
Forta Web Security Assessment - Audit Report

January 5th, 2022

Provided to:



by:



www.dedalo.io - info@dedalo.io

Table of Contents

OVERVIEW	3
SCOPE	3
EXECUTIVE SUMMARY	5
TECHNICAL REPORT	6
SECURITY ISSUES	6
#01 - Weak validation of input parameters leads to enable or disable users Agents7	
#02 - Weak validation of input parameters leads to update any Developer profile	9
#03 - Weak validation of input parameters leads to update any Agent	12
#04 - GraphQL Introspection enabled in production environment	16
#05 - Multiple endpoints with wildcard CORS activated	19
#06 - Weak validation of input parameters allows arbitrary agentId values	21
#07 - Sensitive Information Leak in GraphQL Query	23
#08 - HTML Injection in alerts emails	26

OVERVIEW

Forta engaged Dédalo's team to conduct a white-box security audit of Forta Explorer and Forta Connect web applications and their source code. This process consisted of identifying possible threats and security risks in the components of Forta web applications and its infrastructure. Dédalo performed source code reviews, manual and dynamic tests with multiple tools and techniques to identify security weaknesses.

Every finding is described in this document, including details that we consider necessary to understand the dangers of their possible impact, along with proper recommendations.

SCOPE

This assessment was done on the source code of “Forta Connect” and “Forta Explorer” applications provided by Forta through a private GitHub repository.

The following components were delivered to Dédalo and have been audited:

- GitHub repositories:
 - doiim/forta-connect-frontend
 - doiim/forta-connect-backend
 - forta-protocol/forta-explorer
 - forta-protocol/forta-explorer-api
- Forta Connect running on <https://forta.doiim.com/>
- Forta Explorer running on <https://explorer.forta.network/>

EXECUTIVE SUMMARY

For this audit, we analyzed all of the systems and applications of Forta that we discovered for the defined scope and identified security issues that could negatively affect Forta's business and users' data. This security assessment was conducted between December 16, 2021, and December 28, 2021, and the following days were used for findings triaging and report elaboration.

We performed a complete code review of the in-scope repositories focusing on user-provided inputs, data management, authorization, and authentication management. Along with code review, we performed multiple dynamic tests to validate previous findings and to find web application security issues.

With various techniques and tools, our team has managed to demonstrate missing authorization controls, weak input validations, sensitive information leaks, GraphQL misconfigurations, among other non-critical issues.

General Recommendations

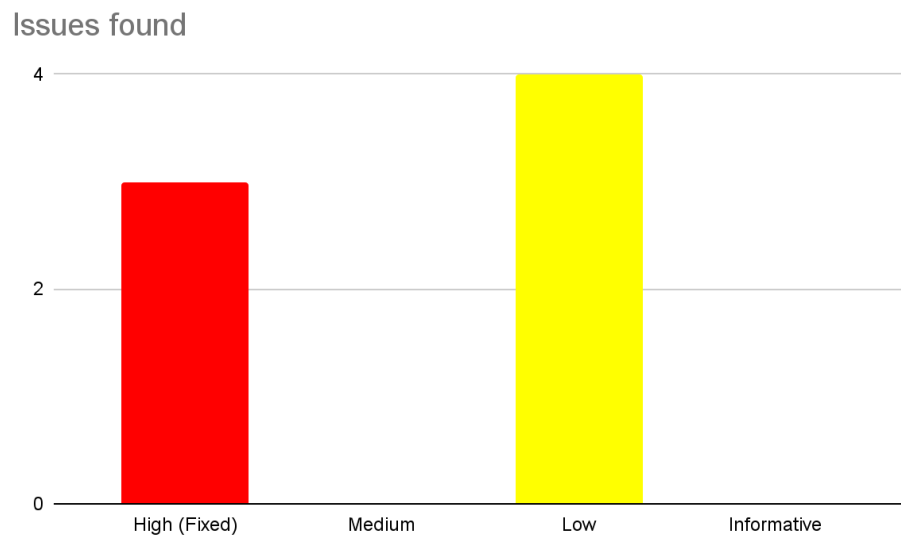
Regarding user-provided Agents, we noticed there is no code review to avoid malicious actions from users. Lack of code verification makes users to not be able to verify if a given Agent is actually running the code referred to the Agent Developer provided repository URL.

While this audit was scoped to only four repositories, we identified several out-of-scope components which are core to the Forta ecosystem and perform sensitive processes with external user-provided data. These components are, but not limited to: Forta Nodes, Agents scheduler, Notifications manager, Agent running environments, a publicly available custom containers registry (Disco). We consider this infrastructure requires a dedicated audit, as this represents a significant surface attack for malicious actors. Weaknesses in these components could result in the following security problems: Agent code tampering, Spam or Denial of Service attacks to the Forta infrastructure or external services exposed to the Internet, Billing exhaustion, etc.

