УНИВЕРЗИТЕТ „ГОЦЕ ДЕЛЧЕВ” – ШТИП
ФАКУЛТЕТ ЗА ИНФОРМАТИКА

Борис Панајотов

КРИЕЊЕ НА ПОДАТОЦИ ВО МРЕЖНИТЕ ПРОТОКОЛИ

- МАГИСТЕРСКИ ТРУД -

Штип, декември 2013 г.

Комисија за оценка и одбрана

Ментор: д-р Александра Милева, доцент
Факултет за информатика
Универзитет „Гоце Делчев“– Штип

Член: д-р Цвета Мартиновска - Банде, вонреден професор
Факултет за информатика
Универзитет „Гоце Делчев“– Штип

Член: д-р Игор Стојановиќ, доцент
Факултет за информатика
Универзитет „Гоце Делчев“– Штип

Членови на Комисија за оценка и одбрана

Претседател: д-р Игор Стојановиќ, доцент
Факултет за информатика
Универзитет „Гоце Делчев“– Штип

Член: д-р Цвета Мартиновска - Банде, вонреден професор
Факултет за информатика
Универзитет „Гоце Делчев“– Штип

Член: д-р Александра Милева, доцент
Факултет за информатика
Универзитет „Гоце Делчев“ – Штип

Научно поле: Компјутерска техника и информатика
Научна област: Информациони системи и мрежи

Датум на одбрана: 18.12.2013

НАУЧНИ ТРУДОВИ ОБЈАВЕНИ НА МЕЃУНАРОДНИ КОНФЕРЕНЦИИ

1. Panajotov, B., & Mileva, A. (2013). Covert Channels in TCP/IP Protocol Stack. ICT Innovations Web Proceedings (стр. 190-199).

Содржина

Краток извадок

Во овој магистерски труд е направена анализа на техниките за креирање и имплементирање на скриени канали во TCP/IP складот со протоколи, класифицирани според засегнатите нивоа и протоколи. Голем дел од протоколите од TCP/IP складот се подложни на злоупотреби со стеганографија. Во анализата како мерки се користени вкупниот број на стеганографски битови пренесени за една секунда (***Raw Bit Rate - RBR***) и вкупниот број на стеганографски битови пренесени по податочна единица на протокол (Protocol Data Unit - PDU), наречена ***Packet Raw Bit Rate -PRBR***. Разбирањето на техниките на скриени канали во мрежните протоколи е основен предуслов за креирање на противмерки за детектирање, елиминирање и ограничување на скриените канали.

Клучни зборови: *скриени канали, стеганографија, заглавје*

Abstract

In this master thesis is given an analysis of techniques for creating and deploying hidden channels in TCP / IP protocols stack, classified according to the affected levels and protocols. Many of the protocols in TCP / IP stack are susceptible to to steganography abuse. For analyzing are used measures like total number of steganography bits transmitted per second (Raw Bit Rate - RBR) and the total number of steganography bits transmitted per protocol data unit (Protocol Data Unit - PDU), called Packet Raw Bit Rate-PRBR. Understanding the techniques of hidden channels in network protocols is essential for creating countermeasures for detecting, eliminating and restricting the hidden channels.

Key Words: *covert channel, steganography, header*

Вовед

Компјутерските мрежи и Интернетот станаа секојдневие во нашиот живот, нешто без што не можеме да замислиме ниту еден ден. Тие го обликуваат нашето слободно и неслободно време, постојано го менуваат и подобруваат квалитетот на живеење и начинот на мислење и правење на нештата, но и преставуваат тивка закана за нашата приватност, за безбедоста на нашите податоци и нашиот идентитет. TCP/IP складот со протоколи е темелот без кој не би можеле денес да сурфаме, да испраќаме и примаме електронска пошта, да читаме електронски книги и весници, да гледаме онлајн видео и аудио записи и сл. Тој е резултат на истражувањата и развојот на протоколите извршени врз експерименталната мрежа со комутација на пакети ARPANET, основана од страна на Агенцијата за напредни истражувачки проекти за одбраната (Defence Advanced Reserch Projects Agency DARPA). Се состои од голема колекција на протоколи кои се издадени како интернет-стандарди од страна на Одборот за интернет-активности (Internet Activities Board - IAB). Бидејќи нема предефиниран модел за TCP/IP складот со протоколи, неговиот работен модел се состои од пет нивоа, и тоа: апликациско ниво, транспортно ниво, Интернет ниво, ниво на мрежен пристап и физичко ниво.

Првобитно наменет за поврзување на универзитетите во САД, Интернетот не е дизајниран од почеток со безбедноста во мислите. TCP/IP складот со протоколи има многу безбедносни пропусти кои постојано се откриваат и соодветно се коригираат. Во годините што поминаа се појавија дистрибуирани напади со одбивање на сервиси (Distributed Denial of Service – DDoS), напад со затровување на DNS кешот, напади со лажирање на изворна адреса, напади врз упатувањето, напади со предвидување на редоследни броеви, напади на автентикацијата и др. (Bellovin, 1989). Се појавија и безбедносни решенија и протоколи како DNSSEC, IPSec, SSL/TLS, SSH и други.

Сепак, покрај вообичаените напади на компјутерските мрежи, постои и уште еден голем безбедносен ризик – користење на стеганографски методи во мрежните протоколи, со помош на кои, покрај нормалната комуникација, ќе се пренесува и скриена комуникација. Стеганографијата во најопшт случај е наука

која ги проучува сите методи за вгнездување на дополнителни скриени содржини во некаков вид на носач, со цел криење на направената промена. Носач може да биде било што, дел од телото на човекот, парче хартија со невидливо мастило, слика, аудио, видео, заглавје на протокол и сл. Модерната стеганографија, позната и како дигитална стеганографија, може да се подели на 4 гранки: стеганографија на дигиталните медиуми, лингвистичка стеганографија, стеганографија на податочни системи и мрежна стеганографија (Zielinska, Mazurczyk, & Szczypiorski , 2013). Скоро сите протоколи од TCP/IP складот со протоколи се подложни на злоупотреби со техники на мрежната стеганографија. Најчесто причина за тоа се редундантните полиња (полиња за падирање, резервирани, недефинирани полиња и сл.) и полињата со случајни податоци во заглавјата на протоколите.

Целта на овој магистерски труд е да даде преглед и анализа на постоечките методи и техники за креирање и имплементација на скриени канали во TCP/IP складот со протоколи, класифицирани според засегнатите нивоа и протоколи. Притоа, освен тунелите во корисен товар, не се опфатени другите техники кои кријат тајна порака во корисниот товар на мрежните протоколи. Не се опфатени ниту чистите прикриени (subliminal) канали во криптосистемите кои може да се користат кај мрежните протоколи. Добар преглед за мрежните скриени канали од 1987 до 2006 год. е даден во (Zander, Armitage, & Branch, 2007), каде што скриените канали се разгледувани според техниките кои се користени за нивно креирање. Може да се погледнат и прегледите дадени во (Llamas, Allison, & Miller, 2005), (Scott, 2008), (Allix, 2007), (Panajotov & Mileva, 2013). Повеќето од скриените канали имаат и своја имплементација, а еден преглед на имплементации на скриени канали е даден во (Smith, 2000). Слична анализа како во овој магистерски труд, но врз неколку имплементации на мрежни скриени канали е дадена во извештајот (Smeets & Koot, 2006).

Разбирањето на техниките на скриени канали во мрежните протоколи е основен предуслов за креирање на противмерки за детектирање, елиминирање и ограничување на скриените канали.

Во првото поглавје се дадени основите на мрежната стеганографија и скриените канали, особините на стеганографските системи, разликите меѓу стеганографија, водени печати и криење на информации (податоци), како и мерките со кои може да се анализираат различните мрежни стеганографски техники. Второто поглавје е посветено на различните стеганографски техники кои се користат за протоколите од Интернет нивото на TCP/IP складот со протоколи, и тоа IPv4, IPv6, ICMP, IGMP, DHCP и ARP. Краток опис на секој протокол, посебно на неговите заглавја, е следено со соодветните стеганографски техники. Третото поглавје е посветено на различните стеганографски техники кои се користат за протоколите TCP и UDP од транспортното ниво. Стеганографските техники за протоколите од апликациско ниво, и тоа: HTTP, FTP, DNS, RTP, SIP се дадени во четвртото поглавје. Петтото поглавје е посветено на различните механизми и техники кои се користат за заштита од скриените канали во мрежните протоколи.

1. Мрежна стеганографија и скриени канали

Отвореният или *јавен (overt) канал* е комуникациски канал во рамките на компјутерски систем или мрежа, наменет за авторизиран пренос на податоци. Од друга страна, *скриен (covert) канал* е кој било комуникациски канал што може да биде искористен од даден процес за пренос на информации на начин кој ги прекршува безбедносните политики на даден систем, дозволувајќи им на информациите да протекуваат до неовластен или непознат приемник, во рамките на легитимна мрежа комуникација (Department of Defence:, 1985). Концептот на скриен канал најпрво бил воведен од страна на Лампсон (Lampson, 1973) во контекст на монолитските системи како механизам со кој од процес на повисоко безбедносно ниво протекуваат информации на процес со пониско безбедносно ниво, кој инаку нема право на пристап на тие информации.

Кој било делен ресурс може потенцијално да биде искористен како скриен канал. Скриените канали најчесто немаат конкретна дефиниција и се ориентирани според дадено сценарио. Ова е затоа што таквите канали можат да постојат помеѓу процеси во оперативниот систем или помеѓу дистрибуирани објекти (Anjan & Abraham, Behavioral Analysis of Transport Layer Based, 2010). Уникатната функционалност на скриените канали е во тоа што тие ја користат легитимната комуникација. Скриените канали не мора да бидат помеѓу две крајни точки, бидејќи можат да дејствуваат на средина од воспоставената врска. *Прикриениот (subliminal) канал* е една од формите на скриените канали, кој се фокусира само на протекувањето на информации во криптографските протоколи. Каналот кој е составен од едноставен скриен канал и прикриен канал се нарекува *хибриден скриен канал*.

Скриените канали првенствено можат да се поделат на *складирачки (storage)* и *временски (timing)* канали. Во случај на складирачките канали, обично еден процес запишува (директно или индиректно) на заеднички ресурс, додека друг процес чита од него. Временскиот канал во суштина е која било техника што пренесува информации со помош на времето на почнување или завршување на дадени настани, па затоа на процесот што ги прима информациите му е потребен

часовник. Посебна потткласа на временски скриен канал е *скриен канал со бројач (counting covert channels)* кој пренесува податоци со броење на појавата на одредени настани (Gray, 1994). Како и сите други канали за комуникација, скриените канали можат да бидат *со шум (noisy)* или *бесшумни (noiseless)*. Кај каналите со шум може да настанат грешки, како супституции, вментувања, бришења и сл. Според бројот на информациските текови помеѓу испраќачот и примачот, постојат *агрегирани (aggregated)* и *неагрегирани (non-aggregated)* скриени канали (Gallagher, P.R., 1993). Според тоа дали помеѓу двете страни кои кумуницираат има меѓујазол или нема, се разликуваат *индиректни* и *директни* скриени канали.

Посебен тип на скриен канал е *тунел во корисен товар (payload tunnel)* кој тунелира еден протокол во корисниот товар на друг протокол. Овој канал најчесто се користи за заобиколување на заштитните ѕидови кои го лимитираат излезниот сообраќај само на неколку апликации (на пример, само HTTP). Тунелот во корисен товар се разликува од останатите скриени канали по тоа што неговата главна примена е да се заобиколат безбедносните филтри, а не да се скрие постоењето на каналот.

Класификацијата дадена во (Wang & Lee, 2005) ја зема предвид димензијата во која се енкодираат податоците (простор, време), но и парадигмата на енкодирањето (базирано на вредност или на транзиција). Според оваа класификација, скриените канали се делат на:

- *Просторни канали базирани на вредност* – испраќачот може да ги менува вредностите на еден или повеќе објекти, а примачот ја екстрахира информацијата со гледање на вредностите. Одговараат на складирачките скриени канали со директно запишување;
- *Просторни канали базирани на транзиција* – испраќачот одредува дали ќе има промена на еден или повеќе објекти или не, а примачот ја дознава информацијата од тоа дали има промена или не. Покажуваат дека складирачки скриени канали може да се формираат и индиректно без испраќачот да има контрола врз вредноста на даден објект;

- *Временски канали базирани на вредност* – испраќачот може да научи или предвиди вредност на некој објект и има контрола кога примачот може да направи опсервации на објектите. Потоа испраќачот чека да се појави вредноста и примачот да направи опсервација.
- *Временски канали базирани на транзиција* – испраќачот може да го контролира редот на модификациите, релативно на опсервациите направени од страна на примачот. Примачот ја екстрахира информацијата од редоследот на настаните, а не од нивните вредности. Одговараат на повеќето временски скриени канали.

Моделот за злоупотреба кај скриените канали се темели на познатиот проблем на затворениците (Prisoners' Problem), дефиниран од Симонс (Simmons, 1983). Двајца затвореници, Алис и Боб, сакаат да комуницираат доверливо и незабележливо преку небезбеден канал - под надзор на чувар (warden). Чуварот може да биде пасивен – само да го следи сообраќајот или активен – може да ја менува содржината на пораките со цел да спречи секаква форма на скриена комуникација. Ако чуварот смета дека пораката на Алис до Боб е безопасна, едноставно може да ја проследи до Боб. Алтернативно, чуварот може да ја промени содржина или може да ја блокира комуникацијата во целост.

Различните стеганографски методи кои се користат во телекомуникациските мрежи се познати како *мрежна стеганографија*. Работата на Rowland (Rowland, 1997) е прв доказ на концепт (proof of concept) за постоењето и можноста за искористување на скриени канали во TCP/IP складот со протоколи, со нивна конкретна имплементација.

Скриените канали во мрежните протоколи се слични на технологиите за криење информации во аудио, визуелни или текстуални содржини, што е предмет на изучување на науката стеганографија. Стеганографијата има потреба од некоја форма што ќе ја прикрива содржината, па така и скриените канали бараат мрежни протоколи што ќе ги користат за криење и пренос на информациите. Мрежните протоколи се идеални за криење на податоци. Повеќето мрежни стеганографски техники за складирачките канали го користат фактот дека постојат редундантни полиња во заглавјата на протоколите, кои може да бидат искористени за криење

на податоци. Wolf (Wolf, 1989) ги предложил резервираните полиња, полето за падирање и недефинираните полиња во рамките на IEEE 802.2, 3, 4 и 5 мрежите да се користат за таа намена. Слично, Handel и Sandford (Handel & Sandford, 1996) ги предложиле резервираните и неискористени полиња од заглавја на други протоколи за истата намена. Друг тип се полињата кои се полнат со случајни податоци, како на пример IP *полето за идентификација* или *почетниот број на секвенца (ISN)* на TCP. Но нивното наивно полнење со податоци ќе биде откриено од страна на пасивен чувар (Murdoch & Lewis, 2005), бидејќи овие полиња природно имаат доволно структура и неуниформност за да бидат ефикасно и надежно диференцирани од немодифицирана PDU. (Girling, 1987) предложил информацијата да се енкодира со модулирање на должината на рамките на нивото на мрежен пристап.

Од гледна точка на напаѓачот, повеќе се преферираат складирачките канали отколку временските канали, поради проблемите со синхронизацијата кои се присутни во временските канали, нивната комплексност и нивниот значително помал пропусен опсег во споредба со складирачките канали. Мрежните скриени канали може да се користат за координирање на DDoS напади, за ширење на компјутерски вируси и црви, за тајна комуникација меѓу терористите и криминалците, но исто така и за безбедно управување со мрежната комуникација (Forte, 2005), за заобиколување на заштитниот ѕид (firewall) во организацијата, за пренесување на податоци за автентикација (deGraaf, Aycock, & Jacobson, 2005) како port knocking, заобиколување на ограничувањето за користење на Интернет во некои земји (Feamster, Balazinska, Harfst, Balakrishnan, & Karger, 2003), подобрување на безбедноста кај VoIP (Mazurczyk & Kotulski, New Security and Control Protocol for VoIP Based on Steganography and Digital Watermarking, 2006), подобрување на квалитетот на сервис кај VoIP (Mazurczyk & Kotulski, 2006) итн. (Jones, Le Moigne,, & Robert, 2004) предлагаат користење на скриен канал во TTL полето IPv4, за следење на назад на IP пакетите без изворна адреса, што е потребно на пример во случај на DoS напад. Повеќето од складирачките скриени канали ефикасно можат да се отстранат преку нормализатори на сообраќај (Fisk, Fisk, Papadopoulos, & Neil, 2002) и (Handly & Paxson, 2001), кои ги менуваат

дојдовните и појдовните пакети со стандардизирање на полињата кои се неискористени или непотребни.

Според (Cox, Miller, Bloom, Fridrich, & Kalker) методите на мрежната стеганографија може да се класифицира во три големи групи:

- методи кои вршат модификација на мрежните PDU;
- методи кои вршат модификација на структурата на протокот со пакети;
- хибридни методи.

Во првата група спаѓаат методите кои ги ги модифицираат полињата од заглавјата на мрежните протоколи, податоците во PDU или нивна комбинација.

Во втората група спаѓаат методите кои имаат влијание на редоследот на пакетите, на доцнењето на пакетите и методите кои воведуваат намерни загуби на пакети со прескокнување на редоследните броеви на пакетите.

Примери на хибридни методи се: LACK (Lost Audio paCKets steganography) (Mazurczyk & Lubacz, LACK—a VoIP steganographic method, 2010), и RSTEG (Retransmission Steganography) (Mazurczyk, Smolarczyk, & Szczypiorski, RSTEG: Retransmission Steganography and Its Detection, 2010), (Mazurczyk, Smolarczyk, & Szczypiorski, Retransmission Steganography Applied, 2010), за кои понатаму ќе се зборува повеќе.

Некои техники се независни од протоколот, како што е техниката на Перкинс (Perkins, 2005) за создавање на скриени канали со користење на збирот од сите битови во пораката. Наједноставниот облик е кога пораки со збир на сите битови под некоја вредност - праг се користат за испраќање на бинарна 0, а над вредноста – праг, за испраќање на бинарна 1. Оваа метода може да се генерализира, така што на почетокот испраќачот и примачот треба да се договорат за максималната можна сума S и бројот на интервали N_i во [0; S] и ако збирот се падне во одреден интервал, тоа ќе одговара на пренос на одредена група на битови. На пример, ако се дадени четири интервали, тие ќе одговараат на пренос на група од 2 бита – 00, 01, 10 и 11, соодветно. Овој канал може да испрати $log_2 N_i$ битови по пакет.

Постојат и техники кои користат стеганографија меѓу неколку протоколи за тајна комуникација, како што е PadSteg (Jankowski, Mazurczyk, & Szczypiorski,

PadSteg: Introducing Inter-Protocol Steganography, 2013), кој ги користи ARP и TCP протоколите заедно со Etherleak ранливоста (неправилно падирање на Ethernet рамка) во рамките на една локална мрежа. Во оваа група спаѓа и класификацијата на пакети предложена од (Dong, Qian, Lu, & Lan, 2012), во која се избира IP пакетот како носач и се избираат неколку полиња од различните протоколи во пакетот (од Интернет, транспортно и/или апликациско ниво) или некои временски карактеристики како особини, потоа пораката се мапира во сортирачки вектор, а тој се мапира во соодветните особини. Ако се изберат t особини, се добива дека може во општ случај да се пренесуваат $log_2 t$ битови на пакет.

Техниките на мрежна стеганографија денес се применуваат и кај апликациите и протоколите кои користат P2P мрежи, како SkyDe (Mazurczyk, Karas, & Szczypiorski, SkyDe: a Skype-based Steganographic Method, 2013), апликација за криење на податоци во Skype.

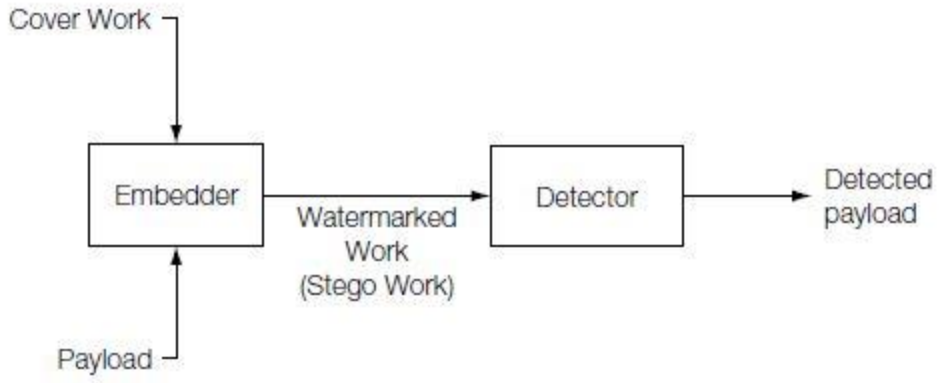
1.1. Карактеристики на стеганографски системи

Стеганографскиот систем (како и системот за водени печати) се состои од *вметнувач (embedder)* и *детектор (detector)*, како што е илустрирано на слика 1. Вметнувачот зема два влеза. Едниот е товарот што сакаме да го вметнеме (водениот печат или тајната порака), а другата е содржината за криење (Cover Work) во која сакаме да го вметнеме товарот. Излезот на вметнувачот се пренесува или снима и се нарекува *стеганограм*. Подоцна, содржината (или некоја друга содржина која не поминала преку вметнувачот) е претставена како влез за детекторот. Повеќето детектори се обидуваат да утврдат дали е присутен товарот и ако е така да ја отпечатат пораката кодирана во него.

Водените печати (watermarking) ги дефинираме како практика на незабележливо менување на дадена содржина, со цел вметнување на порака за таа содржина.

Стеганографијата (steganography) ја дефинираме како практика на неоткривливо менување на содржината, со цел вметнување во тајна порака. Криењето на информации, стеганографијата и водените печати се три тесно поврзани области во кои има голема доза на преклопување и споделување на

многу технички пристапи. Сепак, постојат фундаментални разлики кои влијаат на барањата, а со тоа и на дизајнот на техничкото решение.



Слика 1. Генерички систем за водени печати и стеганографија
Figure 1. A generic watermarking and steganography system (Ingemar J. Cox, 2008)

Криењето на информации (или криење на податоци) е општ термин кој опфаќа широк спектар на проблеми, кои го вклучуваат и вметнувањето на пораки во содржината, како на пример одржување на анонимност при користење на мрежата (Kesdogan, Egner, & Roland, 1998), чување дел од базата на податоци скриена од неавторизирани корисници (Moskowitz & Chang, 1999) и др. Терминот „се крие тука" може да се однесува или на изработката на незабележливи информации (како во водените печати) или за криење на постоењето на тајната информација.

Системите за вметнување на пораки во содржината може да се поделат на: системи со водени печати, во кои пораката е поврзана со скриената содржина и системи без водени печати, во кои пораката не е поврзана со скриената содржина. Тие, исто така, може да бидат независно поделени на системи со стеганографија, во кои постоењето на пораката се чува во тајност и системи без стеганографија, во кои постоењето на пораката не треба да биде тајна. Ова резултира во четири категории на системи за криење на информации, кои се сумирани во табела 1, и тоа:

1. Скриени водени печати (Covert watermarking) - водените печати кодираат информации поврзани со примателот на секоја копија од документите, никој

не знае за нивното постоење, па така изворот на протекување би можел да биде идентификуван.

