Who Goes There? Actively Detecting Intruders With Cyber Deception Tools

@mcdwayne

Hi, I’m Dwayne

- I live in Chicago
- I’ve been a Developer Advocate since 2016
- Co-host of The Security Repo Podcast
- On Twitter @mcdwayne
- mcdwayne@mastodon.social
- Outside of tech, I love improv, karaoke and rock ’n roll

GitGuardian is the code security platform for the DevOps generation.

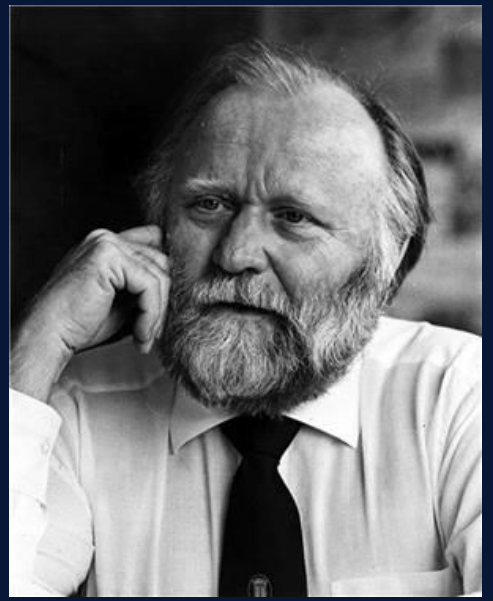
Main focus:
"Where are my hardcoded secrets, have they leaked, and who is working on remediation?"

Dwayne.McDaniel@gitguardian.com

Pardon the interruption…
I need to deploy something real quick

Attackers Want Your Credentials

“I must not fear. Fear is the mind-killer. Fear is the little-death that brings total obliteration. I will face my fear. I will permit it to pass over me and through me. And when it has gone past I will turn the inner eye to see its path. Where the fear has gone there will be nothing. Only I will remain.”



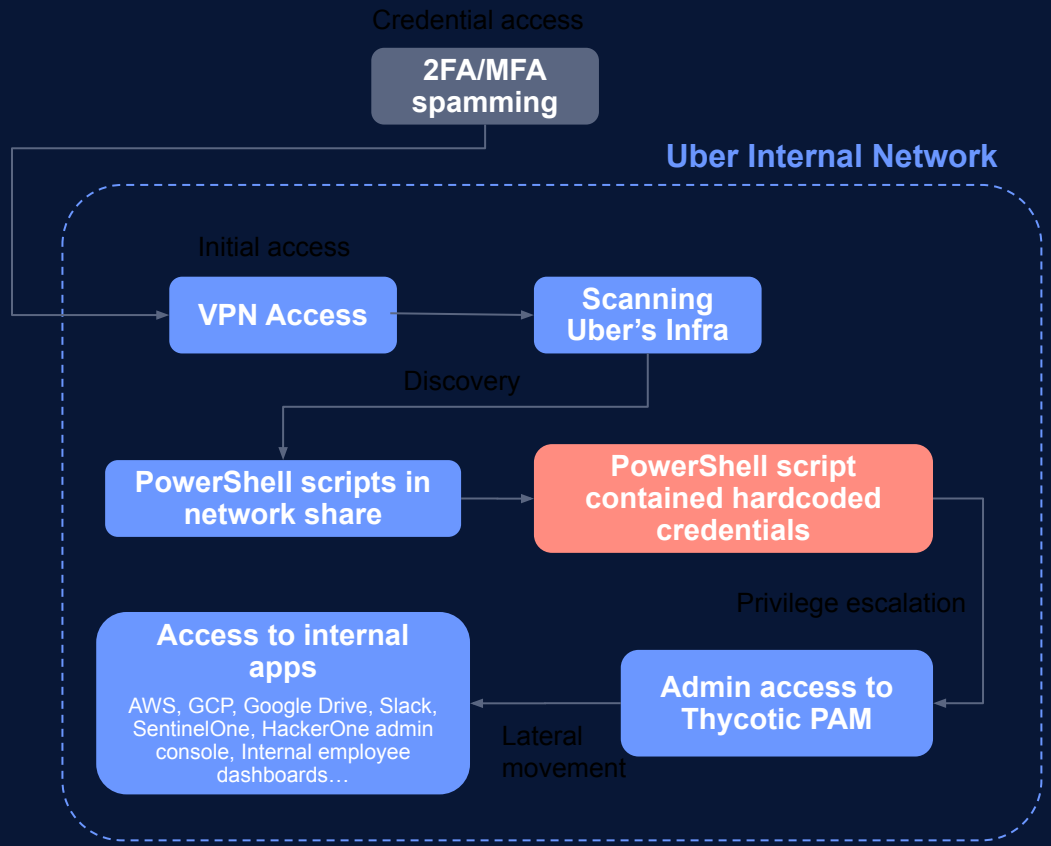
— Frank Herbert, Dune
Bene Gesserit Litany Against Fear

