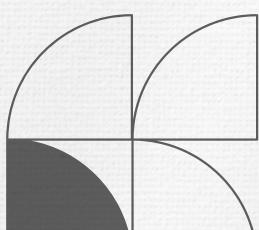
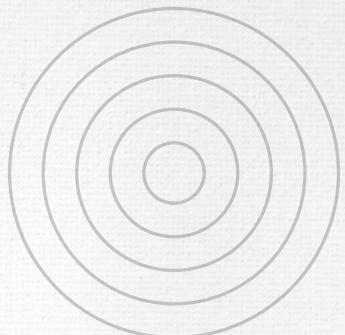


RAPPORT AUDIT DE SÉCURITÉ

{EPITECH}

PROJET _____ **ESP DREAMHABITAT** _____
ID _____ **T-ESP-800 msc2025** _____
Date _____ **04/02/2025** _____



OBJET

Ce document constitue le rapport des tests de vulnérabilités applicatives effectués dans le cadre d'un contrôle de sécurité d'une application web en développement en mode boîte noire.

PERIMETRE

TESTS APPLICATIFS

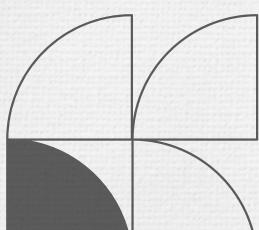
Les URL suivantes ont été analysées :

- <http://localhost:3000> (Serveur/Api)
- <http://localhost:3001> (Client)
- <http://localhost:3306> (Base de données)

LIMITATIONS

Limitations rencontrées:

- Application encore en développement



SYNTHESE

TESTS REALISES

Catégorie	Tests réalisés ?
Analyse de la version des services	
Contrôle des entrées utilisateur	
Analyse des erreurs de configuration	
Gestion des erreurs	
Gestion des identités	
Chiffrement des flux	
Authentification	
Logique métier	
Gestion des sessions	

- NON RÉALISÉ

- VULNERABLE

- SÉCURISÉ

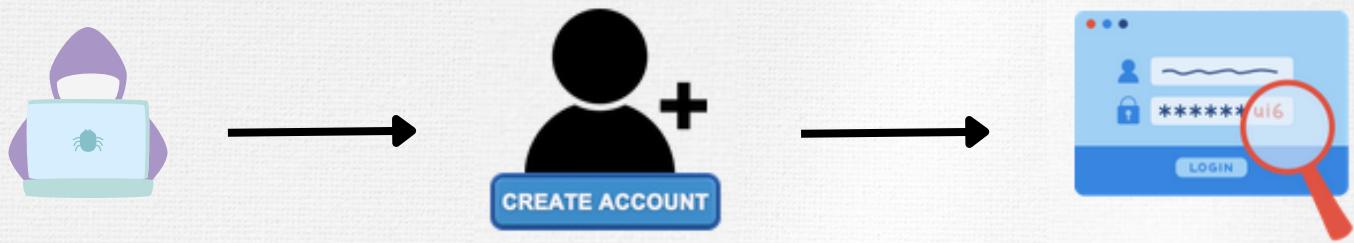
VULNÉRABILITÉS APPLICATIVES

VULNÉRABILITÉ	ID	CRITICAL	HIGH	MEDIUM	LOW
GUESSABLE USER ACCOUNT	01	✓			
WEAK PASSWORD POLICY	02			✓	
WEAK LOCK OUT MECHANISM	03			✓	
CROSS ORIGIN RESSOURCE SHARING	04				✓
INFORMATION DISCLOSURE	05			✓	
INSECURE COMMUNICATION	06			✓	
INSUFFICIENT SECURITY CONTROLS	07	✓			
FINGERPRINT WEB SERVER	08				✓
TESTING SESSION TIMEOUT	09				✓
TESTING FOR JSON WEB TOKEN	10	✓			
BACKUP AND UNREFERENCED FILES	11		✓		

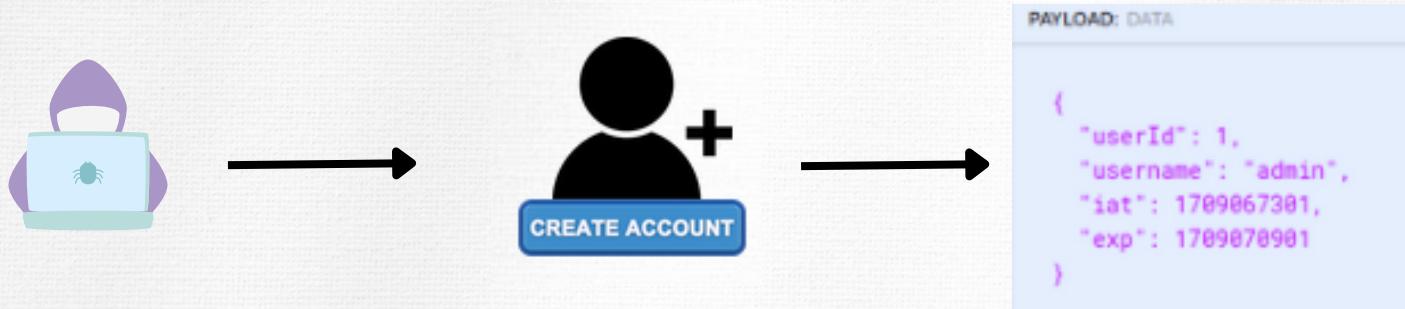
VULNÉRABILITÉ	ID	CRITICAL	HIGH	MEDIUM	LOW
VULNERABLE MODULE	12			✓	
BUSINESS LOGIC VALIDATION	13				✓

ESCALADE DE PRIVILÈGES SCÉNARIOS

- 1) Identification d'utilisateur existant dans la page register Brute force de mot de passe sur le compte identifié



- 2) Création d'un utilisateur Modification du JWT une fois connecté avec le salt découvert



VULNERABILITES

Identity Management Testing

ID: 01	GUESSABLE USER ACCOUNT	CRITICAL
Chemin(s)	http://localhost:3001/register http://localhost:3000/auth/register	
Constat	La réponse du serveur fournit une information permettant de déduire l'existence d'un compte utilisateur. Un attaquant peut ainsi lancer une attaque du type "brute force" permettant de retrouver les identifiants de comptes valides.	
Risque technique	Un attaquant peut potentiellement compromettre un compte utilisateur et réaliser des actions malveillantes.	
Précognition	Il est recommandé de mettre en œuvre une réponse standardisée du serveur qui ne révèle pas d'informations permettant de déduire l'existence ou la validité des comptes utilisateur.	

