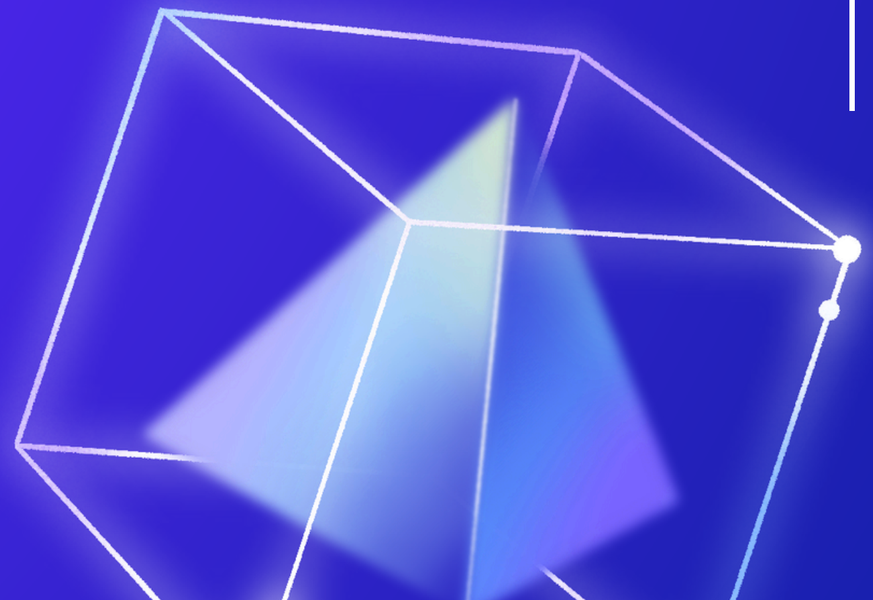
CREATING HONEY TOKENS USING AWS AND TERRAFORM

By Niccolo Arboleda

Reference:
https://github.com/GitGuardian/ggcanary

AGENDA



green apple

deception tool

DECEPTION TECHNOLOGY

“Deception technology is tooling that attracts cybercriminals away from an enterprise's assets and allows an organization to get early warnings of potential cyberattacks in their environment.

HOW IT WORKS

```
"access_key_id": "AKIAU64JG54KI2QPF4LD",
"access_key_secret": "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",
```

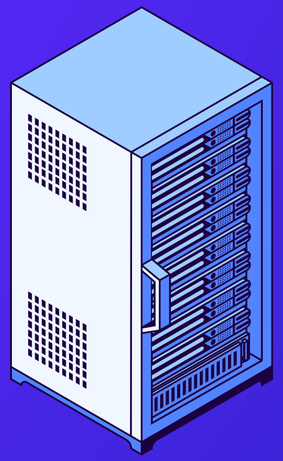
canary/honey token in the form of an AWS key

There’s only one thing more mouth-watering than a green apple for cybercriminals: credentials.

```
"access_key_id": "AKIAU64JG54KI2QPF4LD",
"access_key_secret": "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",
```

Repositories

Dev Stations



Server

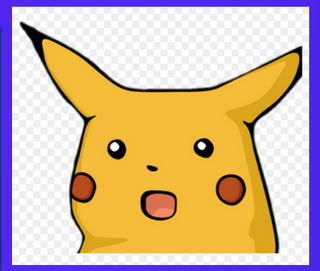
Databases

attacker gets credentials from database

I’m so smart; let me use the credentials I got from the database to log into their account.

attacker uses credentials

AWS



Me in the control room

AWS tells the owner of the token about the attempt.

The database is compromised!

attacker is found out.

PROS

- makes use of psychology.
- helps detect attacks in real-time.
- allows insight into where a cyber attack started.
- relatively cheaper than other deception tools.
- flexible in usage (what it is and its activation).

CONS

- Its use is only for detection, not prevention or response.
- It is not a primary security solution.

