블로그에서 검색 유입을 판별하는 원리

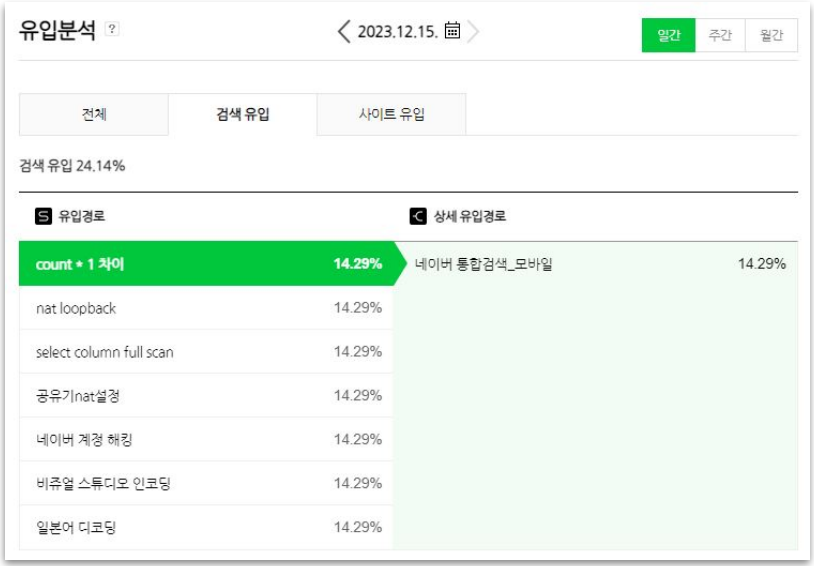
Yun @ Baekjoon Best Conference 2023 Winter

발표자 소개

티스토리 백엔드 4년차 개발자

- Kotlin & Spring Boot 로 일하는 중
- PHP를 혐오함
- 카카오 코딩테스트 문제 검수 담당
- 알고리즘 문제 해결이 취미
- 최근 갑자기 코드포스 오렌지가 되고 싶어서 환장함

블로그 하시는 분?


유입분석
2023.12.15.
일간
주간
월간
전체
검색 유입
사이트 유입
검색 유입 24.14%
유입경로
상세 유입경로
count * 1 차이
14.29%
네이버 통합검색_모바일
14.29%
nat loopback
14.29%
select column full scan
14.29%
공유기nat설정
14.29%
네이버 계정 해킹
14.29%
비쥬얼 스튜디오 인코딩
14.29%
일본어 디코딩
14.29%


2023.08
일간
주간
월간
월간 방문수
147
01
02
03
04
05
06
07
08
09
10
11
이번달
2023년
검색 60
네이버 검색 2
다음 검색 55
구글 검색 2
줌 검색 0
빙 검색 1
야후 검색 0
기타 검색 0
SNS 8
카카오톡 8
카카오스토리 0
페이스북 0
트위터 0
유튜브 0
인스타그램 0
기타 SNS 0
기타 79
직접 유입 17
기타 유입 62
기타 유입 자세히 보기
디바이스
모바일 84%
PC 16%
인기글
월간 조회수
작성일
1 원주율 1000자리 128 2022.11.21
2 루빅스 큐브 신기한 모델링 3 2022.11.22
3 파텍 필립 그랜드마스터 차임 Ref. 5175 제작 과정 영상 2 2022.11.22
4 프로그래밍에서 정수가 2의 N제곱꼴인지 판별하기 2 2022.12.07
5 정규식으로 소수 판별하기 1 2022.11.23
유입 키워드
원주율 1000자리 54
2의 제곱 특징 2
원주율 소수점이하자리 1

HTTP 리퍼러

16개 언어

문서 토론 읽기 편집 역사 보기 도구

위키백과, 우리 모두의 백과사전.

HTTP 리퍼러(HTTP referer)는 웹 브라우저로 월드 와이드 웹을 서핑할 때, 하이퍼링크를 통해서 각각의 사이트로 방문시 남는 흔적을 말한다.

예를 들어 A라는 웹 페이지에 B 사이트로 이동하는 하이퍼링크가 존재한다고 하자. 이때 웹 사이트 이용자가 이 하이퍼링크를 클릭하게 되면 웹 브라우저에서 B 사이트로 참조 주소(리퍼러)를 전송하게 된다. B 사이트의 관리자는 이 전송된 리퍼러를 보고 방문객이 A 사이트를 통해 자신의 사이트에 방문한 사실을 알 수 있다.

웹 사이트의 서버 관리자가 사이트 방문객이 어떤 경로로 자신의 사이트에 방문했는지 알아볼 때 유용하게 사용된다.

하지만, 리퍼러는 조작또한 가능하기 때문에 리퍼러 정보를 사용할 때에는 보안에 항상 주의해야 한다.

한편, HTTP 리퍼러를 정의한 RFC에서 'referrer'를 'referer'라고 잘못 친 것에서 기인하여 HTTP 리퍼러는 'HTTP referer'라고 불린다.[1]

각주 [편집]

1. ↑ "HTTP:The Definitive Guide".

HTTP

지속성 · 압축 · HTTPS

요청 방식

OPTIONS · GET · HEAD · POST · PUT · DELETE · TRACE · CONNECT · PATCH

헤더 필드

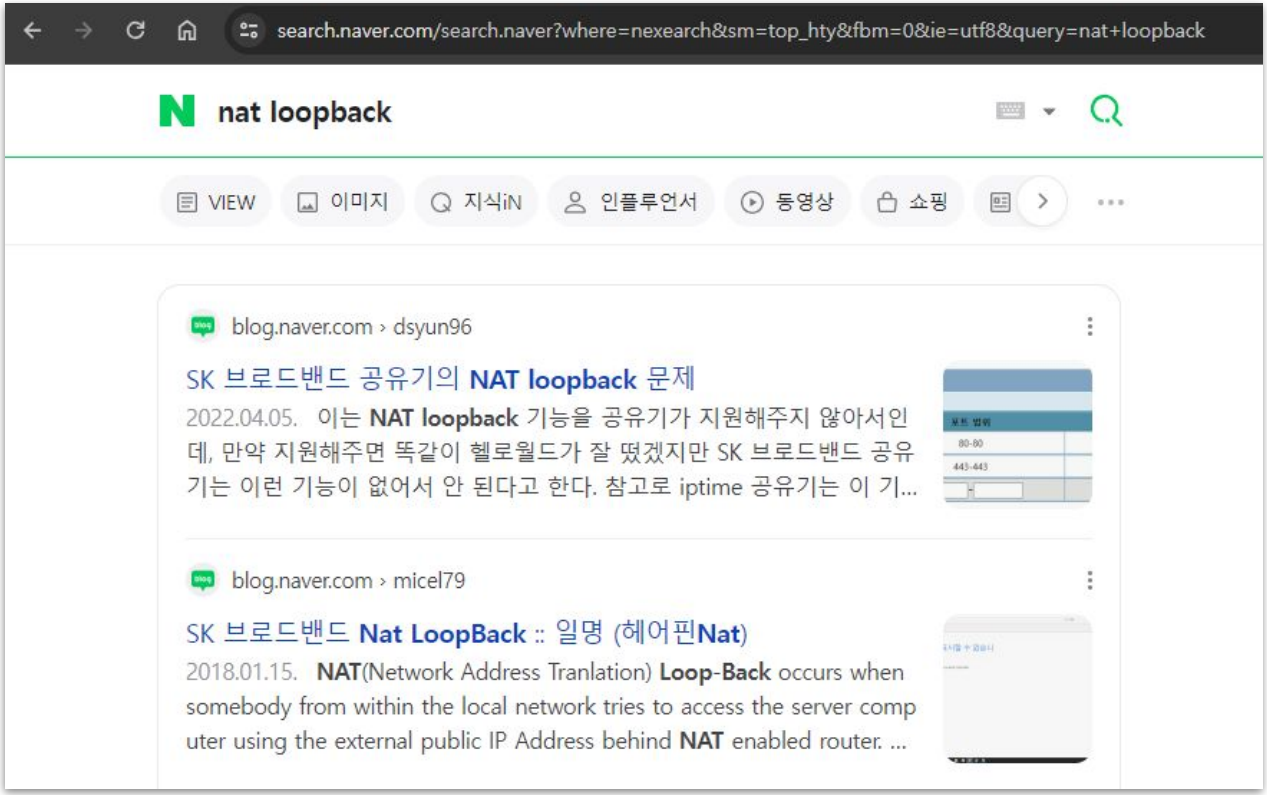
쿠키 · ETag · 위치 · 리퍼러 · DNT · XFF

상태 코드

301 Moved Permanently · 302 Found · 303 See Other · 403 Forbidden · 404 Not Found · 451 Unavailable For Legal Reasons

V · T · E

https://en.wikipedia.org/wiki/HTTP_referer


search.naver.com/search.naver?where=nexearch&sm=top_hty&fbm=0&ie=utf8&query=nat+loopback
nat loopback
VIEW
이미지
지식iN
인플루언서
동영상
쇼핑
blog.naver.com › dsyun96
SK 브로드밴드 공유기의 NAT loopback 문제
2022.04.05. 이는 NAT loopback 기능을 공유기가 지원해주지 않아서인데, 만약 지원해주면 똑같이 헬로월드가 잘 떴겠지만 SK 브로드밴드 공유기는 이런 기능이 없어서 안 된다고 한다. 참고로 iptime 공유기는 이 기...
blog.naver.com › micel79
SK 브로드밴드 Nat LoopBack :: 일명 (헤어핀Nat)
2018.01.15. NAT(Network Address Tranlation) Loop-Back occurs when somebody from within the local network tries to access the server computer using the external public IP Address behind NAT enabled router. ...

search.naver.com/search.naver?where=nexearch&sm=top_hty&fbm=0&ie=utf8&query=nat+loopback

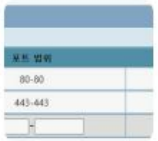
nat loopback

VIEW 이미지 지식iN 인플루언서 동영상 쇼핑

blog.naver.com › dsyun96

SK 브로드밴드 공유기의 **NAT loopback** 문제

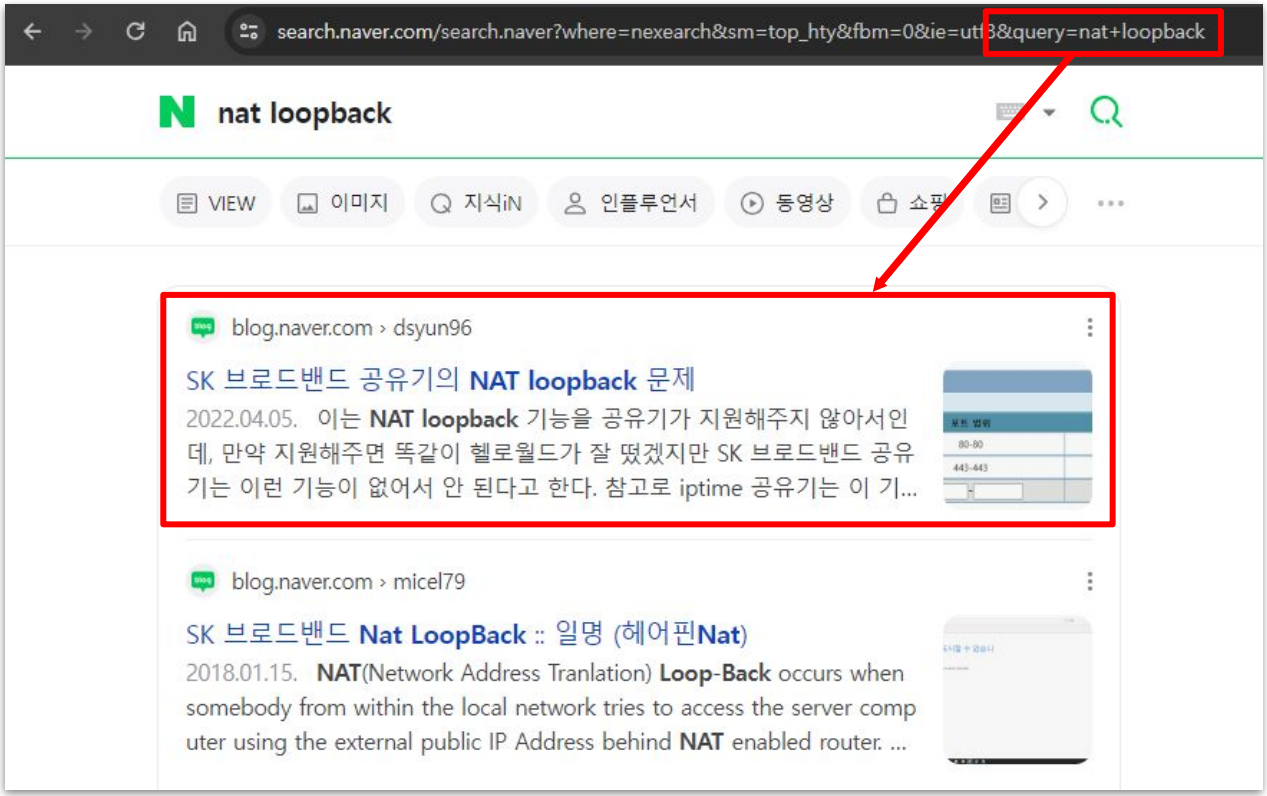
2022.04.05. 이는 **NAT loopback** 기능을 공유기가 지원해주지 않아서인데, 만약 지원해주면 똑같이 헬로월드가 잘 떴겠지만 SK 브로드밴드 공유기는 이런 기능이 없어서 안 된다고 한다. 참고로 iptime 공유기는 이 기...



blog.naver.com › micel79

SK 브로드밴드 **Nat LoopBack** :: 일명 (헤어핀**Nat**)

2018.01.15. **NAT**(Network Address Tranlation) **Loop-Back** occurs when somebody from within the local network tries to access the server computer using the external public IP Address behind **NAT** enabled router. ...


search.naver.com/search.naver?where=nexearch&sm=top_hty&fbm=0&ie=utf8&query=nat+loopback
nat loopback
VIEW
이미지
지식iN
인플루언서
동영상
쇼핑
blog.naver.com › dsyun96
SK 브로드밴드 공유기의 NAT loopback 문제
2022.04.05. 이는 NAT loopback 기능을 공유기가 지원해주지 않아서인데, 만약 지원해주면 똑같이 헬로월드가 잘 떴겠지만 SK 브로드밴드 공유기는 이런 기능이 없어서 안 된다고 한다. 참고로 iptime 공유기는 이 기...
blog.naver.com › micel79
SK 브로드밴드 Nat LoopBack :: 일명 (헤어핀Nat)
2018.01.15. NAT(Network Address Tranlation) Loop-Back occurs when somebody from within the local network tries to access the server computer using the external public IP Address behind NAT enabled router. ...

blog.naver.com/dsyun96/222692746029

윤군

헬로월드

프로필

+ 이웃추가

카테고리

- 전체보기 (1414)
- 잡담 (405)
- 일기 (67)
  - [불쌈] 주간일기 썸띤지
- Language (24)
  - C/C++ (12)
  - Python (2)
  - Java (3)
  - Kotlin (5)
  - JavaScript (1)
  - PHP (1)
- Database (3)
  - N/A (1)
  - Oracle (2)
  - MySQL (0)
- Algorithm (7)
- PS (432)
  - N/A (20)
  - BOJ (265)
  - CodeUp (8)
  - ALGOSPOT (14)
  - dovelet (16)
  - LeetCode (11)
  - Project Euler (20)
  - hacker.org (35)
  - Codeforces (31)
  - AtCoder (12)
- Setting (6)
- 공부 (178)
- 코딩대회 (26)
- 이것저것 (82)
- 일러스트 (79)
- 게임 (105)

RSS 2.0 | RSS 1.0 | ATOM 0.3

Setting

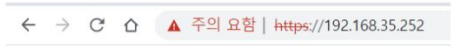
SK 브로드밴드 공유기의 NAT loopback 문제

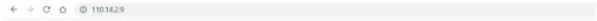
윤군 2022. 4. 5. 23:15 URL 복사 +이웃추가

현재 내 공유기가 사용 중인 공인 IP는 110.14.2.9이다.
여기서 그램 노트북은 LAN으로, 핸드폰과 맥북과 라즈베리파이는 와이파이로 연결되어 있으며 사설 IP로 192.168.35.xxx를 사용한다.

나는 라즈베리파이를 통해 웹서버를 띄웠는데, 외부에서 110.14.2.9 아이피를 통해 80포트로 들어오면 이걸 라즈베리파이가 연결된 192.168.35.xxx 아이피의 80포트로 들어가게끔 포트포워딩을 해준 상태이다.

따라서, 다른 사람들은 110.14.2.9 아이피로 접속하게 되면 문제없이 내 헬로월드를 볼 수 있다.
하지만 내 컴퓨터와 핸드폰은 상황이 다르다. 요 녀석들은 같은 내부망을 사용 중이기 때문에 192.168.35.xxx로 접속하면 헬로월드를 볼 수 있지만 다른 사람처럼 110.14.2.9 아이피로 접속하면 연결조차 되지 않는다.





사이트에 연결할 수 없음

Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder

Preserve log | Disable cache | No throttling

Filter | Invert | Hide data URLs | Hide extension URLs | All | Doc | JS | Fetch/XHR | CSS | Font | Img | Media | Manifest | WS | Wasm | Other

Blocked response cookies | Blocked requests | 3rd-party requests

200 ms 400 ms 600 ms 800 ms 1000 ms 1200 ms 1400 ms 1600 ms 1800 ms 2000 ms 2200 ms 2400 ms

Name

- rd?m=1&px=363&py=220&sx...
- 222692746029
- PostView.naver?blogId=dsyun...
- ExternalWidgetRender.naver?b...

Headers Preview Response Initiator Timing Cookies

General

Request URL:	https://blog.naver.com/dsyun96/222692746029
Request Method:	GET
Status Code:	200 OK
Remote Address:	23.201.36.184:443
Referrer Policy:	unsafe-url

Response Headers Raw

Cache-Control:	no-cache
Connection:	keep-alive
Content-Encoding:	gzip
Content-Length:	1280
Content-Type:	text/html;charset=UTF-8
Date:	Sun, 17 Dec 2023 03:12:54 GMT
Expires:	Thu, 01 Jan 1970 00:00:00 GMT
Referrer-Policy:	unsafe-url
Server:	nfront
Set-Cookie:	JSESSIONID=CD9E262688E1DC24C149E39AB6C3237B.jvm1; Path=/; Secure; HttpOnly
Vary:	Accept-Encoding

Request Headers Raw

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=0.7
Accept-Encoding:	gzip, deflate, br
Accept-Language:	ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7,ja;q=0.6
Connection:	keep-alive
Cookie:	NNB=UPVL54VKQFZGK; BA_DEVICE=bb4c0098-0221-4912-b006-aedb4b45aa11; NSCS=1; _ga_8P4PY65YZ2=GS1.1.1702096252.1.0.1702096252.60.0.0; _ga_GN4BHVX9DS=GS1.1.1702096252.1.0.1702096252.60.0.0; _ga=GA1.2.87269547.1702096253; nx_ssl=2; page_uid=iUHDxsqVOZCssnFPrB8ssssssvl-024440; JSESSIONID=4AF3AC3B0F49CE71E1266F783A80A3A3.jvm1
Dnt:	1
Host:	blog.naver.com
Referer:	https://search.naver.com/search.naver?where=nexearch&sm=top_hty&fbm=0&ie=utf8&query=nat+loopback
Sec-Ch-Ua:	"Not_A Brand";v="8", "Chromium";v="120", "Google Chrome";v="120"
Sec-Ch-Ua-Mobile:	?0
Sec-Ch-Ua-Platform:	"Windows"
Sec-Fetch-Dest:	document
Sec-Fetch-Mode:	navigate
Sec-Fetch-Site:	same-site
Sec-Fetch-User:	?1
Upgrade-Insecure-Requests:	1
User-Agent:	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36

윤군
헬로월드
프로필

+ 이웃추가

카테고리

RSS 2.0 | RSS 1.0 | ATOM 0.3

Setting

SK 브로드밴드 공유기의 NAT loopback 문제

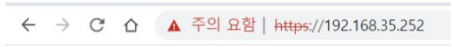
윤군 2022. 4. 5. 23:15

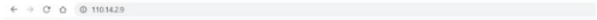
URL 복사 +이웃추가

현재 내 공유기가 사용 중인 공인 IP는 110.14.2.9이다.
여기서 그램 노트북은 LAN으로, 핸드폰과 맥북과 라즈베리파이는 와이파이로 연결되어 있으며 사설 IP로 192.168.35.xxx를 사용한다.