Табела 1. Четири категории за криење на информации. CW се однесува на скриена содржина
Table 1. Four categories of information hiding. CW refers to cover Work (Ingemar J. Cox, 2008)

	Порака зависна од CW	**Порака независна од CW**
Непознато постоење	Covert Watermarking	Steganography (Covert Communication)
Познато постоење	Overt Watermarking	Overt Embedded Communication

2. Стеганографија (Steganography) - пораки скриени во спротивно безопасен пренос. Ова е делот на кој ќе се задржиме понатаму.
3. Отворени водени печати (Overt watermarking) - познато е присуството на воден печат.
4. Отворена вметната комуникација (Overt, embedded communication) - се однесува на познат пренос на помошни, скриени информации што не се поврзани со сигналот во кој се вградени.

Примарната цел на стеганографијата е криење на фактот дека скриената комуникација е присутна во рамките на една безопасна комуникација. Постои и посебна наука наречена *стегоанализата (steganalysis),* чијашто примарна цел е да открие кога се одвива скриената комуникација. За разлика од стеганографијата, при примена на водените печати, нивното постоење во дадена содржина најчесто е познато, но сепак незабележливо за човековите сетила. Вушност, ова дури е пожелно за да се спречи нелегалното користење на содржината. Додека алгоритмите за стеганографија и дигиталните водени печати можат да бидат изградени на заедничка основа на принципи за криење на податоци, својства на системите за стеганографија и стегоанализа се сосема различни од оние на дигиталните водени печати.

Во продолжение се дадени некои карактеристики на стеганографските системи.

Стеганографски капацитет, капацитет на вметнување и ефикасност на вметнување

Капацитетот на вметнување (embedding capacity) е максималниот број на битови кои можат да бидат скриени во дадена скриена содржина. На пример, ако вметнеме скриена порака со менување на најмалку значајниот бит (LSB) на сиви слики, капацитетот на вметнување (во битови) е бројот на пиксели во сликата. *Стеганографскиот капацитет (steganographic capacity)* е максималниот број на битови кои можат да бидат скриени во дадена скриена содржина, така што веројатноста за откривање од страна на противникот е занемарлива. Поради ова стеганографскиот капацитет веројатно ќе биде многу помал од капацитетот на вметнување. Утврдувањето на стеганографскиот капацитет е многу тешка задача дури и за наједноставните шеми за вградување. Повеќето стеганографски шеми може да избегнат да бидат откриени од страна на моменталните стегоанализи со намалување на количината на информации вградени во скриената содржина. Алтернативно, намалувањето на бројот на вметнувачки промени за вметнување на истиот товар го намалува влијанието на вметнувањето и на тој начин води кон побезбедна шема.

Практичните стеганографски шеми треба да имаат употреблив стеганографски капацитет. Додека робустен, воден печат со капацитет од 1 бит, може да биде корисен за апликациите, но стеганографската шема која може да вметнува само 1 бит во сликата не е практична. Примарната цел на новите стеганографски алгоритми е развивање на статистички неоткривливи методи со голем стеганографски капацитет. Важен концепт во стеганографијата е *ефикасноста на вметнувањето (embedding efficiency),* која се дефинира како број на битови од скриената порака кои се вметнати по единица дисторзија (unit distortion). Ако влијанието на сите вметнати модификации е приближно исто, ефикасноста на вметнувањето може да се измери како број на битови од пораката кои се вметнуваат со една вметната промена.

Raw Bit Rate и Packet Raw Bit Rate

Практично, полесно е наместо стеганографски капацитет и капацитет на вметнување да се користи вкупниот број на стеганографски битови пренесени за

една секунда (***Raw Bit Rate - RBR***), а ефикасноста на вметнување да се изрази со вкупниот број на стеганографски битови пренесени по податочна единица на протокол (Protocol Data Unit - PDU), наречена ***Packet Raw Bit Rate -PRBR***. Токму овие мерки се користени при анализата на различните стеганографски методи во TCP/IP складот со протоколи.

2. Интернет ниво

Интернет нивото е одговорно за упатување на пакетите од даден извор до дадена дестинација, преку упатувачи. За таа цел, ова ниво треба да ја познава топологијата на мрежата и да избере соодветен пат преку неа, дури и за големи мрежи. Притоа треба да внимава да ги избегне преоптоварените линии, за да не предизвика застој. На ова ниво од TCP/IP складот со протоколи функционираат повеќе протоколи, како: IP, ICMP, IGMP, ARP, RARP, DHCP и други. Во ова поглавје ќе биде направен преглед на техниките за скриени канали на Интернет ниво, според протоколите на кои се применуваат.

2.1. IP

Интернет протоколот (Internet Protocol - IP) е ненадежен (unreliable) протокол без воспоставување на конекција (connectionless) кој го користи мрежното ниво. Ова значи дека пред да се испратат пакетите не се воспоставува логичка конекција и дека не постои гаранција дека IP пакетите ќе пристигнат до зададената дестинација и нема да бидат изгубени. Интернет протоколот не ја одржува состојбата и информациите помеѓу успешно емитираните пакети. Единствената улогата на Интернет протоколот е да пренесува пакети од една на друга машина. Секој пакет содржи заглавје со контролни информации и корисен товар. Денес во употреба се две верзии на Интернет протоколот: IPv4 и IPv6.

2.1.1. IPv4

IPv4 е најпознатата и најкористената верзија на овој протокол. Дефиниран е во 1981 година во RFC 791 (Information Sciences Institute, 1981). IPv4 користи 32-

битни (четирибајтни) адреси, што го ограничува просторот за адреси на 2^{32} адреси. На табела 2 е прикажано заглавјето на IPv4.

Полето *верзија (Version)* (4 бита) ја дава верзијата на протоколот кој се користи. За оваа верзија, ова поле ќе има вредност 4.

Полето *должина на интернет заглавјето (IHL)* (4 бита) е бројот на 32-битни зборови во заглавјето, заедно со понудените опции. Минималната вредност е пет, што дава минимална должина на заглавјето од 20 октети.

Табела 2. Заглавје на IPv4
Table 2. IPv4 header

DS = поле за идентификациони сервиси
ECN = поле за екплицитна објава на застој

Полињата *DS/ECN* (8 бита) пред воведувањето на диференцираните сервиси биле познати како поле за *тип на сервис*. Првите 6-бита се користат за диференцирани сервиси (Differentiated Services-DS), а останатите 2-бита се резервирани за полето за експлицитно известување на застој (Explicit Congestion Notification-ECN). Полето ECN обезбедува експлицитна сигнализација за појава на застој на сличен начин со frame relay.

Полето за *вкупна должина (Total Length)* (16 бита) ја дава вкупната должина на пакетот, вклучувајќи ги заглавјето и податоците, изразена во октети.

Полето за *идентификација (Identification)* (16 бита) е редоследниот број кој заедно со изворната адреса, одредишната адреса и корисничкиот протокол уникатно го идентификува датаграмот. Исто така, се користи при фрагментација. MTU (максимална преносна единица) е максималната големина на пакетот што мрежата може да ја прифати. Кога пакетот е поголем од MTU, треба да се подели (фрагментира) на неколку помали пакети. Секој од нив содржи поле за идентификација што се користи од страна на приемникот за да го состави оригиналниот пакет, секој фрагментиран пакет од еден оригинален пакет го има истиот број за идентификација.

Полето *знаменца (Flags)* (3 бита) содржи три знаменца, секое кодирано со еден бит, од кои се користат само второто и третото знаменце:

• Доколку DF (не фрагментирај) е поставено на 1, пакетот не смее да се фрагментира при преносот. Ако овој бит е поставен на 1, а големината на пакетот ја надминува големината на MTU на некоја попатна мрежа, пакетот ќе биде отфрлен.

• MF (повеќе фрагменти) се користи за фрагментација и повторно составување и е поставен на 1 за секој фрагмент од еден фрагментиран пакет, само не на последниот фрагмент.

Полето *поместеност на фрагмент (Fragment Offset)* (13 бита) укажува на позицијата на еден фрагмент во оригиналниот пакет, изразено во единици од 64 бита.

Полето *време на живот (TTL),* (8 бита) одредува колку долго му се дозволува на датаграмот да остане во интернет-мрежата, изразено во секунди. Упатувачот може да го намали TTL за еден така што TTL полето е слично со бројачот на скокови. Кога TTL е нула, пакетот се отфрла.

Полето за *протокол (Protocol)* (8 бита) го идентификува типот на следното заглавје во пакетот, кое следи по IP заглавјето. Укажува кој вид на протокол ги користи услугите на IP. Броевите се предефинирани, на пример, 1 е за ICMP, 6 за TCP, 7 за UDP итн.

Полето *сума за проверка на заглавјето* (*Header Checksum*) (16 бита) се користи за откривање на грешки при преносот во заглавјето. Оваа сума се проверува и повторно се пресметува кај секој упатувач, бидејќи TTL полето се менува на секое прескокнување. Податоците не се дел од проверката, додека протоколите на транспортното ниво може да имаат свои полиња за проверка. Алгоритамот за проверка е едноставен: „Сумата за проверка се формира со земање на прв комплемент од збирот со 16-битен прв комплемент на сите 16-битни зборови во заглавјето. За оваа пресметка полето за сума за проверка се иницијализира на вредност нула".

Полето *изворна адреса (Source Address)* (32 бита) е адресата на изворната машината што ги емитира пакетите.

Полето *одредишна адреса (Destination Address)* (32 бита) е адресата на дестинацијата.

Полето за *опции (Options)* (променлива должина) е дополнително поле. Може да содржи информации за безбедноста, рутирањето итн. Ги содржи опциите побарани од корисникот кој ги испраќа податоците.

Полето *дополнување* (*Padding*) (променлива должина) се користи за да се потврди дека должината на заглавјето на датаграмот е целоброен производ на 32 бита.

Полето за *податоци* (променлива должина) мора да биде со должина од целоброен производ на 8 бита. Максималната должина на датаграмот (податочно поле + заглавје) изнесува 65535 октети. Ги содржи податоците од горните слоеви.

Скриени канали во IPv4

Складирачки скриени канали

Една група на стеганографски техники за IPv4 ги користи полињата од IPv4 заглавјето кои имаат некаква редундантност или не се користат при преносот, како *идентификација, знаменца, поместеност на фрагмент* и *опции.* (Rowland, 1997) го користи 16-битното поле *идентификација* за криење на податоци и во него ја сместува ASCII вредноста на знаците помножена со 256. Иако *PRBR* за овој канал е 16, Rowland во својата имплементација користи само 8 бита, а

останатите ги поставува на нула. Полето *идентификација* се користи само при фрагментација и треба да се постави на нула кога не е во употреба. Сепак дестинацијата прифаќа и нефрагментирани пакети со ненултни вредности за ова поле. Ако при преносот на пакетот настане фрагментација, примателот ќе ја добие истата информација од секој фрагмент.

Друга група на стеганографски техники ја користат редундантноста на стратегијата на фрагментирање (Ahsan, Covert channel analysis and data hiding in TCP/IP, 2002), (Ahsan & Kundur, Practical data hiding in TCP/IP, 2002). Кај негфрагментиран пакет сите полиња за фрагментација се нула (*MF*=0, *DF*=0, *поместеност на фрагмент* =0). Ако страните кои комуницираат го знаат MTU во нивната мрежа, тие можат да го користат *DF* битот за испраќање на 1-бит на податоци по пакет или комбинација на *DF* битот и *идентификација* за испраќање на 17-битни податоци по пакет, со испраќање на пакети кои не ја надминуваат големината на MTU. Ако страните кои комуницираат не го знаат MTU, може сè уште да испраќаат 8 бита по пакет, со пополнување на првите 8 бита од *идентификација* полето (останатите 8 се генерираат случајно) со резултатот на XOR-рирање на првите фиксни 8-бита од IPv4 заглавјето и 8 бита податоци. Единствен услов е да нема опции во заглавјето. (Cauich, Gomez, & Watanabe, 2005) ги користат полињата *идентификација* и *поместеност на фрагмент* за поставување на скриените пораки. Нивниот метод обезбедува 29 битови во секој датаграм кој не е фрагментиран (*PRBR*=29), но ова може да работи само за два соседни јазли. Прво, тие проверуваат дали приманиот датаграм има фрагментација (*MF* = 1). Во негативен случај, тие го користат битот што не се користи во *знаменца*, како индикатор дали датаграмот пренесува порака или не, и потоа ги стават податоците во полињата *идентификација* и *поместеност на фрагмент*.

Некои од техниките кои се користат од страна на Mazurczyk и Szczypiorski (Mazurczyk & Szczypiorski, Steganography in handling oversized IP packets, 2009) (Mazurczyk & Szczypiorski, Evaluation of steganographic methods for oversized IP packets, 2012), исто така, го користат процесот на фрагментација. Авторите предлагаат:

- поделба на оригиналниот IPv4 пакет во предефиниран број на фрагменти (на пример, парен број ќе биде бинарна 0, а непарен број ќе биде бинарна 1) – со што се испраќа 1 бит за пакет;
- подесување на вредноста која се вметнува во полето *поместеност на фрагмент* (на пример, парна вредност ќе биде бинарна 1, а непарна вредност ќе биде бинарна 0) – на овој начин се испраќаат $N_F - 1$ битови за пакет, каде што N_F е бројот на фрагменти за тој пакет;
- користење на легитимен фрагмент со стеганограм вметнат во корисниот товар и тогаш се испраќаат $N_F \cdot F_S$ битови за пакет, каде што N_F е бројот на фрагменти за тој пакет, а F_S е големината на фрагментот.

Скриени канали може да се креираат и со полиња од заглавјето на протоколот кои се менуваат за време на преносот, скриен канал со шум кој овозможува пренесување на 1 бит по пакет со користење на полето *TTL*, предложен од (Qu, Su, & Feng, 2004). (Zander, Armitage, & Branch, 2006) предложиле подобрување на 1-бит-по-пакет скриените канали енкодирање во полето *TTL*, со анализирање на почетната *TTL* вредност и нормалната *TTL* вредност која се јавува во мрежата. Тие предложиле користење на 2 различни почетни вредности за *TTL* во пакетите, голема вредност за претставување на бинарна 1, а мала вредност за претставување на бинарна 0. Бидејќи TTL може да кодира до 255 вредности, а најчесто повеќето патеки помеѓу машините во интернет се инфериорни до 40, напаѓачот може да испрати пакети со висока почетна TTL вредност од 255 (енкодира 1) и почетна ниска вредност од 142 (енкодира 0). На пример, ако постојат 20 прескокнувања помеѓу А и Б, Б ќе добие пакети со TTL од 235 или 122, кодирајќи единици или нули.

(Abad, 2001) демонстрирал како фундаментална грешка во дизајнот на Интернет сумата за проверка може да му дозволи на злонамерен корисник да креира скриен канал во 16-битното поле *сума за проверка на заглавјето* со помош на хеш колизии (*PRBR*=16).

IPv4 има механизми за откривање на PMTUD (Path MTU Discovery) кои користат пробни пораки со поставен DF бит и ICMP за примање на нотификации (Mazurczyk & Szczypiorski, Steganography in handling oversized IP packets, 2009)

(Mazurczyk & Szczypiorski, Evaluation of steganographic methods for oversized IP packets, 2012). На овој начин може да се користат пробните пораки во PMTUD за пренос на тајната порака и може да се предизвика испраќање на лажни ICMP пораки кај примателот.

Полињата на изворната и одредишната адреса се, исто така, корисни. Пакет со лажна изворишна адреса може да кодира податоци, како и пакет со лажна дестинациска адреса. Во вториот случај напаѓачот мора да ја надгледува мрежата за да ја добие содржината на пакетот. Исто така, може да се чуваат податоци во полето за опции на заглавјето, со непостоечки опции. Валидните опции можат да се користат со пополнување без нули кои ги енкодираат податоците. Во полето за податоци може да се сместат податоци за скриените канали кога е поставен RST обележувачот. Податоците, исто така, може да се додадат на крајот од полето за податоци. Во второто сценарио дополнителните податоци се неочекувани од страна на серверот и мора да бидат отстранети пред да стигнат до него.

Во статиите (Trabelsi, El-Sayed, Frikha, & Rabie, Traceroute Based IP Channel for Sending Hidden Short Messages, 2006), (Trabelsi, El-Sayed, Frikha, & Rabie, A novel covert channel based on the IP header record route option, 2007) е предложен скриен канал со кој може да се испраќаат кратки пораки во полето *опција за снимање на рута* (Record Route Options) кое може да има големина до 40B. Авторите развиле и практична имплементација, која се темели на вообичаена апликација која го користи ова поле – traceroute командата. Во (Trabelsi & Jawhar, Covert File Transfer Protocol Based on the IP Record Route Option, 2010) е даден дури и скриен протокол за пренос на текстуални датотеки или кратки пораки, кој го користи истото поле *опција за снимање на рута.* Овој протокол користи сесиски ориентирани механизми кои нудат TCP слични особини вгнездени во IPv4 полето *опции* и во еден IP пакет може да вметне до 34B скриена порака.

Една имплементација на скриен канал во IPv4 која го користи полето *изворна адреса* е дадена со B0CK (vecna, 2000). Кај оваа имплементација и покрај тоа што полето *протокол* во IP пакетот е поставено на IGMP, пакетот не содржи енкапсулирано заглавје од IGMP, додека пакетот е наполнет со 124 бајти на нули

во 20 бајтното IP заглавје. Ранливоста може да се намали со забрана на лажните пакети од страна на мрежните протоколи.

Временски скриени канали

Временските скриени канали користат паметен начин за кодирање на информациите поврзан со почеток или крај на одредени настани, без промена на различни полиња во IP заглавјето. На пример, скриениот канал може да се дефинира на следниот начин: ако еден пакет се испрати во даден временски интервал, се кодира бинарна 1, инаку, ако не се испрати, се кодира бинарна 0 (Padlipsky, Snow, & Karger, 1978). Ако секундата е поделена во N_i интервали, *RBR*=$N_i bps$. Оваа идеја била имплементирана од страна на (Cabuk, Brodley, & Shields, 2004). Клиентска програма слуша на дадена порта за пристигнување на првото PDU. Дополнително на податочните битови, серверот испраќа и битови за синхронизација и корекција на грешки. Во истиот труд се објаснети и некои техники за детекција на скриени канали во IP. (Berk, Giani, & Cybenko, 2005) го користат доцнењето меѓу последователните пакети за енкодирање на тајната информација. Во нивниот систем, сите доцнења се складираат и се пресметува средна вредност секогаш кога доаѓа ново доцнење. Секое доцнење над средната вредност се декодира како бинарна 1, а секое доцнење под средната вредност се декодира како бинарна 0. На овој начин со *n* испратени пакети може да се кодираат *n*-1 битови.

(Servetto & Vetterli, 2001) воведуваат намерни загуби во нумериран проток на пакети за создавање на скриени канали со користење на фантомски пакети. Тие прескокнуваат еден секвентен број кај испраќачот, па така нема изгубени кориснички податоци. Доколку се појави загуба на пакет во даден временски интервал, тоа ќе одговара на испраќање на еден бит. Авторите од (Mazurczyk & Szczypiorski, Steganography in handling oversized IP packets, 2009) (Mazurczyk & Szczypiorski, Evaluation of steganographic methods for oversized IP packets, 2012) ги користат истите техники за креирање на фантомски фрагменти, со прескокнување на една вредност на *поместеност на фрагмент*. Овие автори предлагаат уште еден временски скриен канал со користење на различни брзини на фрагментација

(на пример, една брзина ќе биде бинарна 1, а друга ќе биде бинарна 0). Во овој случај се испраќаат $log_2 h$ битови на пакет, каде h е бројот на брзини со кои се прави фрагментацијата.

(Ahsan, Covert channel analysis and data hiding in TCP/IP, 2002) и (Ahsan & Kundur, Practical data hiding in TCP/IP, 2002) покажале како може да се креира временски скриен канал со сортирање на пакетите. Ако се испраќаат *n* пакети и ако мрежата го гарантира редоследот на пакетите при испорака, тогаш со промена на редоследот на пакетите може да се испратат $log_2 n!$ бита. За ова да биде можно, потребна е референца која ќе ги поврзе броевите на сортираните пакети со нивниот природен редослед. За таа цел може да се користи 32-битното поле *редоследен број (Sequence Number)* кај *заглавјето за автентикација* (*Authentication Header*) и полињата *должина на опциони податоци (Option Data Length)* и *опциони податоци (Option Data)* кај IPSec. Во (Ahsan & Kundur, Practical data hiding in TCP/IP, 2002) е даден и алгоритам за проценка на најдобрата секвенца во ресортирањето, кој може да се примени во реална мрежа која не го гарантира редоследот на пакетите при испорака. Авторите на (Mazurczyk & Szczypiorski, Steganography in handling oversized IP packets, 2009) (Mazurczyk & Szczypiorski, Evaluation of steganographic methods for oversized IP packets, 2012) го користат истиот метод за сортирање на фрагменти на даден пакет.

(Galatenko, Grusho, Kniazev, & Timonina, 2005) предложиле временски скриен канал со шум со сортирање на пакети, така што дестинациските адреси да се подредени. Низа од дестинациски адреси во растечки редослед ќе биде бинарна 1, а во опаѓачки ќе биде бинарна 0. Бројот на потребните пакети ќе зависи од ратата на грешка на каналот.

Во (Allix, 2007) е даден пример на следниот временски скриен канал. Ако напаѓачот има контрола над две машини А и Б, секоја од нив има врска со истиот сервер Ц, редоследот на податоци испратени од страна на секоја машина може да се користи како скриен канал. На пример, ако машината А испраќа пакет, а потоа машината Б испрати два, ова може да се толкува како 0. Ако машината А испрати два пакети, а потоа машината Б еден, ова може да се толкува како 1.

Најочигледниот скриен канал со бројач кој може да се замисли е претставен со бинарен код (Allix, 2007). Се избира произволен број λ. За даден временски интервал, ако бројот на пренесени податоци е инфериорен во однос λ, се кодира бинарна 0, ако бројот на префрлени податоци е супериорен или еднаков на λ, се кодира 1.

(Danezis, 2005) предложил индиректен складирачки скриен канал со шум, што го користи полето *идентификација*. Двете страни комуницираат преку меѓујазел со оперативен систем кој глобално ја зголемува вредноста на ова поле за еден за секој испратен пакет. Основна цел е да се натера меѓујазелот да ги препраќа пакетите меѓу двете страни, а за таа цел авторите предлагаат користење на ICMP Echo Request пораките или механизамот за контрола на застој кај TCP. Во даден временски интервал испраќачот испраќа *n* пакети на меѓујазелот и го присилува и тој да врати *n* пакети. Во бројот *n* е енкодирана тајната информација. На почетокот од секој временски интервал примачот го присилува меѓујазелот да испрати 1 порака и со пресметување на разликата во полето *идентификација*, меѓу два последователни примени пакети кај примачот, примачот го дознава *n.* Според авторот на овој начин може да се пренесат 16 бита во секунда.Во прилог е дадена сумарна табела со сите различни стеганографски техники за протоколот IPv4.