Présentation des preuves :

Request

```
Pretty Raw Hex
1 POST /auth/register HTTP/1.1
2 Host: localhost:3000
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101
   Firefox/122.0
4 Accept: application/json, text/plain, */*
5 Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/json; charset=utf-8
8 Content-Length: 38
9 Origin: http://localhost:3001
10 Connection: close
11 Referer: http://localhost:3001/
12 Sec-Fetch-Dest: empty
13 Sec-Fetch-Mode: cors
14 Sec-Fetch-Site: same-site
15 X-PwnFox-Color: blue
16
17 {
    "username": "admin",
    "password": "test"
}
```

Response

```
Pretty Raw Hex Render
1 HTTP/1.1 409 Conflict
2 Access-Control-Allow-Origin: *
3 X-DNS-Prefetch-Control: off
4 X-Frame-Options: SAMEORIGIN
5 Strict-Transport-Security: max-age=15552000; includeSubDomains
6 X-Download-Options: noopen
7 X-Content-Type-Options: nosniff
8 X-XSS-Protection: 1; mode=block
9 Content-Type: text/html; charset=utf-8
10 Content-Length: 23
11 ETag: W/"17-p1+q6GBlRsMkKi37TAOMIcNCg0g"
12 Date: Thu, 24 Apr 2025 11:39:41 GMT
13 Connection: close
14
15 username already in use
```

Il est possible de faire du guessing sur le nom d'un utilisateur

Présentation des preuves :

Request

Pretty Raw Hex

```
1 POST /auth/login HTTP/1.1
2 Host: localhost:3000
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:127.0) Gecko/20100101 Firefox/127.0
4 Accept: application/json, text/plain, */*
5 Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/json; charset=utf-8
8 Content-Length: 39
9 Origin: http://localhost:3001
10 Connection: keep-alive
11 Referer: http://localhost:3001/
12 Sec-Fetch-Dest: empty
13 Sec-Fetch-Mode: cors
14 Sec-Fetch-Site: same-site
15 X-PwnFox-Color: blue
16 Priority: u=1
17
18 {
    "username": "admin",
    "password": "admin"
}
```

Response

Pretty Raw Hex Render JSON Web Tokens

```
1 HTTP/1.1 200 OK
2 Access-Control-Allow-Origin: *
3 X-DNS-Prefetch-Control: off
4 X-Frame-Options: SAMEORIGIN
5 Strict-Transport-Security: max-age=15552000; includeSubDomains
6 X-Download-Options: noopen
7 X-Content-Type-Options: nosniff
8 X-XSS-Protection: 1; mode=block
9 Content-Type: application/json; charset=utf-8
10 Content-Length: 180
```

Utilisateur admin:admin identifié avec des droits administrateurs

VULNERABILITES

Authentication Testing (WSTG-AUTHN)

ID: 02	WEAK PASSWORD POLICY	MEDIUM
Chemin(s)	http://localhost:3001/register http://localhost:3000/auth/register	
Constat	La politique de mot de passe n'est pas assez robuste.	
Risque technique	Un attaquant pourrait compromettre des comptes en tentant de se connecter avec des mots de passe triviaux ou en réalisant une attaque du type "brute force".	
Précognition	Il est recommandé de renforcer les critères de complexité et de longueur des mots de passe. En mettant en place des exigences strictes telles que l'utilisation de caractères spéciaux, de chiffres et de combinaisons de cas.	

Présentation des preuves :

Request	Response
<pre>Pretty Raw Hex 1 POST /auth/register HTTP/1.1 2 Host: localhost:3000 3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101 Firefox/122.0 4 Accept: application/json, text/plain, /* 5 Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3 6 Accept-Encoding: gzip, deflate, br 7 Content-Type: application/json; charset=utf-8 8 Content-Length: 40 9 Origin: http://localhost:3001 10 Connection: close 11 Referer: http://localhost:3001/ 12 Sec-Fetch-Dest: empty 13 Sec-Fetch-Mode: cors 14 Sec-Fetch-Site: same-site 15 X-PwnFox-Color: blue 16 17 { "username": "users", "password": "abcd" 18 }</pre>	<pre>Pretty Raw Hex 1 HTTP/1.1 201 Created 2 Access-Control-Allow-Origin: * 3 X-DNS-Prefetch-Control: off 4 X-Frame-Options: SAMEORIGIN 5 Strict-Transport-Security: max-age=15552000; includeSubDomains 6 X-Download-Options: noopen 7 X-Content-Type-Options: nosniff 8 X-XSS-Protection: 1; mode=block 9 Content-Type: text/html; charset=utf-8 10 Content-Length: 12 11 ETag: W/"c-4Vmhvov3D5N5n+0DRiUz0via3TE" 12 Date: Mon, 04 Mar 2024 11:39:41 GMT 13 Connection: close 14 15 User created</pre>

Response
<pre>Pretty Raw Hex Render 1 HTTP/1.1 201 Created 2 Access-Control-Allow-Origin: * 3 X-DNS-Prefetch-Control: off 4 X-Frame-Options: SAMEORIGIN 5 Strict-Transport-Security: max-age=15552000; includeSubDomains 6 X-Download-Options: noopen 7 X-Content-Type-Options: nosniff 8 X-XSS-Protection: 1; mode=block 9 Content-Type: text/html; charset=utf-8 10 Content-Length: 12 11 ETag: W/"c-4Vmhvov3D5N5n+0DRiUz0via3TE" 12 Date: Thu, 24 Apr 2025 11:39:41 GMT 13 Connection: close 14 15 User created</pre>

Politique de mots de passe faible (4 caractères acceptés)

VULNERABILITES

Authentication Testing (WSTG-AUTHN)

ID: 03	WEAK LOCK OUT MECHANISM	MEDIUM
Chemin(s)	http://localhost:3001/login http://localhost:3000/auth/login	
Constat	L'application ne dispose pas de mécanisme de verrouillage après un certain nombre de tentatives de connexion infructueuses.	
Risque technique	L'absence de mécanismes de verrouillage de compte pourrait permettre à un attaquant de réaliser des attaques du type "brute force" pour compromettre des comptes.	
Précognition	Il est essentiel de mettre en place un mécanisme de verrouillage de compte après un nombre défini de tentatives de connexion infructueuses. Ce dispositif renforcera la sécurité en limitant les attaques par force brute et en réduisant les chances d'accès non autorisé aux comptes utilisateur.	

Présentation des preuves :

5. Intruder attack of http://localhost:3000

Results Positions Payloads Resource pool Settings

Filter: Showing all items

Request ^	Payload	Status code	Response received
900	OSP22	401	9
901	OUTLN	401	9
902	OWA	401	9
903	OWA_PUBLIC	401	6
904	OWNER	401	6
905	PANAMA	401	6
906	PATROL	401	6
907	PERFSTAT	401	6
908	PLEX	401	9
909	SUPERSECRET	401	7
910	PM	401	7

Request Response

Pretty Raw Hex

```
1 POST /auth/login HTTP/1.1
2 Host: localhost:3000
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101 Firefox/122.0
4 Accept: application/json, text/plain, */*
5 Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/json; charset=utf-8
8 Content-Length: 43
9 Origin: http://localhost:3001
10 Connection: keep-alive
11 Referer: http://localhost:3001/
12 Sec-Fetch-Dest: empty
13 Sec-Fetch-Mode: cors
14 Sec-Fetch-Site: same-site
```

② ⚙️ ← → Search

Finished

Multiples tentatives de brute force non bloquée

VULNERABILITES

Client-Side Testing (WSTG-CLNT)

ID: 04	CROSS ORIGIN RESSOURCE SHARING	LOW
Chemin(s)	http://localhost:3001/*	
Constat	Bien qu'une sécurité CORS soit en place, elle autorise toutes les sources (*), ce qui peut permettre à des sites tiers d'effectuer des actions privilégiées et de récupérer des informations sensibles.	
Risque technique	Attaques par des sites tiers, compromettant la confidentialité des données.	
Précognition	Il est recommandé de revoir et de restreindre la configuration de la politique de partage des ressources entre origines (CORS) pour limiter l'accès uniquement aux sources légitimes nécessaires. En restreignant les accès à des domaines spécifiques plutôt que d'utiliser une autorisation pour toutes les sources (*).	

