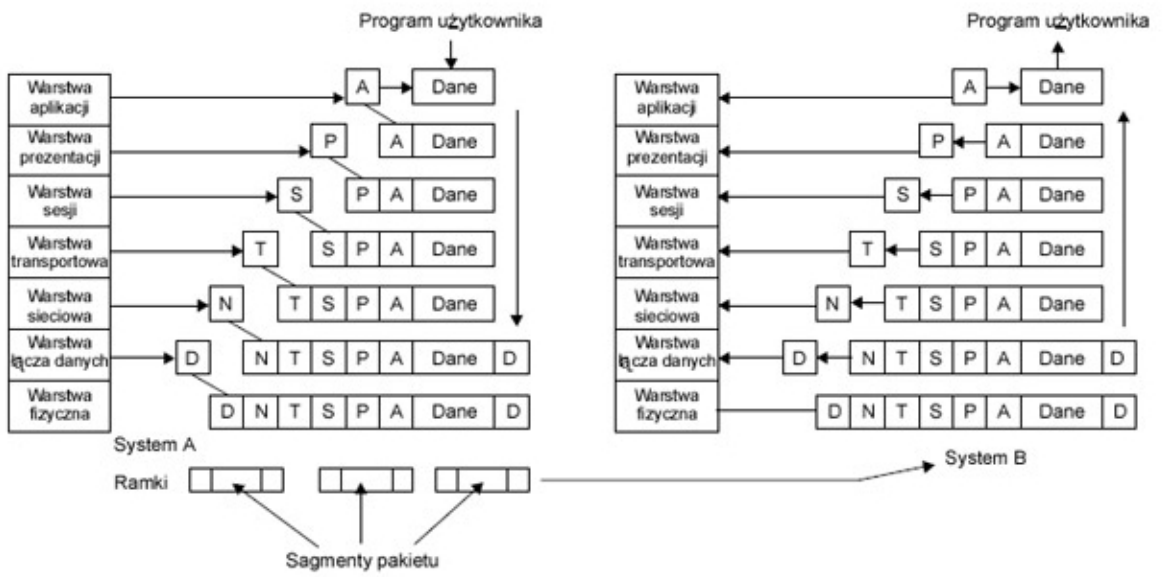
realizacji połączeń pomiędzy komputerami, protokół musi zapewniać wybór optymalnej drogi ukierunkowanej rodzajem transmisji. Na rysunku 3 przedstawiono drogę pakietów danych poprzez warstwy modelu OSI.



Rysunek 3. Droga pakietu przez warstwy modelu OSI.

(źródło: http://m6.mech.pk.edu.pl/~habel/dydaktyka/Kk/511e/t2/3.jpg)

Protokoły komunikacyjne definiowane są do konkretnej warstwowej architektury sieci. Każda z poszczególnych warstw określa swój protokół do obsługi funkcji lub podsystemu procesu komunikacyjnego. Najpopularniejszymi protokołami komunikacyjnymi są: model OSI opracowany przez ISO, SNA firmy IBM, AppleTalk firmy Apple, DECnet firmy DEC oraz protokoły Internetu np. TCP/IP. Wszystkie wymienione protokoły mają swoje odzwierciedlenie na każdym poziomie i spełniają zadania, które służą realizacji komunikacji pomiędzy systemami. Rysunek 4 pokazuje tworzenie, transmisję i odtwarzanie pakietów.



Rysunek 4. Tworzenie, transmisja i odtwarzanie pakietów.

(źródło: http://m6.mech.pk.edu.pl/~habel/dydaktyka/Kk/511e/t2/7.jpg)

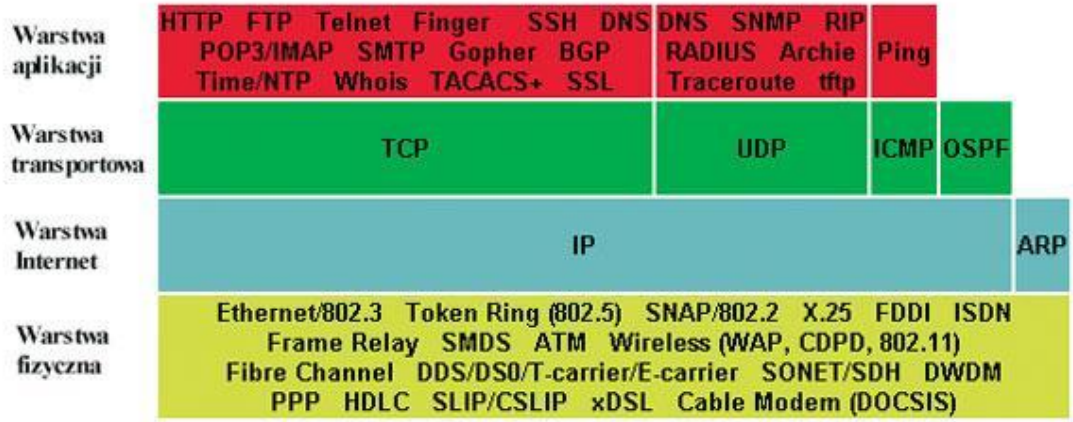
TCP/IP (*ang. Transmission Control Protocol/Internet Protocol*) jest to rodzina protokołów sieciowych, które umożliwiają łączenie komputerów, systemów operacyjnych i programów w jedną ogólnoświatową, uniwersalną sieć. Bez znaczenia jest tutaj jakie system operacyjny jest zainstalowany na danej maszynie. Protokół TCP/IP powstał w końcowych latach 60-tych ubiegłego wieku. Początkowo istniał jako projekt badawczy finansowany przez Departament Obrony Stanów Zjednoczonych. Głównym celem postawionym dla tego projektu było stworzenie takiej struktury sieciowej, która funkcjonowałaby bez zakłóceń mimo zniszczenia części jej fizycznej infrastruktury. W początkowych latach istnienia protokołu był on wykorzystywany jedynie w celach badawczych oraz wojskowych i naukowych. W latach 80-tych XX wieku TCP/IP stał się podstawowym protokołem stosowanym we współczesnych systemach operacyjnych. Jednak na powszechne wykorzystanie musiał poczekać, aż do rozwinięcia się internetu. W czasach obecnych TCP/IP łączy ze sobą ogromną ilość komputerów funkcjonujących w globalnej pajęczynie. Wszystkie one mają połączenie z siecią, która jest dostępna niemal dla każdego i z dowolnego punktu świata. Dobrym posunięciem przy tworzeniu TCP/IP było to, że posiada on otwartą architekturę czyli każdy może poznać sposób jego działania i może go wykorzystać w swoim oprogramowaniu, a żadna organizacja (firma) nie posiada do niego praw autorskich. Funkcjonowanie protokołów TCP/IP jest dosyć skomplikowane, jednak całą specyfikację tego protokołu, zasadę jego funkcjonowania można przeanalizować w dokumentach RFC (*ang. Requests for Comments*). Specyfikację RFC możemy pobrać ze stron *www.rfc-editor.org*.

Do zalet protokołu TCP/IP możemy zaliczyć:

- jest to standard przemysłowy czyli należy on do standardu otwartego, w związku z czym nie jest on kontrolowany przez żadną z firm,
- zawiera zestaw narzędzi, który pozwala na łączenie różnych systemów operacyjnych,
- zbudowany jest o architekturę klient-serwer, która jest skalowalna niezależna od platformy czyli może być poszerzana lub zwężana zależnie od wymagań sieci.

TCP/IP ma strukturę warstwową co pokazuje rysunek 5:

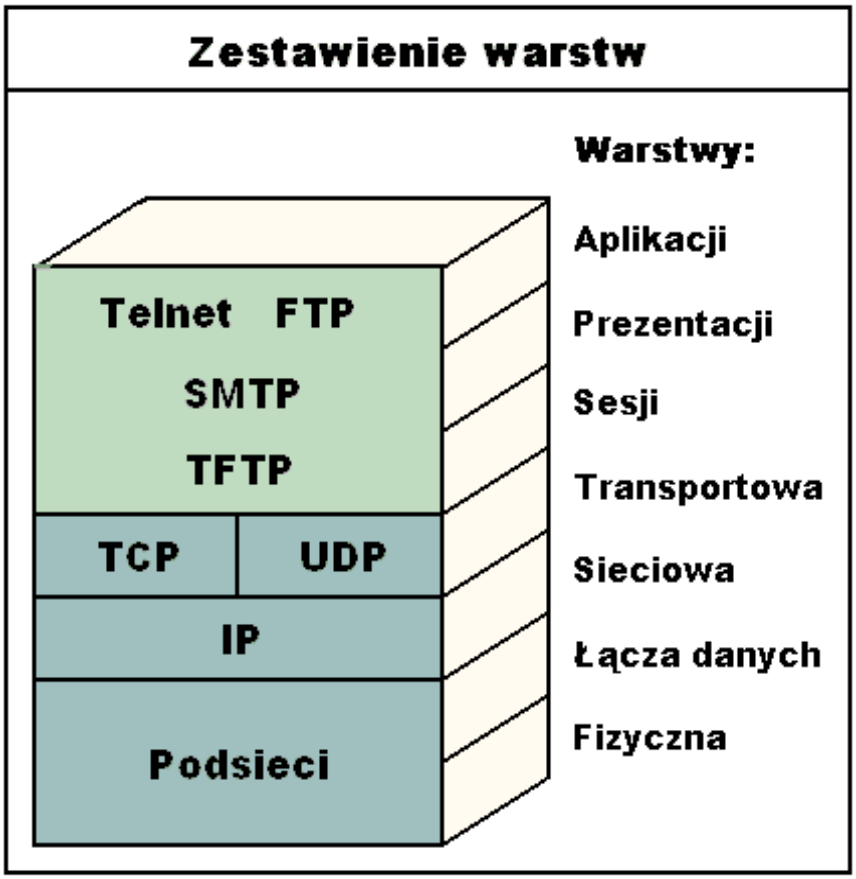
- warstwa aplikacji (programów użytkowych)
- warstwa transportowa
- warstwa sieci (warstwa intersieci, Internet)
- warstwa łącza (dostępu do sieci)



Rysunek 5. Struktura warstwowa protokołu TCP/IP.

(źródło: http://g1.pcworld.pl/news/thumb/7/4/74893)

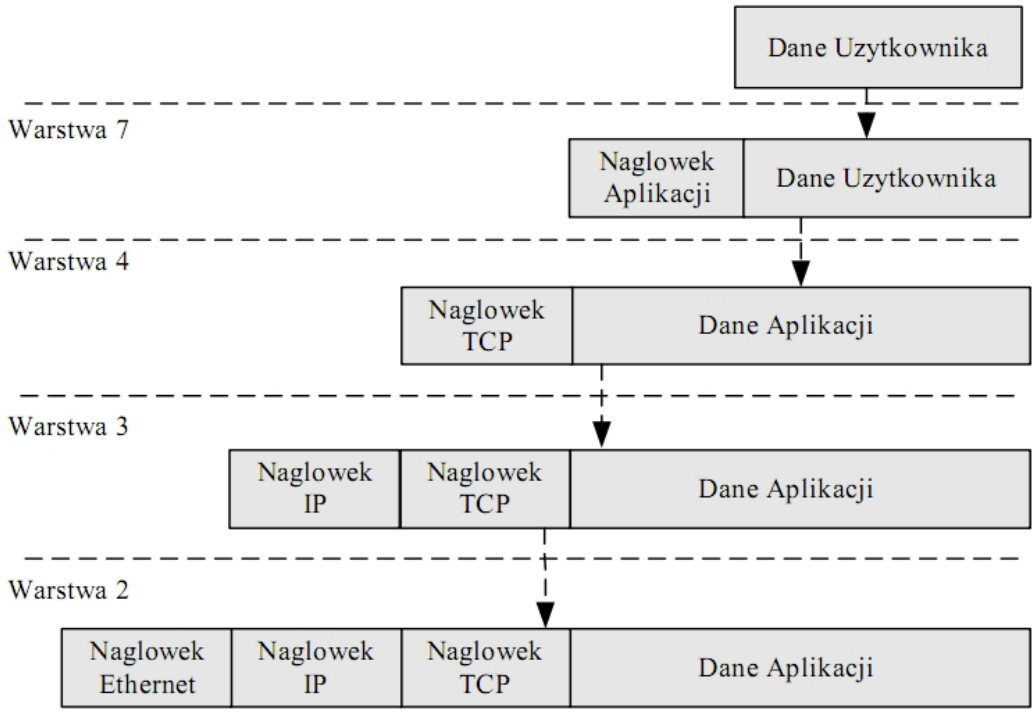
Protokół TCP/IP może zostać również opisany przy pomocy opisanego w punkcie 1.1 modelu ISO/OSI. Ze względu na duże skomplikowanie działania tego protokołu, TCP/IP opisywany jest przez czterowarstwowy model. W modelu tym najważniejszymi warstwami jest warstwa sieciowa i transportowa, kolejne warstwy są ze sobą połączone i tworzą dwie warstwy nazywane jako warstwa dostępu do sieci oraz warstwa aplikacji. Model uproszczony pełnie takie same funkcje co model siedmiowarstwowy możemy to zobaczyć na rysunku 6.



Rysunek 6. Porównanie modeli siedmiowarstwowego modelu OSI z modelem czterowarstwowym.

(źródło: http://majusek.fm.interia.pl/grafika/warstwy.gif)

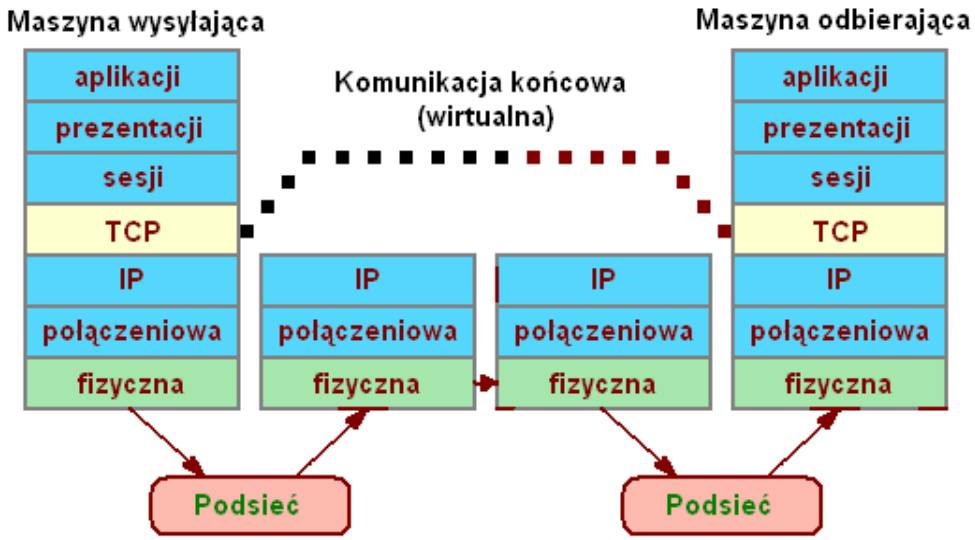
W protokole TCP/IP przy przesyłaniu informacji pomiędzy warstwami następuje tzw. kapsułkowanie (*ang. Encapsulation*) najprościej możemy go opisać jako dodawanie informacji kontrolnych w postaci nagłówków. Przy odbieraniu danych kapsułkowanie wykonywane jest w odwrotnej kolejności czyli przy przechodzeniu przez poszczególne warstwy usuwane są wcześniej dodane nagłówki, a następnie przesyłane do warstw leżących powyżej w celu dalszego przetwarzania. W przypadku kiedy przesyłane dane są niewielkie poprzez kapsułkowanie może powstać więcej danych wykorzystywanych przez protokół niż danych użytecznych. Aby nie dopuścić do takiej sytuacji używany jest inny protokół, np. UDP *(ang. User Datagram Protocol*). Przebieg kapsułkowania możemy przeanalizować na rysunku 7.