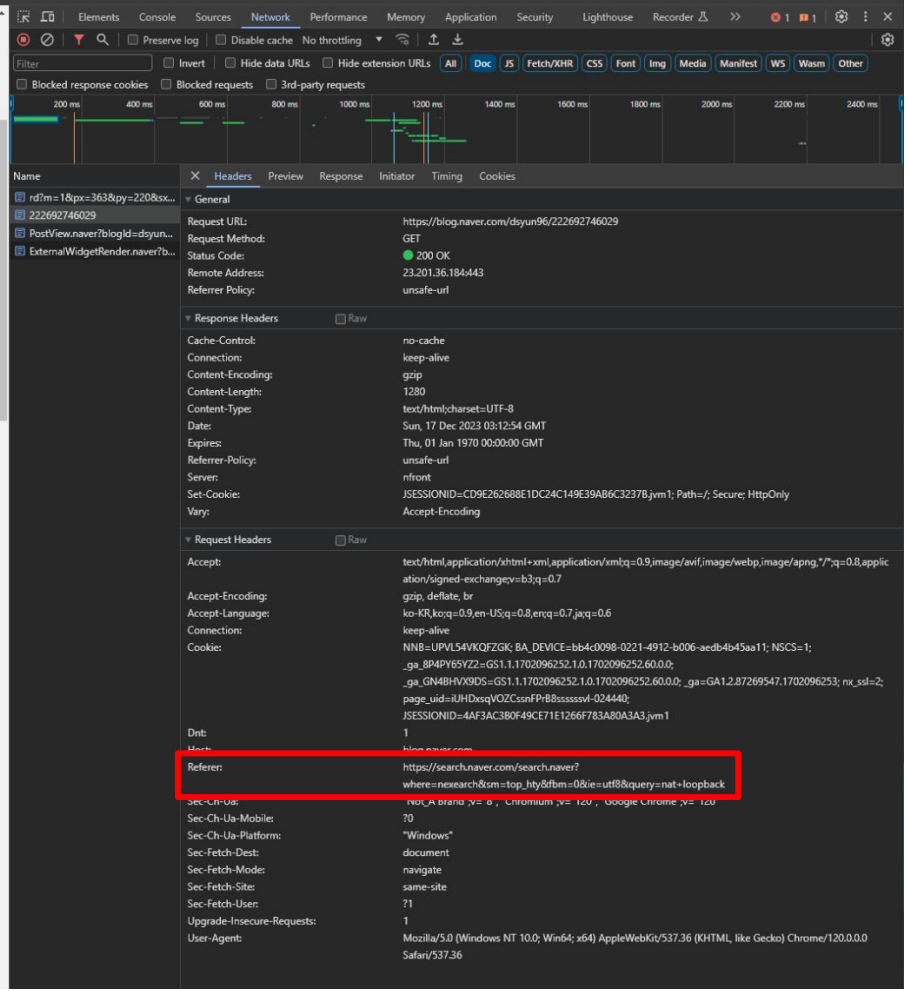
나는 라즈베리파이를 통해 웹서버를 띄웠는데, 외부에서 110.14.2.9 아이피를 통해 80포트로 들어오면 이걸 라즈베리파이가 연결된 192.168.35.xxx 아이피의 80포트로 들어가게끔 포트포워딩을 해준 상태이다.

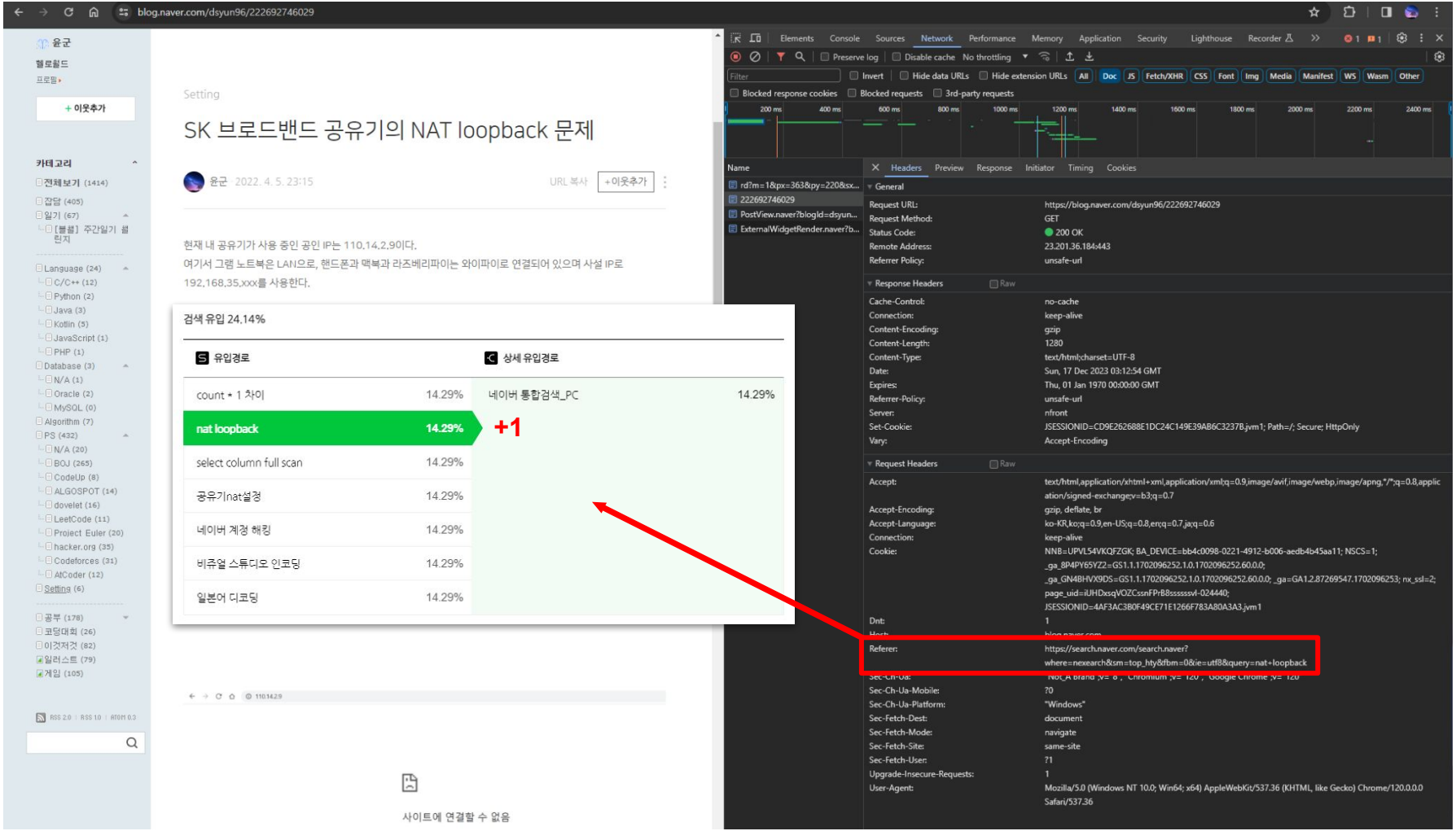
따라서, 다른 사람들은 110.14.2.9 아이피로 접속하게 되면 문제없이 내 헬로월드를 볼 수 있다.
하지만 내 컴퓨터와 핸드폰은 상황이 다르다. 요 녀석들은 같은 내부망을 사용 중이기 때문에 192.168.35.xxx로 접속하면 헬로월드를 볼 수 있지만 다른 사람처럼 110.14.2.9 아이피로 접속하면 연결조차 되지 않는다.





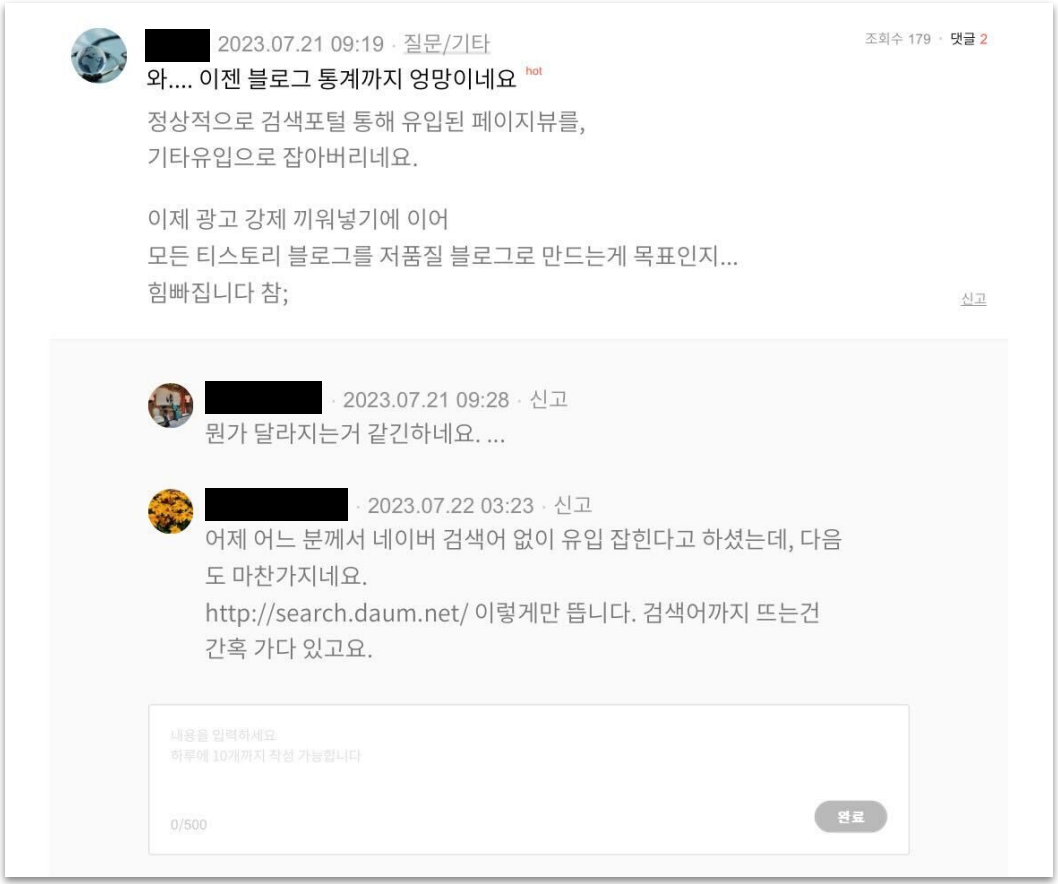
사이트에 연결할 수 없음




blog.naver.com/dsyun96/222692746029
윤군
헬로월드
프로필
+ 이웃추가
카테고리
전체보기 (1414)
잡담 (405)
일기 (67)
[불쌈] 주간일기 썸띤지
Language (24)
C/C++ (12)
Python (2)
Java (3)
Kotlin (5)
JavaScript (1)
PHP (1)
Database (3)
N/A (1)
Oracle (2)
MySQL (0)
Algorithm (7)
PS (432)
N/A (20)
BOJ (265)
CodeUp (8)
ALGOSPOT (14)
dovelet (16)
LeetCode (11)
Project Euler (20)
hacker.org (35)
Codeforces (31)
AtCoder (12)
Setting (6)
공부 (178)
코딩대회 (26)
이것저것 (82)
일러스트 (79)
게임 (105)
Setting
SK 브로드밴드 공유기의 NAT loopback 문제
윤군 2022. 4. 5. 23:15
URL 복사
+이웃추가
현재 내 공유기가 사용 중인 공인 IP는 110.14.2.90이다.
여기서 그램 노트북은 LAN으로, 핸드폰과 맥북과 라즈베리파이는 와이파이로 연결되어 있으며 사설 IP로 192.168.35.xxx를 사용한다.
검색 유입 24.14%
유입경로
상세 유입경로
count * 1 차이 14.29%
nat loopback 14.29%
select column full scan 14.29%
공유기nat설정 14.29%
네이버 계정 해킹 14.29%
비쥬얼 스튜디오 인코딩 14.29%
일본어 디코딩 14.29%
네이버 통합검색_PC 14.29%
+1
사이트에 연결할 수 없음
Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder
Preserve log Disable cache No throttling
Filter Invert Hide data URLs Hide extension URLs All Doc JS Fetch/XHR CSS Font Img Media Manifest WS Wasm Other
Blocked response cookies Blocked requests 3rd-party requests
200 ms 400 ms 600 ms 800 ms 1000 ms 1200 ms 1400 ms 1600 ms 1800 ms 2000 ms 2200 ms 2400 ms
Name
rd?m=1&px=363&py=220&sx...
222692746029
PostView.naver?blogId=dsyun...
ExternalWidgetRender.naver?b...
Headers Preview Response Initiator Timing Cookies
General
Request URL: https://blog.naver.com/dsyun96/222692746029
Request Method: GET
Status Code: 200 OK
Remote Address: 23.201.36.184:443
Referrer Policy: unsafe-url
Response Headers Raw
Cache-Control: no-cache
Connection: keep-alive
Content-Encoding: gzip
Content-Length: 1280
Content-Type: text/html;charset=UTF-8
Date: Sun, 17 Dec 2023 03:12:54 GMT
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Referrer-Policy: unsafe-url
Server: nfront
Set-Cookie: JSESSIONID=CD9E262688E1DC24C149E39AB6C3237B.jvm1; Path=/; Secure; HttpOnly
Vary: Accept-Encoding
Request Headers Raw
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=0.7
Accept-Encoding: gzip, deflate, br
Accept-Language: ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7,ja;q=0.6
Connection: keep-alive
Cookie: NNB=UPVL54VKQFZGK; BA_DEVICE=bb4c0098-0221-4912-b006-aedb4b45aa11; NSCS=1; _ga_8P4PY65YZ2=GS1.1.1702096252.1.0.1702096252.60.0.0; _ga_GN4BHVX9DS=GS1.1.1702096252.1.0.1702096252.60.0.0; _ga=GA1.2.87269547.1702096253; nx_ssl=2; page_uid=iUHDxsqVOZCssnFPrB8ssssssvl-024440; JSESSIONID=4AF3AC3B0F49CE71E1266F783A80A3A3.jvm1
Dnt: 1
Referer: https://search.naver.com/search.naver?where=nexearch&sm=top_hty&fbm=0&ie=utf8&query=nat+loopback
Sec-Ch-Ua: "Not_A Brand";v="8", "Chromium";v="120", "Google Chrome";v="120"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-site
Sec-Fetch-User: ?1
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36

원리는 이게 전부입니다. 쉽죠?

좀 더 심화된 내용으로
들어가봅시다.


[redacted] 2023.07.21 09:19 · 질문/기타
조회수 179 · 댓글 2
와.... 이젠 블로그 통계까지 엉망이네요 hot
정상적으로 검색포털 통해 유입된 페이지뷰를,
기타유입으로 잡아버리네요.
이제 광고 강제 끼워넣기에 이어
모든 티스토리 블로그를 저품질 블로그로 만드는게 목표인지...
힘빠집니다 참;
신고
[redacted] · 2023.07.21 09:28 · 신고
뭔가 달라지는거 같긴하네요. ...
[redacted] · 2023.07.22 03:23 · 신고
어제 어느 분께서 네이버 검색어 없이 유입 잡힌다고 하셨는데, 다음도 마찬가지네요.
http://search.daum.net/ 이렇게만 뜹니다. 검색어까지 뜨는건 간혹 가다 있고요.
내용을 입력하세요
하루에 10개까지 작성 가능합니다
0/500
완료

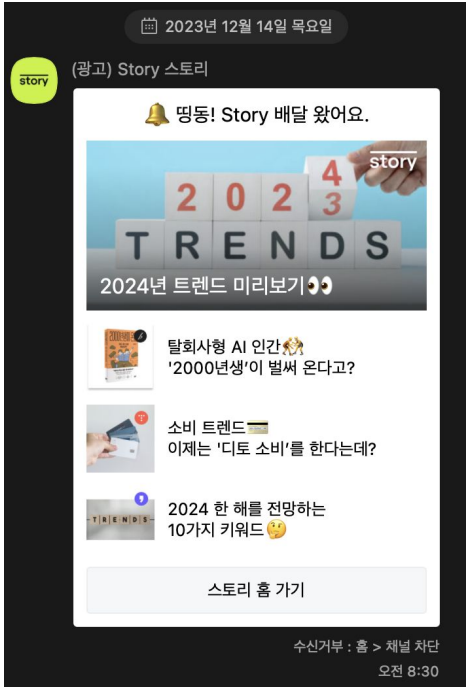
원인은 바로 며칠 전에 추가된
자동로그인 프로세스

자동로그인 탄생 배경

티스토리, 브런치(現 브런치스토리), 카카오스토리가 묶여서스토리홈 탄생

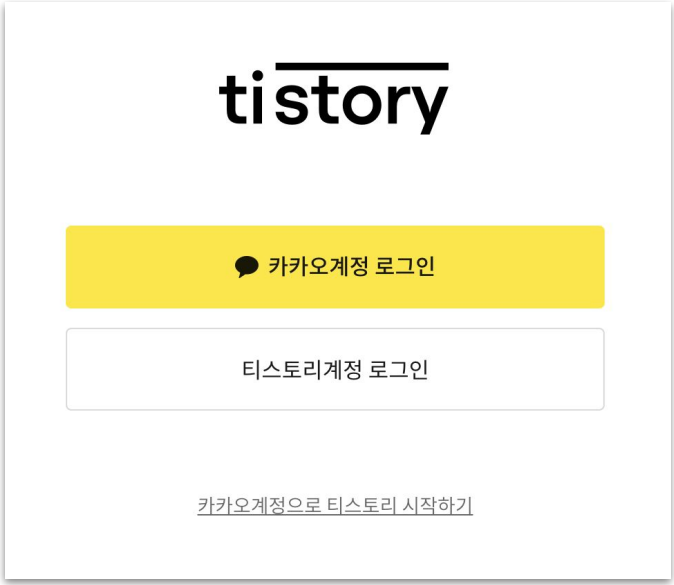
자동로그인 탄생 배경

티스토리, 브런치(現 브런치스토리), 카카오스토리가 묶여서스토리홈 탄생
카카오톡에 스토리 채널이 생겨서 카카오톡 → 티스토리 유입 동선 발생



자동로그인 탄생 배경

티스토리, 브런치(現 브런치스토리), 카카오스토리가 묶여서스토리홈 탄생
카카오톡에 스토리 채널이 생겨서 카카오톡 → 티스토리 유입 동선 발생
티스토리는 카카오 계정으로 로그인이 가능한 상황



자동로그인 탄생 배경

티스토리, 브런치(現 브런치스토리), 카카오스토리가 묶여서스토리홈 탄생
카카오톡에 스토리 채널이 생겨서 카카오톡 → 티스토리 유입 동선 발생
티스토리는 카카오 계정으로 로그인이 가능한 상황

카카오톡에서는 어차피 로그인이 되어있으니까,
티스토리로 유입될 때 자동으로 로그인이 되면 편하지 않을까?!

