
AWS
re:Invent

CTD15

How Rovio Uses Amazon CloudFront for Secure API Acceleration

Mika Linnanoja
Senior Cloud Engineer
Rovio Entertainment Corporation

AWS re:Invent

aws

Agenda

- Me & Rovio
- What we had
- Why do this?
- Solution
- Challenges and findings
- Results
- Q & A

Me & Rovio

- 12 years in ICT
- Background in QA, releasing, Continuous Integration
- Keep Rovio Games cloud services platform up and running 24x7x365

Rovio

- Games first entertainment company
- Angry Birds
- Founded 2003 in Finland
- ~400 employees in FI, SWE, UK, US & CN

What we had

- Fully cloud native REST API based server infrastructure on AWS EC2 accessed by all Rovio games from all around the globe
- ~1000 instances altogether
- Single public endpoint URL with 3rd party Load Balancer & Proxy, other instances in private subnets
- DNS pointing directly to Elastic IPs of LB cluster instances, external health monitoring

Why do this?

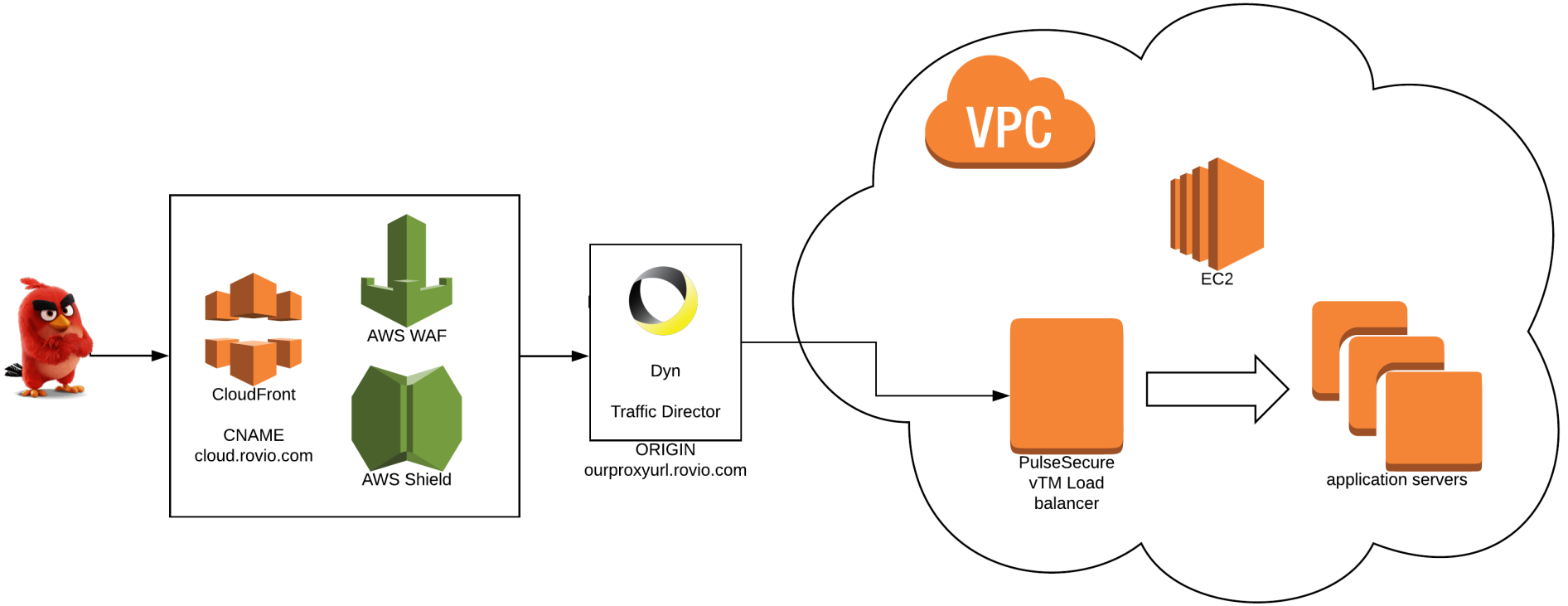
- What if somebody decides to attack our platform?
- No way to mitigate any kind of distributed denial of service attack
- Cannot scale LB cluster endlessly. License limit in max bandwidth & database considerations
- Worst case: Complete loss of business, no in-app purchases nor ads in any game
- We have been *extremely* lucky

Solution: AWS edge services

- Talk with our AWS Solutions Architect => How about you try CloudFront CDN + AWS Shield Advanced + WAF?

- Benefits not just DDoS mitigation (Which we sold this to our management team):
  - Strong DDoS protection on 1st connection point outside our VPC
  - Reduced latencies in API usage i.e. acceleration
  - Possible cost savings due to caching of some responses and reduced data transfers
  - Future proofing our stack: HTTP/2 + IPv6 out of the box

Solution: What



Solution: How

- New CloudFront distribution. Tech known due to earlier use as static media CDN
- Control DNS for origin server (A records)
- Figure out optimal CloudFront settings for dynamic API calls
- Enable AWS Shield & WAF Web ACL to protect and filter unwanted accesses
- Remove extra traffic filtering and access control from existing load balancer
- A way to roll back quickly in case of unforeseen consequences

Challenges and findings

- In production use since February 2018 – going strong!
- No game to server e2e latency and connection quality measurements in place (yet)
- Think of gradual rollout - per game, region or client application
- Adding in effect another proxy in front of our existing proxy brought light to a HTTP 100 Continue bug => hotfix for load balancer
- Pricing model with high request rates and volumes
- Latency improvements are real for us based on our testing scenario

