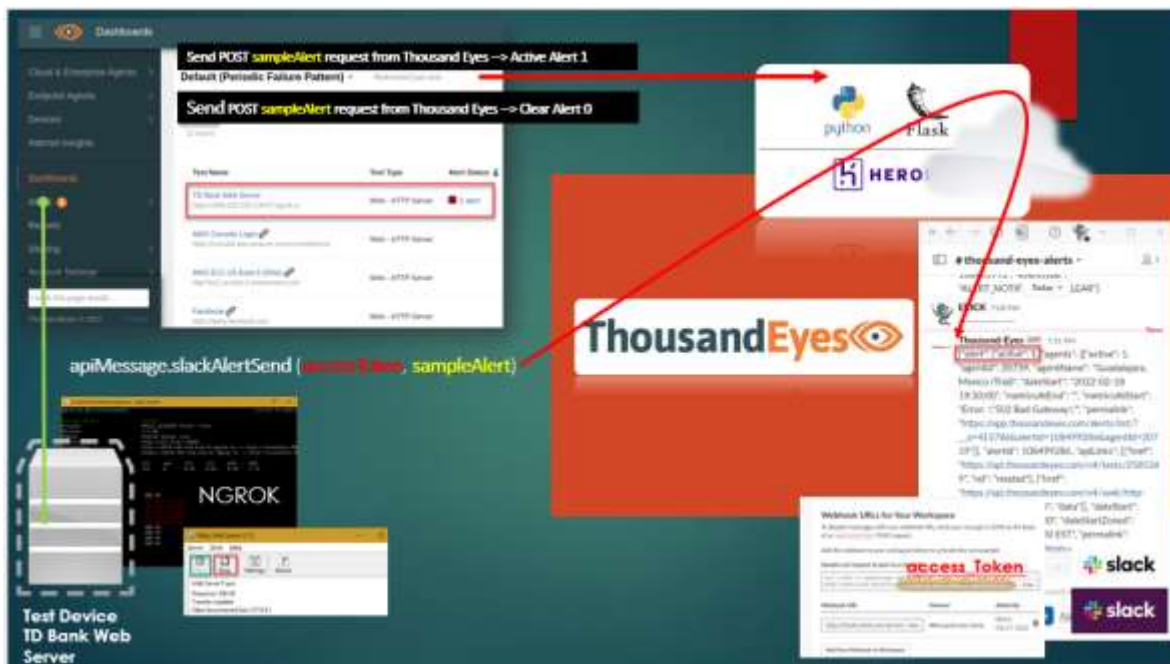


ThousandEyes

OVERVIEW

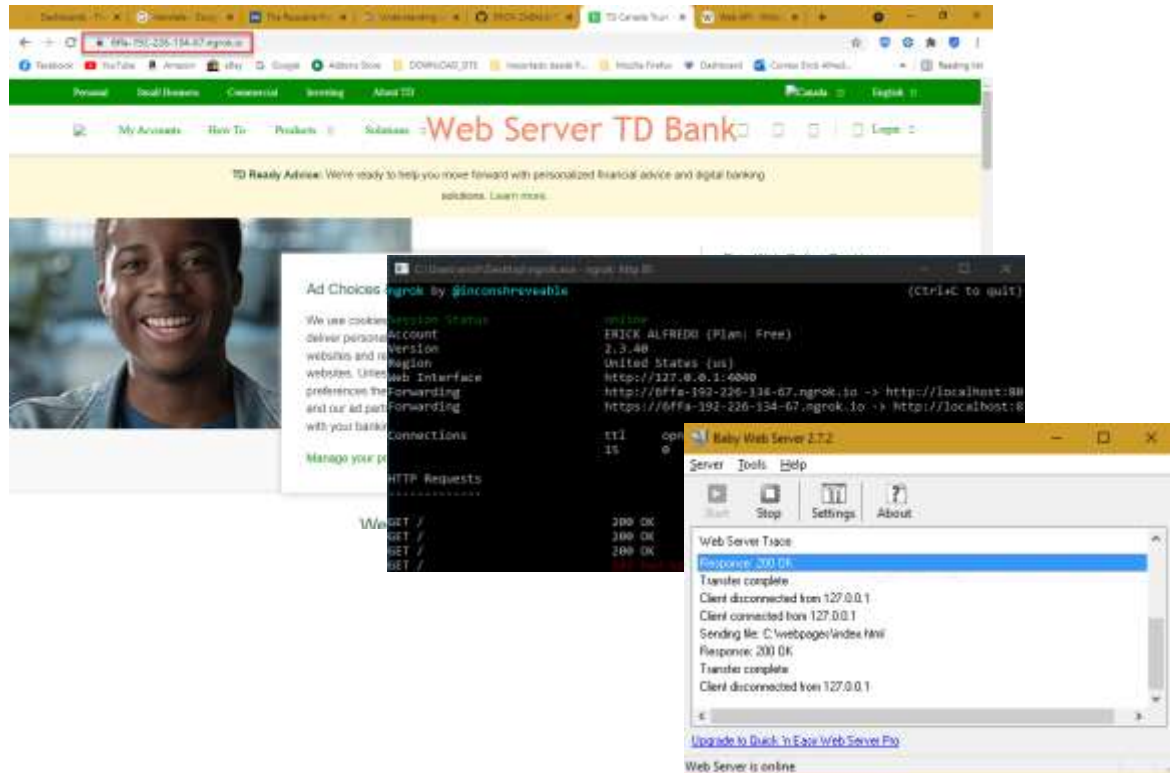
This example is an integration between Thousand Eyes and Slack using Python. Thousand Eyes sends alert notifications via our webhook service and those notifications are received in the Web Server API deployed as an application in the cloud of Heroku. The Web Server API is hosted in the Heroku Cloud send the notifications to Slack device using an incoming webhook of Slack that is a simple way to post messages from external sources (third parties).



PREREQUISITES