Uber Breach – September 2022

Uber

One exposed secret leads to many others!

Some mistakes are more expensive than others. One hardcoded secret giving access to Uber's PAM solution led to a an organization wide IT takeover…

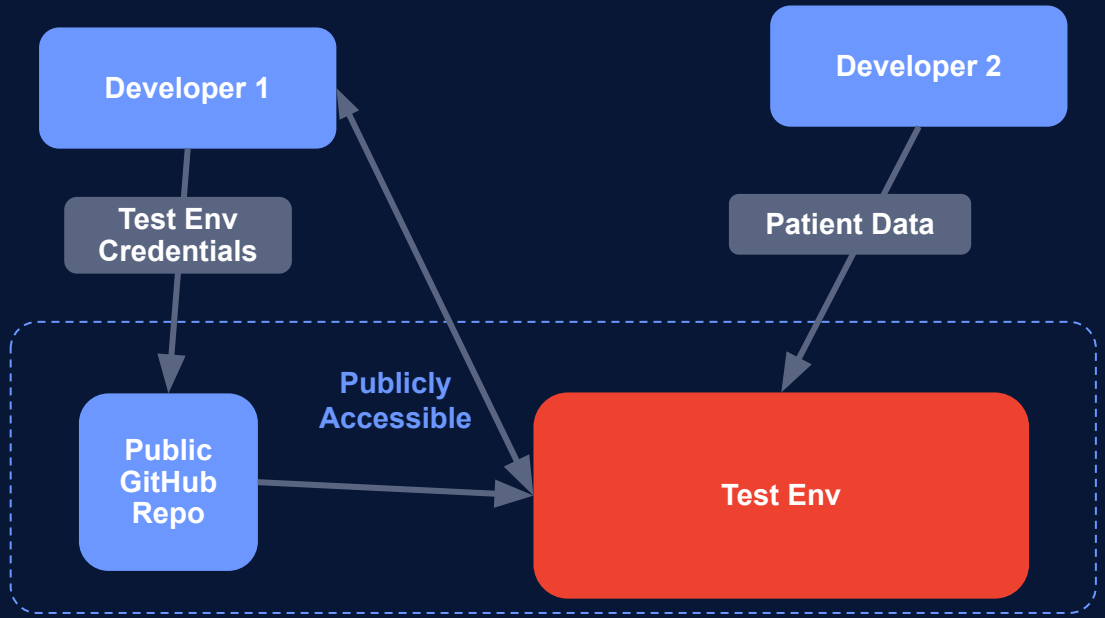


AstraZeneca – November 2022

Unknown number of affected end users

AstraZeneca

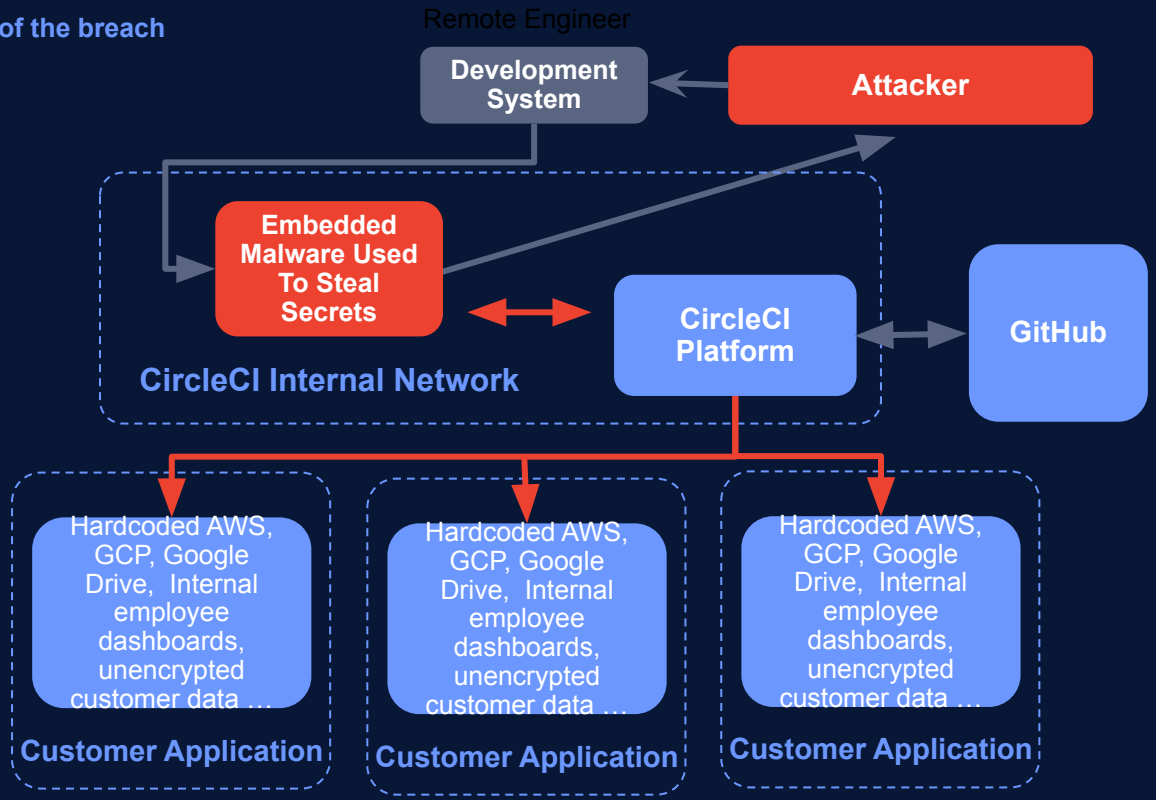
Test environment credentials pushed to GitHub. Another user pushed real patient data to the test environment. Credentials were exposed for over a year before reported. Number of patients affected was never disclosed.



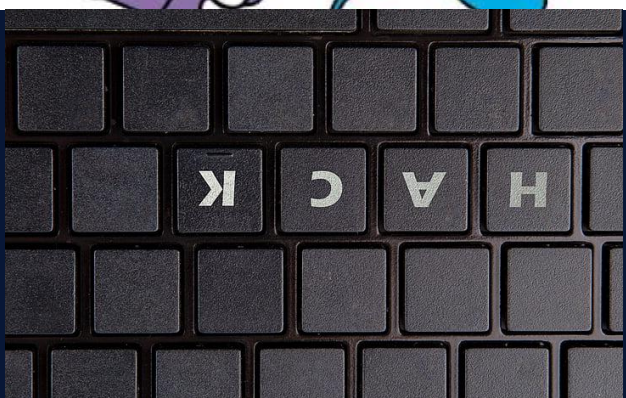
CircleCI – January 2023

Use of a honeytoken alerted the security team of the breach

Customer secrets stolen, unauthorised access to GitHub repos and third-party systems




Data
information
computer
browse
internet
analyze
knowledge
big data
data storage
intellect


HACK

We Know How Attackers Behave

What Attackers Want

1. Access To Data
2. Machine Resources
3. Anything That Leads To 1 or 2

In the 2023 edition of The State of Secrets Sprawl

10M secrets found exposed
in 2022 in public GitHub repositories

More than 67% increase compared to 6 Million in 2021

On average, 5.5 commits out of 1,000 exposed at least one secret **+50% compared to 2021**

