Objectif du lab (SAM)

Utiliser AWS SAM (Serverless Application Model), avec un template YAML, puis déployer depuis le YAML (infra as code) avec SAM CLI.

👉 Un template SAM ne s'importe PAS directement dans la console API Gateway comme OpenAPI.
👉 SAM = CloudFormation : On déploies le YAML, et AWS crée automatiquement l'HTTP API + Lambda.

- Lambda Python
- HTTP API
- Route GET /hello
- Tout défini dans template.yaml
- Déploiement automatique

Sommaire

1 Prérequis sur Windows

On dois avoir :

- AWS CLI v2
- AWS SAM CLI

Installation SAM CLI (Windows)

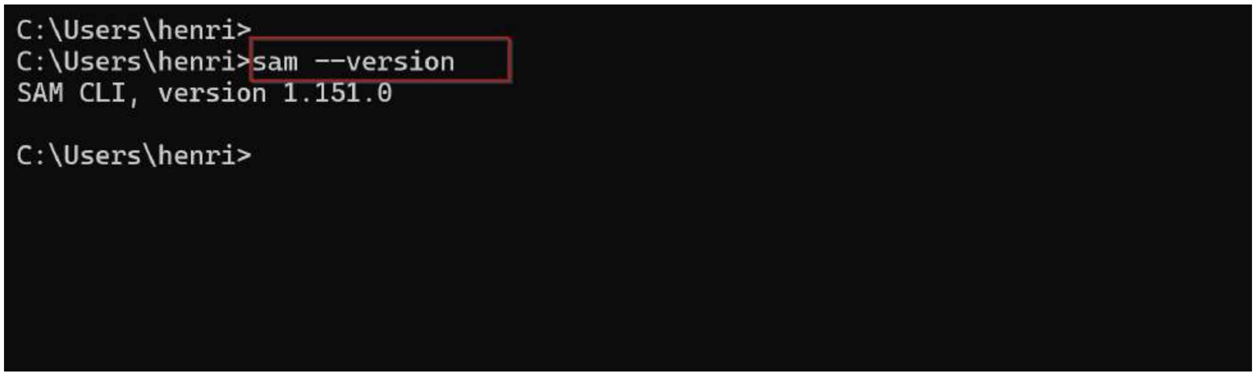
👉 Méthode simple (MSI) :

- https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/install-sam-cli.html

Dans le cas de ce lab, j'ai téléchargé et installé : AWS_SAM_CLI_64_PY3.msi

Vérifier :

```
sam --version
```



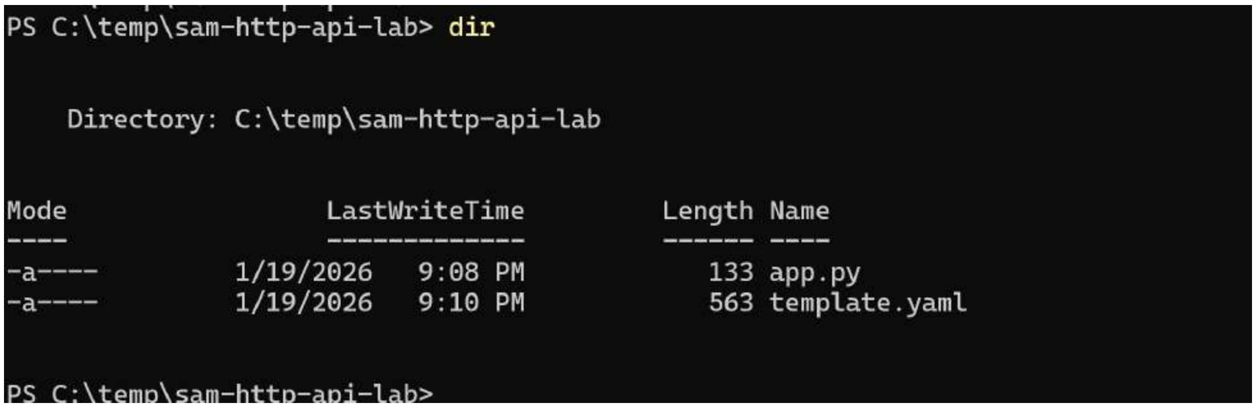
```
C:\Users\henri>
C:\Users\henri>sam --version
SAM CLI, version 1.151.0

C:\Users\henri>
```

2 Structure du projet

Crée ce dossier :

```
sam-http-api-lab/
```

```
├── template.yaml
└── app.py
```



```
PS C:\temp\sam-http-api-lab> dir


    Directory: C:\temp\sam-http-api-lab


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         1/19/2026   9:08 PM            133 app.py
-a----         1/19/2026   9:10 PM            563 template.yaml


PS C:\temp\sam-http-api-lab>
```

---

3 Code Lambda Python (app.py)

```
def lambda_handler(event, context):
    return {
        "statusCode": 200,
        "body": "Hello from HTTP API via SAM"
    }
```

