Министерство науки и высшего образования Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

“НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО”

Факультет ПИиКТ

ОТЧЁТ

По лабораторной работе №4

«Анализ трафика компьютерных сетей с помощью утилиты Wireshark»

По предмету: Компьютерные сети

Студент:

Степанов М.А.

Группа P33301

Преподаватель:

Алиев Т. И.

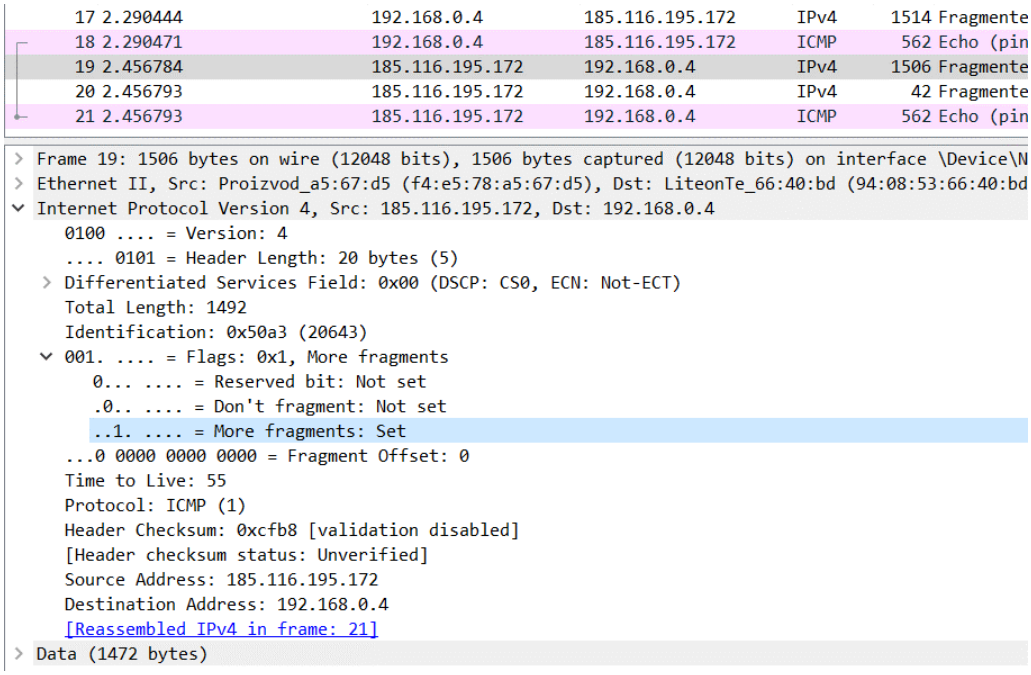
Санкт – Петербург

2023

Анализ трафика утилиты *ping*.

Вариант: smart-sts.kz

Пакеты делятся на фрагменты размером по 1480 байт максимум, а на признак последнего фрагмента указывает флаг *More fragments* в заголовке IPv4.



Количество фрагментов при передаче ping-пакета равно (N div 1480 + 1), где N – количество передаваемых байт.

TTL пакетов можно задать при помощи флага *-i* утилиты *ping*. В поле данных ping-пакета содержится последовательность ASCII символов (повторяется английский алфавит).

Анализ трафика утилиты *tracert.*

В заголовке IP содержится 20 байт, в поле данных – 64 байта.

```
Internet Protocol Version 4, Src: 185.116.195.172, Dst: 192.168.0.4
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
```

```
Internet Control Message Protocol
    Type: 0 (Echo (ping) reply)
    Code: 0
    Checksum: 0xff87 [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence Number (BE): 119 (0x0077)
    Sequence Number (LE): 30464 (0x7700)
    [Request frame: 882]
    [Response time: 150,906 ms]
  > Data (64 bytes)
```

Утилита последовательно отправляет по 3 пакета, каждый раз увеличивая TTL на 1, по истечение которого, пакет возвращается и дает информацию о последнем узле на его пути. Таким образом выстраивается маршрут до пункта назначения.