Latency measurements

```
$ ./pingtestanalyzer.py
HTTP GET mean latencies from [ EU Ireland (eu-west-1) ]
- smoke.rovio.com (Via CloudFront)
274.1 ms [50249 tests]
- <ORIGINURL>.rovio.com (Direct)
508.1 ms [49159 tests]

HTTP GET mean latencies from [ US North Virginia (us-east-1) ]
- smoke.rovio.com (Via CloudFront)
147.6 ms [50870 tests]
- <ORIGINURL>.rovio.com (Direct)
101.9 ms [51106 tests]

HTTP GET mean latencies from [ US Oregon (us-west-2) ]
- smoke.rovio.com (Via CloudFront)
323.1 ms [49950 tests]
- <ORIGINURL>.rovio.com (Direct)
514.2 ms [49099 tests]

HTTP GET mean latencies from [ ASIA Singapore (ap-southeast-1) ]
- smoke.rovio.com (Via CloudFront)
558.9 ms [48877 tests]
- <ORIGINURL>.rovio.com (Direct)
1316.6 ms [45607 tests]

HTTP GET mean latencies from [ SOUTH AMERICA Sao Paulo (sa-east-1) ]
- smoke.rovio.com (Via CloudFront)
511.4 ms [49084 tests]
- <ORIGINURL>.rovio.com (Direct)
764.2 ms [47931 tests]
```

- EU Ireland (eu-west-1):
  46% decrease
- US East (us-east-1, same datacenter):
  45% increase
- US West us-west-2:
  37% decrease
- Asia Singapore ap-southeast-1:
  58% decrease
- South America sa-east-1:
  33% decrease

=> Globally helps significantly

Q & A – Ask me anything

aws re:Invent

aws

Extra Materials for Q & A

Mandate to improve our platform

- Technology choices and implementation of the low level infra fully in team hands
- Possible extra costs involved => Buy-in from mgmt and technology organization: DDoS protection!
- ”Do whatever you want to as long as uptime stays at 99.99 %”

QA for CloudFront distribution

- Rigorous testing using all available test automata in our CI & CD system, thousands of testcases
- cURL/httpie became our best friends – service testing endpoint that returns received HTTP headers is nice
- Custom game build that connects to new endpoint from our service platform SDK
- Dummy user testing using subcontractor

Testing tools: stats

```
$ httpstat --http1.1 https://smoke.rovio.com/adsdepot/1/tests/headers
Connected to 54.192.97.35:443 from 192.168.250.2:50759

HTTP/1.1 200 OK
Content-Type: application/json;charset=utf-8
Content-Length: 435
Connection: keep-alive
Date: Tue, 09 Oct 2018 11:15:18 GMT
Request-Id: 6357629814916248092
X-Cache: Miss from cloudfront
Via: 1.1 f9a0ddc3860252ab6c4d02ab024b4891.cloudfront.net (CloudFront)
X-Amz-Cf-Id: Iof8jAqRQGMmECs4C9SS8Ml9c9UUCOjIC7pfsiW3ZpS3SYhH2kGgSQ==

Body stored in: /var/folders/6l/22zs2p7d6j31y812024wr2b00000gp/T/tmpRtmO83

  DNS Lookup   TCP Connection   Server Processing   Content Transfer
[     5ms    |      21ms      |       128ms       |        1ms       ]
             |                |                   |                  |
    namelookup:5ms            |                   |                  |
                        connect:26ms              |                  |
                                      starttransfer:218ms            |
                                                                 total:219ms
```

httpstat

Testing tools: headers

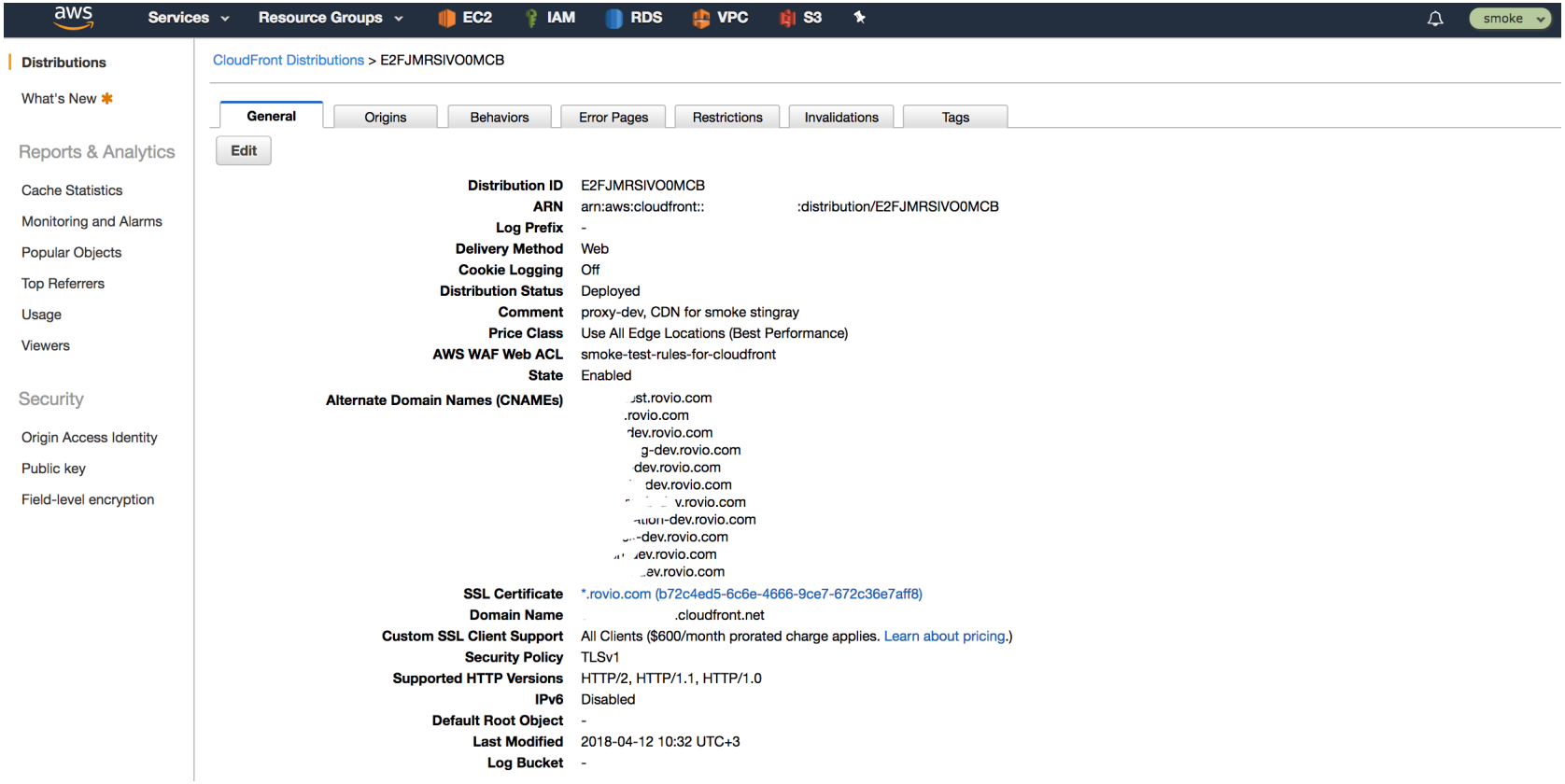
```
$ http https://smoke.rovio.com/adsdepot/1/tests/headers
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 470
Content-Type: application/json;charset=utf-8
Date: Tue, 09 Oct 2018 11:16:43 GMT
Request-Id: 1575855414616325623
Via: 1.1 b475a5f7d95ff68ca0dc588e3c9a3231.cloudfront.net (CloudFront)
X-Amz-Cf-Id: VQoA8i_G3wuD_TOs8TlgAvkGz-aR01BRXPB_Dgw_9PuSShXx2IRAMg==
X-Cache: Miss from cloudfront

{
    "bodyForm": null,
    "headers": {
        "Accept": "*/*",
        "Accept-Encoding": "gzip, deflate",
        "Connection": "keep-alive",
        "Host": "smoke.rovio.com",
        "User-Agent": "HTTPie/0.9.9",
        "X-Cluster-Client-Ip": "205.251.218.157",
        "X-Forwarded-For": "[redacted], 205.251.218.157",
        "X-GeoIP-City-Name": "Helsinki",
        "X-GeoIP-City-Name-Base64": "SGVsc2lua2k=",
        "X-GeoIP-Country-Code": "FI",
        "X-GeoIP-Country-Name": "Finland",
        "X-GeoIP-State-Code": "13",
        "X-Is-Https": "true",
        "X-Real-IP": "[redacted]
    },
    "parameters": {}
}
```

