

Министерство науки и высшего образования
Российской Федерации

Федеральное государственное бюджетное
образовательное учреждение высшего образования

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

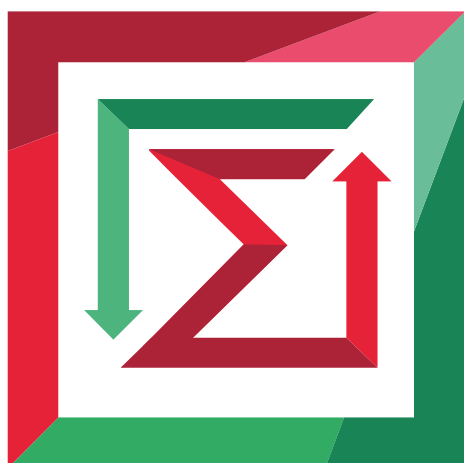


Теоретической и прикладной математики

Лабораторная работа № 1

по дисциплине «ОСНОВЫ WEB-ПРОГРАММИРОВАНИЯ»

HTTP-ЗАПРОСЫ



Факультет:	ПМИ
Группа:	ПМИ-02
Студент:	Сидоров Даниил
Преподаватель:	Цыгулин Алексей Александрович.

Новосибирск

2026

1. Цель работы

Познакомиться с протоколом HTTP, увидеть внутреннюю структуру обмена клиента и сервера.

2. Задание 1. Telnet

HTTP – текстовый протокол взаимодействия. Браузер работает аналогично терминалу связи. Мы используем сетевой протокол **telnet** для того, чтобы увидеть, что происходит при взаимодействии клиента и сервера. Для обмена данными между сервером и браузером воспользуемся сайтом международного института компьютерных технологий <http://www.iict.ru/>. В командной строке выполним команду **telnet iict.ru 80**:

```
C:\Users\danys>telnet iict.ru 80_
```

В открывшемся окне терминала набираем следующее:

GET / HTTP/1.1

Host: iict.ru

После отправки запроса получили ответ:

```
HTTP/1.1 200 OK
Date: Sat, 01 Oct 2022 06:53:29 GMT
Content-Type: text/html; charset=utf-8
Transfer-Encoding: chunked
Connection: keep-alive
Keep-Alive: timeout=10
Vary: Accept-Encoding
Server: Apache
Set-Cookie: 442579f83eab32ca792786558bb9435d=4ff81262266e1d4315c004860139e0bc; path=/; HttpOnly
Set-Cookie: iict_ja_blank_tpl=iict_ja_blank; expires=Thu, 21-Sep-2023 06:53:28 GMT; Max-Age=30672000; path=/
P3P: CP="NOI ADM DEV PSAi COM NAV OUR OTRo STP IND DEM"
Cache-Control: private, no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Expires: Wed, 17 Aug 2005 00:00:00 GMT
Last-Modified: Sat, 01 Oct 2022 06:53:29 GMT
Pragma: no-cache

8b0
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ru-ru" lang="ru-ru">
<head>
  <script type="text/javascript">
    var siteurl='/';
    var tplurl='/templates/iict_ja_blank/';
    var isRTL = false;
  </script>
  <base href="http://iict.ru/" />
  <meta http-equiv="content-type" content="text/html; charset=utf-8" />
  <meta name="generator" content=" Joomla! - Open Source Content Management" />
  <title>A A</title>
  <link href="/?format=feed&type=rss" rel="alternate" type="application/rss+xml" title="RSS 2.0" />
</head>
```

Разберем полученный ответ:

HTTP/1.1 200 OK - код состояния и пояснение к нему.

Date: Sat, 01 Oct 2022 06:53:29 GMT - дата создания ответа от сервера.

Content-Type: text/html; charset=utf-8 - указание типа текстовых данных и используемой кодировки.

Transfer-Encoding: chunked - механизм передачи данных в протоколе передачи гипертекста, надёжно доставляющий данные от сервера клиенту.

Connection: keep-alive - использование одного TCP-соединения для отправки и получения многократных HTTP-запросов и ответов.

Keep-Alive: timeout=10 - как долго сервер готов держать соединение открытым, прежде чем принудительно закроет соединение.

Vary: Accept-Encoding - информирует о поведении сервера в отношении кэширования представления запрошенного ресурса.

Server: Apache - используемый веб-сервер.

Set-Cookie: - заголовки, которые используются для отправки cookies с сервера.