369	15.485007	192.168.0.4	185.116.195.172	ICMP	106	Echo (ping) request id=0x0001, seq=83/21248, ttl=1 (no response found!)
370	15.496083	192.168.0.1	192.168.0.4	ICMP	134	Time-to-live exceeded (Time to live exceeded in transit)
371	15.498140	192.168.0.4	185.116.195.172	ICMP	106	Echo (ping) request id=0x0001, seq=84/21504, ttl=1 (no response found!)
372	15.506244	192.168.0.1	192.168.0.4	ICMP	134	Time-to-live exceeded (Time to live exceeded in transit)
373	15.508431	192.168.0.4	185.116.195.172	ICMP	106	Echo (ping) request id=0x0001, seq=85/21760, ttl=1 (no response found!)
374	15.510285	192.168.0.1	192.168.0.4	ICMP	134	Time-to-live exceeded (Time to live exceeded in transit)
379	16.522863	192.168.0.4	185.116.195.172	ICMP	106	Echo (ping) request id=0x0001, seq=86/22016, ttl=2 (no response found!)
380	16.535990	178.69.32.1	192.168.0.4	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
381	16.537750	192.168.0.4	185.116.195.172	ICMP	106	Echo (ping) request id=0x0001, seq=87/22272, ttl=2 (no response found!)
382	16.543931	178.69.32.1	192.168.0.4	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
383	16.545669	192.168.0.4	185.116.195.172	ICMP	106	Echo (ping) request id=0x0001, seq=88/22528, ttl=2 (no response found!)
384	16.682396	178.69.32.1	192.168.0.4	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
385	17.562818	192.168.0.4	185.116.195.172	ICMP	106	Echo (ping) request id=0x0001, seq=89/22784, ttl=3 (no response found!)
386	17.574860	212.48.204.158	192.168.0.4	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
387	17.577049	192.168.0.4	185.116.195.172	ICMP	106	Echo (ping) request id=0x0001, seq=90/23040, ttl=3 (no response found!)
388	17.589149	212.48.204.158	192.168.0.4	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
389	17.590814	192.168.0.4	185.116.195.172	ICMP	106	Echo (ping) request id=0x0001, seq=91/23296, ttl=3 (no response found!)
390	17.596678	212.48.204.158	192.168.0.4	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)

Генерируемые ICMP-пакеты отличаются от генерируемых утилитой ping размером и содержимым данных, а также установленным TTL. Полученные пакеты были двух типов: «ошибочные» (черные) и «корректные» (розовые), и оба этих типа необходимы – черные информируют о промежуточных узлах маршрута, а розовые – о финальном. Если убрать ключ *-d*, будет генерироваться дополнительный трафик в виде доменных имен узлов:

```
Трассировка маршрута к smart-sts.kz [185.116.195.172]
с максимальным числом прыжков 30:

  1     8 ms     1 ms     2 ms  RT.HomeLAN [192.168.0.1]
  2    13 ms     5 ms     7 ms  ip.178-69-32-1.avangarddsl.ru [178.69.32.1]
  3    14 ms    13 ms     5 ms  bbn.212-48-204-158.nwtelecom.ru [212.48.204.158]
  4     6 ms     6 ms     7 ms  185.140.148.31
  5     *        *        *     Превышен интервал ожидания для запроса.
  6    69 ms   100 ms   100 ms  188.254.19.166
  7     *        *        *     Превышен интервал ожидания для запроса.
  8    86 ms   100 ms   100 ms  141.101.186.18
  9     *        *      132 ms  comp131-26.2day.kz [85.29.131.26]
 10     *        *        *     Превышен интервал ожидания для запроса.
 11   124 ms   100 ms   100 ms  89.38.164.98
 12     *        *        *     Превышен интервал ожидания для запроса.
 13   134 ms   100 ms   100 ms  pkz37.hoster.kz [185.116.195.172]
```

Анализ HTTP-трафика

Пара пакетов «запрос-ответ»:

30	5.897996	192.168.0.4	185.116.195.172	HTTP	548	GET /ru/ HTTP/1.1
40	5.998675	185.116.195.172	192.168.0.4	HTTP	988	HTTP/1.1 200 OK (text/html)

