https://tinyurl.com/dwayne-corn

# Secrets Security End-To-End


@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

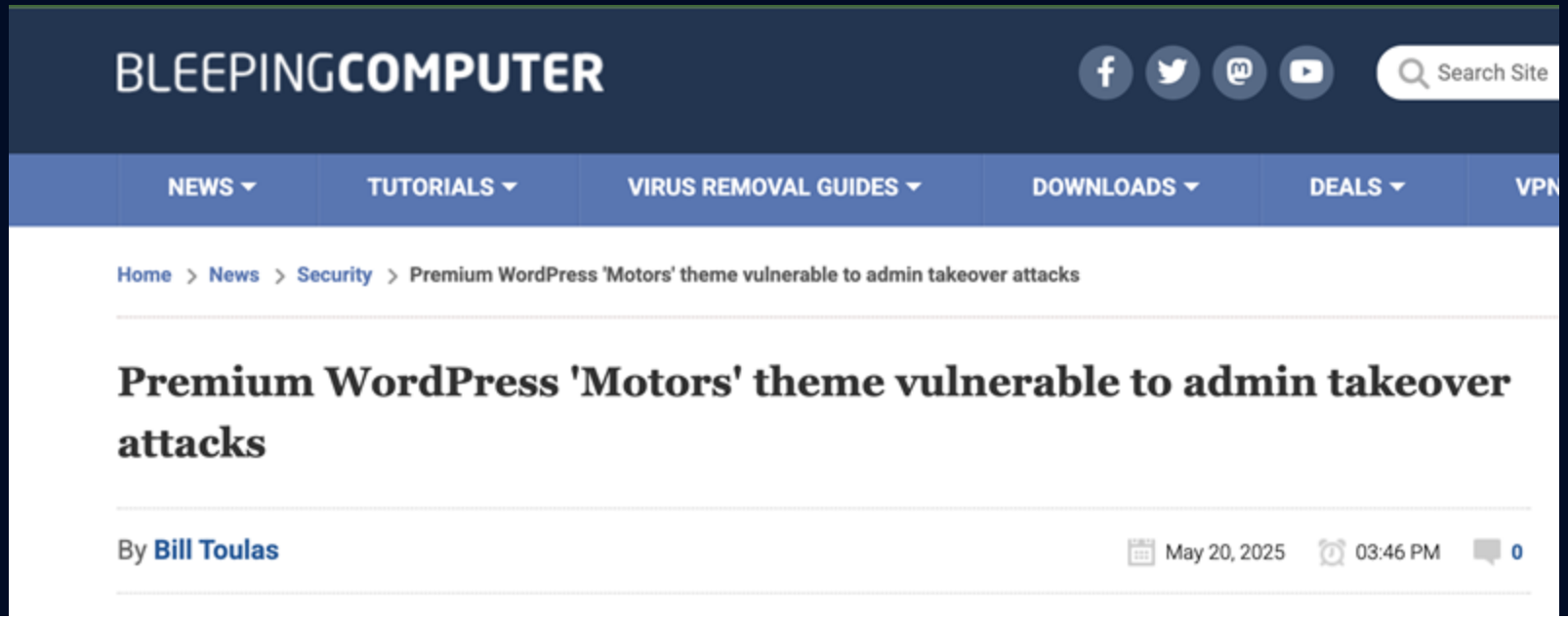
Forbes

# Data Reveals Identity-Based Attacks Now Dominate Cybercrime

One of the most critical shifts is the move toward continuous identity verification. Traditionally, authentication has been treated as a one-time event—users log in once and are then trusted indefinitely. But with attackers now impersonating legitimate users, more companies are adopting real-time behavioral monitoring to detect anomalies.

Another major change is the adoption of just-in-time privileges. Instead of giving employees permanent administrative access, organizations are limiting high-risk permissions to the exact moment they're needed—then revoking them immediately afterward.


@mdwayne-real.bsky.social




"This (vulnerability) is due to the theme not properly validating a user's identity prior to updating their password," explains Wordfence.

"This makes it possible for unauthenticated attackers to change arbitrary user passwords, including those of administrators, and leverage that to gain access to their account."