httpie client

API Acceleration details

- In theory: CloudFront keeps connections alive towards Origin server if connections from each PoP more frequent than HTTP Keep-Alive
  => in a high volume global users scenario savings in e.g. TLS negotiation
- Extra tidbit: Due to Keep-Alive memory and CPU utilization at EC2 load balancer solution dropped very significantly
  => smaller instances OK
- Works for our case: lots of small payload HTTP requests (JSON), no binary data e.g. image uploads/downloads via APIs

Technical Setup: CF

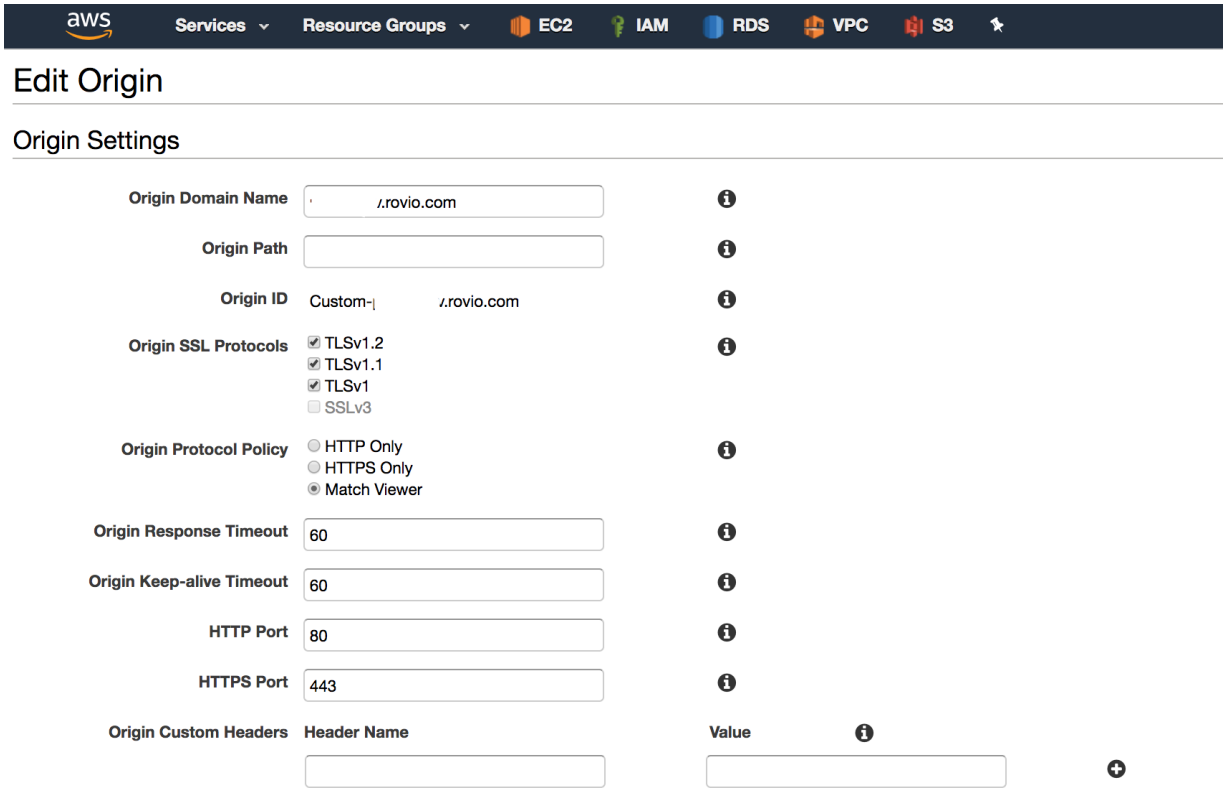


Technical Setup: CF

Relevant main details for cloudfront distribution

- Custom SSL cert for HTTPS access (imported)
- SNI in use due to legacy game clients
- Alternate Domain Names (CNAMEs) for all subdomained admin websites
- All TLS protocols
- HTTP/2 enabled
- IPv6 disabled due to geo mapping issues