- Create a trial account at <https://www.thousandeyes.com/signup/>. Write "Cisco" in the company field.
- Configure a Test (Web Server TD Bank) with an alert that will trigger and clear alerts for your testing in Thousand Eyes.
- Create a Slack instance <https://slack.com/get-started#/createnew>.
- Create a Heroku account at <https://id.heroku.com/login>.

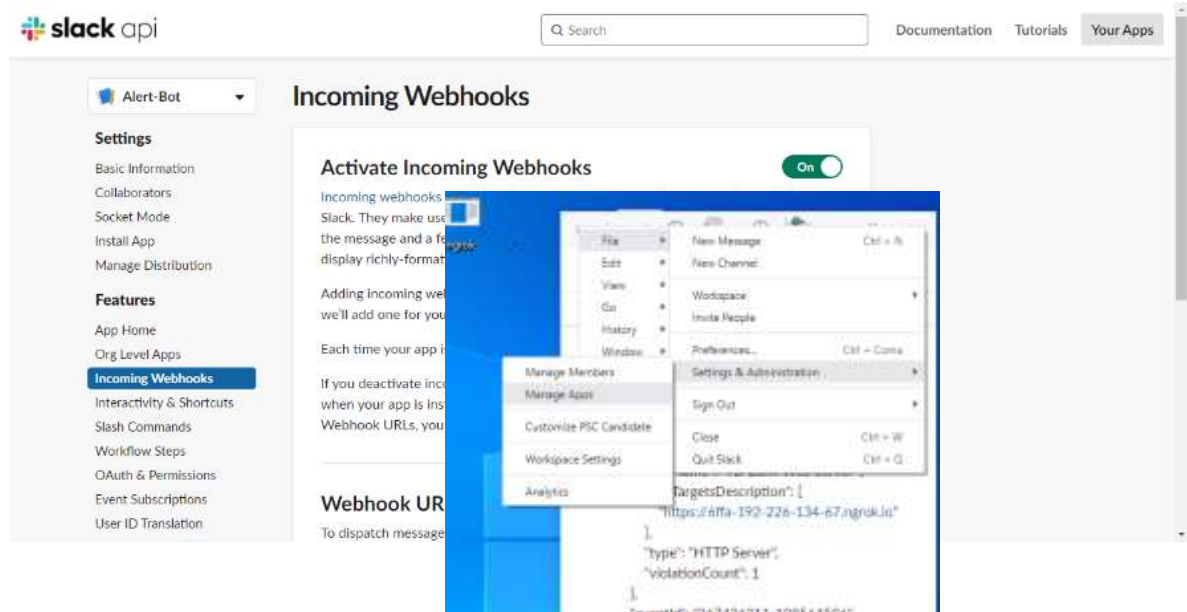
- Deploy a Web Server TD Bank in your PC to make testing with Thousand Eyes, you can use the folder:webServerTdBank and the app:babywebssl.rar and ngrok (<https://ngrok.com/download>) to publish the service to internet.