TECHNOLOGY USED

- TERRAFORM
- AWS S3 BUCKET
- AWS CLOUDTRAIL
- AWS DYNAMODB
- AWS LAMBDA
- SLACK

Source: https://blog.gitguardian.com/from-vulnerability-to-advantage-turn-exposed-secrets-into-your-best-allies-to-detect-intrusion/

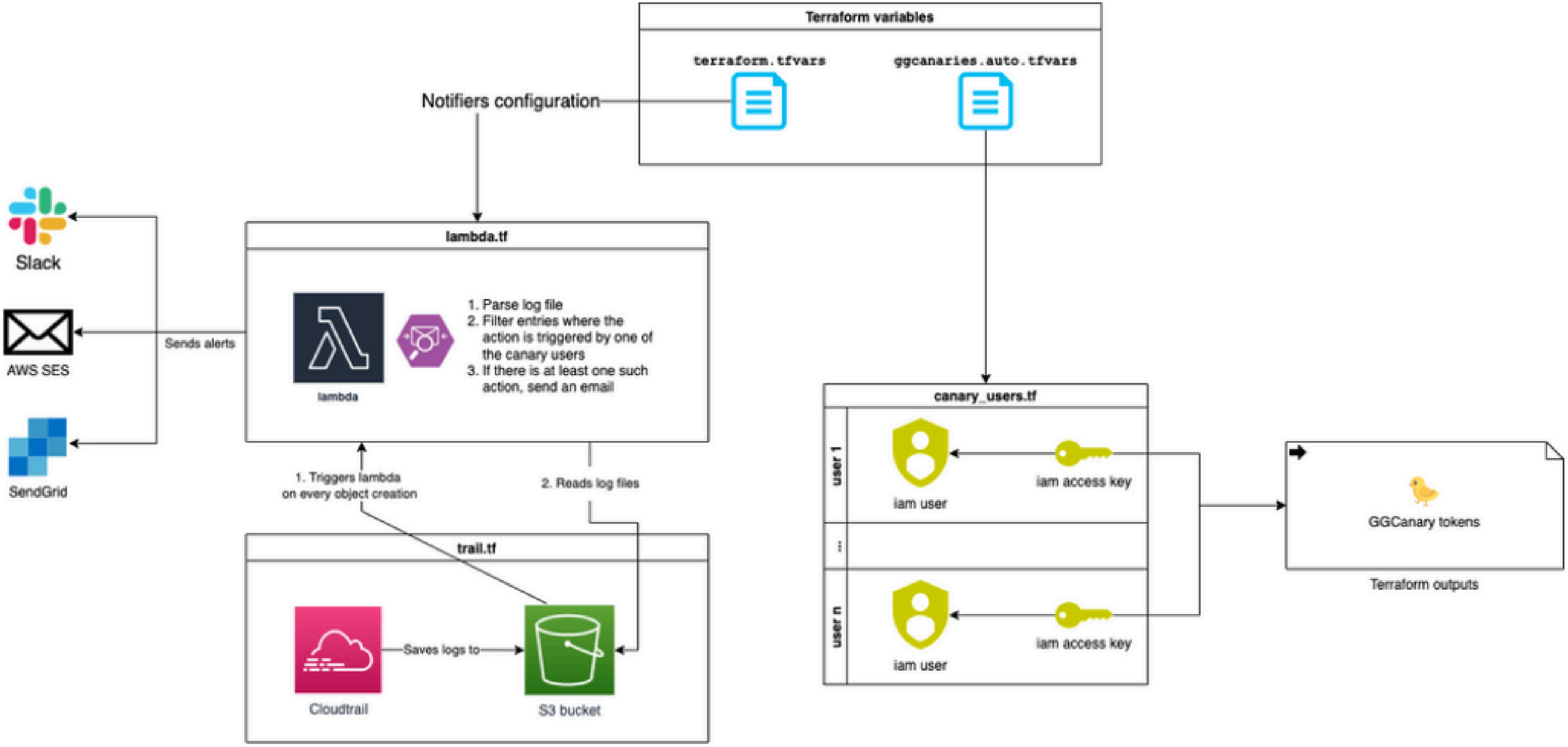