자동로그인 탄생 배경

티스토리, 브런치(現 브런치스토리), 카카오스토리가 묶여서스토리홈 탄생
카카오톡에 스토리 채널이 생겨서 카카오톡 → 티스토리 유입 동선 발생
티스토리는 카카오 계정으로 로그인이 가능한 상황

카카오톡에서는 어차피 로그인이 되어있으니까,
티스토리로 유입될 때 자동으로 로그인이 되면 편하지 않을까?!

자동로그인 탄생 🎉

자동로그인 동작 방식

1. 유저가 개별 유저블로그에 접근

자동로그인 동작 방식

1. 유저가 개별 유저블로그에 접근
2. 서버에서 자동로그인을 시켜야 하는 요청인지 판단

자동로그인 동작 방식

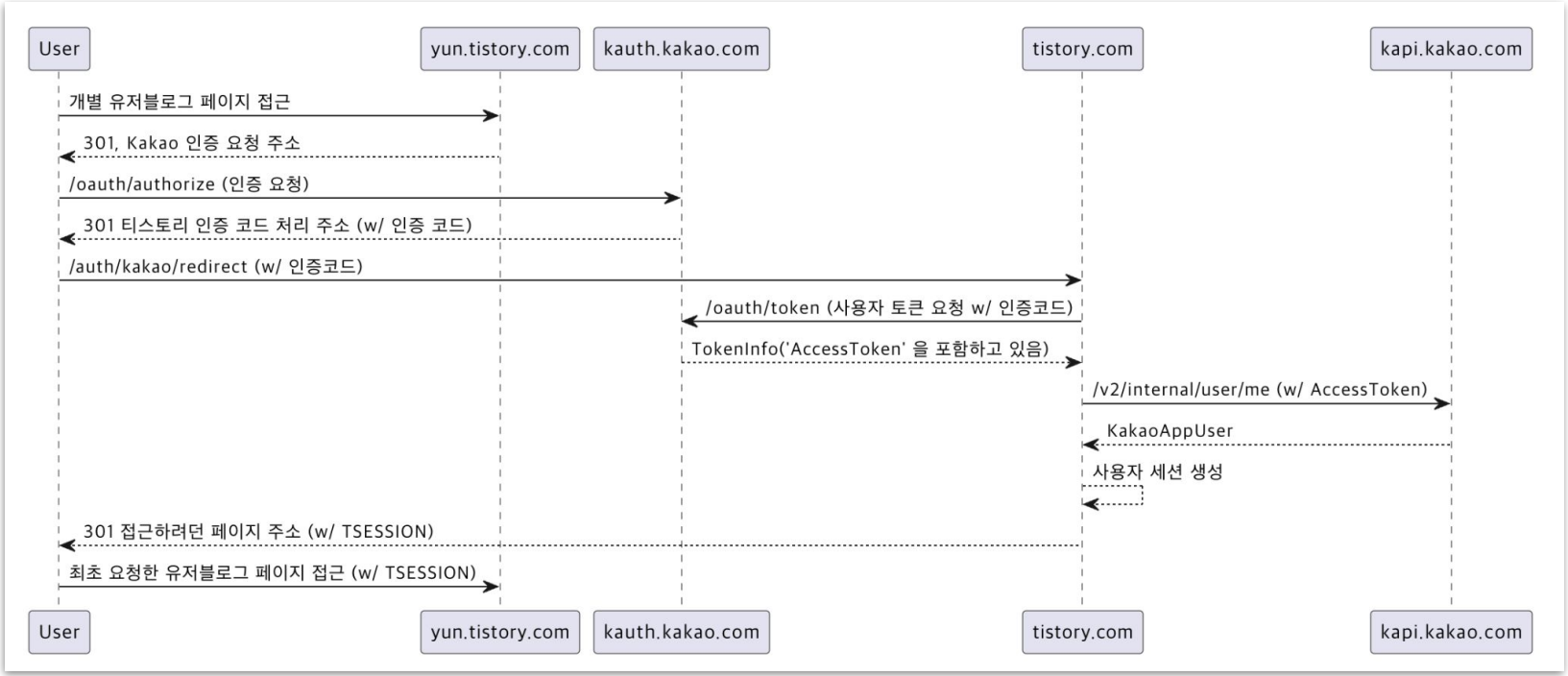
1. 유저가 개별 유저블로그에 접근
2. 서버에서 자동로그인을 시켜야 하는 요청인지 판단
   - 자동로그인 기능을 켜둔 상태인지
   - 자동로그인 방지를 위한 쿠키가 없는지
   - 티스토리 세션이 없는지
   - 봇 요청이 아닌지 등등

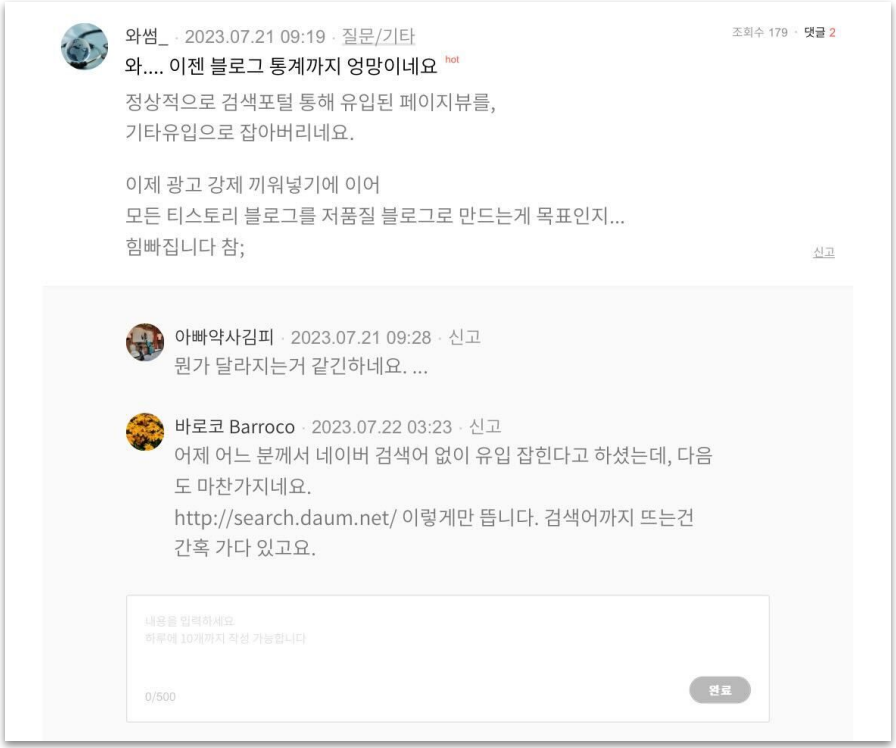
자동로그인 동작 방식

1. 유저가 개별 유저블로그에 접근
2. 서버에서 자동로그인을 시켜야 하는 요청인지 판단
   - 자동로그인 기능을 켜둔 상태인지
   - 자동로그인 방지를 위한 쿠키가 없는지
   - 티스토리 세션이 없는지
   - 봇 요청이 아닌지 등등
3. 자동로그인을 시켜야 한다면, 인증서버인 kauth.kakao.com으로 리다이렉트하여 자동로그인 요청

자동로그인 동작 방식

1. 유저가 개별 유저블로그에 접근
2. 서버에서 자동로그인을 시켜야 하는 요청인지 판단
   - 자동로그인 기능을 켜둔 상태인지
   - 자동로그인 방지를 위한 쿠키가 없는지
   - 티스토리 세션이 없는지
   - 봇 요청이 아닌지 등등
3. 자동로그인을 시켜야 한다면, 인증서버인 kauth.kakao.com으로 리다이렉트하여 자동로그인 요청
4. 성공 여부에 따라 티스토리 세션을 만들고 1번에서 요청한 url으로 리다이렉트


User
yun.tistory.com
kauth.kakao.com
tistory.com
kapi.kakao.com
개별 유저블로그 페이지 접근
301, Kakao 인증 요청 주소
/oauth/authorize (인증 요청)
301 티스토리 인증 코드 처리 주소 (w/ 인증 코드)
/auth/kakao/redirect (w/ 인증코드)
/oauth/token (사용자 토큰 요청 w/ 인증코드)
TokenInfo('AccessToken' 을 포함하고 있음)
/v2/internal/user/me (w/ AccessToken)
KakaoAppUser
사용자 세션 생성
301 접근하려던 페이지 주소 (w/ TSESSION)
최초 요청한 유저블로그 페이지 접근 (w/ TSESSION)
User
yun.tistory.com
kauth.kakao.com
tistory.com
kapi.kakao.com



자동로그인이랑 검색어 유실이랑 무슨 상관일까?

원인 분석을 위한 삽질의 시작

테스트 상황 #1

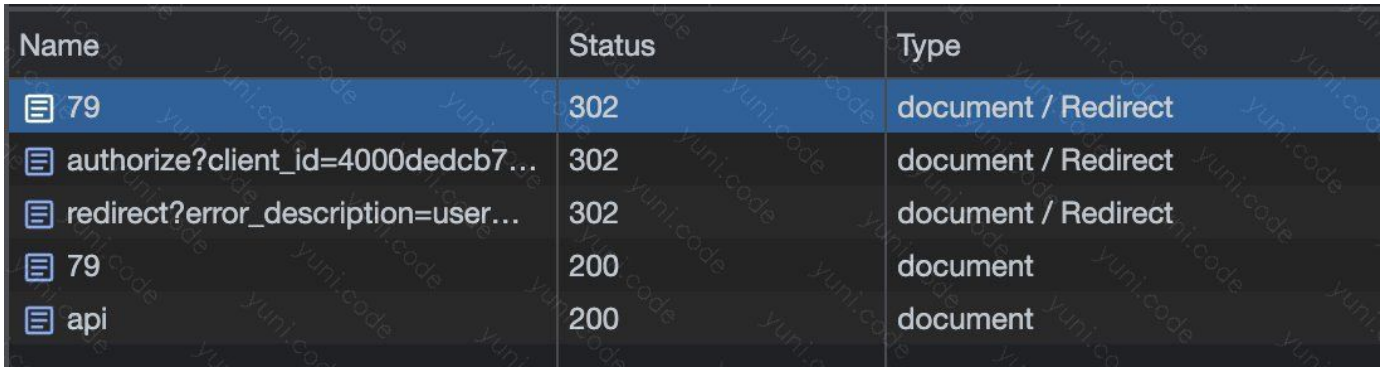
step 0. 최초 상황

현재 유저가 아래의 페이지에 머무르고 있는 상황을 가정.
원래라면 유저블로그에 접근했을 때 이미 자동로그인 프로세스를 거치고 '자동로그인 방지 쿠키'가 만들어졌겠지만,
테스트를 위해 이 페이지에서 쿠키를 모두 지우고 진행.

테스트 상황 #1

step 1. 다른 게시글 클릭

121번 게시글에서 79번 게시글을 누르자, 정상적으로 자동로그인을 타는 모습.

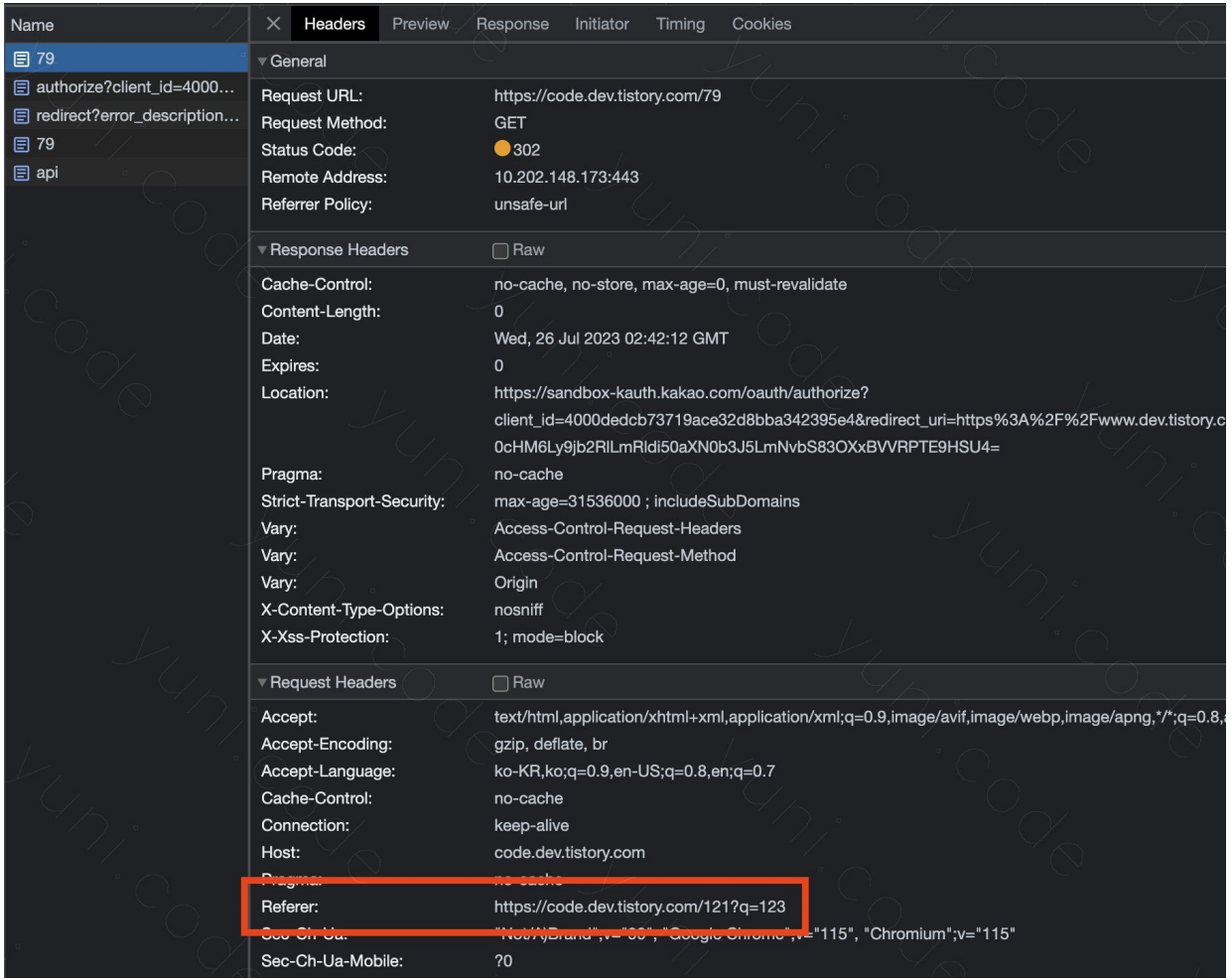


Name	Status	Type
79	302	document / Redirect
authorize?client_id=4000dedcb7…	302	document / Redirect
redirect?error_description=user…	302	document / Redirect
79	200	document
api	200	document

테스트 상황 #1

step 1. 다른 게시글 클릭

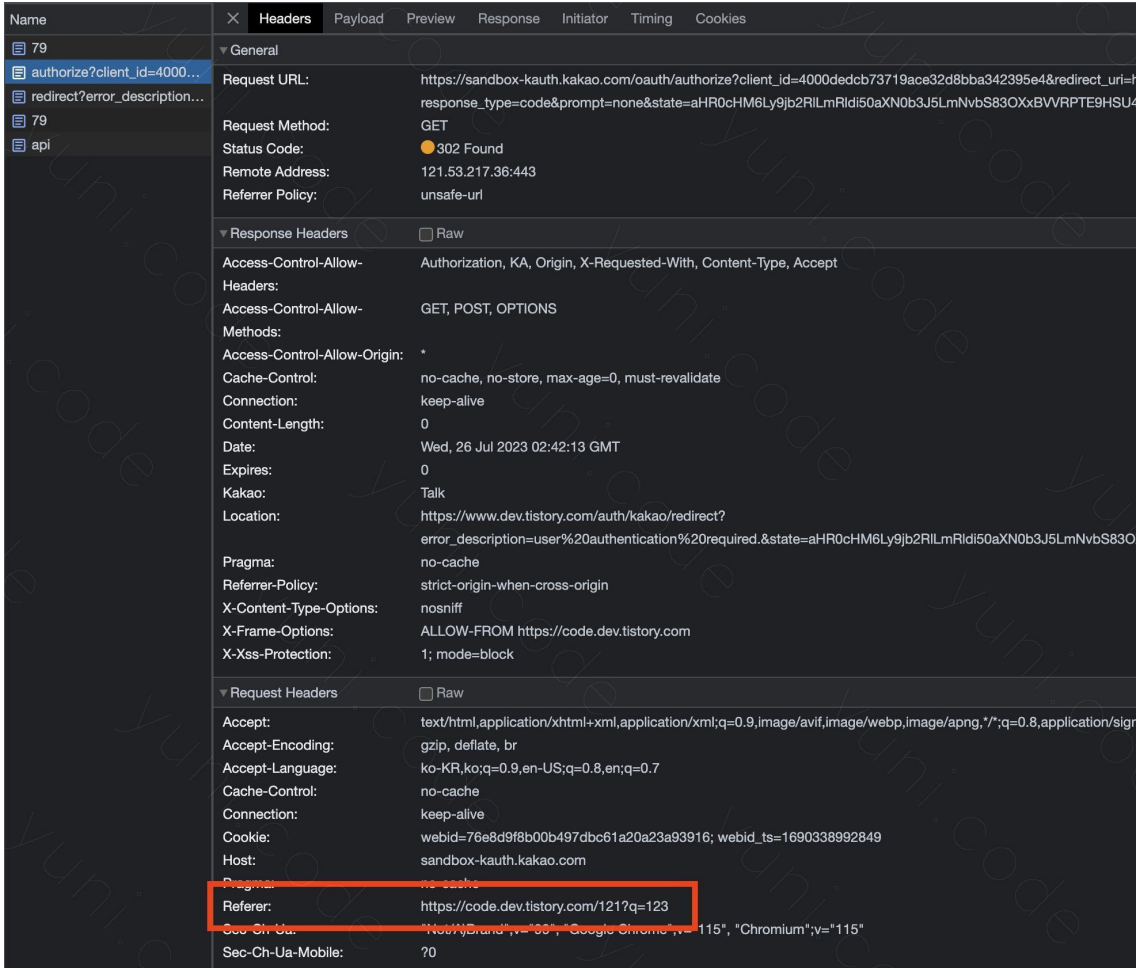
121번 게시글에서 79번 게시글을 누르자, 정상적으로 자동로그인을 타는 모습.



