



Projektová dokumentácia

ISA Projekt

Tunelovanie dátových prenosov cez DNS dotazy

Kičinka Kristián (xkicin02)

Obsah

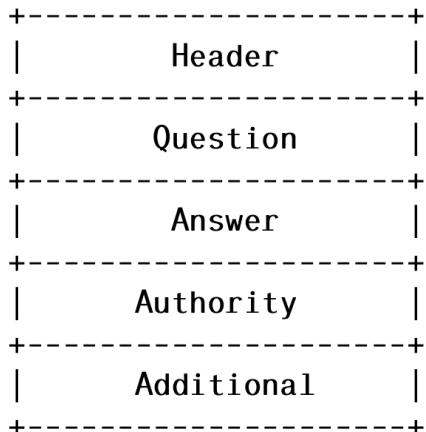
<i>Úvod do problematiky</i>	3
<i>Návrh aplikácie</i>	4
<i>Popis implementácie</i>	5
DNS Sender	5
DNS Receiver.....	6
<i>Obmedzenia/nedostatky</i>	6
<i>Testovanie.....</i>	6
Test 01	7
Test 02	8
Test 03	11
Test 04	13
Test 05	15
<i>Meranie.....</i>	17
<i>Zdroje.....</i>	18

Úvod do problematiky

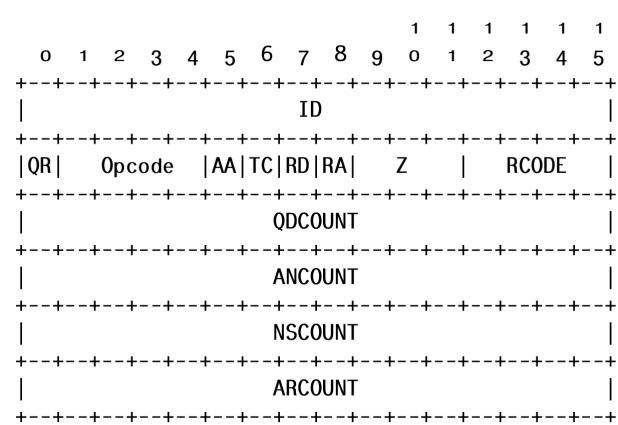
Cieľom projektu je vytvoriť funkčný nástroj, ktorý by realizoval tunelovanie dát prostredníctvom DNS komunikácie. Tunelovanie DNS je metóda, ktorá umožňuje posielat' ľubovoľné dátu prostredníctvom DNS protokolu. Tento protokol neboli pôvodne navrhnutý na takéto účely, avšak je možné využiť jeho vlastnosti na takéto účely. Do DNS query zabalíme komunikáciu, ktorú chceme posielat' na DNS receiver. V odpovedi sa vráti komunikácia, odoslaná z receivera. [1]

Systém DNS záznamov slúži na internete a v počítačových sietiach na prekladanie doménových názvov na IP adresy, zabezpečuje tak zjednodušenie komunikácie a pohybu na internete. [3]

V nasledujúcich obrázkoch je zobrazená štruktúra DNS hlavičky a DNS paketu. [2] V projekte využívame DNS paket na prenos informácií. Na to aby sme mohli odoslať korektný DNS paket je nutné najprv nastaviť správne DNS hlavičku a vytvoriť z dát korektnú DNS query. V našom prípade v hlavičke nastavujeme počet DNS otázok na 1 a povoľujeme rekurziu nastavením RD v hlavičke na 1.



Obrázok 1 (Štruktúra DNS paketu)



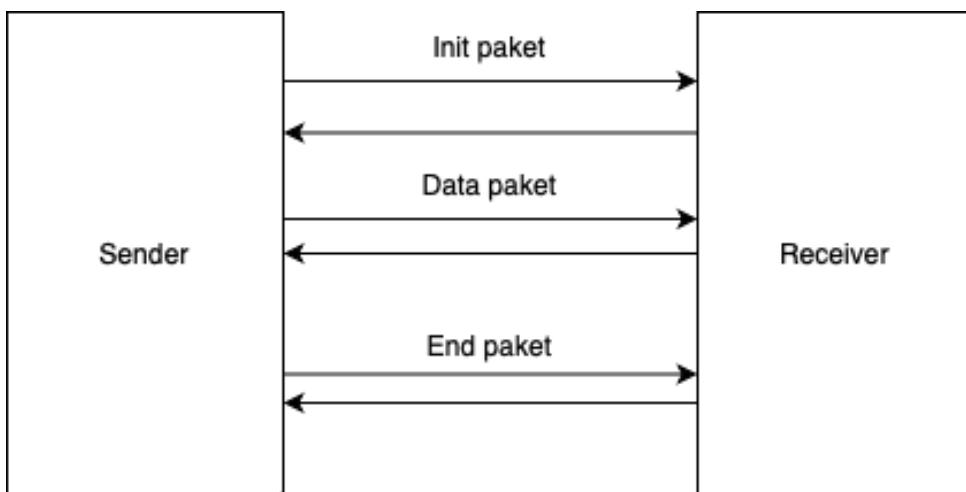
Obrázok 2 (Štruktúra hlavičky DNS paketu)

Návrh aplikácie

Celková aplikácia DNS tunelovania sa skladá z dvoch samostatných modulov. Jedná sa o modul *dns_sender* a *dns_receiver*.

Hlavnou úlohou senderu vo všeobecnosti je odosielanie dát receiveru prostredníctvom DNS queryes. Po spustení modulu so správnymi parametrami sa začína proces odosielania požiadavkou. Program otvorí súbor zadaný ako parameter, a načíta dátu. Následne sú dátu zakódované do formátu base_32, nakoľko je nutné zabezpečiť aby DNS query obsahovala len korektné znaky. Po procese zakódovania sa k dátam pridáva *base_host*, ktorý je zadný používateľom aplikácie. V ďalšom kroku program transformuje predošlé dátu do formátu vhodného pre DNS query. Po vykonaní tohto spracovania sa pristupuje k otvoreniu soketu, vytvoreniu a nastaveniu DNS paketu a samotnému odoslaniu spracovaných dát modulu *dns_receiver*. Po odoslaní paketu sender čaká na odpoveď od receivera. Až po prijatí odpovede sa posielajú ďalší paket. Celkovo rozlišujeme 3 typy paketov, ktoré preposielame a spracúvame. Jedná sa o Init paket, Data paket a End paket. Init paket umožňuje nadviazanie spojenia, dopĺňa informácie, potrebné pre správne nastavenie prenosu *dns_receivera*. Obsahuje cestu k súboru, ktorý má byť uložený na strane receivera. Dátové pakety sú odosielané bezprostredne po potvrdení Init paketu. Obsahujú samotné dátu, ktoré boli načítané zo vstupného súboru a spracované. End paket je odoslaný v prípade ak už boli všetky dátu zo vstupného súboru načítané a odoslané.

Modul dns receivera zabezpečuje príjem DNS paketov a ich nasledovné spracovávanie. Spustením tohto modulu s korektnými parametrami sa spúšťa proces zachytávania DNS komunikácie. Prijatím paketu program pristúpi k spracovaniu prijatého paketu. Pri spracovaní sa okrem iného určuje typ daného paketu, či sa jedná o Init, Data alebo End paket. Následne sa porovnáva prijatý base host s base hostom zadaným ako parameter *dns_receivera*. Ak sa zhodujú môže sa pokračovať v ďalšom spracovaní, už konkrétnie pre daný typ paketu. Modul pracuje v slučke, až do ukončenia používateľom.



Obrázok 3 (Model komunikácie medzi modulmi aplikácie)

Popis implementácie

DNS Sender

Modul dns sendera je implementovaný v súbore *sender/dns_sender.c*. Modul zabezpečuje načítavanie a odosielanie dát DNS receiveru. Pri spustení programu je nutné nastaviť korektné argumenty. Podrobnejší popis spustenia modulu je bližšie popísaný v priloženom README.md súbore. Po spustení programu sa najprv spracovávajú argumenty programu. Jednotlivé argumenty sa uložia do štruktúry *SenderArguments*. Spracovanie a uloženie argumentov zabezpečuje funkcia *parseSenderArguments()*, ktorá sa nachádza v súbore *program_arguments.c*. Vo funkcii sa najprv inicializuje vyššie spomenutá štruktúra *SenderArguments*, následne sa prechádzajú argumenty programu v poli *argv[]* na fixných pozíciách a ich hodnota je ukladaná do štruktúry.

Ak načítanie a spracovanie argumentov prebehlo úspešne pristupuje sa k vytvoreniu a odoslaniu Init paketu. Odosielanie paketov zabezpečuje funkcia *sendSenderData()*, nachádzajúca sa v súbore *dns_sender.c*. Funkcia prijíma ako parameter argumenty modulu, data payload, jeho veľkosť a typ paketu. Pri type paketu rozlišujeme INIT_PACKET, DATA_PACKET a END_PACKET. Vo funkcii *sendSenderData()* najprv vykonáme zakódovanie dát do formátu base32. Na to nám slúži funkcia *base32_encode()* umiestnená v súbore *base32.c*. Ďalej sa za zakódované dátá pridá base host z argumentov modulu prostredníctvom funkcie *strcat()*. Následne sa vyberá IP adresa, na ktorú má byť cieľová komunikácia smerovaná. IP adresa môže byť zadaná v argumentoch, alebo je zvolená implicitná adresa. Pri situácii, kedy je adresa zadaná používateľom, sa na základe typu paketu volajú príslušné funkcie *sendInitPacket()*, *sendDataPacket()* alebo *sendEndPacket()*.

Funkcia *sendInitPacket()* požaduje IP adresu, v tomto prípade zadanú používateľom. Požadované sú aj zakódované dátá a base host. Pri init pakete sa za dátá považuje DST_FILEPATH z argumentov programu. Pri end pakete dátá predstavuje reťazec „END_PACKET“. Dátá v data pakete sú dátá načítané funkciou *loadData()* umiestnenou v súbore *dns_sender.c*. Táto funkcia otvorí súbor na základe cesty z sender argumentov, ak nie je zadaná, číta sa z *stdin*. Dátá sú čítané binárne, aby bolo možné zabezpečiť prenos súborov ako napr. obrázok alebo video. Zo súboru je čítaná stále fixná dĺžka znakov, nakoľko je nutné pri prenose dát dátá deliť do paketov a odosielat postupne. Okrem samotných načítaných dát sa eviduje aj ich dĺžka. V prípade, že IP adresa nie je zadaná používateľom sa prechádza k získaniu implicitnej IP adresy. Túto operáciu zabezpečuje funkcia *getImplicitDNSserverIP()*.