Technical Setup: CF

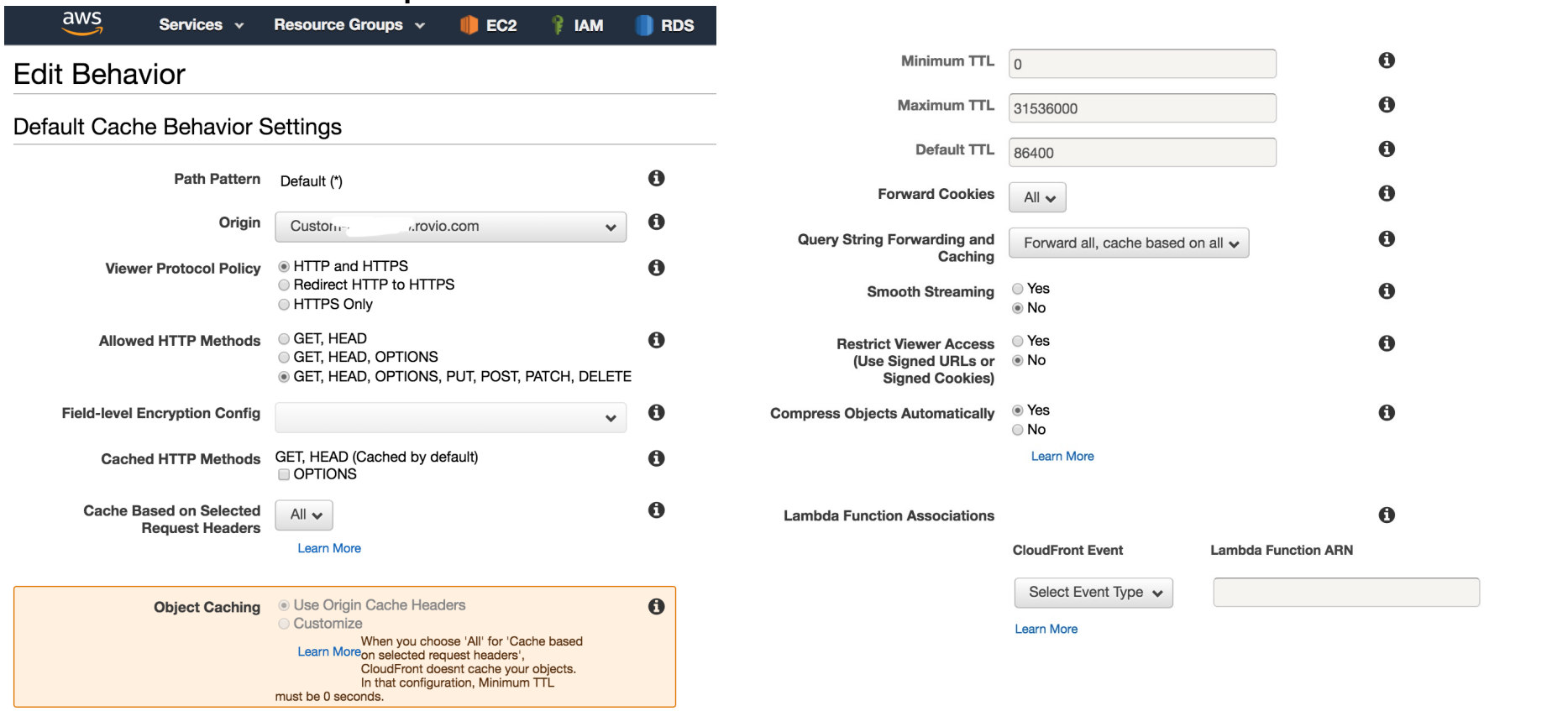


Technical Setup: CF

Origin details

- TLS v1.0, 1.1, 1.2 - waiting for 1.3
- Protocol selection as match viewer – both HTTP and HTTPS cases
- Maximum response timeout 60 s
- Maximum keep-alive 60 s – IMPORTANT
- Default ports
- No extra header adding

Technical Setup: CF



aws

Technical Setup: CF

Behaviors setup

- Protocols separately
- All HTTP methods – IMPORTANT
- “Cache based on selected request headers”: All - IMPORTANT
- Above for disabling caching for API traffic
- Forward all cookies so admin website logins work
- Compression (gzip) enabled for all traffic

Cache Stats (prod)

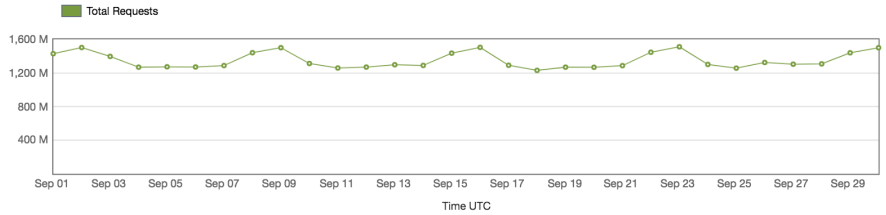
CloudFront Cache Statistics Reports

Start Date 2018-09-01 Granularity Daily (any period in previous 60 Web Distribution EZ873V07KXYX1 (cloudtest.rovi

End Date 2018-09-30 Viewer Location All Locations Update CSV

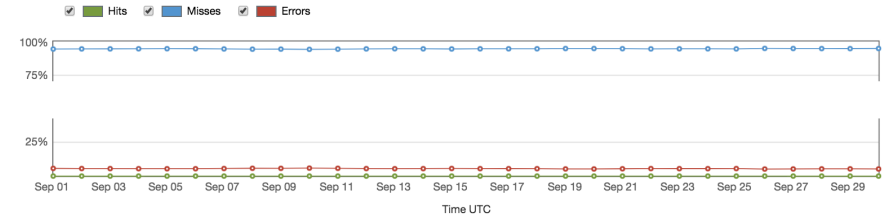
The following charts show selected values from CloudFront access logs. In the access logs for a distribution, each row corresponds with one viewer request. If you choo
your web distributions that had activity during the specified period and that have not been deleted. Data for deleted distributions and RTMP distributions is not available.
viewers making the request while Usage Reports are based on CloudFront Billing Region.