Présentation des preuves :

Request

Pretty Raw Hex

```
1 GET /user HTTP/1.1
2 Host: localhost:3000
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101 Firefox/122.0
4 Accept: application/json, text/plain, */*
5 Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 auth:
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJlc2VySWQiOjEsInVzZXJuYW1lIjoiYWRtaW4iLCJpYXQiOjE3MD
k1NzUyNzEsImV4cCI6MTcwOTU3ODg3MX0.kGdNvEZwpBxzqI9AWXX1Phj3qJ9UdZsSb9NK0LFTgx
8 Origin: http.hacker
9 Connection: close
10 Referer: http://localhost:3001/
11 Sec-Fetch-Dest: empty
12 Sec-Fetch-Mode: cors
```

Response

Pretty Raw Hex Render

```
1 HTTP/1.1 200 OK
2 Access-Control-Allow-Origin: *
3 X-DNS-Prefetch-Control: off
4 X-Frame-Options: SAMEORIGIN
5 Strict-Transport-Security: max-age=15552000; includeSubDomains
6 X-Download-Options: noopen
7 X-Content-Type-Options: nosniff
8 X-XSS-Protection: 1; mode=block
9 token:
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJlc2VySWQiOjEsInVzZXJuYW1lIjoiYWRtaW4iLCJpYXQiOjE3MD
k1NzU1NjMsImV4cCI6MTcwOTU3OTE2M30.PxTC1P-2rUTQG2SH0KUAEwYLSmly-MXP4Ty5WsDg3c
10 Content-Type: application/json; charset=utf-8
11 Content-Length: 1177
12 ETag: W/"499-oUecV12n2ktj+e/w80/RmCcrLsg"
```

Requête envoyée depuis une origine potentiellement malveillante

VULNERABILITES

Testing for Error Handling (WSTG-ERRH)

ID: 05	INFORMATION DISCLOSURE	MEDIUM
Chemin(s)	http://localhost:3000 http://localhost:3001/	
Constat	En provoquant des erreurs, des informations sensibles telles que des détails de la base de données et des routes peuvent être obtenues.	
Risque technique	Exposition d'informations sensibles, facilitant des attaques ultérieures. Ces informations permettent potentiellement d'identifier des vulnérabilités exploitables.	
Précognition	Pour prévenir la divulgation d'informations sensibles par le biais d'erreurs, il est crucial de mettre en place des mécanismes robustes de gestion des erreurs qui ne révèlent pas de détails critiques tels que des informations sur la base de données ou des routes spécifiques.	

Présentation des preuves :

Unhandled Rejection (TypeError): e.response.data[0].constraints is undefined

```
./src/store/actions/account.actions.ts/register/<
src/store/actions/account.actions.ts:71

68 |     }
69 |     console.log(e.response.data);
70 |     if (e.response.data.length !== undefined) {
> 71 |         return dispatch(addNotification("Error", e.response.data[0].constraints.length));
  72 |     ^
73 |         return dispatch(addNotification("Error", e.response.data));
74 |     }
```

[View compiled](#)

This screen is visible only in development. It will not appear if the app crashes in production.
Open your browser's developer console to further inspect this error.

Erreur générée lors de la création d'un utilisateur avec un username existant

Request

```
Pretty Raw Hex
1 POST /auth/register HTTP/1.1
2 Host: localhost:3000
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101 Firefox/122.0
4 Accept: application/json, text/plain, /*
5 Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/json;charset=utf-8
8 Content-Length: 46
9 Origin: http://localhost:3001
10 Connection: close
11 Referer: http://localhost:3001/
12 Sec-Fetch-Dest: empty
13 Sec-Fetch-Mode: cors
14 Sec-Fetch-Site: same-site
15 X-PwnFox-Color: blue
16
17 {
    "username": "users",
    "password": "*****"
18 }
```

Response

```
Pretty Raw Hex Render
1 HTTP/1.1 400 Bad Request
2 Access-Control-Allow-Origin: *
3 X-DNS-Prefetch-Control: off
4 X-Frame-Options: SAMEORIGIN
5 Strict-Transport-Security: max-age=15552000; includeSubDomains
6 X-Download-Options: noopen
7 X-Content-Type-Options: nosniff
8 X-XSS-Protection: 1; mode=block
9 Content-Security-Policy: default-src 'none'
10 Content-Type: text/html; charset=utf-8
11 Content-Length: 867
12 Date: Thu, 24 Apr 2025 11:39:41 GMT
13 Connection: close
14
15 <!DOCTYPE html>
16 <html lang="en">
17   <head>
18     <meta charset="utf-8">
19     <title>
20       Error
21     </title>
22   </head>
23   <body>
24     <pre>
SyntaxError: Unexpected string in JSON at position 40<br>
  &nbs; &nbs;at JSON.parse (<anonymous>)<br>
  &nbs; &nbs;at parse
(/usr/src/app/node_modules/body-parser/lib/types/json.js:89:19)<br>
  &nbs; &nbs;at /usr/src/app/node_modules/body-parser/lib/read.js:121:18<br>
  &nbs; &nbs;at invokeCallback (/usr/src/app/node_modules/raw-body/index.js:224:16)<br>
  &nbs; &nbs;at done (/usr/src/app/node_modules/raw-body/index.js:213:7)<br>
  &nbs; &nbs;at IncomingMessage.onEnd
(/usr/src/app/node_modules/raw-body/index.js:273:7)<br>
  &nbs; &nbs;at IncomingMessage.emit (events.js:198:15)<br>
  &nbs; &nbs;at endReadableNT (_stream_readable.js:1139:12)<br>
  &nbs; &nbs;at processTicksAndRejections (internal/process/task_queues.js:81:17)
</pre>
23   </body>
24 </html>
```

Erreur générée permettant de découvrir des routes côté serveur

VULNERABILITES

Configuration and Deployment Management Testing (WSTG-CONF)

ID: 06	INSECURE COMMUNICATION	MEDIUM
Chemin(s)	http://localhost:3001/*	
Constat	Le site fonctionne en HTTP au lieu de HTTPS, exposant les données des utilisateurs à des interceptions potentielles.	
Risque technique	Les données échangées entre le navigateur de l'utilisateur et le serveur web ne sont pas cryptées. Cela signifie que des tiers pourraient potentiellement intercepter et lire ces données lors de leur transmission. Risque accru d'attaques du type Man-in-the-Middle.	
Précognition	Il est impératif de migrer le site vers HTTPS afin de sécuriser la communication et de protéger les données des utilisateurs contre les interceptions malveillantes. En implémentant HTTPS, toutes les données échangées seront chiffrées, réduisant ainsi considérablement le risque d'attaques	