Табела 3. Стеганографски техники за IPv4
Table 3. Steganographic techniques for IPv4

Статија	Зафатени полиња/ техника	RBR (bps)	PRBR (bitovi/paket)	Тип
(Rowland, 1997)	*идентификација*		16	складирачки, просторен канал базиран на вредност
(Ahsan, Covert channel analysis and data hiding in TCP/IP, 2002), (Ahsan & Kundur, Practical data hiding in TCP/IP, 2002)	*DF*		1	складирачки, просторен канал базиран на вредност
	сортирање на пакети	$k \cdot log_2 n!$, каде kn е број на испратени пакети во 1 секунда, а n е бројот на пакети што се сортираат		временски, базиран на транзиција

(Ahsan, Covert channel analysis and data hiding in TCP/IP, 2002)	*DF идентификација*		17	складирачки, просторен канал базиран на вредност
	верзија IHL идентификација		8	складирачки, просторен канал базиран на вредност
(Cauich, Gomez, & Watanabe, 2005)	*идентификација поместеност на фрагмент*		29	складирачки, просторен канал базиран на вредност
(Mazurczyk & Szczypiorski, Steganography in handling oversized IP packets, 2009) (Mazurczyk & Szczypiorski, Evaluation of steganographic methods for oversized IP packets, 2012)	предефиниран број фрагменти		1	складирачки, просторен канал базиран на транзиција
	подесување на вредноста на *поместеност на фрагмент*		$N_F - 1$, каде N_F е број на фрагменти	складирачки, просторен канал базиран на вредност
	користење на легитимен фрагмент со стеганограм вметнат во корисниот товар		$N_F \cdot F_S$, каде N_F е број на фрагменти, а F_S е големината на фрагментот	складирачки, просторен канал базиран на вредност
	користење на различни брзини на фрагментација		$log_2 h$, каде h е број на различни брзини	временски, базиран на транзиција
	намерно загубени фрагменти	n каде n е број на временски интервали во кои има или нема загубени фрагменти во 1 секунда		временски, базиран на транзиција
	сортирање на фрагменти	$k \cdot log_2 n!$, каде kn е број на испратени фрагменти во 1 секунда, а n е бројот на фрагменти што се сортираат		временски, базиран на транзиција
(Qu, Su, & Feng, 2004)	*TTL*		1	складирачки, просторен канал базиран на

				вредност
(Zander, Armitage, & Branch, 2006)	*TTL*		1	складирачки, просторен канал базиран на вредност
(Abad, 2001)	*сума за проверка на заглавјето*		16	складирачки, просторен канал базиран на вредност
(Trabelsi, El-Sayed, Frikha, & Rabie, Traceroute Based IP Channel for Sending Hidden Short Messages, 2006), (Trabelsi, El-Sayed, Frikha, & Rabie, A novel covert channel based on the IP header record route option, 2007)	*опција за снимање на рута*		40*8	складирачки, просторен канал базиран на вредност
(Trabelsi & Jawhar, Covert File Transfer Protocol Based on the IP Record Route Option, 2010)	*опција за снимање на рута*		34*8	складирачки, просторен канал базиран на вредност
B0CK (vecna, 2000)	*изворна адреса*		32	складирачки, просторен канал базиран на вредност
(Padlipsky, Snow, & Karger, 1978) (Cabuk, Brodley, & Shields, 2004)	испраќа/не испраќа во даден временски интервал	N_i каде N_i е број на интервали во 1 секунда		временски, базирани на транзиција
(Berk, Giani, & Cybenko, 2005)	користење на доцнењето меѓу последователните пакети	$\frac{n-1}{n}$, каде n е бројот на испратени пакети во 1 секунда		временски, базиран на транзиција
(Servetto & Vetterli, 2001)	намерни загуби на пакети	n каде n е број на временски интервали во кои има или нема загубени пакети во 1 секунда		временски, базиран на транзиција
(Galatenko, Grusho, Kniazev, & Timonina, 2005)	сортирање на пакети со подредени дестинациски адреси	kn пакети испратени во 1 секунда, каде n е број на пакети за еден		временски, базиран на транзиција

		бит		
(Allix, 2007)	користење на 2 машини и 1 сервер	$\frac{n}{3}$ каде n е вкупниот број на испратени пакети од двете машини во 1 секунда		временски, базиран на транзиција, со бројач
(Allix, 2007)	броење на настани во даден временски интервал	N_i каде N_i е број на интервали во 1 секунда		временски, базиран на транзиција, со бројач
(Danezis, 2005)	меѓујазел кој глобално го зголемува полето *идентификација*	16		временски канал базиран на вредност

2.1.2. IPv6

Последната верзија на IP, IPv6 користи 128 бита за адреса и е стандардизиран во 1998 година како RFC 2460 (Deering & Hinden, 1998). Пакетот се состои од контролирани информации за адресирање и упатување, како и товар кој се состои од податоците на корисникот. Контролата на информациите во IPv6 пакетите е поделена на задолжително фиксно заглавје и опционални заглавја кои можат да бидат продолжени (Murphy, IPv6 / ICMPv6 Covert Channels, 2008). Товарот на IPv6 пакетот претставува датаграм или сегмент на повисоко ниво (транспортен протокол), но може да биде и податок за интернет нивото (на пример, ICMPv6) или за нивото на поврзување (на пример, OSPF). Упатувачите не ги фрагментираат IPv6 пакетите, како што тоа го прават за IPv4. Хостовите се потребни за да го спроведат откривањето на MTU патеката и за да ги искористат предностите на поголемиот MTU од 1280 октети. Хостовите може да користат фрагментација за испраќање на пакети поголеми од набљудуваниот MTU.

IPv6 пакетите обично се пренесуваат преку протоколот на нивото за мрежен пристап, но исто така, може да биде протокол за тунелирање на повиското ниво, како што е IPv4 кога се користи 6to4 или Teredo технологијата за транзиција. Главното IPv6 заглавје е прикажано на табела 4. Дополнително, постојат уште неколку заглавја кои може да ги има во пакетот, и тоа: заглавје со опции за скокови (*Hop-by-Hop Options*), заглавје со опции за дестинацијата (*Destination Options*), заглавје за упатување (*Routing*), заглавје за фрагментација (*Fragment*),

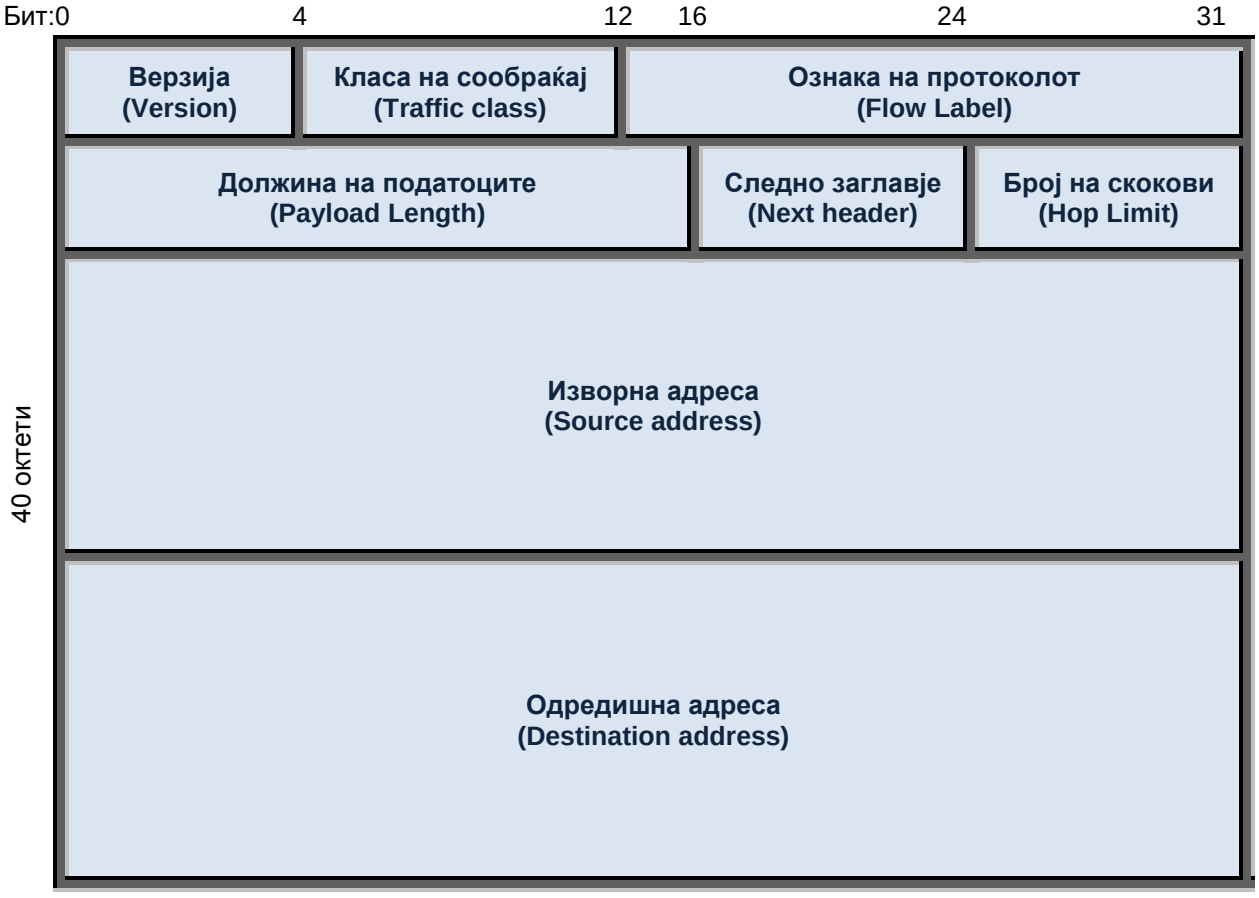
заглавје за автентикација (*Authentication Header - AH*) и заглавје за енкапсулиран безбедносен товар (*Encapsulating Security Payload - ESP*).

Полето ***верзија*** *(Version)* (4 бита) укажува на верзијата на протоколот, во овој случај верзија 6.

Полето *класа на сообраќај (Traffic class)* е еквивалентно на полето DS/ECN во IPv4.

Полето *ознака за тек (Flow Label)* (20 бита) може да се користи од страна на хостот за означување на пакетите за кои бара специјален третман од страна на упатувачите во мрежата, како и за управување со QoS (Quality of Service). Полето е игнорирано од страна на уредите кои не го поддржуваат тоа.

Табела 4. Заглавје на IPv6
Table 4. IPv6 header



DS = поле за идентификациони сервиси
ECN = поле за екплицитна објава на застој

Должина на товарот (Payload Length) (16 бита) укажува на должината на пакетот кој следува по заглавјето. Ова е вкупната должина на сите заглавја за проширување плус PDU од транспортното ниво.

Полето *следно заглавје (Next Header)* (8 бита) укажува на типот на заглавјето кое следува по IPv6 заглавјето.

Полето *број на скокови* (*Hop Limit*) (8 бита) е еквиваленто на полето *TTL* во IPv4 протоколот.

Полето *изворна адреса* (128 бита) е адресата на јазолот кој го генерира пакетот.

Полето *одредишна адреса* (128 бита) е адресата на планираниот примател на пакетот. Ова, всушност, не мора да биде планираното крајно одредиште доколку е присутно заглавје за упатување.

Скриени канали во IPv6

Складирачки скриени канали

Во 2005 година, (Graf, 2003) предложил да се користи модифицирање на полето *опции (Options)* во заглавјето со опции за дестинација, кои се TLV (тип-должина-вредност) енкодирани.

Lucena и соработниците (Lucena, Lewandowsk, & Chapin, 2005) анализирале неколку скриени канали за складирање на податоци во заглавјето на IPv6 и за некои од нив испраќачот мора да го пресмета *ICV* заедно со скриените податоци. Манипулација на главното и дополнителните заглавја кај IPv6 може да се направи на неколку начини:

- со поставување на лажен сообраќај во полето *класа на сообраќај (Traffic class);*
- со поставување на лажен тек во полето *ознака за тек*;
- со поставување на лажна изворна адреса во *изворна адреса*;
- со поставување на почетна вредност во полето *број на скокови* и манипулирање со вредноста на последователните пакети;

- со поставување на валидна вредност во *следно заглавје* и додавање на екстра дополнително заглавје или преку зголемување на вредноста во полето *должина на товарот* и додавање на дополнителни податоци на крајот од пакетот;
- со модифицирање полињата *должина на опциони податоци (Option Data Length)* и *опциони податоци (Option Data)* во заглавјето со опции за скокови;
- со поставување на вредност во 4-битното поле *резервирано (Reserved)* кај тип 0 упатувањето или со фабрикација на адреси до 2048 бајти на пакет во заглавјето за упатување;
- со поставување на вредност во 2-битното и 8-битното поле *резервирано (Reserved),* лажно *следно заглавје*, или со вметнување на цел лажен фрагмент во заглавјето за фрагментација;
- со модифицирање полињата *должина на опциони податоци (Option Data Length)* и *опциони податоци (Option Data)*, фабрикување на една или повеќе опции или лажна вредност за падирање во заглавјето со опции за дестинација;
- со поставување на вредност во 16-битното поле *резервирано (Reserved),* или со креирање на лажно заглавје со големина до 1022 бајти на пакет кај заглавјето за автентикација;
- со креирање на лажно заглавје со големина до 1022 бајти на пакет кај ESP заглавјето или со поставување на лажна вредност за падирање до 255 бајти на пакет.

Временски скриени канали

Бидејќи IPSec е задолжително за IPv6 и кај него може да се користи сортирањето на пакети и фрагменти. IPv6 има механизам за откривање на PMTU со PLPMTUD (Packetization Layer Path MTU Discovery) на тој начин што почнува со испраќање на пакети со релативно мала должина и ако тие поминат се продолжува прогресивно со поголеми должини (Mazurczyk & Szczypiorski, Steganography in handling oversized IP packets, 2009) (Mazurczyk & Szczypiorski,

Evaluation of steganographic methods for oversized IP packets, 2012). На овој начин може да се користат пробните пораки во PLPMTUD за пренос на тајната порака и може да се предизвика испраќање на лажни ICMP пораки кај примателот. Во друг труд (Mazurczyk & Szczypiorski, Evaluation of steganographic methods for oversized IP packets, 2012) истите автори предложиле употреба на RSTEG (Retransmission Steganography) за стеганографски PLPMTUD кој користи намерни препраќања за испраќање на стеганограми. Главната идеја зад RSTEG е да не се потврдува успешно приман пакет, со цел да се предизвика повторно испраќање. Пакетите кои повторно се испраќаат носат стеганограм во полето со податоци.

Во прилог е дадена сумарна табела со сите различни стеганографски техники за протоколот IPv6.

Табела 5. Стеганографски техники за IPv6
Table 5. Steganographic techniques for IPv6

Статија	Зафатени полиња/ техника	RBR (bps)	PRBR (bitovi/paket)	Тип
(Graf, 2003)	*опции* (заглавје со опции за дестинацијата)		max length of the field	складирачки, просторен канал базиран на вредност
(Lucena, Lewandowsk, & Chapin, 2005)	*класа на сообраќај*		8	складирачки, просторен канал базиран на вредност
	ознака за тек		20	складирачки, просторен канал базиран на вредност
	изворна адреса		128	складирачки, просторен канал базиран на вредност
	број на скокови		1	складирачки, просторен канал базиран на вредност
	должина на товарот		$\leq (2^{16}$- GT) *8, каде GT е големината на корисничкиот товар	складирачки, просторен канал базиран на вредност

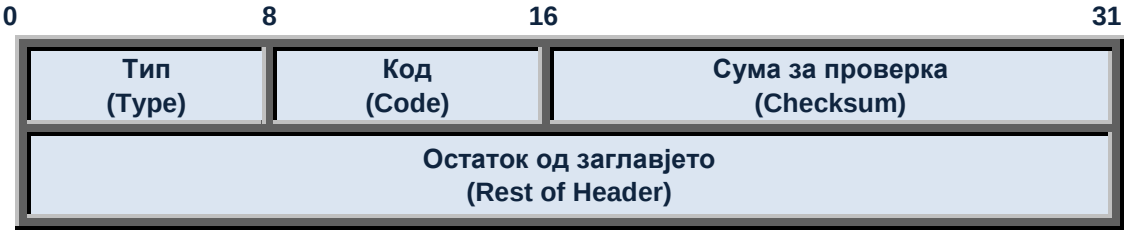
	резервирано (заглавје за упатување)		4	складирачки, просторен канал базиран на вредност
	или со фабрикација на адреси (заглавје за упатување)		≤ 2048*8	складирачки, просторен канал базиран на вредност
	резервирано (заглавје за фрагментација)		2+6	складирачки, просторен канал базиран на вредност
	следно заглавје (заглавје за фрагментација)		8	складирачки, просторен канал базиран на вредност
	резервирано (заглавје за автентикација)		16	складирачки, просторен канал базиран на вредност
	креирање на лажно заглавје (заглавје за автентикација)		≤ 1022*8	складирачки, просторен канал базиран на вредност
	креирање на лажно заглавје (ESP заглавје)		≤ 1022*8	складирачки, просторен канал базиран на вредност
	лажна вредност на падирање (ESP заглавје)		≤ 255*8	складирачки, просторен канал базиран на вредност
(Mazurczyk & Szczypiorski, Evaluation of steganographic methods for oversized IP packets, 2012)	намерни препраќања на пакети	$\leq n \cdot L$ каде n е бројот на препратени пакети во 1 секунда, а L е максималната големина на товарот		хибриден од просторен канал базиран на вредност и временски канал базиран на транзиција

2.2. ICMP

Internet Control Message Protocol - ICMP е дизајниран да обезбеди повратни информации за проблемите во комуникациската средина и користи 8 бајтно заглавје (табела 6). ICMP е дефиниран во RFC 792 (Postel, INTERNET CONTROL MESSAGE PROTOCOL, 1981) и ICMP пораките се испраќаат енкапсулирани во IP пакети. Постојат повеќе кодирани пораки во рамките на ICMP за правилно дијагностицирање на проблемите со мрежата и протокот на сообраќајот. ICMP пораките се испраќаат под многу различни околности, како што се недостижните дестинации или мерење на доцнењето. Едноставните сервиси за решавање на проблеми во мрежата, како што се пинг и барањето на патот, користат експлицитни ICMP пораки за да се соберат информации за мрежата.

Две од најчестите ICMP пораки се ICMP Echo Request и ICMP Echo Response, кои вообичаено се користат за да се дијагностицираат проблеми со мрежата. Постапката се состои од испраќање на порака со ICMP Echo Request до одреден хост, кој одговара со порака ICMP Echo Response. Протоколот, исто така, овозможува враќање на податоци со променлива должина на испраќачот во полето за дополнителни податоци. Некои IP опции како алармирање на упатувачот, зачувување на рутата и временската ознака можат да се користат кога се создава една ICMP порака за ехо-барање, која обезбедува канал за скриени врски помеѓу субјектите кои комуницираат. ICMP заглавјето е прикажано на табела 6.

Табела 6. Заглавје на ICMP
Table 6. ICMP header



Поле *тип* (*Type*) (8 бита) – тип на порака и тој го одредува форматот на останатиот дел од податоците во ICMP. Пораките ехо-барање и ехо-одговор се

запишуваат како 0x8 и 0x0 соодветно и се користат од страна на пинг за да се утврди дали хостот е онлајн и е на располагање.

Поле *код* (*Code*) (8 бита) - поттип на дадениот тип.

Поле *сума за проверка* (*Checksum*) (16 бита) - проверка на грешки во податоците. Се пресметува од ICMP заглавјето заедно со податоците, со зададена вредност 0 за оваа поле. Алгоритамот за проверка е наведен во RFC 1071.

Поле *остатокот од заглавјето* (32 бита) - се разликуваат врз основа на типот на ICMP.

ICMPv6 (ICMP за IPv6) е дефиниран во 2006 година во RFC 4443 (Conta, Deering, & M. Gupta, 2006) и се користи од страна на јазлите на IPv6 за пријавување на грешки кои се случуваат во обработката на пакетите и за вршење на други функции на Интернет нивото, како што се известување за грешки, дијагностика (ICMPv6 "пинг") итн. ICMPv6 е составен дел на IPv6 мрежите и основниот протокол (сите пораки и однесувања кои се бараат со оваа спецификација) мора да биде целосно имплементиран од страна на секој јазол на IPv6.

Објавени се неколку додатоци кои ги дефинираат новите типови на пораки од ICMPv6, како и нови опции за постоечките типови на ICMPv6 пораки. Протоколот за пронаоѓање на сосед (NDP) е протокол за откривање на јазли во IPv6 кој ги заменува и подобрува функциите на ARP. Безбедниот протокол за пронаоѓање на сосед (SEND) е продолжување на NDP со дополнителна безбедност. MRD овозможува откривање на мултикаст упатувачи. Пораките на ICMPv6 може да се класифицираат во две категории: пораки за грешки и пораки за информации. Пораките на ICMPv6 се пренесуваат со IPv6 пакетите во кои IPv6 вредноста за следното заглавје за ICMPv6 е поставена на 58, а заглавјето има слична структура со ICMP заглавјето.

Скриени канали во ICMP

Повеќето скриени канали кои вклучуваат користење на ICMP во голема мера се складирачки канали, каде што неискористените полиња се користат за скриена комуникација. ICMP како скриен канал нуди многу бенефиции поради

целокупната едноставност. Постојат само неколку полиња во рамките на повеќето ICMP пораки, кои овозможуваат брзи имплементации и едноставно подесување на каналот. Она што навистина го прави ICMP протоколот остварлив за скриените канали е употребата на полињата за податоци и носивост во рамките на одредени пораки. Со генерирање на пакети врз основа на специфично кодирани пораки и вградување на пораката од постоечкиот скриен канал во податочното поле овозможува ICMP да послужи како една алтернативна употреба за скриените канали. Овие едноставни фактори овозможуваат ICMP да се смета како невидлив сообраќај.

Проектот Локи (daemon9, AKA, & route, Project Loki, 1996) и (daemon9, Loki2 (the implementation), 1997) демонстрира скриен канал со произволно тунелирање во корисниот товар на ICMP Echo Request ICMP и Echo Response пораките. Дополнително, Loki клиентот му дозволува на оддалечен напаѓач да обвитка и пренесе команди во корисниот товар на ICMP порака, а Loki серверот ги одвиткува и извршува командите, испраќајќи ги резултатите до напаѓачот, повторно обвиткани во ICMP пораки. Овој канал работи за секој мрежен уред кој не ја филтрира содржината на ICMP Echo сообраќајот. Проектот Локи ги покажува способностите на ICMP за пренесување на скриени информации, користејќи го податочното поле во ICMP Echo пораките и тој е првиот пример на тунел во корисен товар. Ова поле е наменето за снимање на информациите за рутата или чување на евиденција за тајмингот, за да се пресмета времето на една обиколка. Податочното поле стандардно зафаќа 24 или 56 бајти, во зависност од оперативниот систем на хостот. Сепак, протоколот овозможува тоа да биде многу подолго, со што се дава произволно висок пропусен опсег. Проверката на полето за податоци од страна на оперативниот систем, заштитниот sid или централните упатувачи на хостот е доста ретка, па така оваа поле може да содржи произволни податоци. Авторот сугерира три опции за компајлирање на алатката. Стандардната опција компајлира програма за извршување која може да се покрене како видлив демон. Втората опција овозможува да се компајлира како покренувачки кернел модул. За максимална невидливост тие препорачуваат рекомпајлирање на кернелот на оперативниот систем со подобрениот пинг на

Локи. Локи2 користи енкрипција за да го прошири кодирањето на пораката. Опциите вклучуваат слабо кодирање со XOR и посилна криптографија со Diffie - Hellmann и Blowfish.

