

GET, POST 방식 패킷 분석

노트북: 숙제

만든 날짜: 2019-07-24 오전 12:06

업데이트: 2019-07-24 오전 12:06

작성자: 이종민

8	12.93444016	192.168.122.1	192.168.122.10	TCP	74	38634 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=3356870072 TSecr=0 WS=128
9	12.93467518	192.168.122.10	192.168.122.1	TCP	74	80 → 38634 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=16758374 TSecr=3356870072 WS=128
10	12.93470429	192.168.122.1	192.168.122.10	TCP	66	38634 → 80 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=3356870072 TSecr=16758374
11	12.93482850	192.168.122.1	192.168.122.10	HTTP	394	GET / HTTP/1.1
12	12.93498202	192.168.122.10	192.168.122.1	TCP	66	80 → 38634 [ACK] Seq=1 Ack=329 Win=30080 Len=0 TSval=16758375 TSecr=3356870072
13	12.93875825	192.168.122.10	192.168.122.1	HTTP	1101	HTTP/1.1 200 OK (text/html)
14	12.93880178	192.168.122.1	192.168.122.10	TCP	66	38634 → 80 [ACK] Seq=329 Ack=1036 Win=31360 Len=0 TSval=3356870076 TSecr=16758378
17	17.94410452	192.168.122.10	192.168.122.1	TCP	66	80 → 38634 [FIN, ACK] Seq=1036 Ack=329 Win=30080 Len=0 TSval=16763384 TSecr=3356870076
18	17.94426208	192.168.122.1	192.168.122.10	TCP	66	38634 → 80 [FIN, ACK] Seq=329 Ack=1037 Win=31360 Len=0 TSval=3356875082 TSecr=16763384
19	17.94447826	192.168.122.10	192.168.122.1	TCP	66	80 → 38634 [ACK] Seq=1037 Ack=330 Win=30080 Len=0 TSval=16763384 TSecr=3356875082

3 way hand shake 과정을 맺어서 HTTP 1.1 에 연결을 시도 해서 200 OK 정상적인 형태로 접속 이 완료가 된 것에 대한 응답을 받고 FIN, ACK 를 통해 연결을 종료하는 것을 볼 수 있다.

24	20.10069436	192.168.122.1	192.168.122.10	TCP	74	38636 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=3356877238 TSecr=0 WS=128
25	20.10099497	192.168.122.10	192.168.122.1	TCP	74	80 → 38636 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=16765541 TSecr=3356877238 WS=128
26	20.10101389	192.168.122.1	192.168.122.10	TCP	66	38636 → 80 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=3356877238 TSecr=16765541
27	20.10113896	192.168.122.1	192.168.122.10	HTTP	459	GET /get.php?id=123&pwd=4567&merong=lol HTTP/1.1
28	20.10129288	192.168.122.10	192.168.122.1	TCP	66	80 → 38636 [ACK] Seq=1 Ack=394 Win=30080 Len=0 TSval=16765541 TSecr=3356877239
29	20.10291550	192.168.122.10	192.168.122.1	HTTP	343	HTTP/1.1 200 OK (text/html)
30	20.10294318	192.168.122.1	192.168.122.10	TCP	66	38636 → 80 [ACK] Seq=394 Ack=278 Win=30336 Len=0 TSval=3356877240 TSecr=16765543
33	25.10805639	192.168.122.10	192.168.122.1	TCP	66	80 → 38636 [FIN, ACK] Seq=278 Ack=394 Win=30080 Len=0 TSval=16770548 TSecr=3356877240
34	25.10902622	192.168.122.1	192.168.122.10	TCP	66	38636 → 80 [FIN, ACK] Seq=394 Ack=279 Win=30336 Len=0 TSval=3356882246 TSecr=16770548
35	25.10949708	192.168.122.10	192.168.122.1	TCP	66	80 → 38636 [ACK] Seq=279 Ack=395 Win=30080 Len=0 TSval=16770549 TSecr=3356882246

GET 방식

3 way hand shake 맺어서 GET 방식으로 연결해서 정상적인 형태 200 OK, 파라미터 값 id, pwd, hidden에 대한 값이 보이는 걸 볼 수 있다.

해당 응답을 받으면 FIN, ACK 를 통해 세션을 종료한다.

