**СКРИТО ПРЕДАВАНЕ НА ДАННИ С ПРОТОКОЛ TCP**

**І. Теоретична част**

Мрежовите стеганографски методи съществено се отличават от компютърните стеганографски методи по използваните контейнери за предаване на тайното съобщение. В мрежовата стеганография за носители на секретните данни не се използват мултимедийни файлове-контейнери, а протоколните единици на мрежовите протоколи от всички нива на модела OSI и протоколния стек TCP/IP.

Голямото разнообразие от протоколи и услуги в различните нива на OSI модела, дава широко поле за разработване на разнообразни техники за скрито предаване на информация. Едно възможно приложение на скрит канал с използване на TCP/IP е манипулиране на дължината на TCP-сегмента. В предаващата страна се образува TCP сегмент, чиято дължина съвпада с ASCII кода на поредния символ от скрития текст. В приемната страна се изчислява дължината на TCP сегмента и по обратен алгоритъм се възстановява текста на скритото съобщение.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 15 | | | | | | 16 | 31 |
| Source port | | | | | | | Destination port | |
| Sequence Number | | | | | | | | |
| Acknowledgement Number (ACK) | | | | | | | | |
| Offser Reserved | U | A | P | R | S | F | Windows | |
| Cheksum | | | | | | | Ugrent Pointer | |
| Options and padding | | | | | | | | |

Фиг. ХХ1 Заглавна част на TCP сегмент

TCP-сегментът е избран за контейнер по две причини. На първо място, защото дължината му може да е променлива и на второ, защото нито в заглавната част на TCP-сегмента (фиг. ХХ1), нито в IP-дейтаграмата (фиг. ХХ2) секретният текст няма да се запише в явен вид. Дължината на TCP-сегментът се изчислява от стойностите на полетата обща дължина Total Length и Header Length на IP-дейтаграмата, като се предполага, че предадените TCP сегменти и IP-дейтаграмите не съдържат полета Options и Padding.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0 |  | 15 | 16 | |  | 31 |
| Version | Header lenght | Type of service | Total length (in bytes) | | | |
| Identification | | | Flags  3bits | Fragment offset | | |
| Time to live (TTL) | | Protocol | Header cheksum | | | |
| Source IP address | | | | | | |
| Destination IP address | | | | | | |
| Options | | | | | | |
|  | | | | | | Padding |

Фиг. ХХ2Заглавна част на IP пакет

Нека да е нужно да се скрие символа «Z», който има десетичен код 90 от ASCII таблицата.

В предаващата страна се образува TCP сегмент, който трябва да пренася потребителските (явни) данни. Дължината на предаваните явни данни трябва да съвпада с кода на скрития символ.

За прехвърляне на отритите данни се формира блок с дължина от 90 байта. Към потребителските данни се добавя TCP хедър с дължина 20 байта (5 думи по 4 байта). Получения TCP сегмент с обща дължина от 110 байта се предава на IP-протокола, който добавя 20-байтова IP-заглавната част и формира IP-дейтаграмата.

Общата дължина на IP-дейтаграмата се записва в полето Обща дължина (в този случай е с дължина 130 байта). Получената IP-дейтаграма се предава на каналния слой (OSI модела) и след това се предава по явния канал.

В приемащата страна за извличане на скрития символ е необходимо да се изчисли дължината на данните, пренасяни от TCP сегмента. За получаване на тази информация, от общата дължина IP-дейтаграмата (поле Total Lenght) се изважда дължината на IP заглавната част: 130-5\*4 = 110 байта. От получената стойност на общата дължина на TCP-сегмента се изважда стойноста на офсетните данни (поле Data Offset)- 110-5\*4 = 90 байта. Получената стойност на дължината на явния текст се използва за код на приетия секретен символ, т.е. на приемната част ще бъде фиксиран символа «Z».

Реалната IP заглавна част на получената от мрежата дейтаграма може да изглежда по следния начин (фиг. ХХ3)



Фиг. ХХ3Заглавна част на получената дейтаграма с двоични данни

Тук полето на общата дължина съдържа двоичното число 0000000010001110, което е еквивалентно на десетичното число 142. Полето за дължина на IP хедъра съдържа двоично то число 0101, т.е. 5 в десетичен вид. Това означава, че дължината на IP-заглавието е пет четирибайтови думи (20 байта). По този начин, дължината на TCP сегмента се равнява на 142 – 20 = 122 байта. На фиг. ХХ4 посочените области са оцветени в сиво.