@mdwayne-real.bsky.social

The New York Times

## How did the leak happen?

The New York Times commented that *"The underlying event related to last week's posting occurred in January 2024 when a credential to a cloud-based third-party code platform was inadvertently made available".*

# US Treasury says Chinese hackers stole documents in 'major incident'

## Events Timeline

- **December 2, 2024**: BeyondTrust detected anomalous behavior and started investigating potential unauthorized access.
- **December 5, 2024**: BeyondTrust identified that an API key for their Remote Support SaaS had been compromised. The company revoked the compromised key, notified impacted customers, and suspended the affected instances.


@mdwayne-real.bsky.social
@mdwayne-real.bsky.social



@mdwayne-real.bsky.social


@mdwayne-real.bsky.social

# Hi. I’m Dwayne.

Dwayne McDaniel
Senior Developer Advocate
dwayne.mcdaniel@gitguardian.com

{

“Hometown” : “Chicago”,

- “Mission” : “Help people figure stuff out”,
- “Developer-advocate-since” : “2014”,
- “Host” : “The Security Repo Podcast”,
- “Socials” : {

“mcdwayne@mastodon.social”,

“www.linkedin.com/in/dwaynemcdaniel” },

“Other-interests” : {“crochet”, “karaoke”,

“rock and roll concerts”, “music in general”}

}

@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

# About GitGuardian

**GitGuardian is an enterprise platform helping teams solve Non-Human Identity security crisis**

- Secrets Detection and Remediation Platform
- Developer Tooling for Prevention
- Honeytokens
- Public Monitoring of GitHub
- NHI Governance

# What Attackers Want:

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


@mdwayne-real.bsky.social

VS

Human

Non-human


@mdwayne-real.bsky.social


Production software environments are composed of a large number of applications which need to be identified, resulting in “non-human identities” or NHI.

Application identities are often associated with secrets, which are used as credentials similarly to the way humans authenticate into computer systems. Application secrets may be used to authenticate into other applications within the trust domain. They may also be used to authenticate into 3rd party SaaS applications.

@mdwayne-real.bsky.social
@mdwayne-real.bsky.social


# Workload Identity in a Multi System Environment (WIMSE) Architecture

- Workload

A workload is a running instance of software executing for a specific purpose. Workload typically interacts with other parts of a larger system. A workload may exist for a very short durations of time (fraction of a second) and run for a specific purpose such as to provide a response to an API request. Other kinds of workloads may execute for a very long duration, such as months or years. Examples include database services and machine learning training jobs.

https://datatracker.ietf.org/doc/draft-ietf-wimse-arch/

@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

FORBES > INNOVATION

# We're Going Through A Machine Identity Crisis

**Ted Shorter** Forbes Councils Member
**Forbes Technology Council**
COUNCIL POST | Membership (Fee-Based)

Oct 18, 2022, 07:00am EDT

The recent growth of machine identities can also create weaknesses. A report from CyberArk found that machine identities outnumber humans 45-1 and that 68% of non-human identities have access to sensitive data and assets.


@mdwayne-real.bsky.social


@mdwayne-real.bsky.social

@n -real.bsky.social
@m e-real.bsky.social

# What are secrets in software development?

```python
from typing import Dict
import aws_lib

def aws_upload(data: Dict):
    database = aws_lib.connect("AKIAF6BAFJKR45SAWSZ5",
    "hjshnk5ex5u34565AWS654/JKGjhz545d89sjkja")
    database.push(data)
```