---

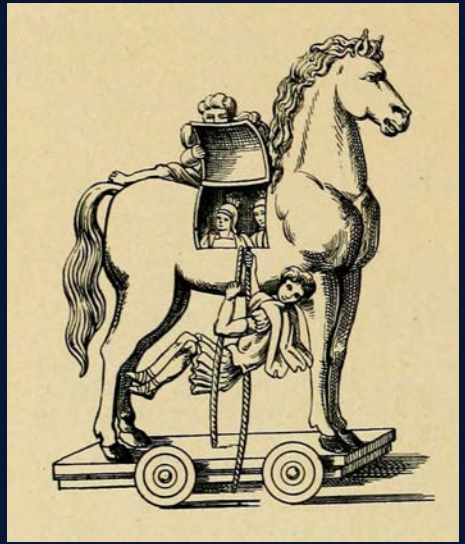
https://www.gitguardian.com/state-of-secrets-sprawl-report-2023

A Brief History Of Cyber Deception

Brief History of Deception

Trojan Horse
~ 1200 BCE

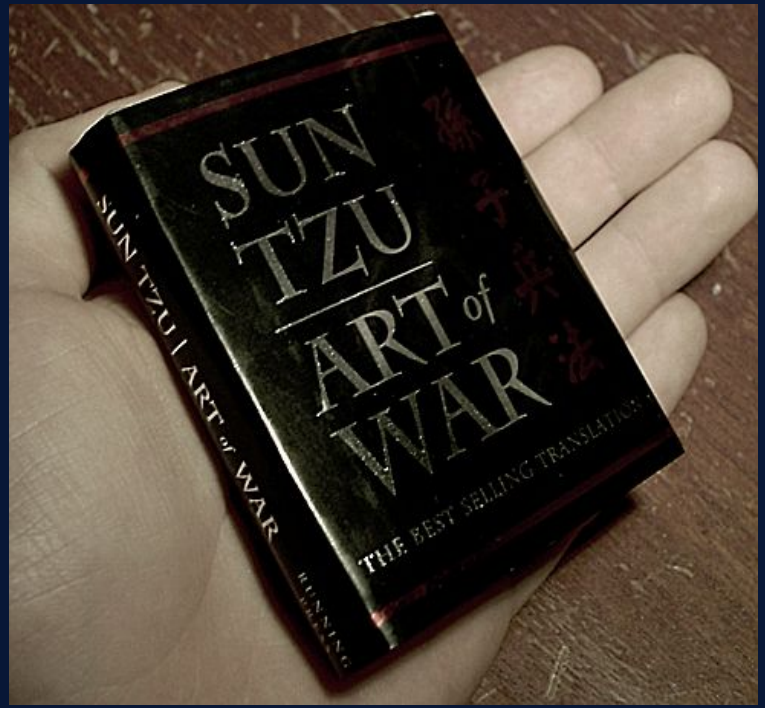
"*I thought I was getting a gift horse from the Achaeans , what I got was defeat*"
- Trojan security officer



Brief History of Deception

Art of War
~ 400 BCE

"*Appear weak when you are strong, and strong when you are weak*"
- Sun Tzu



Brief History of Deception

Ghost Army
1942

"*The first mobile, multimedia, tactical deception unit in U.S. Army history*"

- James Linn - Curator National WWII Museum

Brief History of Deception

The First Honeypot
1985

"Hi, is this the FBI? At my girlfriend's suggestion, I used fake documents to trick someone working with the KGB into keeping their connection to a Lawrence Berkeley National Laboratory computer open long enough to trace their exact location."