Фиг. ХХ4Разпечатка на хедъра на TCP- сегмента

Тук в полето Data Offset е записано двоично число 1000, което в десетичната система е равно на 8. Това означава, че размерът на заглавната част на TCP сегмента е осем четирбайтови думи, т.е. 8 ∙ 4 = 32 байта. По този начин, броят на символите от TCP сегмента е 122-32 = 90 байта. Според кодовата таблица ASCII, десетичният код 90 съответства на символа «Z». Тъй като двата съседни символа на тайният текст се предават в различни TCP сегменти и IP-дейтаграми, може да се каже, че скритата информация е с пространственото разпределение на текста в разни контейнери.

**ІІ. Лабораторно упражнение**

*1. Цел на упражнението*

* Преглед на основните понятия в компютърните мрежи и мрежовата стеганография;
* Използване на програмни средства за анализиране на мрежов трафик;
* Разглеждане и поясняване на принципа на скриване на информацията при предаване на данни в TCP/IP мрежи;
* Отговор на контролни въпроси.

*2. Въвеждащи въпроси*

* Кой е основният принцип в стеганографските методи за защита на информацията?
* Кой е основният принцип в мрежовите стеганографски методи за защита на информацията?
* Кое поле от заглавната част на IP дейтаграмата се използва при маршрутизация?
* По какъв начин се скрива информацията с модификация на дължината на TCP сегмент?
* По какъв начин се извлича секретният текст при обмен с модификация на дължината на TCP сегмент?

*3.Задачи за изпълнение в лабораторното упражнение*

3.1. Изучаване на средствата за анализ на мрежови трафик. Работата с мрежовия анализатор Wireshark и използване на филтри за прихващане на трафик, указан в следващата таблицата:

|  |  |
| --- | --- |
| **Номер по ред** | **Задача** |
| 1 | Прихващане на всички TCP-пакети |
| 2 | Прихващане на всички UDP-дейтаграми, с порт равен на 53 |
| 3 | Прихващане на всички TCP-пакети, с порт равен на 80 |
| 4 | Прихващане на всички TCP-пакети, с порт различен от 80 |
| 5 | Прихващане на целия HTTP трафик |
| 6 | Прихващане на целия ICMP трафик |

3.2. Осъществяване на скрит обмен на информация в TCP/IP мрежи.

3.2.1. Използване на програмите labclnt.exe и labsrv.exe с цел установяване на TCP връзка, стартирани на един или два компютъра.

3.2.2. Изпращане и приемане на скрит текст в съотвествие със задачите описани в следващата таблица. Анализиране на мрежовия трафик.

|  |  |
| --- | --- |
| **Номер по ред** | **Задача** |
| 1 | Изпращане на текста „СТЕГО” |
| 2 | Изпращане на текста „СТЕГАНОГРАФИЯ” |
| 3 | Използване на външен файл за явен текст и изпращане на текста „Стеганографията е наука и изкуство за скриване на информация” |
| 4 | Изпращане на текста „Steganography” |
| 5 | Изпращане на текста “Мрежова стеганография” |
| 6 | Изпращане на текста „Компютърна стеганография” |

*4. Методически указания*

По задача 3.1.

За изпълнение на тази задача е необходимо да се направят следните действия:

1. Стартира се програмата за анализ на мрежови трафик Wireshark. Основният прозорец на програмата е показан на фигура ХХ5.



Фиг. ХХ5 Основен прозорец на програмата Wireshark

2. За да започне анализа на мрежовия трафик трябва да се избере *Capture Options*, в резултат на което се появява диалог за въвеждане на параметрите за прихващане на патеките (фиг. ХХ6).



Фиг. ХХ6 Основен прозорец на програмата Wireshark

В този диалогов прозорец трябва да се укаже мрежовия интерфейс, който ще се прослушва и да се зададат правила за прихващане на пакетите в полето *Capture Filter*. Ако се анализира трафик на програми, стартирани на един компютър се използва адреса loopback (lo) – 127.0.0.1.