Rysunek 7. Kapsułkowanie w protokole TCP/IP w modelu OSI.

(źródło: Zasady komunikacji w sieciach komputerowych – Zbigniew Lipiński)

TCP (*ang. Transmission Control Protocol*) – jest to strumieniowy protokół komunikacji pomiędzy dwoma komputerami. Protokół ten jest częścią protokołu TCP/IP. Protokół TCP jest typem protokołu połączeniowego, który umożliwia detekcję i korekcję przesyłanych pakietów. TCP składa poszczególne pakiety w jedną całość i steruje ich ruchem. Protokół ten zapewnia wiarygodne połączenie z wyższymi warstwami komunikacyjnymi, wykonywane jest to przy pomocy sum kontrolnych i numerów sekwencyjnych pakietów. W przypadku kiedy pakiety zostaną zagubione wykonywana jest retransmisja. Charakterystycznym momentem w protokole TCP jest nawiązanie połączenia, nazywany three-way handshake. Host, który inicjuje połączenie wysyła pakiet z flagą SYN (synchronize), w dalszej kolejności host odbierający go wysyła pakiet z flagami SYN i ACK (potwierdzenie), wtedy to host wysyłający przesyła dane pakietów. Prawidłowe zakończenie przesyłu następuje po wysłaniu flagi FIN.



Rysunek 8. Transmisja pomiędzy warstwami.

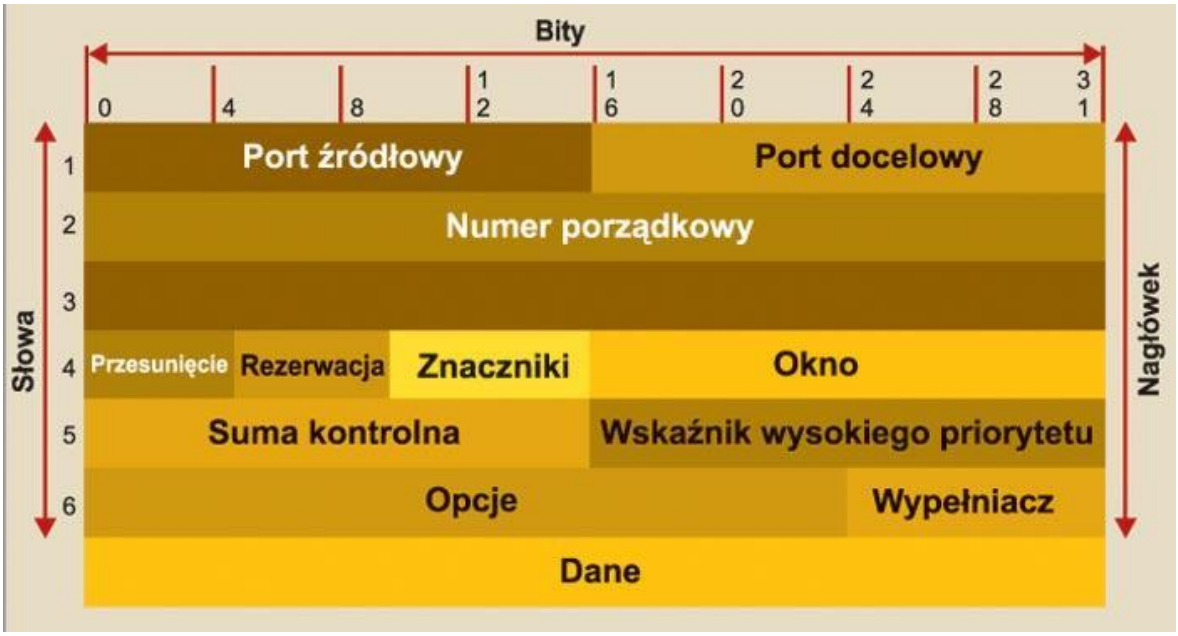
(źródło: http://majusek.fm.interia.pl/str18.htm)

Połączenia procesów warstwy aplikacji protokół TCP sprawdza przy pomocy numeru portu, który jest podany w postaci 16 bitowej liczbie całkowitej. Transport danych wykonywany przez proces odbywa się poprzez tzw. gniazdo. Adres źródłowy i numer portu zestawiony z adresem docelowymi i numerem portu nazywamy asocjacją. Strukturę całego nagłówka TCP przedstawia rysunek umieszczony poniżej.

Podczas połączenia TCP możemy wyróżnić kilka stanów gniazd:

- LISTEN stan ten oczekuje na nawiązanie połączenia TCP,
- SYN-SENT reprezentuje stan oczekiwania na pasujące połączenie po wysłaniu żądania,
- SYN-RECEIVED oczekuje na potwierdzenie ACK po wysłaniu SYN-SEN i SYN-RECEIVED,
- ESTABLISHED stan otwartego połączenia informującego o tym, że dane mogą być przesyłane do warstwy wyższej,
- FIN-WAIT-1 stan oczekujący na żądanie zakończenia połączenia od zdalnego protokołu lub potwierdzającego żądanie zakończenia przesłane wcześniej,
- FIN-WAIT-2 oczekuje na żądanie zakończenia od zdalnego protokołu,
- CLOSE-WAIT reprezentuje stan gniazda, które czeka na żądanie zakończenia przesłane przez lokalnego użytkownika,

- CLOSING stan oczekujący na potwierdzenie zakończenia połączenia przez zdalny protokół,
- LAST-ACK oczekuje na potwierdzenie żądania zakończenia wysłanego wcześniej,
- TIME-WAIT reprezentuje stan oczekiwania okresu czasu potrzebnego do upewnienia się, że zdalny protokół odebrał żądanie przerwania połączenia,
- CLOSED stan braku połączenia.



Rysunek 9. Struktura nagłówka TCP.

(źródło: http://g1.pcworld.pl/news/8/8/88293)

Opis poszczególnych pól nagłówka TCP.

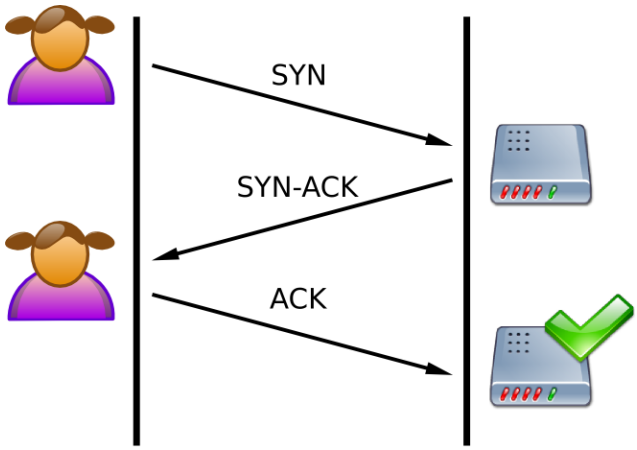
- Port źródłowy jest to numer portu z którego wysyłane są dane. Numer portu jest liczbą całkowitą z przedziału 1-65535.
- Port docelowy jest port do którego mają być dostarczone dane.
- Numer początkowy jest to numer bajtu w strumieniu przesyłanych danych, który identyfikuje pierwszy bajt danych w poszczególnych segmentach TCP. TCP numeruje każdy bajt właśnie takim numerem sekwencyjnym. Numer jest 32 bitową liczbą, która jest cyklicznie zaokrąglana do zera po osiągnięciu liczby 2^{23}-1.
- Numer potwierdzenia zawiera kolejny numer sekwencyjny bajtu, który jest oczekiwany przez nadawcę. Wartość ACK jest wyższy o 1 od ostatniego otrzymanego bajtu. Ponieważ TCP jest protokołem połączeniowym każda strona połączenia musi

przetrzymywać numery sekwencyjne bajtów dla przesyłanych danych.

- Przesunięcie ustawiana jest tu długość nagłówka TCP, liczona jest ona w słowach o długości 32 bitów.
- Rezerwacja wymagane jest aby bity (6) reprezentujące to pole miały wartość 0.
- Znaczniki w tym polu znajdują się flagi nagłówka TCP ma ono długość 6 bitów. Możemy rozróżnić 6 flag które sterują połączeniem:
  - URG kiedy ustawimy ten bit flagowy, wtedy pole urg_ptr jest ważne,
  - ACK ustawienie tego bitu włączy pole ack_seq,
  - PSH włączenie tego bitu skutkuje tym, że dane powinny być niezwłocznie przesłane,
  - RST ustawienie tego bitu spowoduje rozłączenie połączenia,
  - SYN bit ten synchronizuje numery sekwencyjne podczas nawiązywania połączenia,
  - FIN bit oznaczający zakończenie w przesyłaniu danych.
- Okno służy do kontroli przepływu między nadajnikiem a odbiornikiem. Oparta jest na ciągłej numeracji przesyłanych pakietów.
- Suma kontrolna pole to zawiera sumę kontrolną składaną z nagłówka TCP i pseudo-nagłówka o długości 96 bitów.
- Wskaźnik wysokiego priorytetu informuje o tym, że jakieś przesyłane segmenty TCP posiadają wysoki priorytet i oznaczane są znacznikiem URG.
- Opcje pole to definiuje długość opcji, jak i same opcje oraz rodzaj opcji.

Protokół TCP stosuje pewną metodę zwaną retransmisją, która ma zagwarantować, że przesyłane dane nie są tracona ani dublowane. Jednak musi być spełniony jeden podstawowy warunek, a mianowicie odbiorca i nadawca muszą się komunikować ze sobą wysyłając flagę potwierdzenia ACK, która informuje o otrzymaniu danych. Nadawca otrzymujące komunikat potwierdzenia zapisuje sobie tą informację tak aby wysyłać kolejne pakiety. Protokół przed rozpoczęciem transmisji wysyła pewną ilość informacji kontrolnych tzw. handshake. Handshake który jest wykorzystywany w TCP określany jest jako 3-way-handshake. Określenie to bierze się stąd, że podczas wysyłania informacji kontrolnych wymieniane są trzy bloki informacji. Nawiązanie połączenia zaczyna się w momencie kiedy dwa komputery ustalają wartość początkową sekwencji numerycznej zwany skrótowo ISN (*ang. Initial Sequence Number*). Dwa nawiązujące połączenie systemy TCP wymieniają i potwierdzają

wysłane wartości. Przedstawienie wizualne działania 3-way-handshake widoczny jest na rysunku 10.



Rysunek 10. Wizualizacja działania 3-way-handshake.

(źródło: http://upload.wikimedia.org/wikipedia/commons/8/8c/Tcp_normal.png)

IP (*ang. Internet Protocol*) – jest to protokół bezpołączeniowy, a co z tego wynika nie ma zaimplementowanych mechanizmów weryfikacyjnych, które sprawdzają poprawność przesyłanych pakietów. Głównym zadaniem IP jest transport pakietów do adresata. Działa przeważnie w sieciach komputerowych opartych na przełączaniu pakietów. IP przesyła bloki danych tzw. datagramy, a odbiorca identyfikowany jest przy pomocy adresów o stałej długości[2]. W chwili obecnej powszechnie używana jest wersja 4 protokołu IPv4, a adresy występują w postaci 32 bitowej liczby zapisywanej w czterech oktetach. Jednak ze względu na wyczerpujące się możliwości przydzielania nowych adresów opracowano nową wersję protokołu IPv6. W najnowszej wersji zwiększono rozmiar adresu z 32 bitów na 128 bitów. Adres w nowej wersji będzie reprezentowany w postaci heksadecymalnej (szesnastkowej). Dodatkowymi ulepszeniami nowej wersji protokołu IP są: wprowadzenie nowych rozszerzeń, auto konfiguracja, wyeliminowanie wad wersji v4. Protokół IPv6 w globalnej sieci wdrażany jest już od 2000 roku. Na rysunku 10 przedstawiono nagłówek protokołu IP.

2 Asseco Poland, „Specyfikacja protokołów komunikacyjnych", kwiecień 2008

Rysunek 11. Nagłówek protokołu IP.

(źródło: http://g1.pcworld.pl/news/thumb/8/8/88288)

Opis pół nagłówka protokołu IP.

- Wersja zawiera numer wersji protokołu IP.
- IHL (*ang. IP Header Lenght*) zawiera rozmiar nagłówka IP w słowach 32-bitowych.
- Rodzaj usługi (*ang. Type of* Service) zawiera w sobie abstrakcyjne wartości, które mają za zadania podniesienie jakości obsługi pakietu. Pole jest 1 bajtowe, pole to często jest ignorowane. Pole to ma postać pierwszeństwo-D-T-R-C-0. Pierwsze trzy bity określają pierwszeństwo i są ignorowane w większości istniejących współcześnie sieciach. Kolejne cztery bity określają typ usługi, w danym momencie może być ustawiony tylko jeden typ. Po pewnym czasie przedefiniowano to pole i teraz ma ono postać: DSCP-CU. W DSCP oznacza się kategorię usługi (6 najstarszych bitów), dwa ostatnie nie są wykorzystywane.
- Długość całkowita przedstawiona jest tu całkowita długość pakietu razem z nagłówkiem i danymi podana w bajtach. Pole to ma 16 bitów długości,a więc jego maksymalna wartość może wynieść 65535.
- Identyfikacja zawartość tego pola ustawiana jest przez nadawcę i potrzebne jest podczas scalania pakietu.
- Znaczniki są kontrolne 3 bity, które zawierają dane na temat fragmentacji danych. Występują następujące flagi:
  - RF (*ang. Reserved Flag*) w chwili obecnej nie jest wykorzystywana

  - DF (*ang. Don't Fragment*) flaga ta oznacza brak zgodny na fragmentację danych
  - MF (*ang. More Fragment*) ustawienie tej flagi oznacza, że datagram jest częściowo sfragmentowany, a po nim następują dalsze fragmenty
- Przesunięcie fragmentu jest to 13 bitowe pole zawierające informacje o miejscu położenia fragmentów danych w pakiecie. Dane są 8 bajtowe.
- Czas życia określone jest tu czas życia pakietu, wartość czasu zmniejszana o jeden przy przechodzeniu przez każdą stację, działanie takie ma wyeliminować pakiety, które są niemożliwe do dostarczenia.
- Protokół w polu tym określony jest protokół jaki został użyty do przetwarzania danych pakietu.
- Suma kontrolna nagłówka pole o długości 16 bitów, które zawiera sumę kontrolną nagłówka IP.
- Adres nadawcy jest to adres internetowy nadawcy pakietu.
- Adres odbiorcy zdefiniowano tu adres odbiorcy pakietu.
- Opcje zazwyczaj to pole nie jest używane, jednak jeśli trzeba może zawierać np: dane trasy jaką przebył pakiet.

Adres IP podzielony jest na cztery liczby dziesiętne oddzielone kropkami, które mają długość całkowitą 32 bitów. Wyróżnić możemy 5 klas adresowych, które możemy rozpoznać po bitach adresu. Liczba bitów identyfikujących sieci oraz liczba bitów, które identyfikują komputer jest zmienna, zależna od klasy adresowej. Najważniejszymi klasami adresów sieciowych są klasa A, klasa B i klasa C. Obszary adresowania pokazane są na rysunku 11.

Klasa	Obszar adresowy	Maksymalna liczba hostów	Obszar zastosowania
A	1.0.0.0 do 127.255.255.255	16 777 216	mało sieci, wiele hostów
B	128.0.0.0 do 191.255.255.255	65 536	zrównoważona liczba sieci i hostów
C	224.0.0.0 do 239.255.255.255	254	wiele sieci, mało hostów
D	224.0.0.0 do 239.255.255.255	—	adresy grupowe
E	240.0.0.0 do 254.255.255.255	—	niezdefiniowane

Rysunek 12. Obszary adresowe protokołu IP.

