

ASSIGNMENT CHECKLIST & DOCUMENTATION

BENR 3433

No.	Assets	Threat Agent	Attack surface	Attack Goal	Impact	Controls (Mitigation)
1	VMS administrative credential stored in the database	attacker	Improper API implementation of the backend server? Improper access control implementation at the UI, backend and database server.	Obtain user credential?	Confidentiality? Integrity? Availability?	Isolation and segmentation of VMs, patch
2	System configuration of the VMS stored in the database	Attacker/visitor	Physical database storage (loss of VMS configuration) API that could modify the database storage (corruption of VMS configuration)	Service disruption of VMS Service disruption of VMS	Availability Availability and Integrity	Backup and recovery of database, access control of API, monitoring of VM activity
3	VMS host credential stored in the database	Attacker/visitor	API (which API) that could exposed the host' confidential detail API (which API) and business logic that could exposed the visitor records	Obtain the contact detail of the hosts using the VMS for subsequent malicious activities. Obtain the visit detail of the visitors to the compromised host.	Confidentiality Confidentiality	Encryption of data, authentication of API users, isolation and segmentation of VMs
4	Visitor pass	Attacker/visitor	Fake the visitor pass	Cheat the security to enter the restricted area	Integrity	Verification of visitor pass, physical security

INFORMATION SECURITY

<https://vmsgp17.azurewebsites.net/api-docs/>

GROUP 17

MUHAMMAD FAHMI HAFIZI BIN ROSMIDI

MUHAMMAD HAZIQ HANAFI BIN HAFEDI

Public API for security to create a new host account

1.1 API required security approval (e.g. /create/host)

1.2 Testing API without security approval (e.g. /create/test/host)

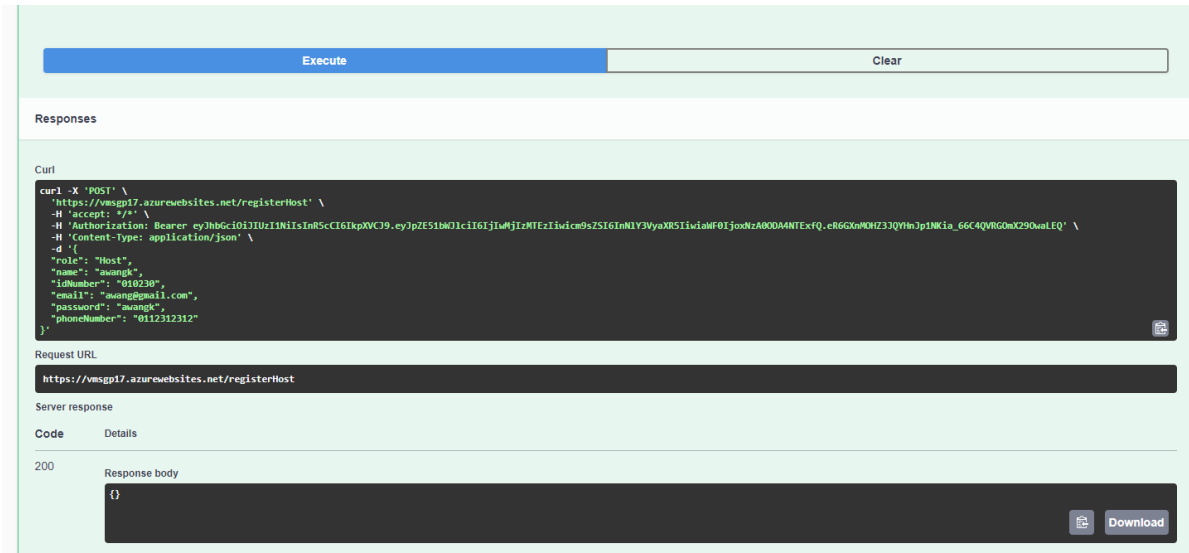


Figure 1.1: register new host with security approval.