테스트 상황 #1

step 2. kauth로 로그인 요청

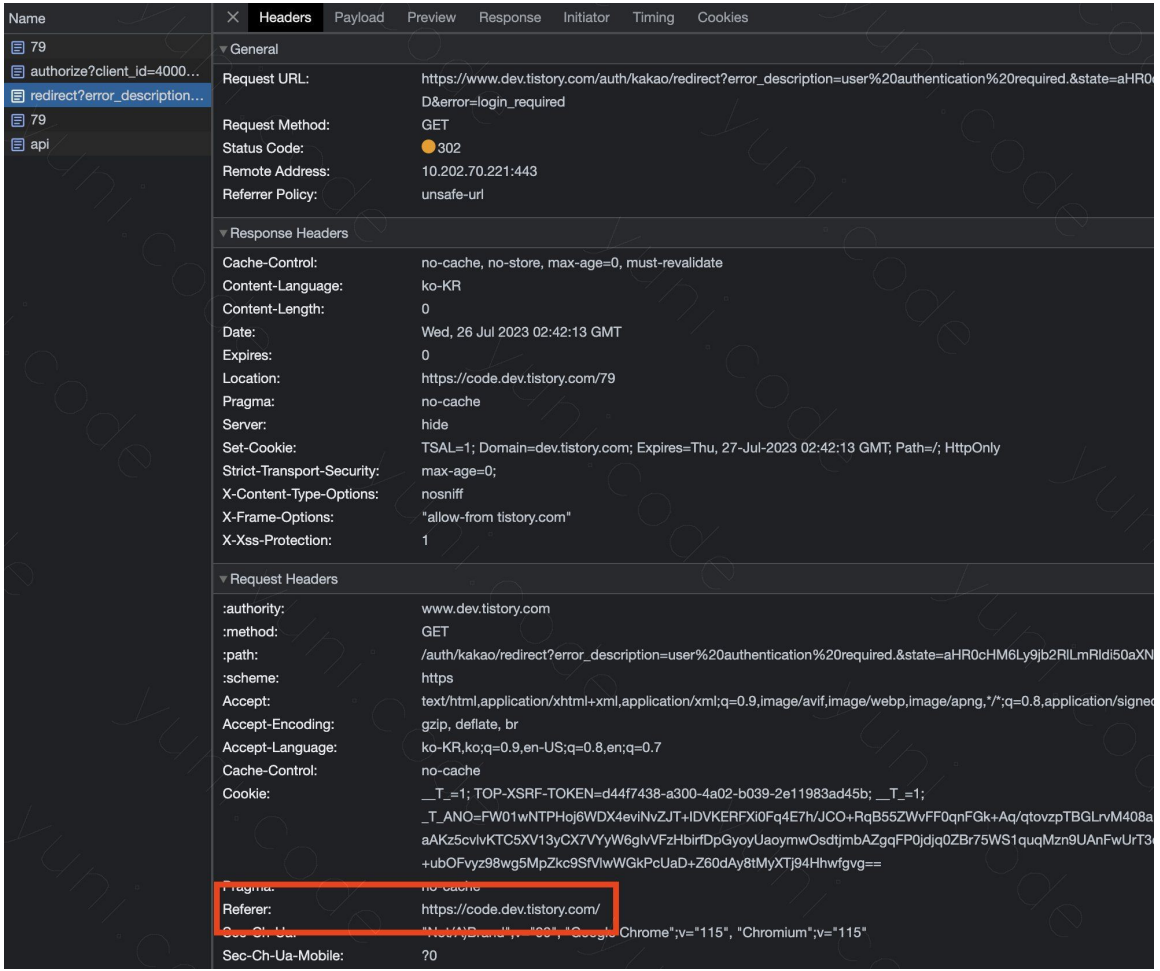
kauth로 redirect 될 때 역시 마찬가지로 referer는 그대로 잘 넘어가는 중.



테스트 상황 #1

step 3. 티스토리 TOP callback

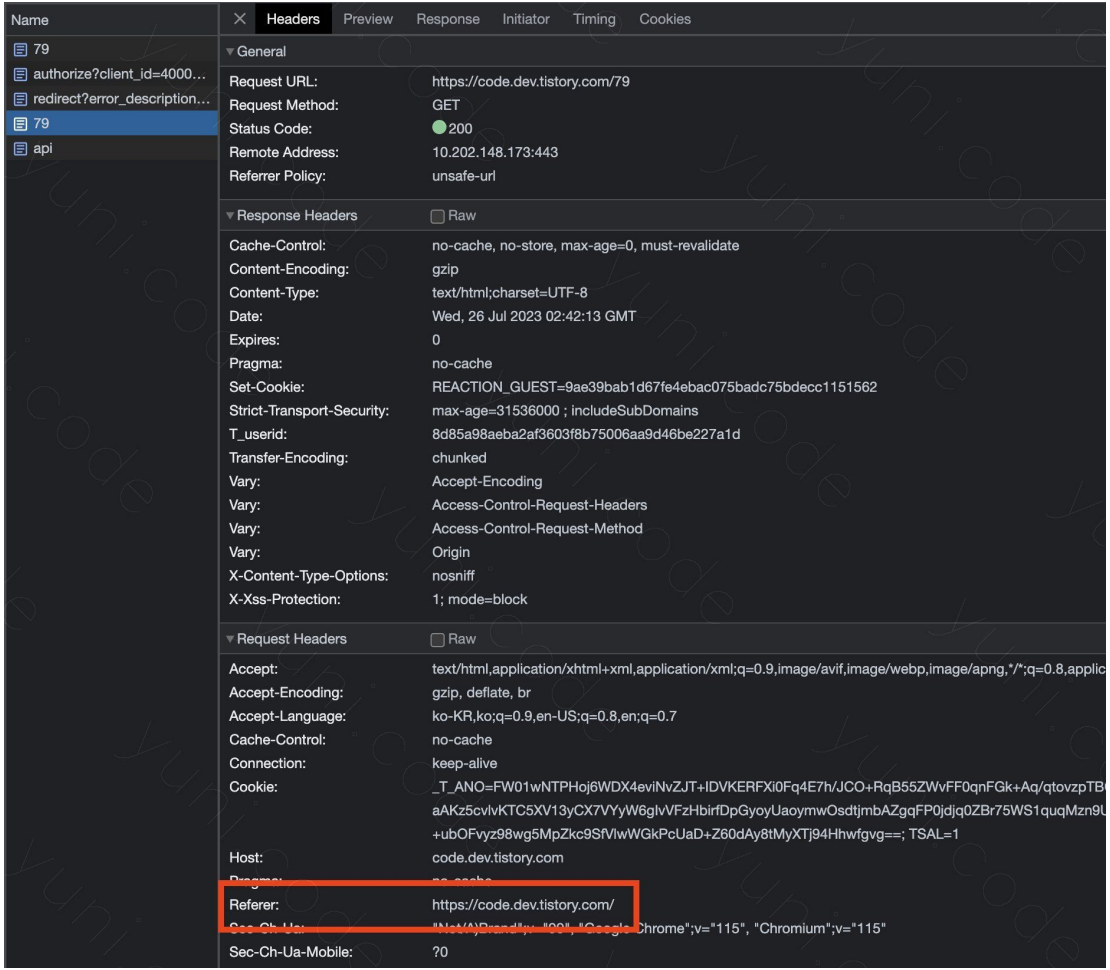
이 부분이 문제의 핵심!
kauth에서 TOP으로 redirect 될 때,
갑자기 path와 query string 정보가
유실되는 것을 확인.



테스트 상황 #1

step 4. 이동하고자 했던 url로 최종 도착

79번 게시글로 이동하는 게 목적이었으니,
최종적으로 79번 게시글에 접근 완료.
하지만 step 3에서 이미 referer에 path와 query string 정보가 사라진 뒤이므로,
여기서도 referer에는 도메인 정보만 남음.



재현은 성공했고…
그럼 무엇이 문제인가?

브라우저의 보안정책?
kauth의 서버 설정?
redirect를 여러 번 거쳐서?

자동로그인을 적용한 브런치는
이런 문제가 없을까?

테스트 상황 #2

step 0. 최초 상황

query string이 referer에 잘 남는지 확인하기 위해 적당히 세팅.
아까와 마찬가지로 이 상태에서 마찬가지로 쿠키는 모두 삭제.

테스트 상황 #2

step 1. 브런치 홈으로 접근

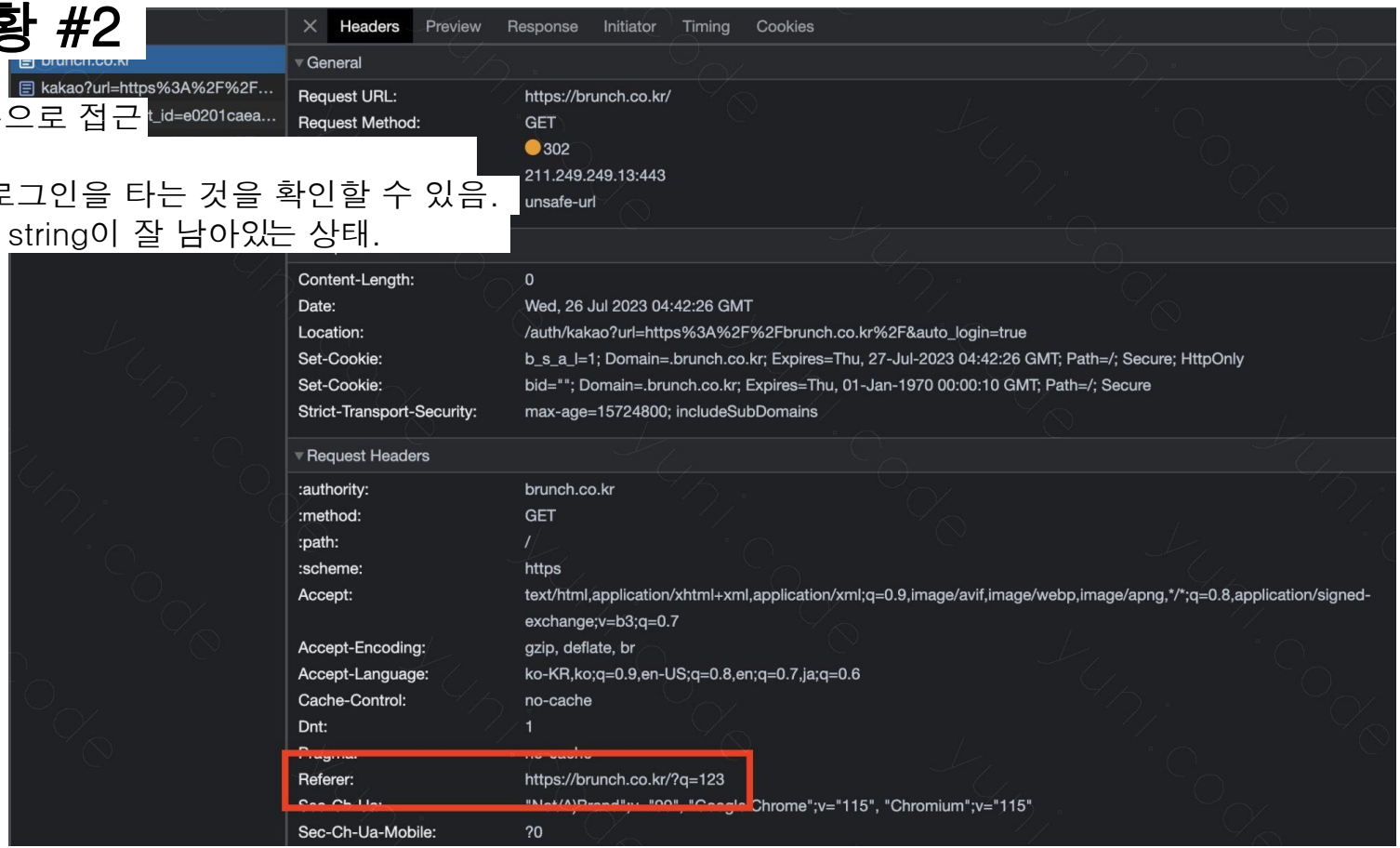
정상적으로 자동로그인을 타는 것을 확인할 수 있음.

Name	Status	Type
brunch.co.kr	302	document / Redirect
kakao?url=https%3A%2F%2Fbrunch....	302	document / Redirect
authorize?client_id=e0201caea90cafb...	302	document / Redirect
kakao?error_description=NOT_CONN...	302	document / Redirect
brunch.co.kr	200	document

테스트 상황 #2

step 1. 브런치 홈으로 접근

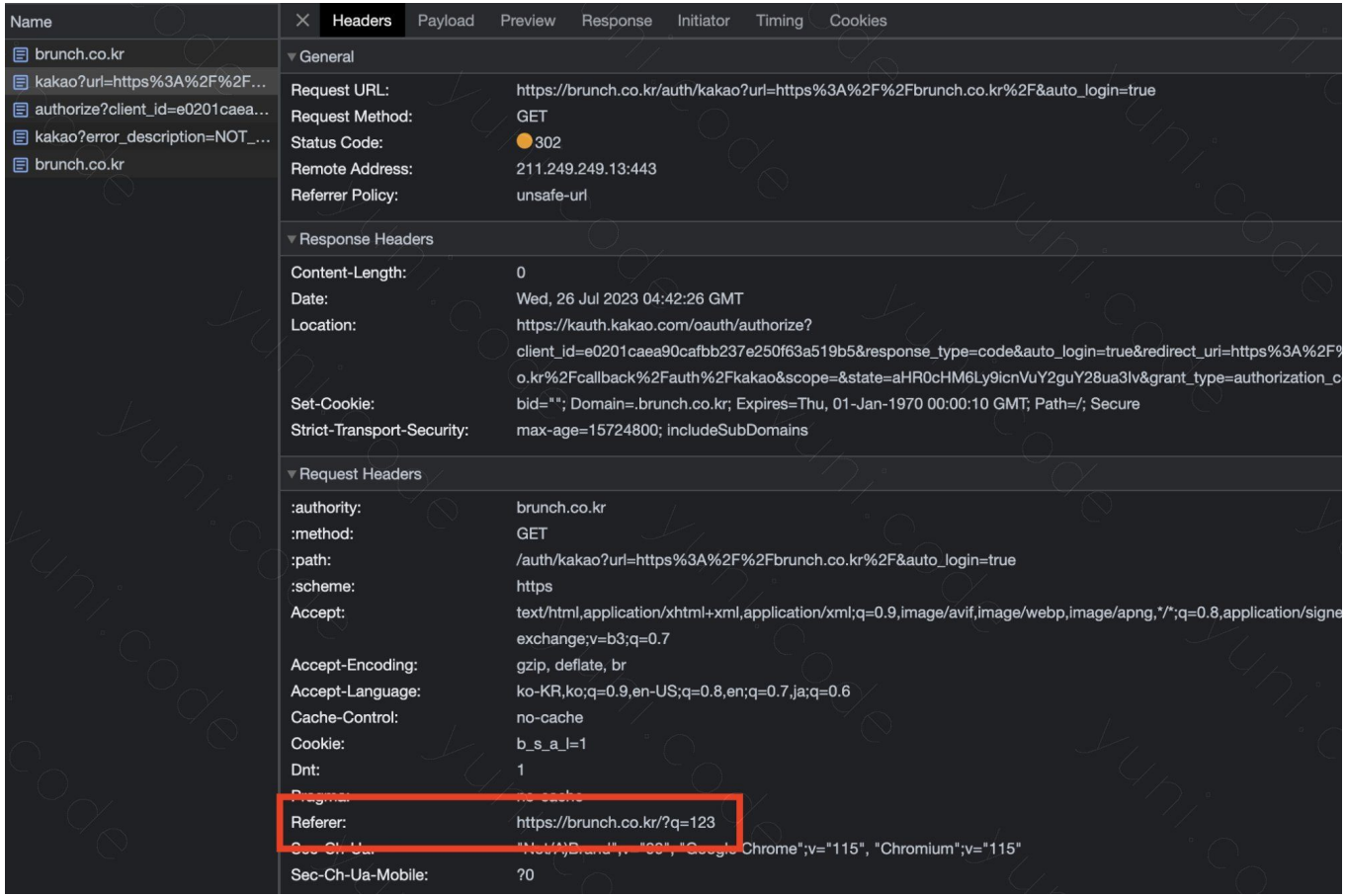
정상적으로 자동로그인을 타는 것을 확인할 수 있음.
referer에도 query string이 잘 남아있는 상태.



테스트 상황 #2

step 2. 브런치의 auth api로 redirect

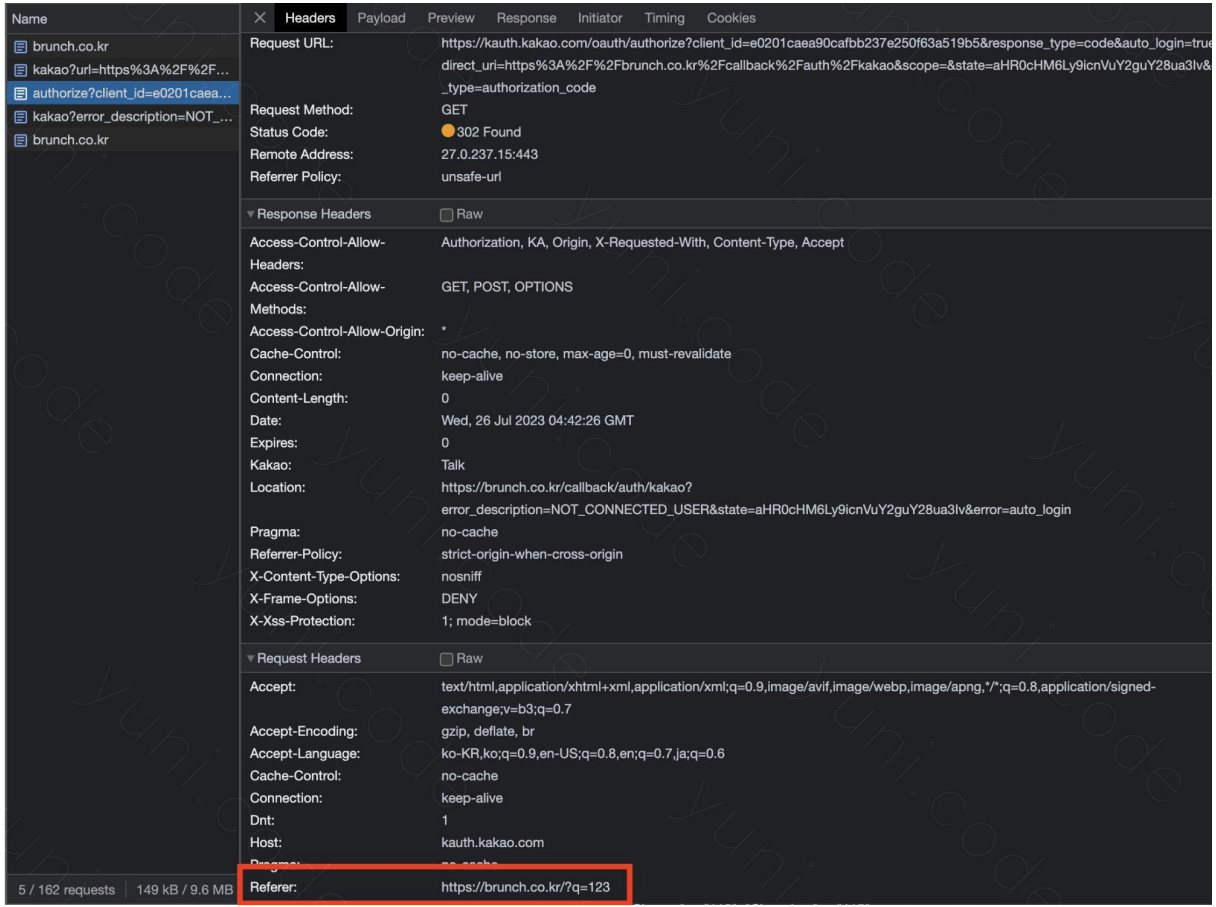
정상.