(źródło: http://g1.pcworld.pl/news/thumb/8/8/88290)

Przy korzystaniu z protokołu IP wykorzystujemy często z tzw. routing. Działanie routingu polega na znalezieniu takiej drogi dla przesyłanych danych, aby dotarł on do punktu docelowego tylko na podstawie adresu IP. Dane przechodząc przez poszczególne stacje kierowane są tam gdzie zdecyduje stacja przesyłowa. Wybór określonej trasy zależy od wielu

kryteriów. Wyróżnić możemy trzy typy routingu:

- statyczny czyli taki gdzie trasa wyznaczana jest na podstawie wpisów w tablicy routingu
- domyślny trasa wyznaczana jest na podstawie domyślnego jednego wpisu
- dynamiczny wpisy w tablicy routingu są aktualizowane automatycznie

Przy routingu statycznym do tablicy routingu komputera wpisany dedykowany router do każdej z sieci. Takie podejście w routingu pozwala nam na śledzenie drogi jaką przebywają pakiety z danymi. W bardzo rozbudowanych sieciach typ ten jest uciążliwy, ponieważ istniałoby bardzo dużo wpisów w takiej tablicy.

W routingu domyślnym do tablicy komputera wpisywany jest tylko jeden adres, pod który wysyłane są wszystkie pakiety danych niepochodzące z własnego obszaru adresowego sieci.

W przypadku routingu dynamicznego komputery i router współpracują ze sobą wymieniając między sobą informacje. Dzięki takiej współpracy komputer posiada informacje o najlepszej aktualnej trasie. Każdy z przesyłanych pakietów danych przechodzi najbardziej optymalną trasą. W komunikacji między tymi dwoma punktami odbywa się za pomocą specjalnego protokołu RIP (*ang. Routing Information Protocol*) lub IGRP (*ang. Interior Gateway Routing Protocol*).

Protokół TCP/IP uregulowany jest odpowiednimi normami i przepisami. Ostatnim dokumentem prawnym w Polsce określającym minimalne wymagania dla systemów teleinformatycznych jest Rozporządzenie Rady Ministrów z dnia 11 października 2005 (Dz.U.05.212.1766). Wymagania te określone są w załączniku nr 1.

Tabela 6 Załącznik nr 1 do wymagań dla protokołów komunikacyjnych.

ZAŁĄCZNIK Nr 1

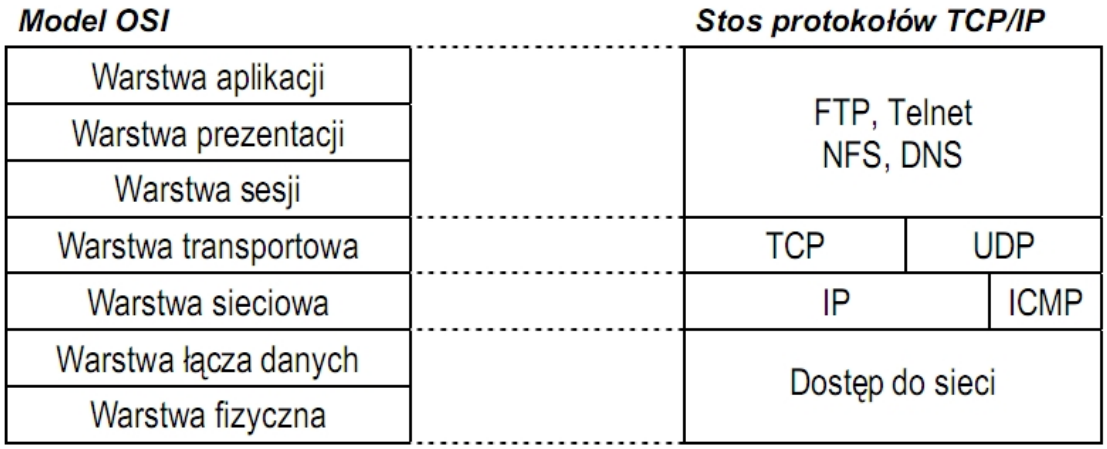
PROTOKOŁY KOMUNIKACYJNE I SZYFRUJĄCE UMOŻLIWIAJĄCE WYMIANĘ DANYCH Z INNYMI SYSTEMAMI TELEINFORMATYCZNYMI UŻYWANYMI DO REALIZACJI ZADAŃ PUBLICZNYCH

Lp.	Nazwa skrócona protokołu oraz jego wersja	Oryginalna pełna nazwa protokołu	Opis protokołu	Organizacja określająca normę lub standard	Nazwa normy, standardu lub dokumentu normalizacyjnego albo standaryzacyjnego
1	2	3	4	5	6
1.	**Do wymiany danych z systemami teleinformatycznymi stosuje się co najmniej jeden z następujących protokołów:**				
1.1	IP wersja 4	Internet Protocol	Protokół komunikacyjny dla Internetu	IETF	RFC 0791
1.2	TCP	Transmission Control Protocol	Strumieniowy protokół komunikacyjny	IETF	RFC 0793
1.3	UDP	User Datagram Protocol	Datagramowy protokół użytkownika	IETF	RFC 0768
1.4	ICMP	Internet Control Message Protocol	Protokół komunikatów kontrolnych Internetu	IETF	RFC 0792
1.5	HTTP wersja 1.1	Hypertext Transfer Protocol	Protokół komunikacyjny sieci WWW	IETF	RFC 2616
2.	**Do wymiany danych z systemami teleinformatycznymi prowadzonej w formie komunikacji pomiędzy klientem i serwerem poczty elektronicznej stosuje się co najmniej jeden z następujących protokołów:**				
2.1	SMTP/MIME	Simple Mail Transfer Protocol/ Multi-Purpose Internet Mail Extensions	Protokoły komunikacyjne wysyłania poczty elektronicznej	IETF	RFC 2045 RFC 2046 RFC 2047 RFC 2048 RFC 2049 RFC 2231 RFC 2646 RFC 2821 RFC 2822 RFC 3023
2.2	POP3	Post Office Protocol	Protokół odbioru wiadomości poczty elektronicznej	IETF	RFC 1939 RFC 1957 RFC 2449
2.3	IMAP	Internet Message Access Protocol	Protokół odbioru wiadomości poczty elektronicznej	IETF	RFC 2342 RFC 2971 RFC 3501 RFC 3502 RFC 3503
3.	**Do szyfrowania wymiany danych z systemami teleinformatycznymi stosuje się co najmniej jeden z następujących protokołów:**				
3.1	SSL wersja 3/TLS	Secure Sockets Layer / Transport Layer Security	Protokół szyfrujący dla sieci WWW	IETF	RFC 2246
3.2	S/MIME wersja 3	Secure Multi-Purpose Internet Mail Extensions	Protokół szyfrujący dla poczty elektronicznej	IETF	RFC 2631 RFC 2632 RFC 2633 RFC 3369
4.	**Do wymiany danych z systemami teleinformatycznymi w zakresie innych usług sieciowych stosuje się co najmniej jeden z następujących protokołów:**				
4.1	DNS	Domain Name System	Protokół komunikacyjny odpowiedzialny za odnajdywanie, informacji o adresach IP	IETF	RFC 1035
4.2	FTP	File Transfer Protocol	Protokół przesyłania plików	IETF	RFC 959
4.3	SOAP wersja 1.2	Simple Object Access Protocol	Protokół wywoływania zdalnego dostępu do obiektów	W3C	
4.4	WSDL wersja 1.1	Web Services Description Language	Język opisu usług sieciowych	W3C	

Rodzina protokołów TCP/IP oczywiści posiada pewne wady, wynikające głównie ze względów bezpieczeństwa. Wady te są szczególnie widoczne wtedy, gdy protokół TCP/IP wykorzystywany jest w sieci ethernetowej w której nie są używane przełączniki, co automatycznie naraża sieć na podsłuchanie lub podrobienie. Protokół ten posiada niski poziom bezpieczeństwa w przypadku wykorzystania takich metod ataku jak sniffing, spoofing, SYN-flood. Bardzo dużą wadą jest także ograniczony zakres adresowy protokołu IP w wersji 4. Jednak aby wyeliminować te ograniczenie opracowano nową wersję protokołu IP nazwaną IPv6.

1.3 Model ISO, a TCP/IP

Model OSI składający się z siedmiu warstw nie wykazuje jak mają być zorganizowane wszystkie usługi sieciowe, model ten daje jedynie potrzebne do organizacji sieci wskazówki. W przeważającej większości zastosowań stosuje się model warstwowy usług sieciowych, który ma odwzorowanie w 7-warstwowym modelu OSI. Tak samo możemy przyporządkować do podstawowego modelu OSI model sieciowy TCP/IP (co jest pokazane na rysunku 13), a więc możemy takie przyporządkowanie wyrazić przez uproszczony model odniesienia. Aplikacje sieciowe przeważnie zajmują się trzema warstwami, które znajdują się najwyżej. Są to warstwa sesji, prezentacji i aplikacji z siedmiowarstwowego modelu odniesienia OSI. Trzy wymienione wcześniej warstwy mogą zostać połączone w jedną zwaną warstwą aplikacyjną. Dwie ostatnie warstwy modelu OSI (fizyczna i łącza danych) także możemy złączyć w jedną warstwę. Wynikiem takich połączeń jest uproszczony model czterowarstwowy.



Rysunek 13. Przyporządkowanie TCP/IP do modelu OSI.

(źródło: http://www.staff.amu.edu.pl/~psi/informatyka/tcpip/osi_tcp.htm)

W uproszczonym modelu warstwowym warstwa dostępu do sieci nie ma określonych standardowych protokołów. Protokoły są przyporządkowywane zależnie od wymagań stawianych sieci oraz przeznaczenia. W sieci TCP/IP wykorzystać możemy różne protokoły dostępu do sieci. Zaliczyć do nich możemy między innymi Token Ring, FDDI, X.25, Ethernet. Do warstwy sieciowej przyporządkowane są protokoły IP i ICMP z modelu TCP/IP . Protokół IP, który funkcjonuje w modelu TCP/IP transportując datagramy w sieci. Taki datagram składa się z danych przekazywanych przez warstwę aplikacyjną oraz nagłówka i bloku końcowego dodanego w warstwie transportowej. IP został opracowany głównie, aby umożliwić sterowanie urządzeniami sieciowymi (routerami). Kolejnymi protokołami, które pracują w warstwie sieciowej są m.in.: ICMP (służy do korygowania błędów), ARP (protokół zwraca adres fizyczny węzła o znanym adresie IP). Do warstwy transportowej przypisane są dwa protokoły służące do transportu TCP i UDP.

- TCP jest protokołem połączeniowym, prowadzącym automatyczne retransmisje w przypadku wystąpienia błędów transmisji. TCP steruje danymi otrzymanymi z warstwy aplikacyjnej.
- UDP jest protokołem bezpołączeniowym, który jednak nie sprawdza poprawności przesyłanych danych i nie przeprowadzający retransmisji. Dlatego protokół ten współpracuje z aplikacjami, które posiadają własne mechanizmy weryfikacyjne.

Warstwa aplikacyjna umieszczona jest najwyżej w hierarchii warstw zarówno w modelu OSI, jak i modelu sieciowym TCP/IP. Architektura protokołu TCP/IP w warstwie aplikacyjnej jest niezależna od platformy sprzętowej i działa jako połączenie klient - serwer. Wygląda to następująco: klient inicjuje (wywołuje) aplikacje, a serwer odpowiada na żądanie klienta. Realizacją przedstawionej koncepcji są podstawowe aplikacje TCP/IP do których zaliczyć możemy TELNET, FTP oraz SMTP. Innymi protokołami wykorzystywanymi w warstwie aplikacji są: protokół zabezpieczający nazwany Kerberos, protokół zarządzania siecią SNMP, prosty protokół przesyłania plików TFTP oraz system adresowania DNS.

Rozdział II

Szyfrowanie i steganografia.

2.1 Szyfrowanie.

Kryptografia oznacza w oryginale tajne pismo, słowo jest złożeniem dwóch słów języka greckiego: krypto - ukryty, tajny, graph - pismo, słowo. Kryptografia zajmuje się szyfrowaniem (kodowaniem) czyli taką zmianą tekstu z postaci jawnej i czytelnej (tekst otwarty) na zupełnie niezrozumiałą (tekst szyfrowany). Zaszyfrowanie tekstu odbywa się tak, aby uprawniony użytkownik mógł odwrócić zastosowaną transformację w procesie odszyfrowywania (dekodowania) czyli odczytać tekst. Z tego opisu wynika, że głównym celem szyfrowania nie jest ukrywanie tekstu, a jedynie jego treści. Za ukrywanie informacji odpowiedzialny jest inny dział kryptografii, tzw. steganografia (steganos oznacza w języku greckim ukryty)[3]. Do dodatkowych zadań kryptografii należą: kontrola integralności, czy uwierzytelnianie.

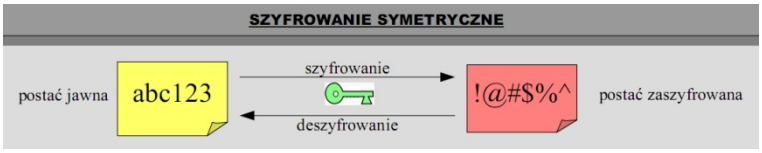
W celu zapewnienia wysokiego stopnia bezpieczeństwa przesyłanych informacji, konieczne jest wprowadzenie odpowiednich metod mogących je zagwarantować bezpieczeństwo przesyłanym danym. Odpowiednie połączenie mechanizmów kryptograficznych i ich wykorzystanie pozwala na tworzenie systemów oferujących cały wachlarz usług zwiększających bezpieczeństwo przesyłanych informacji. Jeszcze nie tak dawno temu kryptografia była wykorzystywana jedynie przez wojsko i wszelkiego rodzaju służby. Osoby prywatne i podmioty gospodarcze kryptografią zainteresowały się znacznie później. Szersze zainteresowanie spowodowane było gwałtownym rozwojem sieci komputerowych, a co za tym idzie szyfrowanie stało się jednym z najważniejszych elementów globalnej sieci.

W kryptografia rozróżniamy dwie podstawowe techniki szyfrowania. Transpozycja czyli przesunięcie polega ona na zmianie nie same znaki, ale ich kolejności ich występowania. Drugą stosowaną techniką szyfrowania jest substytucja, czyli podstawienie, polega ona na zastępowaniu poszczególnych znaków tekstu innymi znakami.

Szyfrowanie możemy podzielić na rodzaje:

- symetryczne w tym rodzaju szyfrowania wykorzystywany jest tylko jeden klucz, zarówno do szyfrowania jak i deszyfrowania,

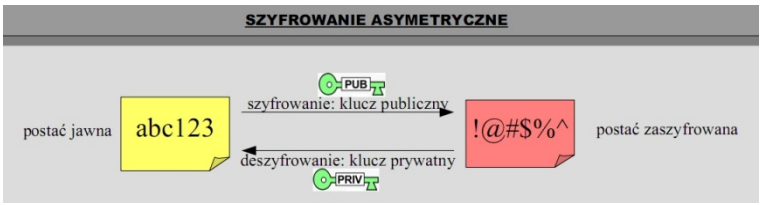
3 http://www.pcworld.pl/artykuly/35081_0_1/Przeglad.zagadnien.kryptografii.html



Rysunek 14. Szyfrowanie symetryczne.

(źródło: Wykład 3 Bezpieczeństwo przesyłu informacji. Szyfrowanie)

– asymetryczne działa przy wykorzystaniu dwóch kluczy, tzn. klucza prywatnego i klucza publicznego, wykorzystywane klucze są różne



Rysunek 15. Szyfrowanie asymetryczne.