Пинг тунелот од (Stødle, 2005) врши тунелирање на TCP врски преку ICMP Echo Request и Echo Response пораки. Други имплементаци на IP-над-ICMP тунели се дадени во (Zelenchuk, 2004), (Thomer, 2005) и (Muench, 2003). Во имплементацијата Skeeve на (Zelenchuk, 2004) скриен испраќач испраќа ICMP Echo Request за да се врати од хост со спуфирана изворна адреса (поставена на адресата на скриениот примач) со тајна порака во корисниот товар. Хостот тогаш испраќа ICMP Echo Response на скриениот примач со истиот корисен товар, како и во барањето. ICMP-Chat (Muench, 2003) спроведува основен механизам за конзолно-базирани разговори, кој користи ICMP пакети за комуникација. Ова решение вклучува шифрирање на пораките со AES-256 и заштита со лозинка на каналот со помош на SHA-256. Овие канали понатаму се испитувани во (Stokes, Yuan, Johnson, & Lutz, 2009).

Друг начин на криење на податоци во ICMP протоколот е 32-битното резервирано поле во ICMP Router Solicitation пораката (Ahsan, Covert channel analysis and data hiding in TCP/IP, 2002). За воспоставување на скриен канал може да се користи и застарената ICMP Address Mask Request порака (Scott, 2008). Барањето нормално го исполнува 32-битното поле за маскирање на адресата со нули, но може да се користи и за пренос на податоци. Сепак, опсегот на скриените канали кои се базираат на оваа техника ќе биде многу ограничен, бидејќи овие пораки се испраќаат од хостот до упатувачот на истата локална мрежа. Алатката V00d00n3t (Murphy, V00d00n3t – Ipv6 / ICMPv6 Covert Channel, 2006) има опција да креира скриени канали не само со IPv6, туку и со ICMPv6.

Табела 7. Стеганографски техники за ICMP
Table 7. Steganographic techniques for ICMP

Статија	Зафатени полиња/ техника	RBR (bps)	PRBR (bitovi/paket)	Тип
Loki (daemon9, AKA, & route, Project Loki, 1996) (daemon9, Loki2 (the implementation), 1997)	ICMP Echo Request ICMP и Echo Response		зависи од ОС и имплементацијата, 24*8, или 56*8, или повеќе	складирачки, просторен базиран на вредност, тунел во корисен

PingTunnel (Stødle, 2005) Skeeve (Zelenchuk, 2004) ICMP-Chat (Muench, 2003)				товар
(Ahsan, Covert channel analysis and data hiding in TCP/IP, 2002)	ICMP Router Solicitation *резервирано*		32	складирачки, просторен базиран на вредност
(Scott, 2008)	ICMP Address Mask Request *адресна маска*		32	складирачки, просторен базиран на вредност

Авторите на (Ray & Mishra, 2008) демонстрирале дека е можен и пренос на голема количина на податоци со софистицирана поддршка на безбедност и надежност. Предложиле протокол кој ќе ги користи ICMP Echo Request пораките, како и техники на отпечатување на оперативен систем за симулирање на однесувањето на TCP/IP складот.

Singh и соработниците (Singh, Lu, Nordstrom, & dos Santos, 2003) предложиле одбрана против ICMP тунелирање. Табела 7 ги сумира различните стеганографски техники за протоколот ICMP.

2.3. IGMP

Internet Group Management Protocol (IGMP) е комуникациски протокол кој се користи од страна на хостовите и соседните упатувачи на IP мрежата за да се воспостави членство во мултикаст групите. IGMP е дефиниран во RFC 3376 (Cain, Deering, Kouvelas, & Thyagarajan, 2002) и е составен дел од IP мултикаст. IP мултикаст овозможува пренос на податоци на подмножество од хост компјутери, кои може да се шират низ различни физички мрежи низ Интернет. Даденото подмножество е познато како мултикаст група. IGMP пакетите се врзуваат во IP датаграмите за пренос, каде што одредишната адреса е мултикаст адреса. Последна верзија е IGMPv3 која имплементира три типа на пораки: прашалник за членство, извештај за членство и групен запис. IGMP се користи во IPv4 мрежите.

Наместо посебна верзија за IPv6 мрежите, управувањето со мултикаст е имплементирано во ICMPv6.

Скриени канали во IGMP

Заглавјето на IGMPv3 содржи полиња кои можат да се искористат за скриена комуникација преку правилно вградување и извлекување на процесите на крајот од комуникацијата (табела 8). Пораките со извештај за членство обезбедуваат 8-битни и 16-битни полиња *резервирано*, кои всушност се поставени на нула и се игнорираат од страна на примачот (Scott, 2008).

Сепак, IGMP протоколот е наменет за комуницирање помеѓу хостовите и упатувачите кои се наоѓаат во иста мрежа. Обично полето за време на живот во IP заглавјето за IGMP пораките се поставува на 1, што ги прави овие пораки неупатувачки надвор од локалната мрежа. Ова го ограничува опсегот на скриените комуникации со користење на овој протокол за вградување на тајни податоци.

Табела 8. Стеганографски техники за IGMP
Table 8. Steganographic techniques for IGMP

Статија	Зафатени полиња/ техника	RBR (bps)	PRBR (bitovi/paket)	Тип
(Scott, 2008)	*резервирано* (извештај за членство)		8 + 16	складирачки, просторен канал базиран на вредност

2.4. DHCP

Dynamic Host Configuration Protocol (DHCP) е мрежен протокол кој му овозможува на серверот автоматски да додели IP адреса на компјутерот од еден дефиниран опсег на броеви кои се конфигурирани за одредена мрежа. Со динамичното адресирање еден уред може да има различна IP адреса со секое наредно поврзување на мрежата. Во некои системи, IP адресата на уредот може да се промени дури и додека уредот е сè уште поврзан на мрежата.

Заглавјето на DHCP е дадено на табела 9.

Табела 9. Заглавје на DHCP
Table 9. DHCP header

OP - 8bytes	**HTYPE** - 8bytes	**HLEN** - 8bytes	**HOPS** - 8bytes
XID (Transaction Identifier) - 4bytes			
SECS - 2bytes		**FLAGS** - 2bytes	
CIADDR (Client IP Address) - 4bytes			
YIADDR (Your IP Address) - 4bytes			
SIADDR (Server IP Address) - 4bytes			
GIADDR (Gateway IP Address) - 4bytes			
CHADDR (Client Hardware Address) - 16bytes			
SNAME (Server Host Name) - 64bytes			
FILE (Boot File Name) - 128bytes			
COOKIE			
Option Value (Option Number, Length, Private-use Options)			

Полето *OP* (8 бајти) го одредува општиот тип на пораката. Вредноста 1 укажува на порака за барање, додека вредноста 2 е одговорот на пораката.

Полето *HTYPE* (8 бајти) го одредува типот на хардвер кој се користи за локалната мрежа.

Полето *HLEN* (8 бајти) ја одредува должината на хардверските адреси во дадената порака.

Полето за *HOPS* (8 бајти) се поставува на 0 од страна на клиентот пред да се пренесе барањето и се користи од страна на агентите за пренос за контролирање на насочувањето на BOOTP и / или DHCP пораките.

Полето *XID* (32 бита) е случаен број, кој е избран од страна на клиентот и се користи од страна на клиентот и серверот за поврзување на пораките за барање со соодветните одговори.

Полето *SECS* (16 бита) го одредува поминатото време откако клиентот го започнал процесот на барање, изразено во секунди.

Поле за *FLAGS* (16 бита) се користи само првиот бит, останатите се резервирани за идна употреба.

Полето *CIADDR* (32 бита) се поставува само ако клиентот е во следниве состојби BOUND (обврзан), RENEW (обновен) или REBINDING (поврзан) и можат да одговорат на ARP барањата.

Полето *YIADDR* (32 бита) е IP адресата која серверот му ја доделува на клиентот.

Полето *SIADDR* (32 бита) е IP адресата на серверот кој испраќа или следниот сервер кој може да се користи во наредниот процес на подигање (bootstrap).

Полето *GIADDR* (32 бита) е IP адресата на портата.

Полето *CHADDR* (128 бита) е хардверската (физичката) адреса на клиентот.

Полето за *SNAME* (512 бита) претставува нулта-прекинат стринг.

Полето *FILE* (1024 бита) е нулта-прекинат стринг. Се користи од страна на клиентот во DHCPDISCOVER пораката и од страна на серверот во DHCPOFFER пораката.

Полето за *Option Value* ги чува опциите за DHCP. Во реалните DHCP пораки секогаш мора да биде присутна барем една опција (т.е. тип на порака), така што ова поле не е празно.

Скриени канали во DHCP

DHCP има можност за пренос на складирачки временски канали, образложени и имплементирани во (Rios, Onieva, & Lopez, 2012):

- Полето *XID* има потенцијал да пренесе до 32 битови со скриени информации. Оваа вредност мора да биде случајно креирана од страна на клиентот, што значи дека не постои соодветен алгоритам за генерирање на идентификатор, како што е генераторот за редоследен број во TCP, со што се отежнува откривањето на скриените канали.
- Полето *SECS* може да се користи за криење на 1 бит информација слично како кај (Giffin, Greenstadt, Litwack, & Tibbetts, 2003).
- Кај полето *CHADDR* може да се искористат преостанатите 10 бајти, кога ова поле користи Етернет MAC адреса од 6B, чијашто должина е дефинирана во *HLEN* полето.

- Полињата *SNAME* и *FILE* се состојат од нулта-прекинати стрингови (`\0'), па сите податоци по овој знак ќе бидат запоставени од серверите и клиентите. Кога не пренесуваат податоци обично овие полиња се поставени на 0, но има ситуации кога може да носат туѓи податоци, посебно за полето *Option Value*, кога има вклучено опција 52 (Overload).
- Полето *Option Value* има променлива должина која обезбедува способност за криење на податоци на неколку начини. Бројот на опции може да се користи за преставување на буква во абецедата или со користење на референтна група од 8 опции и нивно преуредување за претставување на буквите од ASCII множеството на знаци или со користење на типот на опцијата поставен на одредено место може да се енкодира ASCII множеството на знаци. Една од најголемите негативности од употребата на овие методи е дека во зависност од типот на пораката постојат голем број опции кои се задолжителни или не се дозволени. Затоа, можноста за распоредување на кое било од овие решенија ќе зависи од тоа дали воведувањето на неовластени опции во одредени пакети ќе влијаат врз работењето на протоколот. Друг недостаток, кој е карактеристичен за третиот предложен метод, е дека кодирањето на пораки со карактери кои се повторуваат, може да бара создавање на опции кои веќе ги има во рамките на еден ист пакет, во некои случаи, може да биде доста сомнително.

Авторите од (Rios, Onieva, & Lopez, 2012) овие можности ги имаат имплементирано во HIDE_DHCP, кој интегрира 3 методи за креирање на скриени канали со користење на полињата XID, sName, File и Options. Имплементација се заснова на веќе постоечкиот DHCP код.

Табела 10 ги сумира различните стеганографски техники за протоколот DHCP.

Табела 10. Стеганографски техники за DHCP
Table 10. Steganographic techniques for DHCP

Статија	Зафатени полиња/ техника	RBR (bps)	PRBR (bitovi/paket)	Тип
(Rios, Onieva, & Lopez, 2012)	*XID*		32	складирачки, просторен канал базиран на вредност
	SECS		1	временски, базиран на вредност
	CHADDR (ако се користи Етернет MAC адреса)		80	складирачки, просторен канал базиран на вредност
	SNAME		64*8	складирачки, просторен канал базиран на вредност
	FILE		128*8	складирачки, просторен канал базиран на вредност
	Option Value		во HIDE_DHCP до 255*8	складирачки, просторен канал базиран на вредност

2.5. Address Resolution Protocol (ARP)

Address Resolution Protocol (ARP) е протокол кој се користи за мапирање на IP мрежните адреси (32-битна логичка адреса) во физички адреси (на пример, MAC адреса - 48 битна физичка адреса). Протоколот е дефиниран во 1982 година во RFC 826 (Plummer, 1982) и функционира под мрежното ниво како дел од интерфејсот помеѓу мрежата и нивото за поврзување на OSI. ARP кешот претставува табела која обично се користи за одржување на корелација помеѓу секоја MAC адреса и соодветната IP адреса. ARP ги обезбедува правилата за протоколот со составување на оваа корелација и обезбедување на конверзија на адреси во двете насоки.

Основната структура на ARP пакетот е прикажана во следната табела која илустрира случај на IPv4 мрежа која работи на Ethernet. Големината на ARP заглавјето е 28 бајти.

Табела 11. Заглавје на ARP
Table 11. ARP header

Internet Protocol (IPv4) over Ethernet ARP packet		
bit offset	**0 – 7**	**8 – 15**
0	Hardware type (HTYPE)	
16	Protocol type (PTYPE)	
32	Hardware address length (HLEN)	Protocol address length (PLEN)
48	Operation (OPER)	
64	Sender hardware address (SHA) (first 16 bits)	
80	(next 16 bits)	
96	(last 16 bits)	
112	Sender protocol address (SPA) (first 16 bits)	
128	(last 16 bits)	
144	Target hardware address (THA) (first 16 bits)	
160	(next 16 bits)	
176	(last 16 bits)	
192	Target protocol address (TPA) (first 16 bits)	
208	(last 16 bits)	

Полето *тип на хардвер (HTYPE)* го одредува типот на мрежниот протокол. На пример: Етернет се бележи со 1. Големината на ова поле е 2 бајти.

Полето *тип на протокол (PTYPE)* го одредува протоколот за работата на мрежата за која е наменето ARP барањето. За IPv4 има вредност 0x0800. Дозволените вредности на PTYPE го делат просторот за нумерирање со вредностите од типот на Етернет.

Полето *должина на хардверот (HLEN)* (8 бита) е должината на хардверската (MAC) адреса во октети. Големината на Етернет адресите е 6 бајти.

Полето *должина на протоколот (PLEN)* (8 бита) е должината на адресите кои се користат во протоколот од интернет ниво. Големината на IPv4 адресата е 4 бајти.

Полето *операција* (*OPER*) (16 бита) ја одредува операцијата која ја извршува испраќачот: 1 за ARP барање, 2 за ARP одговор.

Полето *хардверска адреса на испраќачот (SHA)* (48 bita) ја содржи MAC адресата на испраќачот.

Полето *протоколна адреса на испраќачот (SPA)* (32 bita) ја содржи IP адресата на испраќачот.

Полето *целна хардверска адреса (THA)* (48 bita) ја содржи MAC адресата на приемникот. Ова поле е игнорирано во барањата.

Полето *целна протоколна адреса (TPA)* (32 bita) ја содржи IP адресата на приемникот.

Скриени канали во ARP

Авторите на (Ji, Fan, & Ma, 2010) предлагаат користење на полето *целна протоколна адреса* за пренесување на тајната порака и тоа само последните 8 бита. На локалната мрежа се разликуваат два типа на хостови - со и без таен агент и иако сите ќе ги примаат ARP пакетите, само оние со таен агент ќе може да ги декодираат. Од последните 8 бита, само *w* битови се користат за пренос на тајната порака, а 8-*w* се користат како индикатор дека има тајна порака. На овој начин се добива складирачки скриен канал со помалку од 8 бита по пакет (табела 12), а бидејќи овие пакети во имплементацијата се испраќаат на почетокот од секоја секунда, RBR=PRBR.

Табела 12. Стеганографски техники за ARP
Table 12. Steganographic techniques for ARP

Статија	Зафатени полиња/ техника	RBR (bps)	PRBR (bitovi/paket)	Тип
(Ji, Fan, & Ma, 2010)	*целна протоколна адреса*	< 8	< 8	складирачки, просторен канал базиран на врендост

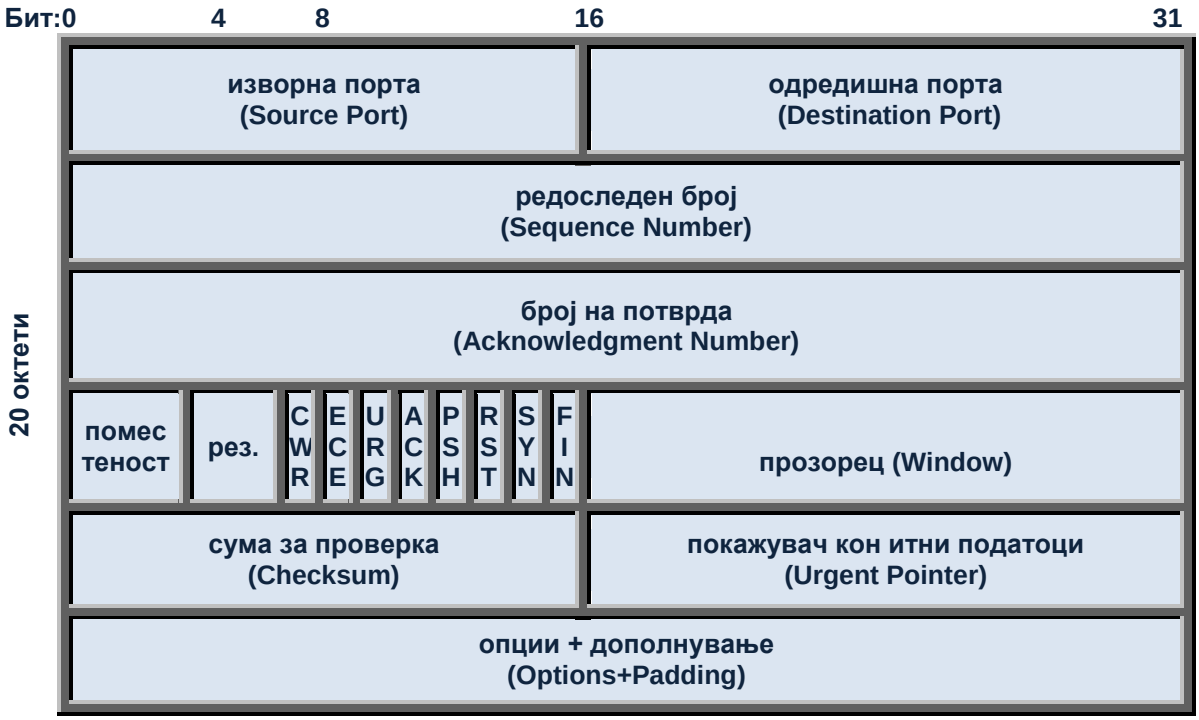
3. Транспортно ниво

Во компјутерски мрежи, транспортното ниво овозможува крај-до-крај комуникациски услуги за апликации во рамките на слоевита мрежна архитектура во протоколите и ги заштитува протоколите од апликациско ниво од деталите на мрежата.

3.1. TCP

Transmission Control Protocol - TCP е протокол ориентиран кон воспоставување на врска, кој овозможува надежна комуникацијата помеѓу два процеса од две машини. TCP обезбедува проверка на грешки, надежна и подредена испорака на октети помеѓу програми кои работат на компјутери поврзани преку една или повеќе мрежи. Ако се открие дека има загуба на податоци или неверодостојност, пакетот се препраќа уште еднаш ако е потребно.

Табела 13. Заглавје на TCP
Table 13. TCP header



TCP прифаќа податоци од протокот за податоци, го дели податокот на помали фрагменти и го додава TCP заглавјето на секој фрагмент, создавајќи TCP

сегменти. Од TCP сегментот се прави IP пакет. Заглавјето има минимална должина од 20 октети, бидејќи треба да ги поддржи сите механизми на протоколот. Во табела 13 е дадено заглавјето на TCP.

Полињата *изворна (Source port*, 16 бита*) и дестинациска (Destination port*, 16 бита*) порта* се користат за идентификација на врската. Изворната порта, изворната адреса, дестинацискатa порта и дестинацискатa адреса се единствени за секоја врска.

Полето *редоследен број* (*Sequence Number*, 32 бита) се користи за идентификација, кои бајти се испратени во протокот на информации. Редоследниот број е првиот податочен октет во овој сегмент, освен кога знаменцето за SYN е поставено на 1. Ако SYN е 1, тогаш ова поле го содржи почетниот редоследен број (ISN) и првиот податочен октет во овој сегмент има редоследен број ISN +1.

Полето *број на потврда* (*Acknowledgment Number*, 32 бита) го содржи редоследниот број на податочен октет кој TCP објектот го очекува следно.

Поле *поместеност на податоците (Data Offset*, 4 бита*)* го покажува бројот на 32-битни зборови во заглавјето, вклучувајќи ги и опциите.

Полето *резервирано* (*Reserved*, 4 бита) е резервирано за идна употреба и не се користи.

Полето *знаменца* (*Flags*, 8 бита) е составено од:

- CWR и ECE се користат за известување и управување со застој.
- URG кога е поставено на 1, значи дека треба да се провери полето *покажувач кон итни податоци*.
- ACK кога е поставено на 1, значи дека треба да се провери полето *број на потврда.*
- PSH кога е поставено на 1, значи дека приманите податоци треба да бидат веднаш препратени до апликацискотo ниво.
- RST кога е поставено на 1, ја ресетира врската.
- SYN кога е поставено на 1, значи дека треба да се синхронизираат редоследните броеви.

- FIN кога е поставено на 1, значи дека испраќачот нема повеќе податоци за испраќање.

Полето *прозорец* (*Window*, 16 бита) го покажува бројот на податочни октети, почнувајќи од редоследниот број кој е наведен во полето за потврда, кој испраќачот е подготвен да го прифати.

Полето *сума за проверка* (*Checksum*, 16 бита) го извршува истиот алгоритам како кај IPv4.

Полето *покажувач кон итни податоци* (*Urgent Pointer*, 16 бита), кога оваа вредност ќе се додаде на редоследниот број на сегментот, го содржи редоследниот број на последниот октет во низата од итните податоци. Тоа му овозможува на примателот да знае колку итни податоци пристигнале. Тоа ретко се користи, но може да се користи за да индицира команди за прекинување.

Полето за *опции* (променлива должина) не е задолжително, а може да се користи за проширувањата на TCP.

Скриени канали во TCP

Складирачки скриени канали

Една од главните намени на TCP протоколот е преуредување на пакетите при пристигнувањето кај примачот и обезбедување на сервис за повторно испраќање на одредени сегменти. Ова се постигнува со полето *редоследен број*. На почетокот се генерира случаен почетен редоследен број (*ISN*) кој се користи во првиот испратен сегмент (SYN сегмент), кога се побарува воспоставување на врска. Примачот нормално одговара со потврдување на приманиот сегмент (SYN/ACK сегмент) и одговара со *ISN*+1 како *број на потврда*. (Rowland, 1997) имплементирал скриени канали во TCP - Covert_TCP, со користење на полињата *почетен редоследен број (ISN)* и *број на потврда*, секое со можност за испраќање на 32 бита по сегмент. Бидејќи случајни вредности се очекуваат за овие полиња, ваквите канали потешко се откриваат. Сепак (Murdoch & Lewis, 2005) развиле шема, Lantra, која генерира ISN за OpenBSD и Linux кој скоро да не може да се разликува од оние генерирани од оригиналниот TCP склад, освен за чувари со знаење на тајниот делен клуч. (Rutkowska, 2004) врз база на Covert_TCP креирала

поробустен надежен канал NUSHU како пасивен скриен канал кој се закачува на постоечка комуникација. NUSHU испраќачот постојано ги менува полињата *редоследен број* и *број на потврда* генерирани од ОС и притоа полето го дели на 3B скриени податоци, 6 бита за секвенционирање и 2 бита како индикатор дали има податоци во податочното поле. Често се испраќаат празни бајти за совпаѓање со нормалниот сообраќај. Притоа ISN се шифрира за да се добие послучајна вредност. Еден начин на детектирање на ваквиот канал со невронски мрежи е даден во (Tumoian & Anikeev, 2005).