Funkcia otvorí súbor „*/etc/resolv.conf*“ vyhľadá riadok začínajúci reťazcom „*nameserver*“ a vyberie IP adresu. V prípade ak sa v súbore nachádza viac IP adres, je stále vybraná len prvá adresa. Ak bola adresa vybraná pristupuje sa taktiež k volaniu vyššie spomínaných funkcií *sendInitPacket()*, *sendDataPacket()* alebo *sendEndPacket()*. V každej z týchto funkcií sa najprv vytvorí štruktúra *sockaddr_in*, do ktorej sa nastaví IP adresa, DNS port (53) a informácia, že ide o IPv4 adresu. V ďalšom kroku je volaná funkcia *sendDataToDnsIP()*, umiestnená v súbore *connection.c*. Posledným krokom je zvýšenie packet_id a v prípade dátových paketov aj chunk_id. Vo funkcii *sendDataToDnsIP()* dochádza k samotnému vytvoreniu a odoslaniu paketov. Pomocou funkcie *createSocket()* sa vytvára a inicializuje soket. Volaním funkcie *initSenderDNSheader()* sa vytvorí a inicializuje DNS hlavička paketu. V hlavičke paketu sa nastaví id paketu, počet DNS queries, rekurzia , a return code. Return code je využívaný pri identifikácii typu paketov. Každý typ paketu má pridelený špecifický return code z nevyužívaných. Init

paket má pridelený kód 12, data paket 13 a end paket kód 14. Následne sa vytvára DNS query pomocou funkcie *createDNSquery()*. Odoslanie paketu zabezpečí funkcia *sendto()*. Po úspešnom odoslaní paketu sa čaká na odpoveď. Ak by odpovede nedorazila, paket je posielaný znova. Ak je prijatá odpoveď ukončuje sa soket a môže sa odosielat ďalší paket. Ak sú posланé celé vstupné dátá, odosiela sa end paket a po prijatí odpovede sa ukončuje činnosť modulu.

DNS Receiver

Modul DNS receivera je implementovaný v súbore */receiver/dns_receiver.c*. Tento modul zabezpečuje zachytávanie DNS paketov posielaných DNS senderom, spracovávanie dát a ich uloženie. Pri spustení modulu je nutné spúštať s istými parametrami. Podrobnejší popis spustenia je popísaný v README.md. Po úspešnom spustení aplikácie sa pristupuje k spracovaniu argumentov programu. Argumenty sa uložia do štruktúry *receiverArguments*. Samotné spracovanie zabezpečuje funkcia *parseReceiverArguments()* umiestnená v súbore *program_arguments.c*. Bezprostredne po spracovaní argumentov sa pristupuje k vytvoreniu a nastaveniu soketu pomocou funkcií *createSocket()* a *setsockopt()*. Následne je vytvorený buffer do ktorého sa bude ukladať prijatý paket. Ďalej je vytvorená a nastavená štruktúra *sockaddr_in*. V nej evidujeme informácie o type adresy dns receivera, IP adresy a portu. Pri nastavení adresy využívame makro „*INADDR_ANY*“. Následne je soketu nastavené meno pomocou funkcie *bind()*. Následne sa pristupuje k zachytávaniu DNS komunikácie pomocou *recvfrom()*. Každý prijatý paket je rozdelený na dns hlavičku a data payload. Z DNS hlavičky zistujeme typ prijatého paketu pomocou return kódu. Ak je identifikovaný typ paketu pristupuje sa k spracovaniu dát, ktoré paket nesie. Pri spracovaní dát sa overuje zhoda base hostov. Ak sa base hosty nezhodujú nedochádza k ďalšiemu spracovaniu. Ak je prijatý paket init paket, obsahuje cestu súboru. Následne je cesta vybraná z paketu a spojená s receiver časťou. Tým vznikne celková cesta k výstupnému súboru. Pri dátovom pakete sa pristupuje k zápisu do out súboru. Funkcia *getDataFromPayload()* spracúva data payload. Odstráni z dát znaky ktoré by bránili dekódovaniu a následne vykoná dekódovanie. Ak sú dátá úspešne zapísané do súboru. Dochádza k odoslaní odpovede DNS senderu. Ako odpoveď je preposielaný prijatý paket avšak s modifikáciou DNS hlavičky. Query response je nastavená na hodnotu 1.

Obmedzenia/nedostatky

- Fixné nastavenie parametrov modulu *dns_sender* a *dns_receiver*, nie je možné meniť poradie parametrov skriptu. Správne poradie je popísané v README.md
- Absencia podpory IPv6 adries
- Pri výpadku komunikácie DNS sender opakuje odoslanie paketu
- Pri odosielaní väčších súborov dochádza sporadicky k zaseknutiu komunikácie, je potrebné znova spustiť oba moduly.
- Pri nastavaní DST_DIRPATH je nutné aby adresár už existoval. Program ho nevytvorí automaticky.

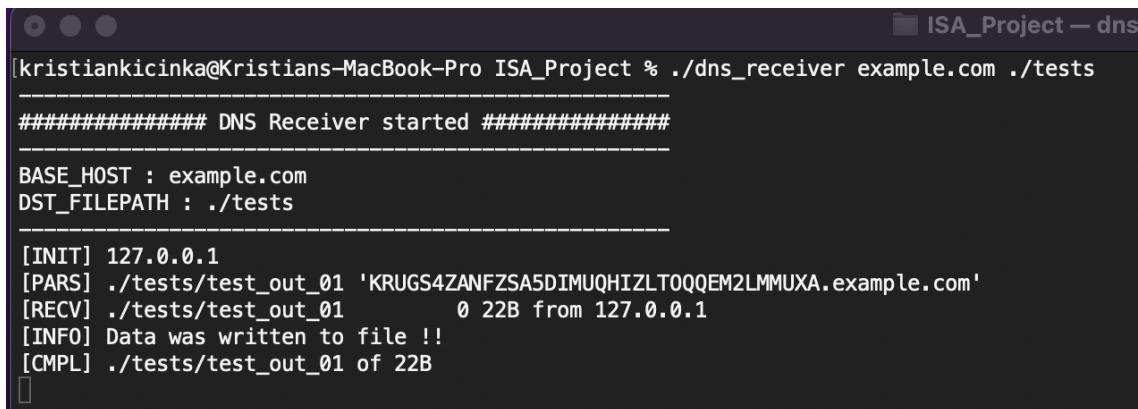
Testovanie

Táto časť je venovaná testovaniu výslednej aplikácie, resp. *dns_sender* a *dns_receiver* modulu. V niekoľkých testoch overíme správnu funkčnosť aplikácií. Jednotlivé testy budú vykonávané na lokálnom zariadení ale aj na serveri eva.fit.vutbr.cz. Na lokálnom zariadení je nainštalovaný operačný systém macOS Ventura (verzia 13.0.1). Pri analýze sietovej komunikácie využívame aplikáciu Wireshark.

Test 01

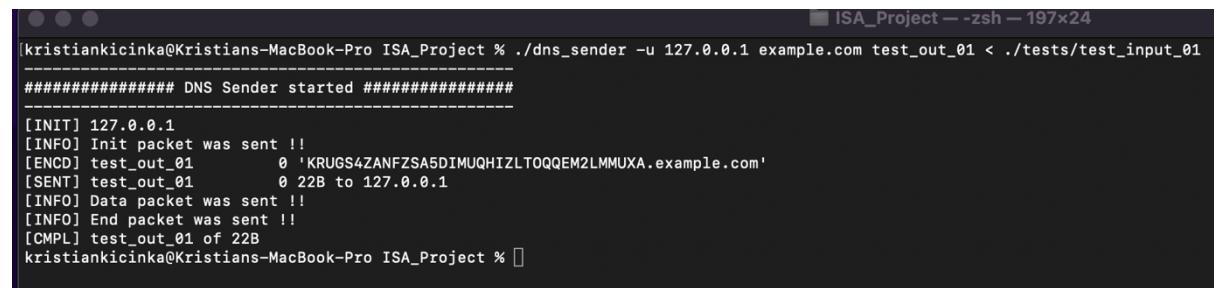
V tomto teste odosielame malý súbor obsahujúci reťazec „This is the test File“ na receiver umiestnený na adresu 127.0.0.1. Ako base host bol zvolený reťazec „example.com“, cieľová zložka DNS receivera je ./tests. Prenesené dátá budú uložené do súboru test_out_01.

Modul DNS receiver spúšťame príkazom „./dns_receiver example.com ./tests“.



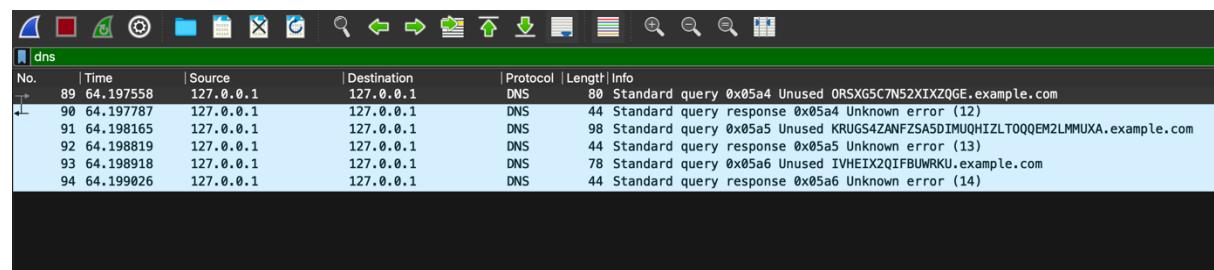
```
[kristiankicinka@Kristians-MacBook-Pro ISA_Project % ./dns_receiver example.com ./tests
#####
BASE_HOST : example.com
DST_FILEPATH : ./tests
#####
[INIT] 127.0.0.1
[PARS] ./tests/test_out_01 'KRUGS4ZANFZSA5DIMUQHIZLTOQQEM2LMMUXA.example.com'
[RECV] ./tests/test_out_01      0 22B from 127.0.0.1
[INFO] Data was written to file !!
[CMPL] ./tests/test_out_01 of 22B
```