테스트 상황 #2

step 3. kauth로 로그인 요청

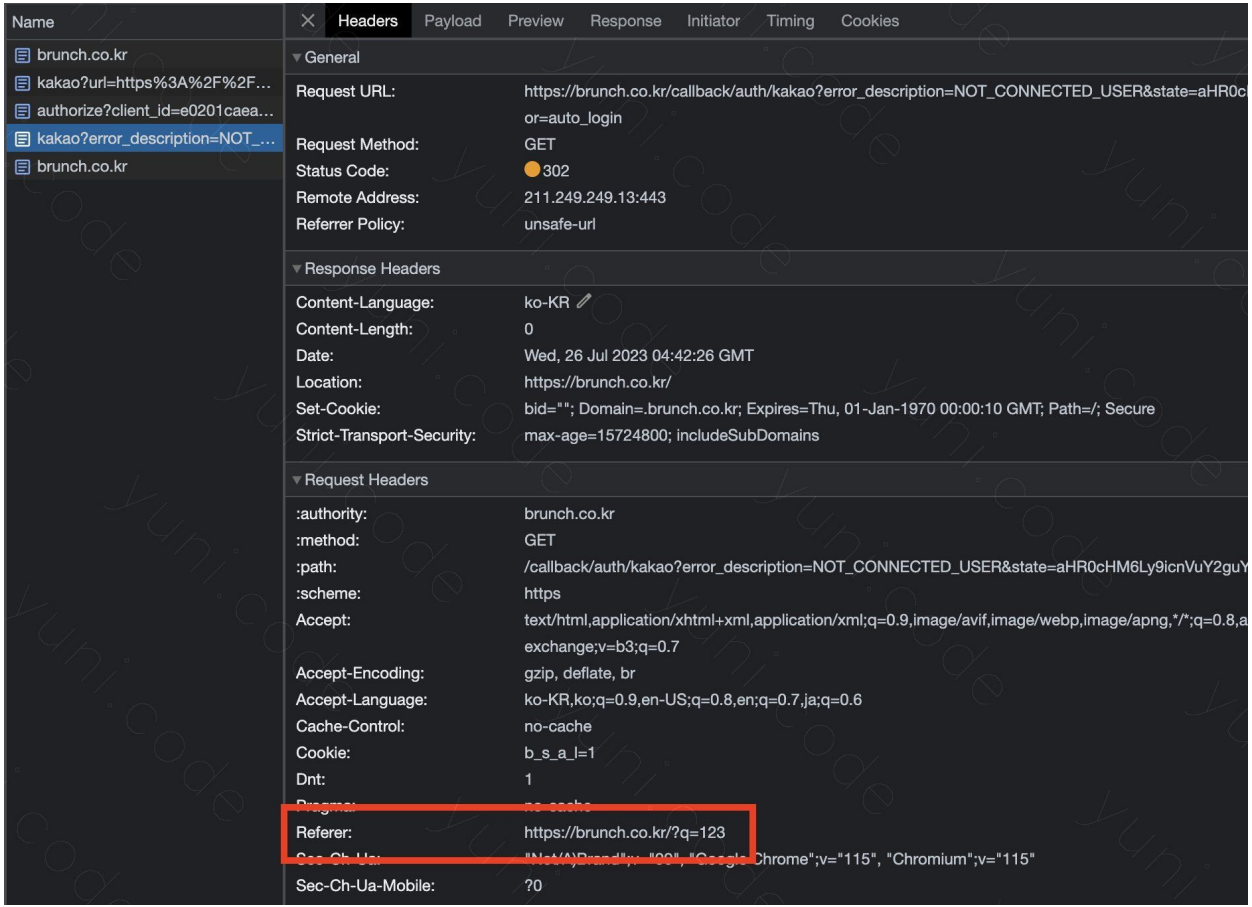
여기까지도 정상.
과연 이 다음에는 그대로 남을까?



테스트 상황 #2

step 4. 브런치 callback

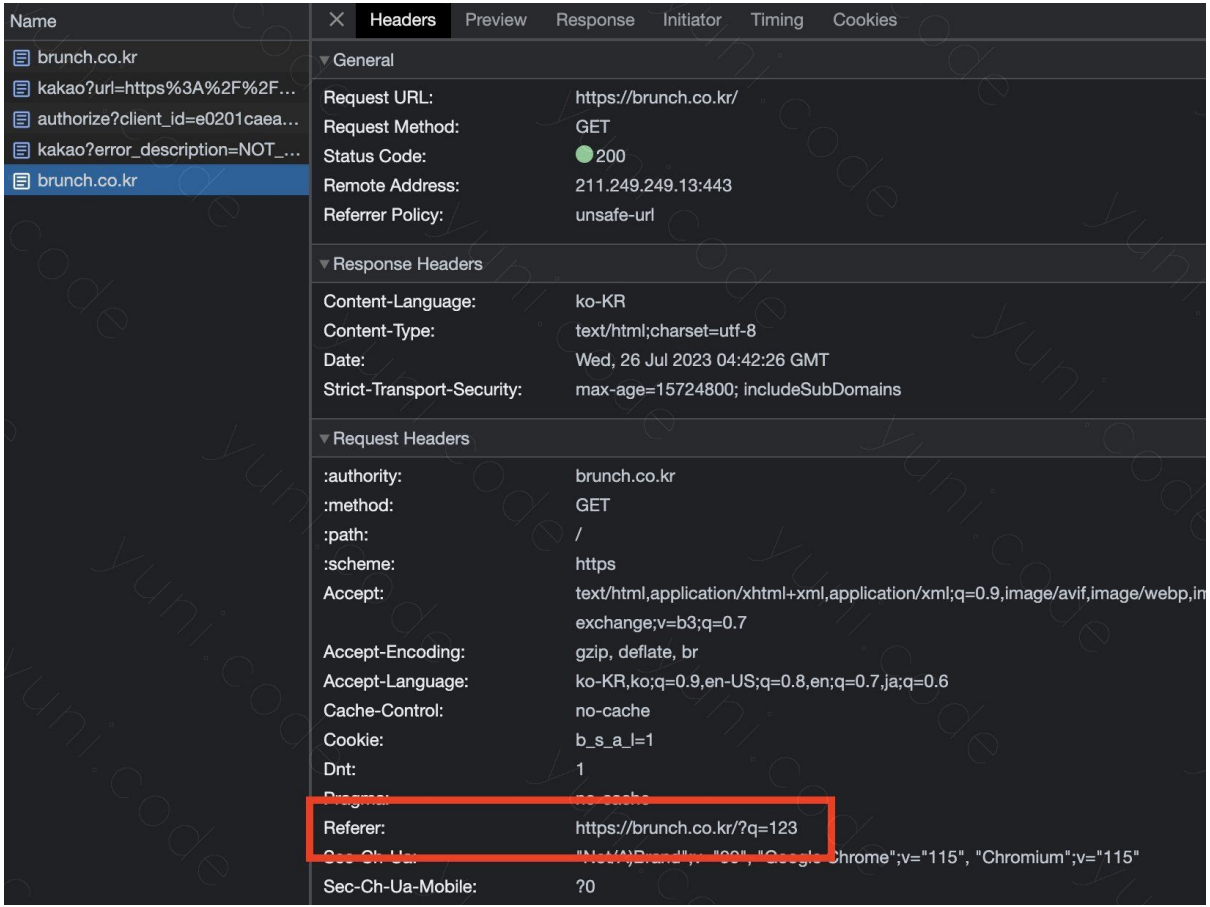
마찬가지로 query string이 잘 남아있다!!



테스트 상황 #2

step 5. 이동하고자 했던 url로 최종 도착

결국 마지막까지 최초 referer 정보가 query string까지 유지하며 요청이 잘 보내졌음.



왜 브런치는 되고, 티스토리는 안 될까?

🤔

우선 눈에 띄는 차이점...

브런치는 다음과 같은 redirect를 가진다.
brunch → kauth → brunch

티스토리는 다음과 같은 redirect를 가진다.
userblog → kauth → TOP → userblog

우선 눈에 띄는 차이점...

브런치는 다음과 같은 redirect를 가진다.
brunch → kauth → brunch

티스토리는 다음과 같은 redirect를 가진다.
userblog → kauth → TOP → userblog

그렇다면, userblog → kauth → userblog로 redirect가 이루어지면 괜찮을까?

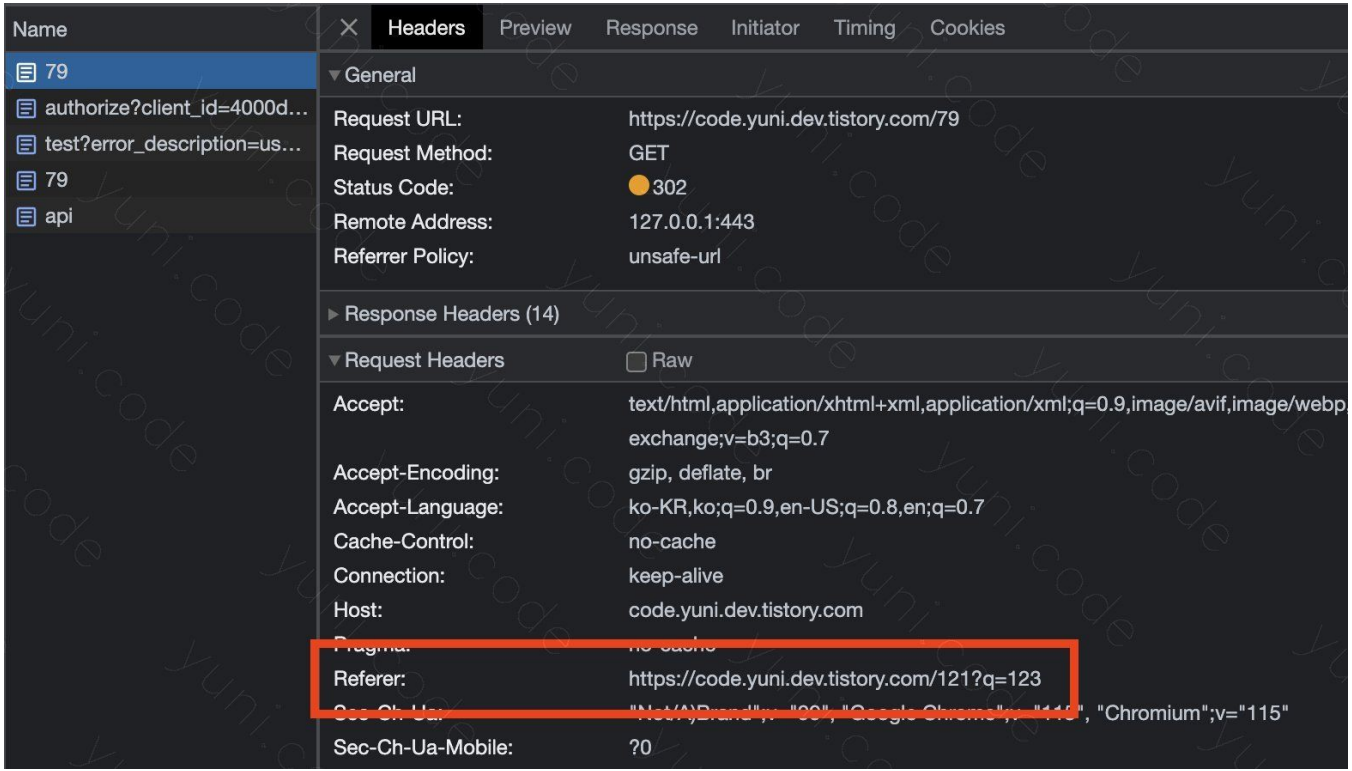
테스트 상황 #3

step 0. 최초 상황

로컬에서 빌드하였고 테스트를 진행(kauth에서 바로 userblog로 받으려면 새로운 test api가 필요).
마찬가지로 현재 사용자는 최초에 이 페이지에서 시작.