P3P: CP="NOI ADM DEV PSAi COM NAV OUR OTRo STP IND DEM " – заголовок, позволяющий сайту информировать браузер о предполагаемом получении личных данных пользователя.

Cache-Control: private, no-store, no-cache, must-revalidate, post – check=0, pre – check=0 - заголовок используется для задания инструкций кэширования как для запросов, так и для ответов.

private - ответ сервера является специфическим для конечного пользователя и не должен кэшироваться различными промежуточными прокси

no-store - кэш не должен хранить никакую информацию о запросе и ответе.

no-cache - необходимость отправить запрос на сервер для валидации ресурса перед использованием закэшированных данных.

must-revalidate - кэш должен проверить статус устаревших ресурсов перед их использованием).

Post - check=0 - Определяет интервал времени в секундах, после которого ресурс должен быть проверен на актуальность.

Pre - check=0 - Определяет интервал времени в секундах, после которого проверка актуальности ресурса должна быть произведена перед его отображением для пользователя.

Expires: Wed, 17 Aug 2005 00:00:00 GMT - заголовок, который содержит дату/время, по истечении которой ответ сервера считается устаревшим.

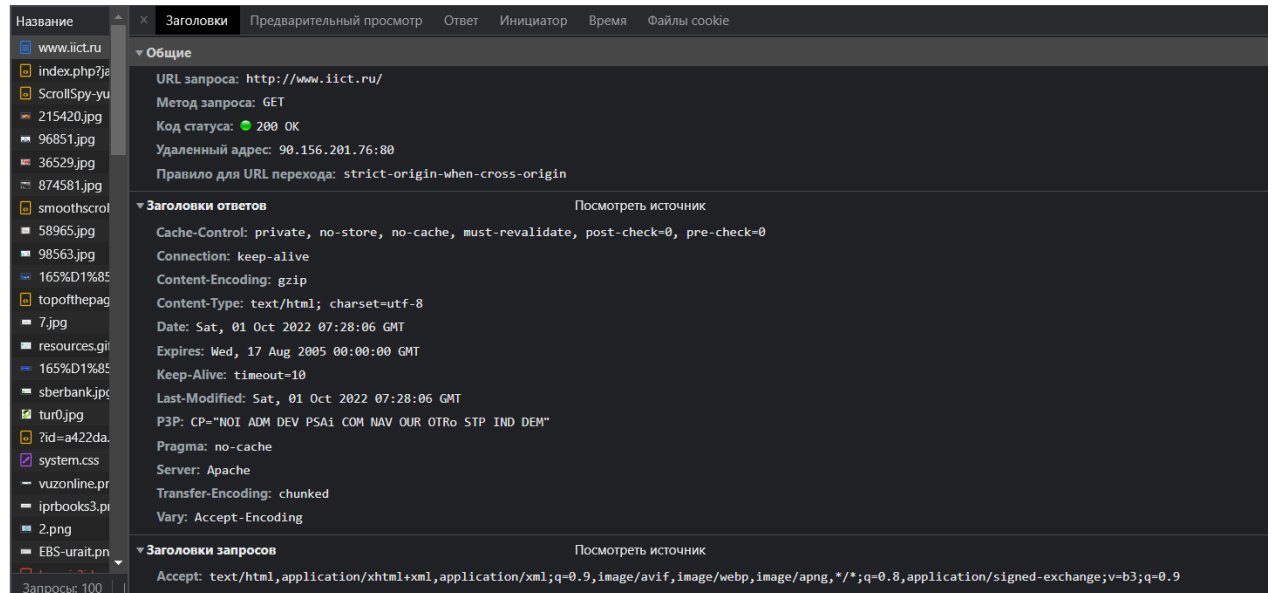
Pragma: no-cache - аналогично заголовку cache-control: no-cache.

Last-modified: Sat, 01 Oct 2022 06.53.29 GMT - дата последних изменений.

3. Задание 2. Браузер

Проделаем предыдущее задание с использованием браузера **Chrome**. Используем вкладку «Сеть» (Network) в средствах разработки для того, чтобы отследить передаваемый запрос и ответ на него.

Откроем сайт **iict.ru** и посмотрим заголовки ответов.



Все заголовки совпадают с ранее полученными. Так же появился новый заголовок **Content-Encoding: gzip** - согласование того, какой формат кодирования использовать для передачи тела.