3. За да се филтрира мрежовия трафик трябва да се въведе правило в полето *Capture Filter*.

Формирането на филтър (правило) ще се поясни с няколко примера:

Ако е необходимо да се прихванат всички ICPM пакети, в полето на филтъра се въвежда *icmp*.

Ако е необходимо да се разгледа целия трафик с изключение на ICMP пакетите, трябва да се добави *not*, с което правилото ще изглежда - *not icmp*.

Правилата могат да бъдат по-сложни, за да могат да се следят няколко параметъра заедно. Например, ако е необходимо да се проследи SSH трафик, трябва да се уточни протокол и порт. В този случай правилото е *tcp port* 22.

В правилата може да участват и логическите операции *and* и *or*. Ако трябва да се прихванат всички пакети с изключение на TCP на порт 22, следва да се въведе правилото - *not tcp and port 22*.

Прихванатите пакети се изобразяват в главния прозорец на програмата във вида, показан на фиг. ХХ7. За всеки пакет в долната част на прозореца може да се получи детайлна информация.



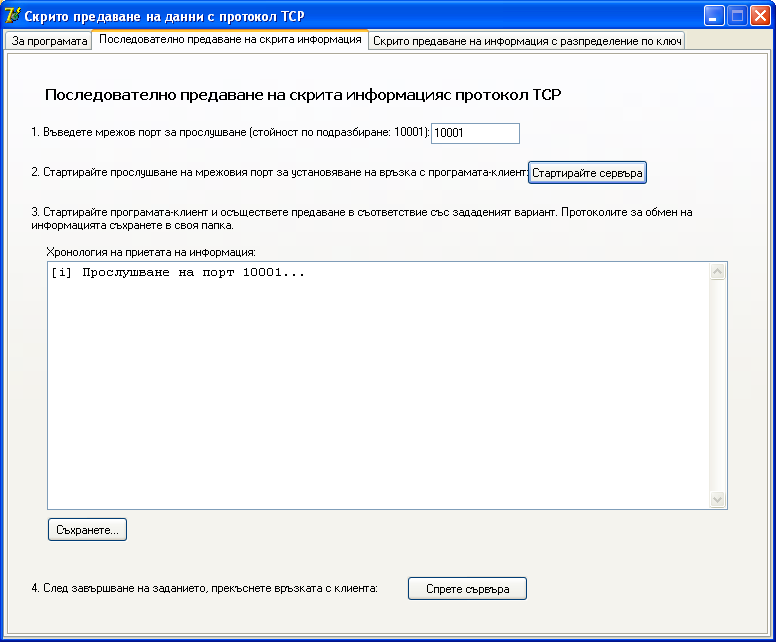
Фиг. ХХ7 Списък на прихванатите пакети

По задача 3.2.

Точка 3.2.1.

За изпълнение на тази задача е необходимо да използват програмата-клиент labclnt.exe и програмата-сървър labsrv.exe. Тези програми могат да бъдат използвани по два начина: стартирани на един компютър или на два, взаимодействащи си чрез локална мрежа или Интернет.

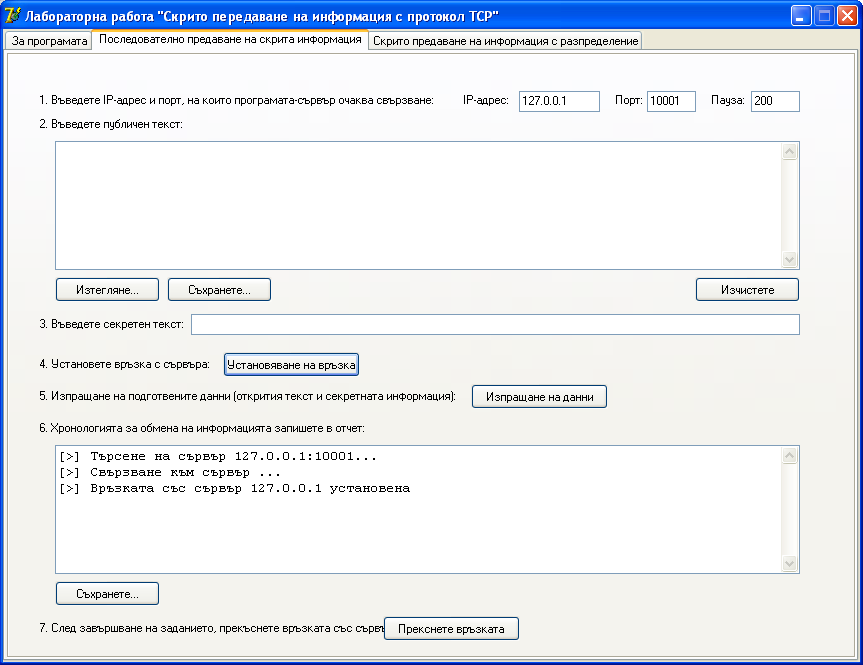
Първо се стартира програмата-сървър, в която трябва да се посочи порта за прослушване от сървъра, а след това се натиска бутон „*Стартирайте сървъра*” (фиг. ХХ8).