Obrázok 4 (Výstup DNS receivera)



```
[kristiankicinka@Kristians-MacBook-Pro ISA_Project % ./dns_sender -u 127.0.0.1 example.com test_out_01 < ./tests/test_input_01
#####
DNS Sender started #####
#####
[INIT] 127.0.0.1
[INFO] Init packet was sent !!
[ENCD] test_out_01      0 'KRUGS4ZANFZSA5DIMUQHIZLTOQQEM2LMMUXA.example.com'
[SENT] test_out_01      0 22B to 127.0.0.1
[INFO] Data packet was sent !!
[INFO] End packet was sent !!
[CMPL] test_out_01 of 22B
kristiankicinka@Kristians-MacBook-Pro ISA_Project % ]
```

Obrázok 5 (Výstup DNS sendera)



No.	Time	Source	Destination	Protocol	Length	Info
89	64.197558	127.0.0.1	127.0.0.1	DNS	80	Standard query 0x05a4 Unused ORSXG5C7N52XIXZQGE.example.com
90	64.197787	127.0.0.1	127.0.0.1	DNS	44	Standard query response 0x05a4 Unknown error (12)
91	64.198165	127.0.0.1	127.0.0.1	DNS	98	Standard query 0x05a5 Unused KRUGS4ZANFZSA5DIMUQHIZLTOQQEM2LMMUXA.example.com
92	64.198819	127.0.0.1	127.0.0.1	DNS	44	Standard query response 0x05a5 Unknown error (13)
93	64.198918	127.0.0.1	127.0.0.1	DNS	78	Standard query 0x05a6 Unused IVHEIX2QIFBUWRKU.example.com
94	64.199026	127.0.0.1	127.0.0.1	DNS	44	Standard query response 0x05a6 Unknown error (14)

Obrázok 6 (Výstup programu Wireshark)

```

> Frame 91: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface lo0, id 0
> Null/Loopback
> Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
> User Datagram Protocol, Src Port: 60143, Dst Port: 53
└ Domain Name System (query)
    Transaction ID: 0x05a5
    Flags: 0x010d Standard query
        0... .... .... = Response: Message is a query
        .000 0.... .... = Opcode: Standard query (0)
        .... 0. .... .... = Truncated: Message is not truncated
        .... 1 .... .... = Recursion desired: Do query recursively
        .... 0.. .... = Z: reserved (0)
        .... ...0 .... = Non-authenticated data: Unacceptable
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
    Queries
        KRUGS4ZANFZSA5DIMUQHIZLTOQQEM2LMMUXA.example.com: type Unused, class Unknown
            Name: KRUGS4ZANFZSA5DIMUQHIZLTOQQEM2LMMUXA.example.com
                [Name Length: 48]
                [Label Count: 3]
                Type: Unused (0)
                Class: Unknown (0x0100)
\[Response In: 92\]

0000  02 00 00 00 45 00 00 5e  c7 b2 00 00 40 11 00 00  . . . E . ^ . . . @ . .
0010  7f 00 00 01 7f 00 00 01  ea ef 00 35 00 4a fe 5d  . . . . . . . . . . . . 5 . J . ]
0020  05 a5 01 0d 00 01 00 00  00 00 00 00 24 4b 52 55  . . . . . . . . . . . . $KRU
0030  47 53 34 5a 41 4e 46 5a  53 41 35 44 49 4d 55 51  GS4ZANFZ SA5DIMUQ
0040  48 49 5a 4c 54 4f 51 51  45 4d 32 4c 4d 4d 55 58  HIZLTOQQ EM2LMMUX
0050  41 07 65 78 61 6d 70 6c  65 03 63 6f 6d 00 00 00  A . example.com . .
0060  01 00

```

Obrázok 7 (Detailné zobrazenie DNS paketu)

```
[kristiankicinka@Kristians-MacBook-Pro ISA_Project % cat tests/test_input_01
This is the test File.%]
kristiankicinka@Kristians-MacBook-Pro ISA_Project % cat tests/test_out_01
This is the test File.%]
kristiankicinka@Kristians-MacBook-Pro ISA_Project % ]
```

Obrázok 8 (Porovnanie vstupu a výstupu)

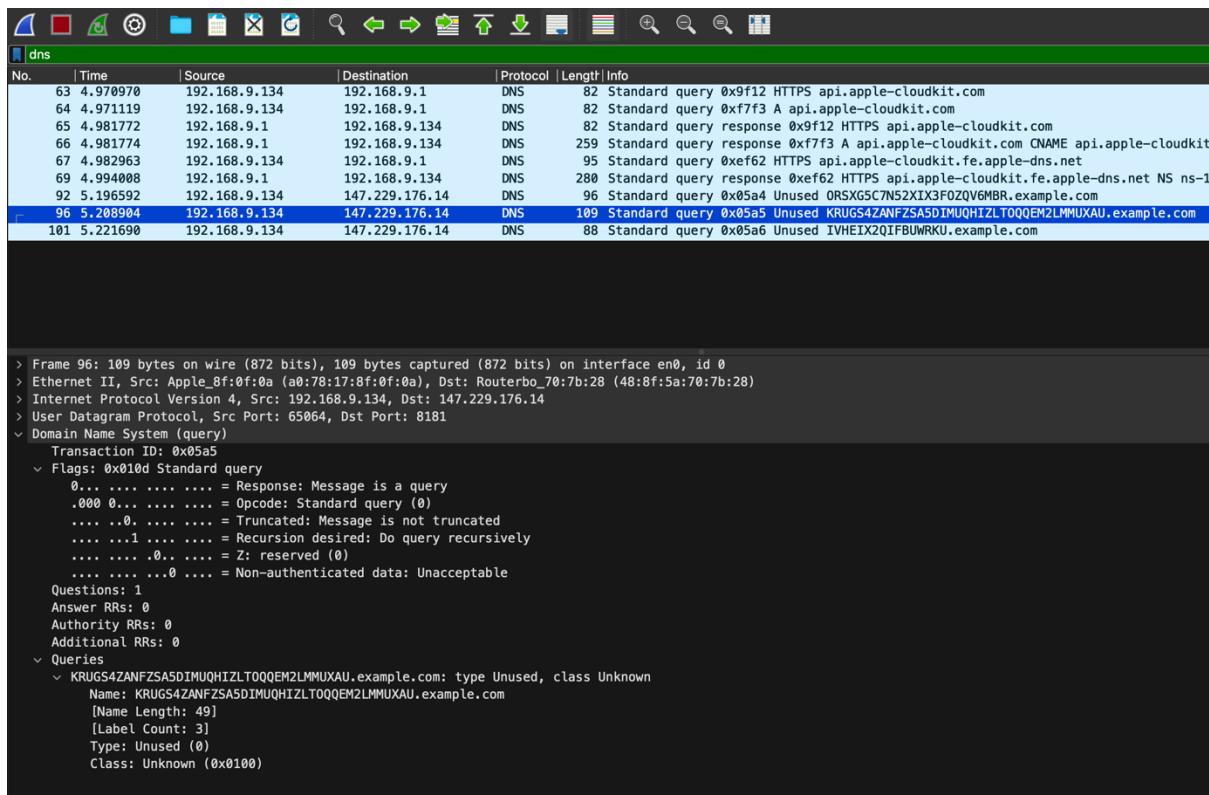
Porovnaním vstupného súboru DNS sendera a výstupného súboru, ktorý vytvoril DNS receiver zistíme, že obsah súboru bol korektnie prenesený DNS tunelom.