## Definition

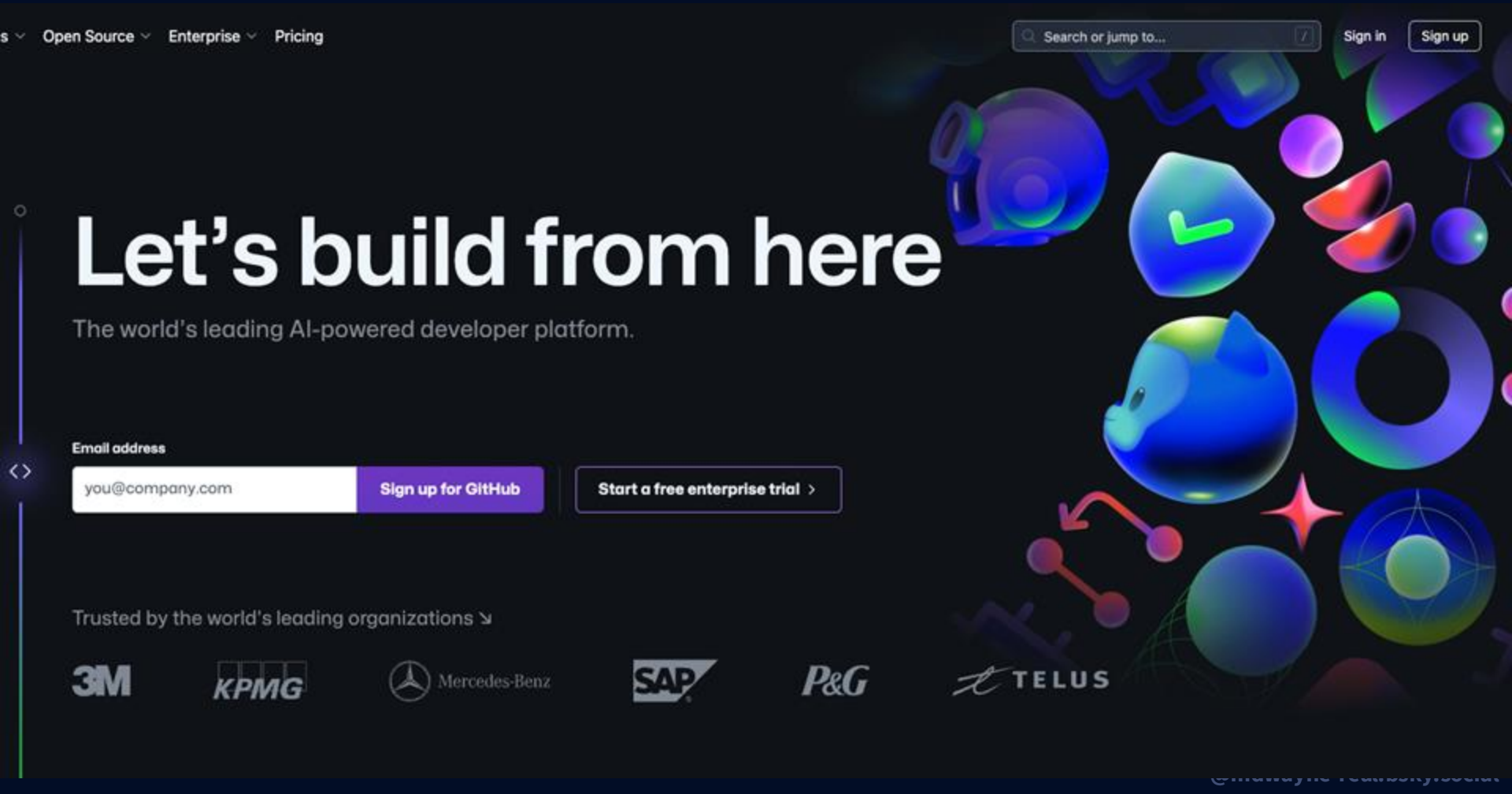
**Secrets authenticate access and encryption of software components**, such as:

- API keys
- Username/password pairs
- Database connection URLs
- Browser session tokens
- Certificates