Фиг. ХХ8 Установяване на порт за прослушване и стартиране на сървъра

След това се стартира и програмата-клиент, с помощта на която ще се предава скритата информация.

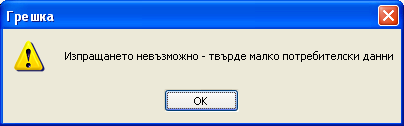
Независимо от това дали двете програми ще бъдат стартирани на един или на два компютъра, в програмата-клиент е необходимо да се укаже IP адреса на компютъра, в локалната мрежа или Интернет, на който е стартирана програмата-сървър. Ако програмите се стартират на един компютър, за адрес на сървъра се използва 127.0.0.1. Посочва се и същия порт, който е използван в програмата-сървър. След натискане на бутона „*Установяване на връзка*”, двете програми са готови за обмен на данни (фиг. ХХ9)



Фиг. ХХ9 Установена връзка между двете програми

Точка 3.2.2.

За прикриване на изпращането на скрития текст е необходим явен текст, който може да се въведе ръчно в съответното поле или да се добави от външен файл с разширение *txt.* Дължината на открития текст е зависима от дължината на скрития. Минималната дължина на явния текст не е точно определена и зависи от ASCII кодовете на символите от скрития текст. Ако явния текст не е с необходимата дължина се появява съобщението показано на фиг. Х10.

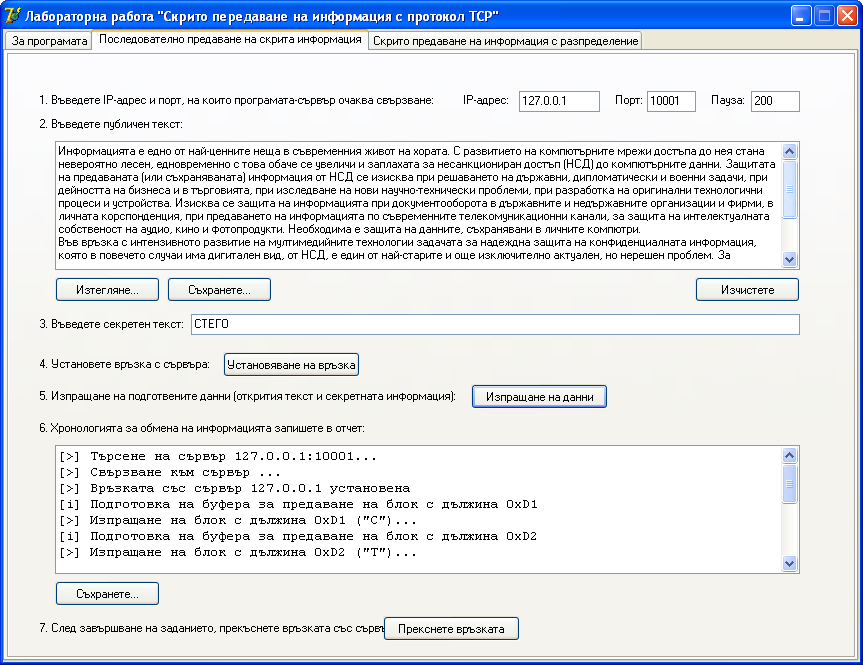


Фиг. Х10 Съобщение за недостатъчна дължина на явния текст

Въвеждането на скрия текст се извършва ръчно в съответното поле. Допустимо е в текста да присъстват букви от кирилица, латиница, цифри и допълнителни символи.