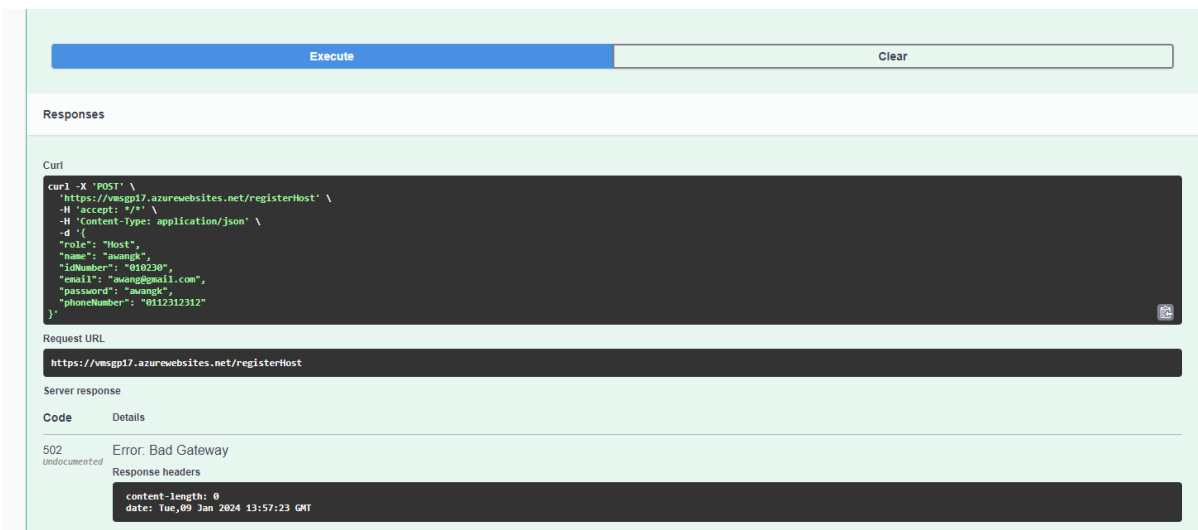


Figure 1.2: register new host without security approval.

Current weakness

The code handles errors differently. In the first `if (err)` block, it logs the error and returns a 401 status, while in the catch block, it logs the error and returns a 500 status. Consistent error handling would be more maintainable

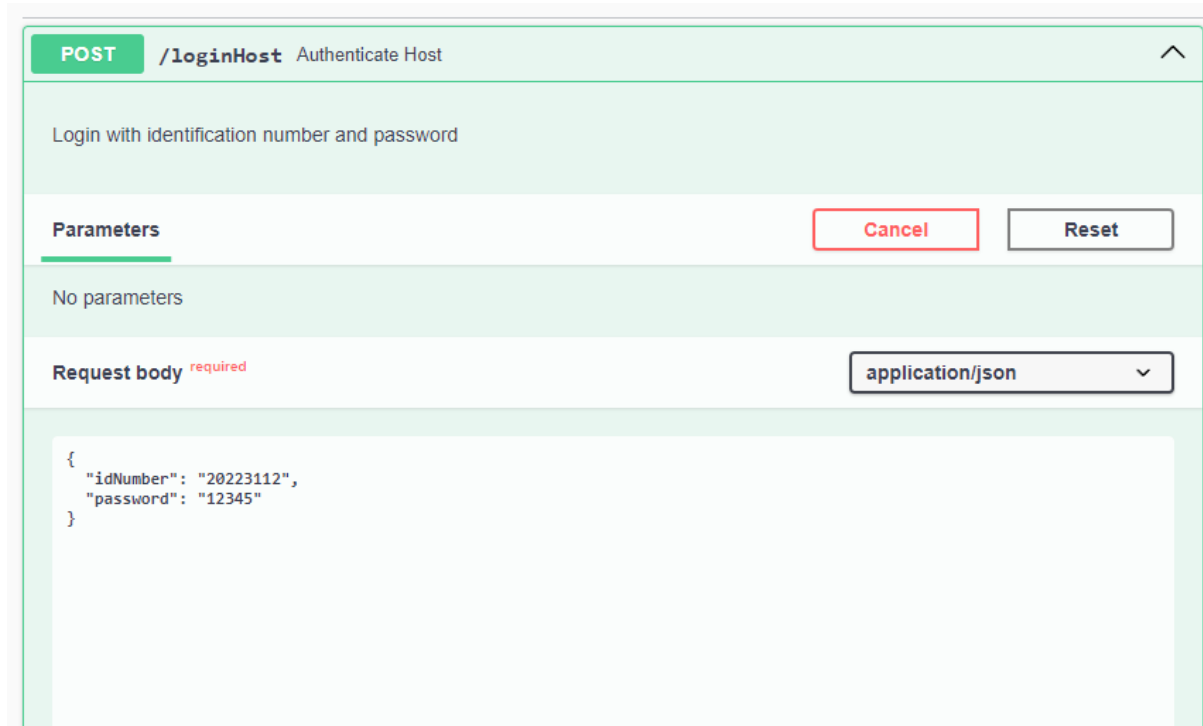
Plan to secure

Avoid exposing detailed error messages to clients. Instead, use generic messages that do not reveal sensitive information