---

4 Template SAM (template.yaml)

👉 Ceci est le cœur du lab

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Description: Simple HTTP API + Lambda (SAM Lab)

Resources:
  HelloFunction:
    Type: AWS::Serverless::Function
    Properties:
      FunctionName: MySimpleLambdaSam
      Runtime: python3.11
      Handler: app.lambda_handler
      CodeUri: .
      MemorySize: 128
      Timeout: 5
      Policies:
        - AWSLambdaBasicExecutionRole
      Events:
        HelloApi:
          Type: HttpApi
          Properties:
            Path: /hello
            Method: GET
```

✅ Ce YAML crée automatiquement :

- Lambda
- HTTP API
- Route
- Permissions API → Lambda

---

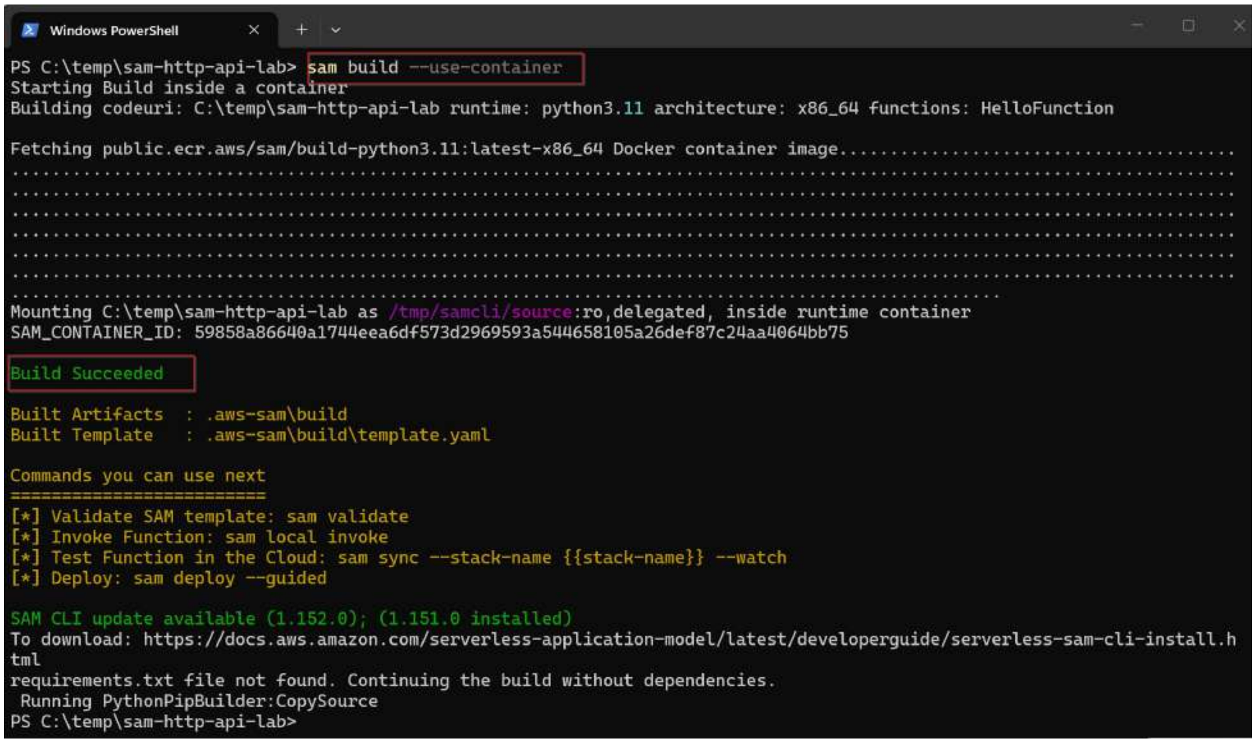
5 Déployer le lab avec SAM

Depuis le dossier du projet :

Étape 1 – Build

```
sam build
 ( ici ne marche pas, car Python est en version 3.13 et SAM supporte
seulement la version 3.11 de python)
donc utiliser python en version 3.11 ou bien utiliser une autre manière de
faire, telle : la commande ci-dessous.

donc utiliser plutôt la commande :
nécessite docker d'installé et docker lancé.
```

```
sam build --use-container
```


Windows PowerShell

```
PS C:\temp\sam-http-api-lab> sam build --use-container
Starting Build inside a container
Building codeuri: C:\temp\sam-http-api-lab runtime: python3.11 architecture: x86_64 functions: HelloFunction

Fetching public.ecr.aws/sam/build-python3.11:latest-x86_64 Docker container image.....................................
..........................................................................................................................
..........................................................................................................................
..........................................................................................................................
..........................................................................................................................
..........................................................................................................................
..........................................................................................................................
...................................................................................................
Mounting C:\temp\sam-http-api-lab as /tmp/samcli/source:ro,delegated, inside runtime container
SAM_CONTAINER_ID: 59858a86640a1744eea6df573d2969593a544658105a26def87c24aa4064bb75