Как видно, код ответа – 200, значит полученные данные корректны. Содержимое поля данных ответа:

```
Line-based text data: text/html (355 lines)
    <!DOCTYPE html>\n
    <html class="nojs html css_verticalspacer" lang="ru-RU">\n
     <head>\n
    \n
      <meta http-equiv="Content-type" content="text/html;charset=UTF-8"/>\n
      <meta name="generator" content="2018.0.0.379"/>\n
      \n
      <script type="text/javascript">\n
       // Update the 'nojs'/'js' class on the html node\n
    document.documentElement.className = document.documentElement.className.replace(/\bnojs\b/g,
    \n
    // Check that all required assets are uploaded and up-to-date\n
     [truncated]if(typeof Muse == "undefined") window.Muse = {}; window.Muse.assets = {"required"
    </script>\n
      \n
      <link rel="shortcut icon" href="images/smart-training-solution-favicon.ico?crc=4048226455"/
      <title>Smart Training Solution</title>\n
      <!-- CSS -->\n
      <link rel="stylesheet" type="text/css" href="css/site_global.css?crc=261265647"/>\n
      <link rel="stylesheet" type="text/css" href="css/index.css?crc=4156362879" id="pagesheet"/>
      <!-- IE-only CSS -->\n
      <!--[if lt IE 9]>\n
      <link rel="stylesheet" type="text/css" href="css/iefonts_index.css?crc=236314972"/>\n
      <![endif]-->\n
      <!-- Other scripts -->\n
```

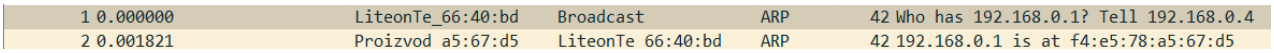
Пара пакетов после перезагрузки страницы:

28	9.764979	192.168.0.4	185.116.195.172	HTTP	620	GET /ru/ HTTP/1.1
32	9.846673	185.116.195.172	192.168.0.4	HTTP	211	HTTP/1.1 304 Not Modified

Код ответа – 304, значит данные на странице не изменились и поле данных отсутствует вовсе.

Анализ ARP-трафика

При первом (относительно очистки arp таблиц) открытии сайты был отправлен ARP-запрос на ближайший маршрутизатор.



1	0.000000	LiteonTe_66:40:bd	Broadcast	ARP	42	Who has 192.168.0.1? Tell 192.168.0.4
2	0.001821	Proizvod a5:67:d5	LiteonTe 66:40:bd	ARP	42	192.168.0.1 is at f4:e5:78:a5:67:d5

Можно заметить, что мы отправляем данный запрос не к конкретному серверу, на котором находится наш сайт, а к ближайшему маршрутизатору (широковещательный запрос, broadcast), поскольку нам не требуется отследить местоположение и существование в целом данного компьютера, достаточно лишь информации о том, что ближайший маршрутизатор знает, куда дальше передавать запрос.

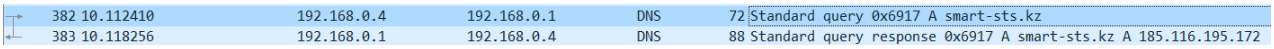
Некоторые запросы содержат mac адреса маршрутизатора и компьютера, с которого происходят запросы. Эти адреса используются для маршрутизации внутри локальной беспроводной сети для составления ARP-таблиц.

Позже данные MAC-адреса используются и в HTTP запросах на сетевом уровне для передачи данных между компьютером и маршрутизатором.

IP адрес же в запросах необходим для нахождения конфликтов в сети и заполнения ARP-таблиц.

Анализ DNS-трафика

При первом (относительно очистки кэша) открытии сайта был послан DNS-запрос (id 732340):



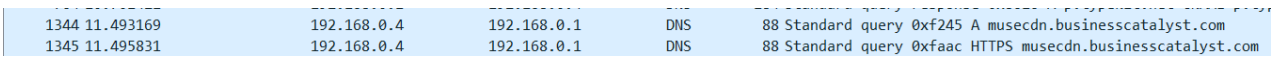
382	10.112410	192.168.0.4	192.168.0.1	DNS	72	Standard query 0x6917 A smart-sts.kz
383	10.118256	192.168.0.1	192.168.0.4	DNS	88	Standard query response 0x6917 A smart-sts.kz A 185.116.195.172

Как видно, адрес получателя отличается от фактического адреса сайта. Это происходит потому, что компьютер не знает этого адреса, он знает лишь доменное имя, поэтому он отправляет DNS запрос на ближайший DNS-сервер, который (через пару строк) вернул ему нужный IP-адрес.