Estimated date of completion

18/1/2023

Checklist 2:



POST /loginHost Authenticate Host

Login with identification number and password

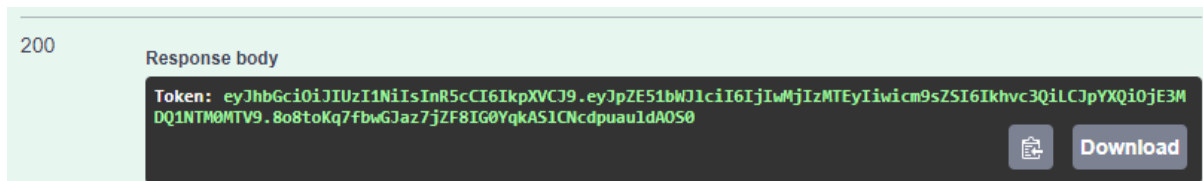
Parameters Cancel Reset

No parameters

Request body required application/json

```
{
  "idNumber": "20223112",
  "password": "12345"
}
```

Figure 2.1: Entering host's idNumber and password.



200

Response body

Token: eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZE51bWJ1ciI6IjIwMjIzMTIiLCJpYXQiOiJlZ3M0MTIwMTV9.8o8toKq7fbwGJaz7jZF8IG0YqkAS1CNcdpuau1dA0S0

Download

Figure 2.2: Token for host being generated.

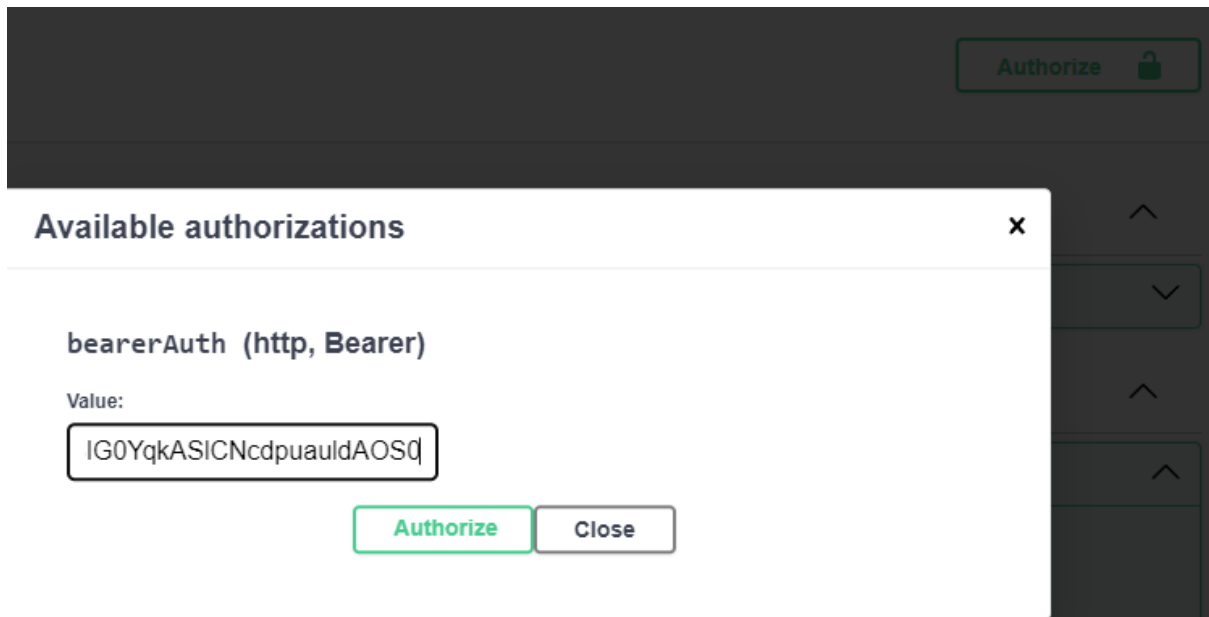


Figure 2.3: enter host's token into authorization value.