Название	Статус	Тип	Инициатор	Размер	Время	Каскад загрузки
www.iict.ru	200	document	Другое	10.6 kB	363 мс	
index.php?j...&jat3action=gzip&jat3type=js&jat3file=t3-assets%...	200	script	(индекс)	(кеш памяти)	0 мс	
215420.jpg	200	jpeg	www.iict.ru/247	(кеш памяти)	0 мс	
96851.jpg	200	jpeg	www.iict.ru/311	(кеш памяти)	0 мс	
36529.jpg	200	jpeg	www.iict.ru/370	(кеш памяти)	0 мс	
874581.jpg	200	jpeg	www.iict.ru/434	(кеш памяти)	0 мс	
58965.jpg	200	jpeg	www.iict.ru/494	(кеш памяти)	0 мс	
98563.jpg	200	jpeg	www.iict.ru/554	(кеш памяти)	0 мс	
165%D1%8579-blue.jpg	200	jpeg	www.iict.ru/645	(кеш памяти)	0 мс	
7.jpg	200	jpeg	www.iict.ru/678	(кеш памяти)	0 мс	
resources.gif	200	gif	www.iict.ru/689	(кеш памяти)	0 мс	
165%D1%8552.jpg	200	jpeg	www.iict.ru/711	(кеш памяти)	0 мс	
sberbank.jpg	200	jpeg	www.iict.ru/722	(кеш памяти)	0 мс	
tur0.jpg	200	jpeg	www.iict.ru/766	(кеш памяти)	0 мс	
ScrollSpy-yui-compressed.js	307	script / Переадр...	(индекс)	350 B	43 мс	
smoothscroll.js	304	script	(индекс)	223 B	173 мс	
topofthepage.js	304	script	(индекс)	224 B	147 мс	
system.css	304	stylesheet	index.php?j...&jat3action=gzip&jat3...	223 B	168 мс	
?id=a422dad7dcf61cb45f9458778f61a8bb620e®=54&or...	200	script	ScrollSpy-yui-compressed.js	2.5 kB	97 мс	
vuzonline.png	304	png	(индекс)	225 B	55 мс	
iprbooks3.png	304	png	(индекс)	224 B	55 мс	
2.png	304	png	(индекс)	224 B	57 мс	
EBS-urait.png	304	png	(индекс)	224 B	60 мс	

В самом начале загрузился документ **www.iict.ru** за 363 мс, его размер составил 10.6 kB, статус документа – 200.

Далее имеем много запросов типа **script, jpeg/gif/png**, а также **stylesheet** – это запросы для загрузки файлов **JS, картинок и CSS** соответственно. Загрузка запросов типа **script** заняло от 0 до 173 мс, загрузка **картинок** от 0 до 60 мс, загрузка **таблицы стилей** 168 мс.

4. Задание 3. Wireshark

Для выполнения задания запустим программу **wireshark**, выберем активный интерфейс **Ethernet**, включим запись сетевого трафика, зайдем на сайт **iict.ru** и выключим запись.

Видим пакеты с протоколом **TCP**. Изначально идут пакеты, направленные к серверу, а после от сервера к браузеру. Время выполнения указано во 2 колонке (от 1 до 3 мс).

Пакет **TCP** — это блок данных, который передаётся в потоке. Порядок пакетов строго определён и не может быть нарушен. Задача протокола **TCP** - управлять отправкой данных и следить за тем, чтобы они были гарантированно приняты получателем. **TCP** разбивает данные на пакеты и нумерует их.