CONFIGURATION

- Create your virtual environment in Python 3.9.2 and added the file runtime.txt, install all dependencies indicated in requirements.txt as well added Procfile file.
- You can work with the same files:
 - apiMessage.py -----> Submit POST Request to Slack if there is any alert in our Thousand Eyes.

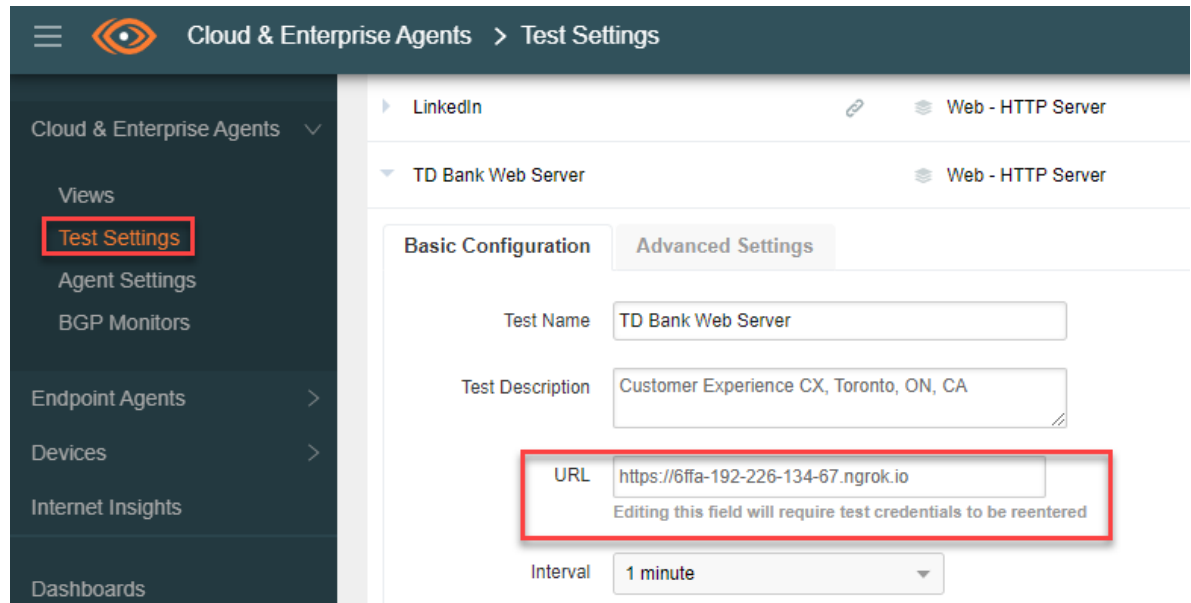
- credentials.py -----> You must to replace with your Slack Token when you activate Incoming Webhook.



- index.py -----> Activate your Web Server API using flask framework to receive POST request from Thousand Eyes Webhook service.
- The "index.py" file allows you to active a Web Server API in Heroku Cloud where you can see the flow request using the line command "**heroku logs --tail**" however you need to install Heroku CLI <https://devcenter.heroku.com/articles/heroku-cli>.

RUNNING

- Activate your Web Server TD Bank with your Ngrok and Babyweb then go to ThousandEyes > Cloud & Enterprise Agents > Test Settings and change your URL "<https://6ffa-192-226-134-67.ngrok.io>"



The screenshot shows the 'Test Settings' page for a 'TD Bank Web Server' agent. The left sidebar lists navigation options: Cloud & Enterprise Agents, Views, Test Settings (highlighted), Agent Settings, BGP Monitors, Endpoint Agents, Devices, Internet Insights, and Dashboards. The main content area shows the 'Basic Configuration' tab with fields for Test Name, Test Description, URL, and Interval. The URL field is highlighted with a red box and contains the value 'https://6ffa-192-226-134-67.ngrok.io'. Below the URL field, a message states: 'Editing this field will require test credentials to be reentered'. The Interval is set to '1 minute'.