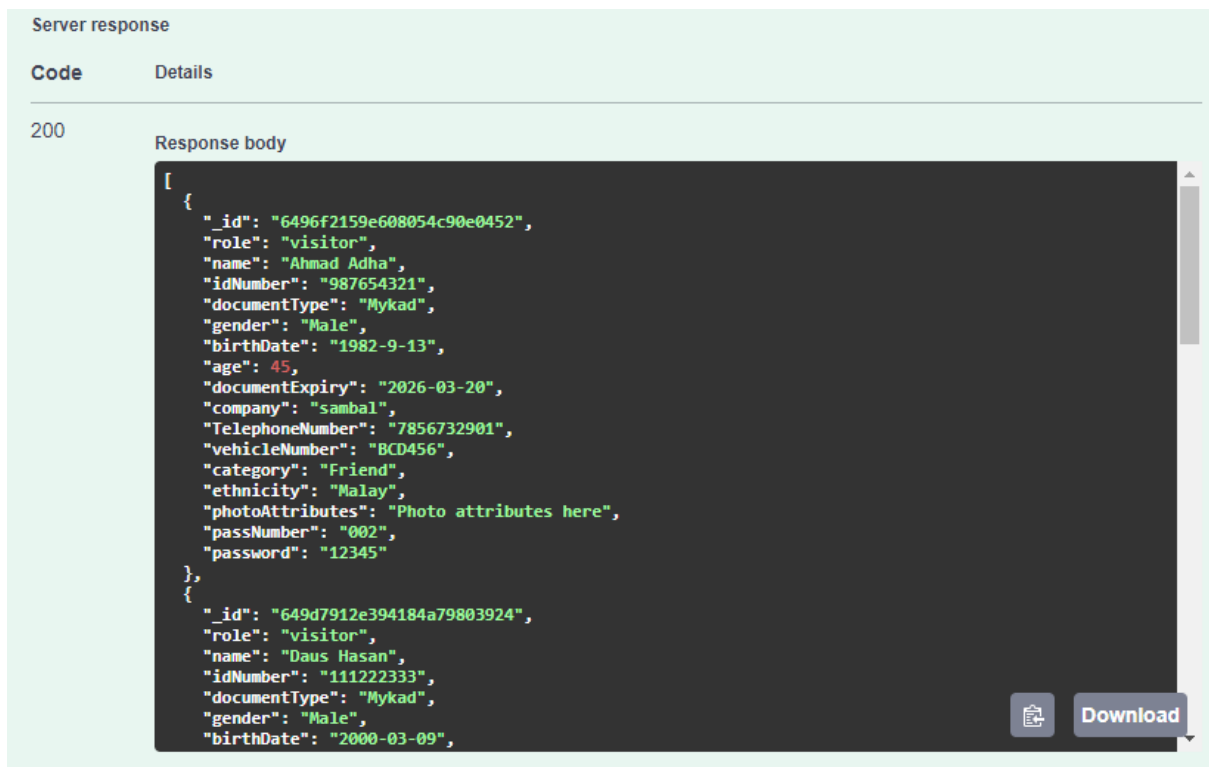


Figure 2.4: List for visitor being generated.

Current weakness

The code assumes that a valid token guarantees a valid user with the necessary permissions.

Plan to secure

Verify that the token has not expired by checking the exp (expiration time) claim. If a token has expired, the server should reject it, and the client must obtain a new token.

Checklist 3:

Figure 3.1: Entering Host's idNumber and password. Then execute.

Figure 3.2: Token for host being generated. Copy all the token.

Figure 3.3: paste all the tokens in the authorization.

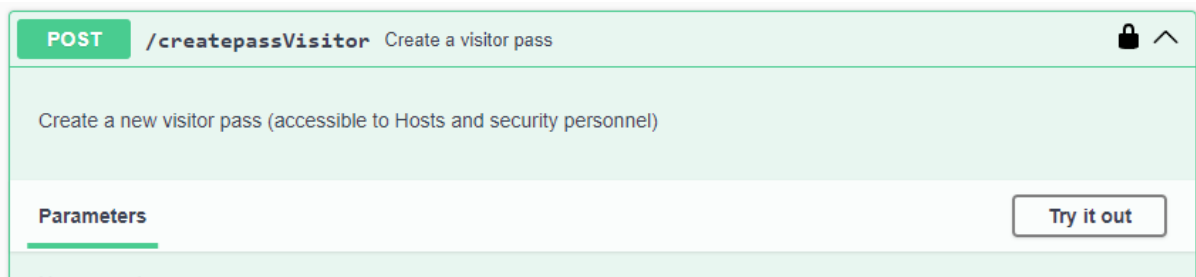


Figure 3.4: Open createpassVisitor.