(Ahsan, Covert channel analysis and data hiding in TCP/IP, 2002) претставува неколку скриени канали кои се добиваат во редундантноста присутна во некои комбинации на шесте знаменца (URG, ACK, PSH, RST, SYN, FIN). Од 64 можни комбинации, само 29 се валидни. Ако URG битот не е поставен, тогаш како скриен канал може да се користи полето *покажувач кон итни податоци* за испраќање на 16 бита по сегмент (Ahsan, Covert channel analysis and data hiding in TCP/IP, 2002), (Hintz, 2003). Може да се користи и полето *резервирано* за пренос на 4 бита по сегмент (Allix, 2007). Полето *сума за проверка* може да се користи со истиот метод како за IPv4, со хаш колизии (Abad, 2001) за пренос на 16 бита по сегмент.

Во повеќето системи полето *изворна порта* за TCP конекцијата се избира помеѓу 1024 и 65535. Напаѓачот може да го конфигурира ова поле со различни вредности за криење на податоци. Исто така, може да се користи и полето *дестинациска порта*. Доколку не постои сервер кој ја набљудува портата, врската едноставно ќе се ресетира, но ќе бидат пренесени 32 бита по сегмент (Allix, 2007).

(Luo, Chan, & Chang, CLACK: A network covert channel based on partial acknowledgment encoding, 2009) со модификација на *број на потврда* имплементирале складирачки скриен канал Clack, додека (Luo, Zhou, Chan, Chang, & Lee, 2011) имплементирале ACKLeaks скриен канал кој вметнува скриена порака во TCP *ACK* битот во една или во повеќе TCP врски, кој се темели на Twelvefold Way резултат од енумеративна комбинаторика (Stanley, 1997).

Временски скриени канали

Giffin и соработниците (Giffin, Greenstadt, Litwack, & Tibbetts, 2003) предложиле временски скриени канали кои ги користат TCP временските печати, со модифицирање на нивниот низок бит. Овој метод го забавува TCP протокот така што временските печати на сегментите се валидни кога се испраќаат. Може да се креираат скриени канали со 1-бит-по-сегмент со споредување на нискиот бит на секој временски печат на TCP со тековниот бит на пораката. Доколку тие се совпаѓаат, сегментот веднаш се испраќа со генериран TCP временски печат, во спротивно е одложен за една временска единица и временскиот печат на TCP се зголемува за 1.

Авторите на (Chakinala, и др., 2006) ја проширале идејата на (Ahsan, Covert channel analysis and data hiding in TCP/IP, 2002) и (Ahsan & Kundur, Practical data hiding in TCP/IP, 2002) и формирале временски скриен канал со сортирање на TCP сегменти и користење на полето *редоследен број*. Авторите формализирале и различни модели за овој тип на скриени канали.

(Luo, Chan, & Chang, Cloak: A ten-fold way for reliable covert communications, 2007) претставуваат класа на временски скриени канали, познати како Cloak, кои ја енкодираат пораката со уникатна дистрибуција на N сегменти врз X TCP текови, користејќи 10 различни методи на енкодирање и симулирање на нормален TCP сообраќај. Нова порака се праќа само по потврдување на претходната. Овој канал се темели на Twelvefold Way резултат од енумеративна комбинаторика (Stanley, 1997).

Еден друг временски канал со имплементација TCP-Script (Luo, Chan, & Chang, TCP Covert Timing Channels: Design and Detection, 2008) вгнездува тајна порака во TCP рој (burst) од сегменти, на тој начин што во одреден временски интервал на енкодирање, T_e, ако пораката е $m \in [1, M]$, каде M е претходно договорено од двете страни, се испраќа TCP рој од m сегменти. Две соседни пораки се одделени со утврден временски интервал T_p.

Табела 14 ги прикажува сумарно различните стеганографски техники кај TCP.

Табела 14. Стеганографски техники за TCP
Table 14. Steganographic techniques for TCP

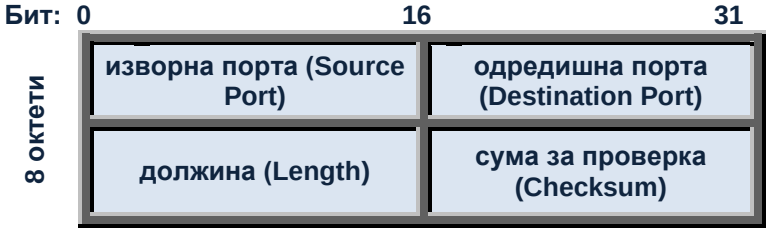
Статија	Зафатени полиња/ техника	RBR (bps)	PRBR (bitovi/ segment)	Тип
(Rowland, 1997)	*редоследен број, број на потврда*		32+32	складирачки, просторен канал базиран на вредност
(Ahsan, 2002) (Hintz, 2003)	*покажувач кон итни податоци*		16	складирачки, просторен канал базиран на вредност
(Abad, 2001)	*сума за проверка*		16	складирачки, просторен канал базиран на вредност
(Allix, 2007)	*резервирано*		4	складирачки, просторен канал базиран на вредност
	изворна порта дестинациска порта		16+16	складирачки, просторен канал базиран на вредност
(Luo, Chan, & Chang, 2009)	*број на потврда*			складирачки, просторен канал базиран на вредност
(Luo, Zhou, Chan, Chang, & Lee, 2011)	*ACK*			складирачки, просторен канал базиран на вредност
(Giffin, Greenstadt, Litwack, & Tibbetts, 2003)	временски печати		1	временски, базиран на вредност
(Chakinala, и др., 2006)	*редоследен број* сортирање на сегменти	$k \cdot log_2 n!$, каде kn е број на испратени сегменти во 1 секунда, а n е бројот на сегменти што се сортираат		временски, базиран на транзиција
(Luo, Chan, & Chang, 2008)	испраќање на рој од сегменти	*n* каде *n* е бројот на $T_e + T_p$ временски интервали во 1 секунда		временски, базиран на транзиција, со бројач

UDP

User Datagram Protocol (UDP) е транспортен протокол кој нуди ненадежен сервис без воспоставување на врска, односно не е гарантирана испораката на пакетите и не се спречува појава на дупликати. Особено е погоден за денешните апликации во реално време. Временски осетливите апликации често користат UDP, бидејќи отфрлањето на пакети е подобро од чекањето за препраќање на пакет, што може да не постои како опција во системите кои работат во реално време. Протоколот е дизајниран од страна на David P. Reed во 1980 година и формално е дефиниран во RFC 768 (Postel, User Datagram Protocol, 1980).

Заглавјето на UDP се состои од 4 полиња, од кои секое е 2 бајти (табела 15). Употребата на полињата за проверка и изворната порта е задолжително во IPv4. Во IPv6 задолжителна е само изворната порта.

Табела 15. Заглавје на UDP
Table 15. UDP header



Полињата *изворна порта (Source Port)* (16 бита) и *одредишна порта (Destination Port)* (16 бита) ги идентификуваат изворната и одредишната порта за дадена комуникација.

Полето *должина (Length)* (16 бита) ја одредува должината во бајти на UDP заглавјето заедно со податоците. Практичната граница за должината на податоците што се изрекува од страна на основниот IPv4 протокол е 65.507 бајти (65.535 - 8 бајти UDP заглавје - 20 бајти IP заглавје). Во IPv6 постојат UDP пакети со големина поголема од 65.535 бајти. RFC 2675 одредува дека полето за должината е наместено на нула, ако должината на UDP заглавјето плус UDP податоци е поголемо од 65.535.

Полето *сума за проверка (Checksum)* (16 бита) го користи истиот алгоритам како кај TCP и IP.

Скриени канали во UDP

UDP нуди 3 полиња за пренос на скриени информации: *изворна порта*, *должина* и *сума за проверка*. Изворната порта може да биде изменета само во случај на динамичен NAT (Thyer, 2008). Користењето на сите полиња овозможува пренесување до 6 бајти по пакет, без гаранција за сигурноста и интегритетот на комуникацијата. Меѓутоа, овој недостаток може да се претвори во предност, бидејќи таквиот сообраќај се открива потешко.

Полето *сума за проверка* не е задолжително, па со неговото присуство или отсуство на заглавјето може да се пренесе еден бит во UDP пакет (Fisk, Fisk, Papadopoulos, & Neil, 2002). Истото поле може да се искористи за криење на податоци со користење на хеш колизии (Abad, 2001). За UDP постојат само складирачки скриени канали (табела 16).

Табела 16. Стеганографски техники за UDP
Table 16. Steganographic techniques for UDP

Статија	Зафатени полиња/ техника	RBR (bps)	PRBR (bitovi/segment)	Тип
(Fisk, Fisk, Papadopoulos, & Neil, 2002)	*сума за проверка*		1	складирачки, просторен канал базиран на вредност
(Abad, 2001)	*сума за проверка*		16	складирачки, просторен канал базиран на вредност
(Thyer, 2008)	*изворна порта, должина сума за проверка*		48	складирачки, просторен канал базиран на вредност

3.3. SSL/TLS

Transport Layer Security (TLS) и неговиот претходник, Secure Sockets Layer (SSL), се криптографски протоколи кои обезбедуваат комуникациска безбедност преку Интернет. TLSv1 е дефиниран во RFC 2246. Тие користат асиметрична криптографија за размена на клучеви, симетрични шифри за доверливост и кодови за автентикација за интегритетот на пораката. Неколку верзии на протоколите се во широка употреба кај апликациите, како што се веб-

прелистувачите, електронската пошта, праќање факсови преку Интернет, инстант пораки и Voice-over-IP (VoIP). Со користење на алатки како OpenSSL, SSL автентикацијата и енкрипцијата можат да бидат вградени во речиси секој тип на апликација.

TLS и SSL работат во основното транспортно ниво, со сегменти кои носат шифрирани податоци. Го користат сервисот што го нуди TCP, а најчесто нивната услуга ја користи HTTP протоколот и тогаш се користи портата 443.

Бидејќи протоколите можат да работат со или без TLS (или SSL), потребно е клиентот да му соопшти на серверот дали сака да воспостави TLS врска или не. Постојат два главни начини за постигнување на ова, едната опција е да се користи различен број на порта за TLS конекција (на пример порта 443 за https). Другата опција е да се користи регуларниот број на портата и да постои барање од клиентот дека серверот ја префрла конекцијата во TLS со користење на посебен механизам за протоколот (на пример START TLS за е-маил протоколот). Откако клиентот и серверот решиле да го користат TLS, тие преговараат за постојана врска со ракување. Во текот на овој процес, клиентот и серверот се договараат за различни параметри, кои се користат за да се воспостави безбедноста на конекцијата.

Дефинирани се три протоколи кои се дел од повисокото ниво на SSL: протоколот за ракување (Handshake Protocol), протоколот за спецификација на промена во шифрата (Change Cipher Spec Protocol) и протоколот за тревожење (Alert Protocol).

Скриени канали во SSL/TLS

(Anjan & Abraham, Behavioral Analysis of Transport Layer Based, 2010), (Anjan, Abraham, & Jadhav, Design of Transport Layer Based Hybrid Covert Channel Detection Engine, 2010) предложиле хибриден скриен канал на транспортно ниво, кој се состои од TCP скриен канал со шум - NCT и прикриен (subliminal) канал во SSL/TLS - SCSL. Конкретно за прикриениот канал може да се искористи кој било алгоритам од SSL/TLS кој користи генерирање на случајни броеви, како на пример Ong-Shamir шемата за дигитални потписи.

4. Апликациско ниво

Апликациското ниво е ниво од TCP/IP складот со протоколи каде што се наоѓаат сите кориснички апликации. Сепак, и на ова ниво има потреба од протоколи за поддршка на корисничките апликации, а некои од нив, како HTTP, FTP, DNS се погодни за примена на техниките на мрежна стеганографија.

4.1. HTTP

Hypertext Transfer Protocol (HTTP) е протокол за размена или пренос на хипертекст, аудио, слики, обичен текст и други видови на податоци. HTTP претставува темел на комуникацијата со податоци за World Wide Web. HTTP за првпат е дефиниран во 1991 година, додека најкористената верзија на HTTP/1.1 е дефинирана во 1999 година во RFC 2616 (Fielding, и др., 1999).

HTTP е клиент/сервер протокол кој работи на принцип барања/одговори и е без состојба. Најтипична употреба е кога комуницираат веб-прелистувач и веб-серверот. Клиентот испраќа порака со HTTP барање до серверот. Серверот, кој обезбедува ресурси како што се HTML датотеки и други содржини, или врши други функции во име на клиентот, враќа порака со одговор до клиентот. Одговорот содржи информации за статусот на барањето, а ги содржи и бараните содржини во главниот дел на пораката.

HTTP сесија е низа од мрежни трансакции со барање-одговор. HTTP клиентот иницира барање (најчесто со GET или POST методата) со воспоставување на конекција преку TCP обично на портата 80. HTTP серверот кој слуша на таа порта, по приемот на барањето, испраќа статусната линија, како што е "HTTP/1.1 200 OK" и сопствена порака. Телото на оваа порака го содржи бараниот ресурс, иако може да биде вратена порака за грешка или други информации.

Скриени канали во HTTP

HTTP е дизајниран да овозможи висок капацитет и ниска латентност, а тоа се идеалните услови за креирање на канал за скриена комуникација. Исто така, дури и најрестриктивните организации го пропуштаат HTTP сообраќајот. Пораките за HTTP барање може да содржат заглавје со повеќе полиња, како *User-agent* и

Referer. Малициозните корисници може да го експлоатираат ова со користење на заглавја за пренос на произволни податоци. Особено интересна карактеристика на HTTP протоколот е телото на објектот. Тоа нормално се користи само во HTTP POST барањата, бидејќи нема вистинска потреба за другите видови на барања. Сепак, во спецификацијата на протоколот не е наведено дека не треба да биде присутно и во други типови на барања. Исто како пораките за HTTP барање, пораките за HTTP одговор можат да содржат заглавје со повеќе полиња. Не постои ограничување од страна на протоколот за големината на HTTP заглавјето или телото на пораката, но сепак веб-серверите воведуваат одредени ограничувања. Така на пример, Apache серверите прифаќаат HTTP заглавја со големина до 8KB, а IIS од 8KB до 16KB во зависност од верзиите.

Денес многу Тројанци и бот мрежи користат HTTP скриени канали за комуникација со нивните автори.

(Bowyer, 2002) вовел основна HTTP стеганографија како начин да се пренесат скриени пораки со тројански коњи, кои ги поминуваат постојаните заштитни ѕидови со состојби. Bowyer предложил користење на HTTP за испорака на украдениот материјал и сугерира приложување на скриената порака на крајот од HTTP GET барањето до веб-серверот кој е контролиран од страна на напаѓачот. Лажниот веб-сервер ќе го отфрли URL сегментот на GET барањето и ќе го анализира само делот од пораката кој следи по барањето. За извлекување на скриената содржина која се наоѓа внатре во заштитената мрежа, Bowyer предлага веб-страница со нормален изглед која содржи скриени пораки во вгнездените слики.

Алатката Reverse WWW Shell демонстрира колку ефективен може да биде HTTP скриен канал во испораката на корисниот товар (vanHauser, 1999). HTTP барањето е всушност повик до мастер програма која слуша за конекции од клиентска роб програма. Во мастер конзолата се внесуваат команди и се извршуваат во робот. Со HTTP 200/OK одговорот се маскираат шел командите до робот. Ова е подложно на детекција од анализаторите на сообраќај.

Bauer (Bauer, 2003) предложил протоколот наречен "Muted Posthorn", кој ја елиминира комуникацијата помеѓу испраќачот и примачот на пораката. Во овој

протокол пораките се испорачуваат од еден на друг веб-сервер преку несомнителни веб-прелистувачи со стандардни HTTP механизмите. Протоколот користи пет HTTP / HTML карактеристики: пренасочувања, колачиња, Referer полето, HTML елементи и активната содржина. HTTP пренасочувачот му соопштува на веб-пребарувачот дека бараниот документ е достапен на друга локација. Оваа локација може да е URL на скрипта со листа од параметри. Максималниот капацитет е 1024B. Колачињата може да се искористат за пренос на тајна порака меѓу два сервера во ист домен од второ ниво, со тоа што се користи Set-Cookie командата и пар клуч-вредност со големина со 4KB за колаче. Исто така, и Referer полето има лимит на пренос до 1024B. HTML елементите предизвикуваат веб прелистувачите автоматски до побаруваат објекти од веб-сервери, без знаење на корисникот. Активните содржини може да предизвикаат редирекција и POST барање без знаење на корисникот.

(Dyatlov & Castro, 2003) предлага складирачки скриени канали со користење на заглавјето и/или телото на HTTP барањето/одговорот. Kwecka (Kwecka, 2006) предложил криење на податоците во HTTP употребувајќи го фактот дека HTTP го третира која било низа на последователни знаци како carriage return line feed [CLRF], празни места [SP] и табови [HT] присутна во заглавјето на ист начин како единствен знак за празно место. Така, на пример, [HT] може да се користи како бинарна 1, а [SP] како бинарна 0 и за преставувањето на 1 ASCII знак може да се користат 8 битови.

(Alman, 2003) покажал како поради слабост во HTTP CONNECT методот, прокси серверите може слепо да пренесат и други информации освен HTTP сообраќајот, па дури и да креираат VPN сообраќај.

Eßer и Freiling (Eßer & Freiling, 2005) предложиле временски скриен канал користејќи HTTP, во кој веб серверот испраќа скриени податоци на клиентот со одложување на одговорот, што би било бинарна 1 или со одговарање веднаш, што би било бинарна 0. Castro (Castro, 2006) предлага користење на HTTP колачињата за создавање на скриени канали во HTTP. (Van Horenbeeck, 2006) предлага тунелирање со HTTP Entity Tag тагот кој му овозможува на клиентот да верифицира дали локално кеширана копија сè уште е тековна и за таа цел развил

алатка Wondjina. Авторот размислувал и за Content-MD5 полето во заглавјето за пренос на податоци, но со него скриениот канал е ограничен на 128 бита по HTTP порака.

Инфранет (Infranet) е работна рамка за користење на HTTP скриени канали за заобиколување на цензурирањето на Интернет (Feamster, Balazinska, Harfst, Balakrishnan, & Karger, 2003). Веб-серверите кои се вклучени во Инфранет добиваат HTTP барања за безопасни веб-странисi, а ја враќаат содржината со криење на цензурираните податоци во нецензурирани слики со користење на стеганографски техники.

Padgett (Padgett, 2001) развил алатка за тунелирање на SSH преку HTTP прокси. (LeBoutillier, 2005) имплементирал алатки за тунелирање на UDP или TCP преку HTTP.

Без постојана врска помеѓу клиентот и серверот проверените податоци не можат да се вклопат во постојниот сообраќај. Сепак, податоците можат да бидат вградени во заглавјата на протоколот од испратените пакети. Техниката која се употребува е port knocking, каде што податоците се кодирани во броевите на UDP портата (Krzywinski, 2004). Броевите на портата секогаш се евидентирани од страна на заштитниот sиd, додека полињата за опциите не се документираат секогаш (deGraaf, Aycock, & Jacobson, 2005). Клиентот на port knocking може да биде едноставна непривилигирана корисничка апликација, па така серверите немаат потреба од способности за фаќање на пакет над основната поддршка за логирање во софтверот на заштитниот sиd. Серверите на port knocking имаат предност бидејќи се невидливи за скенерите на портата и размената на port knocking изгледа како скенирање на портата за обичните аналитичари на сообраќајот.

Поважните стеганографски техники кои се потпираат на HTTP и нивните карактеристики се прикажани во табела 17.

Табела 17. Стеганографски техники за HTTP
Table 17. Steganographic techniques for HTTP

Статија	Зафатени полиња/ техника	RBR (bps)	PRBR (bitovi/PDU)	Тип
(Dyatlov & Castro, 2003)	заглавје и/или тело на HTTP барањето/одговорот		зависи од веб-серверите < 8190 *8 за заглавјето	складирачки, просторен канал базиран на вредност
(Kwecka, 2006)	низа на последователни знаци како line feed [CLRF], празни места [SP] и табови [HT] во заглавјето		зависи од веб-серверите < 8190 *8	складирачки, просторен канал базиран на вредност
(Bauer, 2003)	редирекција		< 1024*8	складирачки, просторен канал базиран на вредност
	Referer		< 1024*8	складирачки, просторен канал базиран на вредност
	колачиња *Set-Cookie*		<4096*8	складирачки, просторен канал базиран на вредност
	HTML елементи и активни содржини			складирачки, просторен канал базиран на вредност
(Van Horenbeeck, 2006)	*Content-MD5*		128	складирачки, просторен канал базиран на вредност
	HTTP Entity Tag			складирачки, просторен канал базиран на вредност
(Castro, 2006)	колачиња			складирачки, просторен канал базиран на вредност
(Eßer & Freiling, 2005)	со одложување или неоложување на одговорот од веб-серверот	*n*, каде *n* е бројот на испратени одговори од веб серверот во 1 секунда		временски, базиран на транзиција

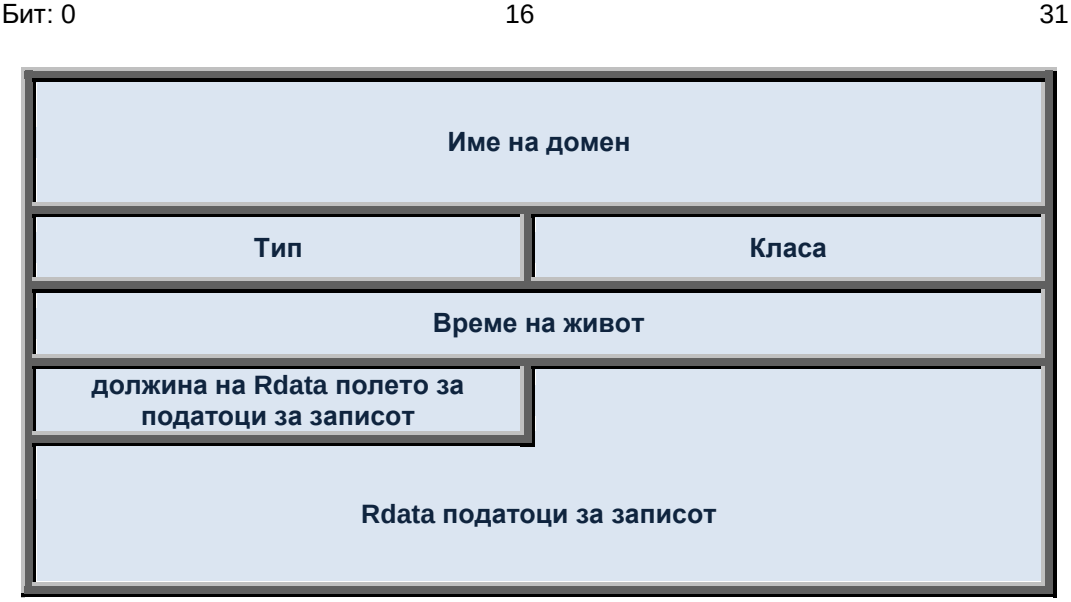
4.2. DNS

Domain Name System – DNS е сервис за пребарување кој обезбедува мапирање помеѓу доменско име на даден хост и неговата IP адреса (RFC 1034 и RFC 1035). DNS е протокол базиран на барања и одговори, додека комуникацијата се одвива со употреба на пораки. Овие пораки содржат заглавје на пораката, кое е присутно во двете пораки за барање и одговор. DNS се заснова на хиерархиска база на податоци која содржи записи на ресурси (resource records - RRs) со името, IP адресата и дополнителните информации за домаќините. Податоците кои се испраќаат од клиентот до серверот вообичаено се енкодирани во името со Base64 или Base32.