Существует несколько типов DNS запросов:

- Прямой
- Обратный
- Рекурсивный
- Итеративный

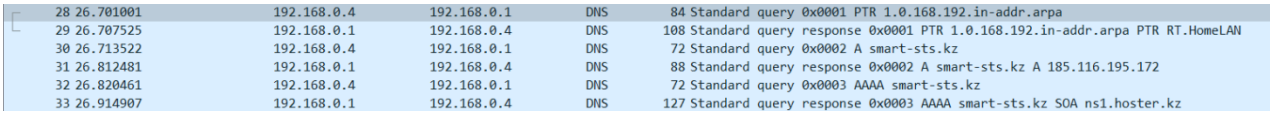
Также при обращении к сайту через браузер могут возникать дополнительные DNS запросы, в случае если некоторые медиа файлы находятся на другом хосте.



1344	11.493169	192.168.0.4	192.168.0.1	DNS	88	Standard query 0xf245 A musecdn.businesscatalyst.com
1345	11.495831	192.168.0.4	192.168.0.1	DNS	88	Standard query 0xfaac HTTPS musecdn.businesscatalyst.com

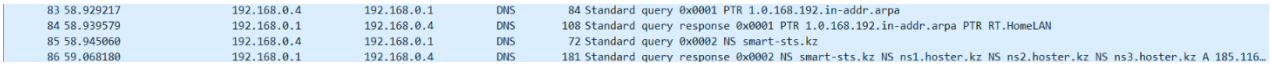
Анализ трафика утилиты *nslookup.*

Обычный DNS-запрос:



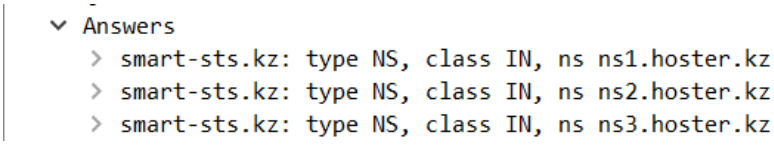
28	26.701001	192.168.0.4	192.168.0.1	DNS	84	Standard query 0x0001 PTR 1.0.168.192.in-addr.arpa
29	26.707525	192.168.0.1	192.168.0.4	DNS	108	Standard query response 0x0001 PTR 1.0.168.192.in-addr.arpa PTR RT.HomeLAN
30	26.713522	192.168.0.4	192.168.0.1	DNS	72	Standard query 0x0002 A smart-sts.kz
31	26.812481	192.168.0.1	192.168.0.4	DNS	88	Standard query response 0x0002 A smart-sts.kz A 185.116.195.172
32	26.820461	192.168.0.4	192.168.0.1	DNS	72	Standard query 0x0003 AAAA smart-sts.kz
33	26.914907	192.168.0.1	192.168.0.4	DNS	127	Standard query response 0x0003 AAAA smart-sts.kz SOA ns1.hoster.kz

NS DNS-запрос:



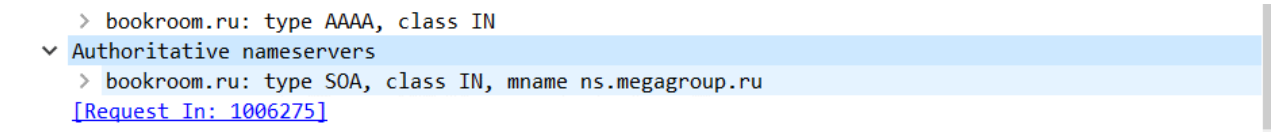
83	58.929217	192.168.0.4	192.168.0.1	DNS	84	Standard query 0x0001 PTR 1.0.168.192.in-addr.arpa
84	58.939579	192.168.0.1	192.168.0.4	DNS	108	Standard query response 0x0001 PTR 1.0.168.192.in-addr.arpa PTR RT.HomeLAN
85	58.945060	192.168.0.4	192.168.0.1	DNS	72	Standard query 0x0002 NS smart-sts.kz
86	59.068180	192.168.0.1	192.168.0.4	DNS	181	Standard query response 0x0002 NS smart-sts.kz NS ns1.hoster.kz NS ns2.hoster.kz NS ns3.hoster.kz A 185.116…

Отличия трафиков в том, что второй запрос содержит информацию не о IP-адресе нашего сайта, а об доменных серверах, которые обслуживают этот сайт (в поле Answers):



```
Answers
  smart-sts.kz: type NS, class IN, ns ns1.hoster.kz
  smart-sts.kz: type NS, class IN, ns ns2.hoster.kz
  smart-sts.kz: type NS, class IN, ns ns3.hoster.kz
```