Build Succeeded

Built Artifacts  : .aws-sam\build
Built Template   : .aws-sam\build\template.yaml

Commands you can use next
=========================
[*] Validate SAM template: sam validate
[*] Invoke Function: sam local invoke
[*] Test Function in the Cloud: sam sync --stack-name {{stack-name}} --watch
[*] Deploy: sam deploy --guided

SAM CLI update available (1.152.0); (1.151.0 installed)
To download: https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/serverless-sam-cli-install.h
tml
requirements.txt file not found. Continuing the build without dependencies.
 Running PythonPipBuilder:CopySource
PS C:\temp\sam-http-api-lab>
```

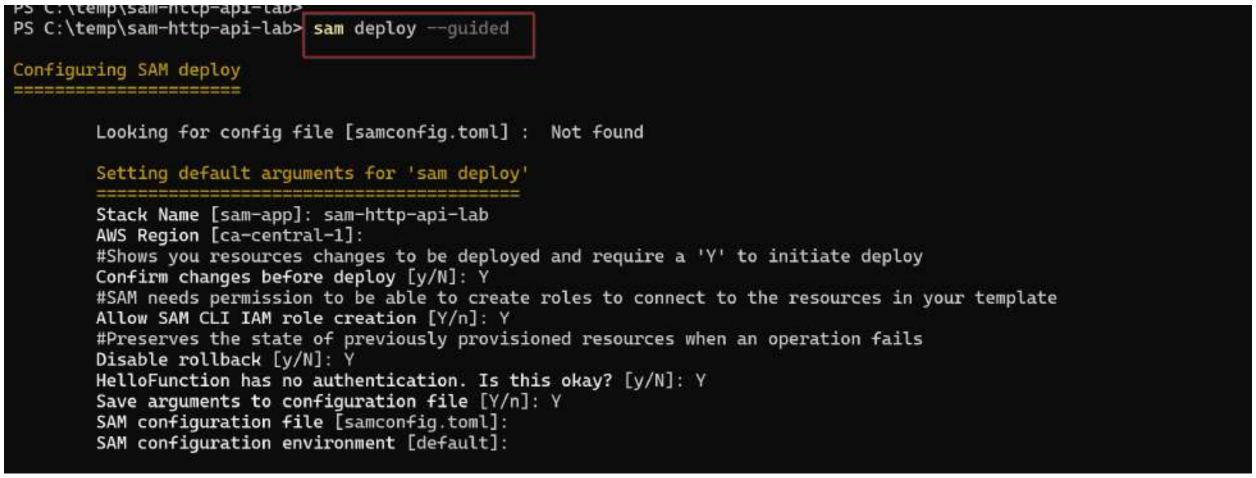
Étape 2 – Deploy guidé (1ère fois)

Il faut ensuite déployer , avec la commande :

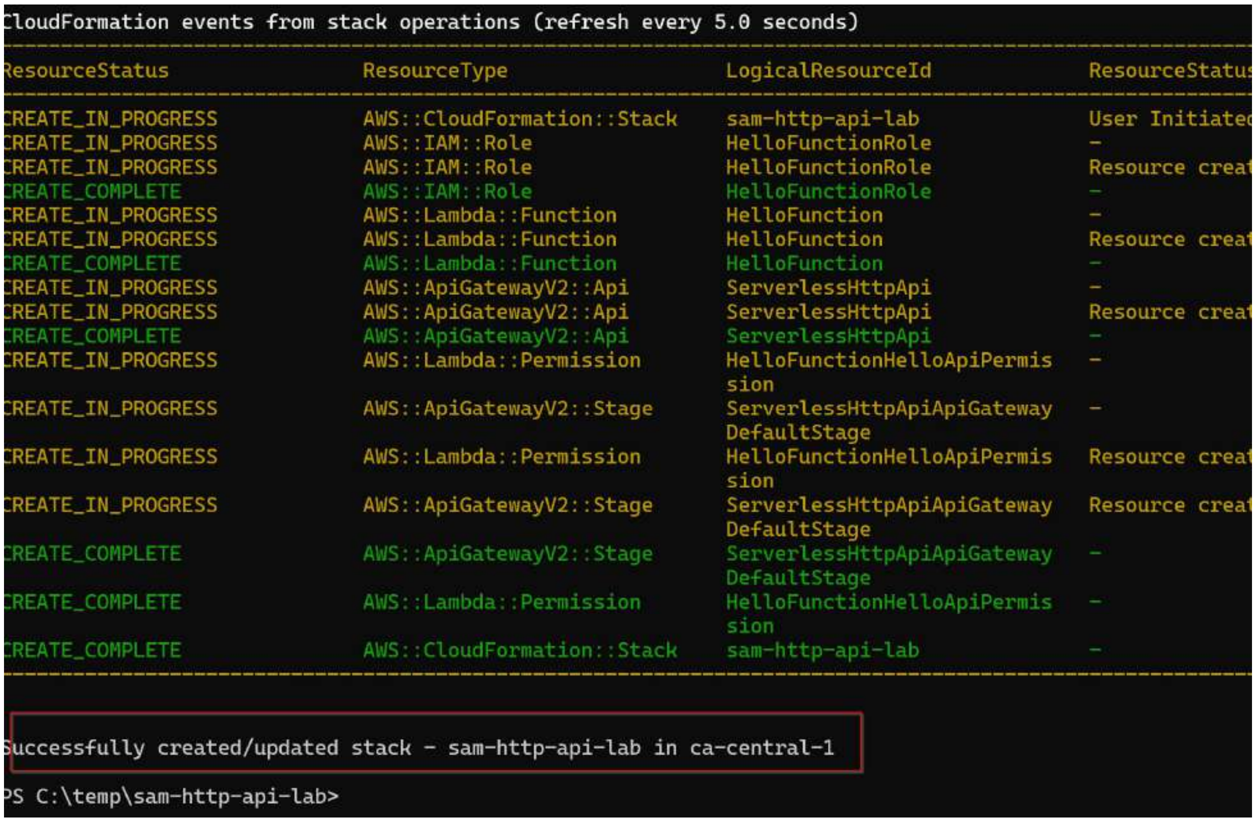
```
sam deploy --guided
```

Réponses recommandées :

- Stack Name : sam-http-api-lab
- Region : ca-central-1
- Confirm changes : Y
- Allow SAM to create IAM roles : Y
- Save arguments : Y