DNS одговорите традиционално не се криптографски потпишани, што доведува до многу можности за напад. Како одговор на ова е креиран Domain Name System Security Extensions (DNSSEC), кој го модифицира DNS за додавање на поддршка за криптографски потпишани одговори.

DNS е чест избор за креирање на тунел со корисен товар, поради тоа што е еден од најмалку филтрираните протоколи во TCP/IP складот со протоколи. Од посебен интерес за креирање на скриени канали се DNS записите за ресурси (Resourse Record - RR), чиј формат е даден на табела 18.

Табела 18. Формат на DNS запис за ресурс
Table 18. Format of DNS resurs record



Полето *име на доменот* (*Domain Name*) – има променлива должина. Иако синтаксата на имињата на домените во пораките е прецизно дефинирана, формата на името на доменот во записот за ресурси е опишана во општи црти. Во суштина, името на доменот во RR мора да соодветствува со формата разбирлива за човекот, која се состои од низа од ознаки со алфанумерички знаци или цртички и секоја двојка ознаки е разделена со точка.

Поле *тип* (*Type,* 16 бита) - го идентификува типот на ресурсот во овој RR.

Поле класа (*Class,* 16 бита) - ја идентификува фамилијата на протоколи. Единствената вообичаено користена вредност во употреба е IN, за интернет.

Поле *време на живот* (*Time to Live,* 32 бита) - обично кога еден запис за ресурс се презема од сервер за имиња, оној кој го побарал записот ќе го стави записот во кеш за да не мора постојано да го прашува серверот за истиот запис. Ова поле го одредува временскиот интервал за време на кое записот може да се чува во кеш меморијата пред повторно да мора да се праша изворот на информацијата.

Поле за *должина на податоците во записот* (*Rdata Field length,* 16 бита) - должина на полето за податоци во записот во октети.

Поле *податоци во записот* (*Rdata*) - низа од октети со променлива должина која го опишува ресурсот. Форматот и должината на оваа информација варира во зависност од типот на записот за ресурс.

Скриени канали во DNS

DNS е подложен на скриени канали кои вршат тунелирање на други протоколи, како на пример IP преку DNS (Табела 19). Особено интересни за пренос на скриени податоци се NS и TXT записите кои може да бидат долги до 255B. Постои и експериментален NULL запис, со должина до 65535B, но во актуелните имплементации должината му изнесува од 300B до 1200B.

NSTX (Nameserver Transfer Protocol) овозможува токму тунелирање на IPv4 преку DNS, но оваа C имплементација има многу грешки (savannah.nongnu.org, 2002). NSTX ги користел TXT записите без енкодирање за симнување и base64 за качување на податоци, а IP пакетите ги делел во парчиња (chunks).

(Kryo, 2010) имплементирал алатка iodine во C за тунелирање на IPv4 преку DNS. Таа го користи NULL типот на записи за симнување без енкодирање и екстензијата EDNS0 на DNS која дозволува користење на пакети подолги од лимитот од 512B избран во RFC 1035. Други типови на записи кои може да се користат се TXT, SRV, MX, CNAME и A, дадени во опаѓачки редослед според бројот на битови кои може да го пренесат. Потребни се клиентска и серверска апликација, и притоа корисникот треба да контролира реален домен и да има сервер со јавна IP адреса на кој ќе се извршува iodine и на кој ќе се делегира подомен. Нагорниот сообраќај е зипуван и енкодиран со Base32 или Base64 во поле *име на домен*.

DNSCat (Pietraszek, 2004) е Java алатка која овозможува бидирекционално тунелирање на IPv4 преку DNS, а во комбинација со PPP може да креира виртуелна приватна мрежа. За надолниот сообраќај ги користи CNAME записите (до 255B) и не бара специјална конфигурација на јадрото на ОС.

Perl алатката OzymanDNS (Kaminsky D. , 2004) енкапсулира TCP проток или SSH во DNS пораки. Користи Base32 енкодирани прашалници, TXT тип на запис и EDNS0 механизмите за екстензија на DNS. Кај овој тип на алатка проблем е што надежен протокол треба да се тунелира преку ненадежен протокол, па апликацијата треба да имплементира повторно испраќање на изгубените DNS пакети. DNS2TCP (Dembour & Collignon, 2008) е уште една алатка за тунелирање на TCP проток или SSH во DNS пораки. Користи TXT записи и Base64 енкодирање.

(Nussbaum, Neyron, & Richard, 2009) имплементираат алатка TUNS (во Ruby) за тунелирање на IP преку DNS. Нагорниот и надолниот сообраќај одат само преку CNAME записите (до 255B), затоа што TXT и NULL записите ретко се користат во реални ситуации. Алатката не ги дели IP пакетите, туку работи со помал MTU од 140B. Авторите ги тестирале сите претходни алатки во различни реални средини и се покажало дека само нивната алатка работи во сите случаи.

(Merlo, Papaleo, Veneziano, & Aiello, 2011) направиле споредбена анализа на перформансите на сите овие скриени канали.

(Anonymous, 2005) предложиле скриени канали со DNS протоколот со користење на негативното кеширање на имиња на хостовите. За домен што не постои, доколку одговорот е во кешот се испраќа NXDOMAIN, а инаку се испраќа NOERROR.

Табела 19. Стеганографски техники за DNS
Table 19. Steganographic techniques for DNS

Статија	Зафатени полиња/ техника	RBR (bps)	PRBR (bitovi/PDU)	Тип
NSTX (savannah.nongnu.org, 2002) OzymanDNS (Kaminsky D. , 2004) DNS2TCP (Dembour & Collignon, 2008)	TXT записи		< 255*8	складирачки, просторен канал базиран на вредност, тунел во корисен товар
iodine (Kryo, 2010)	Нагорен сообраќај *име на домен*		< 255*8	складирачки, просторен канал базиран на вредност, тунел во корисен товар
	Надолен сообраќај NULL, TXT, SRV, MX, CNAME или A типови на записи		< 300*8 до 1200*8 за NULL и зависи од имплементацијата	складирачки, просторен канал базиран на вредност, тунел во корисен товар
DNSCat (Pietraszek, 2004)	Надолен сообраќај CNAME записи		< 255*8	складирачки, просторен канал базиран на вредност, тунел во корисен товар
TUNS (Nussbaum, Neyron, & Richard, 2009)	CNAME записи		< 255*8	складирачки, просторен канал базиран на вредност, тунел во корисен товар
(Anonymous, 2005)	негативно кеширање		1	временски, базиран на вредност

Ако се користи нерекурзивен прашалник нема да се промени состојбата на кешот. За да се формира скриен канал, два хоста треба да се на иста мрежа и да го прашуваат DNS серверот за неколку поддомени кои сигурно не постојат. Ако

поддоменот е во кешот, се третира како бинарна 1, инаку ако не е, се третира како бинарна 0. Овој канал лесно може да се отстрани со оневозможување на кеширањето за дадената зона, со MINIMUM полето во SOA записот на 0.

4.3. FTP

File Transfer Protocol - FTP е протокол за размена на датотеки од еден систем на друг преку кориснички команди (официјална спецификација е RFC 959). Можат да се испраќаат и текстуални и бинарни датотеки, а протоколот обезбедува и начин за контрола на кориснички пристап. Кога еден корисник ќе посака да започне пренос на датотеки, FTP воспоставува TCP врска со целиот систем за размена на контролни пораки. Оваа врска овозможува пренос на корисничката идентификација (user ID) и му овозможува на корисникот да ја наведе датотеката и посакуваните акции врз неа. Штом се одобри преносот на датотеката, се воспоставува втора, податочна врска, без дополнителните трошоци за заглавја или контролни информации на апликациско ниво. Штом е завршен преносот, контролната врска се користи за да се сигнализира завршувањето и да се прифатат нови команди за пренос на датотеки.

Скриени канали во FTP

Zou и соработниците (Zou, Li, Sun, & Niu, 2005) предложиле два скриени канали во FTP (табела 20). Првиот ги енкодира скриените битови директно во FTP командите, на тој начин што, ако има N команди, секоја команда ќе претставува $log_2 N$ битови. NOOP командата се користи за да се заштити FTP контролната врска од истекување на времето и серверот на неа одговара со ресетирање на тајмерот и со порака 200 Command okay. Вториот скриен канал го варира бројот на FTP NOOP командите, кои се испратени во текот на неактивните периоди. Пред испраќање на секој бајт, се испраќа ABOR команда за означување на следниот бајт од скриената порака. Бројот на испратените NOOP команди е еднаков на целобројната вредност на скриените податоци и се движи од 0 до 255. Ова значи дека ако сакате да го испратете бајтот i, се испраќаат i NOOP комади и една ABOR команда.

Табела 20. Стеганографски техники за FTP
Table 20. Steganographic techniques for FTP

Статија	Зафатени полиња/ техника	RBR (bps)	PRBR (bitovi/PDU)	Тип
(Zou, Li, Sun, & Niu, 2005)	FTP команди		за 32 команди, 5 бита/команда	складирачки, просторен канал базиран на транзиција
	i NOOP команди и 1 ABOR команда			временски, базиран на транзиција, со бројач

4.4. RTP

Real-time Transport Protocol – RTP е протокол за испорака на аудио и видео во IP мрежите, дефиниран во 1996 година во RFC 1889 (Schulzrinne, Casner, Frederick, & Jacobson, 1996). Во апликациите како глас преку IP (Voice over IP – VoIP), онлајн игри со повеќе играчи, проток во живо на аудио и видео, проток на складирано аудио и видео, вообичаено аудиото и видеото се пренесуваат во различни текови со RTP, но и со поддршка на контролниот протокол кој оди со него, RTP Control Protocol - RTCP. RTP најдобро може да се објасни како работна рамка која апликациите може директно да ја користат за да имплементираат протокол. Без информации за апликацијата која ќе го користи RTP, што значи дека не е целосен протокол. Од друга страна, RTP наметнува структура и дефинира вообичаени функции за индивидуалните апликации во реално време, за да не мора да се оптоваруваат со нивно дефинирање. Еден дел од RTP функционира на транспортно ниво над UDP, а друг дел функционира на апликациско ниво.

Секој RTP пакет содржи фиксно заглавје и може да содржи дополнителни полиња од заглавјето специфични за апликацијата. Табела 21 го покажува фиксното заглавје. Првите 12 октети се секогаш присутни и се состојат од следните полиња:

Поле за *верзија* (Version, 2 бита) - тековната верзија е 2.

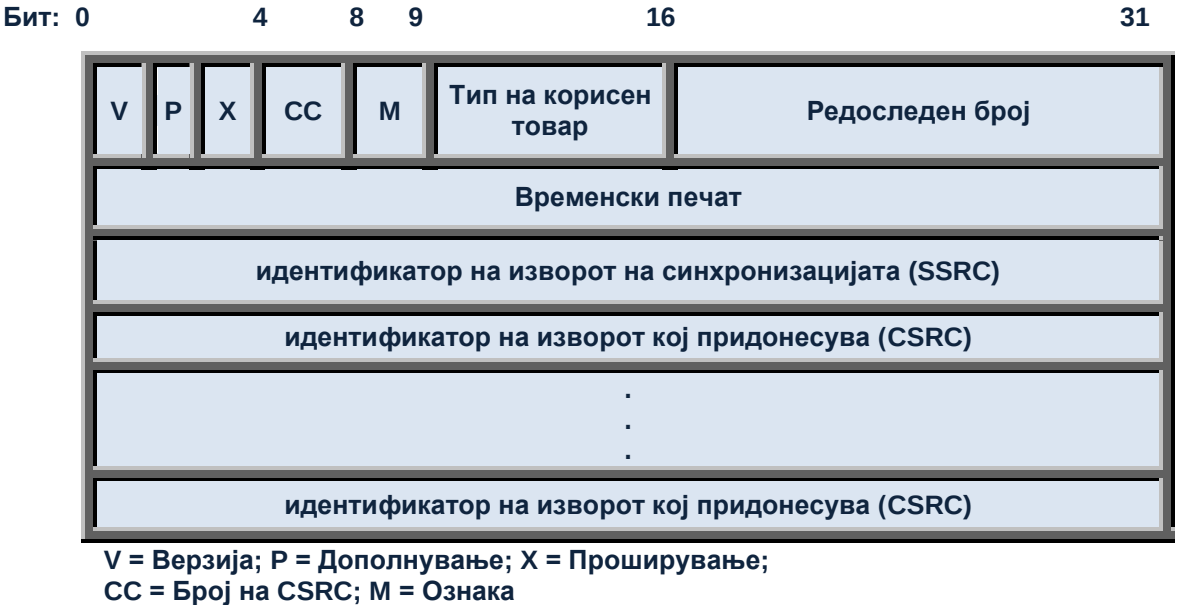
Поле за *дополнување – P бит* (Padding, 1 бит) - означува дали на крајот од корисниот товар се појавуваат октети за дополнување. Ако е така тогаш последниот октет од корисниот товар го содржи бројот на октети за дополнување.

Дополнувањето се користи кога апликацијата бара корисниот товар да биде со должина на целоброен производ на некоја вредност.

Поле за *проширување - X бит* (Extension, 1 бит) - ако е поставено на 1, тогаш по фиксното заглавје следи точно едно заглавје за проширување, кое се користи за експериментални проширувања на RTP.

Поле за *CSRC бројач* (CSRC count, 4 бита) - бројот на идентификатори на извори кои придонесуваат (contributing source - CSRC). Тие следуваат по фиксното заглавје.

Табела 21. Формат на RTP заглавје
Table 21. Format of RTP header



V = Верзија; P = Дополнување; X = Проширување;
CC = Број на CSRC; M = Ознака

Поле за *ознака* (Marker, 1 бит) - се користи за означување на граница од текот на податоци и зависи од типот на корисен товар. За видео, тој се поставува да го означи крајот на кадарот. За аудио, тој се поставува да го означи почетокот на изблик на говор.

Поле за *тип на корисен товар* (Payload Type, 7 бита) - го идентификува форматот на корисен товар во RTP пакетот (заедно со употребата на компресија или шифрирање), кој следи по RTP заглавјето. Во стабилна состојба, еден извор треба да користи само еден тип на корисен товар за време на сесијата, но може

да го смени типот на корисниот товар како одговор на промена во условите, која е откриена од RTCP.

Поле за *редоследен број* (Sequence Number, 16 бита) - овозможува подредување на пакетите во низа од пакети со ист временски печат и детекција на загуба. Секој извор започнува со случаен редоследен број, кој со секој испратен RTP пакет се зголемува за еден. Доколку последователните пакети се логички генерирани во исто време, можат да го имаат истиот временски печат: на пример, неколку пакети кои припаѓаат на истиот видеокадар.

Поле за *временски печат* (Timestamp, 32 бита) - го означува моментот на генерирање на првиот октет на податоците во корисниот товар. Временската единица на ова поле зависи од типот на корисниот товар.

Поле за *идентификатор на изворот на синхронизацијата* (Synchronization Source Identifier – SSRC) - случајно генерирана вредност која уникатно го идентификува изворот во сесијата.

RTP протоколот за пренос на податоци се користи само за пренос на кориснички податоци, обично на мултикаст начин помеѓу учесниците во сесијата. RTCP, исто така, работи на мултикаст начин за да обезбеди повратна врска до RTP изворите на податоци, како и до учесниците во сесијата. Еден RTCP пренос се состои од неколку посебни RTCP пакети групирани во еден UDP датаграм. Во RFC 1889 дефинирани се пет типови на RTCP пакети, и тоа: извештај на испраќачот, извештај на примачот, опис на изворот, збогум и пакет специфичен за апликацијата. За сесии со мал број на учесници, интервалот меѓу последователни RTCP пораки е пет секунди и RTCP комуникацијата не треба да надминува 5% од вкупната комуникација во дадена сесија. Дополнително, постои и безбедна верзија на RTP, наречена Secure RTP, дефинирана во RFC 3711 (Baugher, McGrew, Naslund, Carrara, & Norrman, 2004), во која деловите од RTP пакетот може да се шифрираат и автентицираат со *таг за автентикација*.

Скриени канали во RTP и RTCP

Од анализата се исклучени сите скриени канали кои го користат корисниот товар на RTP пакетот како скриен канал, освен LACK, поради неговата хибридна природа.

Mazurczyk и Szczypiorski (Mazurczyk & Szczypiorski, Steganography of VoIP Streams, 2008) објаснуваат како да се создадат скриени канали во RTP и RTCP, на неколку начини:

- Ако *P* битот е поставен на 1, може да има 1 или повеќе бајти на крајот од корисниот товар, со тоа што последниот бајт го дава бројот на дополнителни бајти заедно со него кои треба да се игнорираат при приемот. Ова ја ограничува големината на ваквиот скриен канал на 255B по RTP пакет;
- Ако *X* битот е поставен на 1, по фиксното RTP заглавје следува продолженото заглавје со променлива должина, кое може да се искористи за скриена порака. Продолженото заглавје е множество на продолжени елементи кои имаат 4-битен *идентификатор* и 4-битна *должина*. Бидејќи *идентификатор* со вредност 0 и 15 се резервирани, можни се најмногу 14 продолжени елементи, секој со максимална должина од 16 бајти. Овој скриен канал има PRBR од 224B;
- Случајно генерираните почетни вредности на *редоследен број* и *временски печат* во првиот RTP пакет;
- Со примена на методата од (Giffin, Greenstadt, Litwack, & Tibbetts, 2003) за создавање на скриени канали со еден бит по RTP пакет со помош на најмалку значајниот бит од полето *временски печат* или од полето *NTP временски печат* од RTCP;
- Може да се добие скриен канал со 160 битови по пакет од блоковите за извештај кои вклучуваат повеќе полиња, во извештајот од примачот и извештајот од испраќачот во RTCP;
- Со полиња за безбедносни механизми во безбедниот RTP или во RTCP, како 80-битниот *таг за автентикација*. Само примачот кој го има клучот може да определи дали тагот за автентикација е погрешен и е можен скриен канал или не. Вообичаена пракса е RTP пакетите со погрешен *таг за автентикација* да се отфрлаат.

Авторите од истиот труд и (Mazurczyk & Lubacz, LACK—a VoIP steganographic method, 2010) укажуваат на еден метод наречен LACK (Lost Audio

Packets Steganographic Method) за создавање на скриени канали, со користење на намерно одложување (а со тоа и загуба) на товарите на пакетите. Товарот на намерно одложените пакети се користи за пренос на тајни информации до приемачите кои се свесни за оваа постапка. Покрај тоа, субјектите кои комуницираат мора да го земат предвид прифатливото ниво на загуба на пакети за IP телефонијата и да не го надминуваат. Првата практична евалуација на овој метод е дадена во (Mazurczyk W. , Lost Audio Packets Steganography: The First Practical Evaluation., 2012).

Табела 22. Стеганографски техники за RTP, RTCP, SRTP
Table 22. Steganographic techniques for RTP, RTCP, SRTP

Статија	Зафатени полиња/ техника	RBR (bps)	PRBR (bits/packet)	Тип
(Mazurczyk & Szczypiorski, Steganography of VoIP Streams, 2008)	*P* = 1, и користење на дополнителни бајти		255*8	складирачки, просторен канал базиран на вредност
	X = 1, продолженото заглавје		224*8	складирачки, просторен канал базиран на вредност
	редоследен број временски печат		32+32 но само во првиот RTP пакет	складирачки, просторен канал базиран на вредност
	блок со извештај кај извештајот од примачот и извештајот од испраќачот кај RTCP		160	складирачки, просторен канал базиран на вредност
	временски печат		1	временски, базиран на вредност
	NTP временски печат кај RTCP		1	временски, базиран на вредност
	таг за автентикација кај SRTP		80	складирачки, просторен канал базиран на вредност
(Bai, Huang, Hou, & Xiao, 2008)	*interarrival jitter* кај RTCP		32	складирачки, просторен канал базиран на вредност
(Lizhi, Yongfeng, Jian, & Bai, 2012)	испраќање само на RTP пакет или на RTP и RTCP		1 или 1/2	временски, базиран на транзиција

Bai и соработниците (Bai, Huang, Hou, & Xiao, 2008) го користат 32-битното *interarrival jitter* поле на RTCP заглавјето за создавање на скриени канали. Тие користат две фази: во првата фаза се пресметува вредноста на jitter полето во тековната мрежа. Во втората фаза тајната порака е модулирана во *interarrival jitter* поле во согласност со претходно пресметаните параметри. Lizhi и соработниците (Lizhi, Yongfeng, Jian, & Bai, 2012) предлагаат нов временски скриен канал, кој го користи Run Length Code и Multi-Zero Code за да се подобри незабележливоста и робустноста. Основната идеја е многу едноставна, ако тековниот стегобит е ист како и претходниот се испраќа еден RTP пакет во спротивно се испраќаат RTCP и RTP пакети.

Табелата 22 ги сумира стеганографските техники кои може да се применат кај протоколите RTP, RTCP и SRTP.

4.5. SIP и SDP

Session Initiation Protocol - SIP, дефиниран во RFC 3261 (Rosenberg, и др., 2002) е сигнализирачки протокол за поставување, промена и прекинување на RTP-базирани мултимедијални сесии помеѓу учесниците кои се поврзани со IP-базирана податочна мрежа. Главниот мотив на развивањето на SIP е да се овозможи глас преку IP (Voice over IP - VoIP). SIP може да поддржува каков било тип на мултимедијална сесија (под мултимедијална се подразбира и сесија кога се пренесува и само еден тип на медиум /media/, вклучително и телеконференцијата).

Се базира на HTTP барање/одговор моделот на трансакции и користи концепти слични на итеративните и рекурзивните пребарувања кај DNS. SIP ги користи повеќето полиња за заглавја, правила за кодирање и статусни кодови на HTTP. Обезбедува формат за прикажување на информации во облик на читлив текст. Има пет основни функции:

- локација на корисникот: корисниците можат да се преместат на друга локација и да пристапуваат до телефонијата или некоја друга апликација од оддалечени локации;
- достапност на корисникот: одредување дали постои желба повикуваната страна да учествува во комуникацијата;

- можности на корисникот: одредување на типот на податоци и параметрите што ќе се користат;
- воспоставување на сесија: воспоставување на повик точка-точка и повици со повеќе учесници, со договорените параметри за сесијата;
- управување со сесијата: ги вклучува задачите за пренесување и прекинување на сесиите, промена на параметрите за сесиите и повикување на сервисите.

SIP го користи Session Description Protocol - SDP, дефиниран во RFC 2327 (Handley & Jacobson, SDP: Session Description Protocol, 1998) за опишување на содржината на сесијата, која може да биде наменета за телефонија, Интернет радио или мултимедијална апликација. SDP вклучува информации за: мултимедијални текови, адреси, порти, типови на корисен товар, извор и времиња за почеток и крај.