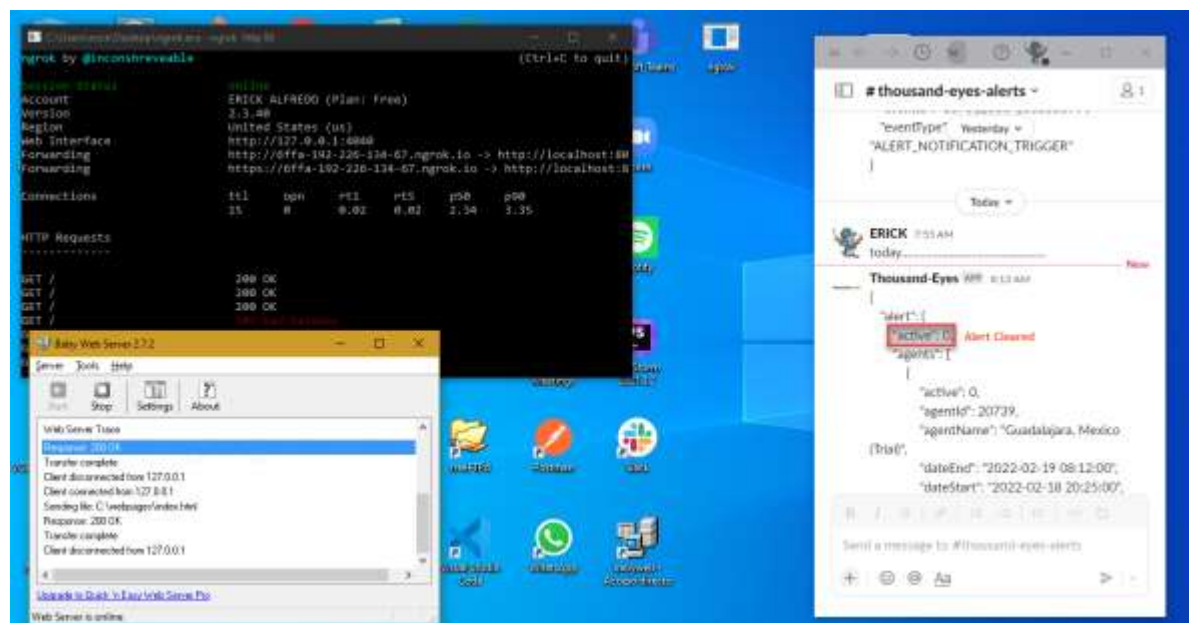
- You will receive an alert cleared in the channel of Slack "**# thousand-eyes-alerts**"
- **Action:** Alert Cleared

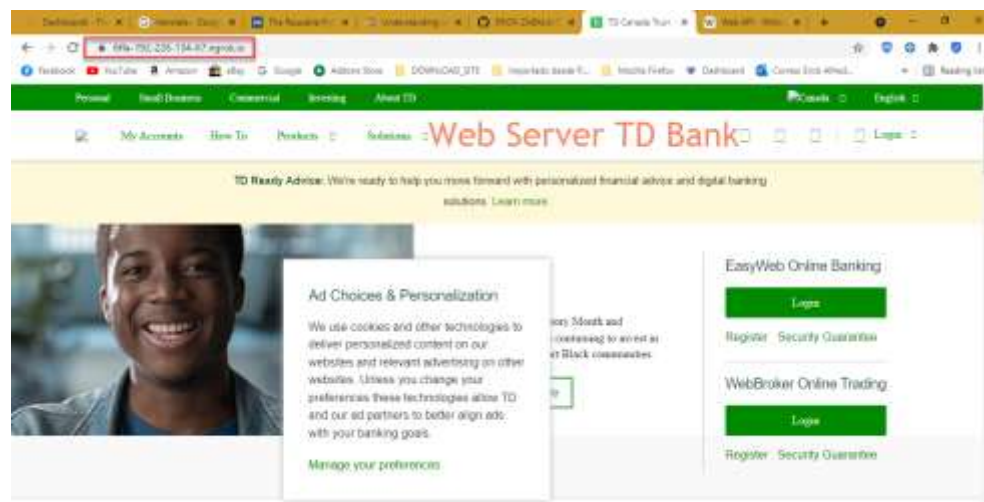
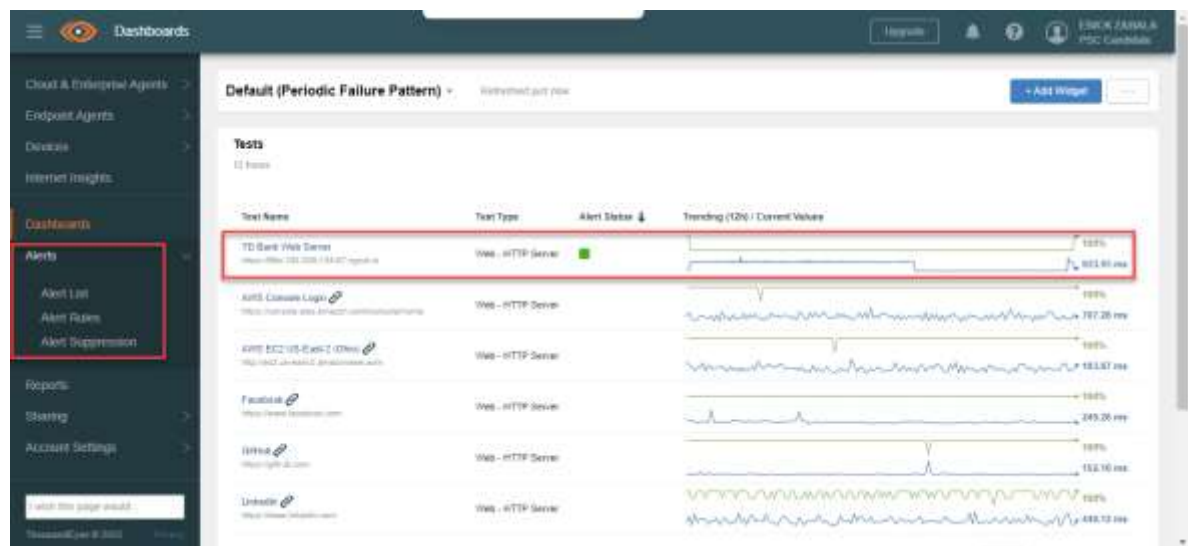
Output:

```
{
  "alert": {
    "active": 0,
    "agents": [
      {
        "active": 0,
        "agentId": 20739,
        "agentName": "Guadalajara, Mexico (Trial)",
        "dateEnd": "2022-02-19 08:12:00",
        "dateStart": "2022-02-18 20:25:00",
        "metricsAtEnd": "Error: \"\\\"\",
        "metricsAtStart": "Error: \"502 Bad Gateway\\\"\",
        "permalink":
        "https://app.thousandeyes.com/alerts/list/?__a=415786&alertId=108505079&agentId=20739"
      }
    ],
    "alertId": 108505079,
    "apiLinks": [
      {
```



```
{
  "href": "https://api.thousandeyes.com/v4/tests/2585349",
  "rel": "related"
},
{
  "href": "https://api.thousandeyes.com/v4/web/http-server/2585349",
  "rel": "data"
}
],
"dateEnd": "2022-02-19 08:12:00",
"dateEndZoned": "2022-02-19 08:12:00 EST",
"dateStart": "2022-02-18 20:25:00",
"dateStartZoned": "2022-02-18 20:25:00 EST",
"permalink": "https://app.thousandeyes.com/alerts/list/?__a=415786&alertId=108505079",
"ruleAid": 415786,
"ruleExpression": "Error is present",
"ruleId": 1927034,
"ruleName": "Webhook Thousand Eyes",
"testId": 2585349,
"testLabels": [],
"testName": "TD Bank Web Server ",
"testTargetsDescription": [
  "https://6ffa-192-226-134-67.ngrok.io"
],
"type": "HTTP Server",
"violationCount": 1
},
"eventId": "367417671-108505079",
"eventType": "ALERT_NOTIFICATION_CLEAR"
}
```