Test 02

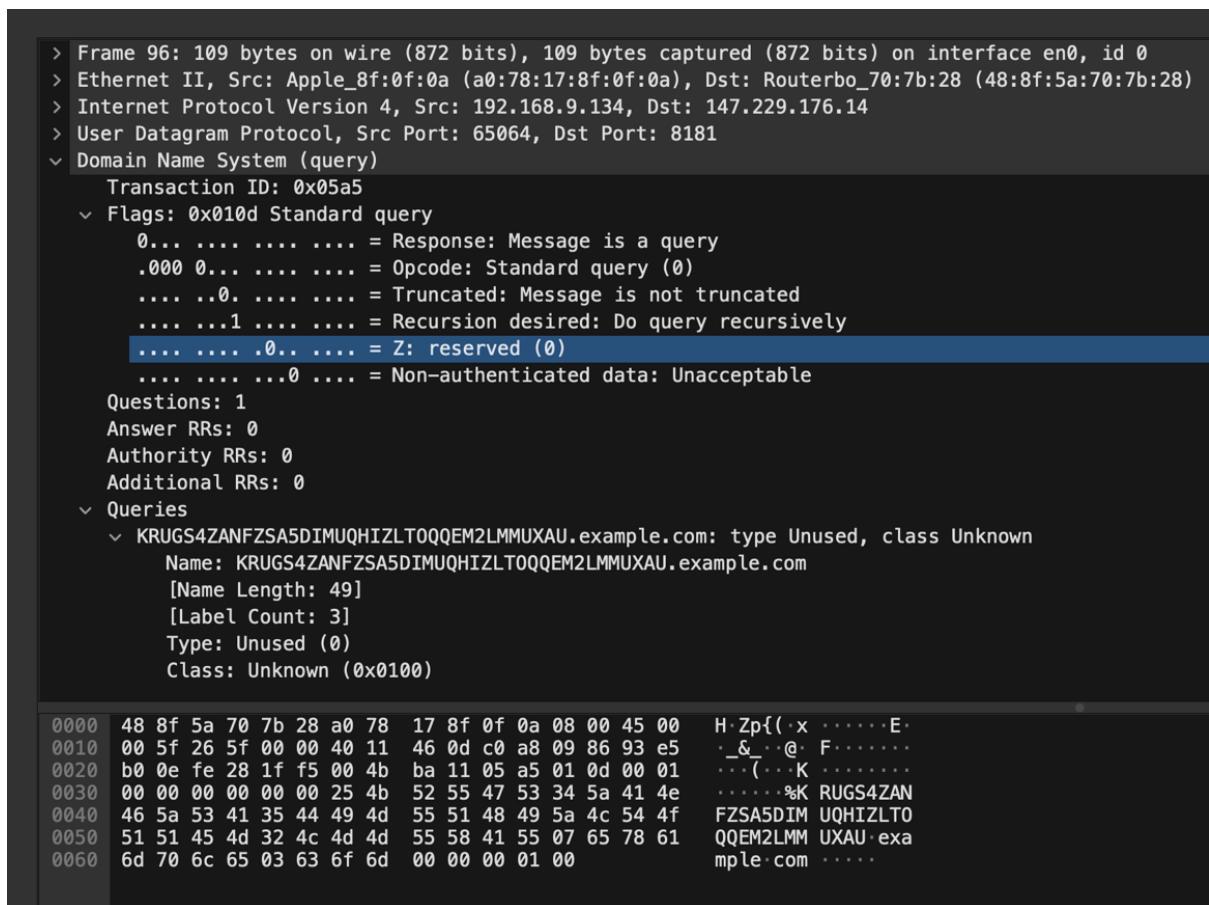
V tomto teste overujeme možnosť posielania dát na vzdialený server. V teste budeme posielat dátá na server eva.fit.vutbr.cz. Server má pridelenú IP adresu 147.229.176.14. Ako base host použijeme reťazec „example.com“. Dátá na receiveri budú ukladané do priečinka tests. Vstupný súbor je zhodný so súborom v predošлом teste. Výstup bude uložený do súboru /tests/test_out_eva_01.

```

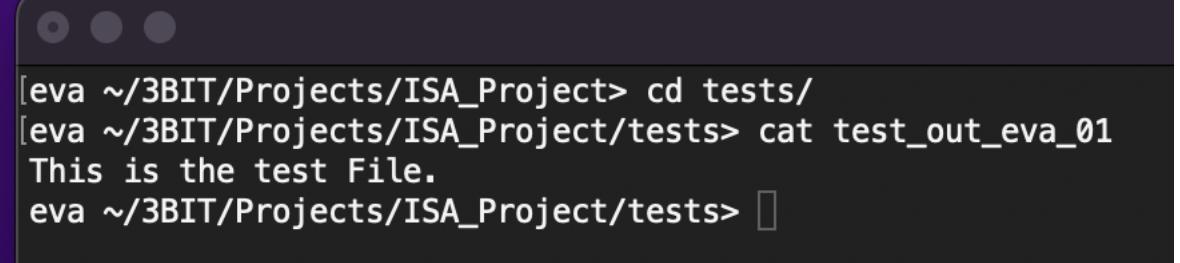
eva ~/3BIT/Projects/ISA_Project> ifconfig
ixl0: flags=8822<BRIDGE,BROADCAST,SIMPLEX,MULTICAST> metric 0 mtu 1500
    options=&#4e07bb-RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING,JUMBO_MTU,VLAN_HWCSSUM,TS04,TS06,LRO,VLAN_HWFILTER,VLAN_HWT50,RXCSUM_IPV6,TXCSUM_IPV6,NOMAP>
    ether 00:25:90:be:33:dc
    media: Ethernet autoselect (100baseT <full-duplex>)
    status: active
    nd6 options=29<PERFORMNUD,IFDISABLED,AUTO_LINKLOCAL>
ixl1: flags=8822<BRIDGE,BROADCAST,SIMPLEX,MULTICAST> metric 0 mtu 1500
    options=&#4e07bb-RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING,JUMBO_MTU,VLAN_HWCSSUM,TS04,TS06,LRO,VLAN_HWFILTER,VLAN_HWT50,RXCSUM_IPV6,TXCSUM_IPV6,NOMAP&gt;
    ether 00:25:90:be:33:dd
    media: Ethernet autoselect
    status: no carrier
    nd6 options=29&lt;PERFORMNUD,IFDISABLED,AUTO_LINKLOCAL&gt;
ix0: flags=8822&lt;UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST&gt; metric 0 mtu 1500
    options=&#4e07bb-RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING,JUMBO_MTU,VLAN_HWCSSUM,TS04,TS06,LRO,WOL_UCAST,WOL_MCAST,WOL_MAGIC,VLAN_HWFILTER,VLAN_HWT50,RXCSUM_IPV6,TXCSUM_IPV6,NOMAP&gt;
    ether ac:1f:b0:ab:fc:7a
    inet 147.229.176.14 netmask 0xffffffff broadcast 147.229.177.255
        inet6 fe80::ac1f:b0ff:feab:fc7%ix0 prefixlen 64 scopeid 0x23
    inet6 fe80::ac1f:b0ff:feab:fc7%ix0 prefixlen 64 scopeid 0x4
    media: Ethernet autoselect (100base-Twimax &lt;full-duplex&gt;)
    status: active
    nd6 options=21&lt;PERFORMNUD,AUTO_LINKLOCAL&gt;
lo0: flags=8043&lt;IP,BROADCAST,RUNNING,MULTICAST&gt; metric 0 mtu 16384
    options=&#4e0003-RXCSUM,TXCSUM,LINKSTATE,RXCSUM_IPV6,TXCSUM_IPV6&gt;
    inet 127.0.0.1 netmask 0xffffffff
        groups: lo
    nd6 options=21&lt;PERFORMNUD,AUTO_LINKLOCAL&gt;
eva ~/3BIT/Projects/ISA_Project&gt; 
</pre>
</div>
<div data-bbox="358 305 634 319" data-label="Caption">
<p>Obrázok 9 (Zobrazenie IP konfigurácie [Eva])</p>
</div>
<div data-bbox="117 328 879 538" data-label="Text">
<img alt="Screenshot of a terminal window showing the execution of ./dns_receiver example.com ./tests. The output includes log messages from the DNS Receiver starting and processing a file named ./tests."/>
<pre>
[eva ~/3BIT/Projects/ISA_Project&gt; ./dns_receiver example.com ./tests
#####
# DNS Receiver started #####
#####
BASE_HOST : example.com
DST_FILEPATH : ./tests
#####
[INIT] 88.146.232.18
[PARS] ./tests/test_out_eva_01 'KRUGS4ZANFZSA5DIMUQHIZLTOQQEM2LMMUXAU.example.com'
[RECV] ./tests/test_out_eva_01          0 23B from 88.146.232.18
[INFO] Data was written to file !!
[CMPL] ./tests/test_out_eva_01 of 23B
</pre>
</div>
<div data-bbox="387 553 606 566" data-label="Caption">
<p>Obrázok 10 (Výstup DNS receivera)</p>
</div>
<div data-bbox="117 575 879 696" data-label="Text">
<img alt="Screenshot of a terminal window showing the execution of ./dns_sender -u 147.229.176.14 example.com test_out_eva_01 &lt; ./tests/test_input_01. The output includes log messages from the DNS Sender starting and sending a data packet."/>
<pre>
[kristiankicinka@Kristians-MacBook-Pro ISA_Project % ./dns_sender -u 147.229.176.14 example.com test_out_eva_01 &lt; ./tests/test_input_01
#####
# DNS Sender started #####
#####
[INIT] 147.229.176.14
[INFO] Init packet was sent !!
[ENCD] test_out_eva_01      0 'KRUGS4ZANFZSA5DIMUQHIZLTOQQEM2LMMUXAU.example.com'
[SENT] test_out_eva_01      0 23B to 147.229.176.14
[INFO] Data packet was sent !!
[INFO] End packet was sent !!
[CMPL] test_out_eva_01 of 23B
kristiankicinka@Kristians-MacBook-Pro ISA_Project % 
</pre>
</div>
<div data-bbox="391 711 603 725" data-label="Caption">
<p>Obrázok 11 (Výstup DNS sendera)</p>
</div>
```



Obrázok 12 (Výstup aplikácie Wireshark)

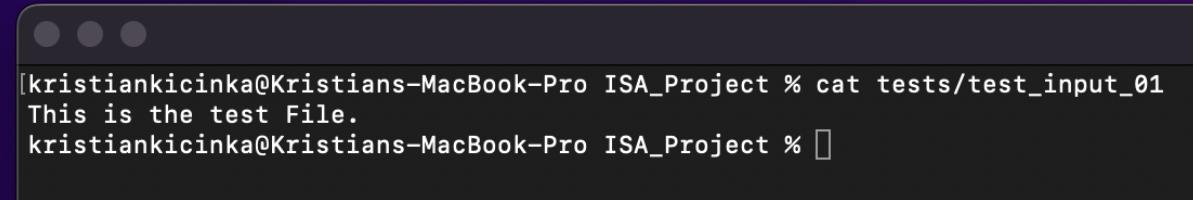


Obrázok 13 (Detailné zobrazenie paketu)



```
[eva ~/3BIT/Projects/ISA_Project> cd tests/
[eva ~/3BIT/Projects/ISA_Project/tests> cat test_out_eva_01
This is the test File.
eva ~/3BIT/Projects/ISA_Project/tests> ]
```

Obrázok 14 (Zobrazenie výstupného súboru)



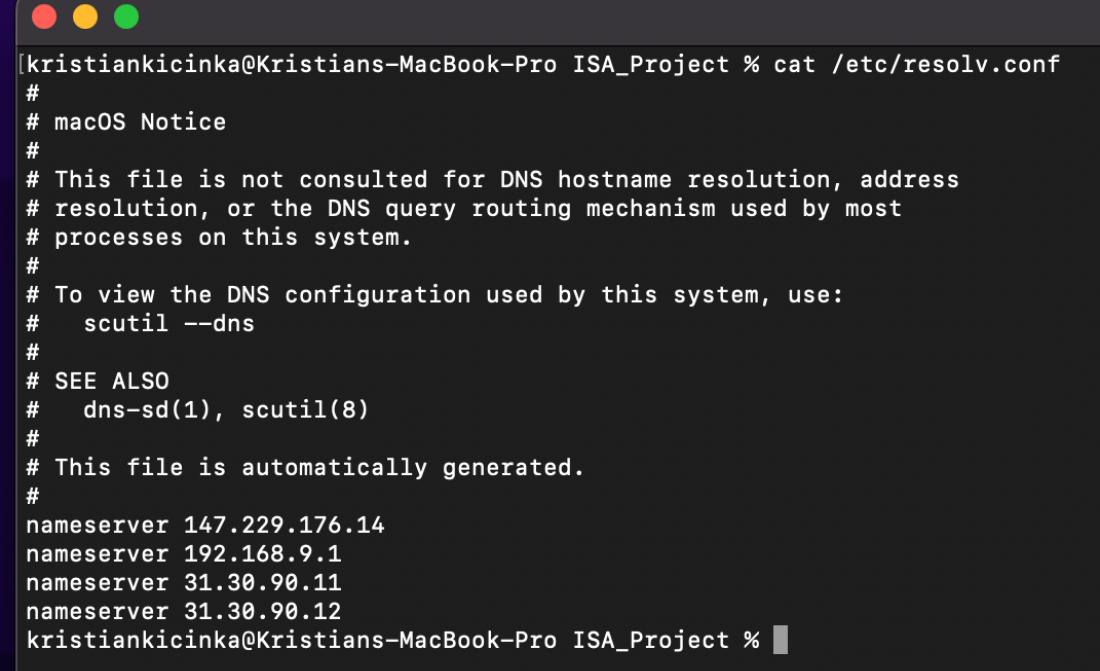
```
[kristiankicinka@Kristians-MacBook-Pro ISA_Project % cat tests/test_input_01
This is the test File.
kristiankicinka@Kristians-MacBook-Pro ISA_Project % ]
```

Obrázok 15 (Zobrazenie vstupného súboru)

Rovnako ako v predošom teste, sa dátá úspešne preniesli, takže môžeme tvrdiť, že komunikácia mimo lokálnu sieť je funkčná.

