

Lab2_b_Be_A_Man_A

Lab 2B-Honors: Origin-Driven Caching

Goal

Implement safe caching for dynamic web/API where:

If origin responds with Cache-Control: public, s-maxage=30 → CloudFront caches 30 seconds

If origin responds with Cache-Control: private, no-store (or no cache-control) → CloudFront does not cache

Cache key includes only what truly varies the response

Students prove correctness using:

x-cache (Hit from cloudfront, Miss from cloudfont)

Age header behavior

and app-visible evidence

What students add to the app (EC2)

Create two endpoints:

1) Public endpoint (cacheable)

GET /api/public-feed

Response must include:

Cache-Control: public, s-maxage=30, max-age=0

s-maxage is for shared caches (CDNs).

Example behavior:

returns "server_time_utc" and "message of the minute"

should change every request at origin, but CloudFront will hold it for 30 seconds

```
nightwolf@nightwolf-Inspiron-7786:~$ curl -v https://larryharrisaws.com/api/public-feed
```

* Host larryharrisaws.com:443 was resolved.

* IPv6: (none)

* IPv4: 18.165.53.33, 18.165.53.121, 18.165.53.56, 18.165.53.63

* Trying 18.165.53.33:443...

* GnuTLS priority: NORMAL:-ARCFOUR-128:-CTYPE-ALL:+CTYPE-X509:-VERS-SSL3.0

* ALPN: curl offers h2,http/1.1

* found 146 certificates in /etc/ssl/certs/ca-certificates.crt

* found 441 certificates in /etc/ssl/certs

* SSL connection using TLS1.3 / ECDHE_RSA_AES_128_GCM_SHA256

* server certificate verification OK

* server certificate status verification SKIPPED

* common name: app.larryharrisaws.com (matched)

* server certificate expiration date OK

* server certificate activation date OK

* certificate public key: RSA

* certificate version: #3

* subject: CN=app.larryharrisaws.com

* start date: Fri, 23 Jan 2026 00:00:00 GMT

* expire date: Sun, 21 Feb 2027 23:59:59 GMT

* issuer: C=US,O=Amazon,CN=Amazon RSA 2048 M04

* ALPN: server accepted h2

```
* Connected to larryharrisaws.com (18.165.53.33) port 443
* using HTTP/2
* [HTTP/2] [1] OPENED stream for https://larryharrisaws.com/api/public-feed
* [HTTP/2] [1] [:method: GET]
* [HTTP/2] [1] [:scheme: https]
* [HTTP/2] [1] [:authority: larryharrisaws.com]
* [HTTP/2] [1] [:path: /api/public-feed]
* [HTTP/2] [1] [user-agent: curl/8.13.0]
* [HTTP/2] [1] [accept: */*]
> GET /api/public-feed HTTP/2
> Host: larryharrisaws.com
> User-Agent: curl/8.13.0
> Accept: */*
>
* Request completely sent off
< HTTP/2 200
< content-type: application/json
< content-length: 213
< date: Fri, 23 Jan 2026 12:56:08 GMT
< cache-control: public, s-maxage=30, max-age=0
< server: Werkzeug/3.1.5 Python/3.9.25
< x-cache: Hit from cloudfront
< via: 1.1 55f299ae81265c35cceef8f686166d6c.cloudfront.net (CloudFront)
< x-amz-cf-pop: TPA52-P2
< x-amz-cf-id: F-pzHfBheBU4Y6XPHemcaMrgBkY1bzLqEZEmssCiTL5KeqvsUDxUGg==
< age: 4
<
{"message_of_the_minute":"Message of the minute: 56 - Fresh from origin at request time!","note":"This should be cached by CloudFront for 30 seconds (s-maxage=30)","server_time_utc":"2026-01-23T12:56:08.435550Z"}
* Connection #0 to host larryharrisaws.com left intact
nightwolf@nightwolf-Inspiron-7786:~$ curl -v https://larryharrisaws.com/api/public-feed
* Host larryharrisaws.com:443 was resolved.
* IPv6: (none)
* IPv4: 18.165.53.63, 18.165.53.56, 18.165.53.121, 18.165.53.33
* Trying 18.165.53.63:443...
* GnuTLS priority: NORMAL:-ARCFOUR-128:-CTYPE-ALL:+CTYPE-X509:-VERS-SSL3.0
* ALPN: curl offers h2,http/1.1
* found 146 certificates in /etc/ssl/certs/ca-certificates.crt
* found 441 certificates in /etc/ssl/certs
* SSL connection using TLS1.3 / ECDHE_RSA_AES_128_GCM_SHA256
* server certificate verification OK
* server certificate status verification SKIPPED
* common name: app.larryharrisaws.com (matched)
* server certificate expiration date OK
* server certificate activation date OK
* certificate public key: RSA
* certificate version: #3
* subject: CN=app.larryharrisaws.com
* start date: Fri, 23 Jan 2026 00:00:00 GMT
```

```

* expire date: Sun, 21 Feb 2027 23:59:59 GMT
* issuer: C=US,O=Amazon,CN=Amazon RSA 2048 M04
* ALPN: server accepted h2
* Connected to larrryharrisaws.com (18.165.53.63) port 443
* using HTTP/2
* [HTTP/2] [1] OPENED stream for https://larrryharrisaws.com/api/public-feed
* [HTTP/2] [1] [:method: GET]
* [HTTP/2] [1] [:scheme: https]
* [HTTP/2] [1] [:authority: larrryharrisaws.com]
* [HTTP/2] [1] [:path: /api/public-feed]
* [HTTP/2] [1] [user-agent: curl/8.13.0]
* [HTTP/2] [1] [accept: */*]
> GET /api/public-feed HTTP/2
> Host: larrryharrisaws.com
> User-Agent: curl/8.13.0
> Accept: */*
>
* Request completely sent off
< HTTP/2 200
< content-type: application/json
< content-length: 213
< date: Fri, 23 Jan 2026 12:56:08 GMT
< cache-control: public, s-maxage=30, max-age=0
< server: Werkzeug/3.1.5 Python/3.9.25
< x-cache: Hit from cloudfront
< via: 1.1 ad2384fd6c429e93b8552aa37936afb4.cloudfront.net (CloudFront)
< x-amz-cf-pop: TPA52-P2
< x-amz-cf-id: Sf4q6uBDzT70xCEdYpKePoFEoiQnREbP5S4f80BzekmwzJ7IS3Pg0w==
< age: 11
<
{"message_of_the_minute":"Message of the minute: 56 - Fresh from origin at request time!","note":"This should be cached by CloudFront for 30 seconds (s-maxage=30)","server_time_utc":"2026-01-23T12:56:08.435550Z"}
* Connection #0 to host larrryharrisaws.com left intact
nightwolf@nightwolf-Inspiron-7786:~$
```

2) Private endpoint (never cache)

GET /api/list (or /api/user-feed)

Response must include:

Cache-Control: private, no-store

This prevents user mixups and stale reads.

```

nightwolf@nightwolf-Inspiron-7786: ~
[{"id":1,"note":"Brotherhood"}]
* Connection #0 to host larrryharrisaws.com left intact
nightwolf@nightwolf-Inspiron-7786: $ curl -v https://larrryharrisaws.com/api/list
* Host larrryharrisaws.com:443 was resolved.
* IPv6: (none)
* IPv4: 18.165.32.79, 18.165.32.107, 18.165.32.13, 18.165.32.110
* Trying 18.165.32.79:443...
* GnuTLS priority: NORMAL:-ARCFOUR-128:-CTYPE-ALL:+CTYPE-X509:-VERS-SSL3.0
* ALPN: curl offers h2,http/1.1
* found 146 certificates in /etc/ssl/certs/ca-certificates.crt
* found 441 certificates in /etc/ssl/certs
* SSL connection using TLS1.3 / ECDHE_RSA_AES_128_GCM_SHA256
* server certificate verification OK
* server certificate status verification SKIPPED
* common name: app.larrryharrisaws.com (matched)
* server certificate expiration date OK
* server certificate activation date OK
* certificate public key: RSA
* certificate version: #3
* subject: CN=app.larrryharrisaws.com
* start date: Fri, 23 Jan 2026 00:00:00 GMT
* expire date: Sun, 21 Feb 2027 23:59:59 GMT
* issuer: C=US,O=Amazon,CN=Amazon RSA 2048 M04
* ALPN: server accepted h2
* Connected to larrryharrisaws.com (18.165.32.79) port 443
* using HTTP/2
* [HTTP/2] [1] OPENED stream for https://larrryharrisaws.com/api/list
* [HTTP/2] [1] [:method: GET]
* [HTTP/2] [1] [:scheme: https]
* [HTTP/2] [1] [:authority: larrryharrisaws.com]
* [HTTP/2] [1] [:path: /api/list]
* [HTTP/2] [1] [user-agent: curl/8.13.0]
* [HTTP/2] [1] [accept: */*]
> GET /api/list HTTP/2
> Host: larrryharrisaws.com
> User-Agent: curl/8.13.0
> Accept: */
>
* Request completely sent off
< HTTP/2 200
< content-type: application/json
< content-length: 32
< date: Fri, 23 Jan 2026 14:13:00 GMT
< expires: 0
< server: Werkzeug/3.1.5 Python/3.9.25
< cache-control: private, no-store
< pragma: no-cache
< x-cache: Miss from cloudfront
< via: 1.1 266ddfb454a8f82cf215fe4e8561c9e.cloudfront.net (CloudFront)
< x-amz-cf-pop: TPAS2-P1
< x-amz-cf-id: Ek7AotXacSMLjJdn6Z_pn62fQZJV9QvoZ_jn4LA11qNq9lVYk2QRUA==
<
[{"id":1,"note":"Brotherhood"}]
* Connection #0 to host larrryharrisaws.com left intact
nightwolf@nightwolf-Inspiron-7786: $ 

```

Terraform: Honors Overlay

1) Reference AWS managed policies (data sources)

Use Terraform data sources to grab managed cache policies by name. HashiCorp documents the data source pattern.

Terraform Registry

Create lab2b_honors_origin_driven.tf:

2) Patch CloudFront behaviors (the Honors behavior matrix)

A) /api/public-feed = origin-driven caching

Use UseOriginCacheControlHeaders (or QueryStrings variant if needed).

B) /api/* = still safe default (no caching)

Keep your earlier “API caching disabled” policy for everything else.

C) /static/* remains aggressive

No change from Lab 2B baseline.

Honors Verification (students must prove origin-driven caching)

1) Prove CloudFront is honoring origin Cache-Control

A) First request should be MISS

```
curl -i https://chewbacca-growl.com/api/public-feed | sed -n '1,20p'
```

Check headers:

Cache-Control: public, s-maxage=30, max-age=0 (from origin)

x-cache: Miss from cloudfront (or similar)

Age: likely absent or 0

x-cache meanings are documented by AWS

```
nightwolf@nightwolf-Inspiron-7786:~$ curl -i https://larryharrisaws.com/api/public-feed | sed -n '1,20p'
% Total    % Received % Xferd  Average Speed   Time   Time  Current
                                         Dload  Upload   Total Spent   Left  Speed
100  213  100  213    0     0   514      0 --:--:-- --:--:-- 513
HTTP/2 200
Content-Type: application/json
Content-Length: 213
Date: Fri, 23 Jan 2026 14:16:51 GMT
Cache-Control: public, s-maxage=30, max-age=0
Server: Werkzeug/3.1.5 Python/3.9.25
X-Cache: Miss from cloudfront
Via: 1.1 78306659e4792bff990acc2996d36c6c.cloudfront.net (CloudFront)
X-Amz-Cf-Pop: TPA52-P1
X-Amz-Cf-Id: rafRpz_ld_kzw6R0yeeLoWjutBgoU-A6K7mmDBtYXoFkxBF0rLLPg==

{"message_of_the_minute": "Message of the minute: 16 - Fresh from origin at request time!", "note": "This should be cached by CloudFront for 30 seconds (s-maxage = 30)", "server_time_utc": "2026-01-23T14:16:51.817934Z"}
nightwolf@nightwolf-Inspiron-7786:~$
```

B) Second request within 30 seconds should be HIT

```
curl -i https://chewbacca-growl.com/api/public-feed | sed -n '1,20p'
```

Expected:

x-cache: Hit from cloudfront

Age: increases on subsequent hits (cache indicator)

Body should remain identical until TTL expires

```

nightwolf@nightwolf-Inspiron-7786:~$ 
nightwolf@nightwolf-Inspiron-7786:~$ 
nightwolf@nightwolf-Inspiron-7786:~$ 
nightwolf@nightwolf-Inspiron-7786:~$ 
nightwolf@nightwolf-Inspiron-7786:~$ curl -i https://larryharrisaws.com/api/public-feed | sed -n '1,20p'
% Total    % Received % Xferd  Average Speed   Time   Time  Current
          Dload  Upload   Total Spent   Left Speed
100  213  100  213     0      0  553       0 --::--- ::--- ::---  554
HTTP/2 200
content-type: application/json
content-length: 213
date: Fri, 23 Jan 2026 14:18:43 GMT
cache-control: public, s-maxage=30, max-age=0
server: Werkzeug/3.1.5 Python/3.9.25
x-cache: Miss from cloudfront
via: 1.1 bd2bca97eb416ec2db082c16ffc2a508.cloudfront.net (CloudFront)
x-amz-cf-pop: TPA52-P1
x-amz-cf-id: EE6YyIRzhSZ9zm2549fW7F_1a-FPDEDvLH6uw1uu4t3mVQX8T3ihQQ==

{"message_of_the_minute": "Message of the minute: 18 - Fresh from origin at request time!", "note": "This should be cached by CloudFront for 30 seconds (s-maxage =30)", "server_time_utc": "2026-01-23T14:18:43.440572Z"}
nightwolf@nightwolf-Inspiron-7786:~$ curl -i https://larryharrisaws.com/api/public-feed | sed -n '1,20p'
% Total    % Received % Xferd  Average Speed   Time   Time  Current
          Dload  Upload   Total Spent   Left Speed
100  213  100  213     0      0  585       0 --::--- ::--- ::---  586
HTTP/2 200
content-type: application/json
content-length: 213
date: Fri, 23 Jan 2026 14:18:43 GMT
cache-control: public, s-maxage=30, max-age=0
server: Werkzeug/3.1.5 Python/3.9.25
x-cache: Hit from cloudfront
via: 1.1 8a2938ef23315abea01680685781527c.cloudfront.net (CloudFront)
x-amz-cf-pop: TPA52-P1
x-amz-cf-id: pQapcS9rm2DDH8UGVnmKhHL6aaZZQrvKSrxz_JBMqStFtFn5GyHBQ==
age: 2

{"message_of_the_minute": "Message of the minute: 18 - Fresh from origin at request time!", "note": "This should be cached by CloudFront for 30 seconds (s-maxage =30)", "server_time_utc": "2026-01-23T14:18:43.440572Z"}
nightwolf@nightwolf-Inspiron-7786:~$ 

```

C) After 35 seconds, it should MISS again

sleep 35

```
curl -i https://chewbacca-growl.com/api/public-feed | sed -n '1,20p'
```

Expected:

x-cache becomes Miss or RefreshHit

Body updates

```

nightwolf@nightwolf-Inspiron-7786:~$ curl -i https://larryharrisaws.com/api/public-feed | sed -n '1,20p'
% Total    % Received % Xferd  Average Speed   Time   Time  Current
          Dload  Upload   Total Spent   Left Speed
100  213  100  213     0      0  503       0 --::--- ::--- ::---  503
HTTP/2 200
content-type: application/json
content-length: 213
date: Fri, 23 Jan 2026 14:21:29 GMT
cache-control: public, s-maxage=30, max-age=0
server: Werkzeug/3.1.5 Python/3.9.25
x-cache: Miss from cloudfront
via: 1.1 ce2dcbbde1975d88f127beb1f5b330c.cloudfront.net (CloudFront)
x-amz-cf-pop: TPA52-P1
x-amz-cf-id: MotNnfgnjZ1TS8esRWBgXWlc9nWSZ3_OIEguL70SErrwDx3JVkb1TQ==

{"message_of_the_minute": "Message of the minute: 21 - Fresh from origin at request time!", "note": "This should be cached by CloudFront for 30 seconds (s-maxage =30)", "server_time_utc": "2026-01-23T14:21:29.852150Z"}
nightwolf@nightwolf-Inspiron-7786:~$ 

```

2) Prove “no-store” never caches (safety proof)

```
curl -i https://chewbacca-growl.com/api/list | sed -n '1,30p'
curl -i https://chewbacca-growl.com/api/list | sed -n '1,30p'
```

```

nightwolf@nightwolf-Inspiron-7786:~$ curl -i https://larryharrisaws.com/api/list | sed -n '1,20p'
% Total    % Received % Xferd  Average Speed   Time   Time  Current
                                         Dload  Upload   Total   Spent   Left  Speed
100  32  100    32     0      0  54      0  --::-- --::-- --::--  54
HTTP/2 200
content-type: application/json
content-length: 32
date: Fri, 23 Jan 2026 14:22:52 GMT
expires: 0
server: Werkzeug/3.1.5 Python/3.9.25
cache-control: private, no-store
pragma: no-cache
x-cache: Miss from cloudfront
via: 1.1 76cf0264e4b32cd3879f2e935e26d070.cloudfront.net (CloudFront)
x-amz-cf-pop: TPA52-P1
x-amz-cf-id: OoJGw-IedvD7DHq8aeBk2I_7o3jkXxaCpDHxwgfy-Uh7gZNcL3Lvg==
[{"id":1,"note":"Brotherhood"}]
nightwolf@nightwolf-Inspiron-7786:~$ curl -i https://larryharrisaws.com/api/list | sed -n '1,20p'
% Total    % Received % Xferd  Average Speed   Time   Time  Current
                                         Dload  Upload   Total   Spent   Left  Speed
100  32  100    32     0      0  73      0  --::-- --::-- --::--  73
HTTP/2 200
content-type: application/json
content-length: 32
date: Fri, 23 Jan 2026 14:22:54 GMT
expires: 0
server: Werkzeug/3.1.5 Python/3.9.25
cache-control: private, no-store
pragma: no-cache
x-cache: Miss from cloudfront
via: 1.1 1ebc98e9cde3bb3dc3ccb51ea8067d7c.cloudfront.net (CloudFront)
x-amz-cf-pop: TPA52-P1
x-amz-cf-id: a-i4h9Xz87_noKCIrdkB8hiQhkf4wTZ6y5cDb0ulsb5Wdle4C4bWrw==

[{"id":1,"note":"Brotherhood"}]
nightwolf@nightwolf-Inspiron-7786:~$ 

```

Expected:

Cache-Control: private, no-store

No meaningful cache hit behavior (Age not growing / no Hit)

Each request should reflect origin state

If a student gets a cache HIT here, it's a fail (potential data leak).

Honors "Make them sweat" incident challenge

Failure Injection: "Origin forgot Cache-Control"

Remove Cache-Control from /api/public-feed

With the managed origin-driven policy, CloudFront should default to not caching

Students must observe:

no Hit from cloudfront

increased origin load

Fix: restore proper Cache-Control

Failure Injection: "Cache fragmentation"

Forward User-Agent / all headers into cache key (bad)

Hit ratio tanks

Fix: whitelist headers; CloudFront warns about forwarding unnecessary headers hurting hit ratio

Student submission checklist (Honors)

Students submit:

1) Terraform diff showing:

use of UseOriginCacheControlHeaders managed cache policy

Refer to file 25-cache-correctness.tf

2) curl -i evidence showing:

Cache-Control present

x-cache transitions (Miss → Hit → Miss)

```
nightwolf@nightwolf-Inspiron-7786:~/Downloads/AWS2025/Terraform/logan/Lab 2$ curl -v https://larryharrisaws.com/api/public-feed
* Host larryharrisaws.com:443 was resolved.
* IPv6: (none)
* IPv4: 18.165.32.13, 18.165.32.79, 18.165.32.107, 18.165.32.110
* Trying 18.165.32.13:443...
* GnuTLS priority: NORMAL:-ARCFOUR-128:-CTYPE-ALL:+CTYPE-X509:-VERS-SSL3.0
* ALPN: curl offers h2,http/1.1
* found 146 certificates in /etc/ssl/certs/ca-certificates.crt
* found 441 certificates in /etc/ssl/certs
* SSL connection using TLS1.3 / ECDHE_RSA_AES_128_GCM_SHA256
* server certificate verification OK
* server certificate status verification SKIPPED
* common name: app.larryharrisaws.com (matched)
* server certificate expiration date OK
* server certificate activation date OK
* certificate public key: RSA
* certificate version: #3
* subject: CN=app.larryharrisaws.com
* start date: Fri, 23 Jan 2026 00:00:00 GMT
* expire date: Sun, 21 Feb 2027 23:59:59 GMT
* issuer: C=US,O=Amazon,CN=Amazon RSA 2048 M04
* ALPN: server accepted h2
* Connected to larryharrisaws.com (18.165.32.13) port 443
* using HTTP/2
* [HTTP/2] [1] OPENED stream for https://larryharrisaws.com/api/public-feed
* [HTTP/2] [1] [:method: GET]
* [HTTP/2] [1] [:scheme: https]
* [HTTP/2] [1] [:authority: larryharrisaws.com]
* [HTTP/2] [1] [:path: /api/public-feed]
* [HTTP/2] [1] [user-agent: curl/8.13.0]
* [HTTP/2] [1] [accept: */*]
> GET /api/public-feed HTTP/2
> Host: larryharrisaws.com
> User-Agent: curl/8.13.0
> Accept: */*
>
* Request completely sent off
< HTTP/2 200
< content-type: application/json
< content-length: 213
< date: Fri, 23 Jan 2026 14:36:00 GMT
< cache-control: public, s-maxage=30, max-age=0
< server: Werkzeug/3.1.5 Python/3.9.25
< x-cache: Miss from cloudfront
< via: 1.1 89396c79b05cfbb1fdb048026d594622.cloudfront.net (CloudFront)
< x-amz-cf-pop: TPA52-P1
```

< x-amz-cf-id: 4fOvjSwQ42s4XZzXOIC4aSnM92Pq_CH7m9s178OPzdgf3ZHyol8SIQ==
<
{"message_of_the_minute":"Message of the minute: 36 - Fresh from origin at request time!","note":"This
should be cached by CloudFront for 30 seconds (s-
maxage=30)","server_time_utc":"2026-01-23T14:36:00.871670Z"}
* Connection #0 to host larryharrisaws.com left intact
nightwolf@nightwolf-Inspiron-7786:~/Downloads/AWS2025/Terraform/logan/Lab 2\$ curl -v <https://larryharrisaws.com/api/public-feed>
* Host larryharrisaws.com:443 was resolved.
* IPv6: (none)
* IPv4: 18.165.32.110, 18.165.32.13, 18.165.32.79, 18.165.32.107
* Trying 18.165.32.110:443...
* GnuTLS priority: NORMAL:-ARCFOUR-128:-CTYPE-ALL:+CTYPE-X509:-VERS-SSL3.0
* ALPN: curl offers h2,http/1.1
* found 146 certificates in /etc/ssl/certs/ca-certificates.crt
* found 441 certificates in /etc/ssl/certs
* SSL connection using TLS1.3 / ECDHE_RSA_AES_128_GCM_SHA256
* server certificate verification OK
* server certificate status verification SKIPPED
* common name: app.larryharrisaws.com (matched)
* server certificate expiration date OK
* server certificate activation date OK
* certificate public key: RSA
* certificate version: #3
* subject: CN=app.larryharrisaws.com
* start date: Fri, 23 Jan 2026 00:00:00 GMT
* expire date: Sun, 21 Feb 2027 23:59:59 GMT
* issuer: C=US,O=Amazon,CN=Amazon RSA 2048 M04
* ALPN: server accepted h2
* Connected to larryharrisaws.com (18.165.32.110) port 443
* using HTTP/2
* [HTTP/2] [1] OPENED stream for <https://larryharrisaws.com/api/public-feed>
* [HTTP/2] [1] [:method: GET]
* [HTTP/2] [1] [:scheme: https]
* [HTTP/2] [1] [:authority: larryharrisaws.com]
* [HTTP/2] [1] [:path: /api/public-feed]
* [HTTP/2] [1] [user-agent: curl/8.13.0]
* [HTTP/2] [1] [accept: */*]
> GET /api/public-feed HTTP/2
> Host: larryharrisaws.com
> User-Agent: curl/8.13.0
> Accept: */*
>
* Request completely sent off
< HTTP/2 200
< content-type: application/json
< content-length: 213
< date: Fri, 23 Jan 2026 14:36:00 GMT
< cache-control: public, s-maxage=30, max-age=0
< server: Werkzeug/3.1.5 Python/3.9.25

< x-cache: Hit from cloudfront
< via: 1.1 8d93f1c4d47d43ea08f1881ff12f45ce.cloudfront.net (CloudFront)
< x-amz-cf-pop: TPA52-P1
< x-amz-cf-id: ndFXjQ5MkPHUYrpQeEbZ3n_lt7rRL2tyjCzysRhoEWEyYrf-a7qDow==
< age: 2
<
{"message_of_the_minute":"Message of the minute: 36 - Fresh from origin at request time!", "note":"This should be cached by CloudFront for 30 seconds (s-maxage=30)", "server_time_utc":"2026-01-23T14:36:00.871670Z"}
* Connection #0 to host larrryharrisaws.com left intact
nightwolf@nightwolf-Inspiron-7786:~/Downloads/AWS2025/Terraform/logan/Lab 2\$ curl -v <https://larrryharrisaws.com/api/public-feed>
* Host larrryharrisaws.com:443 was resolved.
* IPv6: (none)
* IPv4: 18.165.32.79, 18.165.32.110, 18.165.32.13, 18.165.32.107
* Trying 18.165.32.79:443...
* GnuTLS priority: NORMAL:-ARCFOUR-128:-CTYPE-ALL:+CTYPE-X509:-VERS-SSL3.0
* ALPN: curl offers h2,http/1.1
* found 146 certificates in /etc/ssl/certs/ca-certificates.crt
* found 441 certificates in /etc/ssl/certs
* SSL connection using TLS1.3 / ECDHE_RSA_AES_128_GCM_SHA256
* server certificate verification OK
* server certificate status verification SKIPPED
* common name: app.larrryharrisaws.com (matched)
* server certificate expiration date OK
* server certificate activation date OK
* certificate public key: RSA
* certificate version: #3
* subject: CN=app.larrryharrisaws.com
* start date: Fri, 23 Jan 2026 00:00:00 GMT
* expire date: Sun, 21 Feb 2027 23:59:59 GMT
* issuer: C=US,O=Amazon,CN=Amazon RSA 2048 M04
* ALPN: server accepted h2
* Connected to larrryharrisaws.com (18.165.32.79) port 443
* using HTTP/2
* [HTTP/2] [1] OPENED stream for <https://larrryharrisaws.com/api/public-feed>
* [HTTP/2] [1] [:method: GET]
* [HTTP/2] [1] [:scheme: https]
* [HTTP/2] [1] [:authority: larrryharrisaws.com]
* [HTTP/2] [1] [:path: /api/public-feed]
* [HTTP/2] [1] [user-agent: curl/8.13.0]
* [HTTP/2] [1] [accept: */*]
> GET /api/public-feed HTTP/2
> Host: larrryharrisaws.com
> User-Agent: curl/8.13.0
> Accept: */*
>
* Request completely sent off
< HTTP/2 200
< content-type: application/json

```
< content-length: 213
< date: Fri, 23 Jan 2026 14:36:49 GMT
< cache-control: public, s-maxage=30, max-age=0
< server: Werkzeug/3.1.5 Python/3.9.25
< x-cache: Miss from cloudfront
< via: 1.1 9a28211412e847247d8938b4854bacc4.cloudfront.net (CloudFront)
< x-amz-cf-pop: TPA52-P1
< x-amz-cf-id: aHE08YLVi2otNfQRPNaSH-jde5ZShCAwPX1rGCyQI6F_hXcq7zX8AA==
<
{"message_of_the_minute":"Message of the minute: 36 - Fresh from origin at request time!","note":"This
should be cached by CloudFront for 30 seconds (s-
maxage=30)","server_time_utc":"2026-01-23T14:36:49.235007Z"}
* Connection #0 to host larryharrisaws.com left intact
nightwolf@nightwolf-Inspiron-7786:~/Downloads/AWS2025/Terraform/logan/Lab 2$
```

3) One paragraph answering:

Why origin-driven caching is safer for APIs
When you would still disable caching entirely

After spending a decade knee-deep in traditional networking—routing protocols, firewalls, load balancers, MPLS circuits, troubleshooting BGP flaps at 3 a.m.—I've only been working with cloud environments for about four months. From that perspective, origin-driven caching feels like the sanest thing you can do for APIs. In the old world you'd never dream of letting some upstream proxy blindly cache dynamic responses; you controlled exactly what got cached where, usually with very tight TTLs or explicit no-cache directives at the edge. Origin-driven caching gives the application the same level of control: the backend decides via Cache-Control headers whether something is public/s-maxage=30, private/no-store, no-cache/must-revalidate, etc. That prevents the classic nightmare of a CDN serving stale user data, leaking one customer's session to another, or returning yesterday's pricing to a high-value client. You still disable caching entirely (CachingDisabled policy or blanket no-store) when the payload contains anything truly sensitive (tokens, PII, one-time-use codes), when every response is 100% unique per request (no meaningful cache key), or when you're still getting your feet wet in the cloud and you'd rather guarantee every request hits the origin so you can watch logs, debug, and avoid any chance of stale data confusing you while you're learning how the edge actually behaves.