(źródło: Wykład 3 Bezpieczeństwo przesyłu informacji. Szyfrowanie)

Zagrożeniem dla procesu bezpiecznej wymiany danych przy zastosowaniu szyfrowania asymetrycznego jest zastąpienia klucza publicznego jednej z strony kluczem publicznym atakującego. Druga strona jest wtedy przekonana, że korzysta z prawidłowego klucza publicznego odbiorcy i wtedy zaszyfruje ją niewłaściwym kluczem, przez co umożliwi odczytanie danych intruzowi. Można temu zapobiec, należy stworzyć mechanizm certyfikacji klucza, który miałby za zadanie potwierdzanie autentyczności używanego klucza i prawo do korzystania z niego przez daną osobę, firmę czy instytucję.

W kryptografii możemy wyróżnić wiele algorytmów szyfrujących, są to m.in.: DES, Twofish, RSA, AES, RC5, Serpent, IDEA. W tej pracy przedstawię tylko te najpopularniejsze.

Algorytm RSA (nazwa powstała od pierwszych liter nazwisk twórców - Rivest, Shamir i Adelman) – był pierwszym stworzonym algorytmem asymetrycznym, obecnie jest jednym z najczęściej wykorzystywanych algorytmów szyfrujących. Algorytm RSA opiera się na trudności faktoryzacji (rozkładu na czynniki) dużych liczb. Stworzenie szybkiej metody faktoryzacji doprowadziłoby do złamania RSA. Algorytm RSA działa w trzech fazach.

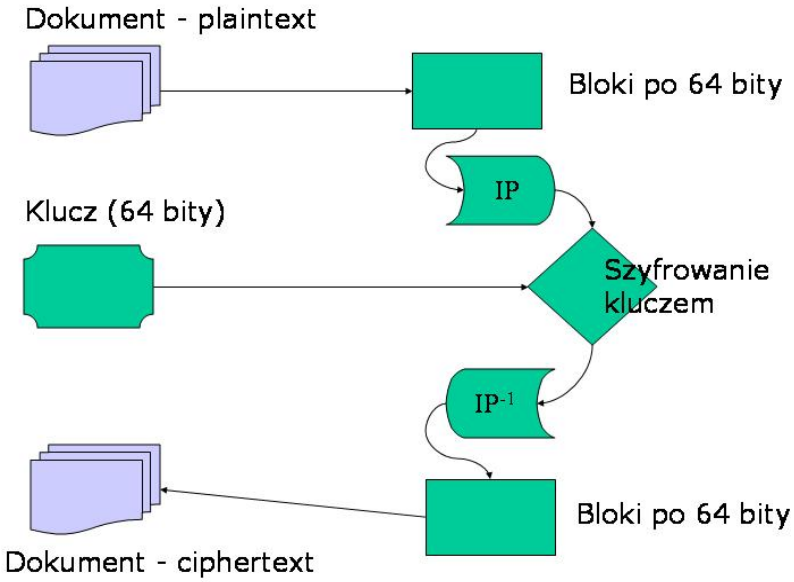
- Faza I – Generowanie dwóch kluczy: klucza publicznego i tajnego. Po wygenerowaniu klucz publiczny jest przesyłany wszystkim zainteresowanym.

Umożliwia on szyfrowanie danych. Klucz tajny umożliwia rozszyfrowanie danych zakodowanych przez klucz publiczny.

- Faza II – Użytkownik otrzymując klucz publiczny może szyfrować swoje dane i przesyłać je w postaci szyfru RSA do odbiorcy, który dysponuje kluczem tajnym, np. do banku. Klucz publiczny nie musi być chroniony, ponieważ przy jego wykorzystaniu nie możemy rozszyfrować informacji. W związku z tym klucz ten może być udostępniany wszystkim zainteresowanym użytkownikom.
- Faza III – Następuje odczyt informacji poprzez odszyfrowanie jej za pomocą klucza tajnego.

W obecnym czasie stosowane są wyłącznie algorytmy szyfrowania 1024, 2048, a nawet 4096 bitowego. Przy zastosowaniu takich algorytmów klucze są praktycznie całkowicie bezpieczne czyli nie można ich złamać przy współczesnym stanie wiedzy o algorytmach komputerowych i przy współczesnych możliwościach obliczeniowych systemów cyfrowych. Własność patentową do algorytmu RSA posiada Massachussets Institute of Technology i obejmuje zarówno szyfrowanie jak i podpisy cyfrowe.

Algorytm DES *(ang. Data Encryption Standard)* – jest to standard szyfrujący dane, który został zaakceptowany w 1977 r. przez Narodowe Biuro Standardów *(National Bureau of Standards)* obecnie Narodowy Instytut Standardów i Technologii (NIST). Główne zastosowanie ma w szyfrowaniu informacji tzw. nie utajnionej, ale ważnej (ang. *Unclassified but Sensitive)*. Algorytm został stworzony w firmie IBM i jest on rozbudowaną wersją szyfrowania LUCIFER. W algorytmie szyfrowanie polega na kodowaniu 64 bitowych bloków danych (8 znaków ASCII), kluczem długości 64-bitów (56 bity klucza + 8 bitów parzystości). Szyfrowanie za pomocą algorytmu DES przebiega następująco. Dane do zaszyfrowania w postaci bloków poddane zostają początkowej permutacji IP, następnie poddawane są złożonemu szyfrowaniu kluczem, a na końcu permutacji odwrotnej IP^{-1} do permutacji początkowej. Schemat szyfrowania przedstawia rysunek 16.



Rysunek 16. Schemat szyfrowania danych przez algorytm DES.

(źródło: http://e-handel.mm.com.pl/crypto/des_0.jpg)

Ostatnim algorytmem, który opisze jest algorytm AES (*ang. Advanced Encryption Standard*) jest młody algorytm, powstał w 1997 roku i następcą algorytmu DES. AES podobnie jak DES wykorzystuje bloki danych tylko, że są one o długości 128, 192 i 256 bitów oraz jest algorytmem symetrycznym. Proces szyfrowania podlega iteracjom, przy czym można wyodrębnić następujące etapy:

- rundę wstępną,
- pewną ilość rund standardowych (ich ilość zależy od długości klucza i wynosi odpowiednio 10, 12 lub 14) z których każda posiada 4 transformacje,
- rundę końcową .

Jak do tej pory algorytm ten jest odporny na wszystkie znane metody ataków, przy tym działa bardzo szybko na różnych platformach co jest spowodowane zoptymalizowaną zwartością kodu. Dodatkowo implementacja tego algorytmu jest bardzo łatwa.

Istnieje jeszcze wiele algorytmów szyfrujących od bardzo prostych po bardziej skomplikowane, jednak ja przedstawiłem w telegraficznym skrócie te najbardziej popularne i wykorzystywane w codziennych zastosowaniach.

2.2 Historia steganografii.

Steganografia to słowo pochodzące z greki słowo i oznacza dokładnie ukryte pismo, skład się dwóch członów: steganos - ukryty, tajny; graphein - pisać, malować, co w odniesieniu do kanału informacyjnego oznacza przesyłanie danych w taki sposób, aby osoby nieuprawnione, które mają dostęp do tych danych nie mogły się zorientować, że istnieją w nich ukryte informacje. Mechanizm steganografii opiera się głównie na zasadzie ukrywania informacji w takich miejscach wiadomości, które nie są wykorzystywane do przesyłania informacji lub miejscach których zmodyfikowanie nie będzie wpływało na główne treści informacyjne. Aby przesyłać ukryte informacje przy pomocy steganografii musimy utworzyć specjalny kanał, zdefiniowany[4] jako kanał komunikacyjny, który może zostać wykorzystany do przesłania informacji w sposób naruszający politykę bezpieczeństwa systemu. Metoda ta w dużej mierze wykorzystuje fakt, że dane przesłane są w taki sposób i w takich protokołach, które zgodnie z ich specyfikacją do tego nie są przewidziane, a jednocześnie narażają system na nieautoryzowane przesyłanie informacji. Steganografia działa zupełnie inaczej niż kryptografia Steganografia tak tworzy informację, że nie można wykazać jej istnienia. Oczywiście najlepszą techniką jest połączenie steganografii z kryptografią razem. Połączenie tych dwóch mechanizmów pozwala na zabezpieczenie się przed się przed następującą sytuacją: strona, która nadzoruje transmisję wykrywa przekaz steganograficzny, jednak ze względu na zastosowanie kryptografii nie może go odczytać.

Jedną z lepszych definicji steganografii napisał Duncan Sellars w książce „An Introduction To Steganography", a brzmi ona tak:
"The goal of steganography is to hide messages inside other harmless messages in a way that does not allow any enemy to even detect that there is a second secret present[5]"
w tłumaczeniu polskim:
Celem steganografii jest ukrycie wiadomości wewnątrz innej wiadomości w taki sposób, aby przeciwnik nawet nie był w stanie wykryć, że występuje druga, sekretna informacja.

Pierwsze zapiski o metodach ukrywania informacji pochodzą z tekstów Herodotusa żyjącego w latach 486-425 p.n.e.. Metody steganograficzne wykorzystywane były także w czasach

4 U. S. Department Of Defense, „Trusted Computer System Evaluation Criteria", 1985
5 Duncan Sellars, An Introduction To Steganography, http://www.cs.uct.ac.za/courses/CS400W/NIS/papers99/dsellars/stego.html

starożytnych między innymi w Egipcie i Chinach. Steganografię udoskonalano przez całe wieki. W czasie trwania dwóch wojen światowych metody steganograficzne wykorzystywane były głównie przez szpiegów. Metody z tamtego okresu opierały się głównie na tekście jako podstawowym nośniku tajnych informacji. Zastosowanie miał tzw. sympatyczny atrament, który służył do oznaczania pojedynczych liter w gazetach lub książkach zaznaczone litery tworzyły przekazywaną wiadomość. W wykorzystywaniu był także specjalny żargon służący do opisywania różnych sytuacji, na przykład zamiast używania słów statek czy port w celu opisania usytuowania wojsk używano słów z żargonu elektrycznego. Technika, która była dostępna w ówczesnym okresie umożliwiała także na przeprowadzanie skomplikowanych operacji typu: umieszczanie całego zminiaturyzowanego zdjęcia w kropce przekazywanego tekstu. Pierwszą próbą mającą na celu systematyzację pojęć związanych ze steganografią i systemami ukrywającymi informacji została podjęta dopiero na konferencji Information Hidding w 1996 roku[6]. Techniki zabezpieczające tajne wiadomości przez ukrywanie tajnych informacji można podzielić na podstawowe grupy:

- ukryte kanały są stosowane w kontekście systemów zabezpieczeń, które są wielopoziomowe jako główny kanał komunikacyjny. Kanały takie nie były z reguły stworzone w celu przesyłania informacji, są jednak wykorzystywane do nieautoryzowanego odczytu informacji z miejsc bardziej strzeżonych wykonywane to jest poprzez przesłanie tych danych do miejsc mniej strzeżonych.
- łączność anonimowa w zasadzie polega ona na ukryciu kontekstu informacji, realizowane to jest na przykład poprzez ukrycie nadawcy i odbiorcy. Zastosowanie anonimowości rozgłaszania polega z kolei na stworzeniu tzw. sieci DC (*ang. Dining Criptographers*), sieć ta jest tworzona na podstawie klasy algorytmów wymyślonych przez Davida Chauma.
- łączność specjalna tworzona jest w celu zapewnienia bardzo wysokiego poziomu bezpieczeństwa i niezawodności. System ten wykorzystywany był głównie przez wojskowe systemy radiokomunikacyjne, które wykorzystywały specyficzną modulację rozproszonego widma.
- steganografia jej podstawowym zadaniem jest ukrycie jak największej ilości tajnych informacji w innej informacji, które jest nośnikiem. Informacja ta musi być przesłana w niezauważony sposób. W steganografii możemy wyodrębnić dwa różne nurty: lingwistyczny polegający na ukrywaniu tekstu w tekście np: za pomocą metody

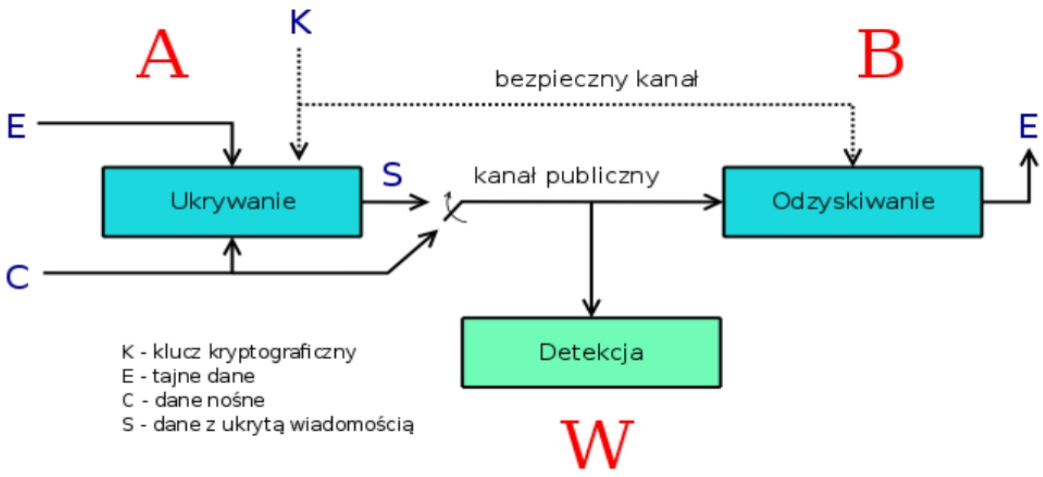
6 Steganologia: współczesne metody ochrony informacji, W.Garbarczuk, P.Kopniak, Politechnika Lubelska, 2005

akrostychu oraz techniczny, nurt ten wykorzystuje osiągnięcia techniki, np: przetwarzanie komputerowe obrazu.

W steganografii każdy kanał informacyjny jest typu punkt-punkt i posiada kilka własnych cech, które go określają: przepustowość, niezawodność, opóźnienie i tryb transmisji. Parametry te określają odpowiednio: ilość informacji które można przesłać kanałem, liczba błędnych bitów odebranych do całkowitej ilości przesłanych bitów (określona w wartości BER), średni czas, który upływa od wysłania informacji do jej odbioru, tryb transmisji może być synchroniczny lub asynchroniczny

Ukryte kanały informacyjne możemy podzielić na:

a) kanały pamięciowe – są to najprostsze i najczęściej wykorzystywane, informacje są przekazywane poprzez miejsca do, których nie są one przeznaczone,
b) kanały terminacyjne – działają na zasadzie uruchamiania i wyłączania odpowiednich procesów systemowych, w zdefiniowanych odstępach czasowych,
c) kanały wyczerpywania zasobów – wartość 0 i 1 definiowana jest poprzez stan zajętości konkretnego zasobu w systemie,
d) kanały czasowe – wykorzystują one czas potrzebny przez system lub proces na wykonanie jakiejś operacji,
e) kanały mocy – informacje w postaci bitów przekazywane są cyklicznie przez zmiany poboru prądu przez urządzania.



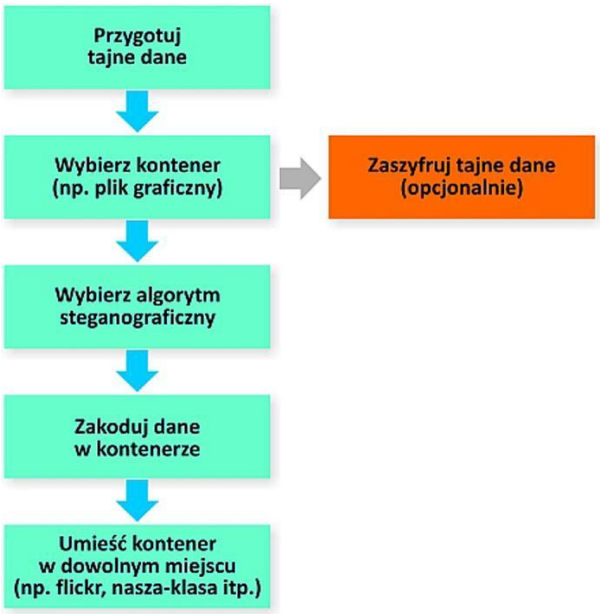
Rysunek 17. Schemat komunikacji steganograficznej.