Total Requests (Millions | Thousands | Not Scaled) Show Details



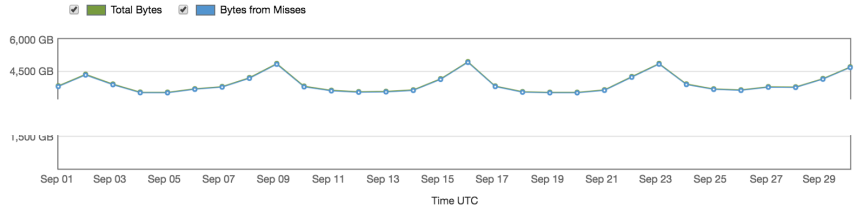
Average: 1,348.3657 M **Total:** 40,450.9705 M **Maximum:** 1,510.2852 M **Minimum:** 1,230.0644 M

Percentage of Viewer Requests by Result Type Show Details



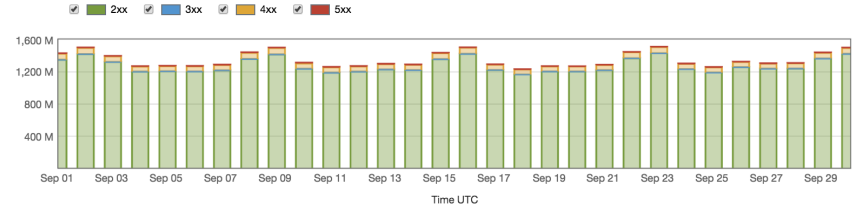
Total Hits: 0 (0%) **Total Misses:** 38,216.17 M (94.48%) **Total Errors:** 2,234.80 M (5.52%)

Bytes Transferred to Viewers (Gigabytes | Megabytes | Kilobytes) Show Details



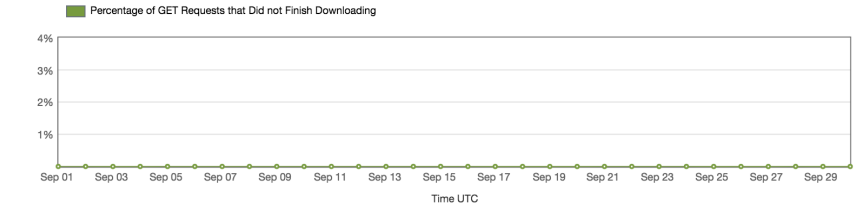
Total Bytes: 117,832.4797 GB **Total Bytes from Misses:** 116,883.1125 GB

HTTP Status Codes (Millions | Thousands | Not Scaled) Show Details



Total 2xx: 38,204.8206 M (94.49%) **Total 3xx:** 14.2417 M (0.04%) **Total 4xx:** 2,028.0383 M (5.02%) **Total 5xx:** 185.1672 M (0.46%)

Percentage of GET Requests that Did not Finish Downloading Show Details

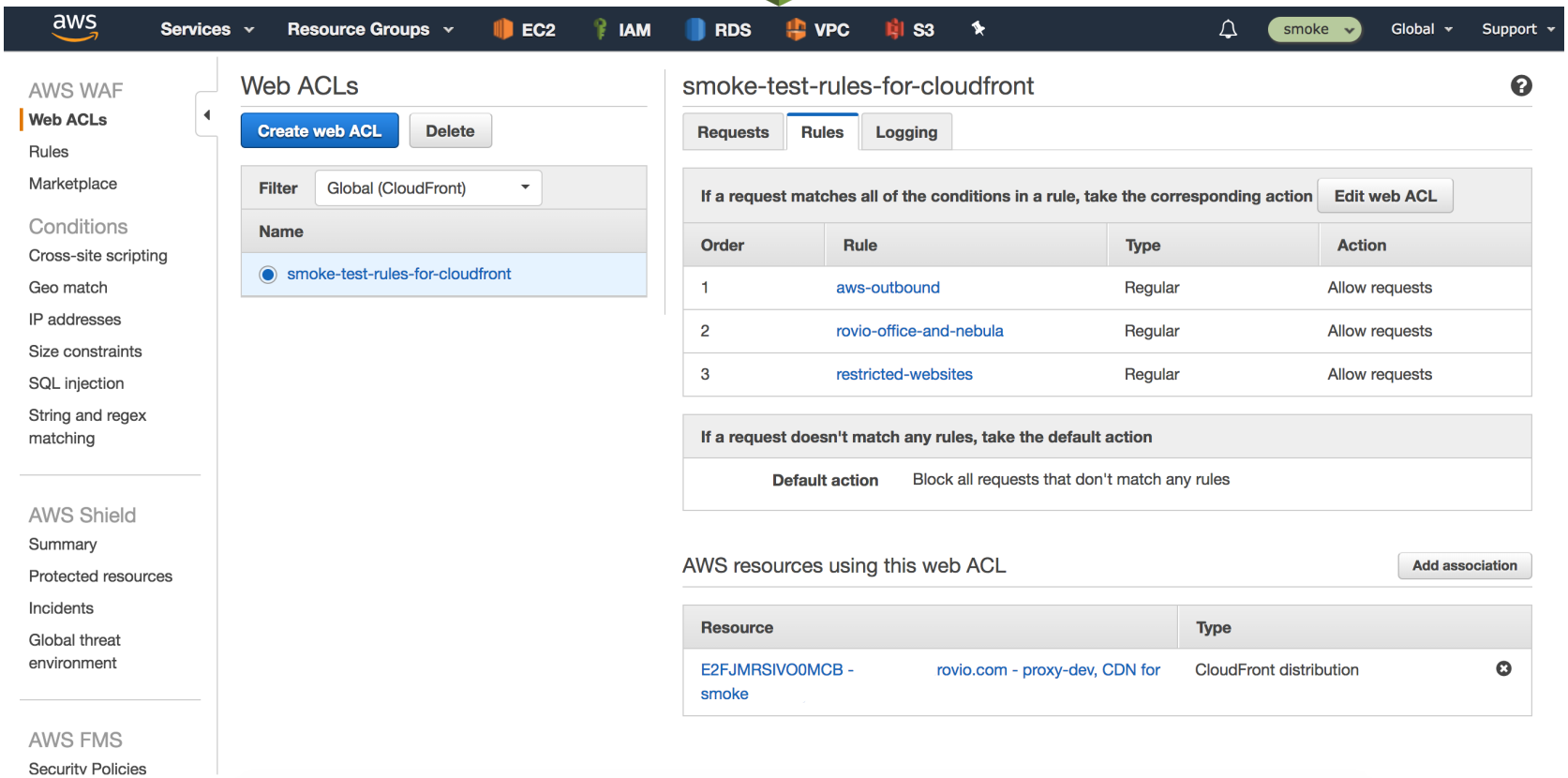


Cache Stats (prod)