39	30.06474549	192.168.122.1	192.168.122.10	TCP	74	38638 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=3356887202 TSecr=0 WS=128
40	30.06504043	192.168.122.10	192.168.122.1	TCP	74	80 → 38638 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=16775505 TSecr=3356887202 WS=128
41	30.06507903	192.168.122.1	192.168.122.10	TCP	66	38638 → 80 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=3356887202 TSecr=16775505
42	30.06521046	192.168.122.1	192.168.122.10	HTTP	517	POST /post.php HTTP/1.1 (application/x-www-form-urlencoded)
43	30.06534815	192.168.122.10	192.168.122.1	TCP	66	80 → 38638 [ACK] Seq=1 Ack=452 Win=30080 Len=0 TSval=16775505 TSecr=3356887203
44	30.06693215	192.168.122.10	192.168.122.1	HTTP	328	HTTP/1.1 200 OK (text/html)
45	30.06696772	192.168.122.1	192.168.122.10	TCP	66	38638 → 80 [ACK] Seq=452 Ack=263 Win=30336 Len=0 TSval=3356887204 TSecr=16775507
48	35.06958166	192.168.122.10	192.168.122.1	TCP	66	80 → 38638 [FIN, ACK] Seq=263 Ack=452 Win=30080 Len=0 TSval=16780509 TSecr=3356887204
49	35.06996622	192.168.122.1	192.168.122.10	TCP	66	38638 → 80 [FIN, ACK] Seq=452 Ack=264 Win=30336 Len=0 TSval=3356892207 TSecr=16780509
50	35.07047078	192.168.122.10	192.168.122.1	TCP	66	80 → 38638 [ACK] Seq=264 Ack=453 Win=30080 Len=0 TSval=16780510 TSecr=3356892207

POST 방식

3 way hand shake를 맺어서 POST 방식 접속 HTTP 1.1 200OK 정상적인 형태. POST 형식은 id, pwd, hidden에 대한 파라미터 값이 보이지 않는다. 해당 응답을 받으면 FIN, ACK를 통해 세션을 종료한다.

POST /post.php HTTP/1.1\r\n	[SEQ/ACK analysis]
Host: www.cccr.net\r\n	[Timestamps]
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0\r\n	TCP payload (392 bytes)
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n	[Hypertext Transfer Protocol]
Accept-Language: en-US,en;q=0.5\r\n	[GET /get.php?id=123&pwd=4567&merong=lol HTTP/1.1\r\n]
Accept-Encoding: gzip, deflate\r\n	Host: www.cccr.net\r\n
Content-Type: application/x-www-form-urlencoded\r\n	User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0\r\n
Content-Length: 34\r\n	Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n
DNT: 1\r\n	Accept-Language: en-US,en;q=0.5\r\n
Connection: keep-alive\r\n	Accept-Encoding: gzip, deflate\r\n
Referer: http://www.cccr.net/\r\n	DNT: 1\r\n
Upgrade-Insecure-Requests: 1\r\n	Connection: keep-alive\r\n
\r\n	Referer: http://www.cccr.net/\r\n
[Full request URI: http://www.cccr.net/post.php]	Upgrade-Insecure-Requests: 1\r\n
[HTTP request 1/1]	\r\n
[Response in 1000 - 40]	[Full request URI: http://www.cccr.net/get.php?id=123&pwd=4567&merong=lol]
File Data: 14 bytes	[HTTP request 1/1]
HTML Form URL Encoded: application/x-www-form-urlencoded	[Response in 1000 - 40]

GET방식과 POST 방식의 패킷을 보면

GET 방식 패킷은 URI, 피라미터 값이 Id, pw, merong의 값을 URI에 전부 다 포함하고 있는걸 볼 수 있다.

반면 POST 방식 패킷은 URI, 피라미터 값이 나타나지 않는 것을 볼 수 있다.

host : www.cccr.net 서버의 도메인 네임을 나타낸다.

user-agent : mozilla firefox 사용자가 어떤 클라이언트를 이용해 요청을 보냈는지 볼 수 있다.

accept : text/html, */* 요청 보낼 때 어떤 타입으로 보내면 좋겠다고 명시한다.

html 형식인 응답을 처리하겠다는 뜻이다. 콤마를 적어서 여러 타입을 동시에 적어줄 수 있는데 html 형식이 아닌 */* 모든 형태의 응답을 처리하겠다는 의미로 보인다.

accept-encoding, accept-language : 언어, 콘텐츠 압축 방식을 나타낸다.

dnt : 요청 헤더는 사용자의 트래킹 설정을 가르킨다. 0은 트래킹 허용한다, 1은 트래킹 허용을 하지 않는 것을 말하는데 현재 설정은 1 이므로 트래킹을 허용한다.

- dnt : 추적 중지, 사이트에서 추적하지 못하게 막는 방식 (firefox에서 기본적으로 적용된 옵션으로 보인다)

connection : 현재 전송이 완료된 후 네트워크 접속을 유지할지 말지를 제어한다.

- keep alive : tcp 연결을 재사용하는 기능이다. tcp를 무한정으로 유지할 수 없으므로 마지막으로 종료된 시점부터 정의된 시간까지 access가 없더라도 세션을 유지하는 구조이다.

referer : 이 페이지 이전의 페이지 주소가 담겨있다. 어떤 페이지에서 현재 페이지로 넘어왔는지 알 수 있다.

- 현재 www.cccr.net 에서 get 또는 post를 사용해서 다른 페이지로 넘어왔기 때문에 www.cccr.net 으로 보인다.