Présentation des preuves :

Request

Pretty Raw Hex

```
1 GET /products HTTP/1.1
2 Host: localhost:3001
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101 Firefox/122.0
4 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 Connection: close
8 Cookie: token=
  eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJc2VySWQiOjEsInVzZXJuYWlIjoiYWRtaW4iLCJpYXQiOjE3MDk1NzUyNzEsImV4cCI6MTcwOTU3ODg3MX0.kGdNvEZwpBxzqI9AWXX1Phj3qJ9UdZsSb9NK0LFTgx
9 Upgrade-Insecure-Requests: 1
```

Response

Pretty Raw Hex Render

```
1 HTTP/1.1 200 OK
2 X-Powered-By: Express
3 Accept-Ranges: bytes
4 Content-Type: text/html; charset=UTF-8
5 ETag: W/"48e-QrEJ0goZpudeTGjEFtKXQVw1fVY"
6 Vary: Accept-Encoding
7 Date: Thu, 24 Apr 2025 11:39:41 GMT
8 Connection: close
9 Content-Length: 1166
10
11 <!DOCTYPE html>
12 <html lang="en">
13   <head>
```

L'ensemble du site du type “Hypertext Transfer Protocol” (HTTP)

VULNERABILITES

Identity Management Testing (WSTG-IDNT)

ID: 07 INSUFFICIENT SECURITY CONTROLS CRITICAL	
Chemin(s)	http://localhost:3001/users http://localhost:3000/user
Constat	Dans le front de l'application les changements de droits ne se font pas en direct. De plus les utilisateurs ont accès à la page d'administration bien qu'ils n'aient pas la possibilité d'y effectuer des actions ou d'y voir le contenu.
Risque technique	Un attaquant pourrait réaliser des actions auxquelles il n'est normalement pas autorisé.
Précognition	Il est essentiel d'améliorer les contrôles de sécurité pour garantir que les changements de droits dans l'application sont appliqués en temps réel et de manière sécurisée. De plus, il est critique de restreindre l'accès à la page d'administration uniquement aux utilisateurs autorisés à y effectuer des actions spécifiques ou à voir son contenu.

Présentation des preuves :

Transformez votre espace avec DREAM HABITAT

Découvrez comment notre technologie IA révolutionne le design d'intérieur et rend la décoration plus accessible que jamais.



Intelligence Artificielle Avancée

Notre IA analyse votre espace et propose des designs personnalisés qui correspondent parfaitement à vos goûts et à votre style de vie.



Redesign Instantané

Obtenez des visualisations de votre espace réinventé en quelques secondes seulement, sans attente ni délai.



Multiple Variations

Explorez différentes versions de votre espace avec plusieurs styles et thèmes pour trouver celui qui vous convient le mieux.

La page users n'est pas bloquée côté front pour un utilisateur qui n'a pas les droits requis, bien qu'il ne soit pas en permission d'accéder aux requêtes get users dans le back

Présentation des preuves : _____

Bon retour !

[Passer à Pro](#)

Connectez-vous à votre compte Dream Habitat.

 [Se connecter avec Google](#)

_____ ou _____

E-mail

Mot de passe

connexion

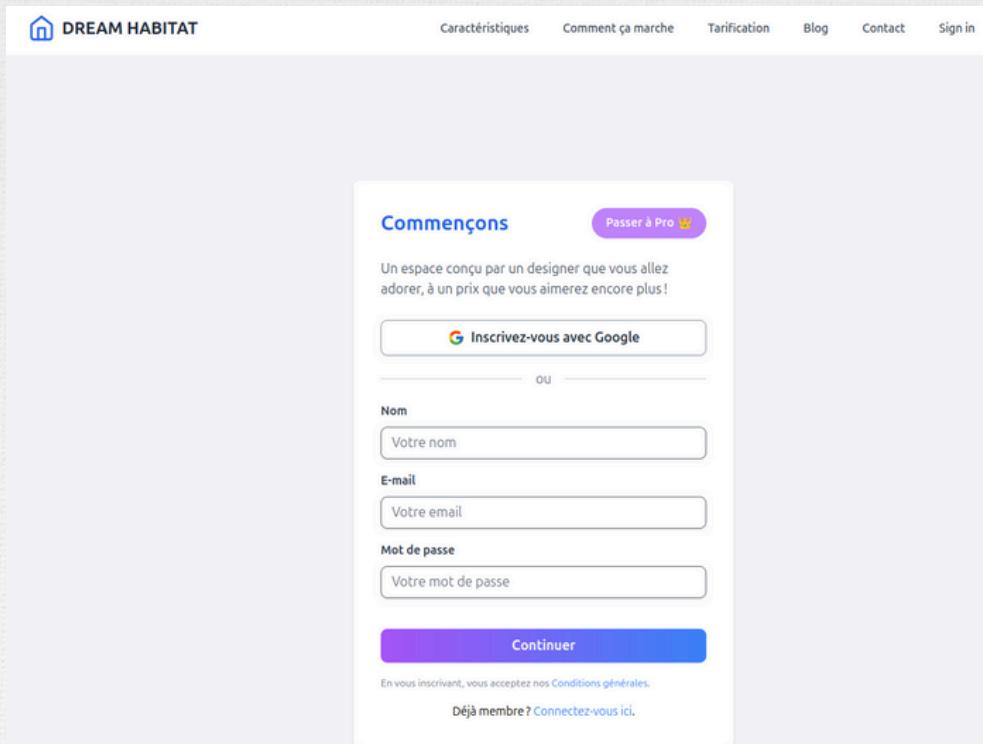
Se souvenir de moi la prochaine fois

[Mot de passe oublié ? Cliquez ici pour réinitialiser.](#)

[Vous n'avez pas de compte ? Inscrivez-vous ici.](#)

Puis connexion avec le compte admin qui a le droit d'observation sur les users

Présentation des preuves :



Puis déconnexion du compte admin pour repasser sur le compte test qui n'avait pas les droits sur la visualisation des users, on peut voir que coté front maintenant il a accès à celle-ci si il passe après un compte admin

A screenshot of a browser's developer tools Network tab. It shows a single request and response pair. The request is a GET to '/user' with status 401 Unauthorized. The response includes standard headers like Access-Control-Allow-Origin, X-DNS-Prefetch-Control, X-Frame-Options, Strict-Transport-Security, X-Download-Options, X-Content-Type-Options, and X-XSS-Protection. It also includes a 'token' header and a long JSON payload containing user data.