TECHNICAL REPORT

SECURITY ISSUES

We found the following issues classified by severity level:




```

Response
Pretty Raw Hex Render
1 HTTP/2 200 OK
2 Date: Thu, 16 Dec 2021 20:16:37 GMT
3 Content-Type: application/json; charset=utf-8
4 X-Powered-By: Express
5 Access-Control-Allow-Origin: *
6 Etag: W/"9b-vD9s9RFzAwnIjU3Melc9F01M7M"
7 Cf-Cache-Status: DYNAMIC
8 Expect-Ct: max-age=604800,
  report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
9 Report-To:
  {"endpoints":[{"url":"https://a.nel.cloudflare.com/report/v3?s=zuxp6z49bdZ%
  2F3NBc5X%2BQe%2BIOZRpR7yv2nbLKji3VuHrQDi%2FxnC6z9Q0rx1uKw9oIyGztFZs8B0CgDmL7hI3
  ShodsouP4McJH5AMtYf%2F0PH7Eqmh0mAMzPhtVgTVXfxxvg%2Fc%3D"}],"group":"cf-nel","ma
  x_age":604800}
10 Nel: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
11 Server: cloudflare
12 Cf-Ray: 6bea9716afa4098e-MIA
13
:14 {
  "data":{
    "updateAgent":{
      "id":86,
      "agentId":
        "0x8dcd31c00e76ba2f8f6edcc65e51e754ce8381edc6491f70d4724d6a818307cf",
      "status":"DISABLED",
      "__typename":"Agent"
    }
  }
}
15
    
```

e.g.: in this evidence we disable the [Doiim Agent](#)

Recommendations

Proper validations should be added in both frontend and backend. Sanitize values before processing them in the backend.


```

Response
Pretty Raw Hex Render
1 HTTP/2 200 OK
2 Date: Thu, 16 Dec 2021 20:54:18 GMT
3 Content-Type: application/json; charset=utf-8
4 X-Powered-By: Express
5 Access-Control-Allow-Origin: *
6 Etag: W/"186-jTLYp+n4SQESUZwV0a16lFNtL2A"
7 Cf-Cache-Status: DYNAMIC
8 Expect-Ct: max-age=604800,
  report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
9 Report-To:
  {"endpoints":[{"url":"https://a.nel.cloudflare.com/report/v3?s=l%2BovG%2FQ9PrYIHZ8kKE68Weycu4
  JDLKjhtrYQl8ZccWY60BSlwem%2B1x0smseCfoHfv82cv2teLeDIWyKMuJFz2R%2F%2FLugVbbNEkRSJ8RmX6oj2BF56c0S
  ld0%2F7hSUAM3Go5I4%3D"}],"group":"cf-nel","max_age":604800}
10 Nel: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
11 Server: cloudflare
12 Cf-Ray: 6beace5eef4802f9-MIA
13
14 {
  "data":{
    "updateDeveloper":{
      "id":43,
      "name":"Mister",
      "description":"New name",
      "url":"https://checker.yahoo.com/",
      "email":null,
      "twitter":null,
      "facebook":null,
      "linkedin":null,
      "github":null,
      "reddit":null,
      "youtube":null,
      "discord":null,
      "publicAddress":"0xa32c603e4e77a50850b52360f9921b23ca7da025",
      "avatar":"bafkreicrgxx23hswzthg3b5hl4qlrdvdiqrpnghyxw67xgmffziav7aayy",
      "__typename":"Developer"
    }
  }
}
15
    
```

The screenshot shows a web browser window with the URL `forta.doiim.com/developer/0xa32C603e4E77a50850b52360F9921B23Ca7DA025`. The page header includes 'Forta Connect BETA' and navigation links for 'Agents', 'Profiles', 'Explorer', 'SDK docs', and 'Forta.org'. The profile section for 'Mister' features a circular profile picture of a yellow bird, the name 'Mister', and a 'New name' button with a link icon. Below this is a section titled 'Agents by Mister' containing a table of active agents.

AGENT ↑↓	LAST UPDATE / ID ↑↓
doiim-agent	6 minutes ago • 16 Dec 2021, 17:49 0x8dcd31c00e76ba2f8f6edcc65e51e7...
Test-Agent	a month ago • 19 Nov 2021, 17:06 0xb7edb1146b15b6d433fbc3817cc7b...
Agent Test	a month ago • 18 Nov 2021, 17:47 0xd9a4a21749a2ee6a675fa1588b33d4...
Guilherme Agent	a month ago • 17 Nov 2021, 11:06 0xd152c53641ee610c62990b3e5c407...