Скриени канали во SIP и SDP

Mazurczyk и Szczypiorski (Mazurczyk & Szczypiorski, Covert Channels in SIP for VoIP Signalling, 2008) предложиле користење на некои токени и полиња во SIP за криење на податоци, и тоа (табела 22):

- Случајно генериран *маг* во *From* полето, кој го формира идентификаторот на SIP дијалог. Мора да биде најмалку 32 бита долг, нема горна граница;
- Случајно генерираниот *branch* во *Via* полето, кој формира идентификатор на трансакцијата, без ограничување во должината;
- Полето *Call-ID*, кое служи за единствена идентификација на повик, се генерира како комбинација од случаен стринг и името или IP адресата на хостот. Случајниот стринг нема ограничување во должината и може да се искористи за скриен канал;
- Полето *CSeq* се состои од комбинација на иницијален редоследен број, кој служи за идентификување и подредување на трансакциите и други полиња и име на метода. 32-битниот иницијален редоследен број може да се користи како скриен канал;

- 8-битното полето *Max-Forwards* служи за детекција на циклус, при одредени услови;
- Некои од полињата *Contact, Subject, Call-Info, Organization, Reply-To, Timestamp, User-Agent;*
- Користење на низи од знаци кои не се печатат (како празни места [SP] или табови [HT]) во полињата од SIP заглавјето, слично на (Kwecka, 2006).

Mazurczyk и Szczypiorski (Mazurczyk & Szczypiorski, Covert Channels in SIP for VoIP Signalling, 2008) предложиле криење на податоците во SDP во некои полиња кои или не се битни или се игнорираат, како:

- *v* - поле за верзијата кое е игнорирано од SIP,
- *o* - сопственик / создавач,
- *s* - поле за име на сесијата, игнорирано од SIP,
- *t* - поле за време на активност на сесијата, игнорирано од SIP,
- *κ* - потенцијален клуч за криптирање, ако се користи безбедна комуникација.

Друг скриен канал го експлоатира фактот дека редоследот на полињата во заглавјата во SIP/SDP пораките зависи од имплементацијата, па така со сортирање на заглавјата може тајно да се испраќаат податоци. На пример, ако полето Call-ID е по CSeq ќе биде бинарна 1, инаку бинарна 0. Бидејќи имињата на полињата не се осетливи на мали и големи букви, скриен канал може да се формира на тој начин што големи букви би се користеле за бинарна 1, а мали за бинарна 0 или обратно. Сепак, ваквиот скриен канал лесно може да се открие.

Авторите на (Mazurczyk & Szczypiorski, Covert Channels in SIP for VoIP Signalling, 2008) предложиле и креирање на скриени канали во SIP и SDP со користење на безбедносните механизми за автентикација и доверливост. SDP содржината вгнездена во SIP INVITE порака може да биде шифрирана и потпишана со S/MIME, а должината и вредноста на овој товар може да се бираат случајно. Шифрираниот дел содржи application/pkcs7-mime бинарна *envelopedData* структура која го енкапсулира шифрираниот SDP опис на сесија и може целосно да се користи за скриен канал. Вториот дел е потписот на товарот

application/pkcs7-signature, па може да се користат неговите битови како втор скриен канал (верификацијата ќе биде неуспешна во таков случај). PRBR во овој случај зависи од користената хаш функција, на пример, ќе биде 256 бита ако се користи SHA-256.

Табела 23. Стеганографски техники за SIP, SDP
Table 23. Steganographic techniques for SIP, SDP

Статија	Зафатени полиња/ техника	RBR (bps)	PRBR (bits/packet)	Тип
(Mazurczyk & Szczypiorski, Covert Channels in SIP for VoIP Signalling, 2008)	*маг* во *From* полето		нема ограничување	складирачки, просторен канал базиран на вредност
	branch во *Via* полето		нема ограничување	складирачки, просторен канал базиран на вредност
	Call-ID		нема ограничување	складирачки, просторен канал базиран на вредност
	првиот дел од *CSeq* полето		32	складирачки, просторен канал базиран на вредност
	Max-Forwards		8	складирачки, просторен канал базиран на вредност
	Contact, Subject, Call-Info, Organization, Reply-To, Timestamp, User-Agent		нема ограничување	складирачки, просторен канал базиран на вредност
	v во SDP		нема ограничување	складирачки, просторен канал базиран на вредност
	o во SDP		нема ограничување	складирачки, просторен канал базиран на вредност
	s во SDP		нема ограничување	складирачки, просторен канал

				базиран на вредност
	t во SDP		нема ограничување	складирачки, просторен канал базиран на вредност
	k во SDP		нема ограничување	складирачки, просторен канал базиран на вредност
	сортирање полиња во заглавјата во SIP/SDP		$log_2 n!$, каде *n* е број на полиња кои се сортираат	временски, базиран на транзиција
	користење име на поле со мали или големи букви на полиња во заглавјата во SIP/SDP		*n,* каде *n* е број на полиња во заглавјето	складирачки, просторен канал базиран на вредност
	envelopedData структура		нема ограничување	складирачки, просторен канал базиран на вредност
	application/pkcs7-signature потписот		256, ако се користи SHA-256	складирачки, просторен канал базиран на вредност

4.6. SSH

Secure Shell - SSH е криптографски клиент-сервер протокол за безбедна размена на податоци, далечинско најавување преку командна линија, извршување преку далечински команди, како и други безбедни мрежни услуги помеѓу два компјутера поврзани на мрежа преку безбеден канал на небезбедна мрежа, сервер и клиент (кои работат со SSH сервер и SSH клиентски програми, соодветно). Спецификацијата на протоколот разликува две главни верзии, кои се наведени како SSH-1 и SSH-2 (RFC 4253). Обезбедува автентикација на сервер, доверливост и интегритет. Освен за безбедно далечинско најавување, се користи и за безбеден SSH трансфер на датотеки (SFTP) и безбедно копирање (SCP). Структурата на SSH пакетот е прикажана на табела 24, а неговата максимална големина е 35000B.

Табела 24. SSH заглавје
Table 24. SSH header

Packet Length	Padding Length	Packet Data	Random Padding	MAC

Полето *должина на пакет (Packet length,* 32 бита*)* ја претставува должината на пакетот во бајти, не вклучувајќи ги полињата за должина на пакет и пораката за проверка на код (MAC).

Полето *должина на дополнување (Padding length,* 8 бита*)* е должината на полето за случајно дополнување во бајти.

Полето за *податоци во пакетот (Packet Data)* ја претставува корисната содржина на пакетот. Ако е договорена компресија, содржината на ова поле е компресирана.

Полето за *случајно дополнување (Random padding)* може да има големина до 255B треба да се наполни со случајни вредности.

Полето *MAC (Message Authentication Code)*, доколку е договорена проверката на пораката, ја содржи MAC вредноста. Вредноста на MAC е пресметана во целиот пакет плус редниот број, со исклучок на MAC полето. Како алгоритми се користат hmac-sha1, hmac-sha1-96, hmac-md5, hmac-md5-96 и none.

SSH е дизајниран како замена за Telnet и други небезбедни далечински протоколи, како што rsh и rexec протоколите на Berkeley, кои испраќаат информации, особено лозинки како чист текст, користат рендерирање за да бидат подложни на следење и откривање користејќи анализа на пакетот. Шифрирањето кое се користи од страна на SSH е наменето да обезбеди доверливост и интегритет на податоците преку небезбедна мрежа. Стандардната TCP порта 22 е доделена за контактирање на SSH сервери.

SSH се состои од 3 компоненти (RFC 4251):

- Transport Layer Protocol – обезбедува автентикација на сервер, доверливост и интегритет, а може и компресија;
- User Authentication Protocol – ја автентицира клиентската страна на серверот;

- Connection Protocol – го мултиплексира шифрираниот тунел на неколку логички канали.

Скриени канали во SSH

Постојат неколку шеми за криење на информации во SSH пакети, дадени во (Lucena, Pease, Yadollahpour, & Chapin, 2005). Скриена порака може да се смести во полето *MAC*, бидејќи менувањето на овие полиња не го нарушува SSH протоколот. Сепак, замената на MAC бара менување на SSH клиентот и серверскиот софтвер за да се игнорираат MAC пресметките. За да се симулира случајноста на MAC вредноста, тајната порака најпрво се компресира и шифрира. Бидејќи големината на MAC полето е до 160 бита, ваквиот канал овозможува пренос на максимум 160 бита на SSH пакет. Недостаток на оваа метода е тоа што нема верификација дали корисниот товар е пренесен точно или не. За да се надмине овој недостаток, може да се избере MAC алгоритам со помала дожина, на пример hmac-md5-96 и остатокот да се дополни со битови од тајната порака. За дадениот алгоритам ќе останат 64 бита за тајната порака.

Друг начин на креирање на скриен канал во SSH е додавање на шифрирана порака на почетокот од пакетот и тоа според авторите, најдобро е да се додаваат по 12B на пакет, за да се зачува големината на Telnet-сличните SSH сесии, за кои забележале дека имаат големина од 48B до 80B. Тие додаваат и дополнителни 4B на CRC код за проверка на интегритетот на тајната порака. Овие два начина авторите ги именуваат како стеганографски техники кои ја задржуваат семантиката на протоколот од апликациско ниво.

Скриена порака може да се смести во полињата *случајно дополнување* (Perkins, 2005).

Табела 25. Стеганографски техники за SSH
Table 25. Steganographic techniques for SSH

Статија	Зафатени полиња/ техника	RBR (bps)	PRBR (bits/packet)	Тип
(Lucena, Pease, Yadollahpour, & Chapin, 2005)	целосна замена на *MAC*		160	складирачки, просторен канал базиран на вредност
	дополнување на *MAC*		160-|хеш резултат|	складирачки, просторен канал базиран на вредност
	додавање на шифрирана порака на почетокот од пакетот		12*8 (имплементација)	складирачки, просторен канал базиран на вредност
(Perkins, 2005)	*случајно дополнување*		$\leq$ 255*8	складирачки, просторен канал базиран на вредност

5. Заштита

Пред да се преземе каква било акција против скриените канали, тие прво треба да се идентификуваат или детектираат. Идентификацијата на скриените канали може да биде ад-хок или базирана на формален метод, како на пример, анализа на информациски тек (Denning, 1976), ориентиран граф на информациски тек (Wu, Ding, Yongji, & Han, 2011), Shared Resource Matrix (SRM) метода (Kemmerer, Shared Resource Matrix Methodology: an Approach to Identifying Storage and Timing Channels,", 1983), Covert Flow Tree (CFT) метода (Kemmerer & Porras, Covert Flow Trees: A Visual Approach to Analyzing Covert Storage Channels, 1991) и други. Идентификацијата се обидува да открие делен ресурс кој потенцијално може да биде искористен за креирање на скриен канал. За разлика од идентификацијата, детекцијата се обидува да открие постоечки скриен канал со испитување на неговата излезна секвенца од настани (Cabuk, S., Brodley, & Shields, 2009).

Откако ќе се идентификува или детектира скриениот канал се достапни следниве контрамерки (Jeng & Abrams, 1987):

- елиминирање на каналот;
- ограничување на пропусниот опсег на каналот;
- следење на каналот;
- документирање на каналот.

Според Jeng & Abrams (1987) постојат две причини за постоењето на скриените канали: пропуст во проектирањето и слабости кои се својствени за дизајнот на системот. Додека скриените канали кои се предизвикани од пропуст може да се коригираат откако ќе бидат откриени, системските не можат да се отстранат без претходно да биде направено редизајнирање на системот. Затоа, идеално скриените канали треба да бидат идентификувани и отстранети за време на фазата на проектирање. Ако скриените канали не се отстранети во фазата на проектирање, следната најдобра опција е да се елиминира можната употреба, бидејќи дури и каналите кои имаат многу мал капацитет може успешно да бидат експлоатирани. Сепак, многу научници сметаат дека скриените канали не може целосно да се елиминираат (Jeng & Abrams, 1987), (Moskowitz & Kang, 1994),

(Zander, Armitage, & Branch, 2007). Доколку скриениот канал не може да се елиминира, потребно е да се намали неговиот капацитет. Вообичаено ова намалување води кон намалување на перформансите на системот, бидејќи најчесто се забавуваат системските механизми или се воведува шум. Исто така, секој скриен канал кој не може да се отстрани треба да се следи или барем да се документира. Детекцијата на скриениот канал е битна за да се обесхрабрат можните корисници и механизмите за детекција може да се извршуваат пред шемите за елиминација, за истите да може да се активираат само во случај на сомнителна активност. На овој начин може да се редуцира негативниот импакт на шемите за елиминација врз перформансите на даден систем.

Се покажало дека необичните облици на сообраќај може да доведат до откривање на скриени канали кои користат заглавја во мрежните протоколи. На пример, повеќекратни ping барања во краток временски интервал може да индицираат присуство на ICMP тунел во корисен товар, како Loki (daemon9, AKA, & route, Project Loki, 1996). Аномалии во некористените полиња од заглавјето може да индицираат скриен канал (Handley & Paxson, Network intrusion detection: Evasion, traffic normalization, and end-to-end protocol semantics, 2001). Постојат и офлајн шеми за детекција базирани на векторските машини за поддршка (support vector machines - SVM) за детекција на ICMP тунели во корисен товар (Sohn, Moon, Lee, Lee, & Lim, 2003) и скриени канали кои ги користат заглавјата на TCP/IP протоколите (Sohn, Seo, & Moon, A study on the covert channel detection of TCP/IP header using support vector machine, 2003).

Обезбедувањето на хостот не може да ги отстрани скриените канали во мрежата, но во некои случаи може да ја спречи нивната експлоатација. Ако хостовите се обезбедени од напад ќе биде невозможна инсталацијата на злонамерен софтвер, модификацијата на софтверот или на мрежниот склад, со што хакерите нема да бидат во можност да ги користат скриените канали. Сепак, довербата во безбедноста на хостот може да биде опасна и најдобро решение е елиминирање на скриените канали кога тоа е можно.

Еден начин за спротивставување на скриените канали е блокирањето на протоколите или портите кои се подложни за имплементирање на скриени канали.

На пример, ICMP сообраќајот е блокиран од страна на многу заштитни sидови, па така пристапот со рендерирање, како што е Loki, не е ефикасен. Некои протоколи како IP, TCP, DNS, не можат да бидат блокирани, бидејќи тие се од витално значење или поради тоа што нивните услуги се премногу важни (на пример, електронска пошта, веб). Истекувањето на класифицирани информации од високо безбедносен систем на систем со ниско безбедносно ниво (класичен скриен канал) може да се спречи при проектирањето на мрежата, каде што е дозволено да се комуницира само помеѓу хостовите на исто безбедносно ниво. Скриените канали со одбивање како Skeeve (Zelenchuk, 2004) работат само доколку е можно спуфирање на IP адресата. Спречувањето на IP спуфирањето ги елиминира таквите канали.

Многу од каналите можат да се отстранат со нормализирање на заглавјата, дополнувањата и екстензиите како што е опишано во (Malan, Watson, Jahanian, & Howell, 2000), (Handley & Paxson, Network intrusion detection: Evasion, traffic normalization, and end-to-end protocol semantics, 2001) и (Fisk, Fisk, Papadopoulos, & Neil, 2002) или посебно за IPv6 протоколот од страна (Lucena, Lewandowsk, & Chapin, 2005) и ICMP тунелирање од страна на (Singh, Lu, Nordstrom, & dos Santos, 2003). Нормализацијата на сообраќајот може да се изврши од страна на крајниот хост или од мрежните уреди како заштитниот sид или прокси. Неискористените или резервираните полиња и дополнувања можат лесно да се решат со нивно поставување на нула, додека непознатите екстензии во заглавјето можат да се отстранат. Некои скриени канали го користат фактот дека одредени полиња во заглавјето не се користат секогаш (и нивната употреба е наведена од страна на други полиња од заглавјето). Овој факт, исто така, може да се користи за нормализација. Голем број други полиња во заглавјето можат да подлегнат на модифицирање под одредени претпоставки. На пример, да се овозможи *DF* битот, полињата *идентификација* и *поместеност на фрагмент* кај IP да се постават на нула ако пакетот е под MTU големина (под претпоставка дека нормализаторот го знае MTU), модифицирање на полето *идентификација* кај IP (под претпоставка дека нормализаторот може да манипулира со сите фрагменти), модифицирање на *редоследен број* кај TCP, *изворна (IP) адреса* и *изворна порта* (под претпоставка

дека нормализаторот може да ги задржи мапирањата помеѓу оригиналот и новите вредности). Полињата *TTL* кај IP и ***временски печат*** во TCP, исто така, може да се модифицираат (под претпоставка дека нормализаторот се наоѓа во или многу блиску до изворот). Истите концепти може да се користат за елиминирање на скриените канали во протоколите на апликациско ниво. (Schear, Kintana, Zhang, & Vahdat, 2006) предложиле елиминирање на скриените канали во HTTP одговорот со ограничување на употребливите заглавја за одговор на фиксна вредност и со проверка на полињата на заглавјата за одговор со соодветните метаподатоци и со барањето на клиентот.

Ако бројот на различни употребливи адреси е мал, капацитетот на скриенитот канал е многу мал. Ограничувањето на бројот на адреси значи ограничување на дозволените хост-до-хост конекции. Ова може да се реализира во затворени мрежи, но не и во отворени мрежи, како што е Интернет. Адресата на испраќачот секогаш треба да биде фиксна (спречување на IP измама), но бројот на дестинациски адреси не може да биде ограничен на мал број. Слично на тоа, употребливите изворни и дестинациски порти тешко може да бидат ограничени.

Друг начин на заштита е испраќање на dummy пакети помеѓу случајни хостови и со тоа се вметнува шум обликот на сообраќајот. Истиот ефект, но со помалку *overhead*, може да го има индиректното упатување (Newman-Wolfe & Venkatraman, 1991). (Girling, 1987) предлага одреден број на можни големини на пакетот, па така бројот на достапни големини на пакетот треба да биде доволно мал за да се ограничи капацитетот на скриениот канал. Сепак, способноста на испраќачот за моделирање на големината на пакетот може да биде ограничена во тековните IP мрежи (големината на UDP пакетот може да се приспособува во рамките на границите на MTU).

Предложени се повеќе решенија за ограничување на капацитетот на каналите: воведување на случаен шум за да се маскираат скриените канали или отворениот канал е принуден да користи фиксни податочни брзини за испраќање на пакет или порака, во комбинација со dummy пакети, или кога не се испраќаат корисни информации се вметнуваат пораки. Временските канали кои користат

секвенцирање на пораките може да се елиминираат со баферирање или одложување на обидите за воспоставување на врска, барањата за услуги и др. Може да се вметнуваат и лажни податоци во мрежата кои ќе служат против прислушувањето, но ова не помага против крајните-хост приемници.

Се смета дека во догледно време мерките против повеќето временски канали нема да може да се имплементираат, па скриените временски канали и каналите кои користат различни податочни брзини ќе останат (Zander, Armitage, & Branch, 2007).

Сите скриени канали кои се заосновани на нестандардна употреба на протоколот може лесно да бидат откриени. Покрај тоа, некои од предложените скриени канали се доста очигледни, бидејќи се користат претходно неискористени полиња, дефинирани пораки или продолжени заглавја кои не се користат повеќе и нивното присуство ќе биде сомнително (на пример, ICMP контрола на проток и продолженото IP заглавје на временскиот печат). Некои скриени канали опишани претходно го користат фактот дека полињата на заглавјето може да имаат произволни вредности во барањата на стандардот. Вообичаен пристап е или да се обучи класификатор за нормално и абнормално однесување, или да се обучи класификатор на нормално однесување и откривање на аномалии. Однесувањето се анализира од збир на протокот на сообраќај, каде што секој проток е опишан со голем број на карактеристики (особини).

(Hintz, 2003) предложил метод за детекција на канали кои користат TCP временски печати со примена на тест за случајност на најмалку значајниот бит кај временските печати и тоа кај мрежи со мали брзини. Премногу случајност може да го открие скриениот канал. Кај мрежите со големи брзини скриениот канал може да биде откриен со односот на различните временски печати и вкупниот број на можни временски печати (во зависност од траењето на врската) и за скриен канал овој однос би бил околу 0,5 (ако LSB не е ист со скриениот бит, се рипа еден такт), а за нормална комуникација околу 1 (се испраќа еден сегмент во еден такт).

Меѓутоа, додека истражувачите имаат развиено голем број контрамерки, искуството во реалноста покажува дека сè уште недостигаат методи за справување со скриените канали. Исто така, некои од предложените контрамерки

може значително да ги намалат перформансите на отворениот канал и затоа нивната примена во реалните мрежи со голема брзина е под знак прашалник.

6. Заклучок

Во овој магистерски труд е направена анализа на постоечките методи и техники за креирање и имплементација на скриени канали во TCP/IP складот со протоколи. Анализата е класифицирана според засегнатите протоколи и соодветните нивоа. Исто така е направен преглед на особините на стеганографските системи, разликите помеѓу стеганографија, водени печати и криење на информации (податоци), како и мерките со кои може да се анализираат различните мрежни стеганографски техники. Опфатени се најзастапените протоколи од интернет нивото, како што се: IPv4, IPv6, ICMP, IGMP, DHCP и ARP, за кои е даден краток опис на протоколот, опис на неговите заглавја и соодветните стеганографски техники. Истото е направено и за протоколите TCP, UDP и SSL/TLS од транспортното ниво, како и за протоколите од апликациско ниво и тоа: HTTP, FTP, DNS, RTP, RTCP, SRTP, SIP и SSH. Предложени се механизми и техники кои се користат за заштита од скриените канали во мрежните протоколи.

На крајот се направени следниве заклучоци:

- Генерално, временските скриени канали имаат помал пропустлив опсег од складирачките временски канали;
- Има многу повеќе складирачки скриени канали, отколку временски скриени канали;
- Од сите протоколи, DNS како протокол кој е најмалку филтриран е најинтересен за креирање на тунел во корисен товар за протоколите IP, TCP, SSH и други. ICMP е, исто така, популарен за оваа намена, но во последно време ваквите канали се спречуваат со целосно или делумно филтрирање на ICMP сообраќајот;
- Кај IPv4 се присутни сите типови на скриени канали според (Wang et al., 2005): има 13 просторни канали базирани на вредност, 1 просторен канал базиран на транзиција, 1 временски канал базиран на вредност и 10 временски канали базирани на транзиција. Со складирачкиот скриен канал на (Trabelsi et al., 2006) се пренесуваат најмногу 40B на пакет;

- Од скриените канали специјално дефинирани за IPv6, 14 се просторни канали базирани на вредност, а само еден е хибриден, односно комбинација на просторен канал базиран на вредност и временски канал базиран на транзиција;
- И за ICMP има само просторни канали базирани на вредност, од кои се разгледани само 6, а 4 од нив се тунел во корисен товар, со можност за пренесување над до 24B или до 56B или повеќе (зависи од имплементацијата) на пакет;
- За IGMP и ARP има само по 1 просторен канал базиран на вредност со можност за пренос на 24 и 8 битови на пакет, соодветно;
- Кај DHCP има 5 просторни канали базирани на вредност и 1 временски канал базиран на вредност, со можност за испраќање и на 255B по пакет;
- За TCP анализирани се 7 просторни канали базирани на вредност со максимум пренесени 64 битови скриена порака на сегмент, 2 временски канали базирани на транзиција и 1 временски канал базиран на вредност;
- Кај UDP има само 3 просторни канали базирани на вредност со максимум пренесени 48 битови скриена порака на сегмент;
- За HTTP се анализирани 9 просторни канали базирани на вредност и 1 временски канал базиран на транзиција, со можности за пренос и на 8KB по порака;
- Кај DNS разгледувани се 4 просторни канали базирани на вредност кои истовремено се и тунели во корисен товар и 1 временски канал базиран на вредност;
- Кај FTP има 1 просторен канал базиран на транзиција и 1 временски канал базиран на транзиција, кој е и канал со бројач;
- За RTP има 6 просторни канали базирани на вредност, 2 временски канали базирани на вредност и 1 временски канал базиран на транзиција, со можност за испраќање до 255B по пакет;

- Кај SIP и SDP има 14 просторни канали базирани на вредност и 1 временски канал базиран на транзиција;
- SSH има 4 просторни канали базирани на вредност, со можност за испраќање до 255B по пакет;
- Складирачките скриените канали со редундантни полиња (некористени, резервирани, со случајна вредност) лесно може да се елиминираат;
- Временските канали тешко се детектираат, бидејќи тешко дека некои од мерките за нивна детекција ќе бидат имплементирани во догледно време.