On peut voir que coté back l'utilisateur n'a bien aucun accès sur la requête get user bien que côté front il ait la vue

Présentation des preuves :

```
Pretty Raw Hex
1 POST /user/ HTTP/1.1
2 Host: localhost:3000
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:127.0) Gecko/20100101 Firefox/127.0
4 Accept: application/json, text/plain, /*
5 Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/json; charset=utf-8
8 auth:
9 eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJlc2VySWQiOjEsInVzZXJuYW1lIjoiYWRtaW4iLCJpYXQiOjE3M
10 Tk5MTE2MjUsImV4cCI6MTcxOTkxNTIyNX0.SjObRzYbDHzyUnTTjXfElUteylz1Qk044A3U3qNXBAQ
11 Content-Length: 58
12 Origin: http://localhost:3001
13 Connection: keep-alive
14 Referer: http://localhost:3001/
15 Sec-Fetch-Dest: empty
16 Sec-Fetch-Mode: cors
17 Sec-Fetch-Site: same-site
18 X-PwnFox-Color: blue
19 Priority: u=1
20
21 {
22     "username": "testxS",
23     "password": "testt"
24 }
```

Response

```
Pretty Raw Hex Render
1 HTTP/1.1 201 Created
2 Access-Control-Allow-Origin: *
3 X-DNS-Prefetch-Control: off
4 X-Frame-Options: SAMEORIGIN
5 Strict-Transport-Security: max-age=15552000; includeSubDomains
6 X-Download-Options: noopen
7 X-Content-Type-Options: nosniff
8 X-XSS-Protection: 1; mode=block
9 token:
```

Un utilisateur avec les droits ADMIN peut créer un utilisateur avec une method POST sachant que cette option n'est pas présente dans l'application web

Présentation des preuves :

```
Pretty Raw Hex
1 POST /user/ HTTP/1.1
2 Host: localhost:3000
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:127.0) Gecko/20100101 Firefox/127.0
4 Accept: application/json, text/plain, /*
5 Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/json; charset=utf-8
8 auth:
9 eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJlc2VySWQiOjEsInVzZXJuYW1lIjoiYWRtaW4iLCJpYXQiOjE3M
10 Tk5MTE2MjUsImV4cCI6MTcxOTkxNTIyNQ.eyJ0bRzYbDHzyUnTTjXfElUteylz1Qk044A3U3qNXBAQ
11 Content-Length: 58
12 Origin: http://localhost:3001
13 Connection: keep-alive
14 Referer: http://localhost:3001/
15 Sec-Fetch-Dest: empty
16 Sec-Fetch-Mode: cors
17 Sec-Fetch-Site: same-site
18 X-PwnFox-Color: blue
19 Priority: u=1
20
21 {
22     "username": "testxS",
23     "password": "testt"
24 }
```

Response

```
Pretty Raw Hex Render
1 HTTP/1.1 201 Created
2 Access-Control-Allow-Origin: *
3 X-DNS-Prefetch-Control: off
4 X-Frame-Options: SAMEORIGIN
5 Strict-Transport-Security: max-age=15552000; includeSubDomains
6 X-Download-Options: noopen
7 X-Content-Type-Options: nosniff
8 X-XSS-Protection: 1; mode=block
9 token:
```

Un utilisateur avec les droits ADMIN peut créer un utilisateur avec une method POST sachant que cette option n'est pas présente dans l'application web

Présentation des preuves :

```
Request
Pretty Raw Hex
1 DELETE /user/22 HTTP/1.1
2 Host: localhost:3000
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:127.0)
Gecko/20100101 Firefox/127.0
4 Accept: application/json, text/plain, /*
5 Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
Content-Type: application/json; charset=utf-8

Response
Pretty Raw Hex Render
1 HTTP/1.1 204 No Content
2 Access-Control-Allow-Origin: *
3 X-DNS-Prefetch-Control: off
4 X-Frame-Options: SAMEORIGIN
5 Strict-Transport-Security: max-age=15552000; includeSubDomains
6 X-Download-Options: noopen
7 X-Content-Type-Options: nosniff
```

Un utilisateur avec les droits ADMIN peut supprimer un utilisateur avec une method DELETE sachant que cette option n'est pas présente dans l'application web

VULNERABILITES

Information Gathering (WSTG-INFO)

ID: 08	FINGERPRINT WEB SERVER	LOW
Chemin(s)	http://localhost:3001/users	
Constat	Des informations techniques concernant le serveur web sont mentionnées dans certains entêtes.	
Risque technique	Ces informations permettent potentiellement d'identifier des vulnérabilités exploitables.	
Précognition	Pour réduire le risque de divulgation d'informations sensibles concernant le serveur web à travers les entêtes, il est recommandé de configurer le serveur afin de limiter ou de supprimer complètement les détails techniques exposés.	

Présentation des preuves :

Request

```
Pretty Raw Hex
1 GET /login HTTP/1.1
2 Host: localhost:3001
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101
   Firefox/122.0
4 Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 Connection: close
8 Upgrade-Insecure-Requests: 1
9 Sec-Fetch-Dest: document
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-Site: none
12 Sec-Fetch-User: ?1
13 X-PwnFox-Color: blue
14 If-None-Match: W/"48e-QrEJ0goZpudeTGjEFtKXQWw1fVY"
15
16
```

Response

```
Pretty Raw Hex Render
1 HTTP/1.1 304 Not Modified
2 K-Powered-By: Express
3 Accept-Ranges: bytes
4 ETag: W/"48e-QrEJ0goZpudeTGjEFtKXQWw1fVY"
5 Date: Thu, 24 Apr 2025 11:39:41 GMT
6 Connection: close
7
8
```

Information sur la technologie visible : Express

Request

Pretty Raw Hex

≡ ln ≡

```
1 GET /login HTTP/1.1
2 Host: localhost:3001
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101
   Firefox/122.0
4 Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 Connection: close
8 Upgrade-Insecure-Requests: 1
9 Sec-Fetch-Dest: document
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-Site: none
12 Sec-Fetch-User: ?1
13 X-PwnFox-Color: blue
14 If-None-Match: W/"48e-QrEJ0goZpudeTGjEFtKXQWw1fVY"
15
16
```

Response

Pretty Raw Hex Render

≡ ln ≡

```
13 <head>
14   <meta charset="utf-8" />
15   <link rel="icon" href="/favicon.ico" />
16   <meta name="viewport" content="width=device-width, initial-scale=1" />
17   <meta name="theme-color" content="#000000" />
18   <meta
19     name="description"
20     content="Web site created using create-react-app"
21   />
22   <link rel="apple-touch-icon" href="logo192.png" />
23   <link rel="manifest" href="/manifest.json" />
24
```

Information sur la technologie visible : react

Request

Pretty Raw Hex

≡ \n ≡

```
1 GET /__webpack_dev_server__/sockjs.bundle.js HTTP/1.1
2 Host: localhost:3001
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101 Firefox/122.0
4 Accept: */*
5 Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 Connection: close
8 Referer: http://localhost:3001/sockjs-node/iframe.html
9 Cookie: token=
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJlc2VySWQiOjEsInVzZXJuYW1lIjoiYWRtaW4iLCJpYXQiOjE3MD
kwNzMAOTIzMmV4cC16MTcwOTA3NzQ5MnQ28mVHV_gtZNbn2uB13S60_eLyihBx6_lrOT6C-4Hal_E
```

Response

Pretty Raw Hex Render

≡ \n ≡

```
1 HTTP/1.1 200 OK
2 X-Powered-By: Express
3 Content-Type: application/javascript
4 Date: Thu, 24 Apr 2025 11:39:41 GMT
5 Connection: close
6 Content-Length: 55544
7
8 !function(t,e){
  "object"==typeof exports&&"object"==typeof module?module.exports=e():"function"==typeof
  define&&define.amd?define([],e):"object"==typeof exports?exports.SockJS=e():t.SockJS=e()
```

Information sur la technologie visible : javascript

VULNERABILITES

Session Management Testing (WSTG-SESS)

ID: 09	TESTING SESSION TIMEOUT	LOW
Chemin(s)	http://localhost:3001/	
Constat	La session n'expire jamais ou bien le temps d'expiration de la session est relativement long (1h+).	
Risque technique	En cas de vol de session, celle-ci restera valide pour l'attaquant.	
Précognition	Il est crucial de tester et de paramétriser correctement le temps d'expiration des sessions utilisateur pour limiter le risque de vol de session. En réduisant le temps d'expiration à un intervalle approprié, par exemple entre 3 et 15 minutes.	