Sensitive files such as **.env, .pem or .crt** are also secrets themselves

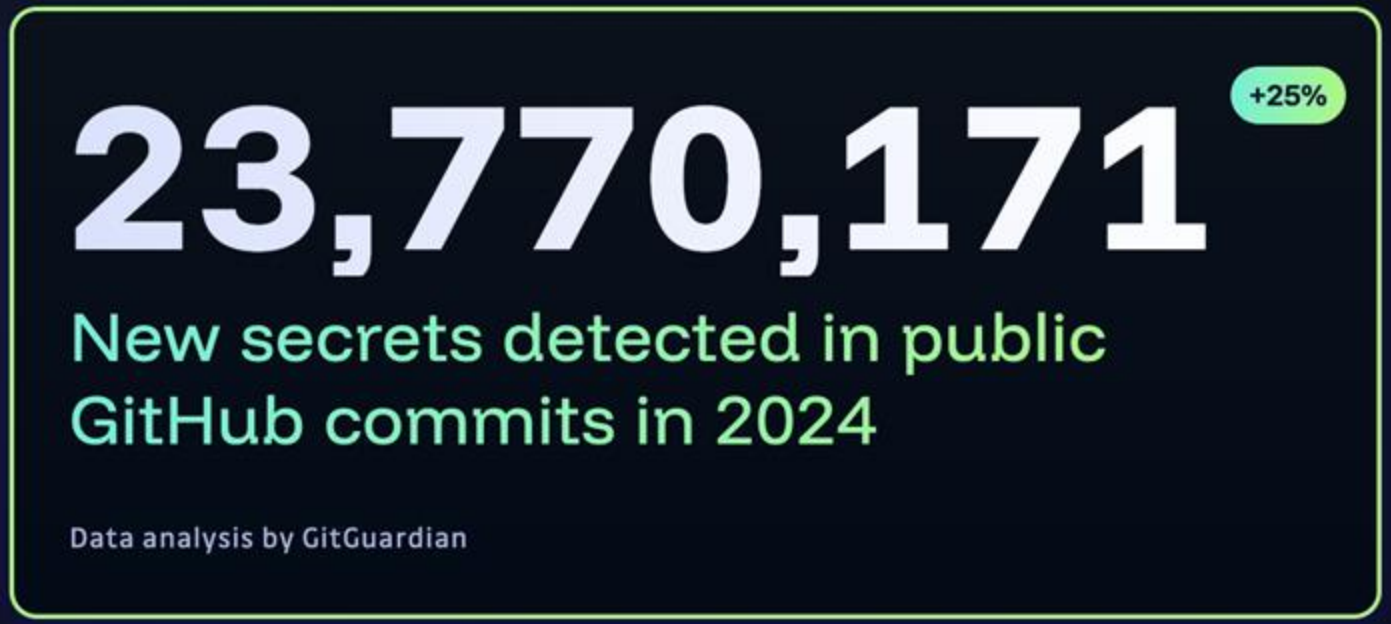
# Credential Leakage Is A Growing Problem


@mdwayne-real.bsky.social


Open Source
Enterprise
Pricing
Search or jump to...
Sign in
Sign up
Let's build from here
The world's leading AI-powered developer platform.
Email address
you@company.com
Sign up for GitHub
Start a free enterprise trial
Trusted by the world's leading organizations
3M
KPMG
Mercedes-Benz
SAP
P&G
TELUS

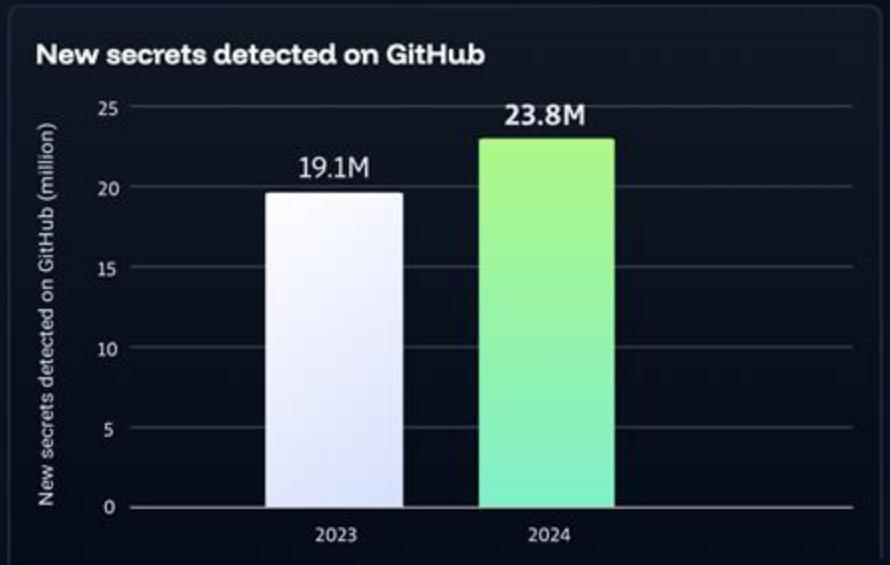

https://www.gitguardian.com/state-of-secrets-sprawl-report-2025


@mdwayne-real.bsky.social




in 2024 GitGuardian scanned **69.6M public repositories** of which at least **4.61% contained a secret.**