Figure 3.5: Fill in all the blank space.



Figure 3.6: it will generate this in the response body.

Current weakness

Our code does not implement anti-CSRF protection for the visitor registration endpoint

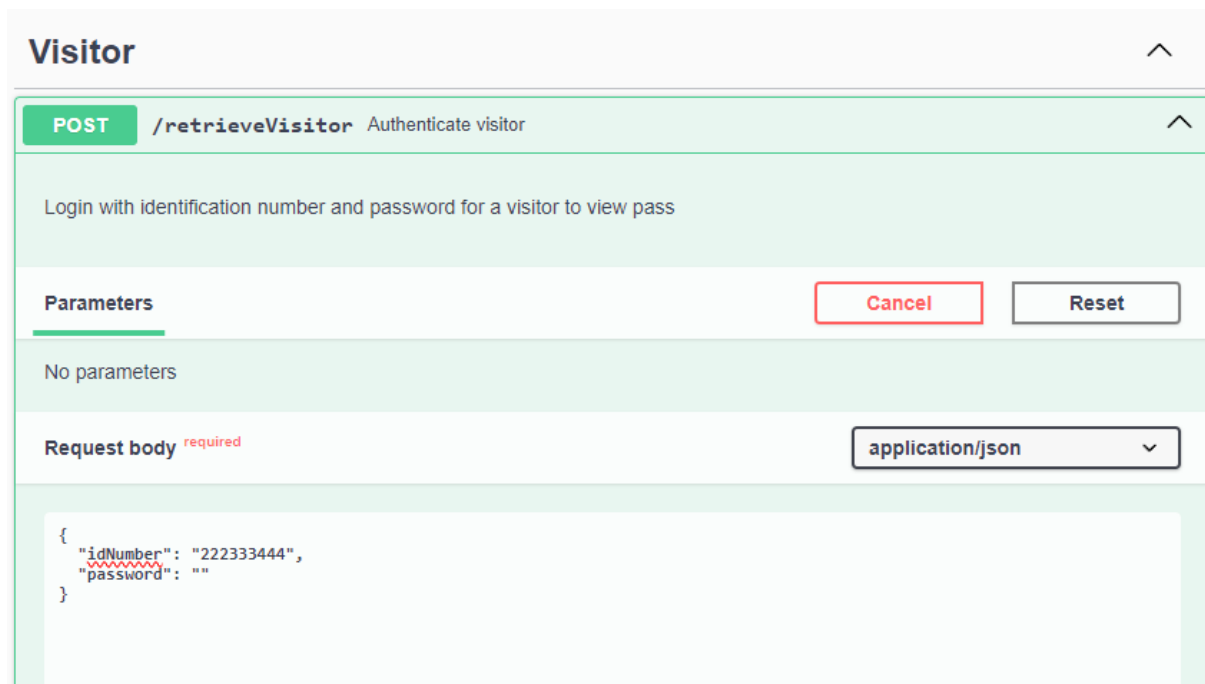
Plan to secure

Adding CSRF tokens or other protective measures can to prevent CSRF attacks.

Estimate date of completion

18/1/2023

Checklist 4:



The screenshot shows a REST client interface for a 'Visitor' endpoint. The method is POST, and the URL is /retrieveVisitor. The description is 'Authenticate visitor'. Below the URL bar, there is a text input field with the placeholder 'Login with identification number and password for a visitor to view pass'. There are 'Parameters' and 'Request body' sections. The 'Parameters' section is empty. The 'Request body' section is set to 'application/json' and contains a JSON object: { "idNumber": "222333444", "password": "" }. There are 'Cancel' and 'Reset' buttons.

Figure 4.1: Visitor entering their idNumber and password.