Présentation des preuves :

```
Request
Pretty Raw Hex
1 GET /user HTTP/1.1
2 Host: localhost:3000
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101
  Firefox/122.0
4 Accept: application/json, text/plain, /*
5 Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 auth:
  eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJlc2VySWQiOjEsInVzZXJuYW1lIjoiYWRtaW4iLCJ
  pYXQiOjE3MDk2MzE2NzAsImV4cCI6MTcwOTYzNTI3MH0.F4fqbkqZ0qhzREeWx6enUzd8-KQkTN7Yc2y
  7nUC0t58
8 Origin: https://localhost:3001
```

Requêtes avec un même cookie avec plus de 15 mins
d'intervalle

VULNERABILITES

Session Management Testing (WSTG-SESS)

ID: 10	TESTING FOR JSON WEB TOKEN	CRITICAL
Chemin(s)	http://localhost:3000/login	
Constat	Cette vulnérabilité concerne le manque de contrôle du JWT par l'application. La signature du jeton est faible. De plus le flag "Secure" n'est pas présent dans les cookies.	
Risque technique	Un attaquant peut forger son propre token JWT afin d'avoir accès à l'application avec des droits élevés.	
Précognition	Pour remédier à la vulnérabilité liée au JSON Web Token (JWT), il est crucial d'améliorer la sécurité de sa gestion. Cela inclut le renforcement de la signature du JWT pour empêcher la falsification et l'ajout du drapeau "Secure" dans les cookies pour assurer la transmission sécurisée sur HTTPS uniquement.	

Présentation des preuves :

Advisory	Request	Response	Path to issue
! JWT weak HMAC secret			
Issue: JWT weak HMAC secret Severity: High Confidence: Certain Host: http://localhost:3000 Path: /user/3			
Issue detail Detected a JWT signed using a well-known HMAC secret key. The key used was @QEGTUI.			
Encoded PASTE A TOKEN HERE eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VySWQiOjEsInVzZXJuYW1lIjoiYWRtaW4iLCJpYXQiOjE3MDkwNjczMDEsImV4cCI6MTcwOTA3MDkwMX0.S-ZxmlxMYztDeI1qcvpWmw8iyPV10PLUuKInFuB-UM	Decoded EDIT THE PAYLOAD AND SECRET HEADER: ALGORITHM & TOKEN TYPE <pre>{ "alg": "HS256", "typ": "JWT" }</pre> PAYLOAD: DATA <pre>{ "userId": 1, "username": "admin", "iat": 1709067301, "exp": 1709070901 }</pre> VERIFY SIGNATURE <pre>HMACSHA256(base64UrlEncode(header) + "." + base64UrlEncode(payload), @QEGTUI.) <input type="checkbox"/> secret base64 encoded</pre> <p>Weak secret!</p>		
Signature Verified		SHARE JWT	

JWT secret signature est très faible

Encoded PASTE A TOKEN HERE

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJc2VySWQiOjQsInVzZXJuYW1lIjoidGVzdC1sImhdCI6MTcyMDMwODE0MSwiZXhwIjoxNzIwMzExNzQxfQ._j4poygwYe7COUrwh_DkjidpjXjvCBEzMDYp3jic7E8
```

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```
{  
  "alg": "HS256",  
  "typ": "JWT"  
}
```

PAYLOAD: DATA

```
{  
  "userId": 4,  
  "username": "test",  
  "iat": 1720308141,  
  "exp": 1720311741  
}
```

Encoded PASTE A TOKEN HERE

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJc2VySWQiOjEsInVzZXJuYW1lIjoiYWRtaW4iLCJpYXQiOjE3MjAzMDgxNDEsImV4cCI6MTcyMDMxMTc0MX0.IteVeixIhExK8Tp9Lgn_WGBrlv6dlyM3FxK09PsWAc0
```

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```
{  
  "alg": "HS256",  
  "typ": "JWT"  
}
```

PAYLOAD: DATA

```
{  
  "userId": 1,  
  "username": "admin",  
  "iat": 1720308141,  
  "exp": 1720311741  
}
```

VERIFY SIGNATURE

```
HMACSHA256(  
  base64UrlEncode(header) + "." +  
  base64UrlEncode(payload),  
  @QEGTUI Weak secret!  
)  secret base64 encoded
```

⌚ Signature Verified

SHARE JWT

Poc usurpation de compte : connexion avec un compte test sans droit, prendre le cookie de l'utilisateur, le décoder et ajouter le weak secret découvert @GEGTUI, remplacer le nom par celui d'un admin ou autre, remplacer dans la page web le token avec celui-ci créé, et rafraîchir la page

VULNERABILITES

Configuration and Deployment Management Testing

ID: 11	BACKUP AND UNREFERENCED FILES	HIGH
Chemin(s)	http://localhost:3000/swagger-stats	
Constat	Des fichiers potentiellement sensibles référencés par le serveur web sont accessibles.	
Risque technique	Certains fichiers référencés peuvent contenir des sources de scripts, des mots de passe, des fichiers de configuration ou d'autres informations sensibles pouvant aider un utilisateur malveillant à préparer des attaques plus avancées.	
Précognition	Pour sécuriser les fichiers sensibles potentiellement accessibles via le serveur web, il est essentiel de mettre en place des mesures strictes de gestion des fichiers de sauvegarde et des fichiers non référencés. Cela inclut la suppression ou le déplacement hors de la racine du serveur web des fichiers contenant des sources de scripts, des mots de passe, des fichiers de configuration et d'autres informations sensibles	

Présentation des preuves :

The screenshot shows the Swagger-stats interface at `localhost:3000/swagger-stats/ux#/requests`. The main content area displays a table of requests by method, two large numerical values (49 and 2669), and a donut chart.