(źródło: Steganografia – praca dyplomowa, Politechnika Warszawska)

Schemat działania komunikacji steganograficznej przedstawiony na rysunku 17, możemy opisać w następujący sposób. Nadawca A chce przesłać tajną informację E do odbiorcy B. Cała komunikacja musi się jednak odbyć kanałem publicznym, który jest kontrolowany przez nadzorcę W. Nadawca A chcąc ukryć fakt komunikacji, stara się wkomponować tajne dane w informacji nośnej C. Aby uzyskać skuteczną postać steganografii nadzorca W nie może rozróżnić informacji poprawnej, nie zawierającej tajnych danych, od przesyłanych informacji S, które zawierają poufne informacje. Jeśli A i B chcą dodatkowo zabezpieczyć przekaz mogą skorzystać z dostępnych funkcji kryptograficznych, które zabezpieczą przekazywane informacje. Mogą wykorzystać zarówno metodę kryptografii symetrycznej w której ustalony jest klucz kryptograficzny oznaczony jako K lub metodę niesymetryczną w której klucze publiczny oznaczony jako Kpub i prywatny oznaczony jako Kpryw. Zastosowanie technik kryptograficznych wpływa ogólnie na poprawę bezpieczeństwa przesyłanych informacji, jednak mogą one wywołać także niepożądane cechy. W przeważającej ilości przekazu tajnych informacji w formie steganograficznej skutkuje zamianą istniejącej już nieważnej części informacji. W związku z tym możemy utracić pewną charakterystyczną postać lub specyficzny histogram wiadomości. Dodatkowo zastosowanie mechanizmów kryptograficznych w stosunku do tajnej informacji powoduje jej zmianę, w wyniku czego ostateczny rozkład bitów jest nieprzewidywalny i prawie zawsze różni od standardowych histogramów określonych dla podmienianych części wiadomości.

2.3 Steganografia w różnych mediach.

Abyśmy mogli ukryć dane potrzebujemy odpowiedniego medium, w którym możemy je ukryć. W steganografii nośnik poufnych informacji nazywa się kontenerem. Każdy z wybranych kontenerów posiada własną pojemność. Pojemność kontenera wyrażana jest stosunkiem liczby bitów przekazu tajnego do liczby bitów, które są konieczne do ich ukrycia. Na przykład w informacji jawnej o wielkości 10 bitów jesteśmy w stanie ukryć jeden bit, wtedy pojemność wynosi 0,1 czyli 10%. Wszystkie dane, które mają być ukryte możemy wyrazić w cyfrowej postaci czyli możemy je zapisać przy pomocy spróbkowanego sygnału o dwóch stanach 0 i 1.



Rysunek 18. Schemat przygotowywania informacji steganograficznej.

(źródło: http://g1.pcworld.pl/news/thumb/1/6/169347)

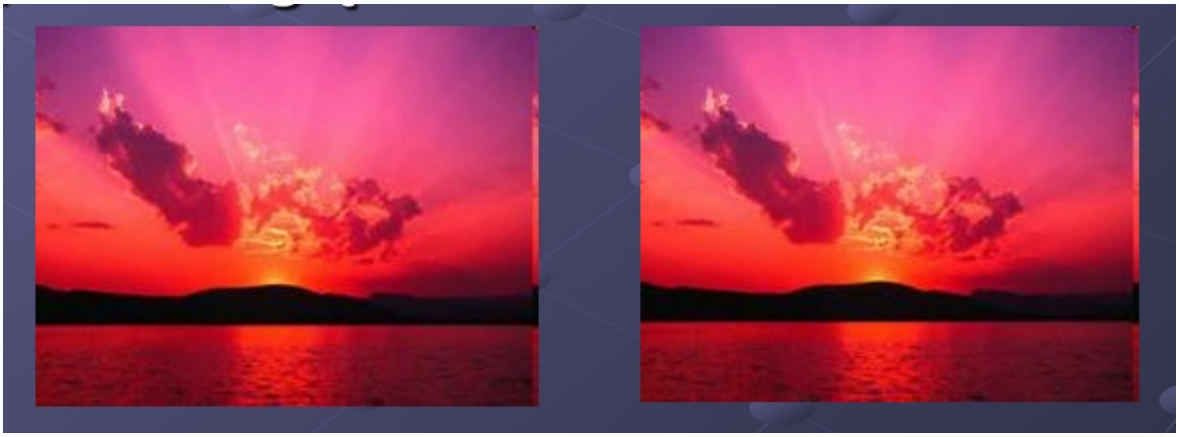
Steganografia w plikach graficznych.

Formaty graficzne możemy podzielić ze względu na ich budowę, są to następujące grupy:

- obrazy bez kompresji stratnej – format bezstratnej kompresji np: BMP charakteryzuje się tym, że obrazy przechowywane są w postaci kombinacji 3 barw składowych: czerwonej, zielonej i niebieskiej. Budowa formatu jest bardzo prosta, składa się z nagłówka oraz opcjonalnie z palety kolorów i danych obrazu.
- obrazy ze stratną kompresją – zastosowana kompresja wpływa w dużym stopniu na jakość obrazu, najpopularniejszym formatem tego typu grafiki jest JPG.

Idea ukrywania tekstu w bitmapie, polega na takim manipulowaniu bitami, aby wynikowy obraz był jak najmniej zmieniony, a jego jakość nie ucierpiała zbytnio. W tym celu powinniśmy zamieniać tylko najmniej znaczące bity, ma to zapewnić jak największą zgodność obrazu z jego oryginalną wersją. Najwyższą zgodność możemy otrzymać zamieniając tylko ostatni bit z każdego koloru. Kolor reprezentowany jest przez liczbę z zakresu od 0 do 255, różni się tylko wartością 1D. Zastosowanie takiej metody jest jednak niezbyt efektywne, związane jest to z tym, że stosunek wielkości zakodowanego tekstu do wielkości całej bitmapy wynosi ok. 11% jest on jednak zależny od wielkości samej bitmapy.

W łatwy sposób możemy zwiększyć efektywność ukrywania informacji w plikach graficznych dokonać tego możemy poprzez zamianę trzech najmniej znaczących bitów, wtedy to stosunek efektywności zwiększa się do granicy 38%. Zmiana tych trzech bitów spowoduje ponadto zmianę wartości piksela maksymalnie o liczbę 7D, przy takiej zmianie jakość przetwarzanego obrazu zmieni się w znikomym stopniu. Najlepszy efekt otrzymamy, kiedy skorzystamy z obrazów posiadających 24 bitową głębię kolorów. Wtedy to wszelkie zmiany poczynione w bitmapie są mało zauważalne, a zarazem możemy w obrazie umieścić więcej ukrytych informacji. Rysunek 19 pokazuje dwa obrazy, jeden oryginalny, a drugi z ukrytą informacją.



Rysunek 19. Porównanie obraz bez ukrytych informacji i z ukrytą informacją.

(źródło: Steganografia – Konrad Ogłaza – Politechnika Krakowska)

Przy stosowaniu metod steganograficznych możemy także wykorzystywać funkcje szyfrowania ukrytych informacji (tekstu), możemy np. skorzystać z popularnych metod kodowania Huffmana czy też Shannon-Fano. Takie kodowanie ukrywanych informacji w dużym stopniu może podnieść stosunek efektywności ukrywania informacji i może osiągać poziom do ok. 63 %.

Steganografia w tekście.

Tekst należy do najstarszych mediów, które były wykorzystywane w steganografii. Wiadomość zapisana w formie elektronicznej lub nawet na papierze może nieść pewną dawkę informacji ukrytych. Już podczas II Wojny Światowej przy pomocy steganografii preparowano specjalne wiadomości tekstowe, w których znajdowała się informacja niejawna.

Informację taką można było odczytać jedynie znając algorytm lub klucz wykorzystany do jej ukrycia. Tekst nie jednak dobry jako medium ponieważ posiada duże ograniczenia:

- pomieści tylko niewielką ilość informacji,
- wykrycie niejawnej informacji jest łatwe do wykonania, nawet w przypadku gdy nie posiadamy odpowiedniego oprogramowania,
- możliwe jest bardzo łatwe zniszczenie lub zmanipulowanie informacją zapisaną w tekście.

Istnieją dwa podstawowe algorytmy steganograficzne działające na kontenerach tekstowych, zaliczyć do nich możemy:

- algorytm osadzający informację poufną w taki sposób, że możliwe staje się przechowywanie tekstu w formie nie cyfrowej. Do tej grupy algorytmów zaliczamy mechanizmy przedstawione przez J.T.Brassila czyli line-shift-coding, word-shift-coding, featue-coding, metodę semantyczną i syntaktyczną.
- algorytm cyfrowy zaliczamy do niego tzw. metodę białych znaków.

Metoda line-shift-coding opiera się na takim manipulowaniu położeniami wiersza tekstu w stosunku do wierszy występującymi nad i pod nim. Przesunięcie musi być stałe i dla tego algorytmu wynosi 1/300 cala. Do zalet tego algorytmu zaliczamy to, że ukryta informacja nie będzie zniszczona nawet po wydrukowaniu dokumentu. Za istotną wadę możemy uznać to, że istnieje duże prawdopodobieństwo wykrycia informacji przez osoby nieuprawnione. Działanie algorytmu polega na tym, że przesunięciu podlegają wybrane wiersze tekstu, więc możemy wyodrębnić trzy stany wiersza:
S=0 czyli brak przesunięcia wiersza
S=1 czyli przesunięcie wybranego wiersza w górę
S=-1 czyli przesunięcie wybranego wiersza w dół.
Line-shift-coding bazuje również na słowach kodowych. Słowa kodowe są reprezentowane bitowo i zawierają w sobie zdefiniowany zbiór wartości np:
S=-1 – wartość bitowa wynosi 1
S=1 – wartość bitowa wynosi 0

Algorytm word-shift-coding opiera się na manipulacji położenia pojedynczych słów lub bloków słownych względem osi X. Dokumenty, które posiadają wyrównanie do krawędzi nadają się do tego doskonale. Możemy rozróżnić dwie odmiany tego algorytmu:

- w każdym z wierszy odnajdywana jest największa i najmniejsza odległość istniejąca pomiędzy poszczególnymi słowami. W dalszym kroku największa odległość jest zmniejszana o wartość, która jest stałą, a najmniejsza wartość jest powiększana o taką samą wartość,
- Wiersz w dokumencie dzieli się trzy bloki słowne. Następnie przesuwany jest tylko blok środkowy względem bloku skrajnie lewego lub bloku skrajnie prawego.

Opisywane metoda manipulowania również bazuje na słowach kodowych. I podobnie jak w metodzie line-shift-coding sława kodowe mogą być zapisane w postaci trzech stanów:
S=0 gdy słowo nie uległo przesunięciu
S=1 gdy słowo uległo przesunięciu w lewo
S=-1 gdy słowo uległo przesunięciu w prawo

Najpopularniejszym kodowaniem jest metoda feature-coding polega ona na wykorzystaniu wysokości liter należących do zbioru: { , , , , , , } L b d f h k l t =. Feature-coding opiera się na zmianie wysokości znaków, które są zawarte w zbiorze L, a słowa kodowe mogą zostać zapisane jako „wydłużenie" lub „skrócenie" wysokości kolejnych znaków. Taki zapisany znak może posiadać jeden z dwóch stanów:
S=0 – wysokość pozostaje niezmieniona
S=1 – wysokość uległa zmianie.
Dzięki takiemu podejściu możliwe staje się zapisywanie słów kodowych binarnie.

Dodawanie tajnych danych w metodzie przebiega w dwóch etapach:

- wysokości poszczególnych znaków ze zbioru L są albo zwiększane albo zmniejszane o losową wartość jest tak robione, aby utrudnić niepowołanej osobie określenie ich wysokości wtedy gdy znajdują się w stanie S=0,
- słowa kodowe są zapisywane wraz modyfikacją ich wysokości, jednak taki zapis przeprowadzane jest tylko dla wybranych znaków ze zbioru L i odbywa się poprzez zwiększenie lub zmniejszenie.

Opisana metoda gwarantuje bardzo wysoki poziom bezpieczeństwa ukrywanych informacji, ponieważ niewielka zmiana w wysokości znaków jest niewidoczna i ignorowana. Aby odczytać ukrytą informację musimy posiadać oryginalny dokument. Istnieje jednak inna możliwość, która wyeliminuje tą konieczność, należy użyć klucza steganograficznego. Klucz

taki będzie wskazywał na znaki zapisane w słowach kodowych.

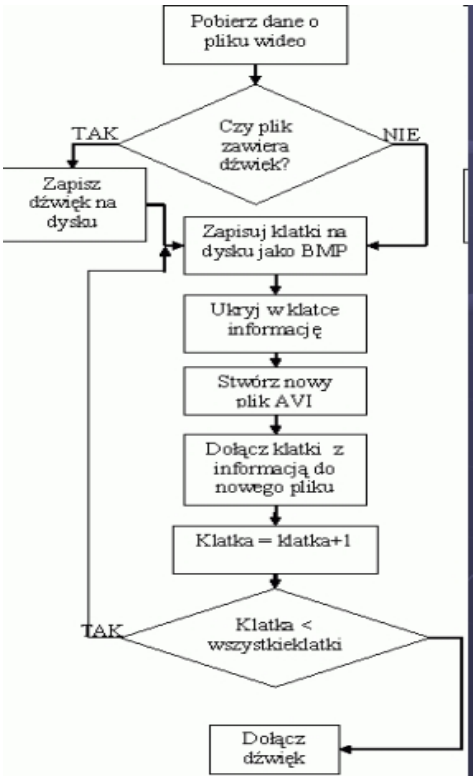
Ostatnim mechanizmem który opiszę jest metoda białych znaków stosowana przy algorytmach cyfrowych. Metoda białych znaków jest jedyna taką metodą ukrywania danych w tekście otwartym, której nie można wykorzystać we wszelkich informacjach, która mają zostać wydrukowane. Wynika to z tego, że białe znaki występujące w tekście są widoczne dla człowieka jedynie w takich sytuacjach kiedy są one umieszczone pomiędzy znakami lub słowami na początku linii. Kiedy białe znaki umieścimy na kocu wiersza są one niewidoczne. Do wyjątków należy sytuacja kiedy tekst jest wyrównany do obu marginesów. Słowo kodowe może być reprezentowane binarnie w przypadku dodania określonej liczby takich znaków na końcu wiersza. Reprezentacja przez spacje może wyglądać następująco:
2 spacje – jest reprezentacja 1 bitu
4 spacje – reprezentują 2 bity
8 spacji – reprezentuje zaś 3 bity
Ilość danych jaką możemy osadzić w tekście jest bardzo ograniczona i można ją zaprezentować poprzez zależność:
l = il_wierszy/8

Metoda ta jest dość prosta, a co się z tym nierozłącznie wiąże posiada wiele ograniczeń:

1. łatwo go wykryć
2. jest podatny na uszkodzenia
3. można ukryć dość małą ilość informacji
4. wielkość kontenera znacznie wzrasta

Steganografia w plikach video.