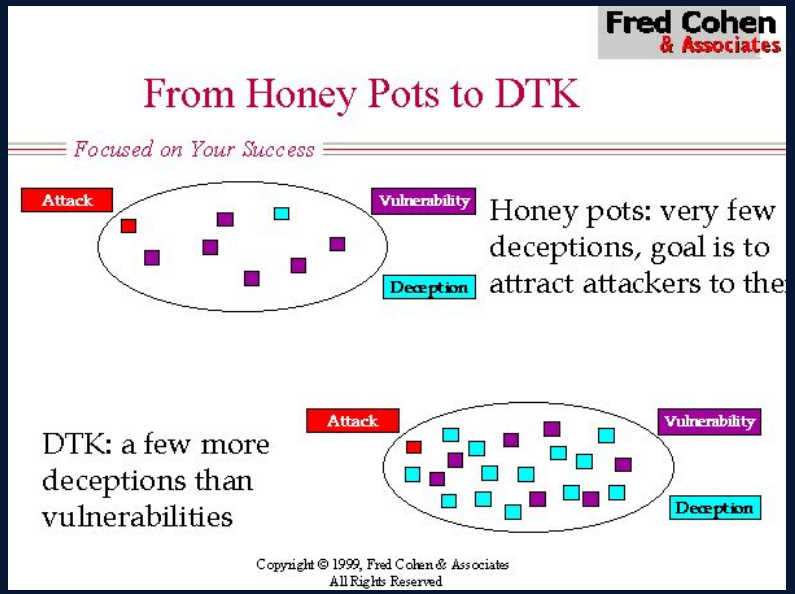
- Cliff Stoll

Brief History of Deception

Fred Cohen's Deception Toolkit 1991

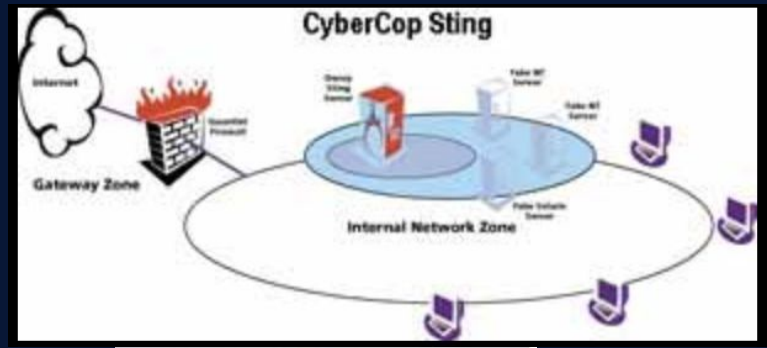
"*Under DTK, deceptions are spread among the normal systems in a network in such a way that unused services on those systems are consumed with deceptions. This increases the likelihood of an intelligence probe encountering a deception rather than a vulnerability*"

- Fred Cohen



Brief History of Deception

First Commercial Honeypots 1998



"These hackers aren't kids on a digital joyride, ... It's clear their motive is financial gain."

- Alfred Huger, Creator of CyberCop Sting

Brief History of Deception

"Honeytokens" is coined 2003

RES: Protocol Anomaly Detection IDS - Honeypots

From: "Augusto Paes de Barros" <augusto () paesdebarros com br>
Date: Fri, 21 Feb 2003 11:17:46 -0000

Lance's point can be expanded in very interesting views. Why use only honeypots "hosts" or "nets", when whe can use accounts, documents, info, etc? I was developing an idea that I call "honeytokens", to use on Windows networks. Basically, information that shouldn't be flowing over the network and, if you can detect it, something wrong is happening.

--
Augusto Paes de Barros, CISSP
http://www.paesdebarros.com.br
augusto () paesdebarros com br

"I was developing an idea that I call 'honeytokens'... *Basically, information that shouldn't be flowing over the network and, if you can detect it, something wrong is happening.*"

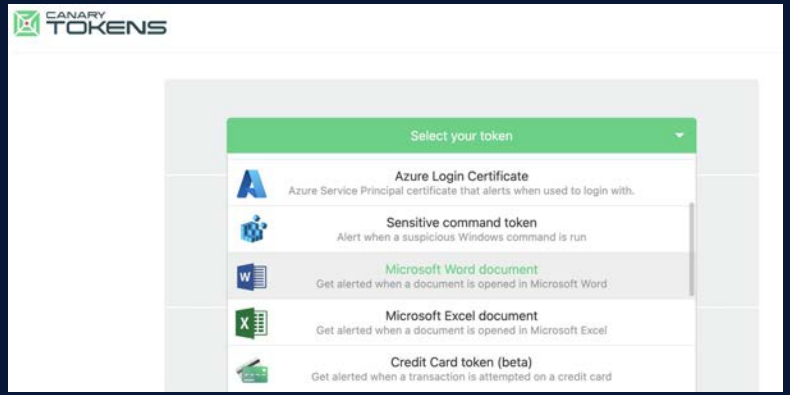
- Augusto Paes de Barros

Brief History of Deception

Canarytokens
2015

"*Added aws token*
Added svn + smtp tokens to generate"

- nickrohrbs, Thinkst developer in a git commit message upon adding aws tokens to the code in 2016.