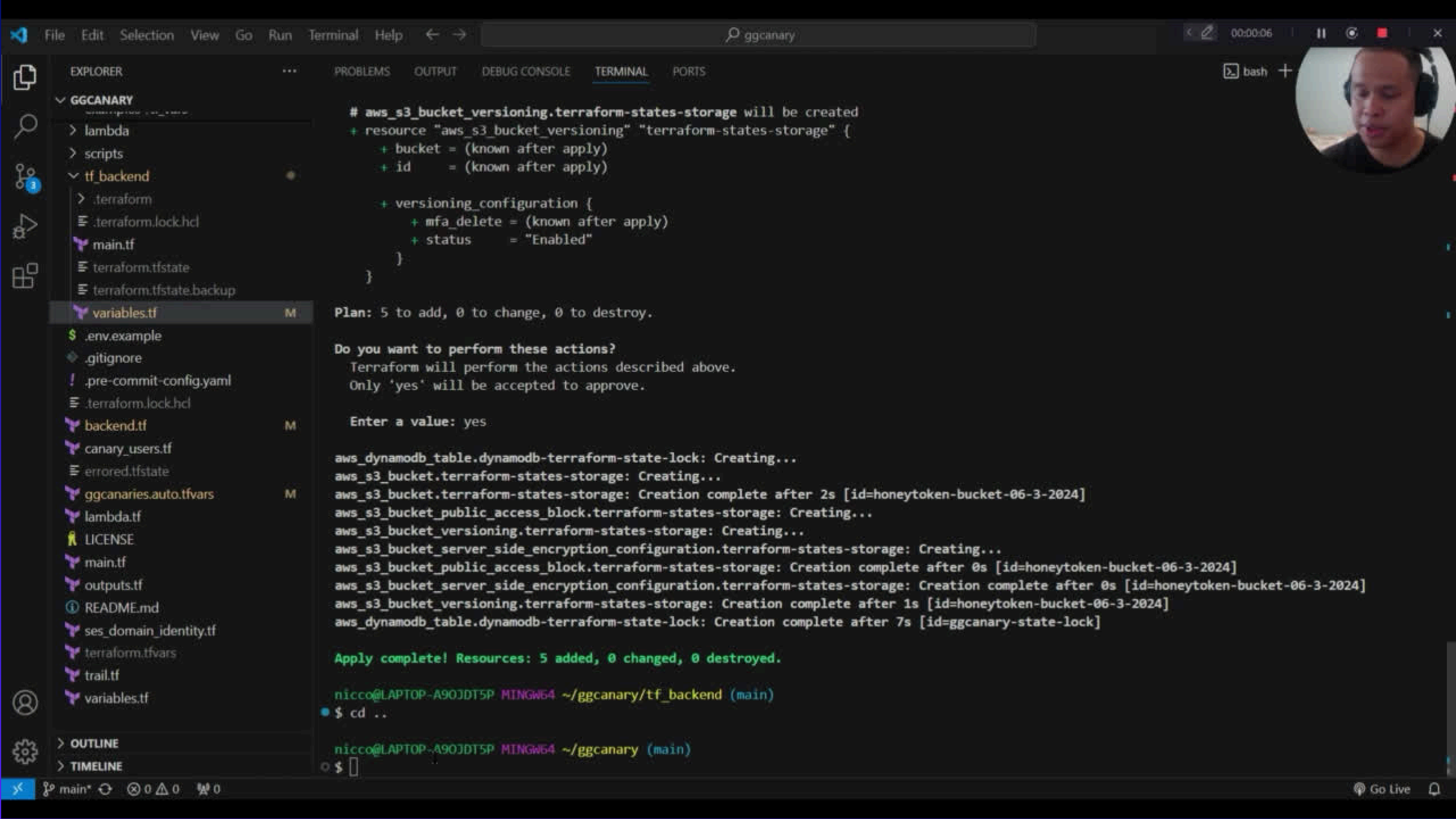
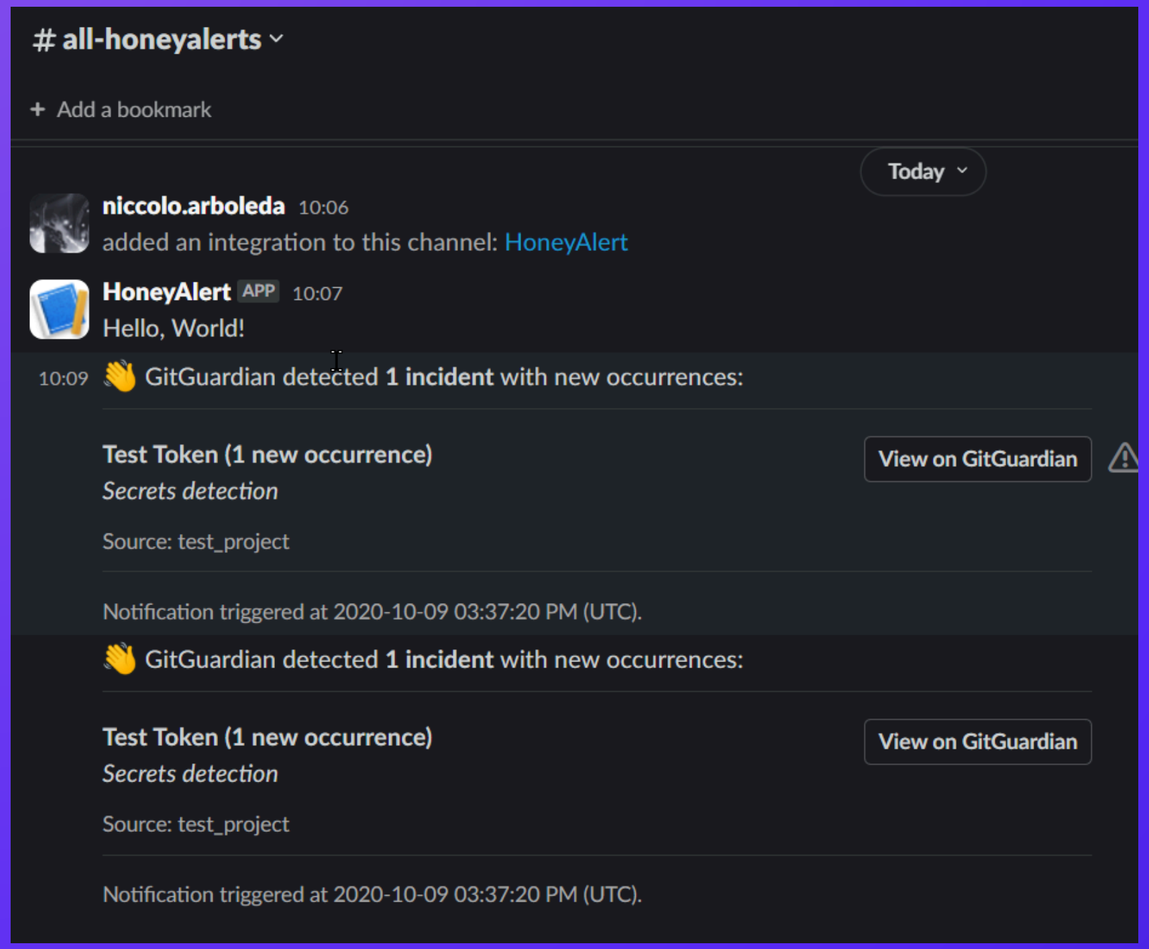
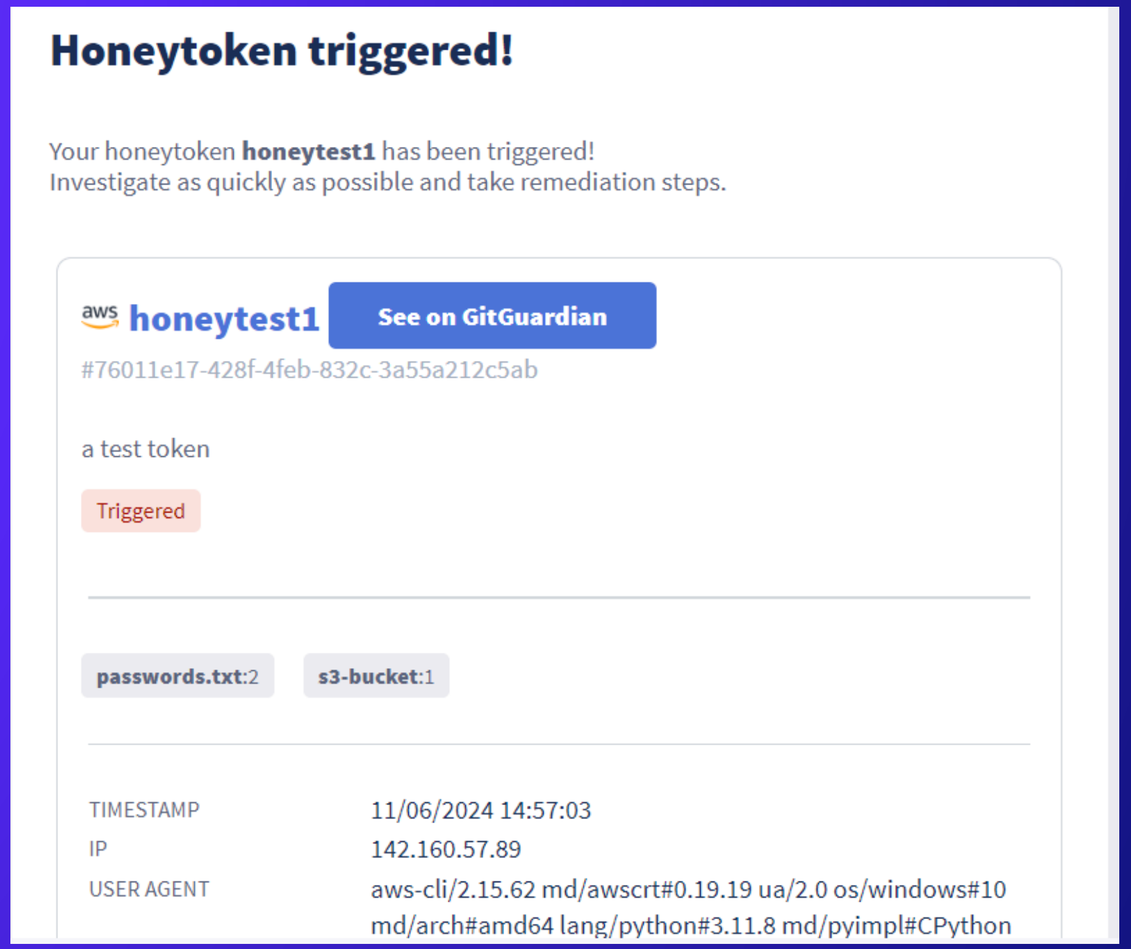