```
PS C:\temp\sam-http-api-lab>
PS C:\temp\sam-http-api-lab> sam deploy --guided

Configuring SAM deploy
======================

        Looking for config file [samconfig.toml] :  Not found

        Setting default arguments for 'sam deploy'
        =========================================
        Stack Name [sam-app]: sam-http-api-lab
        AWS Region [ca-central-1]:
        #Shows you resources changes to be deployed and require a 'Y' to initiate deploy
        Confirm changes before deploy [y/N]: Y
        #SAM needs permission to be able to create roles to connect to the resources in your template
        Allow SAM CLI IAM role creation [Y/n]: Y
        #Preserves the state of previously provisioned resources when an operation fails
        Disable rollback [y/N]: Y
        HelloFunction has no authentication. Is this okay? [y/N]: Y
        Save arguments to configuration file [Y/n]: Y
        SAM configuration file [samconfig.toml]:
        SAM configuration environment [default]:
```



```
CloudFormation events from stack operations (refresh every 5.0 seconds)
-------------------------------------------------------------------------------------------------------------
ResourceStatus              ResourceType                  LogicalResourceId             ResourceStatus
-------------------------------------------------------------------------------------------------------------
CREATE_IN_PROGRESS          AWS::CloudFormation::Stack    sam-http-api-lab              User Initiated
CREATE_IN_PROGRESS          AWS::IAM::Role                HelloFunctionRole             -
CREATE_IN_PROGRESS          AWS::IAM::Role                HelloFunctionRole             Resource creat
CREATE_COMPLETE             AWS::IAM::Role                HelloFunctionRole             -
CREATE_IN_PROGRESS          AWS::Lambda::Function         HelloFunction                 -
CREATE_IN_PROGRESS          AWS::Lambda::Function         HelloFunction                 Resource creat
CREATE_COMPLETE             AWS::Lambda::Function         HelloFunction                 -
CREATE_IN_PROGRESS          AWS::ApiGatewayV2::Api        ServerlessHttpApi             -
CREATE_IN_PROGRESS          AWS::ApiGatewayV2::Api        ServerlessHttpApi             Resource creat
CREATE_COMPLETE             AWS::ApiGatewayV2::Api        ServerlessHttpApi             -
CREATE_IN_PROGRESS          AWS::Lambda::Permission       HelloFunctionHelloApiPermis   -
                                                          sion
CREATE_IN_PROGRESS          AWS::ApiGatewayV2::Stage      ServerlessHttpApiApiGateway   -
                                                          DefaultStage
CREATE_IN_PROGRESS          AWS::Lambda::Permission       HelloFunctionHelloApiPermis   Resource creat
                                                          sion
CREATE_IN_PROGRESS          AWS::ApiGatewayV2::Stage      ServerlessHttpApiApiGateway   Resource creat
                                                          DefaultStage
CREATE_COMPLETE             AWS::ApiGatewayV2::Stage      ServerlessHttpApiApiGateway   -
                                                          DefaultStage
CREATE_COMPLETE             AWS::Lambda::Permission       HelloFunctionHelloApiPermis   -
                                                          sion
CREATE_COMPLETE             AWS::CloudFormation::Stack    sam-http-api-lab              -
-------------------------------------------------------------------------------------------------------------