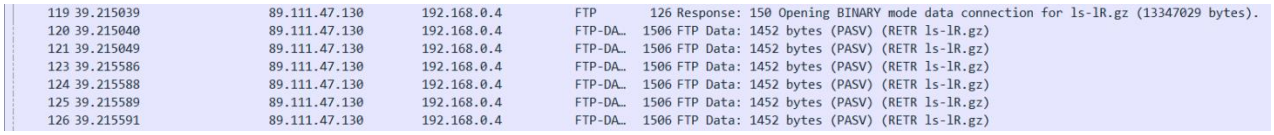
Имя сервера, возвратившего авторитативный отклик:



```
  bookroom.ru: type AAAA, class IN
Authoritative nameservers
  bookroom.ru: type SOA, class IN, mname ns.megagroup.ru
[Request In: 1006275]
```

Анализ FTP-трафика.

Данные фрагментируются на фрагменты по 1460 байт по «протоколу» FTP-DATA, который по факту просто является FTP, настроенным на порт 20:



119	39.215039	89.111.47.130	192.168.0.4	FTP	126	Response: 150 Opening BINARY mode data connection for ls-lR.gz (13347029 bytes).
120	39.215040	89.111.47.130	192.168.0.4	FTP-DA…	1506	FTP Data: 1452 bytes (PASV) (RETR ls-lR.gz)
121	39.215049	89.111.47.130	192.168.0.4	FTP-DA…	1506	FTP Data: 1452 bytes (PASV) (RETR ls-lR.gz)
123	39.215586	89.111.47.130	192.168.0.4	FTP-DA…	1506	FTP Data: 1452 bytes (PASV) (RETR ls-lR.gz)
124	39.215588	89.111.47.130	192.168.0.4	FTP-DA…	1506	FTP Data: 1452 bytes (PASV) (RETR ls-lR.gz)
125	39.215589	89.111.47.130	192.168.0.4	FTP-DA…	1506	FTP Data: 1452 bytes (PASV) (RETR ls-lR.gz)
126	39.215591	89.111.47.130	192.168.0.4	FTP-DA…	1506	FTP Data: 1452 bytes (PASV) (RETR ls-lR.gz)

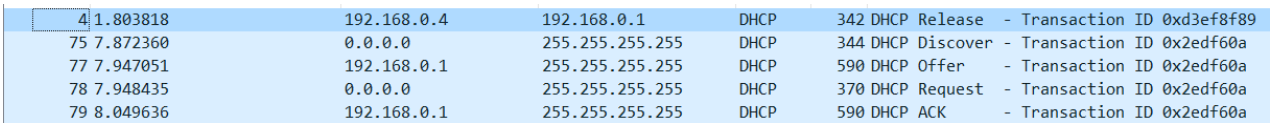
В общем же случае существует 2 порта для подключения по FTP:

«The well known TCP port for FTP control is 21 and for FTP data is 20»

порт 21 для управления (FTP) и порт 20 для передачи данных (FTP-DATA). В нашем случае для получения ответа используется динамический порт

Анализ DHCP-трафика.

Последовательность пакетов:



4	1.803818	192.168.0.4	192.168.0.1	DHCP	342	DHCP Release - Transaction ID 0xd3ef8f89
75	7.872360	0.0.0.0	255.255.255.255	DHCP	344	DHCP Discover - Transaction ID 0x2edf60a
77	7.947051	192.168.0.1	255.255.255.255	DHCP	590	DHCP Offer - Transaction ID 0x2edf60a
78	7.948435	0.0.0.0	255.255.255.255	DHCP	370	DHCP Request - Transaction ID 0x2edf60a
79	8.049636	192.168.0.1	255.255.255.255	DHCP	590	DHCP ACK - Transaction ID 0x2edf60a

Пакеты Discover и Request отличаются тем, что первый еще не знает о местоположении DHCP-сервера, и пытается его найти. Второй же лишь является ответом на Offer, соглашается на предложенный IP-адрес.

Адрес источника и назначения меняется в процессе передачи сообщений, так как некоторые запросы отсылаются широковещательно, а некоторые к конкретному DHCP серверу

Адрес DHCP-сервера: