

Scenario 1:

Κατά τη διαδικασία του User authentication - authorization στην κλήση του userApi στο path "/verify-auth", στην περίπτωση μη διαθεσιμότητας του UserService, το οποίο έχει timeout μηχανισμό για 5 δευτερόλεπτα, εάν δεν είναι διαθέσιμη η υπηρεσία 1 φορά τότε ενεργοποιείται το κύκλωμα του circuit breaker για 10 seconds (@CircuitBreaker(requestVolumeThreshold = 2, failureRatio = 0.5, delay = 10000)) Κατά τη διάρκεια αυτής της περιόδου, η εκτέλεση της μεθόδου θα αποτυγχάνει αμέσως με εξαίρεση CircuitBreakerOpenException, χωρίς να προσπαθήσει να καλέσει τον εξωτερικό πόρο. Η κλήση θα μπορεί να πραγματοποιηθεί επιτυχώς ύστερα από 10 seconds.

Jaeger & Logs:

Πραγματοποιήθηκαν 3 κλήσεις στο endpoint getSongById localhost:8081/api/songs/7003. Η αναμενόμενη λειτουργία του συστήματος είναι στην 1^η να πάρει timeout, στη 2^η CircuitBreakerOpenException που έχουμε ορίσει να επιστρέψει 422 και μόνον εφόσον έχουν περάσει 10 δευτερόλεπτα μπορεί να πραγματοποιηθεί εκ νέου κλήση. Όπως φαίνεται στα screenshots:



1η κλήση:

```
17:32:08.765 [executor-thread-1] INFO  [gr.au.ra.us.in.re.re.UserResource] traceId=5a34c2c0b0f975b88dd93b911253b7b6, spanId=2312aa565cc2ef56, sampled=true - User verify auth called, invocation #0

17:32:08.767 [executor-thread-1] INFO  [gr.au.ra.us.in.re.re.UserResource] traceId=5a34c2c0b0f975b88dd93b911253b7b6, spanId=2312aa565cc2ef56, sampled=true - User verify auth has delay invocation #0

17:32:13.765 [executor-thread-1] ERROR [io.qu.ve.ht.ru.QuarkusErrorHandler] traceId=5a34c2c0b0f975b88dd93b911253b7b6, spanId=2312aa565cc2ef56, sampled=true - HTTP Request to /api/users/verify-auth failed, error id: 977f9ff6-a4ca-41b1-856f-849fb6b2bee2-7
```

Service & Operation	0μs	1.29s	2.57s	3.86s	5.1s
GET /api/songs/{id}					

Service: **content-service** | Duration: **5.14s** | Start Time: **0μs**

Tags

code.function	getSong
code.namespace	gr.aueb.radio.content.infrastructure.rest.resource.SongResource
http.client_ip	127.0.0.1
http.method	GET
http.response_content_length	34
http.route	/api/songs/{id}
http.scheme	http
http.status_code	408
http.target	/api/songs/7003
internal.span.format	otlp
net.host.name	localhost
net.host.port	8081
net.protocol.name	http
span.kind	server

> Process: host.name = MacBook.station | otel.library.name = io.quarkus.opentelemetry | service.version = 1.0.0 | telemetry.sdk.language = java | telemetry.sdk.name = opentelemetry | SpanID: dbfc80fab777dbd7 | ↗

content-service GET	5.08s	54.92ms
GET		

Service: **content-service** | Duration: **5.08s** | Start Time: **54.92ms**

Tags

error	true
http.method	GET
http.response_content_length	5442
http.status_code	500
http.url	http://localhost:8080/api/users/verify-auth
internal.span.format	otlp
net.peer.name	localhost
net.peer.port	8080
net.protocol.name	http
net.transport	ip_tcp
otel.status_code	ERROR
span.kind	client

> Process: host.name = MacBook.station | otel.library.name = io.quarkus.opentelemetry | service.version = 1.0.0 | telemetry.sdk.language = java | telemetry.sdk.name = opentelemetry | SpanID: 3c43ea241a256530 | ↗

user-service GET /api/users/verify-auth	5.07s	63.68ms
GET /api/users/verify-auth		

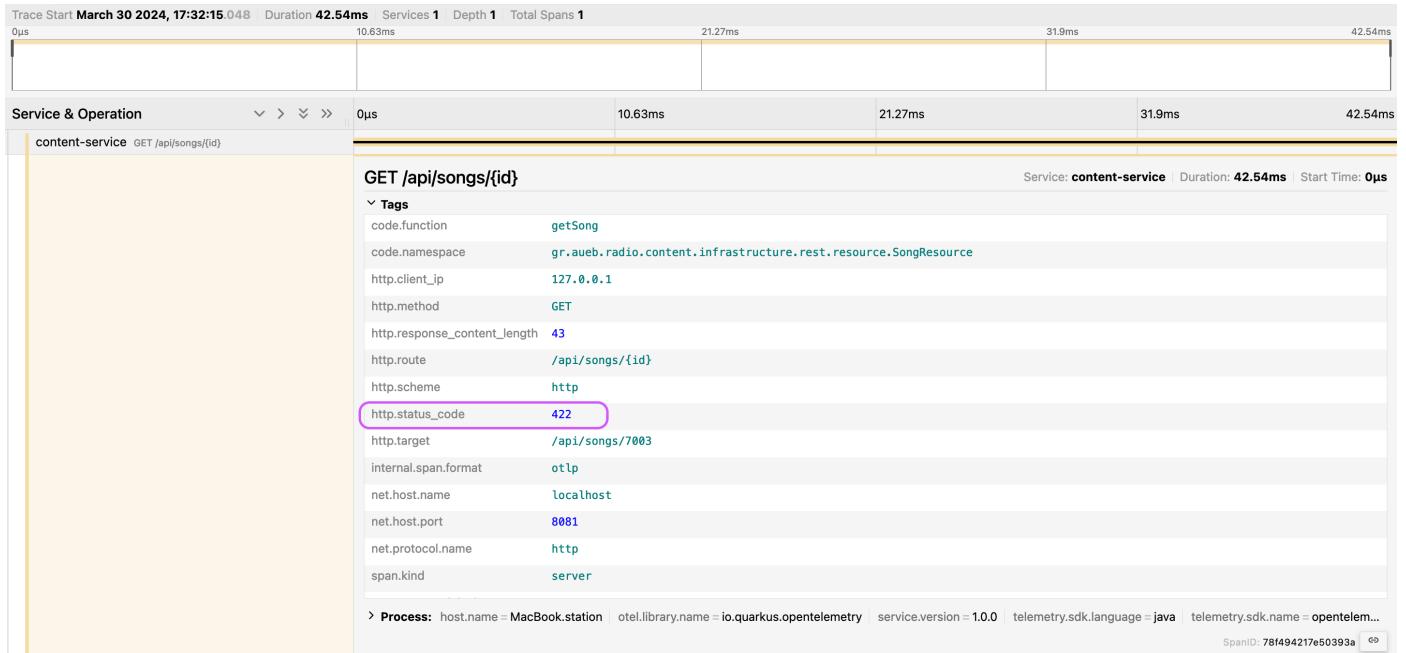
Service: **user-service** | Duration: **5.07s** | Start Time: **63.68ms**

Tags

code.function	verifyAuth
code.namespace	gr.aueb.radio.user.infrastructure.rest.resource.UserResource
error	true
http.client_ip	127.0.0.1
http.method	GET
http.response_content_length	5442
http.route	/api/users/verify-auth
http.scheme	http
http.status_code	500
http.target	/api/users/verify-auth
internal.span.format	otlp
net.host.name	localhost
net.host.port	8080
net.protocol.name	http

> Process: host.name = MacBook.station | otel.library.name = io.quarkus.opentelemetry | service.version = 1.0.0 | telemetry.sdk.language = java | telemetry.sdk.name = opentelemetry | SpanID: 2312aa565cc2ef56 | ↗

2η κλήση:

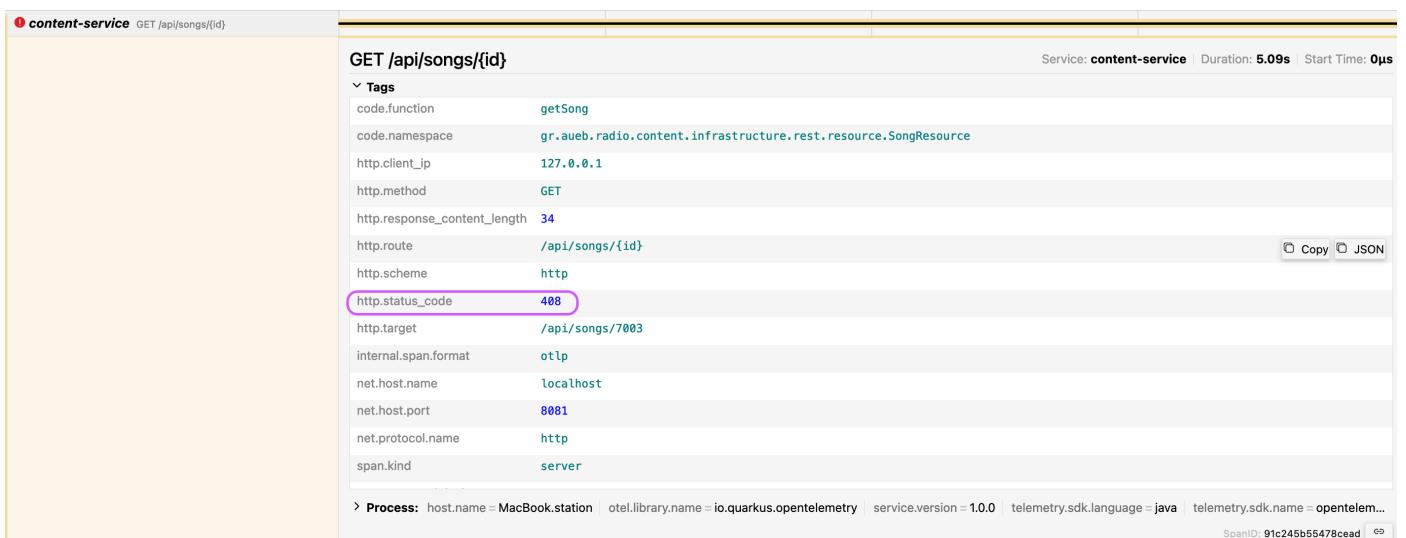


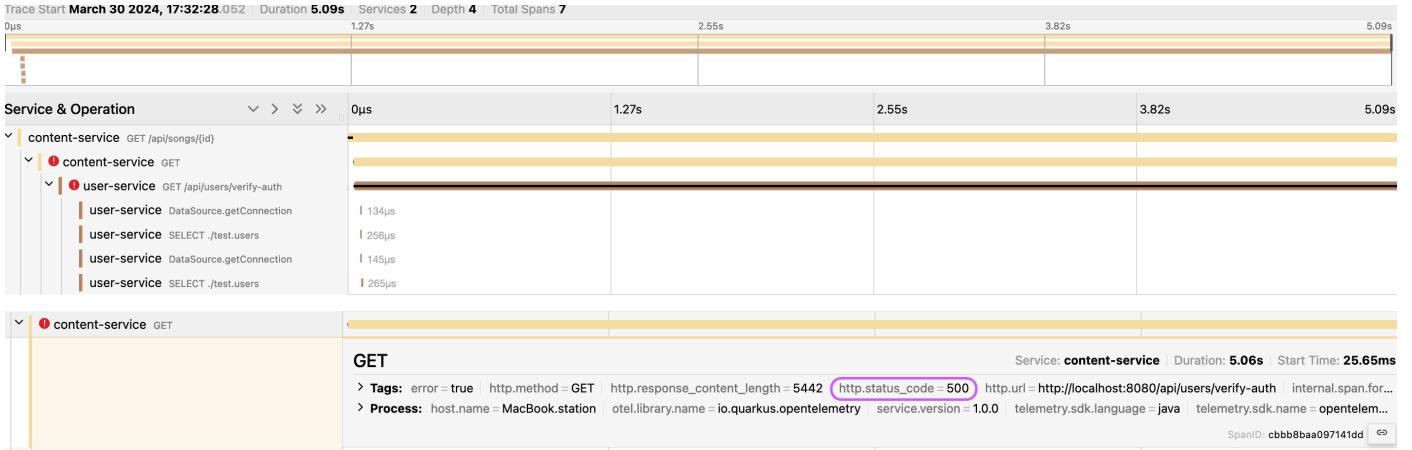
3η κλήση:

```
17:32:28.117 [executor-thread-1] INFO  [gr.au.ra.us.in.re.re.UserResource] traceId=29343a5d72b89aaddc6bc940f9df2329, spanId=8c738bc27ea3c  
cbf, sampled=true - User verify auth called, invocation #0

17:32:28.119 [executor-thread-1] INFO  [gr.au.ra.us.in.re.re.UserResource] traceId=29343a5d72b89aaddc6bc940f9df2329, spanId=8c738bc27ea3c  
cbf, sampled=true - User verify auth has delay invocation #0

17:32:33.119 [executor-thread-1] ERROR [io.qu.ve.ht.ru.QuarkusErrorHandler] traceId=29343a5d72b89aaddc6bc940f9df2329, spanId=8c738bc27ea3  
ccb, sampled=true - HTTP Request to /api/users/verify-auth failed, error id: 977f9ff6-a4ca-41b1-856f-849fb6b2bee2-8
```





Scenario 2:

Στη λειτουργικότητα που αφορά τις προτάσεις περιεχομένου για την δημιουργία εκπομπής, καλείται η υπηρεσία του περιεχομένου (content service), η οποία επιστρέφει τα αποτελέσματα ανάλογα με το είδος της εκπομπής, τη ζώνη μετάδοσης κτλ. Στην περίπτωση μη διαδεσμότητας του πόρου της υπηρεσίας, έχει οριστεί fallback μηχανισμός ο οποίος αναλαμβάνει ως εναλλακτική ενέργεια μα επιστρέψει δεδομένα με τυχαία επιλογή, ώστε στη συνέχεια από αυτά να γίνει διαλογή ανάλογα με τις απαιτήσεις της εκπομπής.

Logs:

Broadcast service

```

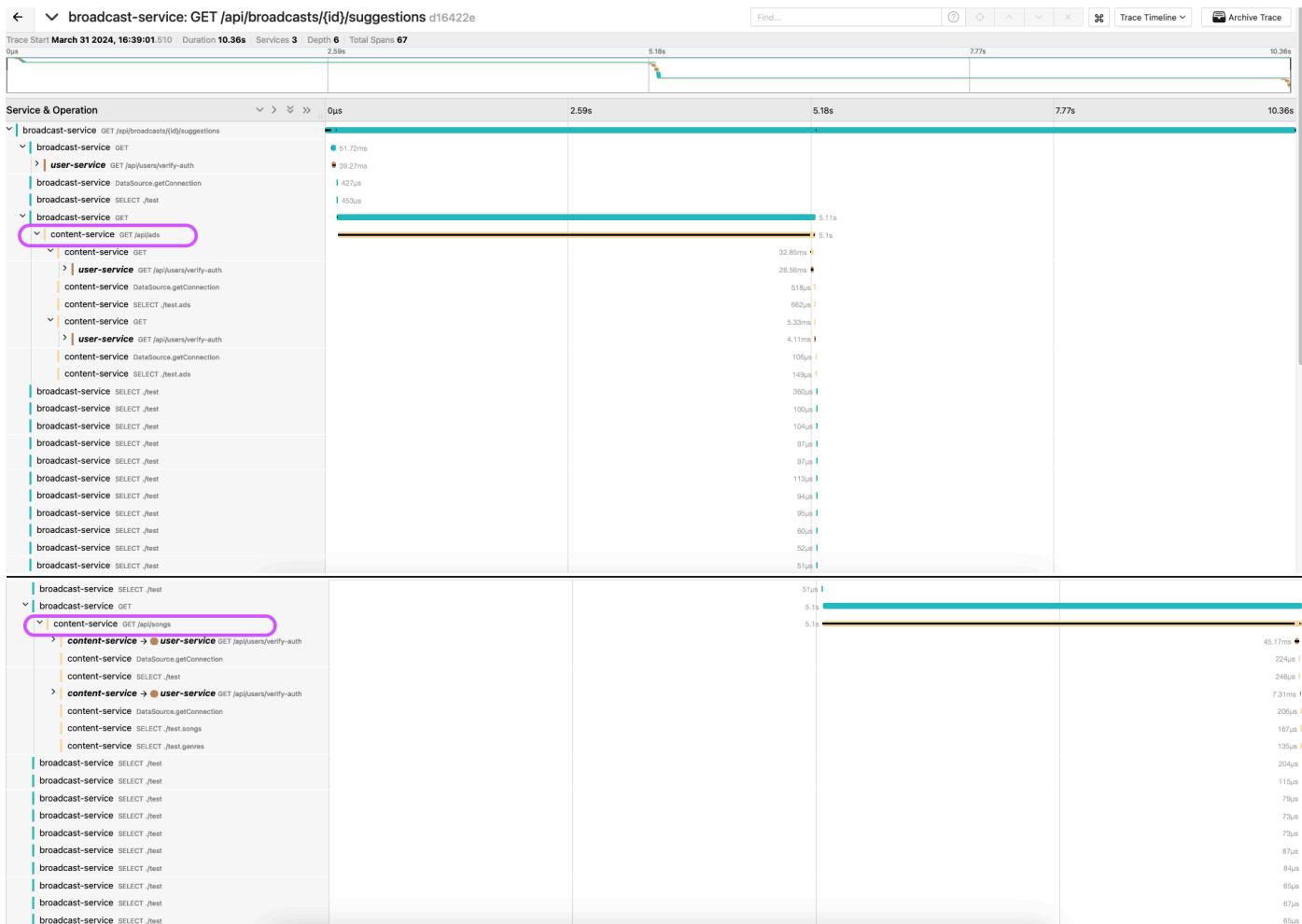
16:39:01.502 [executor-thread-1] INFO  [gr.au.ra.br.in.re.re.BroadcastResource] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=af086c3af1843632, sampled=true
- suggestions process starts
returned id 6001
16:39:01.638 [executor-thread-1] INFO  [gr.au.ra.br.ap.SuggestionsService] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=af086c3af1843632, sampled=true
- suggestAds() call content - getAdsByFilters
16:39:01.638 [executor-thread-1] INFO  [gr.au.ra.br.ap.SuggestionsService] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=af086c3af1843632, sampled=true
- suggestAds() ad retrieved getAdsByFilters from contentApi 0
16:39:01.639 [executor-thread-1] INFO  [gr.au.ra.br.in.se.co.ContentServiceImpl] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=af086c3af1843632, sampled=true
- Call content api invocation #4, getAdsByFilters
16:39:06.749 [executor-thread-1] INFO  [gr.au.ra.br.ap.SuggestionsService] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=af086c3af1843632, sampled=true
- suggestAds() ad retrieved getAdsByFilters from contentApi instead of timezone filter 10
16:39:06.758 [executor-thread-1] INFO  [gr.au.ra.br.ap.SuggestionsService] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=af086c3af1843632, sampled=true
- suggestSongs() call content - getSongsByFilters
16:39:06.759 [executor-thread-1] INFO  [gr.au.ra.br.ap.SuggestionsService] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=af086c3af1843632, sampled=true
- suggestSongs() song retrieved getSongsByFilters from contentApi 0
16:39:06.759 [executor-thread-1] INFO  [gr.au.ra.br.in.se.co.ContentServiceImpl] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=af086c3af1843632, sampled=true
- Call content api invocation #5, getSongsByFilters
16:39:11.860 [executor-thread-1] INFO  [gr.au.ra.br.ap.SuggestionsService] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=af086c3af1843632, sampled=true
- suggestSongs() song retrieved getSongsByFilters from contentApi instead of genre filter 10

```

Content service

```
Tests paused
Press [e] to edit command line args (currently ''), [r] to resume testing, [o] Toggle test output, [:] for the terminal, [h] for more options>
16:39:01.671 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.AdResource] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=a10879f745f81377, sampled=true -
Call in content api searchAds
16:39:01.673 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.AdResource] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=a10879f745f81377, sampled=true -
Content ads search has delay
16:39:01.673 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.AdResource] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=a10879f745f81377, sampled=true -
search() thread sleep 10000
16:39:06.737 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.AdResource] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=a10879f745f81377, sampled=true -
Falling back to fallbackAdRecommendations()
16:39:06.746 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.AdResource] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=a10879f745f81377, sampled=true -
fallbackAdRecommendations() fallbackAds = 10
16:39:06.781 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.SongResource] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=b9d19ec6f5575c9f, sampled=true -
Call in content api searchSongs
16:39:06.781 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.SongResource] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=b9d19ec6f5575c9f, sampled=true -
Content songs search has delay
16:39:06.781 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.SongResource] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=b9d19ec6f5575c9f, sampled=true -
search() thread sleep 10000
16:39:11.840 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.SongResource] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=b9d19ec6f5575c9f, sampled=true -
Falling back to fallbackSongRecommendations()
16:39:11.856 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.SongResource] traceId=d16422e197a0f0c1635d6ef60fa24a73, spanId=b9d19ec6f5575c9f, sampled=true -
fallbackSongRecommendations().fallbackSongs = 10
```

Jaeger



Scenario 3:

Κατά τη διαδικαδία ενημέρωσης των στοιχείων μιας διαφήμισης, απαιτείται να γίνει εντοπισμός των μεταδόσεών της που έχουν προγραμματιστεί, καλώντας το GET /ad-broadcasts?adId={adId} endpoint της broadcast υπηρεσίας. Στην περίπτωση μη διαθεσιμότητας αυτού έχει οριστεί fallback μηχανισμός ο οποίος επιστρέψει μια άδεια λίστα, ως εναλλακτική μη επιτυχούς πραγματοποίησης της κλήσης. Επιπλέον, έχει ενεργοποιηθεί fallback μηχανισμός για τη διαχείριση του timeout exception στο broadcast service.

Logs:

Broadcast service

```

19:23:55.338 [Quarkus Main Thread] INFO  [io.quarkus] traceId=, spanId=, sampled= - Installed features: [agroal, cdi, hibernate-orm, hibernate-orm-panache, jdbch2, narayana-jta, opentelemetry, rest-client, resteasy, resteasy-jsonb, smallrye-context-propagation, smallrye-fault-tolerance, smallrye-health, smallrye-metrics, smallrye-openapi, swagger-ui, vertx]
19:24:35.952 [executor-thread-1] INFO  [gr.au.ra.in.re.re.AdBroadcastResource] traceId=8c3fa8c4dd1f8c8868b17a4b65bf9d9f, spanId=88ab97ece01e2ebb, sampled=true
- AdBroadcast search called
19:24:35.953 [executor-thread-1] INFO  [gr.au.ra.in.re.re.AdBroadcastResource] traceId=8c3fa8c4dd1f8c8868b17a4b65bf9d9f, spanId=88ab97ece01e2ebb, sampled=true
- AdBroadcast search has delay invocation #1
19:24:40.974 [executor-thread-1] INFO  [gr.au.ra.in.re.re.AdBroadcastResource] traceId=8c3fa8c4dd1f8c8868b17a4b65bf9d9f, spanId=88ab97ece01e2ebb, sampled=true
- Fallback - AdBroadcast The request timed out.
19:24:45.977 [executor-thread-1] INFO  [gr.au.ra.in.re.re.AdBroadcastResource] traceId=8c3fa8c4dd1f8c8868b17a4b65bf9d9f, spanId=527f26708575860a, sampled=true
- AdBroadcast search called
19:24:45.979 [executor-thread-1] INFO  [gr.au.ra.in.re.re.AdBroadcastResource] traceId=8c3fa8c4dd1f8c8868b17a4b65bf9d9f, spanId=527f26708575860a, sampled=true
- AdBroadcast search has delay invocation #2
19:24:50.988 [executor-thread-1] INFO  [gr.au.ra.in.re.re.AdBroadcastResource] traceId=8c3fa8c4dd1f8c8868b17a4b65bf9d9f, spanId=527f26708575860a, sampled=true
- Fallback - AdBroadcast The request timed out.

```

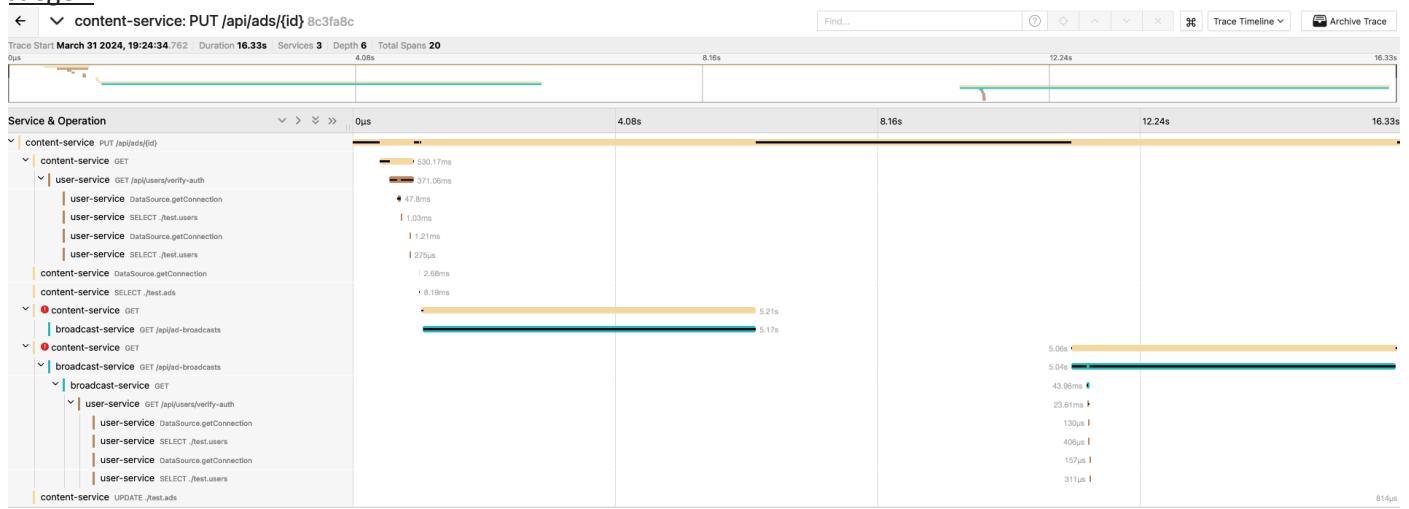
Content Service

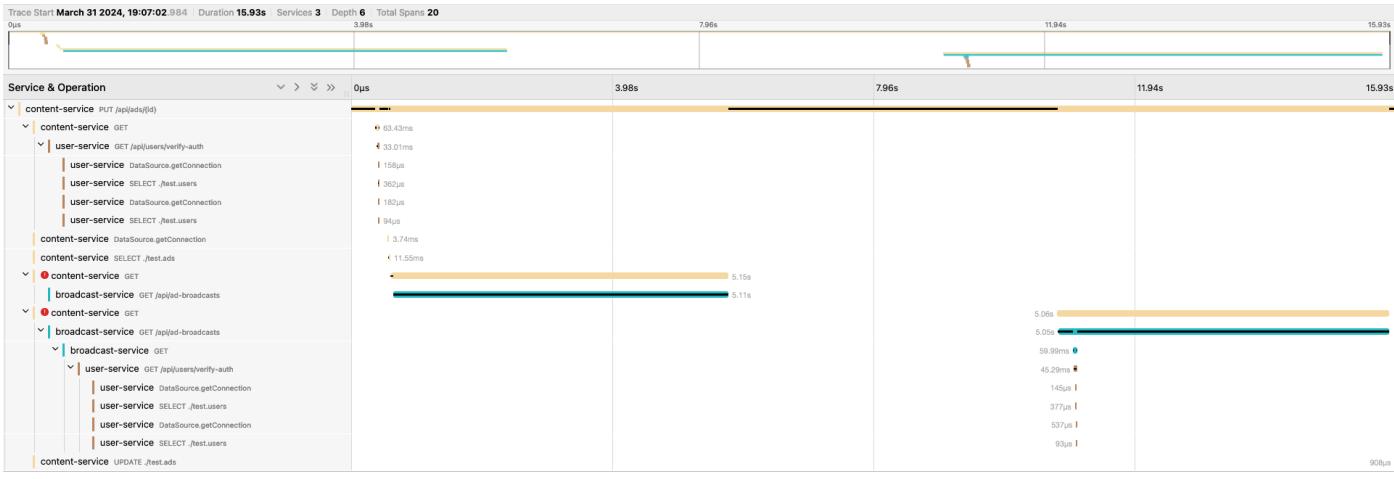
```

19:24:35.821 [executor-thread-1] INFO  [gr.au.ra.co.in.se.br.BroadcastServiceImpl] traceId=8c3fa8c4dd1f8c8868b17a4b65bf9d9f, spanId=7cf023f9c2eeb8b4, sampled=true
e - Calling Broadcast Api getAdBroadcastsByAdId
19:24:51.014 [executor-thread-1] INFO  [gr.au.ra.co.in.se.br.BroadcastServiceImpl] traceId=8c3fa8c4dd1f8c8868b17a4b65bf9d9f, spanId=7cf023f9c2eeb8b4, sampled=true
e - Can't reach broadcast api due to timeout exception
19:24:51.023 [executor-thread-1] INFO  [gr.au.ra.co.in.se.br.BroadcastServiceImpl] traceId=8c3fa8c4dd1f8c8868b17a4b65bf9d9f, spanId=7cf023f9c2eeb8b4, sampled=true
e - Falling back to zeroAdBroadcastsFallback() returning an empty list
adBroadcast0

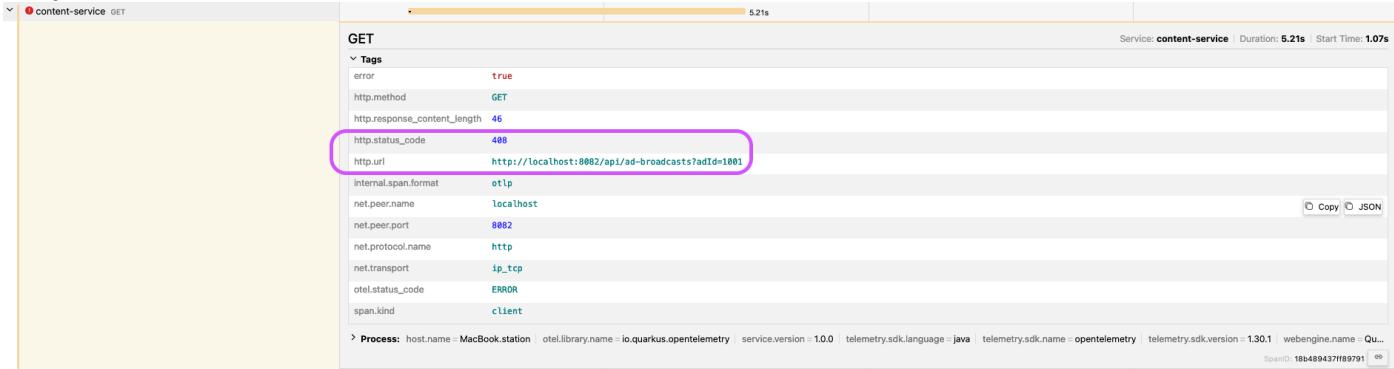
```

Jaeger:

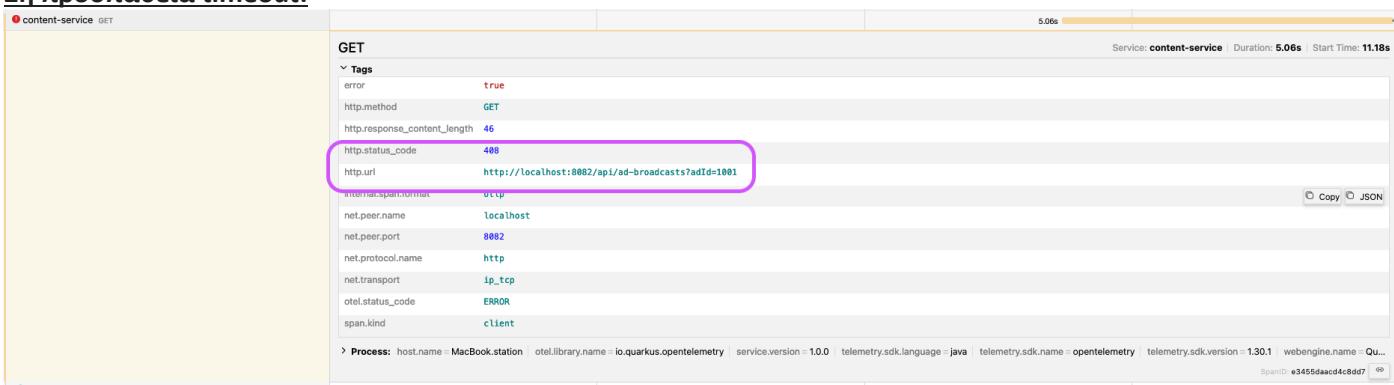




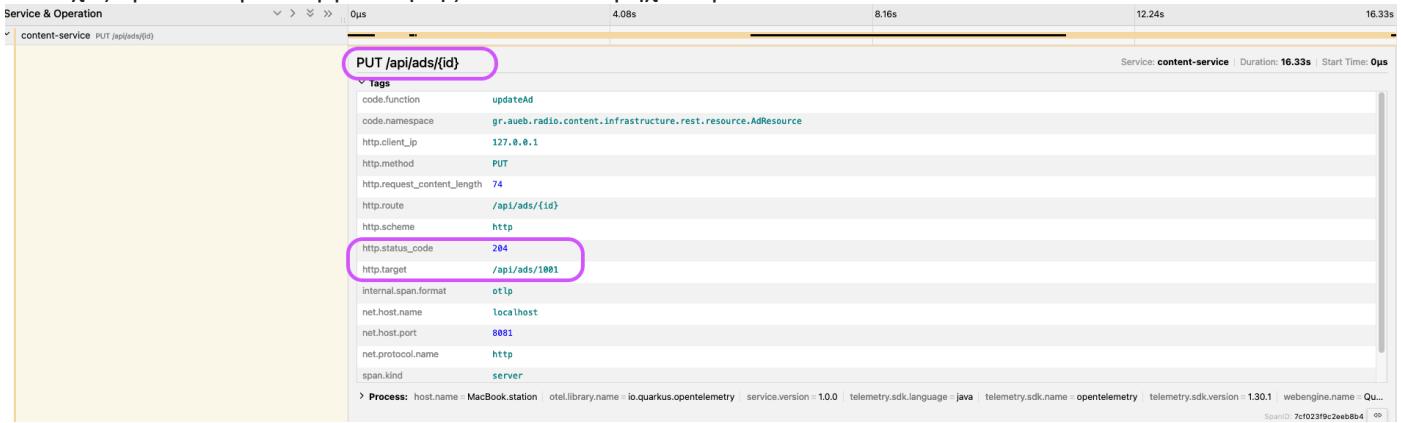
1η προσπάθεια timeout:



2η προσπάθεια timeout:



Επιτυχές update λόγω ενεργοποίησης του fallback μηχανισμού



Scenario 4:

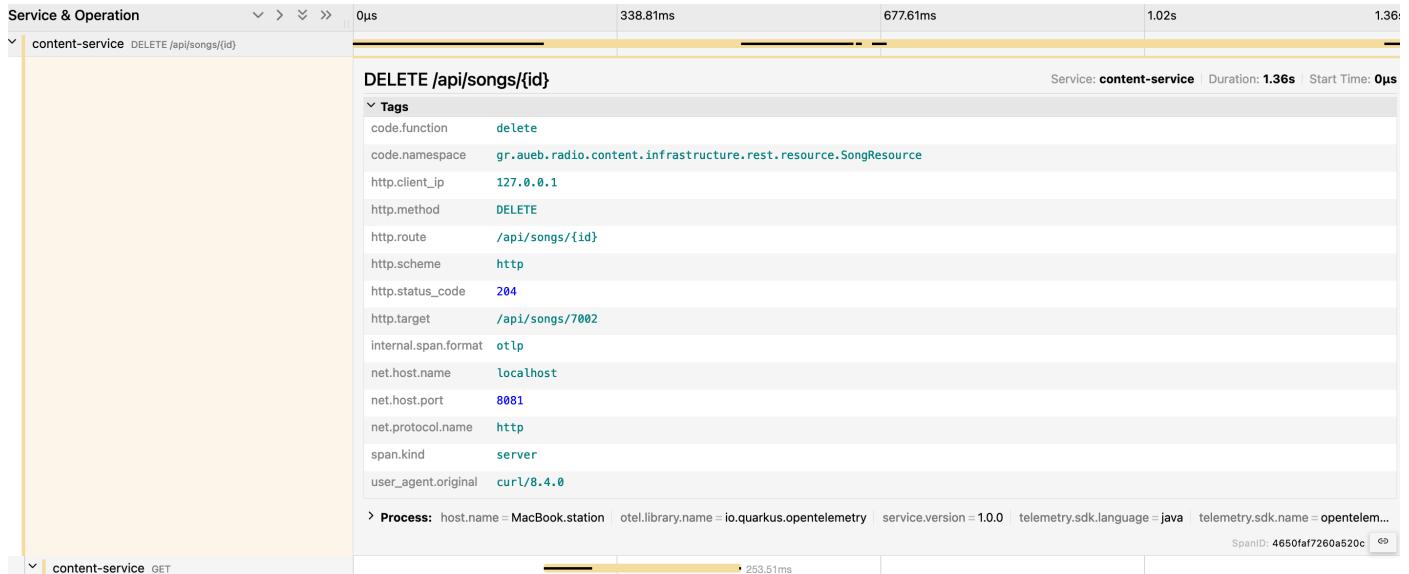
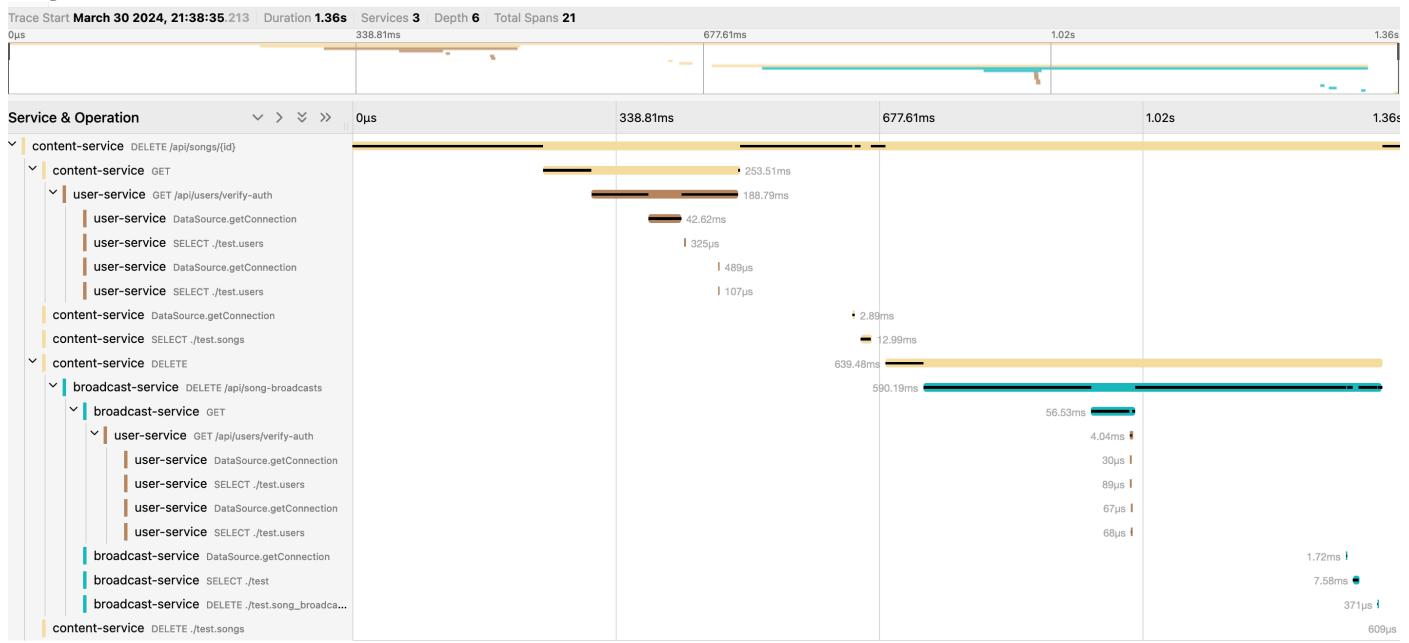
Κατά τη διαγραφή ενός τραγουδιού απαιτείται να βρεθούν και να διαγραφούν αντίστοιχα όλες οι μεταδόσεις που έχουν προγραμματιστεί για το συγκεκριμένο τραγούδι. Συνεπώς είναι απαραίτητο να ρυθμίσουμε το μέγιστο αριθμό ταυτόχρονων κλήσεων για την επιλεγμένη μέθοδο και συγκεκριμένα για την κλήση του interface του Broadcast service deleteSongBroadcastsBySongId (String auth, Integer songId). Πρόκειται για ένα πολύ heavy process, καθώς ένα τραγούδι μπορεί να έχει προγραμματιστεί σε πολλές μεταδόσεις, συνεπώς απαιτείται να οριοθετήσουμε τις παράλληλες κλήσεις θέτοντας @Bulkhead(value = 1) στη διαδικασία του deletion server side, και το BulkheadException διαχειρίζεται ένας fallback μηχανισμός που επιστρέφει κωδικό λάθους 429.

Το συγκεκριμένο test case το πραγματοποιήσαμε χρησιμοποιήθηκε το curl για την διαγραφή 2 τραγουδιών παράλληλα: for id in 7001 7002; do curl -X DELETE http://localhost:8081/api/songs/\$id -H 'Authorization: Basic CHJvZHVjZXI6cHJvZHVjZXI=' & done

Logs:

```
21:38:35.277 [executor-thread-2] INFO  [gr.au.ra.co.in.re.re.SongResource] traceId=5b92eac49d2c8c83acd09c9bd4427984, spanId=e77bbce6db5988e0, sampled=true - Falling back to DeleteConcurrent() - failed to delete songId = 7001
```

Jaeger:



Service & Operation ▾ ▶ ▷ ▸ ▹ 0μs 18.27ms 36.55ms 54.82ms 73.1ms

content-service	DELETE	/api/songs/{id}
-----------------	--------	-----------------

DELETE /api/songs/{id}

Service: **content-service** | Duration: **73.1ms** | Start Time: **0μs**

Tags

- code.function delete
- code.namespace gr.aueb.radio.content.infrastructure.rest.resource.SongResource
- http.client_ip 127.0.0.1
- http.method DELETE
- http.response_content_length 0
- http.route /api/songs/{id}
- http.scheme http
- http.status_code 429
- http.target /api/songs/7001
- internal.span.format otlp
- net.host.name localhost
- net.host.port 8081
- net.protocol.name http
- span.kind server

> **Process:** host.name = MacBook.station | otlp.library.name = io.quarkus.opentelemetry | service.version = 1.0.0 | telemetry.sdk.language = java | telemetry.sdk.name = opentelemetry | SpanID: e77bbce6db5988e0 | ↗

Αντίστοιχη διαχείριση υπάρχει και για το σενάριο διαγραφής **διαφημίσεων**.

Scenario 5:

Κατά τη δημιουργία μιας μετάδοσης διαφήμισης στο Broadcast service, απαιτείται κλήση του content service προκειμένου να αποσταλεί όλο το αντικείμενο του ad και να διαχειριστεί στη συνέχεια όπως απαιτείται από την υλοποίηση. Σε αυτό το σημείο, στην περίπτωση μη διαθεσιμότητας της υπηρεσίας του content το σύστημα θα πραγματοποιήσει 3 προσπάθειες σειριακά (@Retry(maxRetries = 3, delay = 5000)) με αναμονή 5 δευτερολέπτων ανάμεσα στις προσπάθειες.

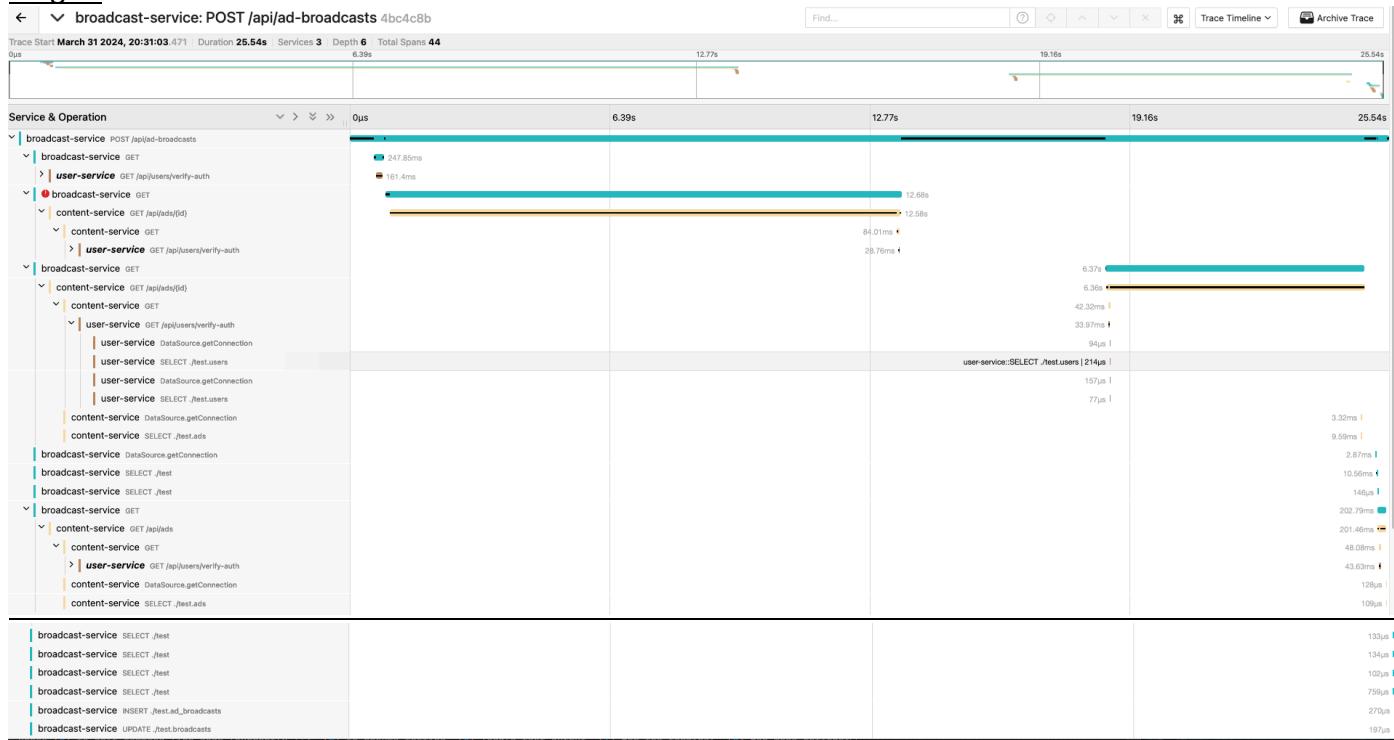
Logs:

```

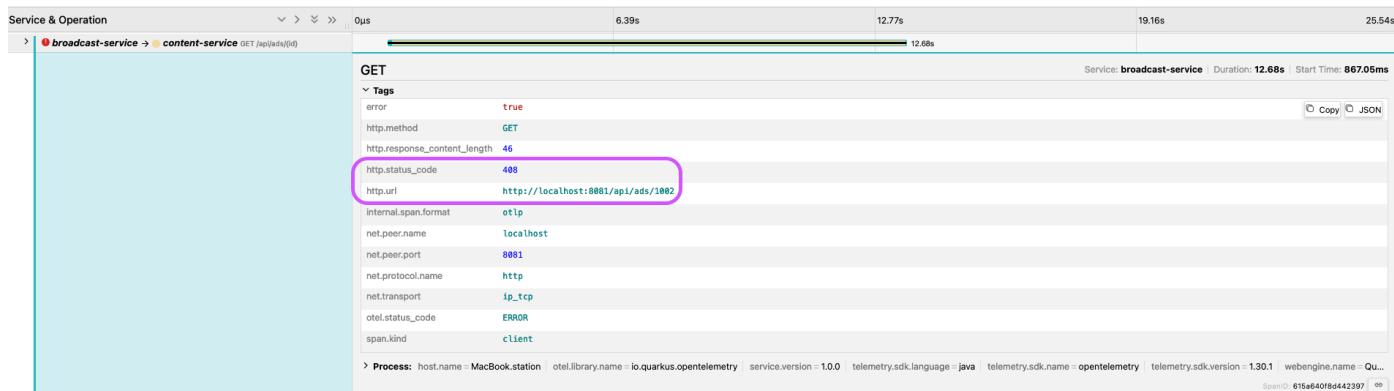
20:31:04.562 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.AdResource] traceId=4bc4c8b1426e960dfbab0be59b43ed84, spanId=26e3eb8ce7511c65, sampled=true - Call content api getAd
20:31:04.564 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.AdResource] traceId=4bc4c8b1426e960dfbab0be59b43ed84, spanId=26e3eb8ce7511c65, sampled=true - getAd has delay invocation #1
20:31:16.899 [executor-thread-1] INFO  [gr.au.ra.co.in.se.us.UserApiRequestInterceptor] traceId=4bc4c8b1426e960dfbab0be59b43ed84, spanId=26e3eb8ce7511c65, sampled=true - Calling user API
with URL: http://localhost:8080/api/users/verify-auth
20:31:17.009 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.AdResource] traceId=4bc4c8b1426e960dfbab0be59b43ed84, spanId=26e3eb8ce7511c65, sampled=true - Fallback - Ad The request timed
out.
20:31:22.097 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.AdResource] traceId=4bc4c8b1426e960dfbab0be59b43ed84, spanId=683f498a6439e08d, sampled=true - Call content api getAd
20:31:22.101 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.AdResource] traceId=4bc4c8b1426e960dfbab0be59b43ed84, spanId=683f498a6439e08d, sampled=true - getAd has delay invocation #2
20:31:22.105 [executor-thread-1] INFO  [gr.au.ra.co.in.se.us.UserApiRequestInterceptor] traceId=4bc4c8b1426e960dfbab0be59b43ed84, spanId=683f498a6439e08d, sampled=true - Calling user API
with URL: http://localhost:8080/api/users/verify-auth
20:31:28.394 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.AdResource] traceId=4bc4c8b1426e960dfbab0be59b43ed84, spanId=683f498a6439e08d, sampled=true - findAd() returning successfully
- id 1002
20:31:28.754 [executor-thread-1] INFO  [gr.au.ra.co.in.re.re.AdResource] traceId=4bc4c8b1426e960dfbab0be59b43ed84, spanId=f721c3d47e985e6e, sampled=true - Call in content api searchAds
20:31:28.755 [executor-thread-1] INFO  [gr.au.ra.co.in.se.us.UserApiRequestInterceptor] traceId=4bc4c8b1426e960dfbab0be59b43ed84, spanId=f721c3d47e995e6e, sampled=true - Calling user API
with URL: http://localhost:8080/api/users/verify-auth

```

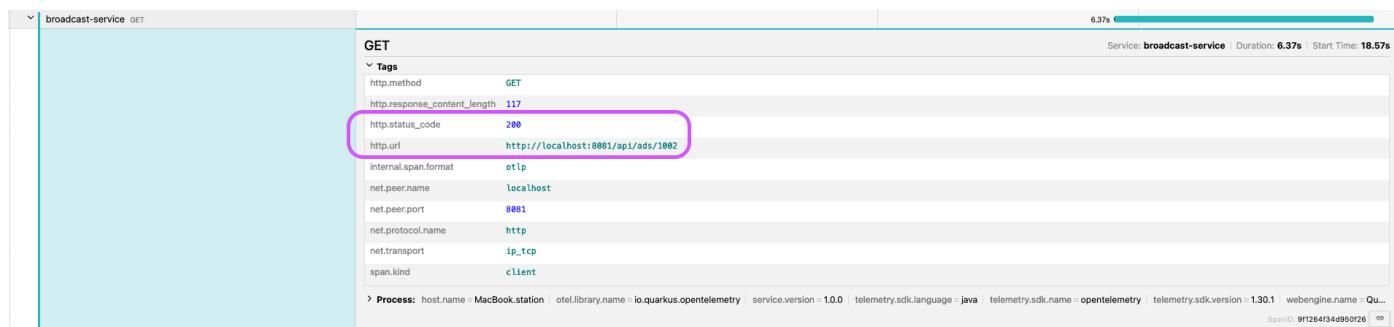
Jaeger:



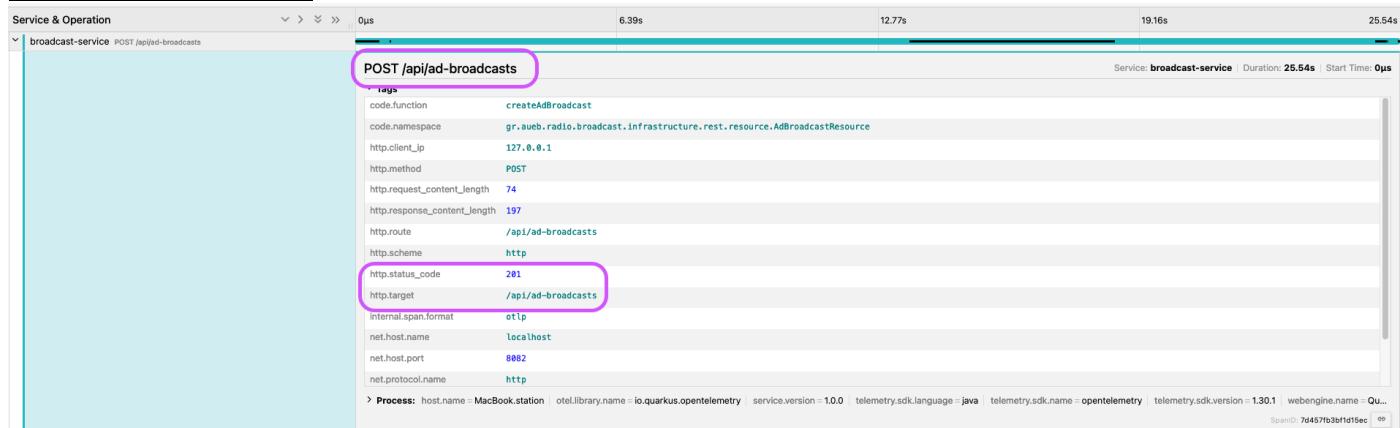
1^η κλήση



2^η κλήση



Δημιουργία μετάδοσης



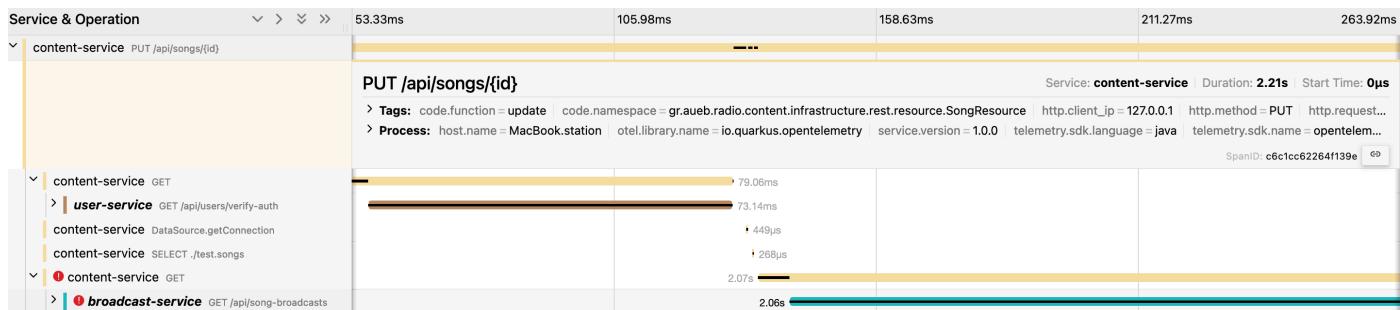
Scenario 6:

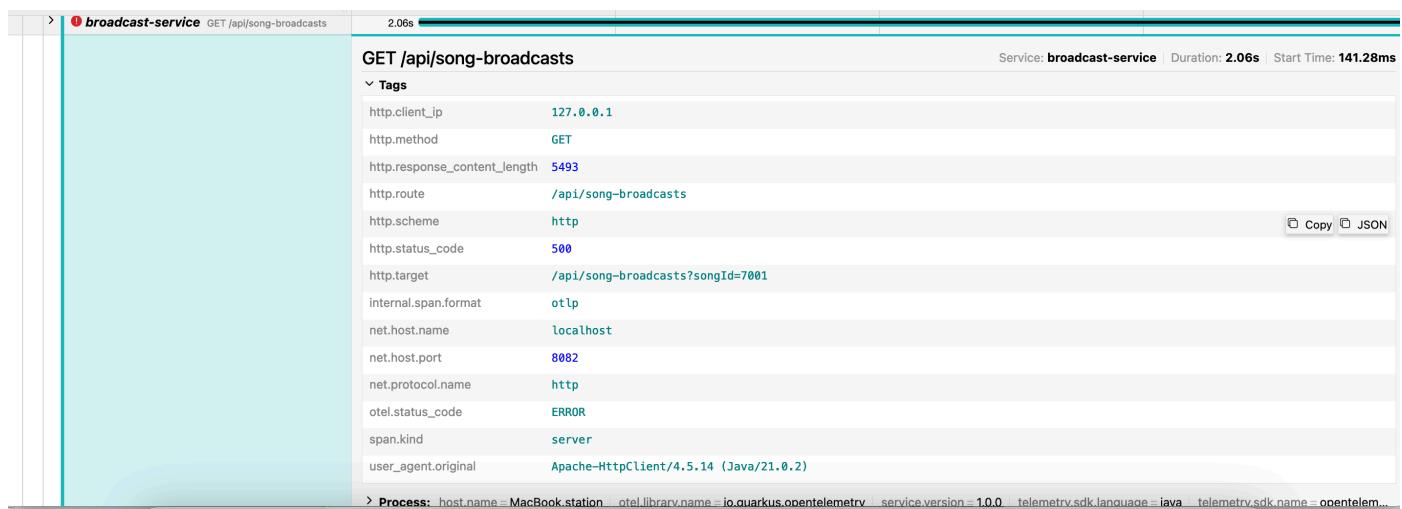
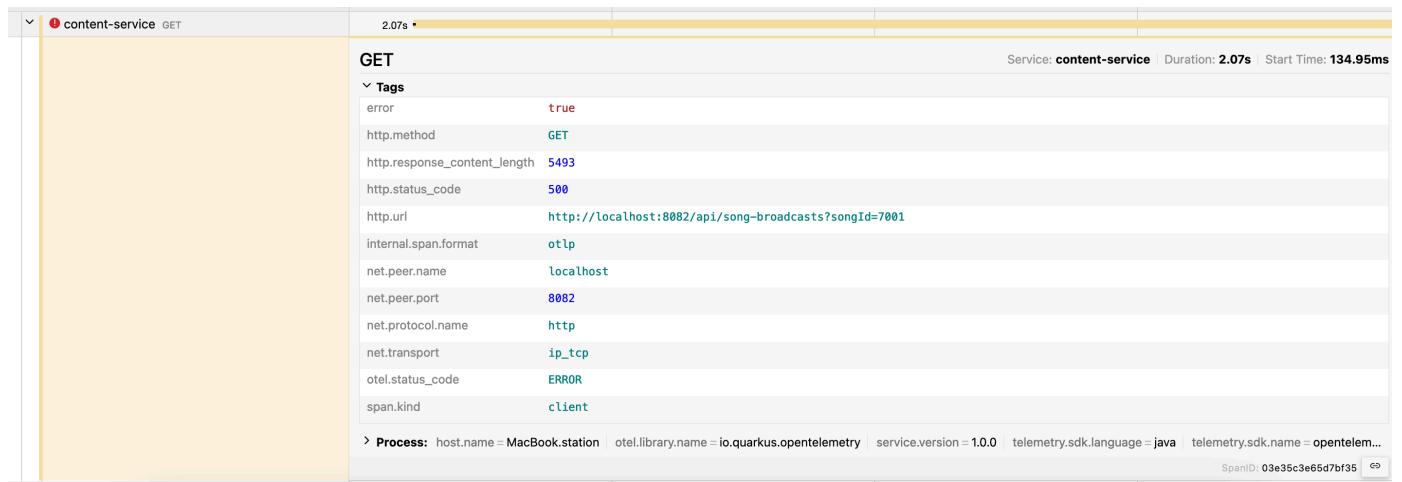
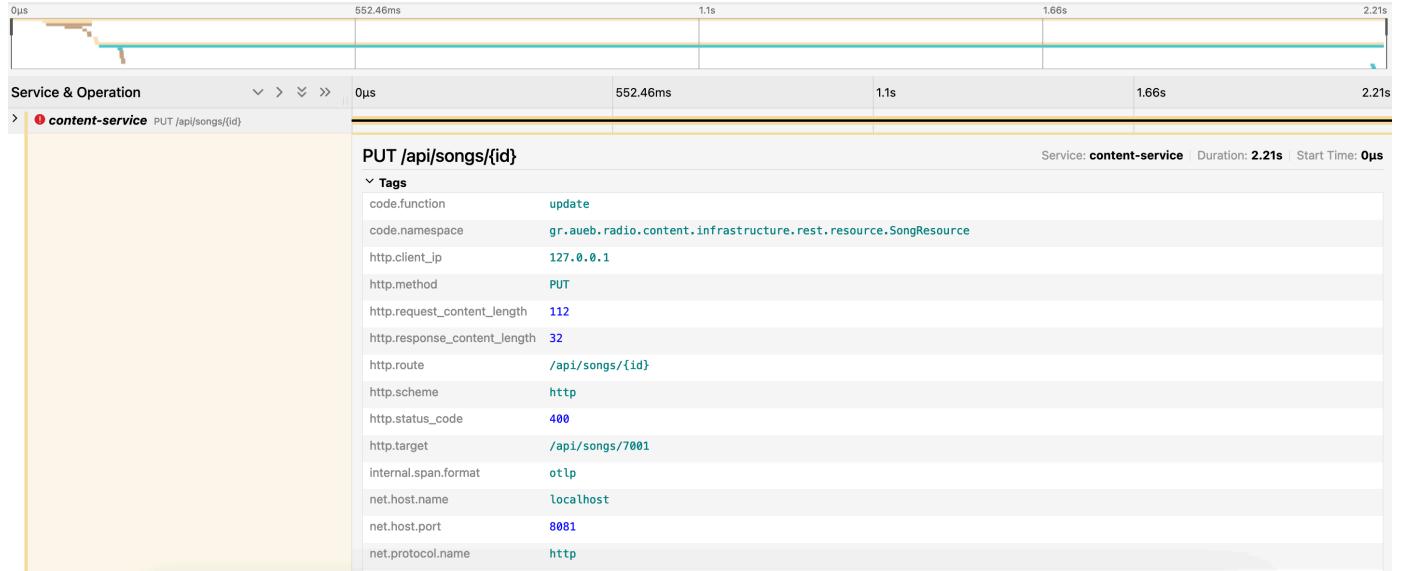
Κατά τη διαδικασία ενημέρωσης (update) ενός τραγουδιού στο content service, πραγματοποιείται επικοινωνία με το broadcast service, προκειμένου να ενημερωθούν αναλόγως και οι μεταδόσεις του συγκεκριμένου τραγουδιού. Για την καλύτερη ευρωστία του συστήματος έχει τοποθετηθεί timeout μηχανισμός στο SongBroadcast resource για διάστημα 2 δευτερολέπτων, συνεπώς εξασφαλίζεται ότι θα αποδεσμευτεί ο πόρος σε εύλογο χρονικό διάστημα.

Logs:

```
14:27:26.150 [executor-thread-1] INFO  [gr.au.ra.br.ap.SongBroadcastService] traceId=540454c3c8bfa39b4022cccee2d5f174, spanId=9b3ff00434a64d5f, sampled=true - SongBroadcastService search has delay
14:27:28.305 [executor-thread-1] ERROR [io.quve.ht.ru.QuarkusErrorHandler] traceId=540454c3c8bfa39b4022cccee2d5f174, spanId=9b3ff00434a64d5f, sampled=true - HTTP Request to /api/song-broadcasts?songId=7001 failed, error id: eced8ac9-ee06-4bc3-b1f9-02bc7ff671c6-1
```

Jaeger:





Scenario 7:

Για λόγους ανθεκτικότητας του συστήματος, έχει τοποθετηθεί στο resource για το creation μιας εκπομπής όριο παράλληλων κλήσεων @Bulkhead(value = 3), καθώς η δημιουργία της εκπομπής αποτελεί μια heavy διαδικασία με πραγματοποιήση πολλών επικοινωνιών ανάμεσα στις μικρο-υπηρεσίες.

Το συγκεκριμένο test case το πραγματοποιήσαμε χρησιμοποιήθηκε το curl για την παράλληλη δημιουργία 4 εκπομπών με το script:

```
curl -X POST -H "Content-Type: application/json" -d '{"type": "PLAYLIST", "genreId": 3, "startingDate": "16-10-2022", "startingTime": "10:00", "duration": 180}' localhost:8082/api/broadcasts/ -H 'Authorization: Basic cHJvZHVjZXI6cHJvZHVjZXI=' &
```

```
curl -X POST -H "Content-Type: application/json" -d '{"type": "PLAYLIST", "genreId": 3, "startingDate": "16-10-2023", "startingTime": "10:00", "duration": 180}' localhost:8082/api/broadcasts/ -H 'Authorization: Basic cHJvZHVjZXI6cHJvZHVjZXI=' &
```

```
curl -X POST -H "Content-Type: application/json" -d '{"type": "PLAYLIST", "genreId": 3, "startingDate": "16-10-2024", "startingTime": "10:00", "duration": 180}' localhost:8082/api/broadcasts/ -H 'Authorization: Basic cHJvZHVjZXI6cHJvZHVjZXI=' &
```

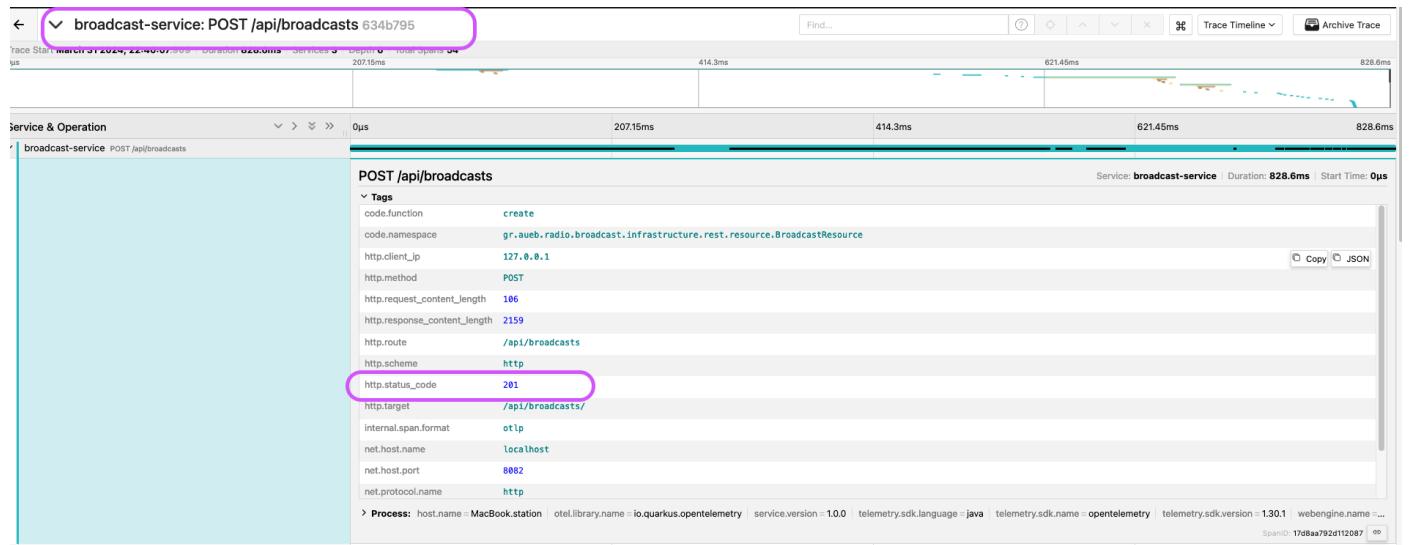
```
curl -X POST -H "Content-Type: application/json" -d '{"type": "PLAYLIST", "genreId": 3, "startingDate": "16-11-2024", "startingTime": "10:00", "duration": 180}' localhost:8082/api/broadcasts/ -H 'Authorization: Basic cHJvZHVjZXI6cHJvZHVjZXI=' &
```

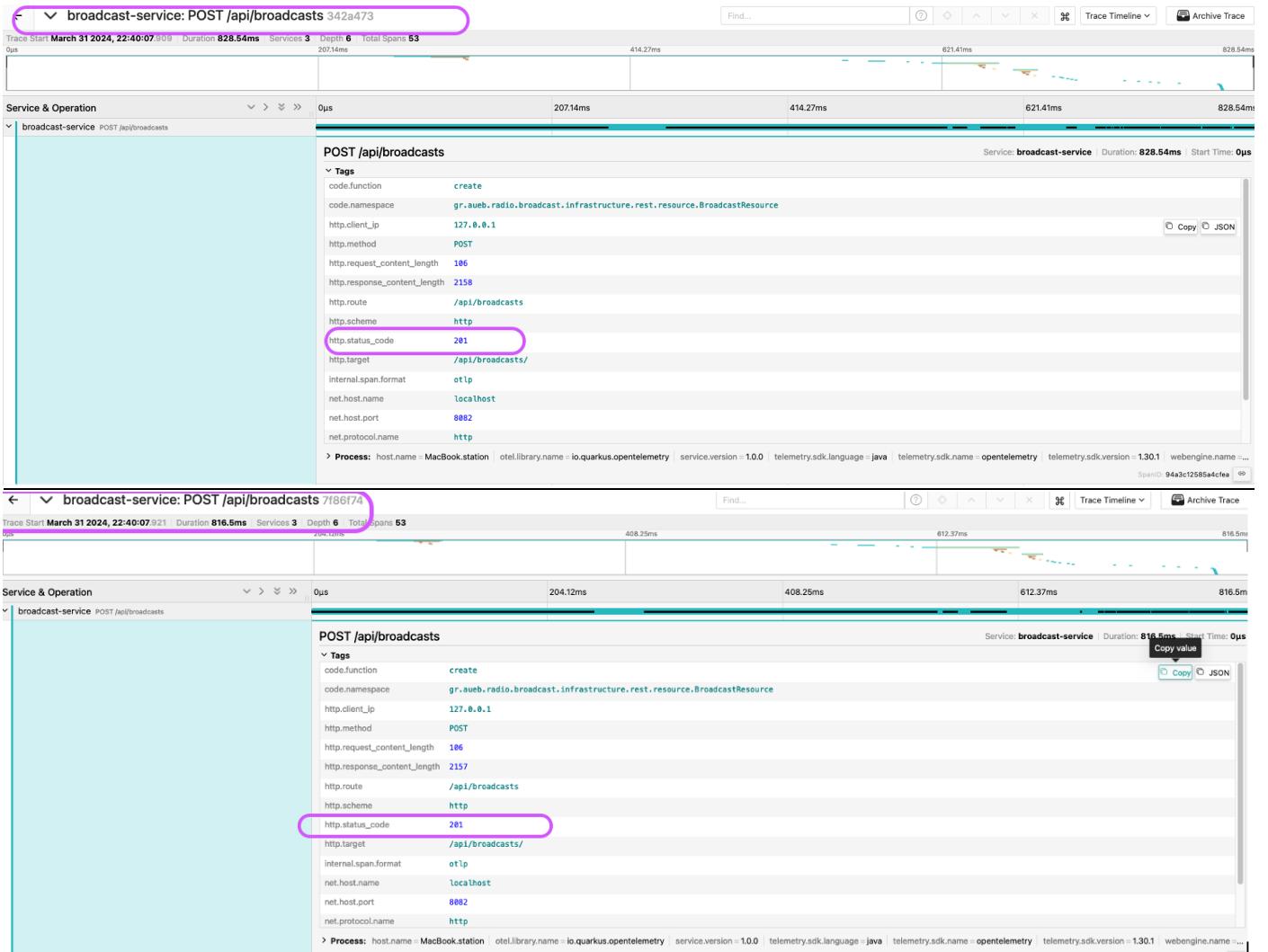
Logs:

```
22:40:08.001 [executor-thread-3] INFO  [gr.au.ra.br.in.re.re.BroadcastResource] traceId=634b795e6af3dc75f016aadc23217c65, spanId=17d8aa792d112087, sampled=true
- Create Broadcast
22:40:08.001 [executor-thread-2] INFO  [gr.au.ra.br.in.re.re.BroadcastResource] traceId=4f34f6a330e700eeb4d17f3022df1805, spanId=03e27cb55c7dafa3, sampled=true
- Falling back to createConcurrentFallback() - failed to create broadcast too many requests
22:40:08.001 [executor-thread-4] INFO  [gr.au.ra.br.in.re.re.BroadcastResource] traceId=342a4731e60484a31357716fa637e26b, spanId=94a3c12585a4cfcea, sampled=true
- Create Broadcast
22:40:08.001 [executor-thread-1] INFO  [gr.au.ra.br.in.re.re.BroadcastResource] traceId=7f86f7468fd503e451d3727650c1e7d5, spanId=cfc1c9961ea8bda4, sampled=true
- Create Broadcast
```

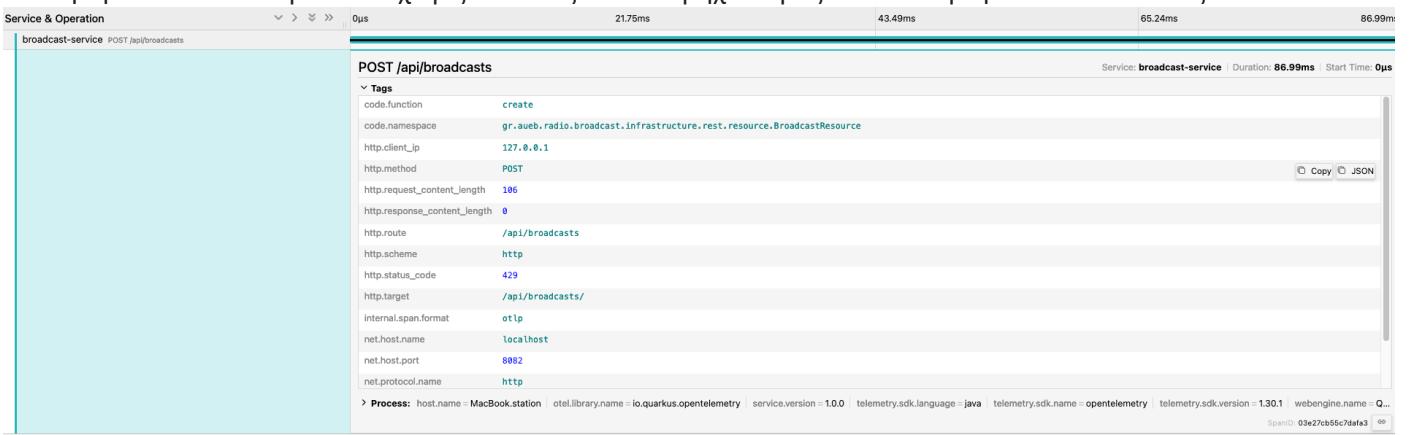
Jaeger:

Επιτυχημένες κλήσεις έγιναν 3





4^η κλήση: BulkheadException διαχειρίζεται ένας fallback μηχανισμός που επιστρέφει κωδικό λάθους 429



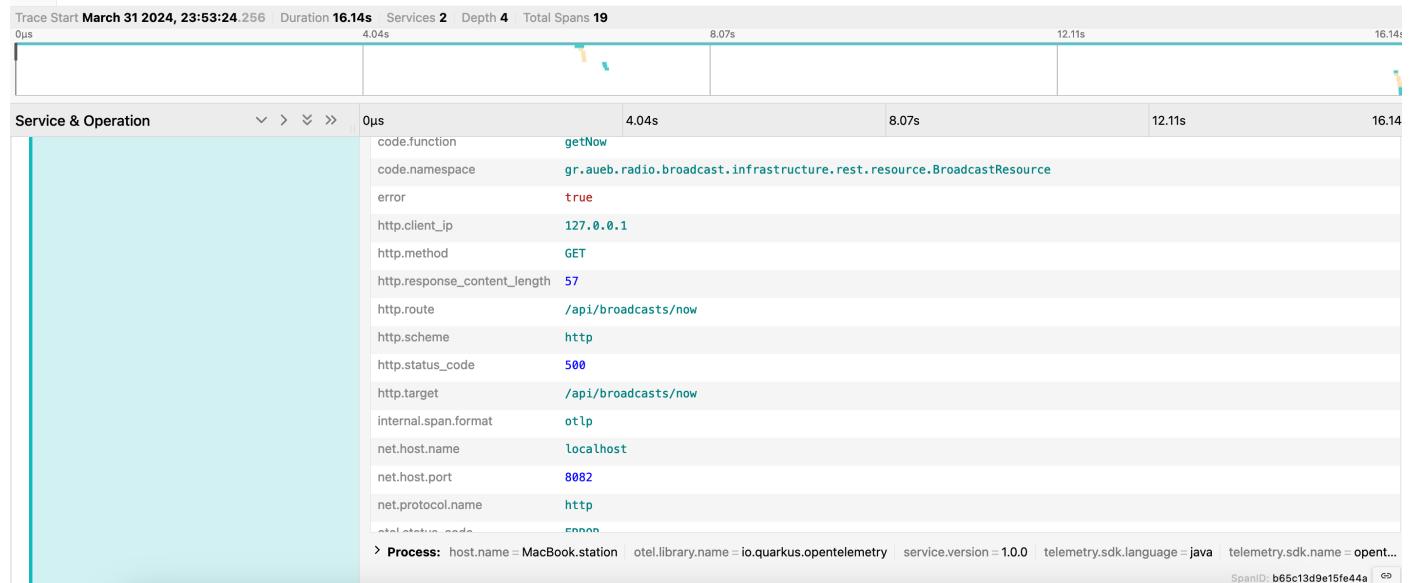
Scenario 8:

Το σύστημα υποστηρίζει την εύρεση της εκπομπής που μεταδίδεται τώρα στην περίπτωση που το επιθυμεί ο χρήστης μέσω του endpoint GET /broadcasts/now. Στο σημέρι αυτό, έχει τοποθετηθεί timeout μηχανισμός για διάστημα 5 δευτερολέπτων, συνεπώς εξασφαλίζεται ότι θα αποδεσμευτεί ο πόρος σε εύλογο χρονικό διάστημα και το σύστημα ξαναπροσπαθεί να επιστρέψει την τρέχουσα εκπομπή, μέχρι να αποτύχει και να διαχειριστεί το timeout μέσω fallback μηχανισμού.

Logs:

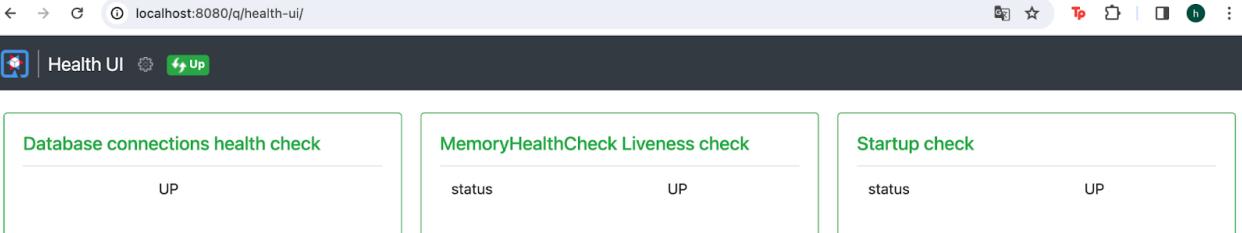
```
23:53:24.350 [executor-thread-1] INFO  [gr.au.ra.br.in.re.re.BroadcastResource] traceId=bde76939fd5de500764338e37d88bc8b, spanId=b65c13d9e15fe4
4a, sampled=true - GetNow invocation #1
23:53:24.351 [executor-thread-1] INFO  [gr.au.ra.br.in.re.re.BroadcastResource] traceId=bde76939fd5de500764338e37d88bc8b, spanId=b65c13d9e15fe4
4a, sampled=true - GetNow has delay
23:53:24.352 [executor-thread-1] INFO  [gr.au.ra.br.in.re.re.BroadcastResource] traceId=bde76939fd5de500764338e37d88bc8b, spanId=b65c13d9e15fe4
4a, sampled=true - GetNow thread sleep 10000
23:53:34.261 [executor-thread-1] INFO  [gr.au.ra.br.in.re.re.BroadcastResource] traceId=bde76939fd5de500764338e37d88bc8b, spanId=b65c13d9e15fe4
4a, sampled=true - GetNow invocation #2
23:53:34.264 [executor-thread-1] INFO  [gr.au.ra.br.in.re.re.BroadcastResource] traceId=bde76939fd5de500764338e37d88bc8b, spanId=b65c13d9e15fe4
4a, sampled=true - GetNow has delay
23:53:34.265 [executor-thread-1] INFO  [gr.au.ra.br.in.re.re.BroadcastResource] traceId=bde76939fd5de500764338e37d88bc8b, spanId=b65c13d9e15fe4
4a, sampled=true - GetNow thread sleep 10000
23:53:40.345 [executor-thread-1] INFO  [gr.au.ra.br.in.re.re.BroadcastResource] traceId=bde76939fd5de500764338e37d88bc8b, spanId=b65c13d9e15fe4
4a, sampled=true - Falling back to fallbackGetNow() - Service unavailable, please try again later
```

Jeager:



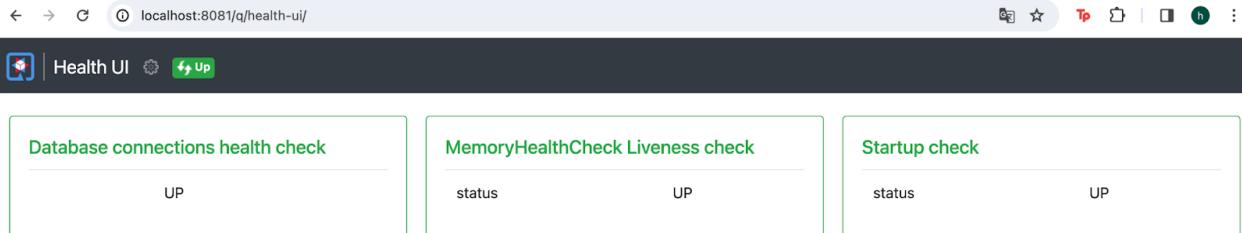
Health checks:

User



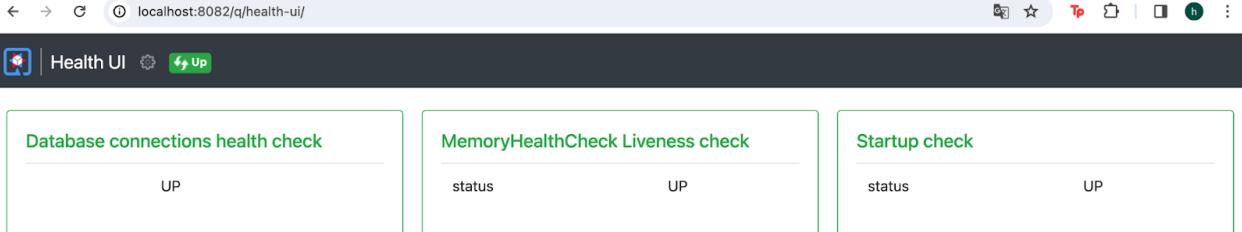
Database connections health check	status	UP
MemoryHealthCheck Liveness check	status	UP
Startup check	status	UP

Content



Database connections health check	status	UP
MemoryHealthCheck Liveness check	status	UP
Startup check	status	UP

Broadcast



Database connections health check	status	UP
MemoryHealthCheck Liveness check	status	UP
Startup check	status	UP

Τα database connection health checks, χρησιμοποιεί το default @Readiness που κοιτάζει αν είναι up η βάση.

Το memory health check χρησιμοποιεί το @Liveness και κοιτάζει αν η ελεύθερη μνήμη του κάθε server είναι τουλάχιστον 10 mb.

Το Start up health check χρησιμοποιεί το @StartUp και σε κάθε ένα microservice κοιτάζει αν παίρνει response από ένα από query στη βάση

- Για το user-service -> αν το `findByName("test") == null` τότε είναι up
- Για το content-service -> αν το `songRepository.findById(-1) == null` τότε είναι up
- Για το broadcast-service -> αν το `broadcastRepository.findById(-1) == null` τότε είναι up