Ukrywanie informacji w plikach video jest praktycznie takie samo jak w plikach graficznych i działa na tej samej zasadzie, jedyna różnica jak wynika z zastosowania tego kontenera jest to, że możliwe jest zaszyfrowanie dużo więcej informacji. Jedynym ograniczeniem jest to, że nie możemy praktycznie stosować jakiejkolwiek kompresji ponieważ może ona zniszczyć część, lub całość ukrytej informacji. Rysunek 20 przedstawia algorytm osadzania informacji ukrytej.



Rysunek 20. Algorytm ukrywania informacji w plikach video.

(źródło: Steganografia – Konrad Ogłaza – Politechnika Krakowska)

Steganografia w dźwięku.

Plik dźwiękowy jest doskonałym kontenerem do ukrycia niejawnych informacji, jednak plik ten musi być odpowiednio przygotowany, aby spełniał swoje zadanie. Dzięki wykorzystywaniu technik steganograficznych o dużym stopniu zaawansowania możliwe jest ukrywanie dużej ilości danych, w stosunkowo małym pliku audio, zachowując jednocześnie bardzo wysoki poziom bezpieczeństwa na wykrycie lub uszkodzenie. Jeśli zastosujemy współczesne przetworniki analogowo-cyfrowe i cyfrowo–analogowe możemy poprawnie wyodrębnić interesujące nas informacje z sygnału analogowego, który powstał z sygnału cyfrowego zawierającego dodatkowe informacje. Użycie steganografii w dźwięku analogowym jest dość trudna w realizacji, ponieważ wymaga ona działania bezpośrednio na sprzęcie o dużej mocy obliczeniowej. Sprzęt taki musi być w stanie na bieżąco dokonywać obliczeń transformaty Fouriera. Przy zastosowaniu sygnału cyfrowego, który zapisany jest na nośnikach w postaci plików wykorzystanie metod steganograficznych jest znacznie prostsze i skuteczniejsze. Przy transmisji danych poprzez sygnał analogowy, występują pewne ograniczenia między innymi niemożliwość składowania sygnału bez uszkodzenia osadzonej w nim informacji za pomocą nośników analogowych. Inaczej sprawa ma się przy informacji

zapisanej w pliku cyfrowym, który może być przesyłany różnymi sposobami, nie jest wymagane odtworzenie pliku w celu wyodrębnienia zawartej w nim informacji. Steganografię na audio przeprowadza się przeważnie na plikach WAVE. Przy otwarciu pliku WAVE odczytujemy z jego nagłówka potrzebne nam informacje między innymi o ilości bitów przypadających na sampel. Z nagłówka możemy się również dowiedzieć czy nie został w stosunku do pliku dźwiękowego wykorzystano jakiś algorytm kompresji. Jeżeli wykryjemy, że plik został skompresowany nie powinniśmy go edytować, ponieważ możemy uszkodzić jego strukturę. Chcąc ukryć informacje w pliku, który jest skompresowany musimy go skonwertować go do formatu WAVE. Potem odnajdujemy koniec nagłówka, oznaczony jest on czterobajtowym znacznikiem data po tym znaczniku znajdują cię dane dźwiękowe, które możemy poddać edycji.

Tabela 7. Dane nagłówkowe pliku WAVE.

(źródło: http://www.yaotzin-steganografia.yoyo.pl/tekst/wavean/wavnag.htm)

Numer bajtu	Nazwa pola	Rozmiar w bajtach	Wartość	Opis
0	ID	4	'RIFF'	Identyfikator pliku RIFF
4	Rozmiar danych	4	Długość pliku - 8	Liczba określająca długość danych w pliku w bajtach
				z pominięciem pierwszych 8 bajtów nagłówka
8	Format ID	4	'WAVE'	Format pliku
12	Opis ID	4	'fmt '	Początek części opisowej pliku
16	Rozmiar opisu	4	.	Rozmiar części opisowej, dla fmt wynosi zwykle 16
20	Format Audio	2	0001h	Rodzaj kompresji. 1 - bez kompresji, 2- kompresja (APCDM, a-law, u-law…) modulacja PCM.
22	Liczba kanałów	2	0x0001, 0x0002	1 - mono, 2 - stereo

24	Częstotliwość	4	8000, 44100, itd.	Częstotliwość próbkowania w Hz
28	Częstotliwość bajtów	4	.	Częstotliwość * Liczba kanałów * Rozdzielczość / 8
32	Rozmiar próbki	2	.	Liczba kanałów * Rozdzielczość / 8
34	Rozdzielczość	2	8, 16, 24, 32	Rozdzielczość w bitach
36	Dodatkowe parametry	x	.	Zwykle brak tego pola
36+x	Dane ID	4	'data'	Początek części z danymi
40+x	Rozmiar danych	4	.	Rozmiar bloku danych
44+x	Dane	.	.	.

Wykorzystanie mechanizmów steganograficznych jest również możliwe w innych mediach, które mogą służyć jako kontener. Do takich kontenerów możemy zaliczyć:

- pliki wykonywalne .exe i .dll
- pliki HTML,
- telefonia VOIP,
- sieci komputerowe.

2.4 Stegoanaliza.

Stegoanaliza jest to nauka, która zajmuje się wykrywaniem ukrytych informacji w kanałach komunikacyjnych. Działania stegoanalizy nie koniecznie muszą prowadzić do odczytania dokładnej treści ukrytych informacji, częściej ma na celu wskazanie istnienia takiego ukrytego kanału steganograficznego. Wykrywanie kanału steganograficznego sprowadza się do przeanalizowania tych części wiadomości lub strumienia danych w celu wykrycia jakiś anomalii. Podejście takie wynika bezpośrednio z faktu, że ukrywana informacja umieszczana jest w miejscach, które nie są przeznaczone do przesyłania informacji lub w miejscu takich danych, które są nadmiarowe. Możemy przedstawić dwa główne sposoby wykrywania anomalii:

- w pierwszym sposobie opieramy się na przeanalizowaniu takich części informacji (np. pól nagłówka TCP/IP), których strukturę znamy i możemy skontrolować ponieważ jest w pełni przewidywalna lub takie części dla których są zdefiniowane wartości wynikające z istniejących standardów lub stosowanych praktyk. Ważną czynnością jest kontrola czy pojawiają się wartości nadmiarowe oraz czy elementy sygnalizujące pojawienie dodatkowych danych mają pokrycie w danych,
- drugi sposób działa ma zasadzie porównania wartości części wiadomości i nadanie im statusu jako prawdopodobnych lub nie dla danego systemu bądź protokołu. Takie podejście stosuje się do wartości które są ściśle określone. Możemy je także stosować do wartości pseudolosowych lub takich których histogram jest charakterystyczny.

Rozdział III
Steganografia w protokole TCP/IP.

W ówczesnym czasie przez sieci komputerowe przesyłana jest ogromna ilość danych, ze względu na niekontrolowany ich przesyła komunikacja sieciowa stała się dużym obszarem gdzie można wykorzystywać metody steganograficzne. Sieci w przeważającej części oparte o protokoły TCP/IP. Ponieważ następuje ciągły rozwój protokołów sieciowych powstają coraz to nowsze techniki wykorzystujące je jako nośnik ukrytej informacji. Każdy z powstających protokołów sieciowych jest tworzony z przeznaczeniem do konkretnych zadań, warunków działania. Pierwsze protokoły, które powstały stworzone zostały z jednym konkretnym zadaniem – miały działać, w obecnych czasach protokoły mają zaimplementowane wiele mechanizmów zapewniających bezpieczeństwo. Jednak w dalszym ciągu możemy zauważyć, że w praktycznie każdym miejscu modelu OSI w obrębie protokołów, które realizują usługi każdej z warstw możemy znaleźć takie elementy, które możemy wykorzystać do naszych celów. Spowodowane to jest tym, że protokoły sieciowe posiadają takie cechy (redundancja przesyłanych danych, brak jednoznaczności), które mogą zostać wykorzystane. W wielu protokołach implementowane są takie mechanizmy, które nie są jeszcze wykorzystywane, a stwarzają furtkę do wykorzystywania ich w celach steganograficznych.

3.1 Analiza nagłówka protokołów IP.

W protokole IP polami, które mogą zostać wykorzystane do przesłania niejawnych informacji są, pokazane jest to na rysunku 21 (żółte pola):

- pole Typ usługi (Type of Service) do wykorzystania od 1 do 8 bitów,
- pole identyfikacyjne do wykorzystania 16 bitów,
- pole flag fragmentacji do wykorzystania 2 z 3 dostępnych bitów,
- pole opcji

Bity 0-3	4-7	8-15	16-18	19-23	24-31
Wersja	IHL	Typ usługi	Długość całkowita		
Identyfikator			Flagi	Przemieszczenie fragmentacji	
TTL		Protokół	Suma kontrolna nagłówka		
Adres źródłowy					
Adres docelowy					
Opcje					Dopełnienie

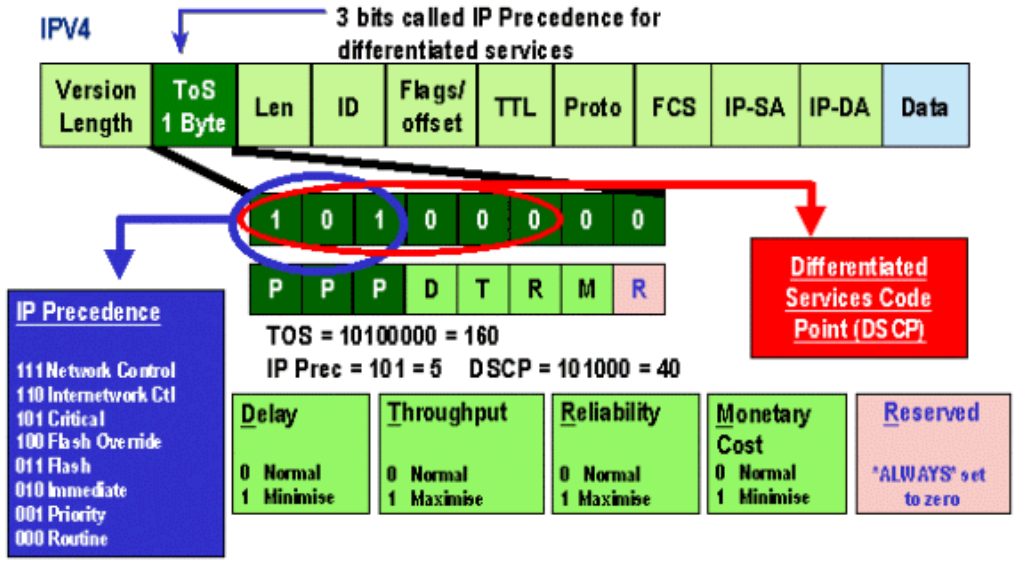
Rysunek 21. Datagram protokołu IP – pola wykorzystywane w steganografii.

(źródło: Steganografia – Politechnika Krakowska, Konrad Ogłaza)

Pole Type of Services jest polem o długości 8 bitów nie posiada jednak jednoznacznej budowy co powoduje, że jego interpretacja jest różna w zależności od implementacji protokołu IP. W wyniku tej niejednoznaczności można w tym polu utworzyć ukryty kanał. Pole to często nie jest wykorzystywane (zwłaszcza w małych sieciach) co stwarza, że pole to jest puste i może zostać wykorzystane w celu przesyłania informacji nie jawnych. W nowoczesnych sieciach jest jednak ono wykorzystywane ma to na celu poprawę jakości działania oraz zapewnienie odpowiedniej przepustowości i niezawodności. Pole TOS we współczesnych sieciach nosi nazwę DSCP. Chcąc za pomocą tego pola przesyłać informacje ukryte mamy kilka możliwości:

- możemy wykorzystać najmłodszy bit (nie wpływa to na transmisję),
- możemy wykorzystać trzy najstarsze bity, które określają pierwszeństwo,
- możemy użyć całego pola Type of Services (8 bitów),
- wykorzystujemy 4 bit, który określa opóźnienie,

Wymienione wyżej możliwości różnią się przede wszystkim swoją pojemnością oraz wykrywalnością. Przy analizie przesyłanych danych, zmodyfikowanych datagramów IP można zidentyfikować nieprawidłowe użycie pola TOS. Mniejszą wykrywalnością cechuje się wykorzystywanie takich bitów, które nie są używane w danej sieci. Wykorzystanie pola Typu usługi niesie za sobą jeszcze jedno niebezpieczeństwo, pole to może zostać wymazane lub usunięte. Zawartość pola ToS przedstawiona została na rysunku 22.



Rysunek 22.Zawartość pola ToS.

(źródło http://www.cisco.com/en/US/tech/tk543/tk762/technologies_white_paper09186a00800b0828.shtml)

Pole identyfikacji służy do identyfikowania każdego przesyłanego datagramu, a co z tego wynika musi być unikalne w obrębie jednej transmisji. Wyjątkiem tu są takie dane, które podlegają fragmentacji czyli każdy z fragmentów posiada taki sam numer identyfikacyjny. Wykorzystać to możemy w niniejszy sposób: musimy zapewnić unikalne numery identyfikacyjne dla różnych datagramów, a więc mamy możliwość na takie spreparowanie tych numerów w których przesyłane ukryte informacje. Dopóki preparowane numery są unikalne komunikacja przebiega prawidłowo. Dodatkowo ukrywanie informacji w polu identyfikacje jest trudne do wykrycia, potrzebne by było do tego wiele informacji: nazwa systemu operacyjnego sprzętu wysyłającego pakiety, informacje o transmisjach IP wykonywanych przez tę maszynę. Informacje te są różne ponieważ, każdy z systemów operacyjnych generuje numery identyfikacyjne dla przesyłanych datagramów w zależności od implementacji stosu TCP/IP. Jeżeli już ukryty kanał w polu identyfikacyjny zostanie wykryty atak na niego jest trudny ponieważ wymaga pracochłonnych działań, między innymi śledzenie całego ruchu w obrębie całej transmisji oraz trzeba by było wygenerować nowe wartości numerów identyfikacyjnych.

W polu flag fragmentacji mamy 3 bity, które określają sposób obsługi datagramu w przypadku jego defragmentacji. W tym celu używane są 2 bity, ostatni z bitów musi zawsze mieć wartość zero (bit oznaczany jako RF). Jednakże zmiana tego bitu nie wpływa na

zachowanie pakietu, zatem można go wykorzystać jako kanał przesyłający ukryte dane. Zastosowanie tego pola nie jest bezpieczne ponieważ tak jak pole TOS wszelkie zmiany w strukturze tego pola są łatwo wykrywalne, a jego usunięcie polega na zerowaniu najstarszego bitu.

Pole opcji w większości przypadków nie jest używane, dodatkowo posiada ono zmienną długość. Implementacja tego pola w protokole IP pozwala na definiowanie własnych opcji, które są numerowane i dzielone według własnego upodobania. Definiowanie własnych opcji stwarza możliwość wykorzystania numerów dotąd nie używanych do przesyłania danych. Pojemność takiego kanału informacyjnego jest zależny od zastosowanych opcji (od jednego bitu do kilkunastu bajtów). Wysyłanie poufnych danych poprzez pole opcji nie jest zbytnio bezpieczne ponieważ w mogą zostać usunięte przez prawidłowo skonfigurowane routery- dzieje się tak tylko w przypadku opcji zdefiniowanych na podstawie dokumentacji RFC. Wykorzystanie własnych opcji nie powoduje ich usunięcia.

3.2 Analiza nagłówka protokołu TCP.

W protokole TCP polami, które wykorzystywane są w steganografii to pola przedstawione na rysunku 23:

Opis nagłówka TCP

	Bity 0–3	4–7	8–15	16–31
0	Port nadawcy			Port odbiorcy
32	Numer sekwencyjny			
64	Numer potwierdzenia			
96	Długość nagłówka	Zarezerwowane	Flagi	Szerokość okna
128	Suma kontrolna			Wskaźnik priorytetu
160	Opcje (opcjonalnie)			
160/192+	Dane			

Rysunek 23. Datagram protokołu TCP – pola wykorzystywane w steganografii.