The screenshot shows the 'Response body' of the POST request. The status is 200. The response is a JSON object: { "Token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZE51bWJ1ciI6IjIyMjMzQ0NCIsInJvbGU0i0i02aXNpdG9yIiwiaWF0IjoxNzA0NTU0Mjc0fQ.8yq54wa4jHrnm-JfeUm92WJCWgNEtZ00krMV3iw_ws", "Visitor Info": { "_id": "65995879887c4bffe8da857f", "role": "visitor", "name": "Amirul Ahmad", "idNumber": "222333444", "documentType": "Mykad", "gender": "Male", "birthDate": "2000-07-02", "age": 29, "documentExpiry": "2023-07-02", "company": "cili", "TelephoneNumber": "0185371321", "vehicleNumber": "WNY2032", "category": "Guest", "ethnicity": "Indian", "photoAttributes": "None", "passNumber": "011", "password": null } } }. There are 'Download' and 'Copy' buttons.

Figure 4.2: visitor's data being generated.

Current weakness

The code contains two `res.send` statements, one inside an if condition and another outside. This is likely unintentional and may lead to unexpected behavior.

Plan to secure

Rebuild the code by using only one `res.send`.

Estimate date of completion

18/1/2023

Checklist 5:

POST /loginAdmin Authenticate administrator personnel

Login with identification number and password

Parameters

No parameters

Request body required

application/json

```
{
  "idNumber": "20223119",
  "password": "67890"
}
```

Figure 5.1: Entering admin idNumber and password. Then execute.

Response body

Token: eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZE51bWJ1ciI6IjIwMjE5Iiwicm9sZSI6ImFkbWludIiwiaWF0IjoxNzA0NTU0OTA4fQ.CWgt6T6Ypr1E8f7Rq1cLSTKrFi-KrnISbsMNurXF56Y

Download

Figure 5.2: Admin token being generated. Copy all the token.

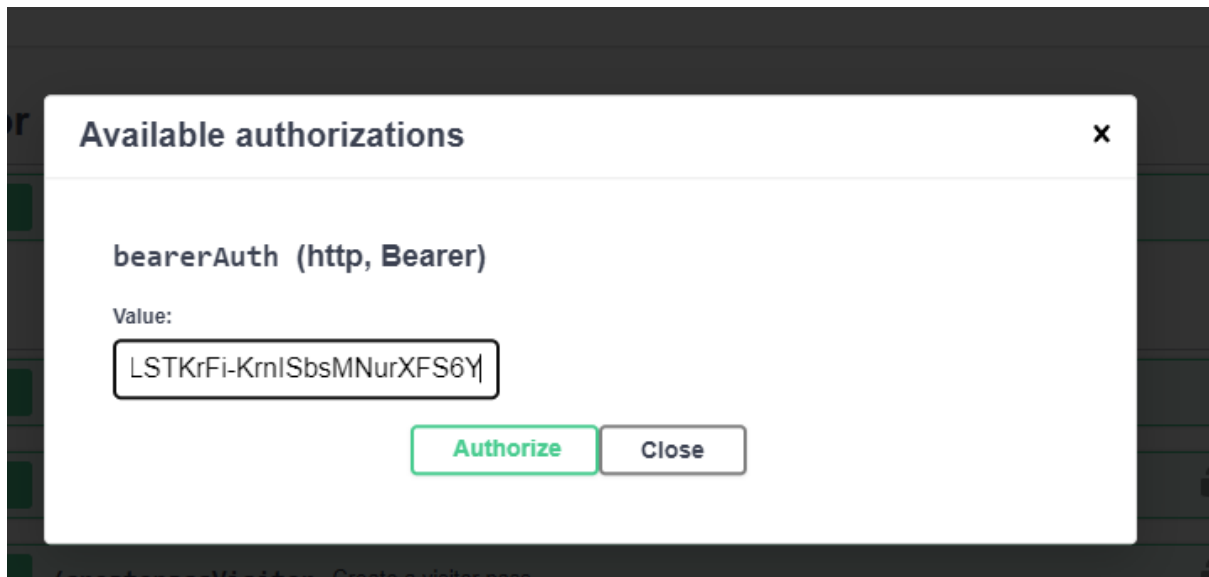


Figure 5.3: Paste the token into authorization value and click authorize.