7 1.538135	192.168.0.5	90.156.201.98	TCP	66 54951 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
9 1.582069	90.156.201.98	192.168.0.5	TCP	66 80 → 54951 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1452 SACK_PERM=1 WS=512
10 1.582197	192.168.0.5	90.156.201.98	TCP	54 54951 → 80 [ACK] Seq=1 Ack=1 Win=262656 Len=0
14 1.629280	90.156.201.98	192.168.0.5	TCP	60 80 → 54951 [ACK] Seq=1 Ack=444 Win=30720 Len=0
15 1.985912	90.156.201.98	192.168.0.5	TCP	1506 80 → 54951 [ACK] Seq=1 Ack=444 Win=30720 Len=1452 [TCP segment of a reassembled PDU]
16 1.985912	90.156.201.98	192.168.0.5	TCP	1506 80 → 54951 [ACK] Seq=1453 Ack=444 Win=30720 Len=1452 [TCP segment of a reassembled PDU]
17 1.985977	192.168.0.5	90.156.201.98	TCP	54 54951 → 80 [ACK] Seq=444 Ack=2905 Win=262656 Len=0
18 1.986078	90.156.201.98	192.168.0.5	TCP	387 80 → 54951 [PSH, ACK] Seq=2905 Ack=444 Win=30720 Len=333 [TCP segment of a reassembled PDU]
19 1.986453	90.156.201.98	192.168.0.5	TCP	1245 80 → 54951 [PSH, ACK] Seq=3238 Ack=444 Win=30720 Len=1191 [TCP segment of a reassembled PDU]
20 1.986485	192.168.0.5	90.156.201.98	TCP	54 54951 → 80 [ACK] Seq=444 Ack=4429 Win=262656 Len=0
21 1.986671	90.156.201.98	192.168.0.5	TCP	1016 80 → 54951 [PSH, ACK] Seq=4429 Ack=444 Win=30720 Len=962 [TCP segment of a reassembled PDU]
22 1.987282	90.156.201.98	192.168.0.5	TCP	1506 80 → 54951 [ACK] Seq=5391 Ack=444 Win=30720 Len=1452 [TCP segment of a reassembled PDU]
23 1.987282	90.156.201.98	192.168.0.5	TCP	434 80 → 54951 [PSH, ACK] Seq=6843 Ack=444 Win=30720 Len=380 [TCP segment of a reassembled PDU]
24 1.987327	192.168.0.5	90.156.201.98	TCP	54 54951 → 80 [ACK] Seq=444 Ack=7223 Win=262656 Len=0
25 1.987642	90.156.201.98	192.168.0.5	TCP	1192 80 → 54951 [PSH, ACK] Seq=7223 Ack=444 Win=30720 Len=1138 [TCP segment of a reassembled PDU]
26 1.987671	192.168.0.5	90.156.201.98	TCP	54 54951 → 80 [ACK] Seq=444 Ack=8361 Win=261632 Len=0
27 1.988107	90.156.201.98	192.168.0.5	TCP	1365 80 → 54951 [PSH, ACK] Seq=8361 Ack=444 Win=30720 Len=1311 [TCP segment of a reassembled PDU]
28 1.988595	90.156.201.98	192.168.0.5	TCP	1123 80 → 54951 [PSH, ACK] Seq=9672 Ack=444 Win=30720 Len=1069 [TCP segment of a reassembled PDU]
29 1.988633	192.168.0.5	90.156.201.98	TCP	54 54951 → 80 [ACK] Seq=444 Ack=10741 Win=262656 Len=0
38 2.100962	90.156.201.98	192.168.0.5	TCP	60 80 → 54951 [ACK] Seq=10761 Ack=947 Win=31744 Len=0
53 2.141651	90.156.201.98	192.168.0.5	TCP	1506 80 → 54951 [ACK] Seq=10761 Ack=947 Win=31744 Len=1452 [TCP segment of a reassembled PDU]
54 2.141651	90.156.201.98	192.168.0.5	TCP	1498 80 → 54951 [PSH, ACK] Seq=12213 Ack=947 Win=31744 Len=1444 [TCP segment of a reassembled PDU]
55 2.141651	90.156.201.98	192.168.0.5	TCP	1502 80 → 54951 [PSH, ACK] Seq=13657 Ack=947 Win=31744 Len=1448 [TCP segment of a reassembled PDU]
56 2.141736	192.168.0.5	90.156.201.98	TCP	54 54951 → 80 [ACK] Seq=947 Ack=15105 Win=262656 Len=0
57 2.141842	90.156.201.98	192.168.0.5	TCP	1506 80 → 54951 [ACK] Seq=15105 Ack=947 Win=31744 Len=1452 [TCP segment of a reassembled PDU]
59 2.141881	192.168.0.5	90.156.201.98	TCP	54 54951 → 80 [ACK] Seq=947 Ack=17862 Win=262656 Len=0
472 2.799393	90.156.201.98	192.168.0.5	TCP	60 80 → 54951 [ACK] Seq=17862 Ack=1454 Win=32768 Len=0
525 2.806626	90.156.201.98	192.168.0.5	TCP	1506 80 → 54951 [ACK] Seq=19314 Ack=1454 Win=32768 Len=1452 [TCP segment of a reassembled PDU]
526 2.806912	90.156.201.98	192.168.0.5	TCP	1498 80 → 54951 [PSH, ACK] Seq=19314 Ack=1454 Win=32768 Len=1444 [TCP segment of a reassembled PDU]
527 2.806912	90.156.201.98	192.168.0.5	TCP	1506 80 → 54951 [ACK] Seq=20758 Ack=1454 Win=32768 Len=1452 [TCP segment of a reassembled PDU]
528 2.806912	90.156.201.98	192.168.0.5	TCP	1506 80 → 54951 [ACK] Seq=22210 Ack=1454 Win=32768 Len=1452 [TCP segment of a reassembled PDU]
529 2.806912	90.156.201.98	192.168.0.5	TCP	1506 80 → 54951 [ACK] Seq=23662 Ack=1454 Win=32768 Len=1452 [TCP segment of a reassembled PDU]
530 2.806912	90.156.201.98	192.168.0.5	TCP	1506 80 → 54951 [ACK] Seq=25114 Ack=1454 Win=32768 Len=1452 [TCP segment of a reassembled PDU]
531 2.806912	90.156.201.98	192.168.0.5	TCP	1506 80 → 54951 [ACK] Seq=26566 Ack=1454 Win=32768 Len=1452 [TCP segment of a reassembled PDU]
532 2.806912	90.156.201.98	192.168.0.5	TCP	1506 80 → 54951 [ACK] Seq=28018 Ack=1454 Win=32768 Len=1452 [TCP segment of a reassembled PDU]
533 2.806991	192.168.0.5	90.156.201.98	TCP	54 54951 → 80 [ACK] Seq=1454 Ack=29470 Win=262656 Len=0