Recommendations

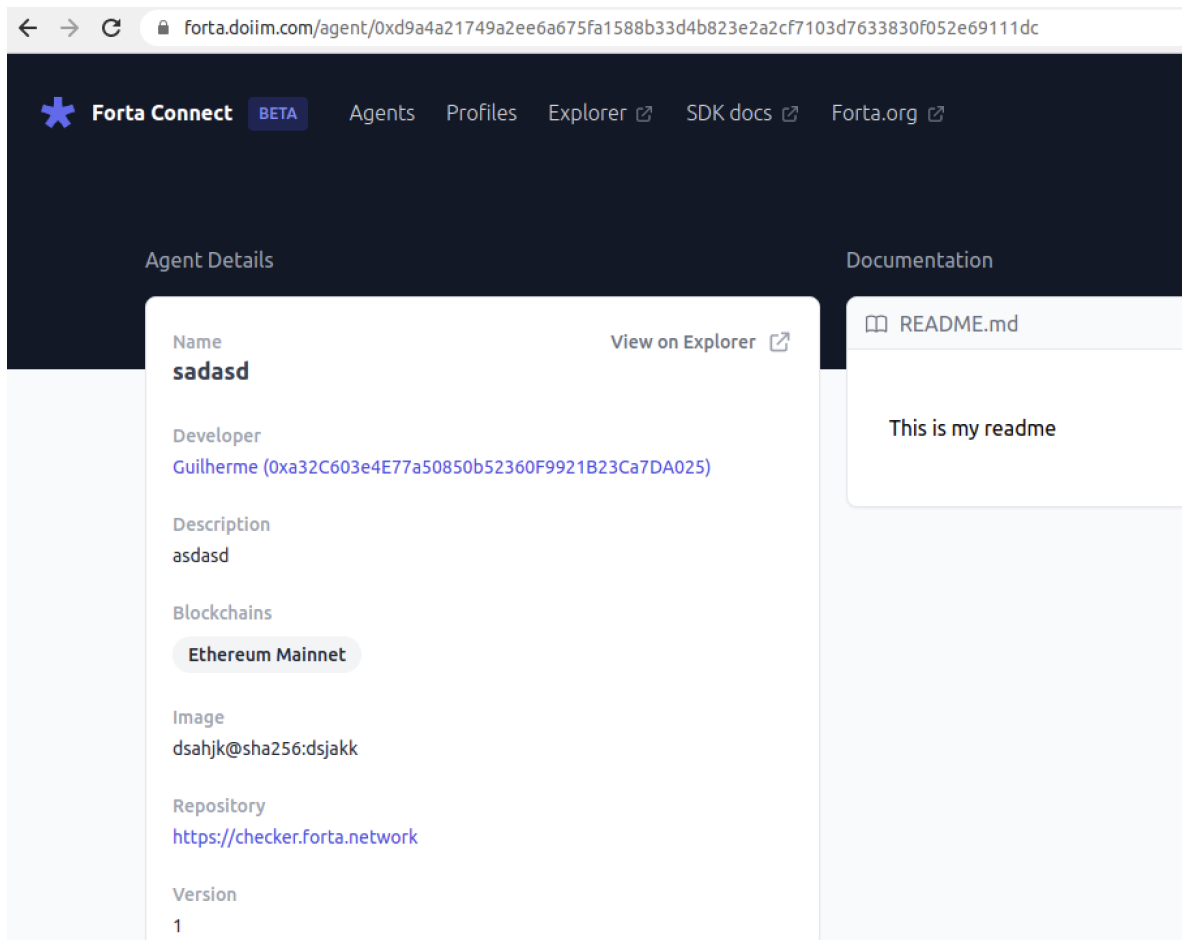
Proper validations should be added in the backend. Correct the “graphql-shield” middleware role checking. Fixes should be done in the following files:

- <https://github.com/doiim/forta-connect-backend/blob/main/src/permissions.ts>
- <https://github.com/doiim/forta-connect-backend/blob/main/src/rules.ts>


```

Response
Pretty Raw Hex Render
1 HTTP/2 200 OK
2 Date: Fri, 17 Dec 2021 18:58:54 GMT
3 Content-Type: application/json; charset=utf-8
4 X-Powered-By: Express
5 Access-Control-Allow-Origin: *
6 Etag: W/"88-/k/4i/8egkBXE+0uGvW5/ay1rBo"
7 Cf-Cache-Status: DYNAMIC
8 Expect-Ct: max-age=604800,
report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
9 Report-To:
{"endpoints":[{"url":"https://a.nel.cloudflare.com/report/v3?s=jDN1J0AImYyb0uvLc0tp%2FSzEWaI
1wkkkjvd12DyxseJEjLFrqTTaYt7giyJrMi4gV2ZRogcfIBDlKgbEd5enprNeIrFJmnpCdDlibfXMaSpXP7Zo6l1EZ37wgriJ
FX10W4%3D"}],"group":"cf-nel","max_age":604800}
10 Nel: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
11 Server: cloudflare
12 Cf-Ray: 6bf262933a51220f-MIA
13
14 {
  "data":{
    "updateAgent":{
      "id":121,
      "agentId":"0xd9a4a21749a2ee6a675fa1588b33d4b823e2a2cf7103d7633830f052e69111dc",
      "__typename":"Agent"
    }
  }
}
15
    
```

Response of the server with the Agent updated.