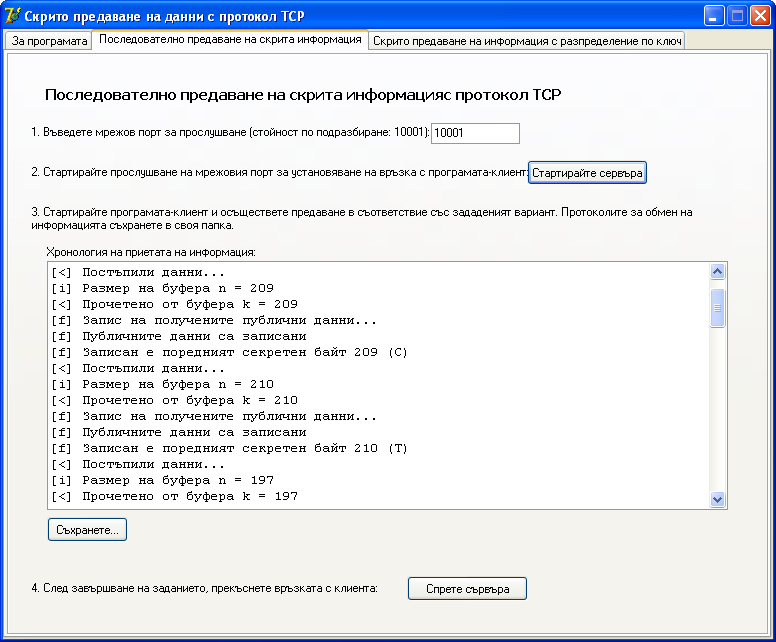
Следва изпращането на данни с натискане на бутона „Изпращане на данни”. В обособени полета от прозорците на двете програми може да се следи изпращането (приемането) на данни.

На фигура Х11 е показано примерно приложение на програмата-клиент за изпращане на скрит текст „СТЕГО”



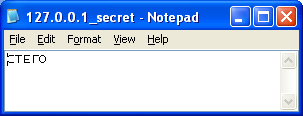
Фиг. Х11 Използване на програмата-клиент за изпращане на скрит текст „СТЕГО”

В програмата-сървър се получават данните от програмата-клиент, като се показва резултата от обработката (фиг. Х12).



Фиг. Х12 Приемане на скрития текст „СТЕГО” от програмата-сървър

В папката, в която се съхранява програмата-сървър, след приемането на данните се създават два файла с име във формат: *ipaddress\_public.txt* и *ipaddress\_secret.txt*, където *ipaddress* е адресът на сървъра. Файлът *ipaddress\_public.txt* съдържа явния текст, а файла *ipaddress\_secret.tx* съдържа скрития текст (фиг. Х13).



Фиг. Х13 Файлът *127.0.0.1\_secret* и неговото съдържание – скрития текст „СТЕГО”

За анализиране на мрежовия трафик, генериран от двете програми, може да се използва програмата Wireshark. За проследяването е необходимо да се приложи коректно правило, филтриращо дължината на TCP сегмента. Тъй като диапазона на знаците от ASCII таблицата, които традиционно се използват за представяне на текст, са в интервала от 32 до 254, то филтъра би трябвало да бъде следния:

*tcp.len >=32 and tcp.len <=254 and tcp.port ==10001*

*5. Контролни въпроси:*

1. Какви са контейнерите в мрежовата стеганография?

2. От колко нива е съставен модела OSI и кои са те?

3. На кои нива от OSI модела работят протоколите от стека TCP/IP?

4. Какви са основните функции на мрежовите анализатори?

5. Кои са основните стъпки за правилното конфигуриране на програмата-клиент labclnt.exe и програмата-сървър labsrv.exe?

6. По какъв начин се извлича секретният текст при обмен с модификация на дължината на TCP сегмент и къде се записва?

7. Какво трябва да се съобразява при съставянето на правилото за филтриране на трафик за прихващане на пакетите, генерирани от програмите за скрито предаване с протокола TCP?