(źródło: Steganografia – Politechnika Krakowska, Konrad Ogłaza)

- pole numer sekwencyjny gdzie możemy wykorzystać 32 bity,
- pola zarezerwowane,
- pole wskaźnik priorytetu – 16 bitów,
- pole opcji.

Numer sekwencyjny wybierany jest zawsze na początku negocjacji przy nawiązywaniu połączenia i wysyłany jest w pierwszym pakiecie TCP. Wartość tego numeru jest losowa i powinna być zmieniana niezależnie od nawiązanych połączeń. Pole to może zostać wykorzystane do przesłania informacji tylko raz podczas nawiązywania połączenia, ponieważ początkowy numer sekwencyjny nie musi być unikalny wobec innych numerów istniejących w innych połączeniach. Jednorazowa możemy więc przesłać 4 bajty informacji (32 bity). Wykrywalność informacji ukrytych w numerze sekwencyjnym jest znikoma, ponieważ trzeba by było monitorować wszystkie nawiązywane połączenia TCP i przeprowadzać ich bieżącą analizę.

Pole zarezerwowane jest polem sześciobitowym obecnie nieużywanym, które musi zawsze posiadać wartość zero (wynika to z dokumentacji RFC). Podobnie, jak ma to miejsce w polach bitów TOS i zarezerwowanej flagi fragmentacji protokołu IP, nieużywane bity mogą zostać wykorzystane do transmisji. Pole to jest najczęściej ignorowane w istniejących sieciach, a datagramy TCP z wartością tego pola różną od zera są przesyłane bezproblemowo. Wykrywanie i uniemożliwianie ukrytej transmisji jest dosyć proste w realizacji, należy tylko wyzerować to pole.

Wskaźnik priorytetu to pole 16-bitowe pole. W obecnych czasach stosowane jest bardzo rzadko, pole to ma zastosowanie tylko gdy jednocześnie zostanie ustawiona flaga URG w nagłówku segmentu TCP. Bity tego pola mogą jednak zostać użyte do przesyłu informacji, a, polega to na zapisaniu przesyłanej informacji do pola ważności, przy jednoczesnym pozostawieniu wyzerowanej flagi URG. Podejście takie powoduje, że wskaźnik priorytetu nie będzie interpretowany. Datagramy, które mają tak zmieniony nagłówek są bez przeszkód przesyłane. Teoretyczna pojemność takiego kanału wynosi 2 bajty. Pole to wskazuje na ostatni istotny bajt w segmencie i jest dodawane do numeru sekwencyjnego. Wykrycie ukrytej transmisji nie jest zbyt trudne, ponieważ pole priorytetu jest używane sporadyczne, więc zapisanie go może budzić podejrzenia. Drugą ważną rzeczą jest to żeby flaga URG wyzerowana. Wykrycie transmisji przy pomocy tego pola polega na obserwacji całej transmisji i wyłapywaniu segmentów bez ustawionej flagi, a ze wskaźnikiem ważności różnym od zera.

Pole opcji umieszczone w nagłówku TCP w odróżnieniu do protokołu IP jest używane ustawiane są w nim trzy opcje: Maximum Segment Size (MSS), SACK-Permitted, No

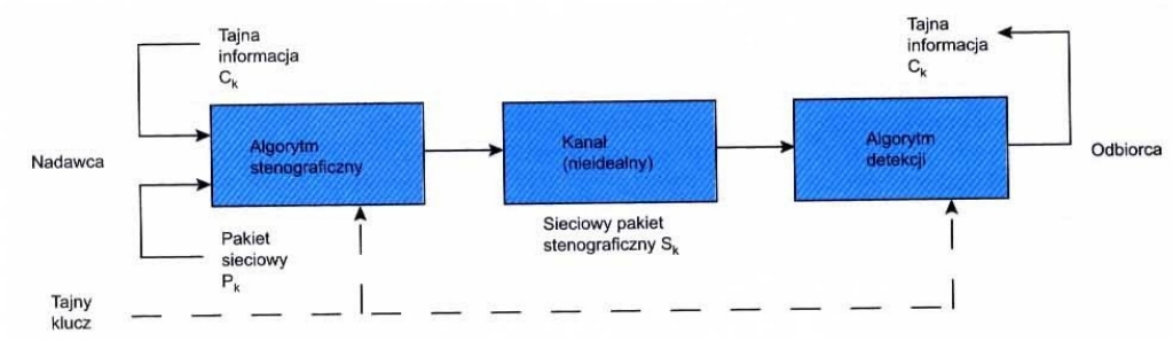
Operation, które przeważnie przesyłane są podczas nawiązywania połączenia. Tak jak w polu opcji protokołu IP możliwe jest zdefiniowanie własnej opcji, co umożliwia nam wykorzystanie tego pola do przesyłu ukrytych informacji. Musimy się jedynie ograniczać wielkością segmentu TCP. Przesyłanie ukrytych informacji może wzbudzić zainteresowanie w przypadku kiedy zdefiniujemy jakieś swoje niestandardowe opcje.

3.3 Kanały informacyjne.

Jak widać z powyższych opisów i rysunków steganografia wykorzystuje pola zawarte w nagłówkach protokołów IP i TCP, tych nagłówków, które przeważnie są nadmiarowe i opcjonalne. Dodatkowo korzystać możemy z pól, które oddziałują na zachowanie tych protokołów lub realizują konkretne zadania. Ramki nagłówków tworzą dodatkowe pasmo przenoszące informacje, które mają zostać ukryte przed nie uprawnionymi użytkownikami. Wskazane pola są dość dobrym sposobem na ukrywanie danych, jednak istnieje jeszcze kilka aspektów, mających ogromne znaczenie, a mianowicie musimy zaimplementować odpowiedni nadajnik i odbiornik, uzgodnić używane narzędzia w sieci. Niemal wszystkie ukryte kanały w protokołach TCP/IP są kanałami pamięciowymi, spowodowane jest to tym, że w nagłówkach tych protokołów znajdują się pola (opisane w punktach 3.1 i 3.2) w których możliwe staje się umieszczenie ukrytych informacja nie zakłócając komunikacji i transmisji. Poza używaniem pól najczęściej wykorzystywanych i najpopularniejszych istnieją jeszcze inne możliwości przekazania tajnej informacji. Jedną z nich jest umieszczenie informacji w polu ID nagłówka IP, w polu o długości 16 bitów możemy umieścić zakodowane wartości znaków w ASCII. Drugą możliwością przy polu ID jest ukrywanie informacji w najstarszym bajcie tego pola przy jednoczesnym losowaniu najmłodszego bajtu. Kolejną możliwością przesłania danych jest wykorzystanie pola TTL nagłówka IP, dane muszą mieć wyznaczone wcześniej wartości, które umożliwią przesłanie jednego bitu ukrytej informacji. Z metod pamięciowych najbardziej rozwiniętą techniką jest NUSHU opracowana została ona przez Jolantę Rutkowską. Technika ta opiera się na ukrywaniu danych w numerach ISN o pojemności 32 bitów, metoda ta nie generuje sztucznego ruchu sieciowego oraz zapewnia kontrolę transmisji i korekcję błędów. NUSHU jest to protokół pasywny czyli nie wymagane jest odesłanie informacji zwrotnych.

3.4 Wykorzystanie protokołów TCP i IP w steganografii.

Jak wynika z powyższego opisu możliwość ukrycia wiadomości wiąże się ze źle zaprojektowanymi protokołami komunikacyjnymi. Schemat przesyłania informacji polega na przesyłaniu ukrytych danych w pakiecie danych. Pakiet ten służy głownie do maskowania obecności tajnych danych. Na samym początku Nadawca musi stworzyć steganograficzny pakiet danych, w którym ukrywa informację skierowaną do Odbiorcy. Pakiet steganograficzny powstaje z połączenia pakietu sieciowego i informacji do ukrycia. Dodatkowym zabezpieczeniem przy przesyłaniu ukrytej informacji może być użycie tajnego klucza, który jest znany tylko Nadawcy i Odbiorcy. Transmisja takiego zmanipulowanego pakietu nie przebiega jednak idealnie, wynika to bezpośrednio z tego, że istnieją pewne przypadkowo wygenerowane pakiety steganograficzne, które mogą zakłócić kolejność przesyłanych pakietów. Taka sytuacja nie jest pożądana ponieważ ma to wpływ na ukrytą informację. Spreparowany pakiet danych przechodzi przez wiele węzłów, w których nie może zostać zauważona ukryta informacja, ani nie może zostać ona odrzucona. Po dotarciu pakietu z ukrytą informacją do Odbiorcy, pakiet poddawany jest detekcji odpowiednim algorytmem w celu wyłuskania pożądanej informacji. Rysunek 24 przedstawia działanie przesyłu takich pakietów.



Rysunek 24. Schemat przesyłu informacji.

(źródło: Hakin9 – 2005/03 – Łukasz Wójcicki)

Jak już zdążyliśmy się dowiedzieć z tej pracy w protokole TCP w jego nagłówku istnieje pole flag, które określa przeznaczenie oraz zawartość segmentu TCP. Istnieją 64 kombinacje przekazywanych flag, niektóre z nich są nadmiarowe, co może być wykorzystane do utworzenia tajnych kanałów. Nadmiarowa kombinacja bitów pola Flag może wyglądać tak, jak w tabeli 8.

Tabela 8. Kombinacja nadmiarowych bitów.

URG	ACK	PSH	RST	SYN	FIN
0	1	1	0	0	1

W przeważającej większości segmenty TCP mają ustawioną flagę ACK na wartość 1, ponieważ połączenie protokołu TCP jest dwukierunkowe. Kombinacja nadmiarowa przedstawiona powyżej możemy wytłumaczyć następująco. Jedna ze stron połączenia chce zakończyć połączenie (flaga FIN = 1), w tym samym czasie przesyłając potwierdzenie otrzymania pakietów (ACK = 1). Ustawiony jest także bit PSH, mający za zadanie natychmiastowe przesłanie żądania do warstwy aplikacji. Flaga URG nie jest w tym momencie ustawiona i do czas kiedy nie otrzyma ona wartości 0, istnieje 16 bitów nadmiarowych informacji, które mogą zostać wykorzystane do przekazania tajnych informacji. Takie sytuacje występują zawsze kiedy bit URG nie zostanie ustawiony. Podobne kombinacje można uzyskać wykorzystując bit SYN. Jednym z ograniczeń jest to, że bity PSH i URG nie mogą być używane jednocześnie.

Steganografia w protokole IP możemy zastosować w inny sposób. Wykorzystujemy do tego komunikat IGMP. Komunikat taki podczas rozsyłania kapsułkowany jest w datagramie IP. Komunikat IGMP ma długość 8 bajtów, na który składają się:

- pole rodzaju komunikatu (8 bitów),
- pole maksymalnego czas u odpowiedzi (8 bitów),
- suma kontrolna (16 bitów),
- adres grupy (32 bity).

Podczas enkapsulacji tego komunikatu wartość pola protokołu w nagłówku IP przyjmuje wartość 2. Komunika IGMP jest kapsułkowany w taki sposób, że część stała nagłówka IP mieści się w 20 bajtach, a komunikat w 8. Niestety datagram w większości przypadków nie mieści się w jednej ramce fizycznej, ze względu przesyłanie datagramów w różnych sieciach fizycznych w których nałożone są ograniczenia wielkości przesyłanych ilości danych. Parametrem, który określa tę wielkość nazywa się MTU (*ang. Maximum Transfer Unit*). Dlatego przesyłane dane, jeśli nie mieszczą się w jednej ramce fizycznej są fragmentowane.

Komunikaty IGMP mogą wystąpić w dwóch postaciach, jako:

- raport przynależności do grupy oraz komunikat o wyjściu z grupy,
- jako komunikat odpytujący o przynależność do grupy.

Przy uwzględnieniu tych komunikatów możemy wydzielić poniższe rodzaje datagramów protokołu IP:

- od hosta do routera (2 rodzaje z dozwoloną fragmentacją i zabronioną fragmentacją),
- od routera do hosta (2 rodzaje z dozwoloną fragmentacją i nie dozwoloną fragmentacją).

Przy takim ułożeniu przesyłanych datagramów, jeden po drugim, dostajemy macierz o wymiarach 16x16. W której możemy zauważyć, że w rzędach 2,5,11,12,13 mamy możliwość umieszczenia swoich informacji.

Tabela 9. Macierz kombinacji nadmiarowych bitów nagłówka IP.

Rząd	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
2	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
5	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0
11	1	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Pokazane powyżej metody wykorzystania pól nie są jedynymi stosowanymi podczas ukrywania informacji, podobnie możemy wykorzystać pola obowiązkowe nagłówków TCP i IP w sieci możemy znaleźć wiele narzędzi, które wykorzystują tą lukę w nagłówkach. Jednym z nich jest program convert_tcp, który wykorzystuje pola identyfikacji w datagramie IP oraz pole numeru sekwencyjnego nagłówka TCP. Program ten działa po systemem Linux, po ściągnięciu go należy go skompilować, robimy to za pomocą podstawowym poleceniem w konsoli Linuxa: *$ cc -c covert_tcp \ convert_tcp.c*. Kiedy mamy już skompilowany program możemy zmodyfikować pole identyfikacyjne nagłówka IP.

Manipulacja tym polem polega na zmianie wartości ASCII interesującego nas znaku, możemy to wykonać przy pomocy poleceń:

$ convert_tcp -source 215.258.58.14 -dest 246.28.147.12 -file dane.txt

odbiorca na swoim sprzęcie musi mieć także uruchomiony program convert_tcp, który pracował będzie jako serwer. Program w trybie serwera uruchamiany jest poleceniem:

$ convert_tcp -source 215.258.58.14 -server -file test.txt

Drugie pole wykorzystywane przez ten program to numer sekwencyjny zawarte jest ono w nagłówku protokołu TCP. Pole służy zapewnieniu niezawodności połączenia transmisji TCP/IP i wykorzystywany jest przy nawiązywaniu połączenie w trybie *3-way-handshake*, który opisany jest we wcześniejszym rozdziale. Jak już wiemy uzgodnienie połączenia przebiega w trzech etapach:

- aplikacja klienta przesyła segment danych synchronizacyjnych (SYN – *ang.synchronize*), które zawierają początkowy numer danych. W segmencie SYN przeważnie przesyłane jest tylko nagłówek IP, nagłówek TCP i ewentualne opcje TCP,
- w kolejnym etapie serwer potwierdza przyjęcie danych synchronizacyjnych SYN od klienta i przesyła własne dane synchronizacyjne SYN. Dane te zawierają kolejny numer powiększony o 1 w stosunku do numeru otrzymanego przez klienta. Dodatkową informacją wysyłaną przez serwer jest potwierdzenie (ACK – *ang. acknowledgment*),
- ostatni etap polega na potwierdzeniu przez klienta otrzymania danych synchronizacyjnych przesłanych przez serwer.