Agent from another Developer modified.

Recommendations

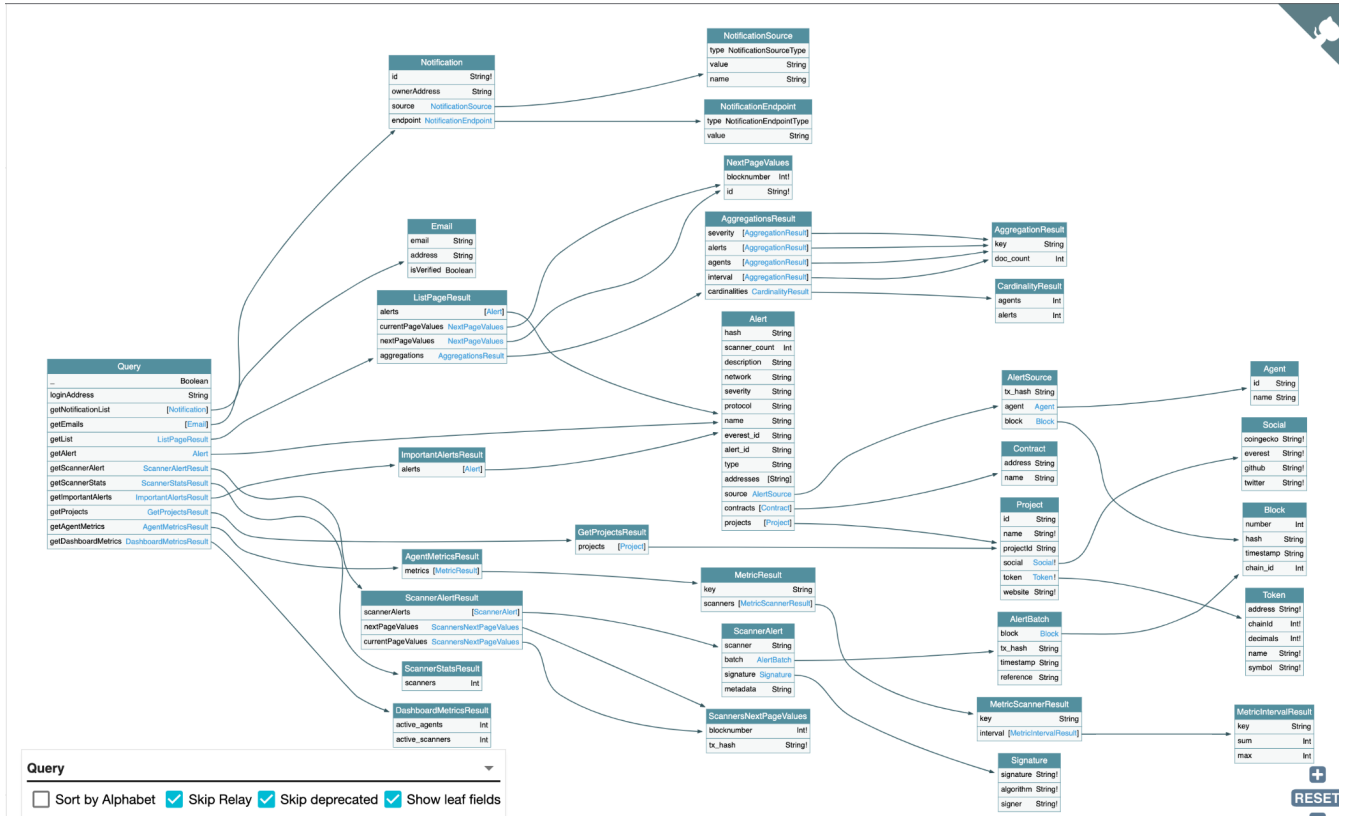
Proper validations should be added in the backend. Correct the “graphql-shield” middleware role checking. Fixes should be done in the following files:

- <https://github.com/doiim/forta-connect-backend/blob/main/src/permissions.ts>
- <https://github.com/doiim/forta-connect-backend/blob/main/src/rules.ts>


```

Response
Pretty Raw Hex Render
1 HTTP/2 200 OK
2 Date: Fri, 17 Dec 2021 22:38:28 GMT
3 Content-Type: application/json; charset=utf-8
4 X-Powered-By: Express
5 Access-Control-Allow-Origin: *
6 Etag: W/"d1bc-7Nl/BatGVTYr34Iv+3e0Qew0WTg"
7 Cf-Cache-Status: DYNAMIC
8 Expect-Ct: max-age=604800,
report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
9 Server: cloudflare
10 Cf-Ray: 6bf3a44bcba908b8-EZE
11
12 {
  "data":{
    "_schema":{
      "queryType":{
        "name":"Query"
      },
      "mutationType":{
        "name":"Mutation"
      },
      "subscriptionType":{
        "name":"Subscription"
      },
      "types":[
        {
          "kind":"OBJECT",
          "name":"Query",
          "description":null,
          "fields":[
            {
              "name":"_",
              "description":null,
              "args":[

```

Recommendations

Disable Debug mode on production environments

#05 - Multiple endpoints with wildcard CORS activated

Severity: Low

Description

Some endpoints on different components of the application have permissive CORS policies configured with wildcards.