Successfully created/updated stack - sam-http-api-lab in ca-central-1

PS C:\temp\sam-http-api-lab>
```

6 Récupérer l'URL de l'API

Après le déploiement, SAM est supposé afficher l'URL de l'API http

Mais si on ne le voit pas dans la fenêtre PowerShell (ou cmd), on peut aussi aller dans la console AWS et le récupérer au niveau de Cloudformation et au niveau de API Gateway:

SAM déploie toujours via la création d'un stack de type Cloudformation. Cloudformation décrit les ressources à déployer dans AWS, à l'intérieur d'un stack, puis par la suite, les ressources sont créés. La ressource principale qui est crée est l'API http.

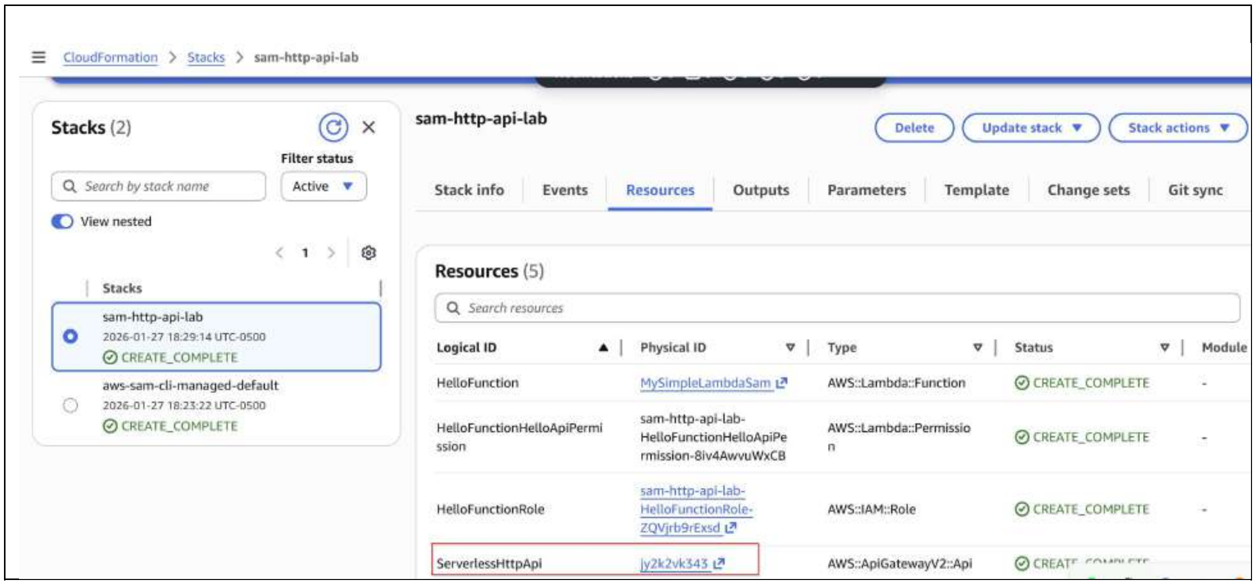
L'API http est crée au niveau de API Gateway

Format général de l'URL qui est crée :