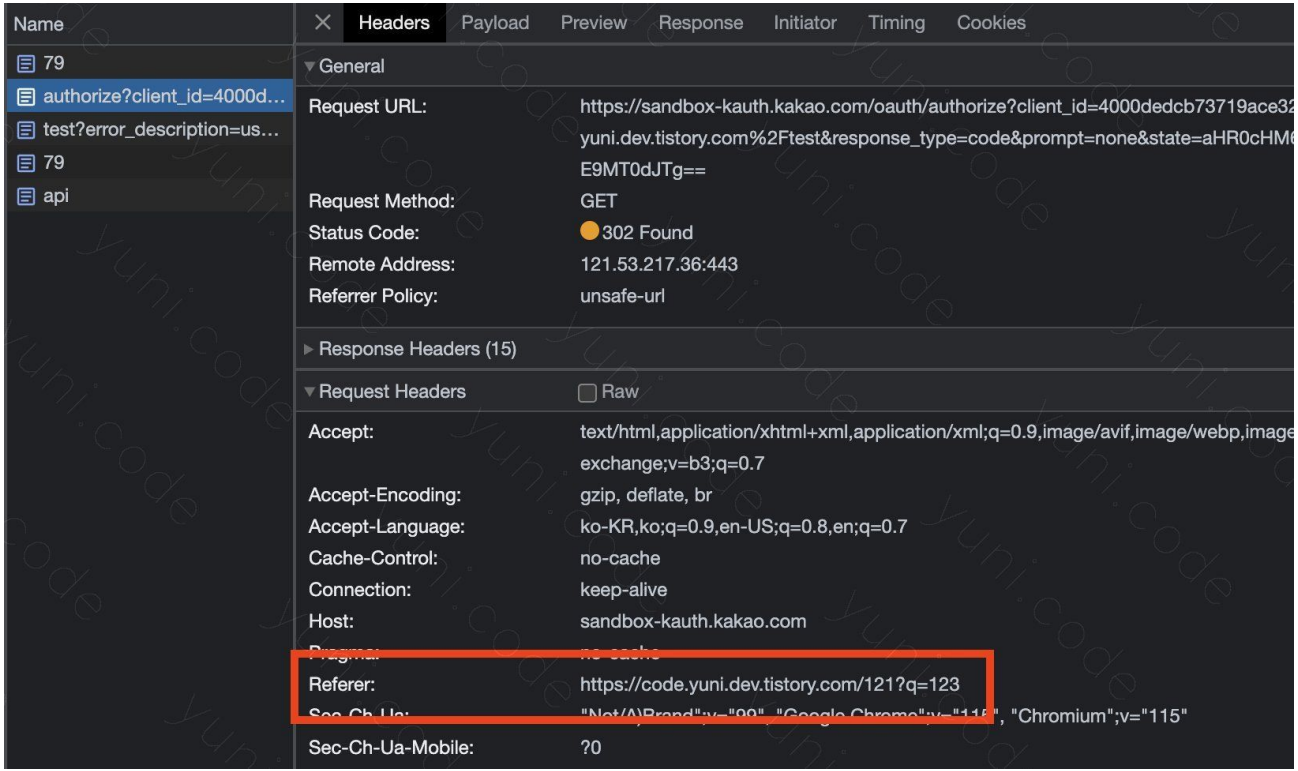
테스트 상황 #3

step 1. 다른 게시글 클릭



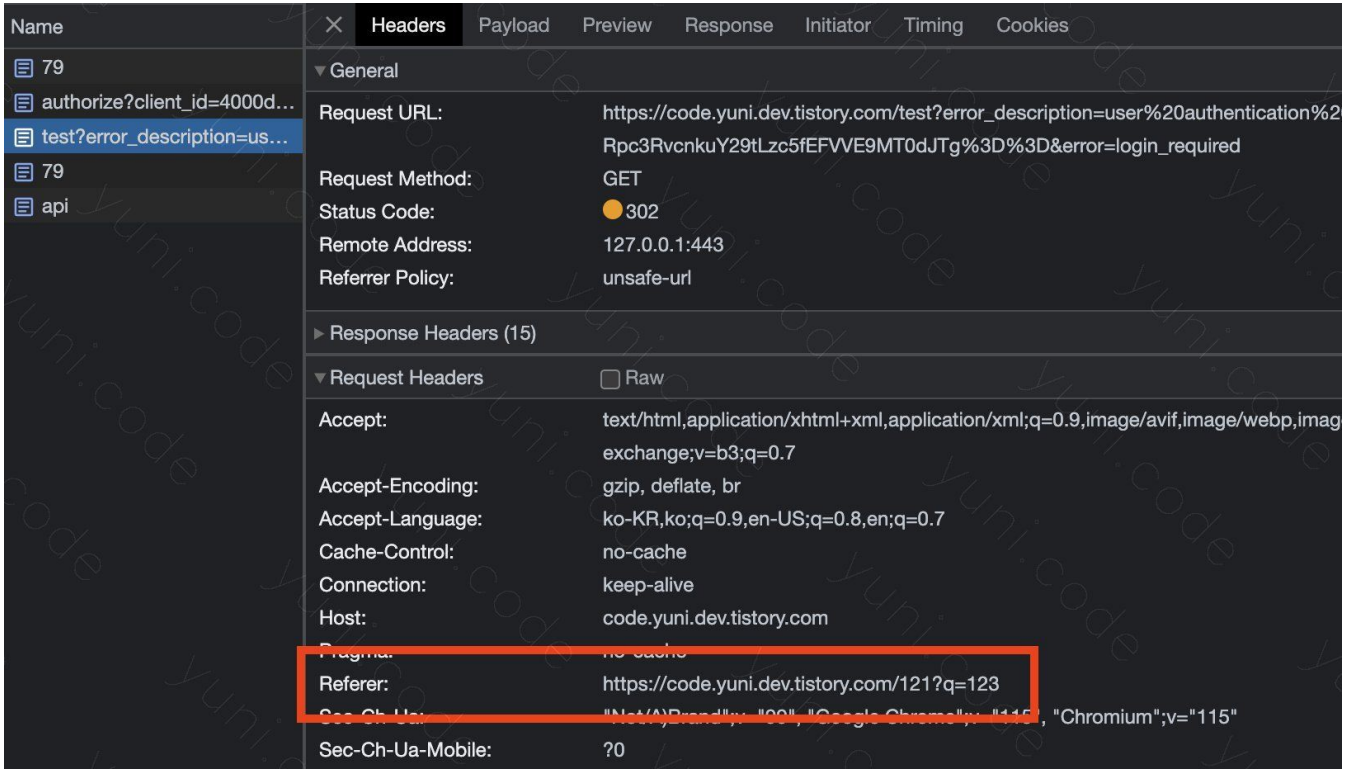
테스트 상황 #3

step 2. kauth로 로그인 요청



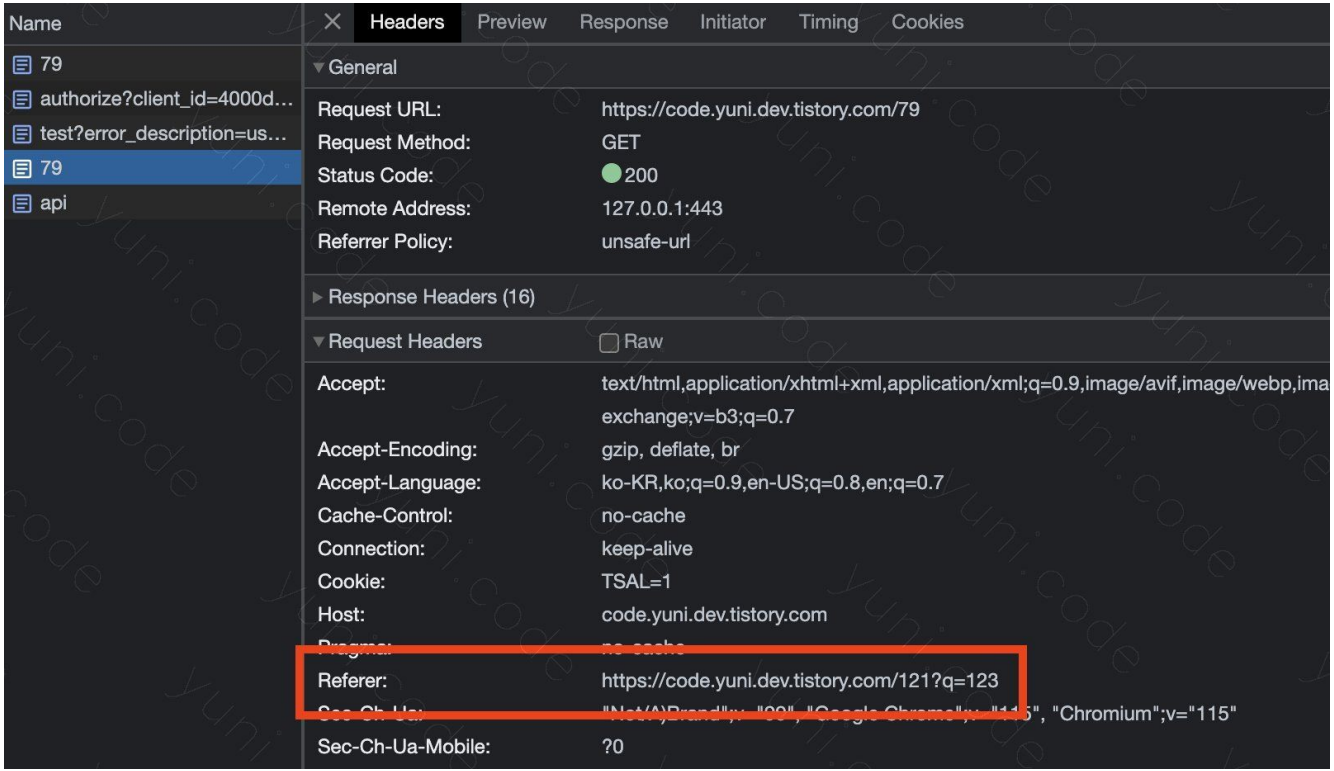
테스트 상황 #3

step 3. /test callback



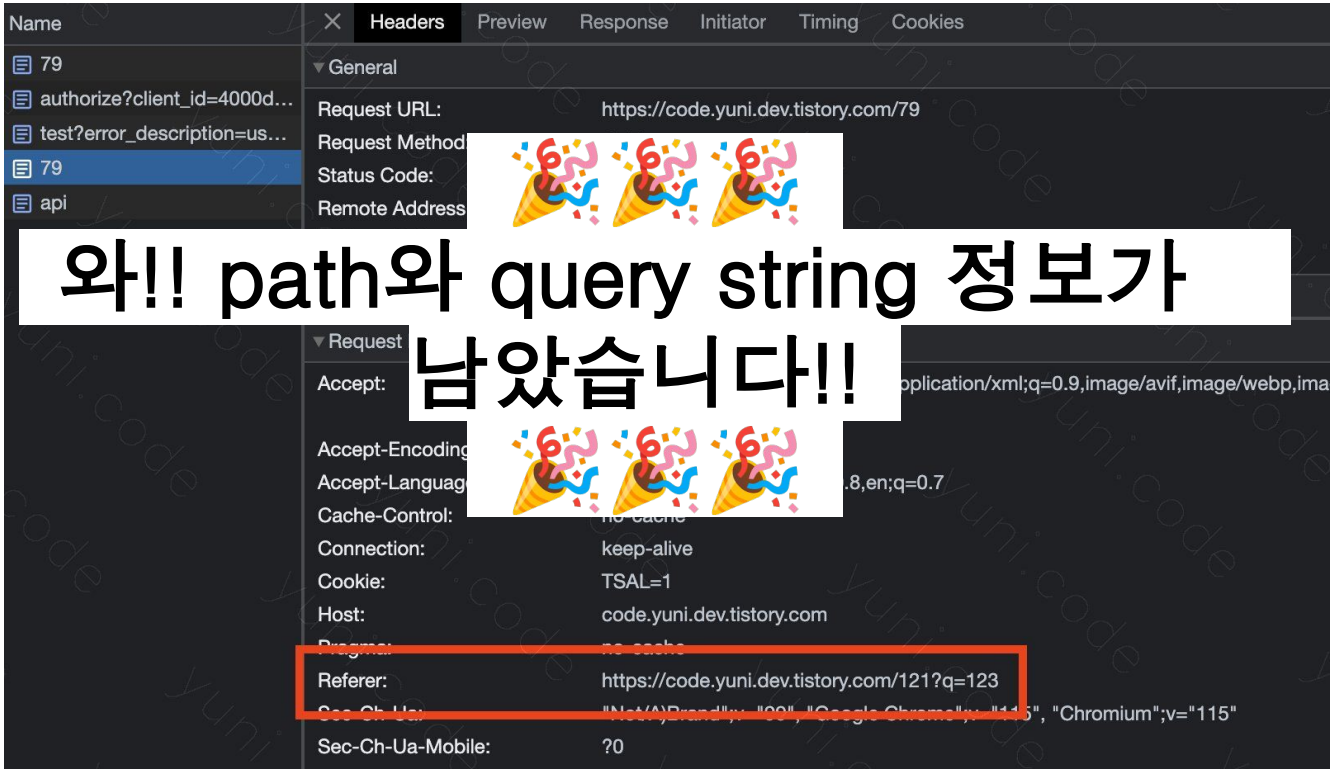
테스트 상황 #3

step 4. 이동하고자 했던 url로 최종 도착



테스트 상황 #3

step 4. 이동하고자 했던 url로 최종 도착



성공은 했지만, 사실 이건 반쪽짜리 테스트

실제 검색유입에서 들어오는 요청의 경우는
userblog에서 userblog로 요청하는 게 아니라
daum이나 naver 등 **다른 도메인에서 들어오는 요청**이기 때문

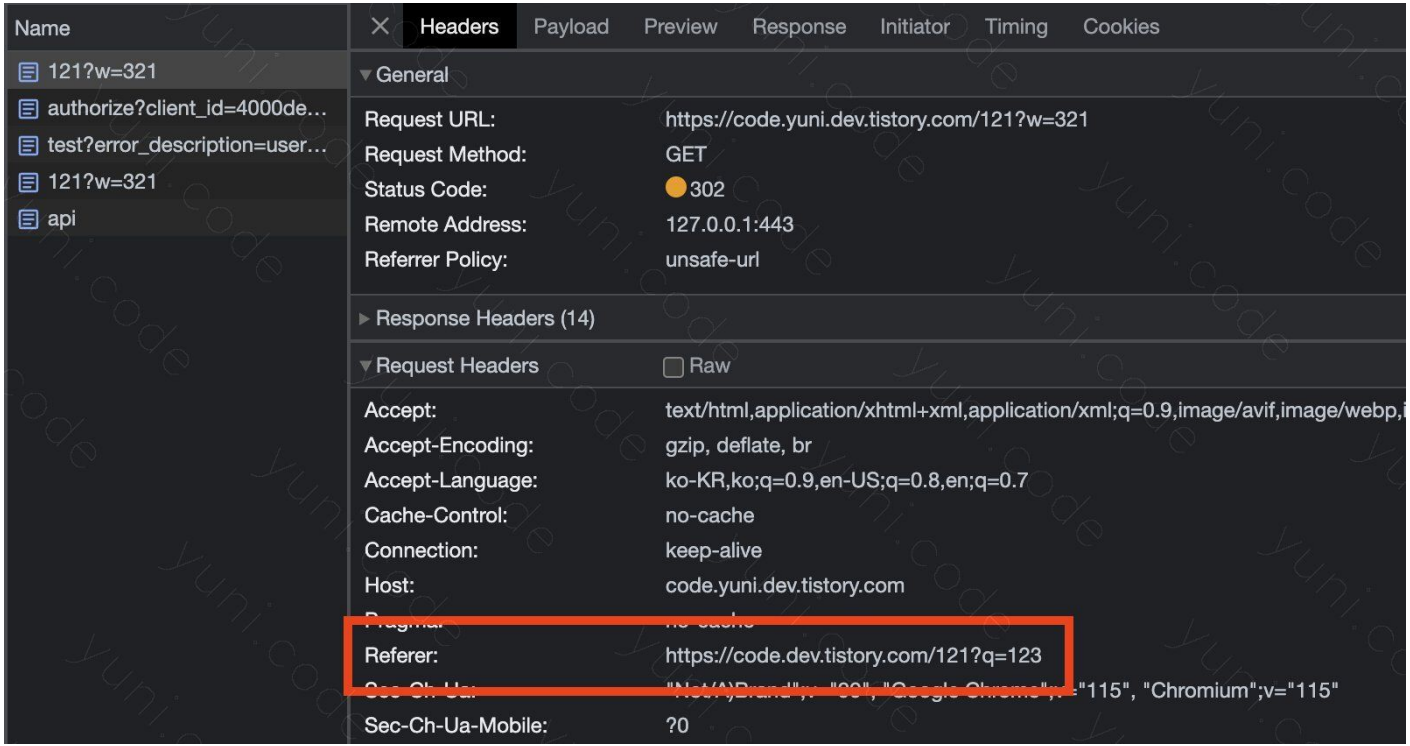
테스트 상황 #4

step 0. 최초 상황

이번에는 dev 도메인 121번 게시글에서, local 도메인 79번 게시글로 요청하는 상황으로 테스트.