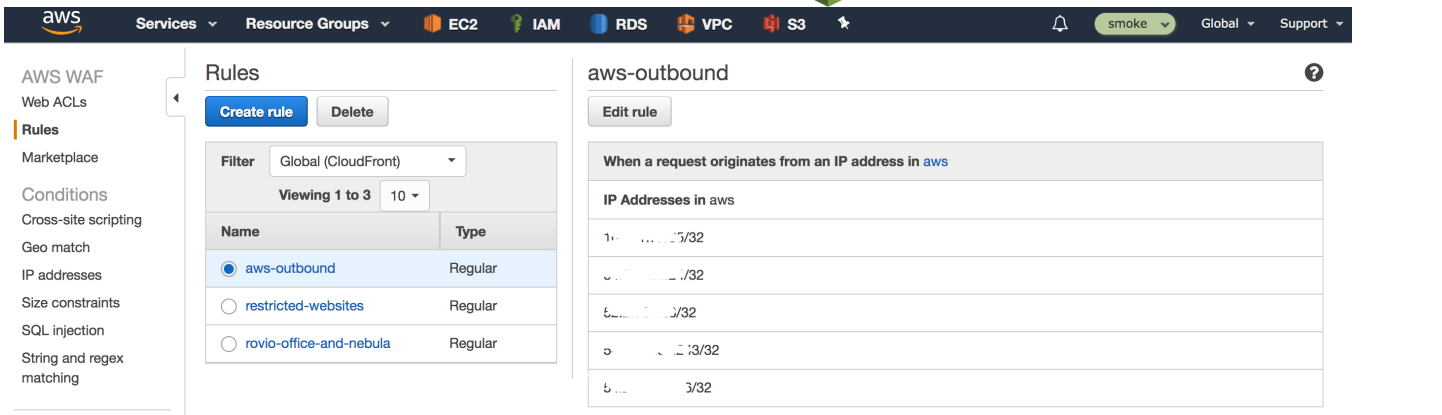
- September stats
  - 40 billion requests in month => ~ 15000 requests / second
  - 117 000 GB transferred => ~ 400 Mbps
- Estimations prior to production use
  - ~20000 requests / second
  - 200 000 GB / month

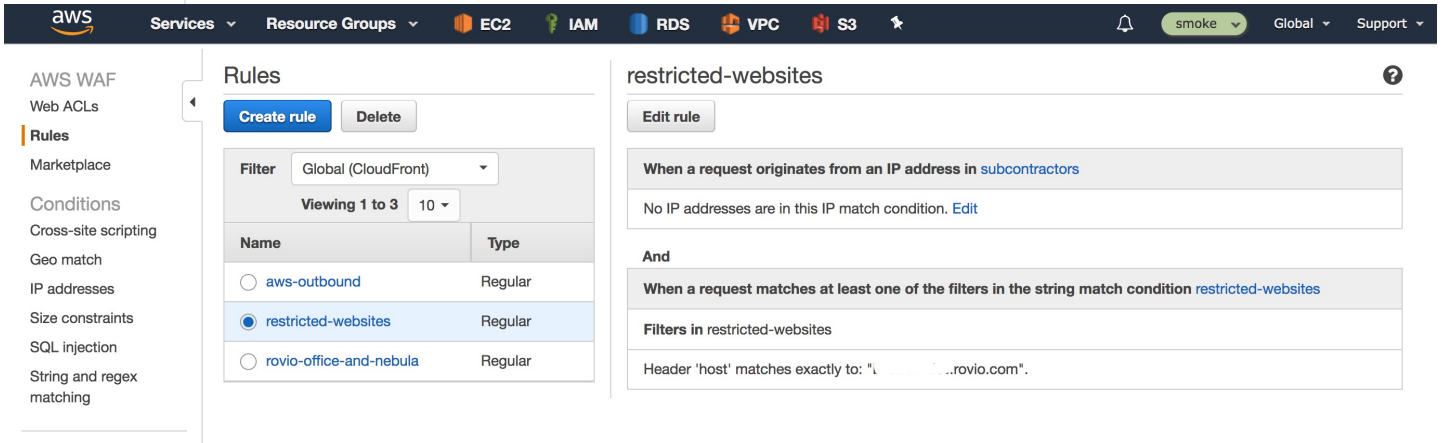
=> Not trivial to get exact incoming/outgoing traffic figures to your endpoint