Requests Table:

Method	Requests	Responses	Apdex Score	Errors	Req rate	Err rate	Success
GET	49	49	0.37	21	0.00	0.00	
POST	2669	2669	0.01	2648	0.03	0.03	
PUT	0	0	0.00	0	0.00	0.00	
DELETE	0	0	0.00	0	0.00	0.00	
PATCH	13	13	0.35	8	0.00	0.00	

Request Count Summary:

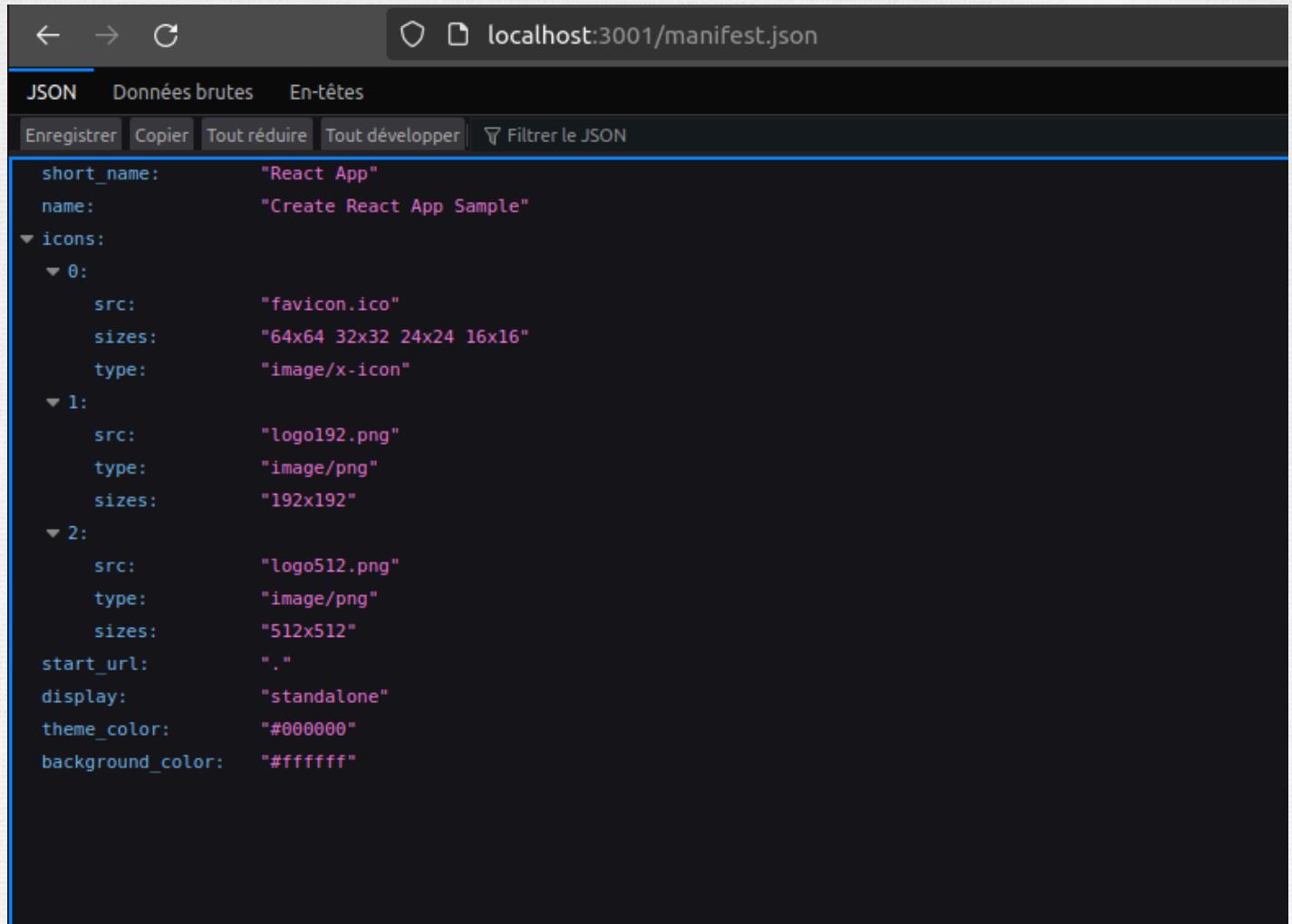
- GET: 49 (1.79%)
- POST: 2669 (98%)

Requests by method:

A donut chart illustrating the distribution of requests by method. The chart is almost entirely orange, representing 98% of requests, with a very small blue slice representing 2% of requests.

Method	Percentage
GET	2%
POST	98%
PUT	0%
DELETE	0%
PATCH	0%

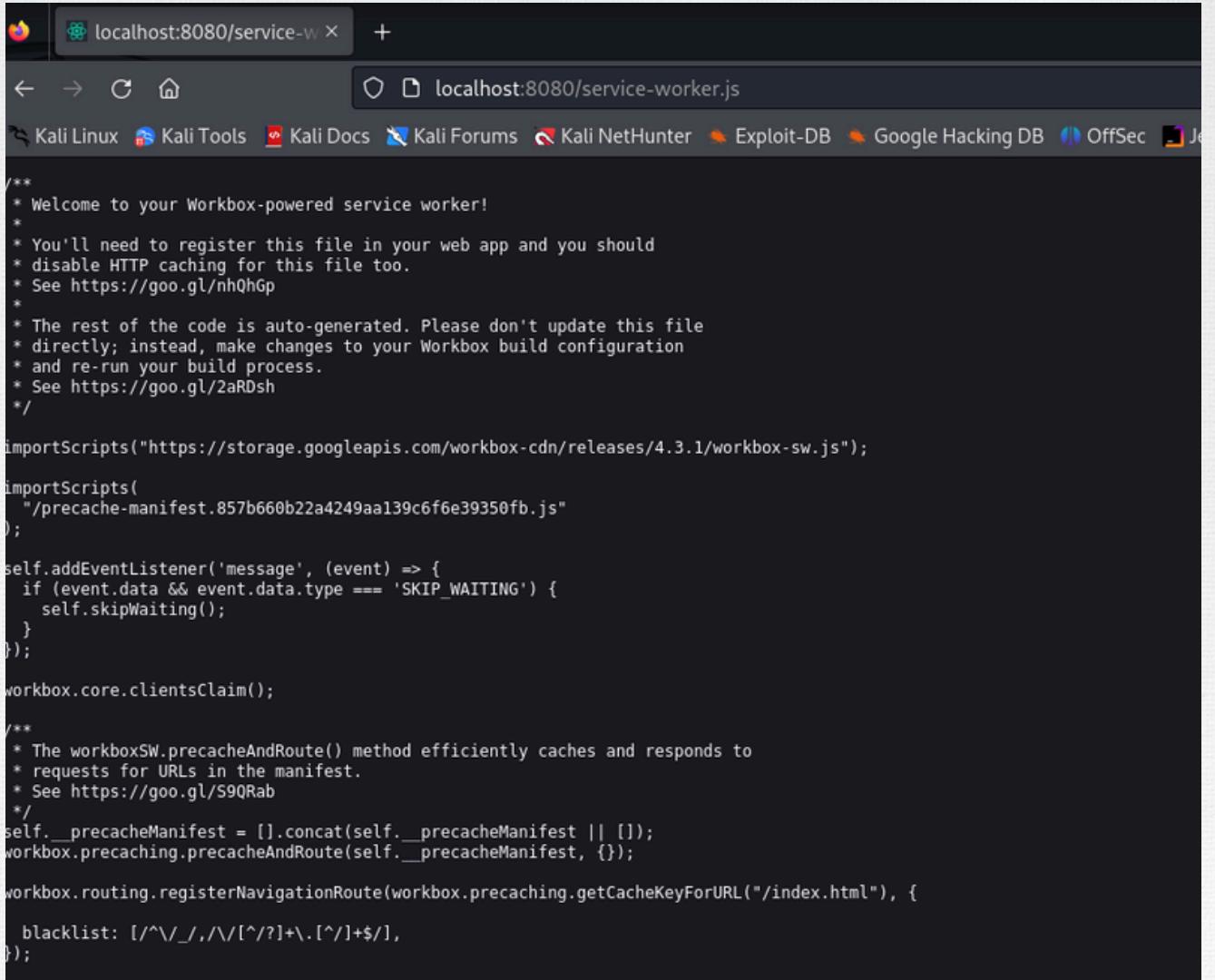
Swagger-stats accessible de manière non-authentifié