Diagram illustrating the ggcanary architecture

DEMO


File Edit Selection View Go Run Terminal Help
ggcanary
00:00:06
EXPLORER
GGCANARY
lambda
scripts
tf_backend
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
variables.tf M
.env.example
.gitignore
.pre-commit-config.yaml
.terraform.lock.hcl
backend.tf M
canary_users.tf
errored.tfstate
ggcanaries.auto.tfvars M
lambda.tf
LICENSE
main.tf
outputs.tf
README.md
ses_domain_identity.tf
terraform.tfvars
trail.tf
variables.tf
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
bash
# aws_s3_bucket_versioning.terraform-states-storage will be created
+ resource "aws_s3_bucket_versioning" "terraform-states-storage" {
+ bucket = (known after apply)
+ id = (known after apply)
+ versioning_configuration {
+ mfa_delete = (known after apply)
+ status = "Enabled"
}
}
Plan: 5 to add, 0 to change, 0 to destroy.
Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.
Enter a value: yes
aws_dynamodb_table.dynamodb-terraform-state-lock: Creating...
aws_s3_bucket.terraform-states-storage: Creating...
aws_s3_bucket.terraform-states-storage: Creation complete after 2s [id=honeytoken-bucket-06-3-2024]
aws_s3_bucket_public_access_block.terraform-states-storage: Creating...
aws_s3_bucket_versioning.terraform-states-storage: Creating...
aws_s3_bucket_server_side_encryption_configuration.terraform-states-storage: Creating...
aws_s3_bucket_public_access_block.terraform-states-storage: Creation complete after 0s [id=honeytoken-bucket-06-3-2024]
aws_s3_bucket_server_side_encryption_configuration.terraform-states-storage: Creation complete after 0s [id=honeytoken-bucket-06-3-2024]
aws_s3_bucket_versioning.terraform-states-storage: Creation complete after 1s [id=honeytoken-bucket-06-3-2024]
aws_dynamodb_table.dynamodb-terraform-state-lock: Creation complete after 7s [id=ggcanary-state-lock]
Apply complete! Resources: 5 added, 0 changed, 0 destroyed.
nicco@LAPTOP-A9OJDT5P MINGW64 ~/ggcanary/tf_backend (main)
$ cd ..
nicco@LAPTOP-A9OJDT5P MINGW64 ~/ggcanary (main)
$
main*
0 0 0
Go Live


all-honeyalerts
+ Add a bookmark
Today
niccolo.arboleda 10:06
added an integration to this channel: HoneyAlert
HoneyAlert APP 10:07
Hello, World!
10:09 GitGuardian detected 1 incident with new occurrences:
Test Token (1 new occurrence)
Secrets detection
Source: test_project
View on GitGuardian
Notification triggered at 2020-10-09 03:37:20 PM (UTC).
GitGuardian detected 1 incident with new occurrences:
Test Token (1 new occurrence)
Secrets detection
Source: test_project
View on GitGuardian
Notification triggered at 2020-10-09 03:37:20 PM (UTC).


Honeytoken triggered!
Your honeytoken honeytest1 has been triggered!
Investigate as quickly as possible and take remediation steps.
aws honeytest1
See on GitGuardian
#76011e17-428f-4feb-832c-3a55a212c5ab
a test token
Triggered
passwords.txt:2
s3-bucket:1
TIMESTAMP 11/06/2024 14:57:03
IP 142.160.57.89
USER AGENT aws-cli/2.15.62 md/awscrt#0.19.19 ua/2.0 os/windows#10 md/arch#amd64 lang/python#3.11.8 md/pyimpl#CPython

KEY TAKEAWAYS

THANK YOU!