Technical Setup: WAF

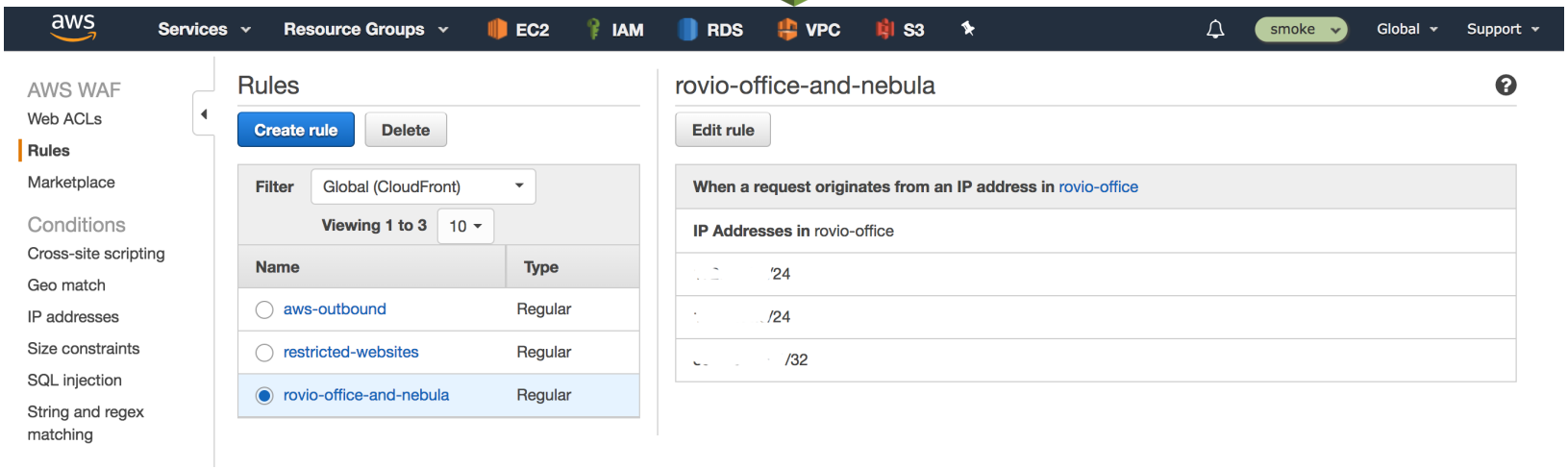


Technical Setup: WAF





Technical Setup: WAF



Technical Setup: WAF 

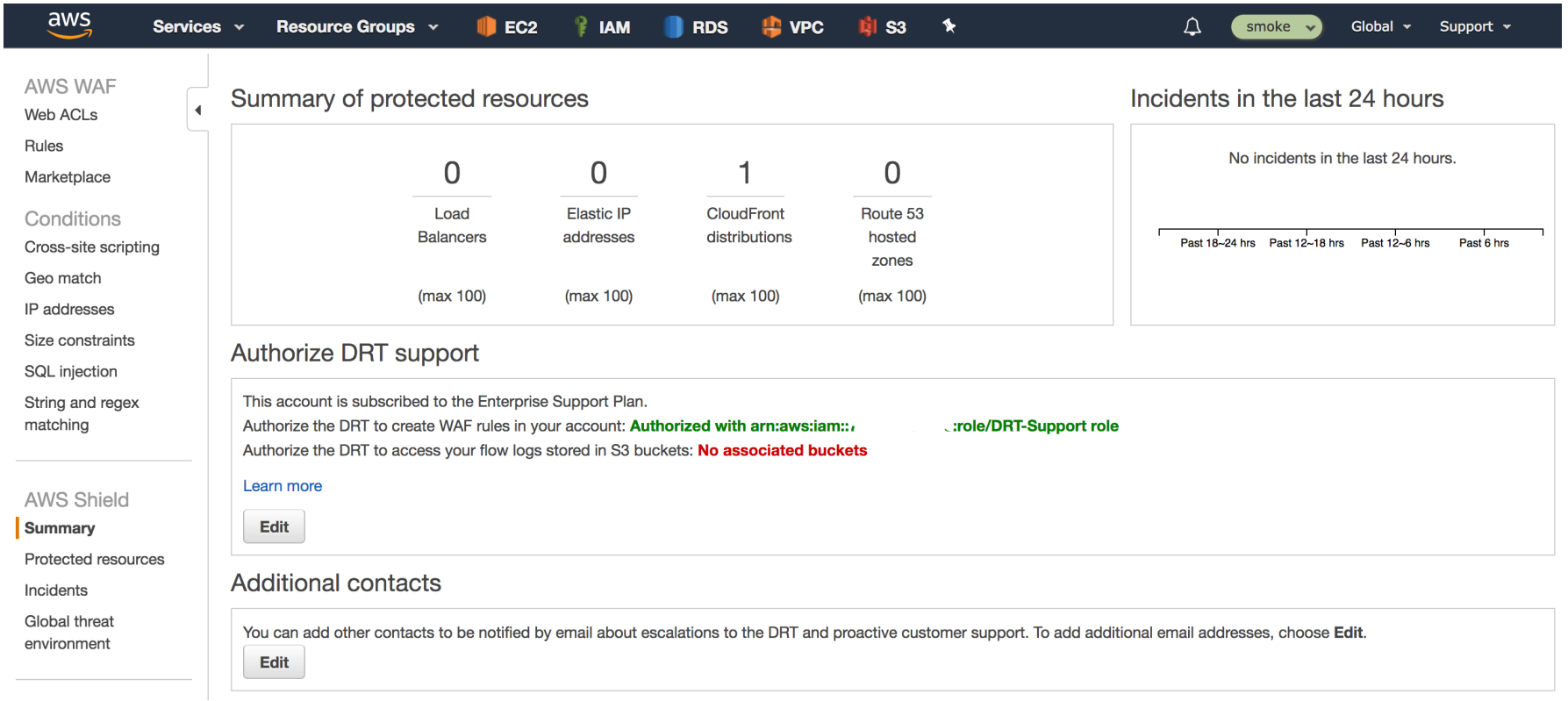
WAF Setup

- Single Web ACL with a few rules linked to CF distribution

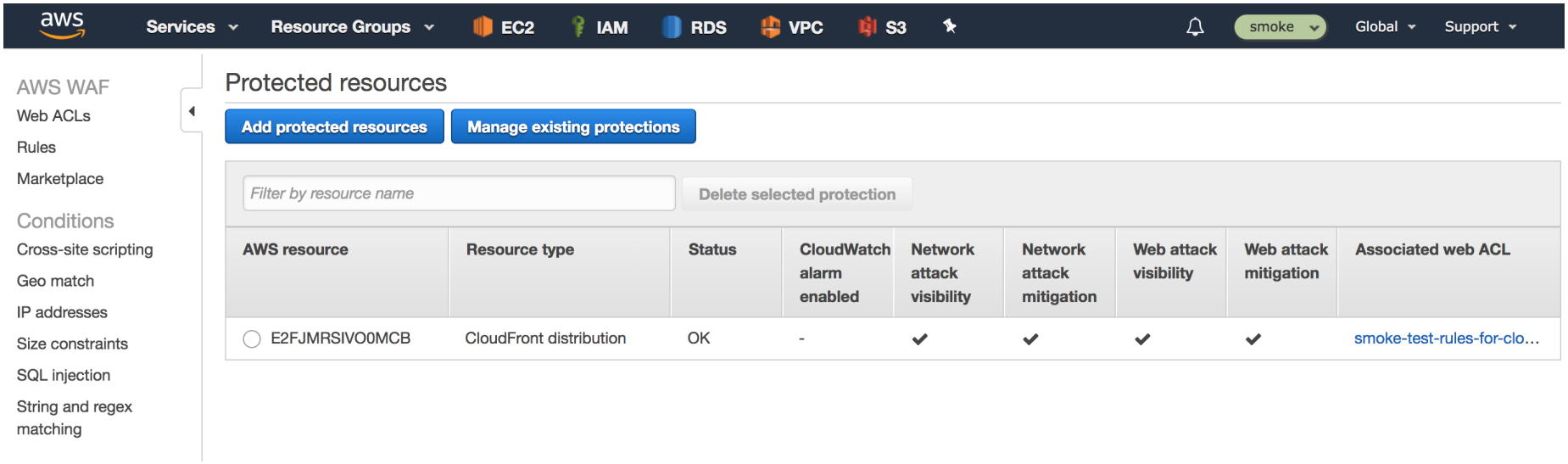
- Dev
  - block all requests by default
  - allow traffic in from few knows IPs (AWS + office)
- Stage and Prod
  - allow all requests by default
  - block restricted endpoints unless from known IPs

- Very simple IP and host header based checks