Если последуем за **TCP** потоком, получим информацию, схожую с заданием 1 и 2.

```
GET / HTTP/1.1
Host: iict.ru
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/105.0.0.0 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7

HTTP/1.1 200 OK
Date: Sat, 01 Oct 2022 07:47:40 GMT
Content-Type: text/html; charset=utf-8
Transfer-Encoding: chunked
Connection: keep-alive
Keep-Alive: timeout=10
Vary: Accept-Encoding
Server: Apache
Set-Cookie: 442579f83eab32ca792786558bb9435d=073cd557243c372170c98f9d091a3dd4; path=/; HttpOnly
Set-Cookie: iict_ja_blank_tpl=iict_ja_blank; expires=Thu, 21-Sep-2023 07:47:40 GMT; Max-Age=30672000; path=/
P3P: CP="NOI ADM DEV PSAI COM NAV OUR OTRP STP IND DEM"
Cache-Control: private, no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Expires: Wed, 17 Aug 2005 00:00:00 GMT
Last-Modified: Sat, 01 Oct 2022 07:47:40 GMT
Pragma: no-cache
Content-Encoding: gzip

9eb
.....X.n...m...X.K...".v.&p..V..A..w...vw...dE@...Z.5P.GP.)
...n.8y%a.f={.R$e+...{
Ss9..w.s...l...o.....o lX...7...M...{.....H .d< .e.s.+...hd.&...m.@...+lh...H.w/H.W.....Dg.z..L.'E..H0...{6...
$.F.#...#.J$.C...Hk..I.....q.u...m...K.$..l.TX.rw.....P...v.m.r...J.G.EnD...J...FV.r..SI..2.oc6.....PP1.f.f..6.J.
*...[e1..A...
Hd$.J..p.{
2.{l
...G6E7.m...
...0..I&=M...<N...<KN..K'..g...7(y.....'.{.O..P..~..{.l.'X~..@9....x..._..7.#.|
Z@.J...s...g&...y...s...v$.F..x.F.x.g.%a.1.....o...V.Z...../s...:MXP.c.c.S1`s.....D.k].B.
9..Fc..pqapZJ...l.....'.....m...d.M.VSc.X...=0C7..Gd...3..B..Z..m|.>..d..^..S...o4.....@.C.
.R...\.\\..Fs)s<.4..n.../(<).O...Z...v...a..U+Vz..D<...po|.....
..s.D...qh...a0X.e.h.l.<.R.<M..@.<.I...lmM...#.V...5..[Z
f...[.....Z."J..j...QK..Op.f...m|.r.(...$L...{.})...uP.....a[D]S.).....m...:Yk8.....F...o.5.
.....
....8aG..5>.....3.u0...5s0.E.p..>u..Bo.....~...e..#.yw.C.0f..f..c..8J."6N..@|yrN..z...4:....|l....q... ..2:~
\\...V|b....."B.....W.....mt...{.9..8,..* ..G.F...=...Q...".oH&..d..d..G..v.Z.....t...GX.p=
$.S...VR...1R*..3.8m7...t.f...9.....F.>.0}
```

Далее видим пакеты с протоколом **HTTP**. Сначала произошел **http** запрос, далее загрузились запросы типа **png/jpg, css, js**. Время выполнения указано во 2 колонке (от 1 до 3.2 мс). После выполнения каждого запроса получаем ответ 200.