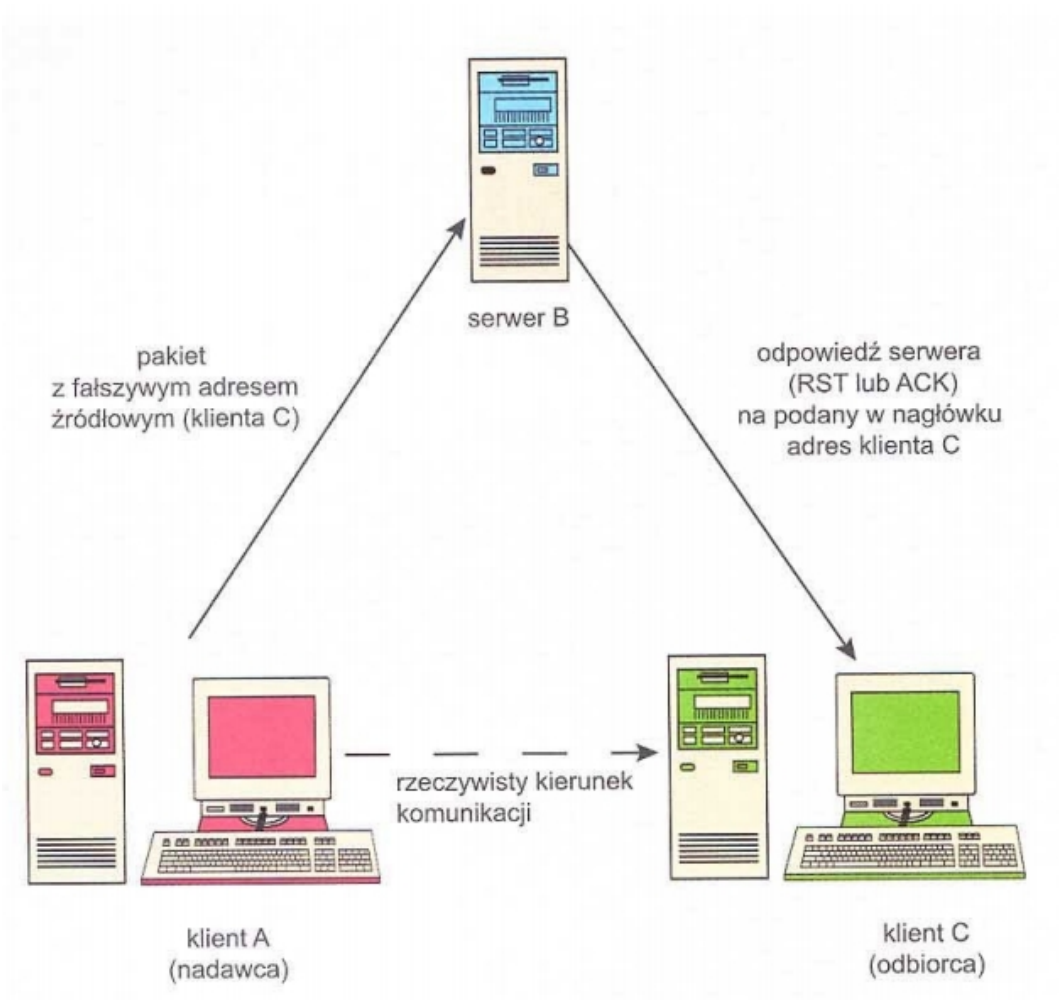
Tak samo jak w poprzednim przypadku mamy możliwość podmiany wartości numeru danych na wartość ASCII. Polecenia są podobne jak w powyższym przykładzie i wyglądają one następująco, odpowiednio dla klienta i serwera:

$ convert_tcp /-source_port 198.168.1.10 -dest 192.168.1.10 -source_port 20
-dest_port 20 -seq -file dane.txt
$ convert_tcp /-source_port 198.168.1.10 -server -seq -file test.txt

Kolejnym sposobem na przesłanie ukrytych informacji za pomocą TCP jest wykorzystanie tzw. odbicia (*ang.bounce*) numeru potwierdzenia. Metoda ta polega na tym, że nadawca najpierw wysyła pakiet w którym znajdują się następujące dane:

- fałszywy adres IP źródła,
- fałszywy numer portu źródłowego,
- fałszywy adres IP odbiorcy,
- fałszywy numer portu odbiorcy,
- segment synchronizacyjny SYN z zakodowane dane.

Schemat działania tej metody przedstawia rysunek 25.



Rysunek 25. Schemat działania metody odbicia.

(źródło: Hakin9 – 2005/03 – Łukasz Wójcicki)

Działanie metody przebiega następująco. Klient A wysyła fałszywe dane z zakodowaną informacją, B odbiera te dane. W danych, które są przesyłane zawarty jest adres serwera C. Serwer B przesyła odpowiedź SYN/RST lub SYN/ACK. Z powodu tego, że zawarty jest w danych fałszywy adres źródłowy, serwer B kieruje odpowiedź z ukrytymi informacjami do serwera C. Odbiorca C po otrzymaniu pakietu dekoduje ukrytą informacje.

Ukryty kanał steganograficzny możemy także stworzyć ręcznie za pomocą dwóch programów SendIP i tcpdump. Program SendIP służy do wysyłania pakietów, a tcpdump umożliwia przechwytywanie ruchu TCP. Przykładowo możemy przesłać wiadomość ukryta w polu identyfikacyjnym datagramu IP. Poszczególne litery przesyłane są w kodzie ASCII. Dla przykładu prześlemy tekst o treści *test*. Kod ASCII dla tej wiadomości będzie wyglądał następując: litera t – 116, e – 101, s – 115 i t – 116. Wiadomość przesyłana jest osobnymi literami. Aby przesłać pierwszą literę wydajemy polecenie:

 # sendip -p ipv4 -li 116 -p tcp -td 80 192.168.1.2

Polecenie to możemy zinterpretować następująco: na początku informujemy, że posługiwać się będziemy protokołami TCP i IP (flagi -p ipv4, -p tcp), następna opcja (-td 80) informuje, że wysyłamy pakiet na port 80 i host o adresie 192.168.1.2. W tym poleceniu najważniejszą wartością jest jednak flaga -li 116, która przesyła literę t w kodzie ASCII. Aby przesłać całą wiadomość musimy powtórzyć to polecenie dla każdej litery z osobna. Osoba odbierająca wiadomość musi posłużyć się programem tcpdump. Odbiorca nasłuchuje na porcie 80 za pomocą polecenia:

 # tcpdump -vvv dst port 80

wiadomość która została ukryta znajduje się w polu id:

16:25:00.491516 192.168.1.1.0 > 192.168.1.2.http:
$ [tcp sum ok] 3843951135:3843951135(0)
win 65535 (ttl 255, id 116, len 40)

taki sposób przesyłania informacji jest uciążliwy i czasochłonny, a ponadto może zostać łatwo wykryty.

Tworząc efektywny kanał steganograficzny na podstawie protokołu TCP/IP musi on spełniać on kilka cech:

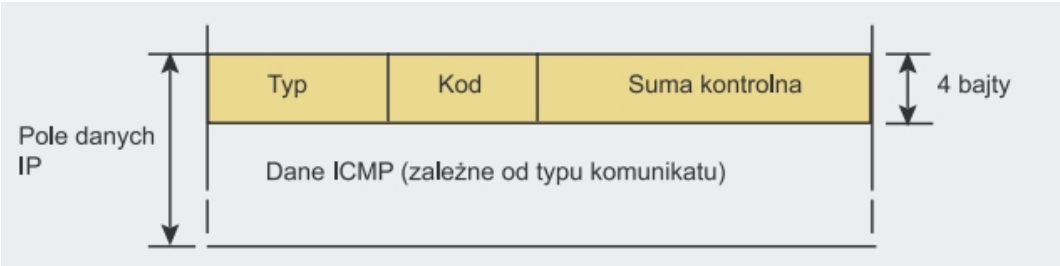
- modyfikacji podlegają pola nagłówków TCP/IP w taki sposób, aby nie wpłynąć na mechanizmy zawarte w protokole TCP/IP, szczególnie na poprawne zachowanie się sesji TCP,
- kanał steganograficzny nie powinien być zdefiniowany w tych polach nagłówka, które zawierają ściśle określone wartości, może to zostać łatwo wykryte,
- powinniśmy tak zmodyfikować pola aby wykrycie informacji było jak najtrudniejsze,
- zastosowanie steganografii w nagłówkach TCP/IP nie powinno powodować wydłużenia czasu generacji pakietów – można a to ławo wykryć,

- przy przesyłaniu dużej ilości informacji musimy uwzględnić dodatkowe narzuty obliczeniowe wynikające z wyznaczania pół nagłówka TCP/IP,
- utworzony kanał steganograficzny musi dodatkowo zapewniać bezpieczeństwo przesyłanych informacji, np: przez szyfrowanie.

3.5 Protokół komunikatów sterujących ICMP w nagłówkach TCP/IP.

Internetowy protokół komunikatów sterujących (ICMP – *ang. Internet control Message Protocol*) jest uznawany za protokół bezpieczny, jedna w połączeniu z protokołem IP może stworzyć ukryty kanał informacyjny. Zadaniem tego protokołu jest dostarczanie komunikatów informujących o tymczasowych błędach w komunikacji. ICMP wykorzystywany jest na przykład w sytuacji, kiedy wymagana jest fragmentacja pakietu IP, a jednocześnie ustawiona jest flaga DF. Ogólnie przyjmując protokół ten służy do zgłaszania błędów komunikacji i pozyskiwania informacji na temat sieci.

Komunikaty protokołu ICMP są kapsułkowane w pakietach IP podobnie jak ma to miejsce w protokołach transportowych wykorzystujących czwartą warstwę modelu OSI czyli w TCP i UDP. Jednak ICMP należy do warstwy sieciowej czyli tak samo jak protokół IP. Protokół ICMP jest integralną częścią protokołu IP, jednak nie wykorzystuje on mechanizmów typu klient-serwer, numerów portów co powoduje, że nie gwarantuje on dostarczenia komunikatu. Polami, które są najważniejsze w pakiecie ICMP są: typ wiadomości i kod wiadomości (dwa pierwsze bajty nagłówka ICMP). Widoczne jest to na rysunku 26.



Rysunek 26. Format komunikatu ICMP.

(źródło: Hakin9 – 2006/02 – Antonio Merola)

Pakiety IP są kapsułkowane w warstwie fizycznej sieci, więc długość pakietu jest ograniczona długością ramki dla danego nośnika tzw. MTU. Pakiety, które są większe od ustalonej wartości MTU podlegają defragmentacji. Jest możliwość zablokowania fragmentacji pakietu

IP poprzez ustawienie flag DF w nagłówku. Jeśli router otrzyma pakiet, który jest zbyt duży oraz ma ustawiony znacznik DF, pakiet wtedy jest odrzucany, a nadawca pakietu otrzymuje komunikat ICMP typu 3 (destination unreachable) z kodem 4 czyli konieczna jest defragmentacja pakietu, ale ustawiony jest znacznik DF. Sekcja danych nagłówka ICMP może posłużyć zatem do utworzenia tajnego kanału informacyjnego. Przykładem wykorzystania nagłówka ICMP jest projekt LOKI.

Wnioski.

Protokół TCP/IP jest najbardziej rozpowszechnionym protokołem komunikacyjnym we współczesnych sieciach. Jednak został on opracowany już w latach siedemdziesiątych co powoduje, że protokoły TCP/IP posiadają wiele wad, które można wykorzystać we własnych celach. Wykorzystanie wszelkich takich błędów może nieść ze sobą poważne zagrożenia. Przykładem może tu być wyciek nie jawnych informacji, a które mogą zagrozić bezpieczeństwu czy to firmy, czy państwa. Zapobieganie takim wyciekom jest dość trudne i czasochłonne. Zależy w głównej mierze od zastosowanych metod ukrywania informacji. Możliwe jest prowadzanie analizy przesyłanych datagramów oraz wykrywanie wszelkich anomalii. Zastosowanie steganografii w TCP/IP umożliwia przesyłanie tajnych informacji w kanałach, które nie są wykrywane przez wiele filtrów. Bardzo często zdarza się, że ukryte kanały komunikacyjne tworzone są w takich polach nagłówkowych protokołów TCP i IP, że mechanizmy wykrywające są praktycznie bezbronne. Jednak ze względu na zwiększający się ciągle przepływ danych wykrywanie wszelkich prób przesyłania poufnych informacji staje się utrudnione.

W pracy przedstawiono możliwości i analizę wykorzystania nagłówków najpopularniejszych protokołów komunikacyjnych czyli TCP i IP do przesyłania tajnych informacji. Wykorzystanie kilku pół nagłówków z datagramów TCP i IP możliwe jest nawet jeśli nie wynika to bezpośrednio ze specyfikacji oraz nie było w zamyśle ich twórców. Jak wynika z przedstawionych faktów do przesyłania ukrytych informacji wykorzystać możemy między innymi: pole identyfikacji nagłówka IP, numer protokołu TCP, pola opcji w TCP i IP, flagi nagłówka IP i inne. Metody ukrywania informacji w protokołach sieciowych ciągle się rozwijają więc wykrywanie ich staje się coraz trudniejsze. Dodatkowe możliwości ukrywania informacji wynikają z dużej ilości istniejących protokołów sieciowych w warstwach wyższych modelu OSI, więc należy przypuszczać, że zostaną one wkrótce wykorzystane szerzej w tworzeniu ukrytych kanałów informacyjnych.

Oprócz ukrywania kanałów steganograficznych w protokołach sieciowych istnieje wiele innych możliwości, które są chętnie wykorzystywane przez użytkowników. Kontenerami stanowiącymi medium transmisyjne dla tajnej informacji mogą być: pliki graficzne, piliki wideo, tekst, plik HTML czy telefonia VOIP.

Bibliografia.

1. Asseco Poland, „Specyfikacja protokołów komunikacyjnych”, kwiecień 2008 http://www.rzoz.gov.pl/download/Specyfikacja_protokolow_komunikacyjnych.pdf 15.12.2010
2. Garbarczuk W., Kopniak P., „Steganologia: współczesne metody ochrony informacji”, Politechnika Lubelska, 2005
3. Hunt C., „TCP/IP. Administracja sieci”, Wydawnictwo RM, 1998
4. Korus P., „Zabawy ze steganografią”, Linux-Magazine, 04/2009
5. Ogłaza K., „Steganografia”, Politechnika Krakowska – Wydział Fizyki, Matematyki i Informatyki Stosowanej, 2009
6. Pacholczyk F., „Steganografia w sieciach IP”, Politechnika Łódzka, 2003
7. Roczniki 2004 – 2010, magazynu Hakin9
8. Rogala M., „Tworzenie ukrytych kanałów informacyjnych w sieciach komputerowych TCP/IP”, Politechnika Wrocławska – Wydział Informatyki i Zarządzania, 2008
9. Scott J., „Diagnozowanie i utrzymanie sieci – Księga eksperta”, Helion, 2000
10. Sellars D., An Introduction To Steganography, http://www.zoklet.net/totse/en/privacy/encryption/163947.html 26.12.2010
11. Szczypiorski K., Lubacz J., Mazurczyk W., „Steganografia sieciowa czyli o wyrafinowanych sposobach ukrytego przekazywania informacji”, Politechnika Warszawska – Instytut Telekomunikacji, 2009
12. Szczypiorski K., „Steganografia w bezprzewodowych sieciach lokalnych”rozprawa doktorska, Politechnika Warszawska – Wydział Elektroniki i Technik Informacyjnych, 2006
13. Tomaszewski M., „Analiza algorytmów ukrywania w dźwięku”, Praca Dyplomowa, Politechnika Białostocka, 2006
14. U. S. Department Of Defense, „Trusted Computer System Evaluation Criteria”, 1985
15. Szewc Ł., Struzik P., Sokołowski D., „Model warstwowy OSI i jego znaczenie w technologii sieci komputerowych" http://m6.mech.pk.edu.pl/~habel/dydaktyka/Kk/511e/t2/index.html 22.12.2010
16. Wasak M. „TCP/IP” http://majusek.fm.interia.pl/ 01.11.2010
17. Groele J., Groele R., „TCP/IP a model OSI” http://www.staff.amu.edu.pl/~psi/informatyka/tcpip/osi_tcp.htm 15.11.2010
18. Sikora A., „Przegląd zagadnień kryptografii” 10.12.2010 http://www.pcworld.pl/artykuly/35081_0_1/Przeglad.zagadnien.kryptografii.html

Spis tabel.

Spis rysunków.