Brief History of Deception

Honeytokens Becomes Default 2nd Line Defence at Google 2023

"*Honeytokens are your early warning signs*"

- Kevin Mandia from Mandiant /Google Cloud - The state of Cybersecurity - Year in Review talk at RSA 2023

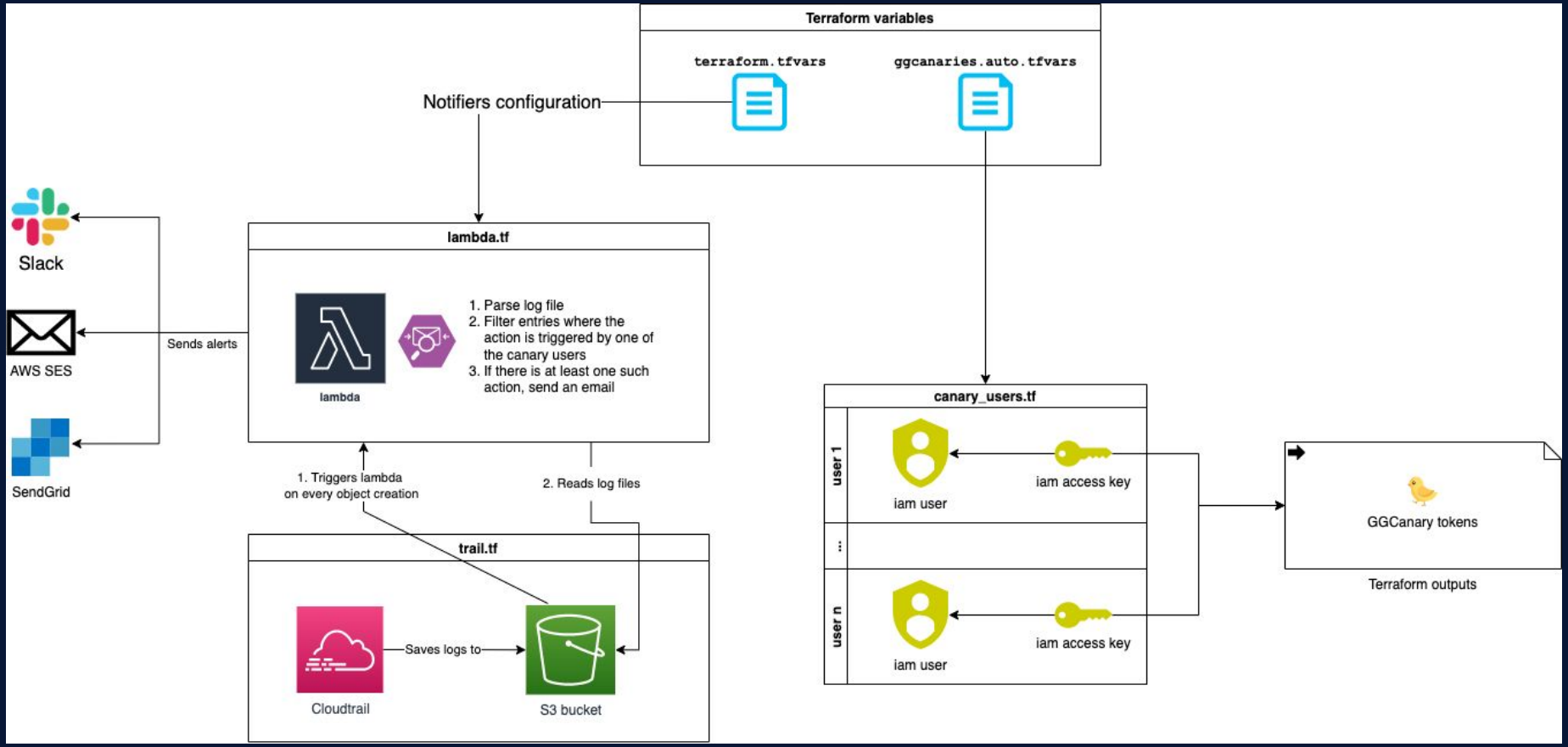
What Is A Honeytoken?

```
provider "aws" {
  region = "us-east-2"
  access_key = "AKIAZ63VNLHLGVPIZC72"
  secret_key = "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"
}
```

Definition

Honeytokens are decoy credentials that do not allow any access to any resources or data. Instead, they trigger alerts that reveal the IP address of the user who attempted to use the honeytoken.