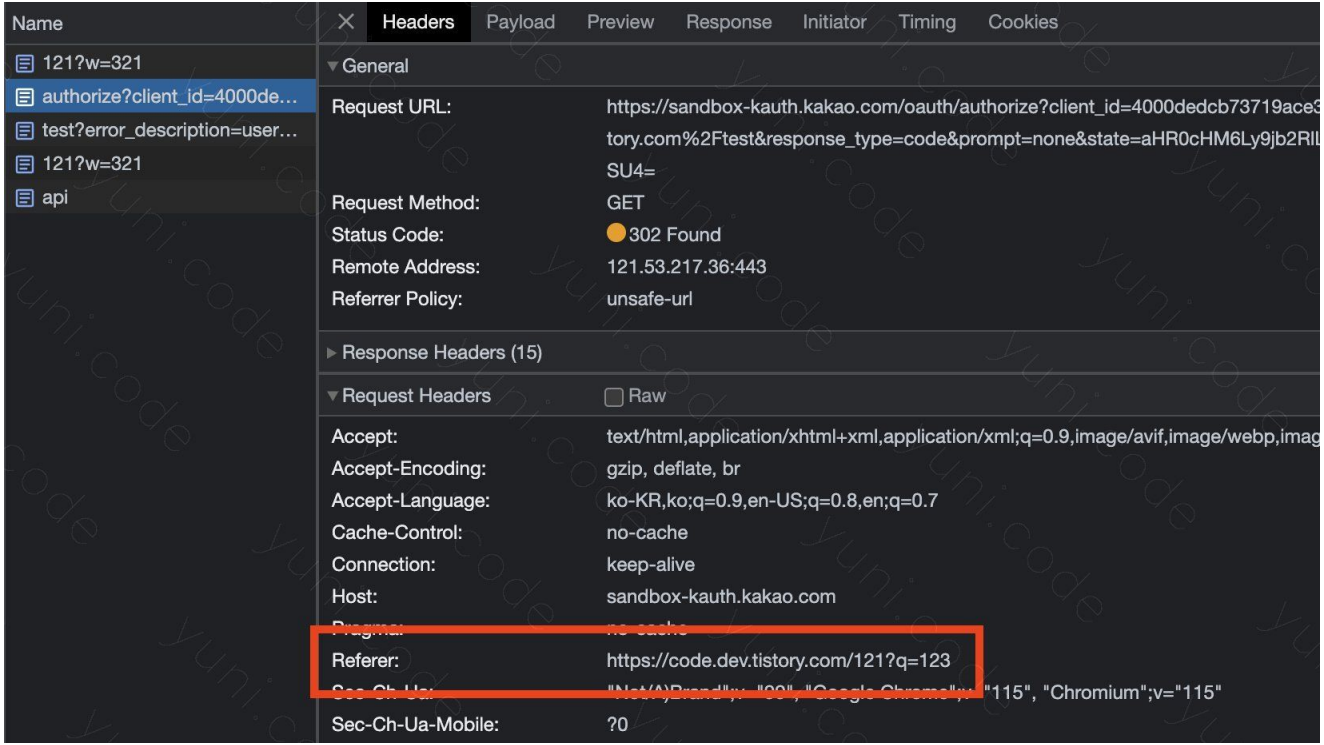
테스트 상황 #4

step 1. local로 향하는 url 클릭



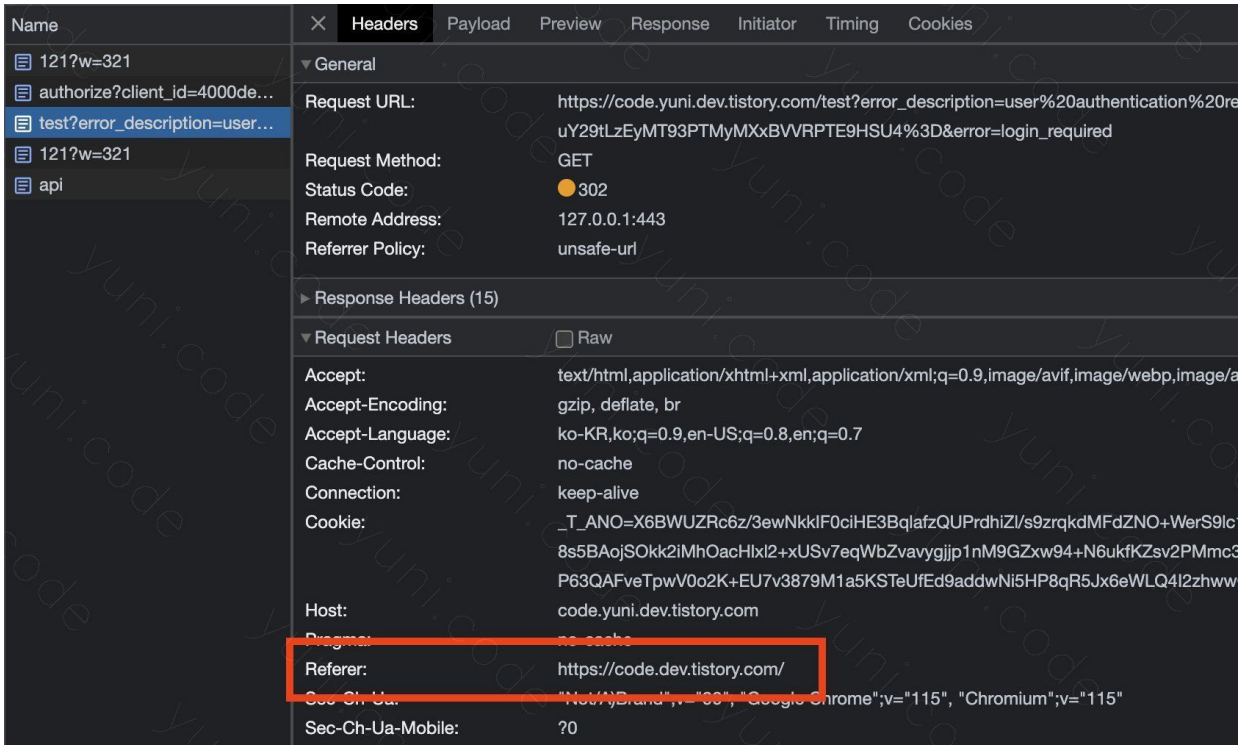
테스트 상황 #4

step 2. kauth로 로그인 요청



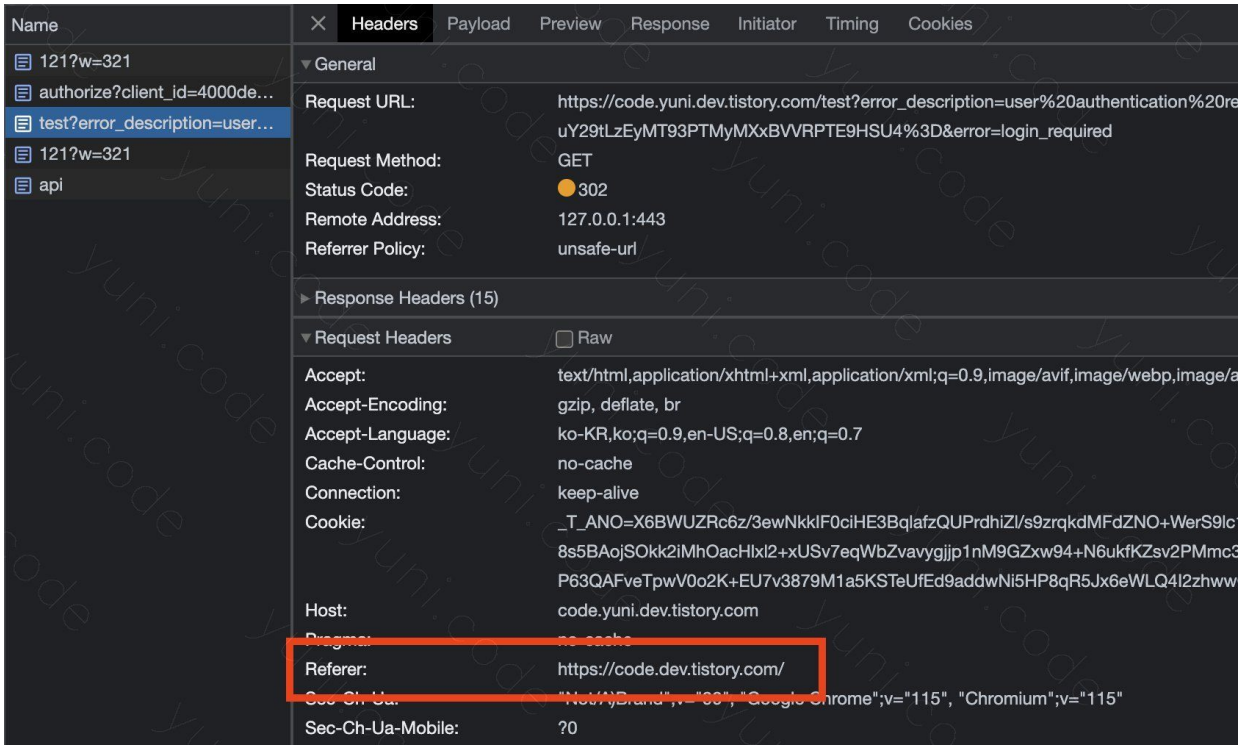
테스트 상황 #4

step 3. /test callback



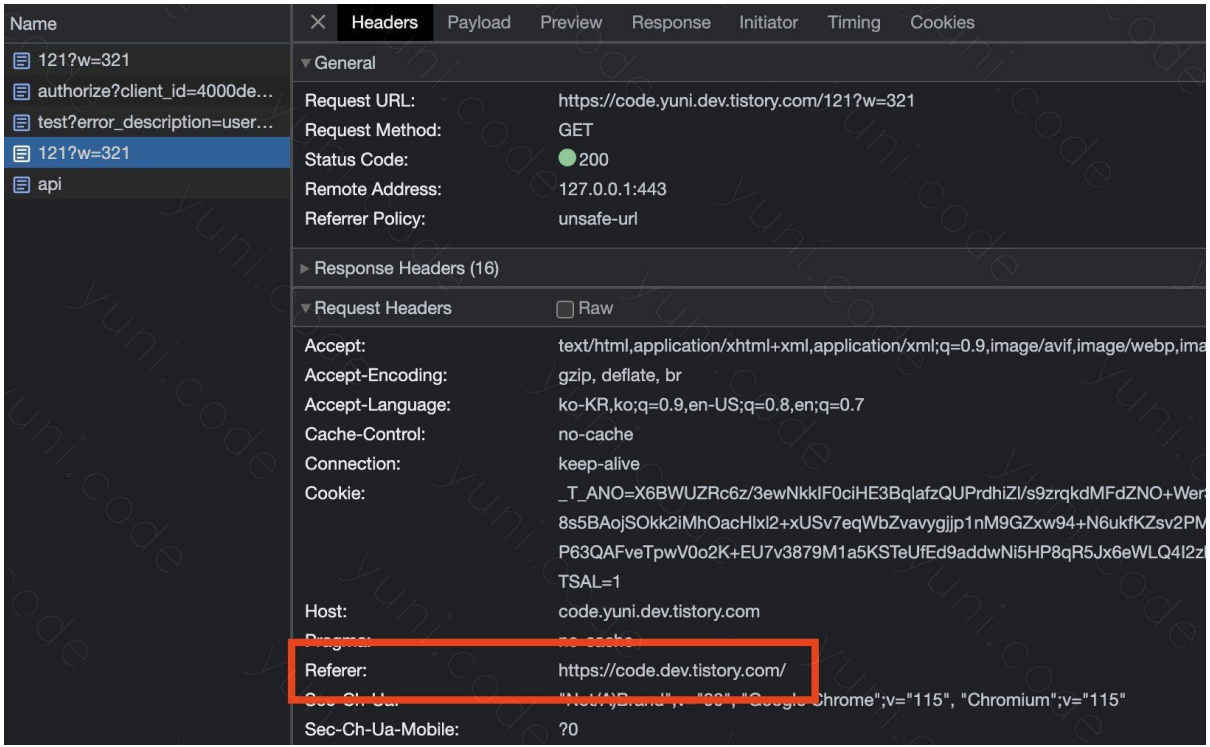
테스트 상황 #4

step 3. /test callback



테스트 상황 #4

step 4. 이동하고자 했던 url로 최종 도착



왜??? 이번엔 안 됐을까?

그 이유는 바로 Referrer-Policy

Referrer-Policy

The `Referrer-Policy` HTTP header controls how much referrer information (sent with the `Referer` header) should be included with requests. Aside from the HTTP header, you can set this policy in HTML.

Header type	Response header
Forbidden header name	no

Syntax

HTTP

```
Referrer-Policy: no-referrer
Referrer-Policy: no-referrer-when-downgrade
Referrer-Policy: origin
Referrer-Policy: origin-when-cross-origin
Referrer-Policy: same-origin
Referrer-Policy: strict-origin
Referrer-Policy: strict-origin-when-cross-origin
Referrer-Policy: unsafe-url
```

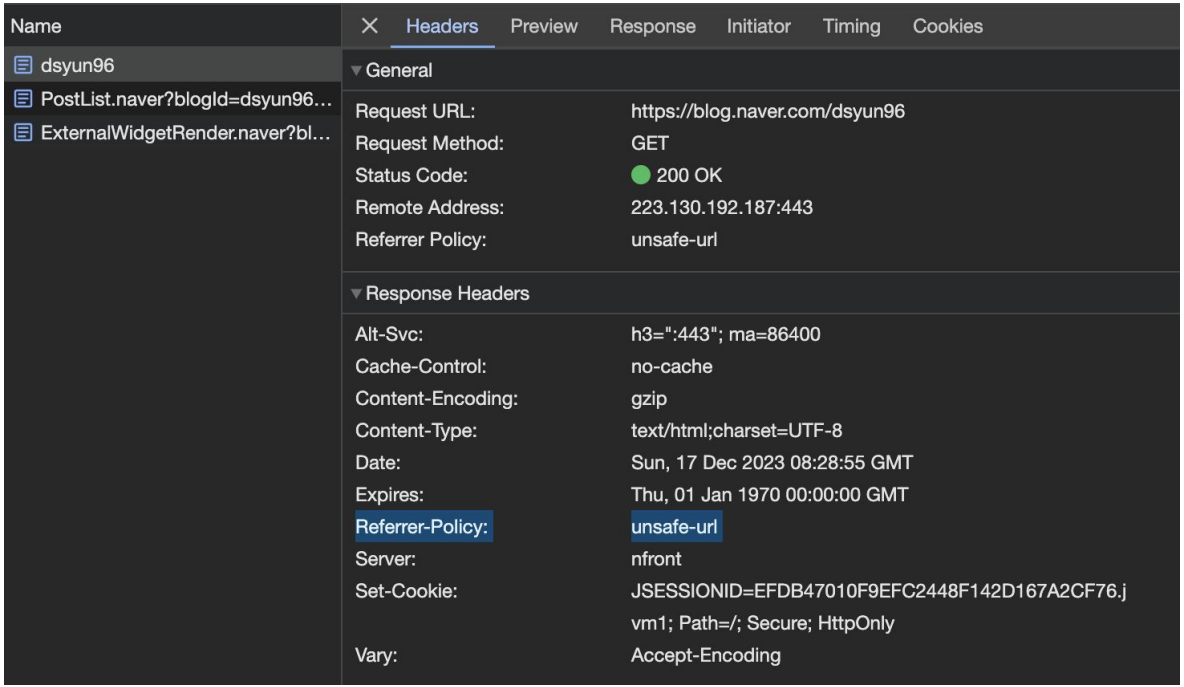

private info
private + sensitive
private + identifying
security-critical
#1 https://music.example/superArtist123
#2 https://news.example/fr/search/?q=science&sort=latest
#3 https://health-blog.example/path/sports-injuries/knee
#4 https://social-network.example/my-account/johndoe86
#5 https://social-network.example/email_verified/?email=johndoe86@gmail.com
#6 https://cloud-storage.example/625x1710s7Gtsr/password_reset

Referrer Policy는 어떻게 적용할까?

- HTTP Header
- HTML
- Javascript

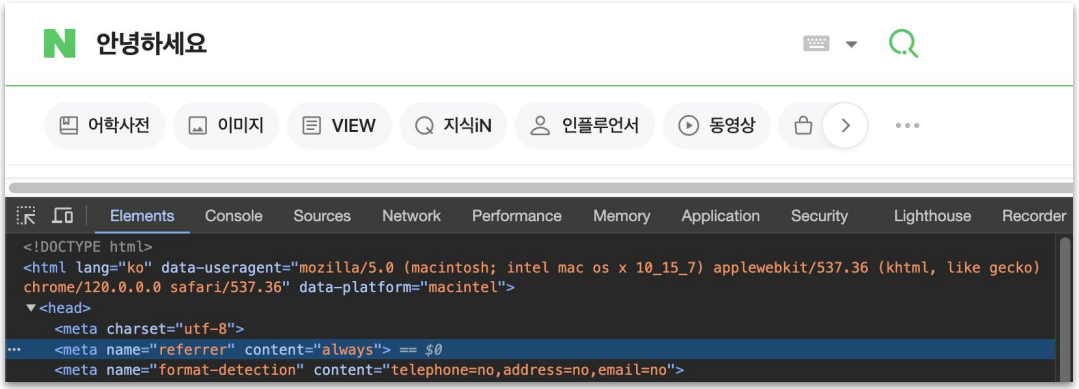
Referrer Policy는 어떻게 적용할까?

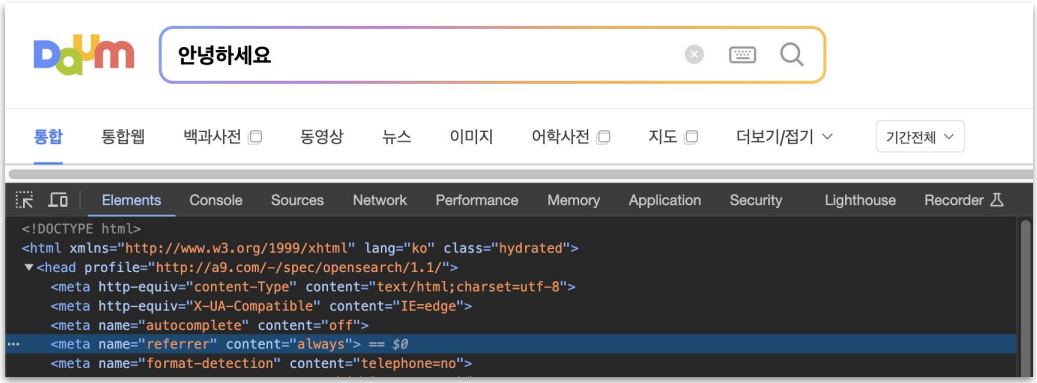
- HTTP Header
- HTML
- Javascript



Referrer Policy는 어떻게 적용할까?

- HTTP Header
- HTML
- Javascript



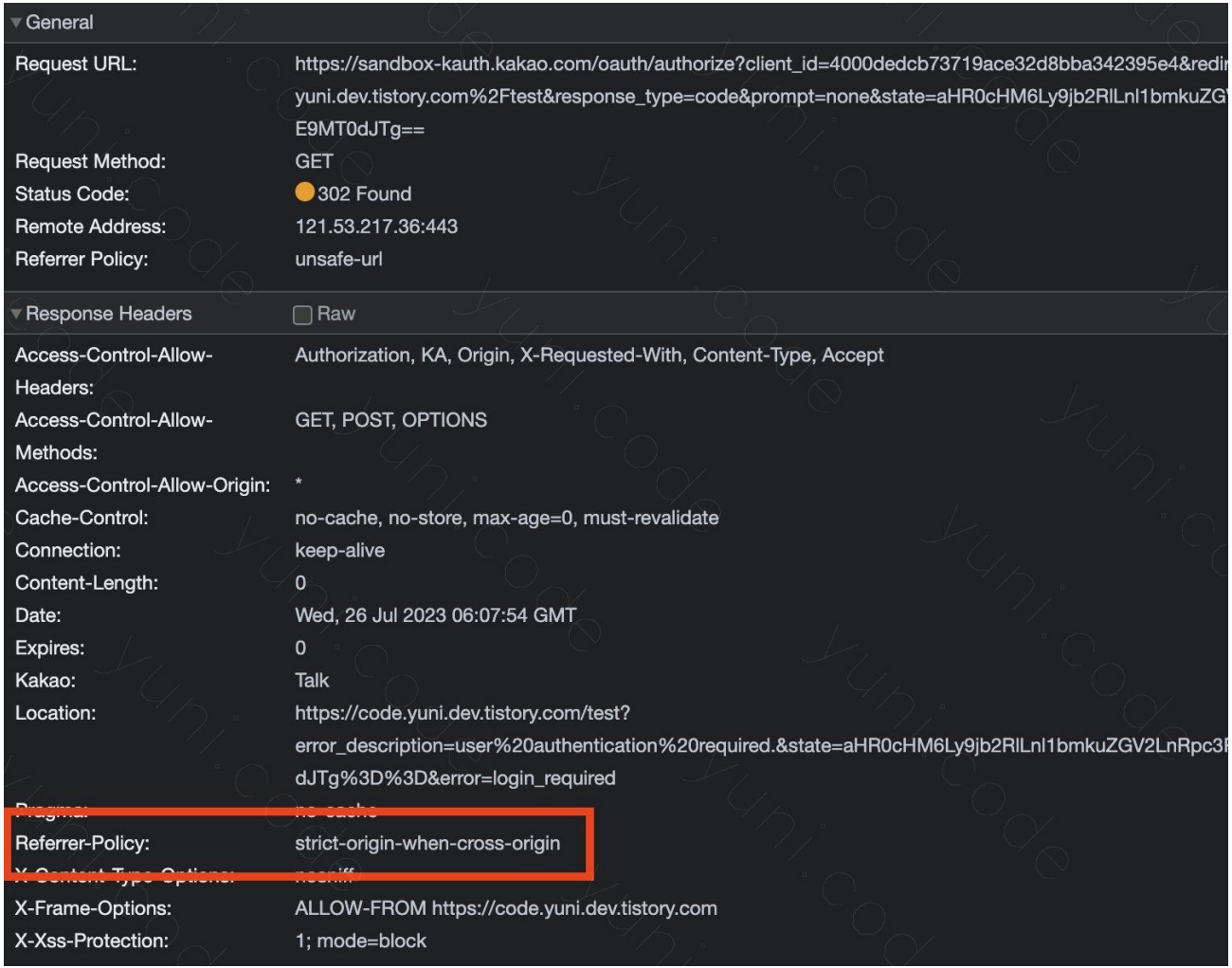


Referrer Policy는 어떻게 적용할까?

- HTTP Header
- HTML
- Javascript

Referrer Policy 적용 우선순위

1. element 레벨 (e.g.)
2. page 레벨 (HTTP Header, meta tag)
3. 브라우저 기본값



▼General

Request URL:	https://sandbox-kauth.kakao.com/oauth/authorize?client_id=4000dedcb73719ace32d8bba342395e4&redir yuni.dev.tistory.com%2Ftest&response_type=code&prompt=none&state=aHR0cHM6Ly9jb2RlLnl1bmkuZG E9MT0dJTg==
Request Method:	GET
Status Code:	302 Found
Remote Address:	121.53.217.36:443
Referrer Policy:	unsafe-url

▼Response Headers ☐ Raw

Access-Control-Allow-Headers:	Authorization, KA, Origin, X-Requested-With, Content-Type, Accept
Access-Control-Allow-Methods:	GET, POST, OPTIONS
Access-Control-Allow-Origin:	*
Cache-Control:	no-cache, no-store, max-age=0, must-revalidate
Connection:	keep-alive
Content-Length:	0
Date:	Wed, 26 Jul 2023 06:07:54 GMT
Expires:	0
Kakao:	Talk
Location:	https://code.yuni.dev.tistory.com/test? error_description=user%20authentication%20required.&state=aHR0cHM6Ly9jb2RlLnl1bmkuZGV2LnRpc3 dJTg%3D%3D&error=login_required
Pragma:	no-cache
Referrer-Policy:	strict-origin-when-cross-origin
X-Content-Type-Options:	nosniff
X-Frame-Options:	ALLOW-FROM https://code.yuni.dev.tistory.com
X-Xss-Protection:	1; mode=block

	No Data	Origin Only	Full URL
`no-referrer`	✓		
`origin`		✓	
`unsafe-url`			✓
`strict-origin`	HTTPS → HTTP	HTTPS → HTTPS, HTTP → HTTP	
`no-referrer-when-downgrade`	HTTPS → HTTP		HTTPS → HTTPS, HTTP → HTTP
`origin-when-cross-origin`		`cross-origin`	`same-origin`
`same-origin`	`cross-origin`		`same-origin`
`strict-origin-when-cross-origin`	HTTPS → HTTP	`cross-origin`, HTTPS → HTTPS, HTTP → HTTP	

cross-origin이란?

origin은 scheme(protocol), hostname, port의 조합을 의미
- scheme : https://
- hostname : www.tistory.com
- port : 443

https://yceffort.kr:443 를 기준으로 비교했을 때,

Origin	비교결과	이유
https://fake.kr:443	`cross-origin`	도메인이 다르다.
https://www.yceffort.kr:443	`cross-origin`	서브도메인이 다르다.
https://blog.yceffort.kr:443	`cross-origin`	서브도메인이 다르다.
http://yceffort.kr:443	`cross-origin`	scheme이 다르다.
http://yceffort.kr:80	`cross-origin`	port가 다르다.
https://yceffort.kr:443	`same-origin`	완전히 같다.
https://yceffort.kr	`same-origin`	포트가 없지만, https의 기본포트 443이 있다고 간주한다.

필요한 정보는 모두 모였으니
파헤쳐 봅시다!

브런치에서 referer의 query string이 잘 남았던 이유

brunch → kauth → brunch

브런치에서 referer의 query string이 잘 남았던 이유

brunch → kauth → brunch

kauth에 요청할 때 brunch.kr 이라는 referer 정보와,
kauth가 다시 redirect로 요청을 보내야 하는 brunch.kr 이same-origin 이라서,
full url을 제공받을 수 있다.

티스토리에서 referer의 query string이 유실됐던 이유

userblog → kauth → TOP → userblog

티스토리에서 referer의 query string이 유실됐던 이유

userblog → kauth → TOP → userblog

kauth에 요청할 때 yun.tistory.com 이라는 referer 정보와,
kauth가 다시 redirect로 요청을 보내야 하는 www.tistory.com 이cross-origin 이라서,
origin 정보만 제공받을 수 있다.

티스토리에서 referer의 query string이 유실됐던 이유

userblog → kauth → TOP → userblog

kauth에 요청할 때 yun.tistory.com 이라는 referer 정보와,
kauth가 다시 redirect로 요청을 보내야 하는 www.tistory.com 이cross-origin 이라서,
origin 정보만 제공받을 수 있다.

userblog → kauth → userblog

이 경우는 same-origin 이므로, full url을 제공받을 수 있다.