```
HttpApiUrl = https://xxxxx.execute-api.ca-central-1.amazonaws.com
```

6.1 Méthode 1 — Via CloudFormation

1️⃣Console AWS → **CloudFormation**
2️⃣Stack : `sam-http-api-lab`
3️⃣Onglet : Ressources :
Cliquer sur le champs : ''ServerlessHttpApi


CloudFormation > Stacks > sam-http-api-lab

Stacks (2)

Filter status: Active

View nested

Stacks
sam-http-api-lab 2026-01-27 18:29:14 UTC-0500 CREATE_COMPLETE
aws-sam-cli-managed-default 2026-01-27 18:23:22 UTC-0500 CREATE_COMPLETE

sam-http-api-lab

Delete | Update stack | Stack actions

Stack info | Events | Resources | Outputs | Parameters | Template | Change sets | Git sync

Resources (5)

Logical ID	Physical ID	Type	Status	Module
HelloFunction	MySimpleLambdaSam	AWS::Lambda::Function	CREATE_COMPLETE	-
HelloFunctionHelloApiPermission	sam-http-api-lab-HelloFunctionHelloApiPermission-8iv4AwvuWxCB	AWS::Lambda::Permission	CREATE_COMPLETE	-
HelloFunctionRole	sam-http-api-lab-HelloFunctionRole-ZQVjrb9rExsd	AWS::IAM::Role	CREATE_COMPLETE	-
ServerlessHttpApi	jy2k2vk343	AWS::ApiGatewayV2::Api	CREAT[illegible]	