Скриените канали кои емитуваат само неколку бајти може тешко да се детектираат, додека постојат и техники за креирање на сосема невидливи скриени канали. Подобрувањето на безбедноста на системот и нормализирањето на сообраќајот го намалуваат ризикот од скриени канали. Повеќето скриени канали се предвидливи, бидејќи ги користат неискористените полиња во заглавјата на протоколите и можат да се елиминираат. Други канали користат случајни податоци (на пример, TCP ISN). Иако ваквите канали се тешки за детектирање, ако се правилно кодирани, тие сè уште може лесно да се елиминираат. Појавата на шум во комуникациските канали го намалува капацитетот на каналот, но скриените канали имаат корист бидејќи ја зголемува варијабилноста за криење на податоци.

Може да се потврди дека можностите за воспоставување на тајни комуникации преку мрежата на скриени канали се скоро бескрајни. Ова е добра причина за да се препорача дека идните истражувачи треба да ги насочат своите напори во развојот на генеричко откривање и механизми за превенција и спротивставување од заканата на овие канали.

Литература

Abad, C. (2001). *IP checksum covert channels and selected hash collision.* Повратено од http://gray-world.net/papers/ipccc.pdf.

Ahsan, K. (2002). *Covert channel analysis and data hiding in TCP/IP.* Master thesis, University of Toronto.

Ahsan, K., & Kundur, D. (2002). Practical data hiding in TCP/IP. *ACM Workshop on Multimedia Security.*

Allix, P. (2007). *Covert channels analysis in TCP/IP networks.* Orsay, France.

Alman, D. (2003). HTTP Tunnels Though Proxies. *SANS Institute 2003.*

Anjan, K., & Abraham, J. (2010). Behavioral Analysis of Transport Layer Based. Во *Recent Trends in Network Security and Applications* (стр. 83-92). Springer-Verlag Berlin Heidelberg.

Anjan, K., Abraham, J., & Jadhav, M. (2010). Design of Transport Layer Based Hybrid Covert Channel Detection Engine. *1*(4).

Anonymous. (2005). *DNS Covert Channels and Bouncing Techniques.*

Bai, L. Y., Huang, Y., Hou, G., & Xiao, B. (2008). Covert Channels Based on Jitter Field of the RTCP Header. *Fourth International Conference on Intelligent Information Hiding and Multimedia Signal Processing.*

Bauer, M. (2003). New Covert Channels in HTTP: Adding Unwitting Web Browsers to Anonymity Sets. Erlangen.

Baugher, M., McGrew, D., Naslund, D., Carrara, E., & Norrman, K. (2004). The Secure Real-time Transport Protocol (SRTP). *IETF, RFC 3711*.

Bellovin, S. (1989). Security problems in the TCP/IP protocol suite. *19*(2), 32-48.

Berk, V., Giani, A., & Cybenko, G. (2005). *Detction of Covert Channel Encoding in Network Packet Delays.* Department of Computer Science, Dartmouth College.

Bowyer, L. (2002). *Firewall Bypass via protocol Steganography.*

Cabuk, S., Brodley, C., & Shields, C. (2004). IP covert timing channels: Design and detection. *11th ACM Conference on Computer and communication Security* (стр. 178-187). Washington DC, USA: ACM Press.

Cabuk, S., Brodley, C., & Shields, C. (2009). IP Covert Channel Detection. *12*(4).

Cain, B., Deering, S., Kouvelas, I., & Thyagarajan, B. F. (2002). *Internet Group Management Protocol.* The Internet Society.

Castro, S. (2006). *Cooking Channels.* hakin9 Magazine.

Cauich, E., Gomez, R., & Watanabe, R. (2005). Data Hiding in Identification and Offset IP Fields. *Springer, 3563*(5), 118-125.

Chakinala, R., Kumarasubramanian, A., Manokaran, R., Noubir, G., C., P. R., & Sundaram, R. (2006). Steganographic communication in ordered channels. *Information Hiding Workshop.*

Conta, A., Deering, S., & M. Gupta. (2006). *Internet Control Message Protocol (ICMPv6).* The Internet Society.

Cox, I., Miller, M., Bloom, J., Fridrich, J., & Kalker, T. (н.д.). *Digital Watermarking and Steganography.* The Morgan Kaufmann Series in Multimedia Information and Systems, Elsevier Inc.

daemon9. (1997). Loki2 (the implementation). *51*(6). Повратено од http://www.phrack.com/issues.html?issue=51&id=6

daemon9, AKA, & route. (1996). Project Loki. *49*(6). Повратено од http://www.phrack.org/issues.html?issue=49&id=6&mode=txt

Danezis, G. (2005). *Covert Communications Despite Traffic Data Retention.* ESAT, University of Leuven.

Deering, S., & Hinden, R. (1998). *Internet Protocol, Version 6 (IPv6).* The Internet Society.

deGraaf, R., Aycock, J., & Jacobson, M. (2005). Improved Port Knocking with Strong Authentication. *21st Annual Computer Security Applications Conference*, (стр. 10 pp. - 462).

Dembour, O., & Collignon, C. (2008). DNS2TCP. *http://hsc.fr/ressources/outils/dns2tcp/*.

Denning, D. (1976). A Lattice Model of Secure Information Flow. *19*(5), 236–243.

Department of Defence:. (1985). Department of defence trusted computer system evaluation. Technical Report DOD 5200.28-ST.

Dong, P., Qian, H., Lu, Z., & Lan, S. (2012). A Network Covert Channel Based on Packet Classification. *14*(2), 109-116.

Dyatlov, A., & Castro, S. (2003). *Exploitation of data streams authorized by a network access control system for arbitrary data transfers: tunneling and covert channels over the HTTP protocol.* USA: Gray-world.

Eßer, H. G., & Freiling, F. C. (2005). *Kapazittsmessung eines verdeckten Zeitkanals ber HTTP.*

Eu-Jin, G., Dan, B., Philippe, G., & Benny, P. (2003). The Design and Implementation of Protocol-based Hidden Key Recovery. *6th Information Security Conference.*

Feamster, N., Balazinska, M., Harfst, G., Balakrishnan, H., & Karger, D. (2003). Infranet: Circumventing Web Censorship and Surveillance. *11TH USENIX SECURITY SYMPOSIUM.*

Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., & Berners-Lee, T. (1999). *Hypertext Transfer Protocol -- HTTP/1.1.* Internet Official Protocol Standards.

Fisk, G., Fisk, M., Papadopoulos, C., & Neil, J. (2002). Eliminating Steganography in Internet Traffic with Active Wardens. *IH '02 Revised Papers from the 5th International Workshop on Information Hiding.* London.

Forte, D. V. (2005). SecSyslog: An Approach to Secure Logging Based on Covert Channels. *First InternationalWorkshop of Systematic Approaches to Digital Forensic Engineering (SADFE).*

Galatenko, A., Grusho, A., Kniazev, A., & Timonina, E. (2005). Statistical Covert Channels Through PROXY Server. *Third International Workshop on Mathematical Methods, Models, and Architectures for Computer Network Security*, *LNCS 3685*, стр. 424-429. St. Petersburg, Russia.

Gallagher, P.R. (1993). *Guide to Understanding Covert Channel Analysis of Trusted.* USA: National Computer Security Center.

Giffin, J., Greenstadt, R., Litwack, P., & Tibbetts, R. (2003). Covert Messaging Through TCP Timestamps. *2nd international conference on Privacy enhancing technologies.* Berlin: 2nd.

Girling, C. (1987). Covert channels in LAN's. *13*(2), 292-296.

Graf, T. (2003). *Messaging over IPv6 Destination Options.* Swiss Unix User Group.

Gray, J. (1994). Countermeasures and tradeoffs for a class of covert timing channels.

Handel, T., & Sandford, M. (1996). Hiding data in the OSI network model. *Information Hiding. LNCS, vol. 1174* (стр. 23-38). Springer Verlag.

Handley, M., & Jacobson, V. (1998). *SDP: Session Description Protocol.* The Internet Society.

Handley, M., & Paxson, V. (2001). Network intrusion detection: Evasion, traffic normalization, and end-to-end protocol semantics. *10th USENIX Security Symposium.*

Handly, M., & Paxson, V. (2001). Network intrusion detection: Evasion, traffic normalization, and end-to-end protocol semantics. *10th USENIX Security Symposium.*

Hintz, A. (2003). *Covert Channels in TCP and IP Headers.*

Information Sciences Institute. (1981). *DARPA INTERNET PROGRAM PROTOCOL SPECIFICATION.* California.

Ingemar J. Cox, M. L. (2008). *Digital Watermarking and Steganography.* Morgan Kaufmann Publishers.

Jankowski, B., Mazurczyk, W., & Szczypiorski, K. (2010). Information Hiding Using Improper Frame Padding. Warsaw.

Jankowski, B., Mazurczyk, W., & Szczypiorski, K. (2013). *PadSteg: Introducing Inter-Protocol Steganography.* Warsaw.

Jeng, A., & Abrams, M. (1987). On Network Covert Channel. *3rd Aerospace Computer Security Conf.*

Ji, L., Fan, Y., & Ma, C. (2010). Covert channel for local area network. *Wireless Communications, Networking and Information Security (WCNIS), 2010 IEEE International Conference*, (стр. 316 - 319). Beijing.

Jones, E., Le Moigne,, O., & Robert, J.-M. (2004). IP Traceback Solutions Based on Time to Live Covert Channel. *12th IEEE Int'l. Conf. Networks*, (стр. 451-457).

Kaminsky, D. (2004). Black Ops of DNS. *The Black Hat Briefings 2004.*

Kaminsky, D. (2004). OzymanDNS. *http://dankaminsky.com/2004/07/29/51/*.

Kaminsky, D. (2008). *Attacking Distributed Systems - The DNS Case Study.*

Kawamoto, D. (2006). *Blackmailers try to black out Million Dollar Homepage.* CNET News.com.

Kemmerer, R. (1983). Shared Resource Matrix Methodology: an Approach to Identifying Storage and Timing Channels,". *1*(3), 256–277.

Kemmerer, R., & Porras, P. (1991). Covert Flow Trees: A Visual Approach to Analyzing Covert Storage Channels. *SE-17*(11), 1166–1185.

Kesdogan, D., Egner, J., & Roland, B. (1998). *Stop-and-Go-MIXes providing probabilistic anonymity in an open system.* 2nd Int. Information Hiding Workshop.

Kryo. (2010). iodine.

Krzywinski, M. (2004, Nov 29). *Portknocking.* Преземено на 24.5.2013 г. од http://www.portknocking.org

Kwecka, Z. (2006). *Application Layer Covert Channel Analysis and Detection.* Edinburgh.

Lampson, B. (1973). A Note on the Confinement Problem. *16* (10), 613-615.

LeBoutillier, P. (2005). *HTTunnel.* Преземено на 4.12.2013 г. од http://sourceforge.net/projects/httunnel/

Lizhi, Y., Yongfeng, H., Jian, Y., & Bai, L. Y. (2012). A Novel Covert Timing Channel Based on RTP/RTCP. *21*(4).

Llamas, D., Allison, C., & Miller, A. (2005). Covert Channels in Internet Protocols: A Survey. *Proceedings of the 6th Annual Postgraduate Symposium about the Convergence of Telecommunications, Networking and Broadcasting (PGNET).* Liverpool.

Lucena, N. (2004). *Syntax and Semantics-Preserving Application-Layer Protocol Steganography.* 6th Information Hidding Wksp.

Lucena, N., Lewandowsk, G., & Chapin, S. (2005). Covert Channels in IPv6. *5th International Workshop Privacy Enhancing Technologies.*

Lucena, N., Pease, J., Yadollahpour, P., & Chapin, S. J. (2005). Syntax and Semantics-Preserving Application-Layer Protocol Steganography. *6th Information Hidding Workshop*, *LNCS 3200*, стр. 164-179. Toronto, Canada.

Lundström, M. (н.д.). MailTunnel.

Luo, X., Chan, E., & Chang, R. (2007). Cloak: A ten-fold way for reliable covert communications. *ESORICS.*

Luo, X., Chan, E., & Chang, R. (2008). TCP Covert Timing Channels: Design and Detection. *IEEE International Conference on Dependable Systems and Networks With FTCS and DCC*, (стр. 420 - 429).

Luo, X., Chan, E., & Chang, R. (2009). CLACK: A network covert channel based on partial acknowledgment encoding. *IEEE ICC.*

Luo, X., Zhou, P., Chan, E., Chang, R., & Lee, W. (2011). A Combinatorial Approach to Network Covert Communications with Applications in Web Leaks. *IEEE/IFIP 41st International Conference on Dependable Systems & Networks (DSN)*, (стр. 474 - 485).

Malan, G., Watson, D., Jahanian, F., & Howell, P. (2000). Transport and Application Protocol Scrubbing. *IEEE Conf. Computer Communications* (стр. 1381–1390).

Mandal, D. (2011). *Covert Channel over ICMP.* Kolkata.

Mazurczyk, W. (2012). Lost Audio Packets Steganography: The First Practical Evaluation. *Journal of Security and Communication Networks.*

Mazurczyk, W., & Kotulski, Z. (2006). *SafeComp. LNCS 4166*, стр. 170-181. Springer.

Mazurczyk, W., & Kotulski, Z. (2006). New Security and Control Protocol for VoIP Based on Steganography and Digital Watermarking. *5*, 417-426.

Mazurczyk, W., & Lubacz, J. (2010). LACK - a VoIP Steganographic Method. *45*(2-3).

Mazurczyk, W., & Lubacz, J. (2010). LACK - a VoIP steganographic method. *45*(2-3), 153-163.

Mazurczyk, W., & Szczypiorski, K. (2008). Covert Channels in SIP for VoIP Signalling. *Communications in Computer and Information Science.*

Mazurczyk, W., & Szczypiorski, K. (2008). Steganography of VoIP Streams. *On the Move to Meaningful Internet Systems: OTM. LNCS 5332.* Springer-Verlag.

Mazurczyk, W., & Szczypiorski, K. (2009). Steganography in handling oversized IP packets. *First International Workshop on Network Steganography (IWNS 2009).* Wuhan, China.

Mazurczyk, W., & Szczypiorski, K. (2012). Evaluation of steganographic methods for oversized IP packets. *Telecommunication Systems, 49.*

Mazurczyk, W., Karas, M., & Szczypiorski, K. (2013). SkyDe: a Skype-based Steganographic Method. *8*(3), 389-400.

Mazurczyk, W., Smolarczyk, M., & Szczypiorski, K. (2010). Retransmission Steganography Applied. Computing Research Repository (CoRR), abs/1007.0767.

Mazurczyk, W., Smolarczyk, M., & Szczypiorski, K. (2010). RSTEG: Retransmission Steganography and Its Detection.

Merlo, A., Papaleo, G., Veneziano, S., & Aiello, M. (2011). A Comparative Performance Evaluation of DNS Tunneling Tools. *4th International Workshop on Computational Intelligence in Security for Information Systems. LNCS 6694*, стр. 84-91. Springer , Heidelberg.

Moskowitz, I. S., & Chang, L. W. (1999). An entropy-based framework for database inference. *3rd Int. Information Hiding Workshop.* 3rd Int. Information Hiding Workshop.

Moskowitz, I., & Kang, M. (1994). Covert Channels — Here To Stay? *9th Annual Conf. Computer Assurance*, (стр. 235-244).

Muench, M. (2003). ICMP-Chat.

Murdoch, S. J., & Lewis, S. (2005). Embedding Covert Channels into TCP/IP. *7th Information Hiding Workshop,.* UK.

Murphy, R. (2006). V00d00n3t – Ipv6 / ICMPv6 Covert Channel. *Slides from DEFCON*.

Murphy, R. (2008). *IPv6 / ICMPv6 Covert Channels.*

Newman-Wolfe, R., & Venkatraman, B. (1991). High Level Prevention of Traffic Analysis. *7th Annual Computer Security Applications Conference*, (стр. 102–109).

Nussbaum, L., Neyron, P., & Richard, O. (2009). On Robust Covert Channels Inside DNS. *Proceedings of IFIP Advances in Information and Communication Technology*, 51–62.

Padgett, P. (2001). *Corkscrew.*

Padlipsky, M. A., Snow, W., & Karger, P. A. (1978). Limitations of End-to-End Encryption in Secure Computer Networks. Massachusetts.

Panajotov, B., & Mileva, A. (2013). Covert Channels in TCP/IP Protocol Stack. *ICT Innovations Web Proceedings*, (стр. 190-199).

Perkins, M. C. (2005). *Hiding out in plaintext : covert messaging with bitwsie summations.* Ames, Iowa.

Phrack Magazine. (1997). Loki 2. *Phrack Magazine, 7*, 6 of 17.

Pietraszek, T. (2004). DNSCat. *http://tadek.pietraszek.org/projects/DNScat/*.

Plummer, D. C. (1982). *An Ethernet Address Resolution Protocol.* INTERNET STANDARD.

Postel, J. (1980). *User Datagram Protocol.* RFC 768.

Postel, J. (1981). *INTERNET CONTROL MESSAGE PROTOCOL.* DARPA INTERNET PROGRAM PROTOCOL SPECIFICATION.

Qu, H., Su, P., & Feng, D. (2004). A Typical Noisy Covert Channel in the IP Protocol. *38th Annual International Carnahan Conference of Security Technology.*

Ray, B., & Mishra, S. (2008). A Protocol for Building Secure and Reliable Covert Channel. *Sixth Annual Conference on Privacy, Security and Trust*, (стр. 246 – 253).

Rios, R., Onieva, J. A., & Lopez, J. (2012). HIDE_DHCP: Covert Communications Through Network Configuration Messages. *HIDE-DHCP: Covert Communications Through Network Configuration Messages* (стр. 162-173). Malaga: Springer Berlin Heidelberg.

Rios, R., Onieva, J. A., & Lopez, J. (2012). HIDE-DHCP: Covert Communications Through Network Configuration Messages. Bo *Information Security and Privacy Research* (стр. 162-173). Malaga: Springer Berlin Heidelberg.

Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Schooler, E. (2002). *SIP: Session Initiation Protocol.* The Internet Society.

Rowland, H. C. (1997). *Covert channels in the TCP/IP protocol suite.* DoIS Documents in Information Science.

Rutkowska, J. (2004). The Implementation of Passive Covert Channels in the Linux Kernel.

savannah.nongnu.org. (2002). NSTX. *http://savannah.nongnu.org/projects/nstx/*.

Sbrusch, R. (2006). *Network Covert Channels:Subversive Secrecy.* SANS Institute.

Schear, N., Kintana, C., Zhang, Q., & Vahdat, A. (2006). Glavlit: Preventing Exfiltration at Wire Speed. *5th Wksp. Hot Topics in Networks (HotNets).*

Schulzrinne, H., Casner, S., Frederick, R., & Jacobson, V. (1996). *RTP: A Transport Protocol for Real-Time Applications.* Internet Official Protocol Standards.

Scott, C. (2008). *Network Covert Channels: Review of Current State and Analysis of Viability of the use of X.509 Certificates for Covert Communications.* London: Egham, Surrey TW20 0EX, England.

Servetto, S., & Vetterli, M. (2001). Communication using phantoms: covert channels in the Internet. *IEEE International Symposium on Information Theory (ISIT).*

Simmons, G. (1983). *The prisoner's problem and the subliminal channel.* Plenum Press.

Singh, A., Lu, C., Nordstrom, O., & dos Santos, A. (2003). Malicious ICMP Tunneling: Defense against the Vulnerability. *8th Australasian Conference on Information Security and Privacy (ACISP).*

Smeets, M., & Koot, M. (2006). *Covert Channels.* University of Amsterdam, Research Report.

Smith, J. C. (2000). *Covert Shells.*

Sohn, T., Moon, J., Lee, S., Lee, D. H., & Lim, J. (2003). Covert channel detection in the ICMP payload using support vector machine. *International Conference on Information and Communications Security*, (стр. 828–835).

Sohn, T., Seo, J.-T., & Moon, J. (2003). A study on the covert channel detection of TCP/IP header using support vector machine. *International Conference on Information and Communications Security*, (стр. 313–324).

Stanley, R. (1997). *Enumerative Combinatorics.* Cambridge: Cambridge University Press.

Stødle, D. (2005). ptunnel - Ping Tunnel.

Stokes, K., Yuan, B., Johnson, D., & Lutz, P. (2009). *ICMP covert channel resiliency.* New York: Rochester Institute of Technology.

Thomer, G. (2005). IP-over-ICMP using ICMPTX.

Thyer, J. S. (2008). Covert Data Storage Channel Using IP Packet Headers. *SANS Institute.*

Trabelsi, Z., & Jawhar, I. (2010). Covert File Transfer Protocol Based on the IP Record Route Option. *5*, 64-73.

Trabelsi, Z., El-Sayed, H., Frikha, L., & Rabie, T. (2006). Traceroute Based IP Channel for Sending Hidden Short Messages. *Advances in Information and Computer Security, First International Workshop on Security, IWSEC 2006.* Kyoto, Japan.

Trabelsi, Z., El-Sayed, H., Frikha, L., & Rabie, T. (2007). A novel covert channel based on the IP header record route option. *1*(4).

Tumoian, E., & Anikeev, M. (2005). Detecting NUSHU Covert Channels Using Neural Networks. Taganrog.

Van Horenbeeck, M. (2006). Deception on the network: thinking differently about covert channels. *Australian Information Warfare and Security Conference.*

vanHauser. (1999). *Placing Backdoors through Firewalls.* The Hacker's Choice.

vecna. (2000). B0CK. *7*(2).

Wang, Z., & Lee, R. (2005). New Constructive Approach to Covert Channel Modeling. Department of Electrical Engineering, Princeton University.

Wolf, M. (1989). Covert channels in LAN protocols. *Workshop on Local Area Network Security (LANSEC89).*

Wu, J., Ding, L., Yongji, W., & Han, W. (2011). A Practical Covert Channel Identification Approach in Source Code Based on Directed Information Flow Graph. *Fifth International Conference on Secure Software Integration and Reliability Improvement (SSIRI).*

Zander, S. (2010). *Performance of Selected Noisy Covert Channels and Their Countermeasures in IP Networks.* Melbourne.

Zander, S., Armitage, G., & Branch, P. (2006). Covert Channels in the IP Time To Live. *Australian Telecommunication Networks and Applications Conference (ATNAC).*

Zander, S., Armitage, G., & Branch, P. (2007). *A survey of covert channels and countermeasures in computer network protocols.* IEEE Communications Surveys and Tutorials.

Zelenchuk, L. (2004). Skeeve — ICMP Bounce Tunnel.

Zielinska, E., Mazurczyk, W., & Szczypiorski, K. (2013). Development Trends in Steganography.

Zou, X., Li, Q., Sun, S., & Niu, X. (2005). The Research on Information Hiding Based on Command Sequence of FTP Protocol. *9th International Conference on Knowledge-Based Intelligent Information and Engineering Systems.*

Борис Панајотов

Криење на податоци во мрежните протоколи

Факултет за информатика, Универзитет „Гоце Делчев“ - Штип