Clients from any browser can access these methods without domain restrictions.

Backend components are:

https://explorer-api.forta.network	Explorer API
https://forta.doiim.com/api/*	Forta Connect backend

Impact

An attacker could craft arbitrary cross-origin requests and exploit this permissive misconfiguration to act on behalf of the victim.

Evidence

As an example, this is a response to a POST request on fortadoiim.com. Regardless of the Origin domain, the server allows the request, and then the response could be captured from a web browser context.

```

Request
-----
1 POST /api/graphql HTTP/2
2 Host: fortadoiim.com
3 Content-Length: 288
4 Origin: https://attacker.com
5 Referer: https://attacker.com
6 Sec-Ch-Ua: " Not A;Brand";v="99", "Chromium";v="96",
  "Google Chrome";v="96"
7 Accept: */*

Response
-----
1 HTTP/2 200 OK
2 Date: Thu, 06 Jan 2022 03:38:55 GMT
3 Content-Type: application/json; charset=utf-8
4 Content-Length: 30
5 X-Powered-By: Express
6 Access-Control-Allow-Origin: *
7 Etag: W/"1e-cf8j4pYEnmurIkjJgu09iO3j6ZU"
8 Cf-Cache-Status: DYNAMIC
    
```

Recommendations

We recommend settling CORS configuration to limit access to specific front-end domains to prevent other websites from accessing Forta APIs.

#06 - Weak validation of input parameters allows arbitrary agentId values

Severity: **Low**

Description

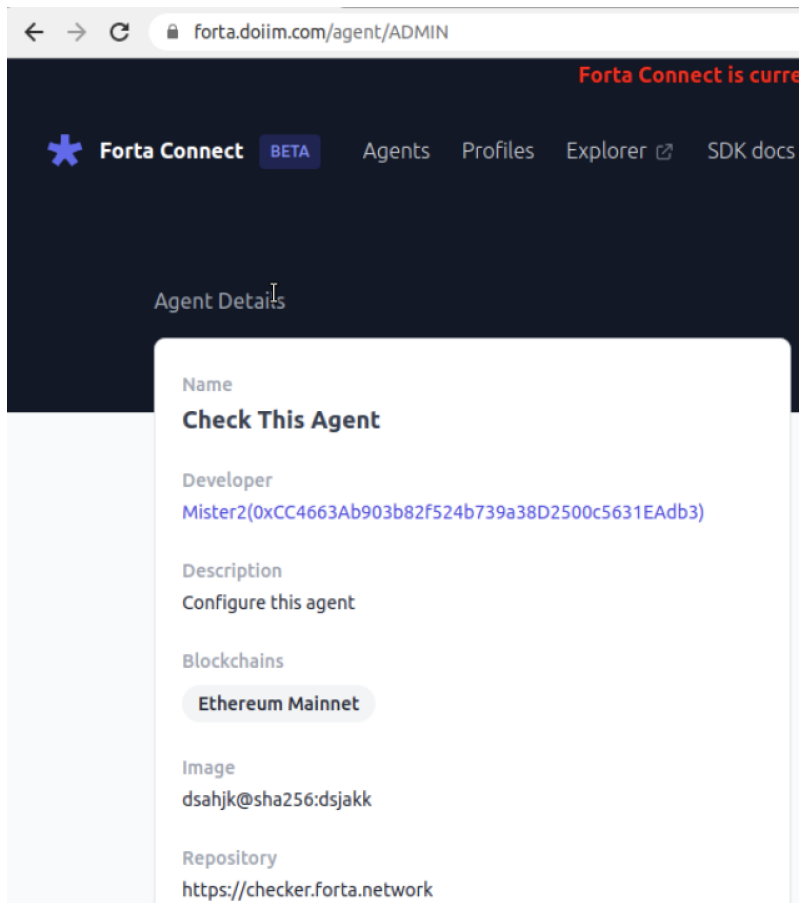
Any authenticated user is allowed to **craft** an arbitrary agentId without the common pattern (0x*) when creating a new **Agent**.

An attacker could manipulate the agentId input and put any value to the Agent Identification, therefore impacting in the Agent URL alias.