11	1.582581	192.168.0.5	90.156.201.98	HTTP	497	GET / HTTP/1.1
30	2.032590	90.156.201.98	192.168.0.5	HTTP	74	HTTP/1.1 200 OK (text/html)
31	2.057266	192.168.0.5	90.156.201.98	HTTP	557	GET /index.php?jst3action=gzip&jst3type=css&jst3file=t3-assets%2Fcss_f698d.css HTTP/1.1
58	2.141842	90.156.201.98	192.168.0.5	HTTP	1399	HTTP/1.1 200 OK (text/css)
469	2.754984	192.168.0.5	90.156.201.98	HTTP	561	GET /images/banners/antiterror/2.png HTTP/1.1
535	2.807080	90.156.201.98	192.168.0.5	HTTP	1266	HTTP/1.1 200 OK (PNG)
538	2.812149	192.168.0.5	90.156.201.98	HTTP	558	GET /images/banners/EBS-urait.png HTTP/1.1
668	2.862797	90.156.201.98	192.168.0.5	HTTP	1269	HTTP/1.1 200 OK (PNG)
682	2.897876	192.168.0.5	90.156.201.98	HTTP	560	GET /images/typo/16-19/Calender.png HTTP/1.1
765	2.952705	90.156.201.98	192.168.0.5	HTTP	798	HTTP/1.1 200 OK (PNG)
774	2.963329	192.168.0.5	90.156.201.98	HTTP	554	GET /images/typo/16-19/14.png HTTP/1.1
788	3.016196	90.156.201.98	192.168.0.5	HTTP	1051	HTTP/1.1 200 OK (PNG)
791	3.022741	192.168.0.5	90.156.201.98	HTTP	554	GET /images/typo/16-19/10.png HTTP/1.1
803	3.073187	90.156.201.98	192.168.0.5	HTTP	1133	HTTP/1.1 200 OK (PNG)
806	3.076638	192.168.0.5	90.156.201.98	HTTP	554	GET /images/typo/16-19/49.png HTTP/1.1
815	3.129497	90.156.201.98	192.168.0.5	HTTP	1140	HTTP/1.1 200 OK (PNG)
816	3.132300	192.168.0.5	90.156.201.98	HTTP	566	GET /images/typo/16-19/product-design.png HTTP/1.1
831	3.186268	90.156.201.98	192.168.0.5	HTTP	1077	HTTP/1.1 200 OK (PNG)

Если мы последуем за **HTTP** потоком, то увидим такие же результаты, как и в прошлых заданиях:

```
GET / HTTP/1.1
Host: iict.ru
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/105.0.0.0 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7

HTTP/1.1 200 OK
Date: Sat, 01 Oct 2022 07:47:40 GMT
Content-Type: text/html; charset=utf-8
Transfer-Encoding: chunked
Connection: keep-alive
Keep-Alive: timeout=10
Vary: Accept-Encoding
Server: Apache
Set-Cookie: 442579f83eab32ca792786558bb9435d=073cd557243c372170c98f9d091a3dd4; path=/; HttpOnly
Set-Cookie: iict_ja_blank_tpl=iict_ja_blank; expires=Thu, 21-Sep-2023 07:47:40 GMT; Max-Age=30672000; path=/
P3P: CP="NOI ADM DEV PSAI COM NAV OUR OTR STP IND DEM"
Cache-Control: private, no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Expires: Wed, 17 Aug 2005 00:00:00 GMT
Last-Modified: Sat, 01 Oct 2022 07:47:40 GMT
Pragma: no-cache
Content-Encoding: gzip

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ru-ru" lang="ru-ru">

<head>
  <script type="text/javascript">
var siteurl='/';
var tplurl='/templates/iict_ja_blank/';
var isRTL = false;
</script>
```

5. Вывод

В ходе выполнения заданий познакомились с протоколами HTTP и TCP с помощью **telnet**, панели разработчика браузера и **wireshark**.