Test 03

V tomto teste overujeme, či je funkčná možnosť posielania dát na implicitný DNS server, bez nutnosti zadávania IP adresy. Najprv je nutné upraviť súbor `/etc/resolv.conf` a pridať tam adresu DNS server, v našom prípade ide o IP adresu serveru Eva. Ako base host sme zvolili reťazec „example.com“ Vstupný ale aj výstupný súbor je zhodný so súborom v predchádzajúcim teste.



```
[kristiankicinka@Kristians-MacBook-Pro ISA_Project % cat /etc/resolv.conf
#
# macOS Notice
#
# This file is not consulted for DNS hostname resolution, address
# resolution, or the DNS query routing mechanism used by most
# processes on this system.
#
# To view the DNS configuration used by this system, use:
#   scutil --dns
#
# SEE ALSO
#   dns-sd(1), scutil(8)
#
# This file is automatically generated.
#
nameserver 147.229.176.14
nameserver 192.168.9.1
nameserver 31.30.90.11
nameserver 31.30.90.12
kristiankicinka@Kristians-MacBook-Pro ISA_Project % ]
```

Obrázok 16 (Výsledný `/etc/resolv.conf` súbor)

```
[eva ~/3BIT/Projects/ISA_Project> ./dns_receiver example.com ./tests
#####
DNS Receiver started #####
#####
BASE_HOST : example.com
DST_FILEPATH : ./tests
#####
[INIT] 88.146.232.18
[PARS] ./tests/test_out_eva_01 'KRUGS4ZANFZSA5DIMUQHIZLTOQQEM2LMMUXAU.example.com'
[RECV] ./tests/test_out_eva_01          0 23B from 88.146.232.18
[INFO] Data was written to file !!
[CMPL] ./tests/test_out_eva_01 of 23B
```

Obrázok 17 (Výstup DNS receivera)

```
[kristiankicinka@Kristians-MacBook-Pro ISA_Project % ./dns_sender example.com test_out_eva_01 < ./tests/test_input_01
#####
DNS Sender started #####
#####
[INIT] 147.229.176.14
[INFO] Init packet was sent !!
[ENCD] test_out_eva_01          0 'KRUGS4ZANFZSA5DIMUQHIZLTOQQEM2LMMUXAU.example.com'
[SENT] test_out_eva_01          0 23B to 147.229.176.14
[INFO] Data packet was sent !!
[INFO] End packet was sent !!
[CMPL] test_out_eva_01 of 23B
kristiankicinka@Kristians-MacBook-Pro ISA_Project % ]
```

Obrázok 18 (Výstup DNS sendera)

No.	Time	Source	Destination	Protocol	Length	Info
158	14.259686	192.168.9.134	147.229.176.14	DNS	96	Standard query 0x05a4 Unused ORSXG5C7N52XIX3FOZOV6MBR.example.com
160	14.288428	192.168.9.134	147.229.176.14	DNS	109	Standard query 0x05a5 Unused KRUGS4ZANFZSA5DIMUQHIZLTOQQEM2LMMUXAU.example.com
167	14.304867	192.168.9.134	147.229.176.14	DNS	88	Standard query 0x05a6 Unused IVHEIX2Q1FBWRKU.example.com

> Frame 160: 109 bytes on wire (872 bits), 109 bytes captured (872 bits) on interface en0, id 0
> Ethernet II, Src: Apple_8f:0f:0a (a0:78:17:8f:0f:0a), Dst: Routerbo_70:7b:28 (48:8f:5a:70:7b:28)
> Internet Protocol Version 4, Src: 192.168.9.134, Dst: 147.229.176.14
> User Datagram Protocol, Src Port: 61410, Dst Port: 8181
> Domain Name System (query)
> Transaction ID: 0x05a5
> Flags: 0x010d Standard query
> Questions: 1
> Answer RRs: 0
> Authority RRs: 0
> Additional RRs: 0
> Queries
> KRUGS4ZANFZSA5DIMUQHIZLTOQQEM2LMMUXAU.example.com: type Unused, class Unknown
Name: KRUGS4ZANFZSA5DIMUQHIZLTOQQEM2LMMUXAU.example.com
[Name Length: 49]
[Label Count: 3]
Type: Unused (0)
Class: Unknown (0x0100)

Obrázok 19 (Výstup aplikácie Wireshark)

V aplikácii Wireshark ale aj vo výstupe DNS receivera môžeme pozorovať odoslané a prijaté dátové pakety na zariadenie s korektnou IP adresou. Týmto sme overili správnosť výberu implicitného DNS servera.

Test 04

V tomto teste overíme možnosť odoslania iných súborov ako textových. Pokúsime sa odoslať obrázok v .png formáte. Ako base host je zvolený reťazec „example.com“. Výsledný súbor bude uložený v `./tests/test_img_eva.png`.

```
eva ~/3BIT/Projects/ISA_Project> ./dns_receiver example.com ./tests
#####
DNS Receiver started #####
BASE_HOST : example.com
DST_FILEPATH : ./tests
[INT] 88.146.232.18
[PARS] ./tests/test_img_eva_01 'RFIE4RNBINAUAAAAGUSCCEKIAAMYAAABOIIAYAAAAXQUXEAAAET7USRCBKR4NV3GRWEQUAUAIXTPBCKYIBGY07FGCXTVBCJK0QDPW2BDSJZZC00ABQAAAAAAAAAAAAAAA
[PARS] ./tests/test_img_eva_01 0 998 from 88.146.232.18
[INFO] Data was written to file !!
[PARS] ./tests/test_img_eva_01 'AAAAAAAAAAAAAAAAAAAAAAAAC7XA2V17JPFPRS22Kw27GN3NT5V3RM17G3Z23B550IH2K5M1J4S3PWEBGUFKFCSCIVB0CFD2X56NAMPAP6YAHVAQXSGVTRIIGI.example.com'
[RECV] ./tests/test_img_eva_01 1 998 from 88.146.232.18
[INFO] Data was written to file !!
[PARS] ./tests/test_img_eva_01 'YTY7UQ150UMR2Z2UDQX2GGCURJRJGNLJJ5IDNEALBFSHFWSCRFAVNKNWLM523VTNG3FSL0G3M42JX3717473V3U6X3W65SHNGQXU6M7TJS30P7GP7PPH7XYP7ZN5T35JLV35IU1VAFBBEMJVA.example.com'
[RECV] ./tests/test_img_eva_01 2 998 from 88.146.232.18
[INFO] Data was written to file !!
[PARS] ./tests/test_img_eva_01 'UEERRCROMAEJQORGLEBINIQDCNBI1JDCEYDIAKDCMYBIAKDK4MWA54IW45LNYSVTTTJ077JSZ3B55JM7ZK4PHHL0G7HRMwKV5BWDR7EVAUGEJQ0TSQZTZND536Q.example.com'
```

Obrázok 20 (Začiatok výstupu receivera)

```
#####
DNS Sender started #####
[INFO] Data was written to file !!
[PARS] ./tests/test_img_eva.png 'N4SWAATNOV7VSHL7RXSRNG0JPQJJVZVRVYEPZIATPYPOA6NUMGKYFNW3HT5VZ73MPM30YLE2K4WA43HFS6ZAQTMDD3AYJUYF3E4LVZ44DPPC4RA6FDV34DHFP6C5W4E7TYGDYH22N7T3LA5MK0W4OY7DCQBY.example.com'
[RECV] ./tests/test_img_eva.png 92 998 from 88.146.232.18
[INFO] Data was written to file !!
[PARS] ./tests/test_img_eva.png 'NXGPKH4MSDVE3R65DDMC4KEG0D1ER0QAKHGLFQCY5LWLMMW0INJK4EJ35HQNDV0EK6TIXWZN63T3CXUHHP5AMF45Z7GCVKUP0R47T6037VC3C3X5YL00JX3S5W12XTFLZFR0WGELET13UXI02N73UHGB6ANNY.example.com'
[RECV] ./tests/test_img_eva.png 93 998 from 88.146.232.18
[INFO] Data was written to file !!
[PARS] ./tests/test_img_eva.png 'LDVY5EAONWFKRX45N4V4RRLUPX3YYQRGBMNEWIPD8VK2PS6TIZVJFM4TWWQXSHWHTV7LE5SLTS7PMNRMX2P6DLEDAAQMGS24VBJSGLWAFDAEEMKBJSABDCFQIAUMQVMACBQKIAUMQK.example.com'
[RECV] ./tests/test_img_eva.png 94 998 from 88.146.232.18
[INFO] Data was written to file !!
[PARS] ./tests/test_img_eva.png 'CVQCIAMARZKCRCQIAUMKWAJCQFDEEKKI0QJASRFCCFEK1ET7W6DY0BEAAAAAAT6X73IKDMAAAAAAAA
[PARS] ./tests/test_img_eva.png 'UEERRCROMAEJQORGLEBINIQDCNBI1JDCEYDIAKDCMYBIAKDK4MWA54IW45LNYSVTTTJ077JSZ3B55JM7ZK4PHHL0G7HRMwKV5BWDR7EVAUGEJQ0TSQZTZND536Q.example.com'
[RECV] ./tests/test_img_eva.png 95 998 from 88.146.232.18
[INFO] Data was written to file !!
[PARS] ./tests/test_img_eva.png 'AADACRXMYB0IRVLTCQAAAABEUKTSEVZBGBAQ.example.com'
[RECV] ./tests/test_img_eva.png 96 248 from 88.146.232.18
[INFO] Data was written to file !!
[CML] ./tests/test_img_eva.png of 9528B
```

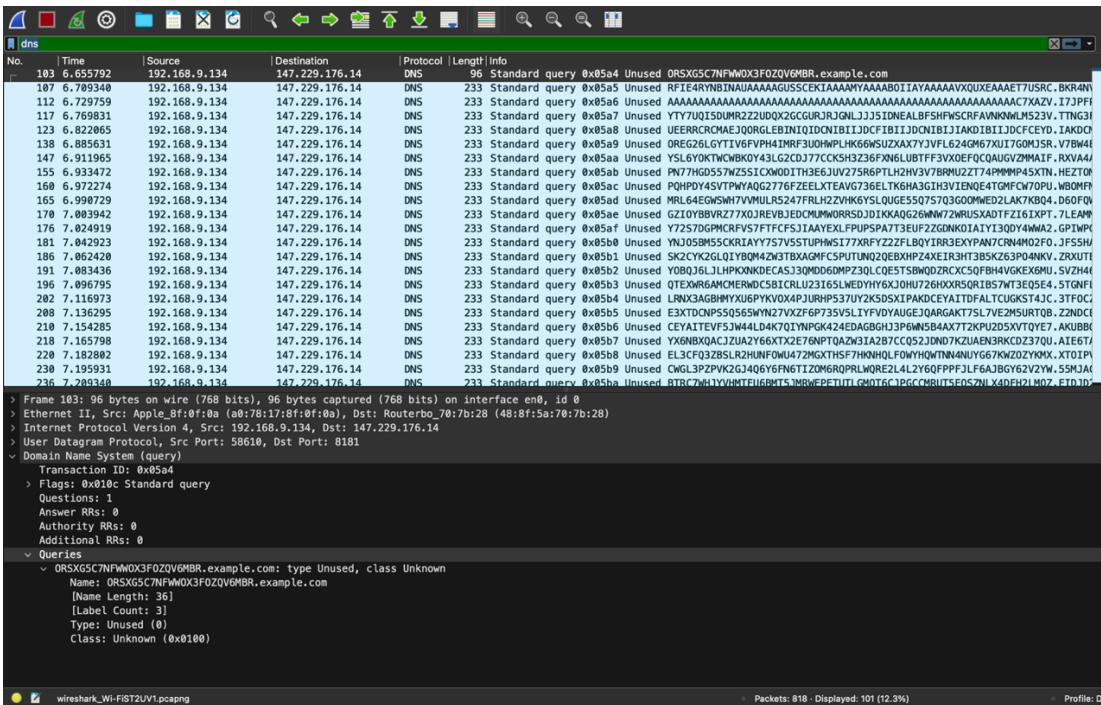
Obrázok 21 (Koniec výstupu receivera)

```
#####
DNS Sender started #####
[INT] 147.229.176.14
[INFO] Init packet was sent !!
[ENDC] test_img_eva.png 0 'RFIE4RNBINAUAAAAGUSCCEKIAAMYAAABOIIAYAAAAXQUXEAAAET7USRCBKR4NV3GRWEQUAUAIXTPBCKYIBGY07FGCXTVBCJK0QDPW2BDSJZZC00ABQAAAAAAAAAAAAAAA
[SEN] test_img_eva.png 0 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENDC] test_img_eva.png 1 'AAAAAAAAAAAAAAAAAAAAAAAAC7XA2V17JPFPRS22Kw27GN3NT5V3RM17G3Z23B550IH2K5M1J4S3PWEBGUFKFCSCIVB0CFD2X56NAMPAP6YAHVAQXSGVTRIIGI.example.com'
[SEN] test_img_eva.png 1 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENDC] test_img_eva.png 2 'YTY7UQ150UMR2Z2UDQX2GGCURJRJGNLJJ5IDNEALBFSHFWSCRFAVNKNWLM523VTNG3FSL0G3M42JX3717473V3U6X3W65SHNGQXU6M7TJS30P7GP7PPH7XYP7ZN5T35JLV35IU1VAFBBEMJVA.example.com'
[SEN] test_img_eva.png 2 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENDC] test_img_eva.png 3 'UEERRCROMAEJQORGLEBINIQDCNBI1JDCEYDIAKDCMYBIAKDK4MWA54IW45LNYSVTTTJ077JSZ3B55JM7ZK4PHHL0G7HRMwKV5BWDR7EVAUGEJQ0TSQZTZND536Q.example.com'
[SEN] test_img_eva.png 3 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENDC] test_img_eva.png 4 '0RE026LGYTIV6FVPH4IMRF3UOHPLHK66WSUZAX7YJFL624GM67XUI7GDMJSRV7BW4ENN2BH7U30345K3M2H2NSH4VXXOR5YY647LVS7PTUB374QPUGB0B2DMNKD05IPLNYSQ27LT33BVPSS56H33GKAPGWY.example.com'
```

Obrázok 22 (Začiatok výstupu sendera)

```
[SEN] test_img_eva.png 91 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENDC] test_img_eva.png 92 'N4SWAATNOV7VSHL7RXSRNG0JPQJJVZVRVYEPZIATPYPOA6NUMGKYFNW3HT5VZ73MPM30YLE2K4WA43HFS6ZAQTMDD3AYJUYF3E4LVZ44DPPC4RA6FDV34DHFP6C5W4E7TYGDYH22N7T3LA5MK0W4OY7DCQBY.example.com'
[SEN] test_img_eva.png 92 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENDC] test_img_eva.png 93 'NXGPKH4MSDVE3R65DDMC4KEG0D1ER0QAKHGLFQCY5LWLMMW0INJK4EJ35HQNDV0EK6TIXWZN63T3CXUHHP5AMF45Z7GCVKUP0R47T6037VC3C3X5YL00JX3S5W12XTFLZFR0WGELET13UXI02N73UHGB6ANNY.example.com'
[SEN] test_img_eva.png 93 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENDC] test_img_eva.png 94 'LDVY5EAONWFKRX45N4V4RRLUPX3YYQRGBMNEWIPD8VK2PS6TIZVJFM4TWWQXSHWHTV7LE5SLTS7PMNRMX2P6DLEDAAQMGS24VBJSGLWAFDAEEMKBJSABDCFQIAUMQVMACBQKIAUMQK.example.com'
[SEN] test_img_eva.png 94 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENDC] test_img_eva.png 95 'CVQCIAMARZKCRCQIAUMKWAJCQFDEEKKI0QJASRFCCFEK1ET7W6DY0BEAAAAAAT6X73IKDMAAAAAAAA
[SEN] test_img_eva.png 95 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENDC] test_img_eva.png 96 'AADACRXMYB0IRVLTCQAAAABEUKTSEVZBGBAQ.example.com'
[SEN] test_img_eva.png 96 248 to 147.229.176.14
[INFO] Data packet was sent !!
[ENDC] End_packet was sent !!
[CML] test_img_eva.png of 9528B
kristiankicinka@Kristians-MacBook-Pro ISA_Project %
```

Obrázok 23 (Koniec výstupu sendera)



Obrázok 24 (Výstup aplikácie Wireshark)

```

> Frame 170: 233 bytes on wire (1864 bits), 233 bytes captured (1864 bits) on interface en0, id 0
> Ethernet II, Src: Apple_8f:0f:0a (a0:78:17:8f:0f:0a), Dst: Routerbo_70:7b:28 (48:8f:5a:70:7b:28)
> Internet Protocol Version 4, Src: 192.168.9.134, Dst: 147.229.176.14
> User Datagram Protocol, Src Port: 53779, Dst Port: 8181
> Domain Name System (query)
    Transaction ID: 0x05ae
    Flags: 0x010d Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
    Queries
        <--> GZI0YBBVRZ77X0JREVBJEDCMUMWORRSJDJKKAQG26WNW72WRUSXADTFZI6IXPT.7LEAMMXAQACR535XYWPBXSACQ3VBQGTTN
            Name: GZI0YBBVRZ77X0JREVBJEDCMUMWORRSJDJKKAQG26WNW72WRUSXADTFZI6IXPT.7LEAMMXAQACR535XYWPBXSACQ
            [Name Length: 173]
            [Label Count: 5]
            Type: Unused (0)
            Class: Unknown (0x0100)

0000 48 8f 5a 70 7b 28 a0 78 17 8f 0f 0a 08 00 45 00 H·Zp{(.x .....·E·
0010 00 db 2e 1f 00 00 40 11 3d d1 c0 a8 09 86 93 e5 ····.·@ = ····.·
0020 b0 0e d2 13 1f f5 00 c7 51 69 05 ae 01 0d 00 01 ····.·Qi···.·
0030 00 00 00 00 00 00 3f 47 5a 49 4f 59 42 42 56 52 ····.·?G ZIOYBBVR
0040 5a 37 37 58 4f 4a 52 45 56 42 4a 45 44 43 4d 55 Z77XOJRE VBJEDCMU
0050 4d 57 4f 52 52 53 44 4a 44 49 4b 4b 41 51 47 32 MWORRSJD JKKAQG2
0060 36 57 4e 57 37 32 57 52 55 53 58 41 44 54 46 5a 6WNW72WR USXADTFZ
0070 49 36 49 58 50 54 3f 37 4c 45 41 4d 4d 58 41 51 I6IXPT77 LEAMMXAQ
0080 41 43 52 35 33 35 58 59 57 50 42 58 53 41 43 51 ACR535XY WPBXSACQ
0090 33 56 42 51 47 54 54 4e 52 42 52 54 36 36 33 52 3VBQGTTN RBRT663R
00a0 44 54 59 54 47 51 57 54 32 48 47 44 48 4a 36 45 DTYTGWt 2HGDHJ6E
00b0 32 34 4f 51 51 57 21 48 48 42 45 4d 49 56 41 46 240QQW!H HBEMIVAF
00c0 42 56 4d 42 54 42 46 37 50 48 56 4a 35 46 57 44 BVMBTBF7 PHVJ5FWD
00d0 37 55 4f 34 44 57 46 41 07 65 78 61 6d 70 6c 65 7U04DWFA ·example
00e0 03 63 6f 6d 00 00 00 01 00 ·com···.·
```

Obrázok 25 (Detailné zobrazenie DNS paketu)

Pri porovnaní výstupov DNS receivera a sendera vidíme, že veľkosti vstupného a výstupného súboru sa zhodujú. Dáta boli preposlané úspešne.