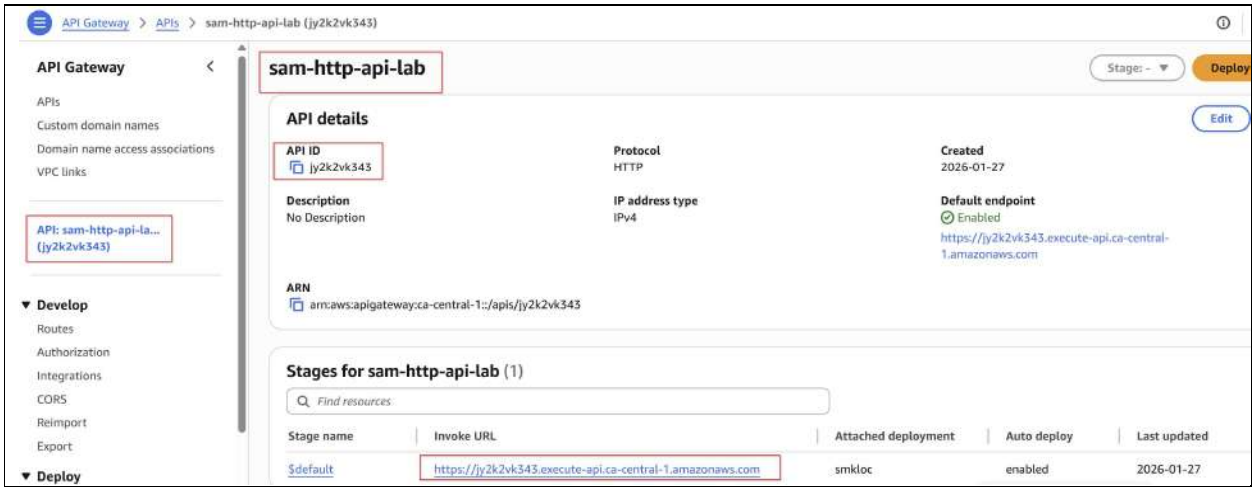
URL : https://jy2k2vk343.execute-api.ca-central-1.amazonaws.com

6.2 Méthode 2— Via API Gateway (HTTP API)

1️⃣ Console AWS → **API Gateway**
2️⃣ **HTTP APIs**
3️⃣ Sélectionner votreAPI
4️⃣ Onglet **Stages** → `$default`

👉 **Invoke URL** = URL de votre API

URL : https://jy2k2vk343.execute-api.ca-central-1.amazonaws.com



sam-http-api-lab

API details

API ID jy2k2vk343	**Protocol** HTTP	**Created** 2026-01-27
Description No Description	**IP address type** IPv4	**Default endpoint** Enabled https://jy2k2vk343.execute-api.ca-central-1.amazonaws.com

ARN arn:aws:apigateway:ca-central-1::/apis/jy2k2vk343

Stages for sam-http-api-lab (1)

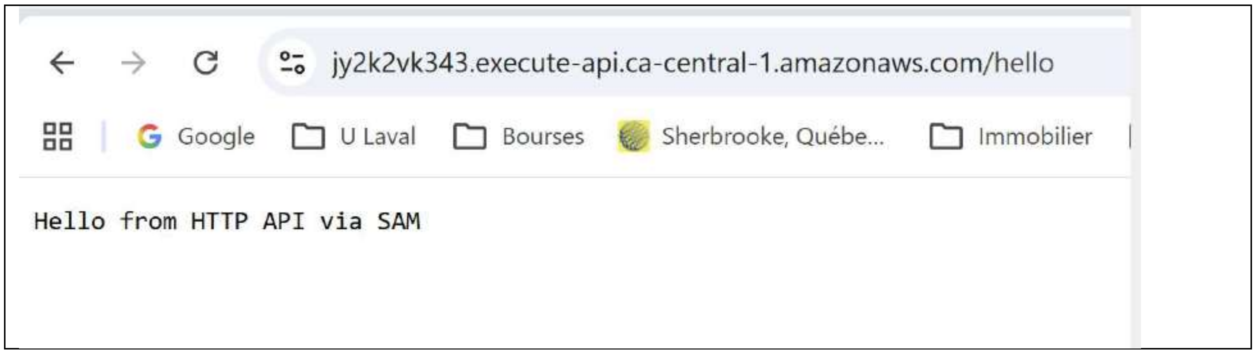
Stage name	Invoke URL	Attached deployment	Auto deploy	Last updated
$default	https://jy2k2vk343.execute-api.ca-central-1.amazonaws.com	smkloc	enabled	2026-01-27

6.3 Tester via le browser

Taper la commande :

`GET` https://jy2k2vk343.execute-api.ca-central-1.amazonaws.com/hello

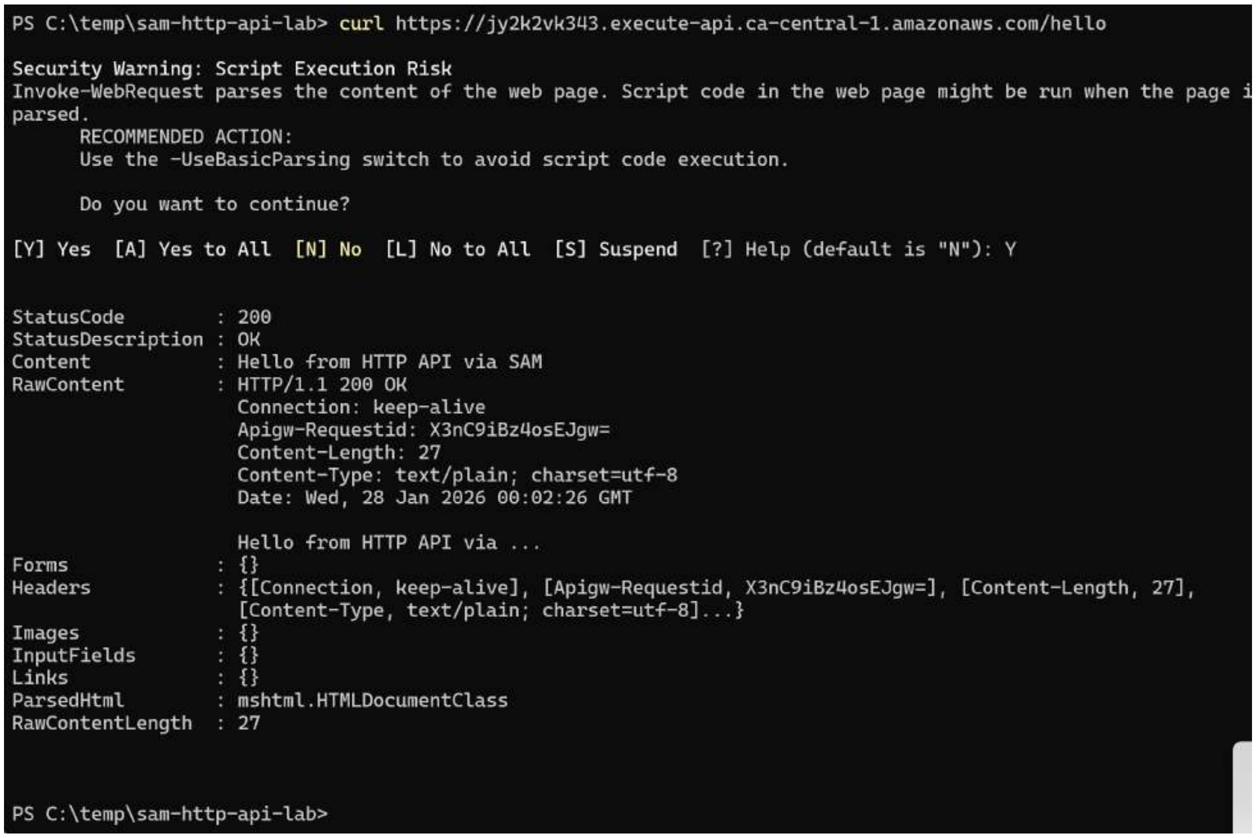
`Reponse :`