Technical Setup: Shield Advanced



Technical Setup: Shield Advanced



Technical Setup: Shield Advanced

Shield Advanced

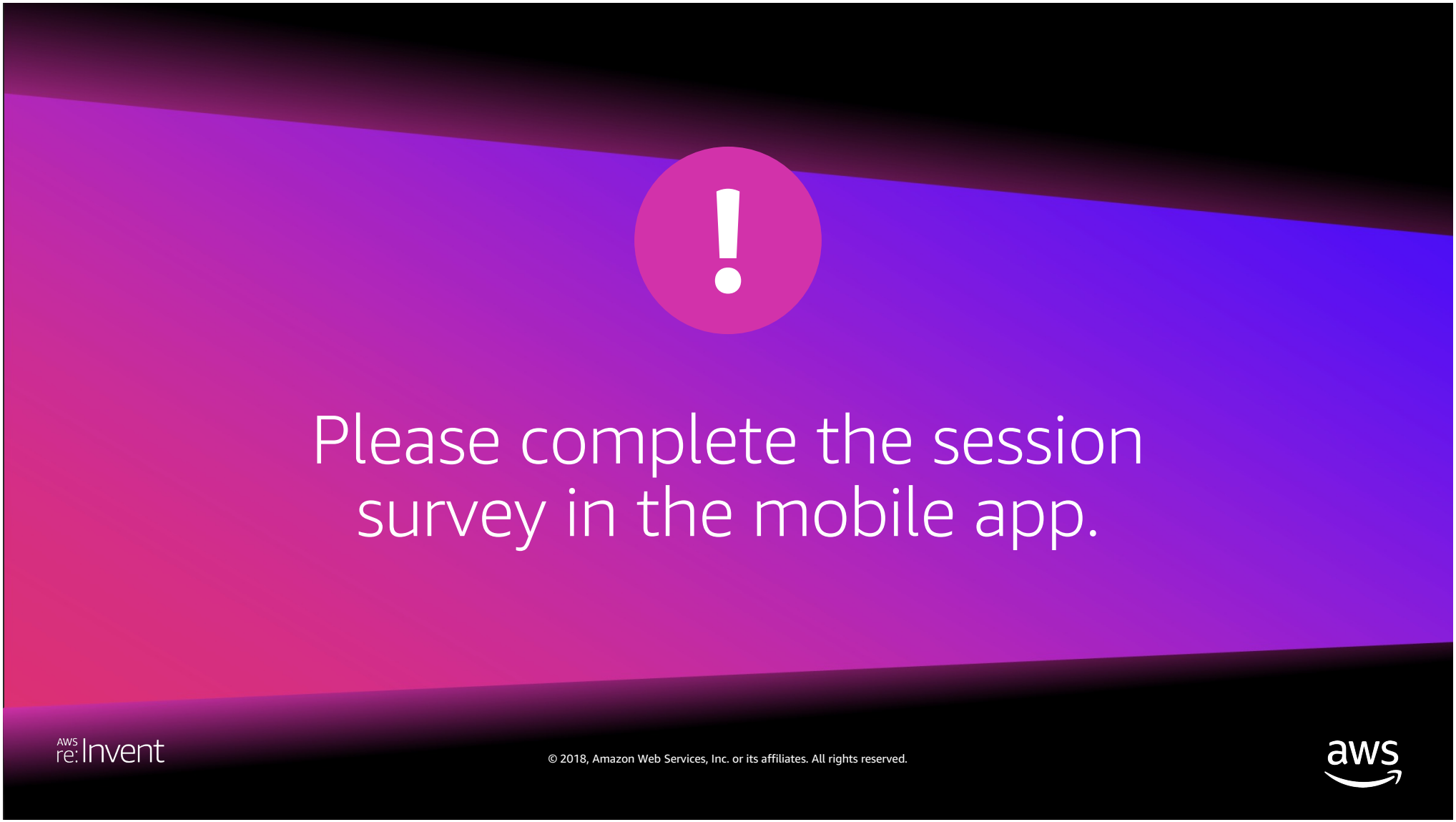
- Enable once per organization umbrella
- Subscription fee once, data traffic per usage
- Toggle on for all supported resources – CF, ALB/ELB, EIP
- DDoS Response Team (DRT) Cloudformation stack

Thank you!

Mika Linnanoja
mika.linnanoja@rovio.com

AWS re:Invent

aws


Please complete the session survey in the mobile app.
AWS re:Invent
© 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
aws