Test 05

V tomto teste overujeme možnosť odoslania väčšieho textového súboru. Vstupný súbor je naplnený generovaným textom. Súbor obsahuje 5 odstavcov generovaného textu. Ako base host je zvolený opäť reťazec „example.com“. Výsledný výstupný súbor je uložený v `./tests/test_out_eva_02`.

```
ISA_Project — ssh xkicin02@eva.fit.vutbr.cz — 197x22
[eva ~/3BIT/Projects/ISA_Project> ./dns_receiver example.com ./tests
#####
DNS Receiver started #####
BASE_HOST : example.com
DST_FILEPATH : ./tests

[INT] 88.146.232.18
[PARS] ./tests/test_out_eva_02 'JRXZEZLNEBUXA43VNUQGI33MNSZCA43J0QQC3LF00WCAY3PNZZWYK3UMV2HKZLSEB0W12LQNFZWG2L0M40K3DJO0XCAUDSMFSXGZL000QGIYLQNFRHK4Z0EBEW45DFM5SXEIDJNYQHGYLQNFSW4LRABJIGQYI.example.com'
[RECV] ./tests/test_out_eva_02 0 998 from 88.146.232.18
[INFO] Data was written to file !!
[PARS] ./tests/test_out_eva_02 'NSW3DVOQMGYLVNNUWE5LTBWM63DFON2GSZJANZXG3B0EBCHK2LTERBH3DWNFXGC40EBHHK3TDEB5G4DJM2XGIDUN5ZH133SEB3GK3BANVUSAZDB0BUE5LTBZW63DMNFRWS5DVMRUW4LQKJZ2WY3A.example.com'
[RECV] ./tests/test_out_eva_02 1 998 from 88.146.232.18
[INFO] Data was written to file !!
[PARS] ./tests/test_out_eva_02 'MEQH15LS0BUXGIDNMFYW4JYMEBRXK4TT0VZSA43J0QQC3LF00WCAY3VNRS4DJO0QGCLBANFXH1ZLSMR2W1DJM0WCAZTFNRUXGLRABJCHK2LTBRW63TENFWK3TU0VWSAYLVM52W1DJM0QG2YLHNZQS4Y.example.com'
[RECV] ./tests/test_out_eva_02 2 998 from 88.146.232.18
[INFO] Data was written to file !!
[PARS] ./tests/test_out_eva_02 'MWhXA2LSEBZHK5D0VWS54IC00VWGYLNEBXQIIDB0JRXKIDBEBSXGS5BAONXWY3DJMNUX15LENFXCAZLNFZ23SEFYFEC2L0MVW4IDMWVCA3LBONZC1DR0VUXGIDNMF2XE2LTB3GK2DJM2WYJANRQW21.example.com'
```

Obrázok 26 (Začiatok výstupu DNS receivera)

```
ISA_Project — ssh xkicin02@eva.fit.vutbr.cz — 197x22
[INFO] Data was written to file !!
[PARS] ./tests/test_out_eva_02 'EBSG54ZAOBQX5DV0JUWK3TUEBW63TUMVZSY1DOMFWZGZLUOVZCA4TJMRUW5LMOVZSA3LVMXACCSJNYQG3300ZQWY3DJO0XCAUDSMFSXGZL000QGIYLQNFRHK4Z0EBEW45DFM5SXEIDJNYQHGYLQNFSW4LRABJIGQYI.example.com'
[RECV] ./tests/test_out_eva_02 27 998 from 88.146.232.18
[INFO] Data was written to file !!
[PARS] ./tests/test_out_eva_02 'OUQGY3DN5ZH12LTB2XILBAMRUVG5DNUQGCSBAMR2SLRAJZ2WY3D0EBYH3DWNFXGC4RAMVWKG2L0MVWGXIDTMW54IAKIFSW4ZLBNYQHMLMEBWC43TMEQHCK5LJ0MQG2YLV0JUXGIDWVUGSY3VNRQS4A.example.com'
[RECV] ./tests/test_out_eva_02 28 998 from 88.146.232.18
[INFO] Data was written to file !!
[PARS] ./tests/test_out_eva_02 'FMRWS3TJMECATLBMVRW3TBMQVZ66LRAJZ2WY3D0EBYXK2LTBSGSYLNFYQES3RANXW45TBNRNG5Z0EBJWKZBAMVWGS5BAMR2WSLBA0B5WY3DNFZ2GK43ROVSSAYMEAFGMYLVMNUWE5LTB3GK3A.example.com'
[RECV] ./tests/test_out_eva_02 29 998 from 88.146.232.18
[INFO] Data was written to file !!
[PARS] ./tests/test_out_eva_02 'FQ0GS3TUMVZG15LNEBRGXKY2MEBSGSYLNFYQEM5LTNSAZDNEQGYZLPFQ0GS3LQMVZG12LF00QGS3RMEB0WY2LROVW21DTNF2CAYLNMV2CYIDGMV2W02LB0Q0GK5JMEBXXEY3JFYQAUSL0ORSW0ZLSBYGK3A.example.com'
[RECV] ./tests/test_out_eva_02 30 998 from 88.146.232.18
[INFO] Data was written to file !!
[PARS] ./tests/test_out_eva_02 'NSW45DF0NYXKZJA0Z5WYIDMWVGS5B0EBEW4IDEMFYGSYTVMQG5LHOVSSA3TPNYQHGYLQNFSW4LQK.example.com'
[RECV] ./tests/test_out_eva_02 31 558 from 88.146.232.18
[INFO] Data was written to file !!
[CMP] ./tests/test_out_eva_02 of 31248
```

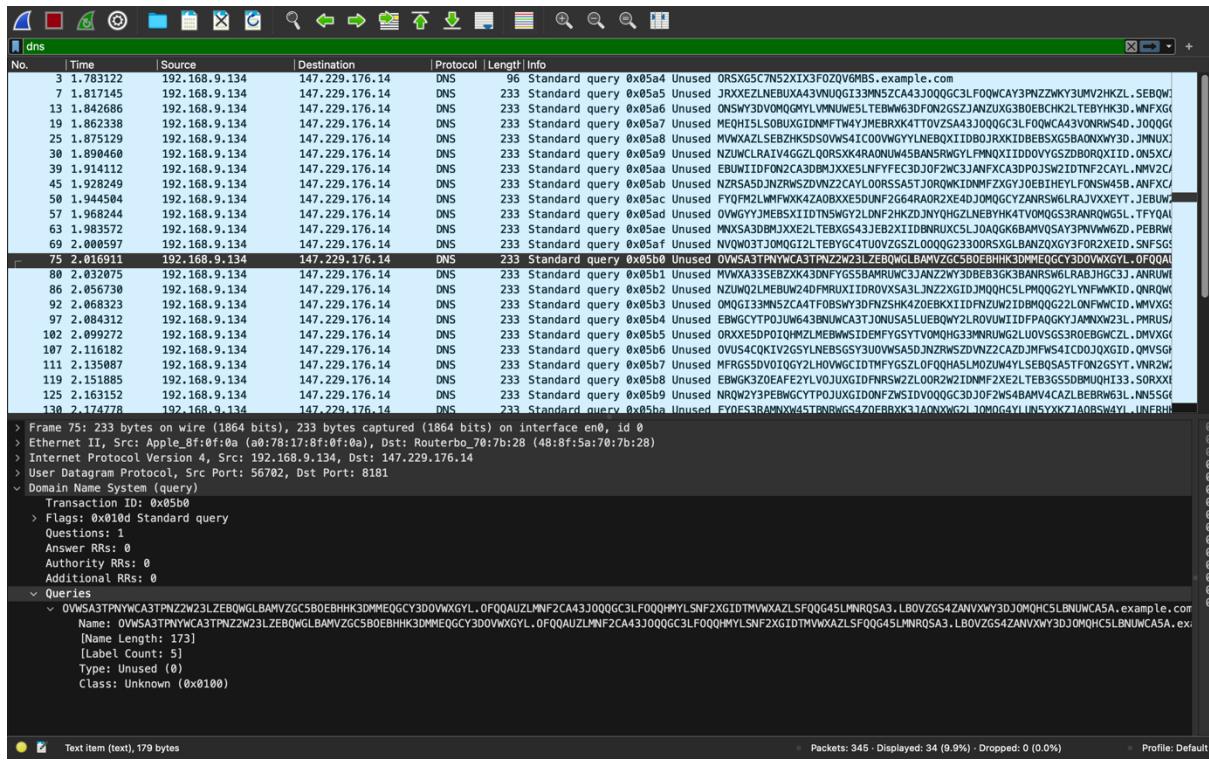
Obrázok 27 (Koniec výstupu DNS receivera)

```
ISA_Project — zsh — 199x4
[kristiankicinka@Kristians-MacBook-Pro ISA_Project % ./dns_sender example.com test_out_eva_02 < ./tests/test_input_02
#####
DNS Sender started #####
[INT] 147.229.176.14
[INFO] Init packet was sent !!
[ENCD] test_out_eva_02 0 'JRXZEZLNEBUXA43VNUQGI33MNSZCA43J0QQC3LF00WCAY3PNZZWYK3UMV2HKZLSEB0W12LQNFZWG2L0M40K3DJO0XCAUDSMFSXGZL000QGIYLQNFRHK4Z0EBEW45DFM5SXEIDJNYQHGYLQNFSW4LRABJIGQYI.example.com'
[SEN] test_out_eva_02 0 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_out_eva_02 1 'NSW3DVOQMGYLVNNUWE5LTBWM63DFON2GSZJANZXG3B0EBCHK2LTERBH3DWNFXGC40EBHHK3TDEB5G4DJM2XGIDUN5ZH133SEB3GK3BANVUSAZDB0BUE5LTBZW63DMNFRWS5DVMRUW4LQKJZ2WY3A.example.com'
[SEN] test_out_eva_02 1 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_out_eva_02 2 'MEQH15LS0BUXGIDNMFYW4JYMEBRXK4TT0VZSA43J0QQC3LF00WCAY3VNRS4DJO0QGCLBANFXH1ZLSMR2W1DJM0WCAZTFNRUXGLRABJCHK2LTBRW63TENFWK3TU0VWSAYLVM52W1DJM0QG2YLHNZQS4Y.example.com'
[SEN] test_out_eva_02 2 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_out_eva_02 3 'MWhXA2LSEBZHK5D0VWS54IC00VWGYLNEBXQIIDB0JRXKIDBEBSXGS5BAONXWY3DJMNUX15LENFXCAZLNFZ23SEFYFEC2L0MVW4IDMWVCA3LBONZC1DR0VUXGIDNMF2XE2LTB3GK2DJM2WYJANRQW21.example.com'
[SEN] test_out_eva_02 3 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_out_eva_02 4 'NZUWCLR1AV40GZLQR0SXK4RAONUW45BAN5RWGYLFMNQXIIDDOVYGSZD0RQXI0D0N5CA4DSN5UWIZL00QWAU43VNZ2CA2LOEBRXK3DQMEEQHC6LJEBXWmZTJMNW0C1DEMVFZWK4TVNZ2CA3LPNRWGS5BAMFVG3I.example.com'
```