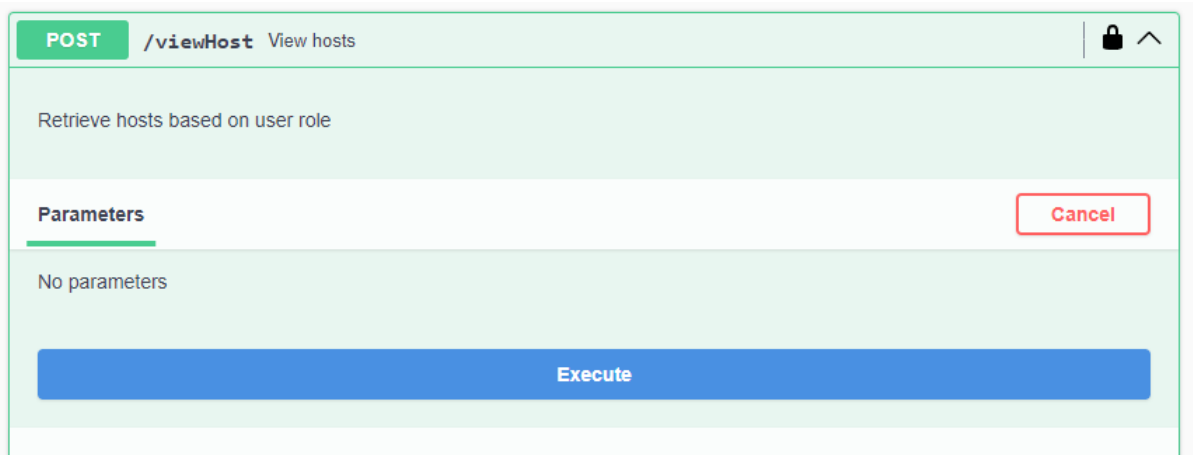


Figure 5.4: Go to viewHost and click execute.



Figure 5.5: The list of host being generated in the response body.

Current weakness

The code lacks comprehensive error handling for database connections and queries. It's essential to include try-catch blocks to handle potential errors during database operations

Plan to secure

Avoid exposing detailed error messages to users. Return generic messages to clients to avoid leaking sensitive information.

Estimate date of completion

18/1/2023

Checklist 6:

- 1.0 Additional APIs(manage account roles (security/host) by authenticated administrator and etc)
- 2.0 Public API for authenticated security to retrieve the contact number of the host from the given visitor pass (the pass should only review destination host to visit to the public)

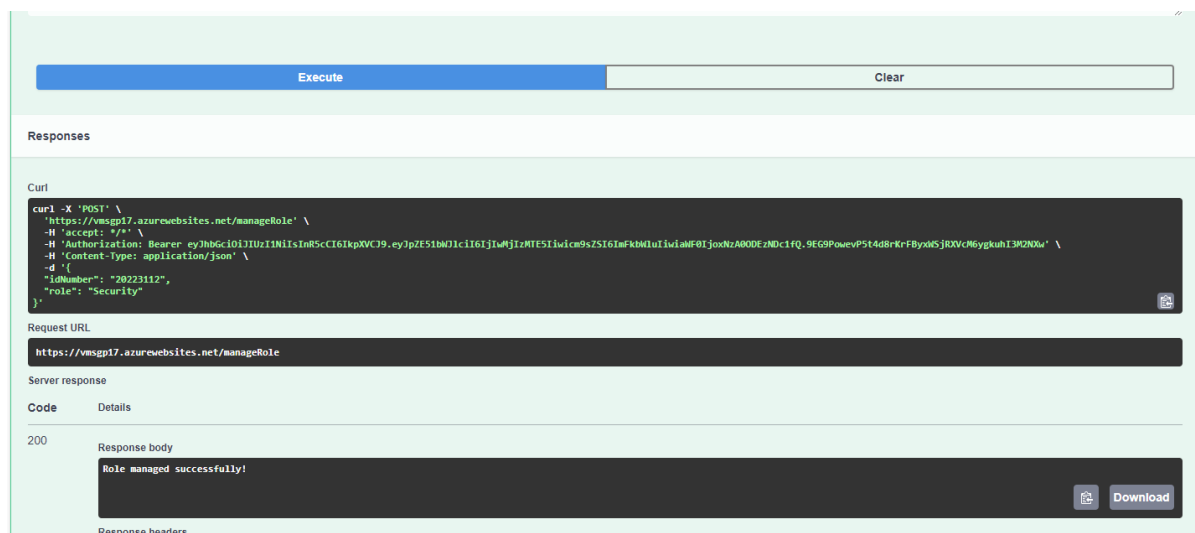


Figure 6.1: change role with security role.