Evidence

```

Request
Pretty Raw Hex ↵ ↵ ⋮ Select extension... ▼
1 POST /api/graphql HTTP/2
2 Host: forta.doim.com
3 Cookie: _ga=GAL.1.107361307.1639661195; _ga_3ERDDVRGQQ=
  GS1.1.1639670007.3.1.1639670502.0
4 Content-Length: 2050
5 Accept: */*
6 Authorization: Bearer
  eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJwYXlsb2FkIjp7ImkiOiJo50Cwic2x1ZyI6IjIwTUZMTY0LTJkNmUtNDcxYy04MDE2LTEyYzA5N2Y0MDJlNyIsInB1YmtpY0FkZHI3c3MiOiIweENDNDY2M0FiOTAzYjgyZjUyNGI3MzlhMzhEMjUwMG1NjMxRUFkYjMiLCJyb2xLIjoiREVWRUxPUEVSIn0sImhhdCI6MTYzOTY2MTQxMiwicXhwIjoxNjM5ODM0MjE5fQ.4I1IF8D4ws9ILSWNH0r_AhiFUT7yA1m-wRqfQ8HZGKs
7 User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like
  Gecko) Chrome/85.0.4183.121 Safari/537.36
8 Content-Type: application/json
9 Origin: https://forta.doim.com
10 Sec-Fetch-Site: same-origin
11 Sec-Fetch-Mode: cors
12 Sec-Fetch-Dest: empty
13 Referer:
  https://forta.doim.com/agent/0x8dcd31c00e76ba2f8f6edcc65e51e754ce8381edc6491f
  70d4724d6a818307cf
14 Accept-Encoding: gzip, deflate
15 Accept-Language: en-US,en;q=0.9
16
17 {
  "variables":{
    "agentId":"RANDOMVALUE",
    "description":"Configure this agent",
    "manifestIPFS":
      "bafkreigexqkgshaecsagj24c2rsvljlu5pweqa2qmler2toznyqqqb6ukq",
    "metaTransaction":{
      "signature":
        "0x3f1129c9dc16dc028643482391419ald1372e2fb17249278bc516af33ccad9a523d8f
        f7131013757c49f1c0875c993bf28814f8416aa7eb9e4dbff52228f9fbc1c",
      "request":{
        "value":0,
        "gas":1000000,
        "nonce":"0",
    
```



Recommendations

Proper validations should be added in both frontend and backend. Sanitize values before inserting them into the DB.

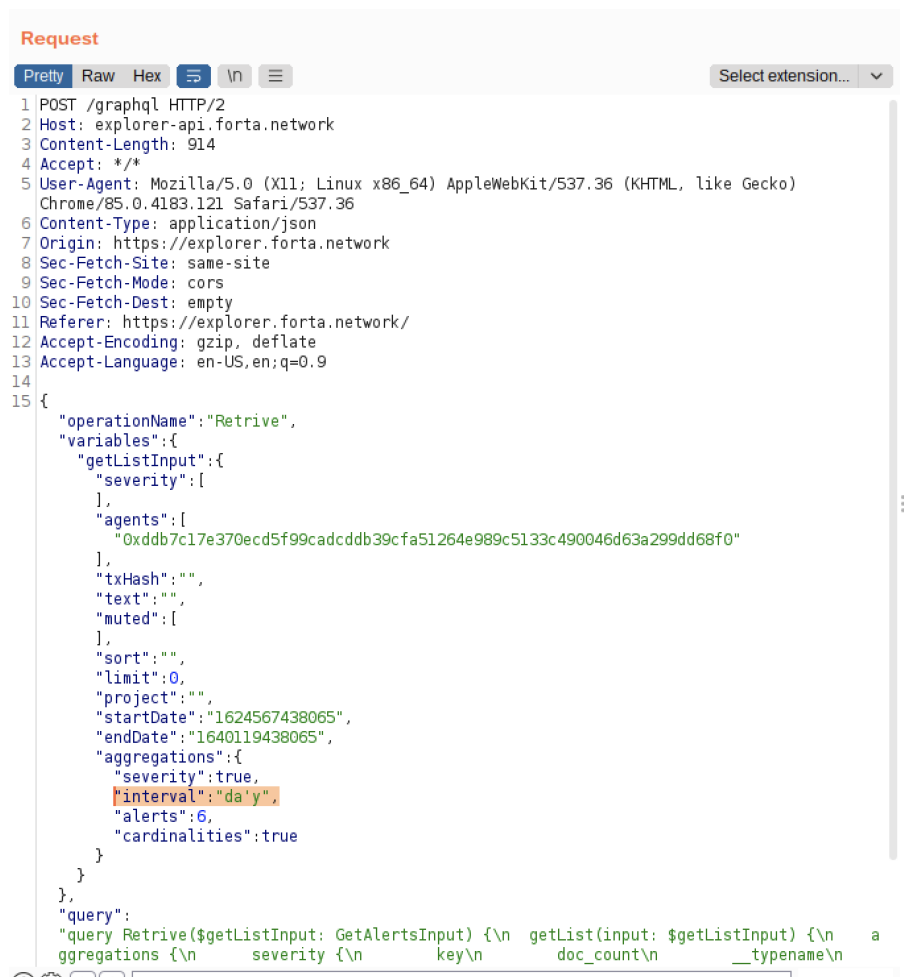
#07 - Sensitive Information Leak in GraphQL Query

Severity: **Low**

Description

Any unauthenticated user is allowed to **modify** input variables leading to **Internal Server Errors** leaking information about internal endpoints, AWS security tokens and other information. This occurs because GraphQL debug mode is enabled.