- Turn off your Web Server TD Bank using "Babyweb" application, you will receive an alert active "**active**": 1," in the channel of Slack "**# thousand-eyes-alerts**"



- **Action:** Alert Active

Output:

```
{
  "alert": {
    "active": 1,
    "agents": [
      {
        "active": 1,
        "agentId": 20739,
        "agentName": "Guadalajara, Mexico (Trial)",
        "dateStart": "2022-02-19 08:22:00",
        "metricsAtEnd": "",
        "metricsAtStart": "Error: \"502 Bad Gateway\"",
        "permalink":
"https://app.thousandeyes.com/alerts/list/?__a=415786&alertId=108562975&agentId=20739"
      }
    ],
    "alertId": 108562975,
    "apiLinks": [
      {
        "href": "https://api.thousandeyes.com/v4/tests/2585349",
        "rel": "related"
      },
      {
        "href": "https://api.thousandeyes.com/v4/web/http-server/2585349",
        "rel": "data"
      }
    ],
    "dateStart": "2022-02-19 08:22:00",
    "dateStartZoned": "2022-02-19 08:22:00 EST",
    "permalink": "https://app.thousandeyes.com/alerts/list/?__a=415786&alertId=108562975",
    "ruleAid": 415786,
    "ruleExpression": "Error is present",
    "ruleId": 1927034,
    "ruleName": "Webhook Thousand Eyes",
    "testId": 2585349,
    "testLabels": [],
    "testName": "TD Bank Web Server ",
    "testTargetsDescription": [
      "https://6ffa-192-226-134-67.ngrok.io"
    ],
    "type": "HTTP Server",
    "violationCount": 1
  },
  "eventId": "367422871-108562975",
  "eventType": "ALERT_NOTIFICATION_TRIGGER"
}
```

Send a message to thousand-eyes-alerts

The screenshot shows a Windows desktop with several open applications:

- ngrok by @inconshreveable:** A terminal window showing the ngrok service is running. It displays account information for ERICK ALFREDO (Plan: Free) and lists connections, HTTP requests, and a log of GET requests to the local web server.
- Web Server:** A small application window titled "Web Server" showing a log of activity, including client connections and responses.
- Telegram:** A chat window for the group "#thousand-eyes-alerts" showing a recent alert from ERICK ALFREDO at 9:21 AM. The alert details include agent information, a date start, and a permalink to the alert.

The screenshot displays the ThousandEyes dashboard with the "Alerts" section selected in the sidebar. The main area shows a table of tests with their names, types, alert status, and performance metrics.

Test Name	Test Type	Alert Status	Trending (12h) / Current Value
TD Bank Web Server	Web - HTTP Server	1 alert	100% / 200.00 ms
AMS Console Login	Web - HTTP Server		100% / 125.75 ms
AMS EC2 US-East-1-000	Web - HTTP Server		100% / 200.71 ms
Facebook	Web - HTTP Server		100% / 271.36 ms
Netflix	Web - HTTP Server		100% / 175.12 ms
Unilever	Web - HTTP Server		100% / 417.36 ms

The screenshot shows a web browser displaying an error message:

Failed to complete tunnel connection

The connection to <https://8ffa-192-228-134-67.ngrok.io> was successfully tunneled to your ngrok client, but the client failed to establish a connection to the local address localhost:80.

Make sure that a web service is running on localhost:80 and that it is a valid address.

The error encountered was: `dial tcp [::1]:80: connect: No connection could be made because the target machine actively refused it.`

SEARCHING WITH THIS CODE

The intention to make this code is to generate curiosity, with them coming new ideas and as consequence arrive the creativity, finally you are in the innovation. Reach out via Cisco Webex for any support questions.

REPOSITORY

That is the link reference of repository:

ERICK-ZABALA/Thousand-Eyes Private

<https://github.com/ERICK-ZABALA/Thousand-Eyes.git>
`git@github.com:ERICK-ZABALA/Thousand-Eyes.git`
`gh repo clone ERICK-ZABALA/Thousand-Eyes`