The screenshot shows a browser window displaying the contents of a JSON file at the URL `localhost:3001/manifest.json`. The browser interface includes standard navigation buttons (back, forward, refresh) and a toolbar with options like "Enregistrer" (Save), "Copier" (Copy), "Tout réduire" (Minimize All), "Tout développer" (Expand All), and "Filtrer le JSON" (Filter JSON). The main content area is a JSON viewer showing the following data:

```
short_name: "React App"
name: "Create React App Sample"
icons:
  0:
    src: "favicon.ico"
    sizes: "64x64 32x32 24x24 16x16"
    type: "image/x-icon"
  1:
    src: "logol92.png"
    type: "image/png"
    sizes: "192x192"
  2:
    src: "logo512.png"
    type: "image/png"
    sizes: "512x512"
start_url: "."
display: "standalone"
theme_color: "#000000"
background_color: "#ffffff"
```

Fichier de configuration accessible : manifest.json



The screenshot shows a web browser window with the URL `localhost:8080/service-worker.js` in the address bar. The page content displays the source code of a service worker script. The code is well-commented, explaining the purpose of various sections like importing workbox, adding event listeners, and configuring precaching and routing.

```
/*  
 * Welcome to your Workbox-powered service worker!  
 *  
 * You'll need to register this file in your web app and you should  
 * disable HTTP caching for this file too.  
 * See https://goo.gl/nhQhGp  
 *  
 * The rest of the code is auto-generated. Please don't update this file  
 * directly; instead, make changes to your Workbox build configuration  
 * and re-run your build process.  
 * See https://goo.gl/2aRDsh  
 */  
  
importScripts("https://storage.googleapis.com/workbox-cdn/releases/4.3.1/workbox-sw.js");  
  
importScripts(  
  "/precache-manifest.857b660b22a4249aa139c6f6e39350fb.js"  
);  
  
self.addEventListener('message', (event) => {  
  if (event.data && event.data.type === 'SKIP_WAITING') {  
    self.skipWaiting();  
  }  
});  
  
workbox.core.clientsClaim();  
  
/*  
 * The workboxSW.precacheAndRoute() method efficiently caches and responds to  
 * requests for URLs in the manifest.  
 * See https://goo.gl/S9QRab  
 */  
self.__precacheManifest = [].concat(self.__precacheManifest || []);  
workbox.precaching.precacheAndRoute(self.__precacheManifest, {});  
  
workbox.routing.registerNavigationRoute(workbox.precaching.getCacheKeyForURL("/index.html"), {  
  blacklist: [/^/_/, /[^?]+\.[^/]+$/],  
});
```

Fichier de configuration accessible : service-worker.js

VULNERABILITES

Vulnerable and Outdated Components

ID: 12	VULNERABLE MODULE	HIGH
Version	Node version 11 (docker-image mhart/alpine-node@11)	
Constat	Les versions affectées de ce package sont vulnérables à l'injection de code en raison de la gestion incorrecte des variables d'environnement sous Linux lorsque le processus s'exécute avec des privilèges élevés que l'utilisateur actuel n'a pas.	
Risque technique	Cette vulnérabilité permet potentiellement à des utilisateurs non privilégiés d'injecter du code dans un processus Node.js privilégié par le biais de variables d'environnement telles que NODE_BIND_SERVICE. à travers des variables d'environnement telles que NODE_OPTIONS.	
Précognition	Pour sécuriser le module vulnérable utilisant Node.js version 11 dans l'image Docker mhart/alpine-node@11, il est crucial de mettre à jour vers une version plus récente de Node.js qui corrige cette vulnérabilité spécifique. En outre, il est recommandé de limiter les privilèges du processus Node.js autant que possible et de ne pas exposer des variables d'environnement sensibles comme NODE_OPTIONS à des utilisateurs non privilégiés.	

Présentation des preuves :

HIGH SEVERITY

NEW

🛡️ Code Injection

Vulnerable module: node

Introduced through: node@11.8.0

Detailed paths

- Introduced through: docker-image|mhart/alpine-node@11.8 > node@11.8.0

Overview

node is a JavaScript runtime built on Chrome's V8 JavaScript engine.

Affected versions of this package are vulnerable to Code Injection due to the incorrect handling of environment variables on Linux when the process is running with elevated privileges that the current user lacks (does not apply to `CAP_NET_BIND_SERVICE`).

Remediation

Upgrade `node` to version 18.19.1, 20.11.1, 21.6.2 or higher.

References

- GitHub Commit
- GitHub Commit
- GitHub Commit
- RedHat Bugzilla Bug

[Code Injection vulnerability report](#)

Lien résumant les failles de cette version de package et comment les exploiter:

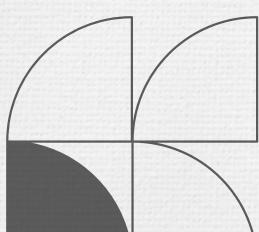
<https://nodejs.org/en/blog/vulnerability/february-2024-security-releases/#code-injection-and-privilege-escalation-through-linux-capabilities-cve-2024-21892---high>

RECOMMANDATIONS

1. Configurer CORS de manière plus restrictive pour limiter les sources autorisées.
2. Implémenter HTTPS pour garantir une communication sécurisée.
3. Renforcer les vérifications de rôle côté frontend pour prévenir le contournement via proxy.
4. Examiner et sécuriser les fichiers de sauvegarde, en s'assurant qu'ils ne sont pas accessibles publiquement.
5. Corriger les erreurs de gestion pour éviter la divulgation d'informations sensibles.
6. Restreindre l'accès aux fichiers de configuration pour limiter la divulgation d'informations.
7. Bloquer la mise en production/mise en ligne si le développement est cours ou si un trop grand nombre de vulnérabilités est détecté
8. Mettre en place un contrôle d'accès strict basé sur le principe du moindre privilège
9. Auditer régulièrement les dépendances et bibliothèques tierces
10. Implémenter une stratégie de gestion des incidents de sécurité

POINT D'ATTENTION

Ce rapport met en lumière plusieurs vulnérabilités potentielles sur le site "DreamHabitat". Les recommandations fournies visent à renforcer la sécurité globale de l'application. Il est recommandé de prendre des mesures correctives immédiates pour minimiser les risques potentiels.



ANNEXES

Référence:

Ce document s'appuie sur la méthodologie et les tests décrits dans le guide "OWASP Web Security Testing Guide"