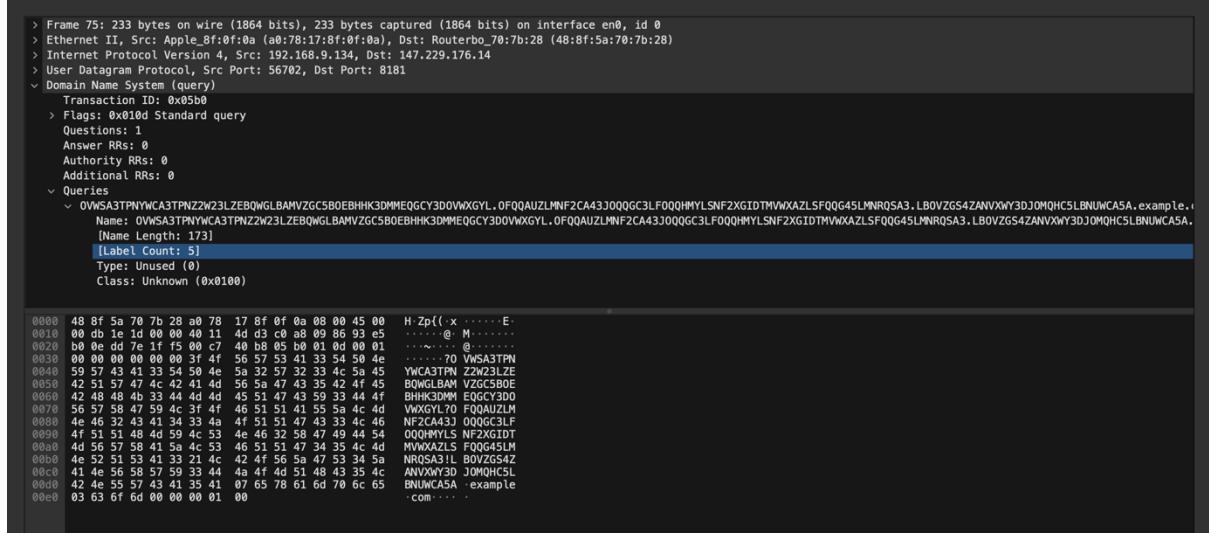
Obrázok 28 (Začiatok výstupu DNS sendera)

```
ISA_Project — zsh — 199x24
[SEN] test_out_eva_02 26 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_out_eva_02 27 'EBSG54ZAOBQX5DV0JUWK3TUEBW63TUMVZSY1DOMFWZGZLUOVZCA4TJMRUW5LMOVZSA3LVMXACCSJNYQG3300ZQWY3DJO0XCAUDSMFSXGZL000QGIYLQNFRHK4Z0EBEW45DFM5SXEIDJNYQHGYLQNFSW4LRABJIGQYI.example.com'
[SEN] test_out_eva_02 27 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_out_eva_02 28 'OUQGY3DN5ZH12LTB2XILBAMRUVG5DNUQGCSBAMR2SLRAJZ2WY3D0EBYH3DWNFXGC4RAMVWKG2L0MVWGXIDTMW54IAKIFSW4ZLBNYQHMLMEBWC43TMEQHCK5LJ0MQG2YLV0JUXGIDWVUGSY3VNRQS4A.example.com'
[SEN] test_out_eva_02 28 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_out_eva_02 29 'FMRWS3TJMECATLBMVRW3TBMQVZ66LRAJZ2WY3D0EBYXK2LTBSGSYLNFYQES3RANXW45TBNRNG5Z0EBJWKZBAMVWGS5BAMR2WSLBA0B5WY3DNFZ2GK43ROVSSAYMEAFGMYLVMNUWE5LTB3GK3A.example.com'
[SEN] test_out_eva_02 29 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_out_eva_02 30 'FQ0GS3TUMVZG15LNEBRGXKY2MEBSGSYLNFYQEM5LTNSAZDNEQGYZLPFQ0GS3LQMVZG12LF00QGS3RMEB0WY2LROVW21DTNF2CAYLNMV2CYIDGMV2W02LB0Q0GK5JMEBXXEY3JFYQAUSL0ORSW0ZLSBYGK3A.example.com'
[SEN] test_out_eva_02 30 998 to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_out_eva_02 31 'NSW45DF0NYXKZJA0Z5WYIDMWVGS5B0EBEW4IDEMFYGSYTVMQG5LHOVSSA3TPNYQHGYLQNFSW4LQK.example.com'
[SEN] test_out_eva_02 31 558 to 147.229.176.14
[INFO] Data packet was sent !!
[CMP] test_out_eva_02 of 31248
kristiankicinka@Kristians-MacBook-Pro ISA_Project %
```

Obrázok 29 (Koniec výstupu DNS sendera)



Obrázok 30 (Výstup aplikácie Wireshark)



Obrázok 31 (Podrobné zobrazenie DNS paketu)

Ako v predošom prípade pri porovnaní veľkostí vstupného súboru v DNS senderi a výstupného súboru pri DNS receiveri, zistíme, že veľkosti sú zhodné. Dáta boli korektne prenesené.

Meranie

V tejto časti vykonáme meranie času potrebného na prenos súborov z modulu DNS sendera na DNS receiver. Pri meraní využijeme príkaz „time“, ktorý napišeme pred spúšťací príkaz sendera. Využili sme nastavenia z testu 04 ale aj 05 a odmerali čas potrebný na prenos testovacích súborov.

```
[INFO] Data packet was sent !!
[ENCD] test_img_eva.png      92 'N4SWAATNOV7V5HLX7RXSRNG0JPQJJVZRVYEPZIATPYPQA6NUMGKYFNW3HT5VZ73MPM3OYLE2K4WA43HF
ple.com'
[SENT] test_img_eva.png      92 99B to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_img_eva.png      93 'NXGPKM4MSDVE3R65DDMC4KEGDICRQAKGHLFQCY5LWLMW0INJK4EJJ5HQNDNVOEK6TIXWZN63T3CXUHH
ple.com'
[SENT] test_img_eva.png      93 99B to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_img_eva.png      94 'LDVY5EAONWFKRX45N4V4RRULPX3YYQRGBMYNEWIPDBVKK2PS6TIZVJFM4TWUQXSHWHTV7LE5SLTS7PMNRM
ple.com'
[SENT] test_img_eva.png      94 99B to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_img_eva.png      95 'CVQCIAUMAARZKCRQCIAMKWAJCSQFDCEEKIUQJASRFCFEKIET7W6DYOBAAAAAAAT6X73IKDMAAAAAAAA
ple.com'
[SENT] test_img_eva.png      95 99B to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_img_eva.png      96 'AAADACRXXMYBOIRVLTCQAAAABEUKTSEVZBGBAQ.example.com'
[SENT] test_img_eva.png      96 24B to 147.229.176.14
[INFO] Data packet was sent !!
[INFO] End packet was sent !!
[CMPL] test_img_eva.png of 9528
./dns_sender example.com test_img_eva.png < ./tests/test_input_img.png 0.00s user 0.03s system 1% cpu 1.488 total
kristiankicinka@Kristians-MacBook-Pro ISA_Project %
```

Obrázok 32 (Výstup merania pre test 04)

```
[INFO] Data packet was sent !!
[ENCD] test_out_eva_02      27 'EBSGS4ZAOBXE5DV0JUWK3TUEBW63TUMVZSY1DOMFZWGZLUOVZCA4TJMRLW5LM0VZSA3LVOVMXACCSJNYQG03OOZQWY3DJOMXCATLBMRWIK3TB0MQGS4DT0VWSA5TFNRUXILBAMNXW43FMN2GK5DVMVZCAZI.example.com'
[SENT] test_out_eva_02      27 99B to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_out_eva_02      28 '0UGGY33CN5ZHI2LTB2XILBAMRUW5GDVNUGQCSBAMR2WSLRAJZ2WY3DBEBYHK3DWNFXG4RAMVWKG2LGMVXGIIDTMVWS4IAKIFSW4ZLBNNQHMZLMEBWc43TMEQHCSLJDMQ22YLVOJUXGIDwMVUGSY3VRQS3A.example.com'
[SENT] test_out_eva_02      28 99B to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_out_eva_02      29 'MFRWS3TJMEXCATLBMRWIK3TB0MQGSY2LCMVZG61RAJZ2WY3DBEBYXK2LTEBSGSYLNFYQES3RAMNXW45TBNRWS4ZOEBJWKZBAMWGS5BAMR2WSLBA0BSWY3DFNZ2GK43ROVSSAYJMEAFGMYLVMNUWE5LTEB3GK3A.example.com'
[SENT] test_out_eva_02      29 99B to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_out_eva_02      30 'FQGGS3TJM/ZG15LNExBKGYZMBGSYLNFYQEM5LTMNSSAZDVNNEQQYZLPFQQGS3LQMVZG12LF0QQGS3RMWBQWY2LROVQW2IDTNF2CAYLNMV2CY1DGMV2W02L80QQGK5JMEBXEY3JFYQAUSSL0ORSWOZLSEBYGK3A.example.com'
[SENT] test_out_eva_02      30 99B to 147.229.176.14
[INFO] Data packet was sent !!
[ENCD] test_out_eva_02      31 'NRW45DFONYKZJAOFWNC3JA02SWYIDWMVWGS5BOBEW4IDEMFYGSYTVOQM0C5LHOVSSASTPNYQHGYLQNFWS4LQK.example.com'
[SENT] test_out_eva_02      31 65B to 147.229.176.14
[INFO] Data packet was sent !!
[INFO] End packet was sent !!
[CMPL] test_out_eva_02 of 3124B
./dns_sender example.com test_out_eva_02 < ./tests/test_input_02 0.00s user 0.01s system 2% cpu 0.487 total
kristiankicinka@Kristians-MacBook-Pro ISA_Project %
```

Obrázok 33 (Výstup merania pre test 05)

Zdroje

- [1] <https://www.root.cz/clanky/tunelujeme-provoz-pomoci-dns-cesta-ven-ze-site/>
- [2] <https://mislove.org/teaching/cs4700/spring11/handouts/project1-primer.pdf>
- [3] https://cs.wikipedia.org/wiki/Domain_Name_System