Full guide on how to create the serverless auto check-in system

Introduction/Concepts

AWS Lambda+Github Webhook+Google Spreadsheet:
https://blog.techbridge.cc/2018/09/14/lambda-and-github-api/

―――

Steps/Methods

Github:

1. Repository settings → webhook → add webhook
   a. Payload url: aws API Gateway deployed stage invoked url + func name (e.g.
      https://escrfx0jpf.execute-api.ap-east-1.amazonaws.com/dev1/comment-issue)
   b. Content type: Application/JSON
   c. Which events would you like to trigger this webhook?
      i. ☑ Let me select individual events.
         1. ☑ "Issue comments"

2. Settings → Developer settings → Personal access tokens → Generate new token
   a. Scopes: repo, user, notifications (should be enough?) or check all options
   b. Copy the access token and create endpoint by appending the token

(e.g. endpoint ='https://api.github.com/repos/kittychengyuenpui/lambda-check-in-test/issues?
access_token=' + token;)

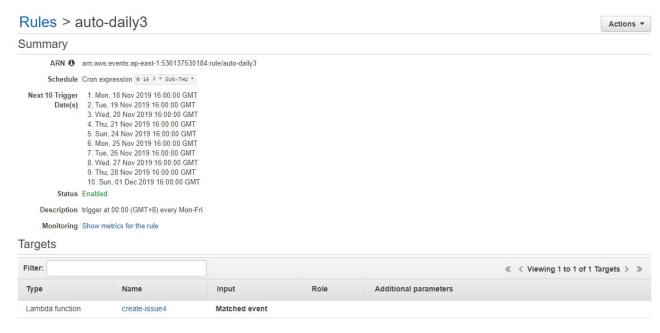―――

Google:

1. Create a service account: service-account-creds.json
2. Share the spreadsheet with the service account
   (service-account-kc2019@silent-thunder-258702.iam.gserviceaccount.com)
3. Transfer ownership to service account (Optional?)

―――

Lambda API:

1. Create api gateway trigger
2. Create method (post)
3. Enable CORS
4. Deploy api

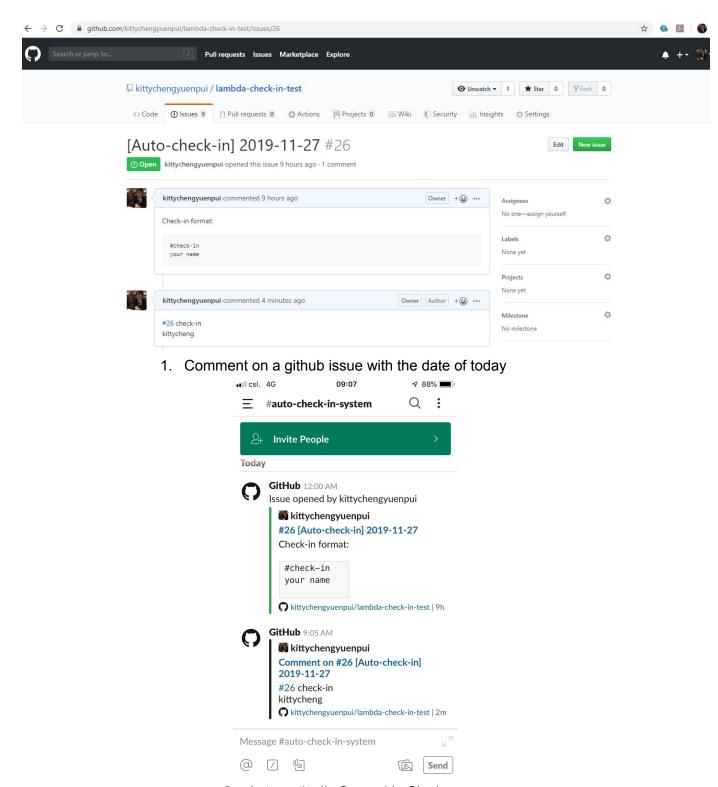―――

Lambda Function:

1. Create lambda functions in aws (create issue and comment issue)
2. Add cloudwatch event to trigger the create issue function while adding api gateway to
   trigger the comment issue function
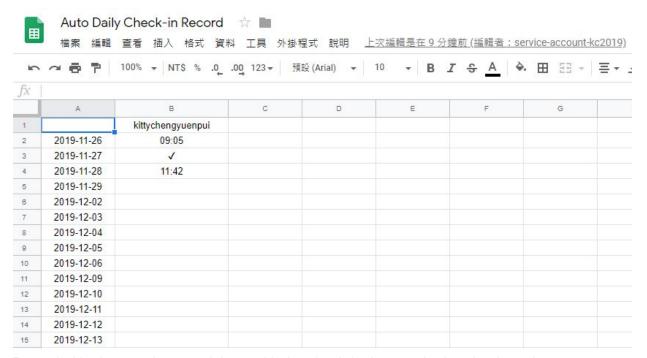   a. Set the cloudwatch event schedule: Cron (0 16 ? * SUN-THU *)

Rules > auto-daily3

Actions ▼

Summary

ARN ⓘ	arn:aws:events:ap-east-1:536137530184:rule/auto-daily3
Schedule	Cron expression `0 16 ? * SUN-THU *`
Next 10 Trigger Date(s)	1. Mon, 18 Nov 2019 16:00:00 GMT
	2. Tue, 19 Nov 2019 16:00:00 GMT
	3. Wed, 20 Nov 2019 16:00:00 GMT
	4. Thu, 21 Nov 2019 16:00:00 GMT
	5. Sun, 24 Nov 2019 16:00:00 GMT
	6. Mon, 25 Nov 2019 16:00:00 GMT
	7. Tue, 26 Nov 2019 16:00:00 GMT
	8. Wed, 27 Nov 2019 16:00:00 GMT
	9. Thu, 28 Nov 2019 16:00:00 GMT
	10. Sun, 01 Dec 2019 16:00:00 GMT
Status	Enabled
Description	trigger at 00:00 (GMT+8) every Mon-Fri
Monitoring	Show metrics for the rule

Targets

Filter: [   ]   « ‹ Viewing 1 to 1 of 1 Targets › »

Type	Name	Input	Role	Additional parameters
Lambda function	create-issue4	Matched event		

The above cloudwatch event set as triggering the create issue function on every Monday to Friday at 12:00 am (GMT+8)

3. Update create issue via inline code while updating comment issue with aws cli, because "If the deployment package of your Lambda function "comment-issue" > 3 MB, it will be too large to enable inline code editing"
   a. Download aws cli
   b. aws configure: aws-rootkey-for-aws-cli.csv + default region name: ap-east-1 + default output format: json
4. Update commands in aws cli (for comment issue):
   aws-lambda-update-function-commands.txt
   a. cd comment-issue
   b. 7z a -tzip -r ../comment-issue *
   c. aws lambda update-function-code --function-name comment-issue --zip-file fileb://comment-issue.zip
5. Set Environment variables for sensitive data in Lambda function (e.g. github personal access token → key:token, value: xxx → process.env.token)

Results

1. Comment on a github issue with the date of today

2. Automatically Synced in Slack

3. Recorded in the google spreadsheet with the check-in time marked under the column of your name and on the row of the date of today by service account

_____

Resources

Simple Google Spreadsheet Access (node.js):
https://github.com/theoephraim/node-google-spreadsheet

_____

Further Implementation

1. If create new google spreadsheet (e.g. once per month), update spreadsheet id in lambda function: comment-issue
2. If add a new github user, add the user's username in the first row of the spreadsheet