Evidence



```

Request
Pretty Raw Hex ↵ \n ☰ Select extension...
1 POST /graphql HTTP/2
2 Host: explorer-api.forta.network
3 Content-Length: 914
4 Accept: */*
5 User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko)
  Chrome/85.0.4183.121 Safari/537.36
6 Content-Type: application/json
7 Origin: https://explorer.forta.network
8 Sec-Fetch-Site: same-site
9 Sec-Fetch-Mode: cors
10 Sec-Fetch-Dest: empty
11 Referer: https://explorer.forta.network/
12 Accept-Encoding: gzip, deflate
13 Accept-Language: en-US,en;q=0.9
14
15 {
  "operationName":"Retrive",
  "variables":{
    "getListInput":{
      "severity":[
    ],
      "agents":[
        "0xddb7c17e370ecd5f99cadcd9cfa51264e989c5133c490046d63a299dd68f0"
      ],
      "txHash":"","
      "text":"","
      "muted":[
    ],
      "sort":"","
      "limit":0,
      "project":"","
      "startDate":"1624567438065",
      "endDate":"1640119438065",
      "aggregations":{
        "severity":true,
        "interval":"da'y",
        "alerts":6,
        "cardinalities":true
      }
    }
  },
  "query":
    "query Retrive($getListInput: GetAlertsInput) {\n  getList(input: $getListInput) {\n    a
    ggregations {\n      severity {\n        key\n        doc_count\n        __typename\n
    
```

```

Response
Pretty Raw Hex Render
1 HTTP/2 200 OK
2 Date: Tue, 21 Dec 2021 21:02:10 GMT
3 Content-Type: application/json; charset=utf-8
4 X-Powered-By: Express
5 Access-Control-Allow-Origin: *
6 Etag: W/"c15-gHJyDjb9IkGbnmcSXl7AwH53ils"
7 Cf-Cache-Status: DYNAMIC
8 Expect-Ct: max-age=604800,
report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
9 Server: cloudflare
10 Cf-Ray: 6c140cbdef0608f5-EZE
11
12 {
  "errors": [
    {
      "message": "Request failed with status code 400",
      "locations": [
        {
          "line": 2,
          "column": 3
        }
      ],
      "path": [
        "getList"
      ],
      "extensions": {
        "code": "INTERNAL_SERVER_ERROR",
        "exception": {
          "config": {
            "url": "/forta_alerts/_search",
            "data": {
              "{\n\"aggs\":{\n\"severity\":{\n\"terms\":{\n\"field\":\n\"alert.severity.keyword\", \"order\
er\":{\n\"_count\":\n\"desc\", \"size\":5}},\n\"alerts\":{\n\"terms\":{\n\"field\":\n\"aler
t.name.keyword\", \"order\":{\n\"_count\":\n\"desc\", \"size\":6}},\n\"interval\":{\n\"d
ate_histogram\":{\n\"field\":\n\"alert_timestamp\", \"interval\":\n\"day\"}},\n\"agents
_cardinality\":{\n\"cardinality\":{\n\"field\":\n\"source.agent.id.keyword\"}},\n\"aler
ts_cardinality\":{\n\"cardinality\":{\n\"field\":\n\"alert.name.keyword\"}},\n\"size\":
0,\n\"from\":0,\n\"query\":{\n\"bool\":{\n\"must\":{\n\"range\":{\n\"alert_timestamp\":{\n
gte\":\n\"1624567438065\", \"format\":\n\"epoch_millis\"}},\n\"range\":{\n\"alert_tim
estamp\":{\n\"lte\":\n\"1640119438065\", \"format\":\n\"epoch_millis\"}},\n\"bool\":{\n
\"should\":{\n\"match\":{\n\"source.agent.id\":\n\"0xddb7c17e370ecd5f99cadddb39cfa51
264e989c5133c490046d63a299dd68f0\"}}}}}}},\n\"sort\":{\n\"source.block.number\":{\n
\"order\":\n\"desc\"}},\n\"id.keyword\":{\n\"order\":\n\"desc\"}}}}",
            "headers": {