하지만 실제 플로우는

브런치의 경우 :
daum/naver → brunch → kauth → brunch

티스토리의 경우 :
daum/naver → userblog → kauth → TOP → userblog

하지만 실제 플로우는

브런치의 경우 :
daum/naver → brunch → kauth → brunch

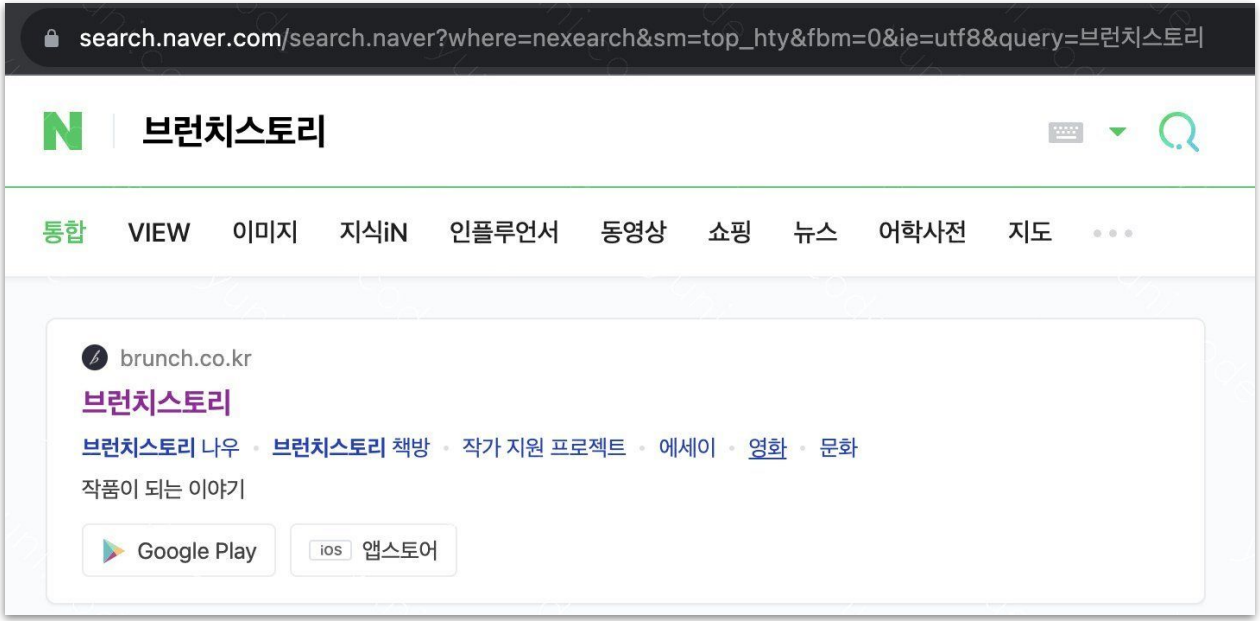
티스토리의 경우 :
daum/naver → userblog → kauth → TOP → userblog

kauth에 들어오는 요청의 referer 정보는 daum/naver 이므로,
무조건 cross-origin 이 된다.

테스트... 해봐야겠지?

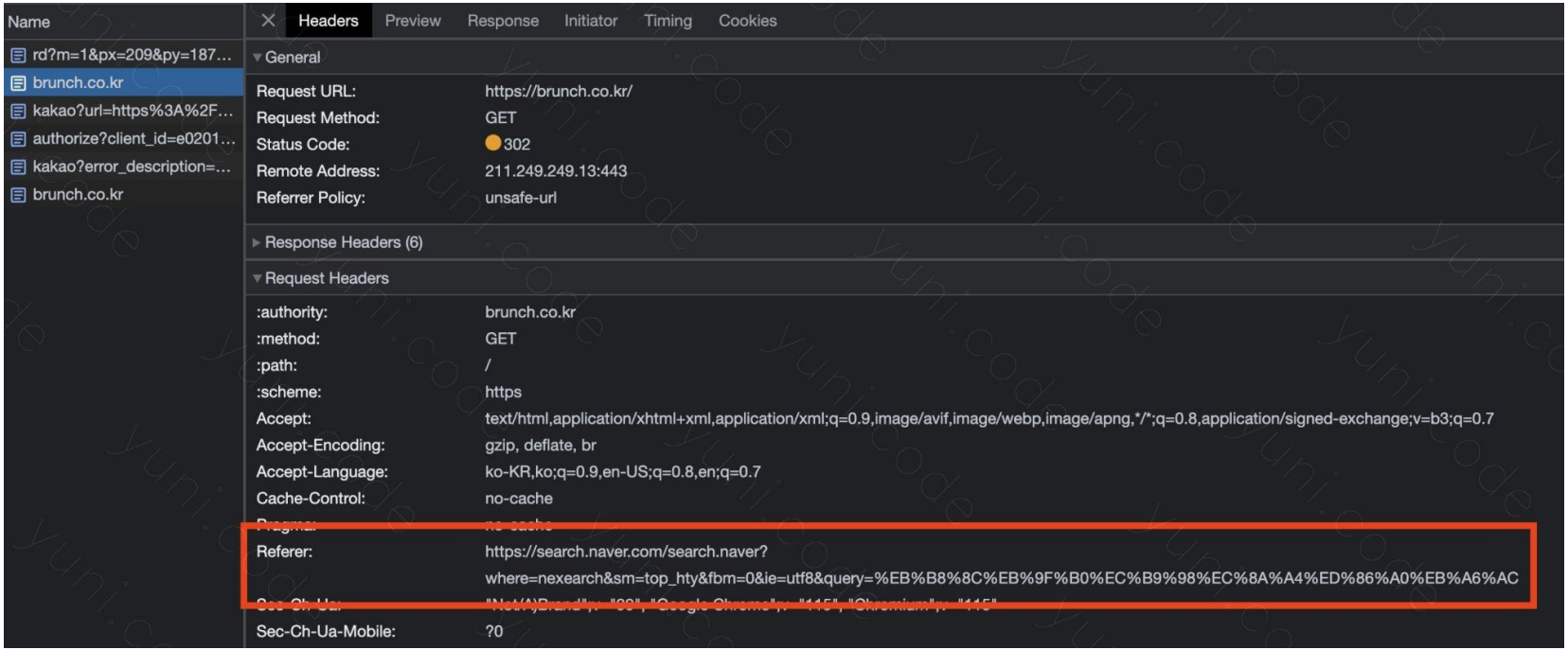
테스트 상황 #5

step 0. 최초 상황



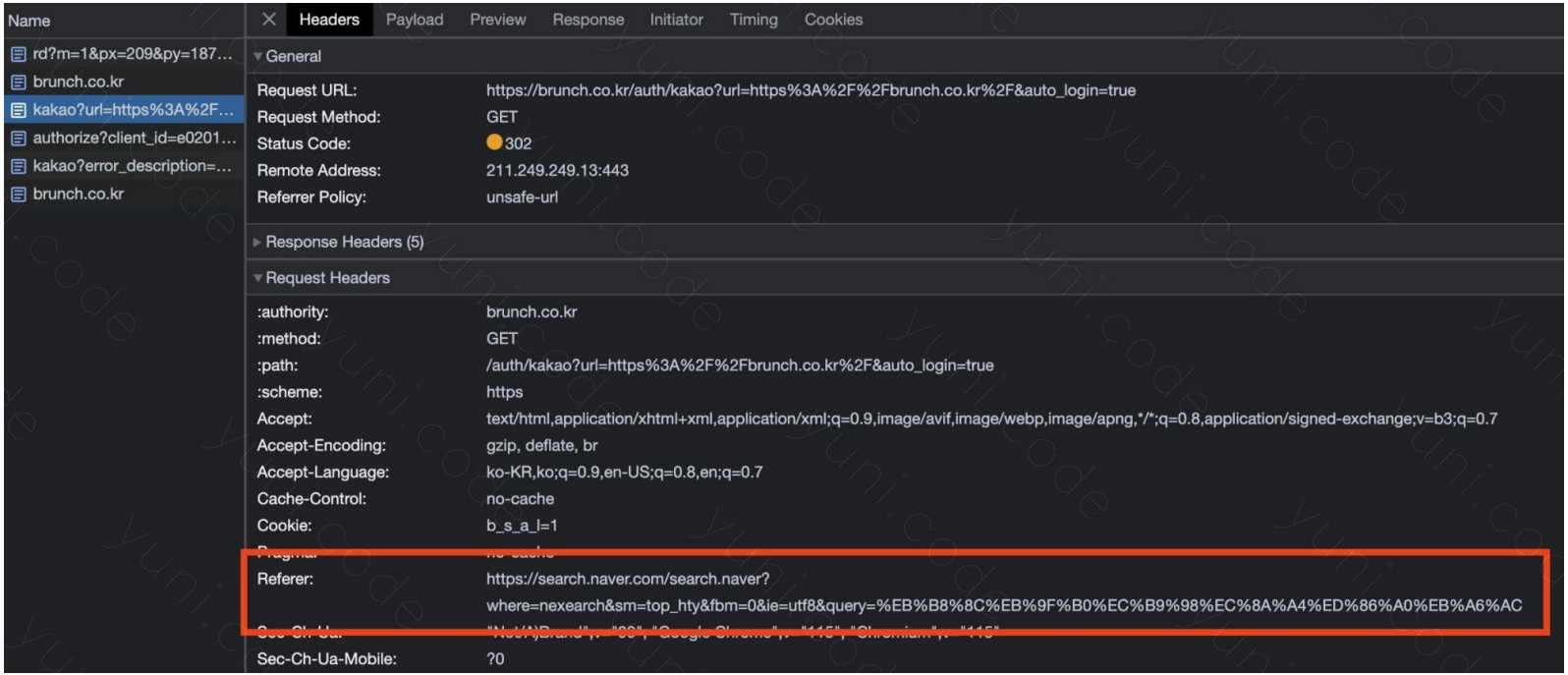
테스트 상황 #5

step 1. 브런치스토리로 접근



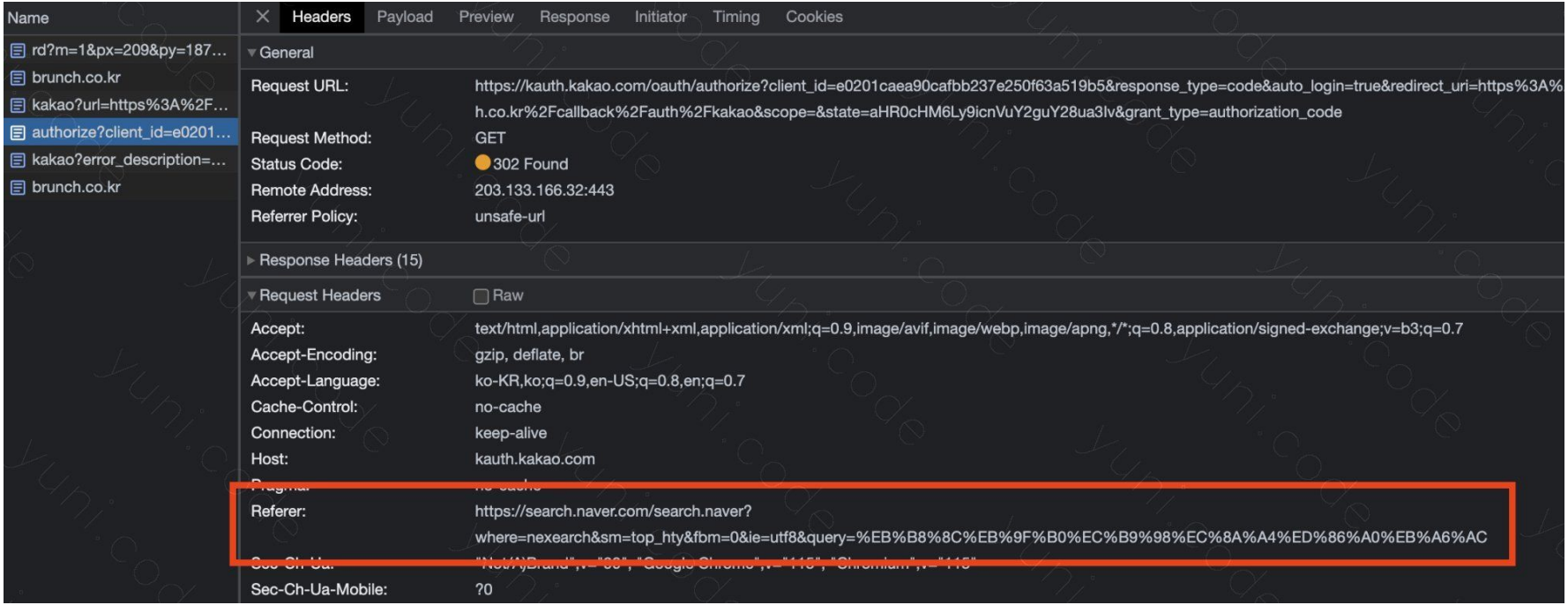
테스트 상황 #5

step 2. 브런치스토리의 auth api로 redirect



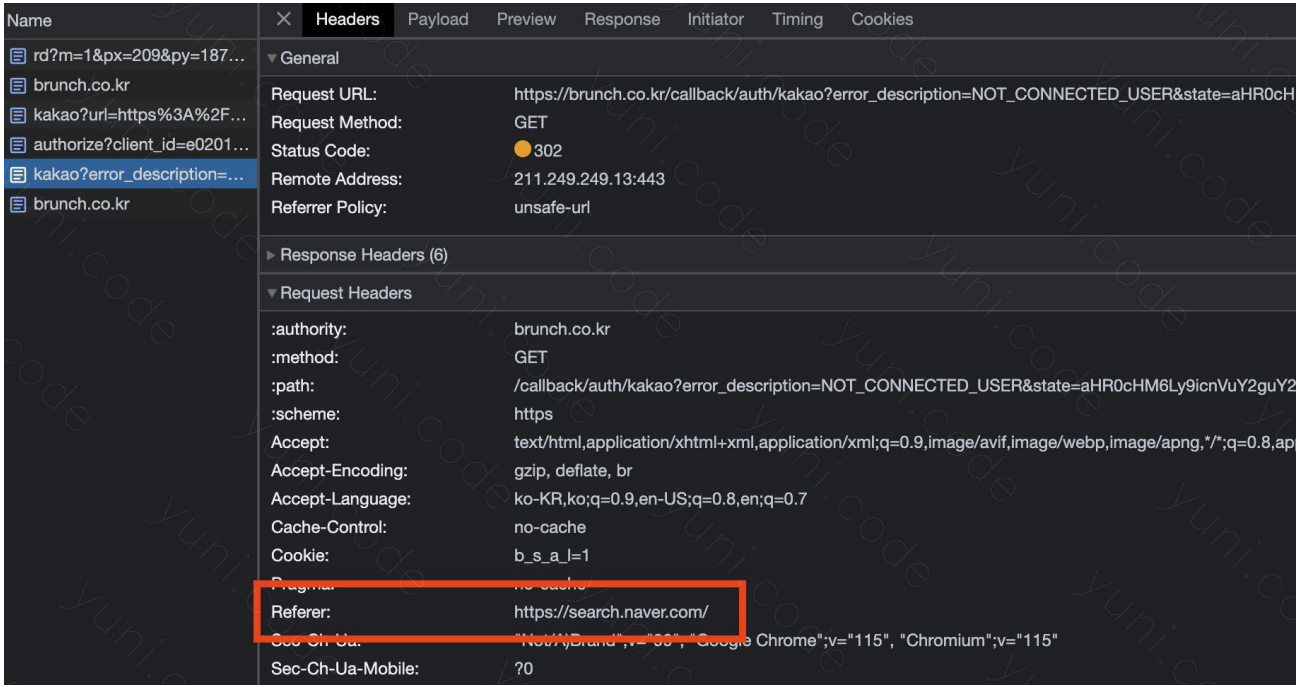
테스트 상황 #5

step 3. kauth로 로그인 요청



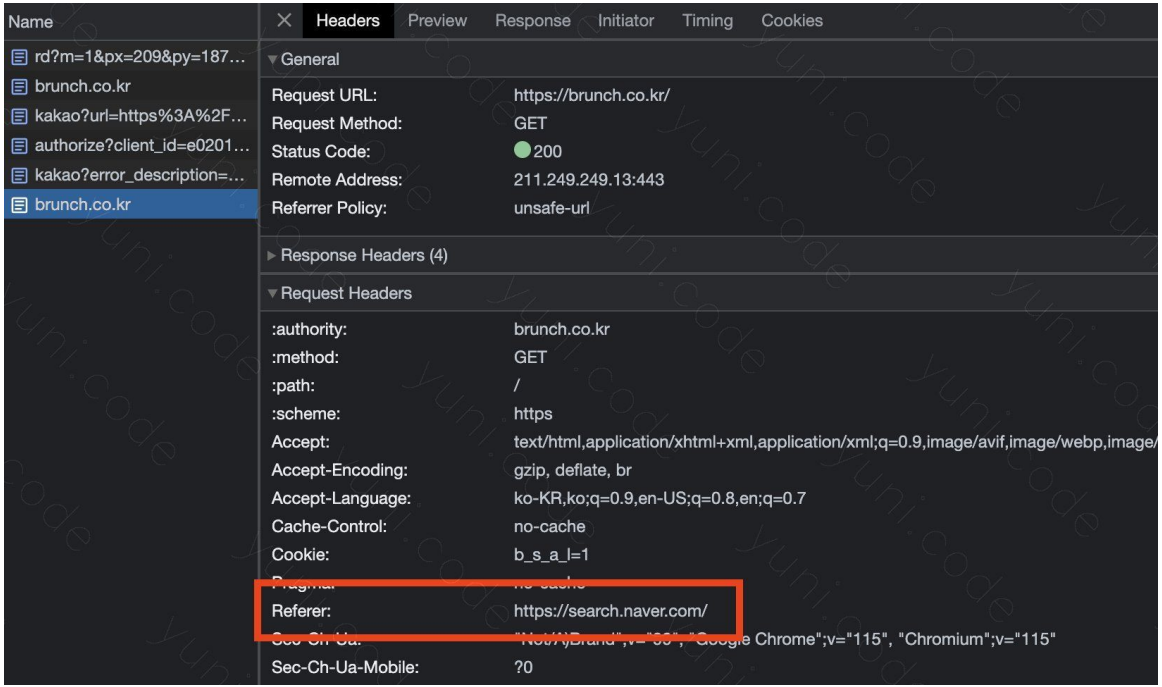
테스트 상황 #5

step 4. 브런치스토리 callback



테스트 상황 #5

step 5. 이동하고자 했던 url로 최종 도착



증명 완료

그럼 이걸 어떻게 해결해야 하는가?

그럼 이걸 어떻게 해결해야 하는가?

1. kauth 담당 부서에 Referrer-Policy 변경 요청하기

그럼 이걸 어떻게 해결해야 하는가?

1. kauth 담당 부서에 Referrer-Policy 변경 요청하기 → 어떤 사이드이펙트가 생길지 모른다

그럼 이걸 어떻게 해결해야 하는가?

1. kauth 담당 부서에 Referrer-Policy 변경 요청하기 → 어떤 사이드이펙트가 생길지 모른다
2. kauth로 자동로그인 요청을 보낼 때 redirect uri에 query string으로 세팅하고,
   TOP에서 받아서 이를 다시 referer 정보에 세팅하기

그럼 이걸 어떻게 해결해야 하는가?

1. kauth 담당 부서에 Referrer-Policy 변경 요청하기 → 어떤 사이드이펙트가 생길지 모른다
2. kauth로 자동로그인 요청을 보낼 때 redirect uri에 query string으로 세팅하고,
   TOP에서 받아서 이를 다시 referer 정보에 세팅하기 → 가능할 것으로 보임 (PoC 필요, 아직 안 해봄)

E.O.D.

참고 자료

https://yceffort.kr/2020/09/referer-and-referrer-policy