```
Hello from HTTP API via SAM
```

6.4 . Tester via powershell (avec CURL)

curl https://jy2k2vk343.execute-api.ca-central-1.amazonaws.com/hello

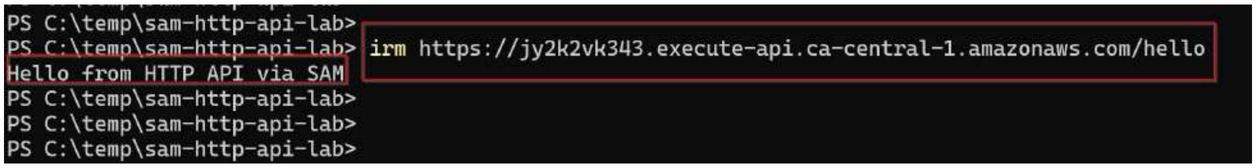


```
PS C:\temp\sam-http-api-lab> curl https://jy2k2vk343.execute-api.ca-central-1.amazonaws.com/hello

Security Warning: Script Execution Risk
Invoke-WebRequest parses the content of the web page. Script code in the web page might be run when the page i
parsed.
      RECOMMENDED ACTION:
      Use the -UseBasicParsing switch to avoid script code execution.

      Do you want to continue?

[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "N"): Y


StatusCode        : 200
StatusDescription : OK
Content           : Hello from HTTP API via SAM
RawContent        : HTTP/1.1 200 OK
                    Connection: keep-alive
                    Apigw-Requestid: X3nC9iBz4osEJgw=
                    Content-Length: 27
                    Content-Type: text/plain; charset=utf-8
                    Date: Wed, 28 Jan 2026 00:02:26 GMT

                    Hello from HTTP API via ...
Forms             : {}
Headers           : {[Connection, keep-alive], [Apigw-Requestid, X3nC9iBz4osEJgw=], [Content-Length, 27],
                    [Content-Type, text/plain; charset=utf-8]...}
Images            : {}
InputFields       : {}
Links             : {}
ParsedHtml        : mshtml.HTMLDocumentClass
RawContentLength  : 27


PS C:\temp\sam-http-api-lab>
```

6.5 Tester via IRM

Avec la commande : Invoke Rest Method (IRM)

irm https://jy2k2vk343.execute-api.ca-central-1.amazonaws.com/hello

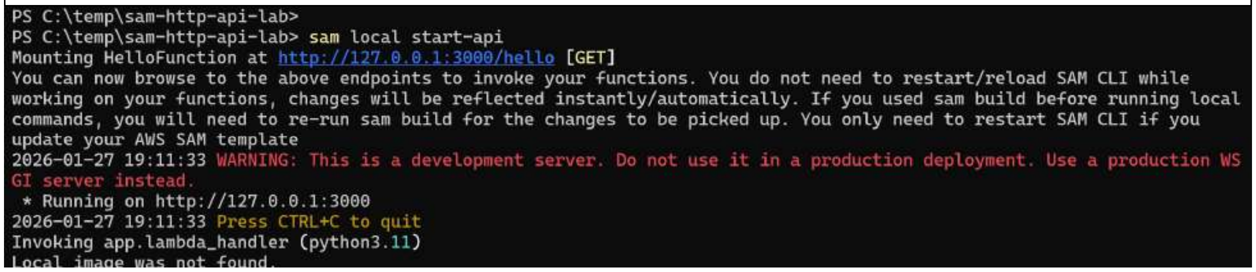


```
PS C:\temp\sam-http-api-lab>
PS C:\temp\sam-http-api-lab> irm https://jy2k2vk343.execute-api.ca-central-1.amazonaws.com/hello
Hello from HTTP API via SAM
PS C:\temp\sam-http-api-lab>
PS C:\temp\sam-http-api-lab>
PS C:\temp\sam-http-api-lab>
```

6.6 Test local

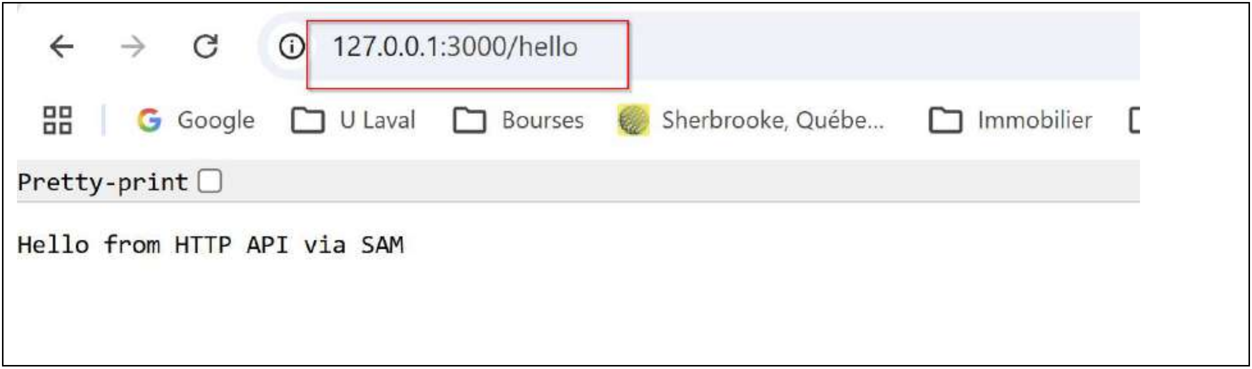
Tester localement avec la commande :

```
sam local start-api
```



```
PS C:\temp\sam-http-api-lab>
PS C:\temp\sam-http-api-lab> sam local start-api
Mounting HelloFunction at http://127.0.0.1:3000/hello [GET]
You can now browse to the above endpoints to invoke your functions. You do not need to restart/reload SAM CLI while
working on your functions, changes will be reflected instantly/automatically. If you used sam build before running local
commands, you will need to re-run sam build for the changes to be picked up. You only need to restart SAM CLI if you
update your AWS SAM template
2026-01-27 19:11:33 WARNING: This is a development server. Do not use it in a production deployment. Use a production WS
GI server instead.
 * Running on http://127.0.0.1:3000
2026-01-27 19:11:33 Press CTRL+C to quit
Invoking app.lambda_handler (python3.11)
Local image was not found.
```

Puis saisir dans le browser :



```
Hello from HTTP API via SAM
```

7 Voir le résultat dans la console AWS

Après le deploy :

- Lambda → MySimpleLambdaSam
- API Gateway → HTTP API auto-créée
- CloudFormation → stack sam-http-api-lab

8 References AWS:

https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/install-sam-cli.html

https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/what-is-sam.html