Honeytokens look identical to real credentials to an attacker.



https://github.com/GitGuardian/ggcanary

Honeytoken Options

vs

Open Source - The DIY Route

- Complete DIY - see previous diagram - requires Lambda knowledge and time to tinker with it
- GitGuardian/ggcanary - Requires Terraform and AWS
- spacesiren/spacesiren - Requires AWS how
- thinkst/canarytokens - Requires Docker experience

Commercial Options - Off The Shelf

- Canarytokens.org - free, one off honeytokens
- Thinkst - Canary.tools - The paid version of Canarytoken.org
- GitGuardian Honeytoken Module - Requires GitGuardian account
- Microsoft Sentinel - Azure specific
- CrowdStrike - Requires CrowdStrike
- Proofpoint - Identity Threat Defense Shadow

Honeytoken Best Practices

Honeytoken Best Practices

Do:

Put honeytokens in your private environments

- Since they don't go to anything, there is no legit reason someone would attempt to use one
  - Code, CI environments, Jira, Slack, Vault

Honeytoken Best Practices

DO NOT:

Put honeytokens in public places

- AWS, GitHub, GitLab, GitGuardian and many other public scanners are always on the lookout for public keys and will trigger them by scanning them

Honeytoken Best Practices

Do:

Use a 1:1 ratio of honeytokens to repo/environment

- **Keep it simple. When an alarm goes off, make it easy to tell exactly where, and only where, that honeytoken was embedded.**

Honeytoken Best Practices

Do:

Use automation to scale deployment

- One off honeytokens have value, but blue teams should be worried about defense at scale
  - Bash or Python scripting should be all you need
  - An example

    https://github.com/mcdwayne/honeytoken-putter

Honeytoken Best Practices

Do:

Think in terms of 'Blue Team'

- Use the IP, UserAgent, and other data points to block access
- The goal is not to track down the individual attacker, it is to guard your stuff
- If you think other credentials are at risk, time to rotate them

Honeytoken Best Practices

Do:

Remember this is a journey, not a one off exercise

- **Start with one repo. Worry about scale and automation once you understand and are comfortable with this, or any tech.**

Let’s check on our honeytoken from earlier…

In Conclusion


Data
information
computer
browse
internet
analyze
knowledge
intellect
big data
data storage


H A C K

```
provider "aws" {
  region = "us-east-2"
  access_key = "AKIAZ63VNLHLGVPIZC72"
  secret_key = "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"
}
```

Definition

Honeytokens are decoy credentials that do not allow any access to any resources or data. Instead they trigger alerts that reveal the IP address of the user who attempted to use the honeytoken.

Honeytokens look identical to real credentials to an attacker.

Honeytoken Options

vs

Honeytoken Best Practices

DO NOT:

Put honeytokens in public places

- AWS, GitHub, GitLab, GitGuardian and many other public scanners are always on the lookout for public keys and will trigger them by scanning them

Hi, I’m Dwayne

Dwayne McDaniel

- I live in Chicago
- I’ve been a Developer Advocate since 2016
- Co-host of The Security Repo Podcast
- On Twitter @mcdwayne
- mcdwayne@mastodon.social
- LinkedIn @dwaynemcdaniel
- Happy to chat about anything, hit me up
- Outside of tech, I love improv, karaoke and going to rock and roll shows!

Who Goes There? Actively Detecting Intruders With Cyber Deception Tools

@mcdwayne

Toyota – October 2022

Found by a third party security researcher

TOYOTA

Private T-Connect repo, that included a key to production data server, cloned and pushed to a public GitHub repo. This exposed data of 296,000 customers for 5 years.