```

```

Response
Pretty Raw Hex Render
ts_cardinality\":{\"cardinality\":{\"field\":{\"alert.name.keyword\"}},\"size\":0,\"from\":0,\"query\":{\"bool\":{\"must\":{\"range\":{\"alert.timestamp\":{\"gte\":{\"1624567438065\"},\"format\":{\"epoch_millis\"}}},\"range\":{\"alert.timestamp\":{\"lte\":{\"1640119438065\"},\"format\":{\"epoch_millis\"}}},\"bool\":{\"should\":{\"match\":{\"source.agent.id\":{\"0xddb7c17e370ecd5f99cadcddb39cfa51264e989c5133c490046d63a299dd68f0\"}}}}},\"sort\":{\"source.block.number\":{\"order\":{\"desc\"}},\"id.keyword\":{\"order\":{\"desc\"}}}},\"headers\":{\"Content-Type\":\"application/json\", \"Host\": \"vpc-prod-forta-alerts-e4dk7rvfraouyc7uvnygan5hii.us-east-1.es.amazonaws.com\", \"Content-Length\":808, \"X-Amz-Security-Token\": \"IQoJb3JpZ2luX2VjEjMaCXVzLWVhc3QtMSJIMEYCIQDTbAUCILFRYDhgqVM7Ve6fPRnSZ8wF7ExpKoZLADVDSAIhAPaLP5SuP/DNxbZegwmk7gqYqVD/PcNjVvjbX1+wHYnUkusDCBwQARoMMIzNTQxNDEzNjMwIgxWhxIc3H5tw68WC70qyAMZon409gYtK9soytBcvKCRbH5xikj5bbVaFfw6CSv++DQqJkm5wfCzoGmD3VK8yMLQL/2frdXB7rH2qA4yHrmqZGQL7ChMQvGWpsZ7BglN5e517GmNY4RFmLBPgR8g4PiSTCb/Wzfjeja7iq54iENkUnzxohvn5LzEiP402yLAI6fm/ArnXUM4f/thr9V/vOREaWeZay0x5nuYj8dP+3gLeRFRknsGd9UQWV1CKZh819WuhsNsUlVzpp+1HttZU0ya3yNKZQnao4cbx7BgU4BUpUWq+CmONfwlMRLHzMu/ZASf1o0HRZxbsdfnFGurGfckzP+b2TNmL5nNN3GIrQ1FIuRi71rrq8Hiyw4hjY0a1j/h/JXHDR+tyoxa444e5KGiTv2yUx40eEedopc25KH964DH0vYAZfH5ncJTCUZkri9JmJJqiNryJGSfL8PFvxiSv/mbj0w2HqpNyv/YR453LzeZK110GmwhKztG8+rTlFz0ACz5HYuzASVmFLdQc6q2tbtxZJzZZf3j5NFK6WFEX1qHAVrVqvQfj/2vD5oIFKkxc6BJ3glrEYuk1bnXkvALS69Nc+AWbIMU21cm/CMHQRBg2t/FL8wkrIjgY6pAFWHHCpDmoaD00093q0WZ51k8JE3ygJyJlvo+XQQCbn/7Fdl+bIfTeashpv5G5gWUBsbPXTdvs4vi0z00Xqu8D1bfFeFAYIRYeIsQ3MXczbyc6GxTZccsLExSxeNEHU5alLWQFYkBz+EnMwLL5+m76jSyWBAQGV2m08QvwFTfdH+CvhuIT9/eRGg+kFwigNL7HLTIOnrC2dXRLe9L3CTPzV2RUBAw==\", \"X-Amz-Date\": \"20211221T210210Z\", \"Authorization\": \"AWS4-HMAC-SHA256 Credential=ASIARZQ5EL370NUA3IXA/20211221/us-east-1/es/aws4_request, SignedHeaders=content-length;content-type;host;x-amz-date;x-amz-security-token, Signature=53b8a3d17f039b17dc3d6cae8f949048c8e2a5022130c79054d40f38ff484bd75\", \"User-Agent\": \"axios/0.21.4\" }, \"baseUrl\": \"https://vpc-prod-forta-alerts-e4dk7rvfraouyc7uvnygan5hii.us-east-1.es.amazonaws.com\", \"transformRequest\": [ null ], \"transformResponse\": [ null ]

```

Recommendations

Disable Debug mode on production environments.

#08 - HTML Injection in alerts emails

Severity: **Low**

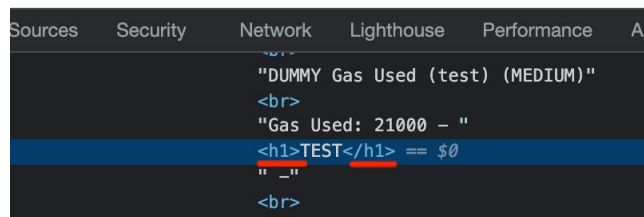
Description

When creating new Agents, in the Findings values, HTML is allowed, therefore HTML is rendered when receiving email notifications.

An attacker could reformat e-mail notifications to redirect them to malicious sites or persuade users to perform malicious actions or to be victim of phishing attacks

Evidence

Email with rendered HTML:



Agent code submitted with embedded HTML code:

```
findings.push(Finding.fromObject({
  name: "DUMMY Gas Used (test)",
  description: `Gas Used: ${gasUsed} - <h1>TEST</h1> -`,
  alertId: "FORTA-1",
  severity: FindingSeverity.Medium,
  type: FindingType.Suspicious
}))
```

Recommendations

Proper input sanitization should be performed before processing them in the backend and rendered in emails or frontend.