@mdwayne-real.bsky.social




**70%**

of the secrets leaked in 2022 are still valid

**15%**

commit authors leaked a secret

**1.9 Million**

pro-bono alert emails sent

@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

# Key findings from Verizon's 2025 Data Breach Investigations Report

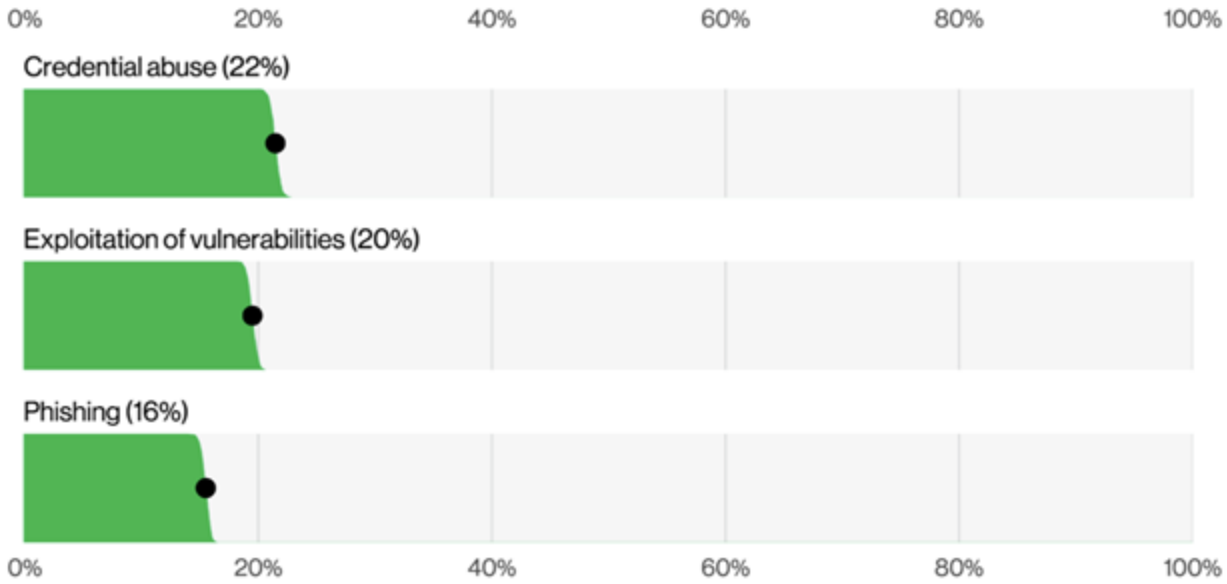



**Figure 5.** Known initial access vectors in non-Error, non-Misuse breaches (n=9,891)


@mdwayne-real.bsky.social

# Key findings from Verizon's 2025 Data Breach Investigations Report

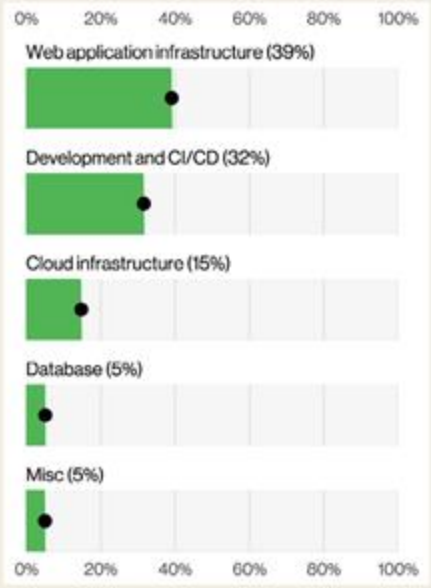



**Figure 12.** Top categories of exposed secrets in public git repos (n=441,780)

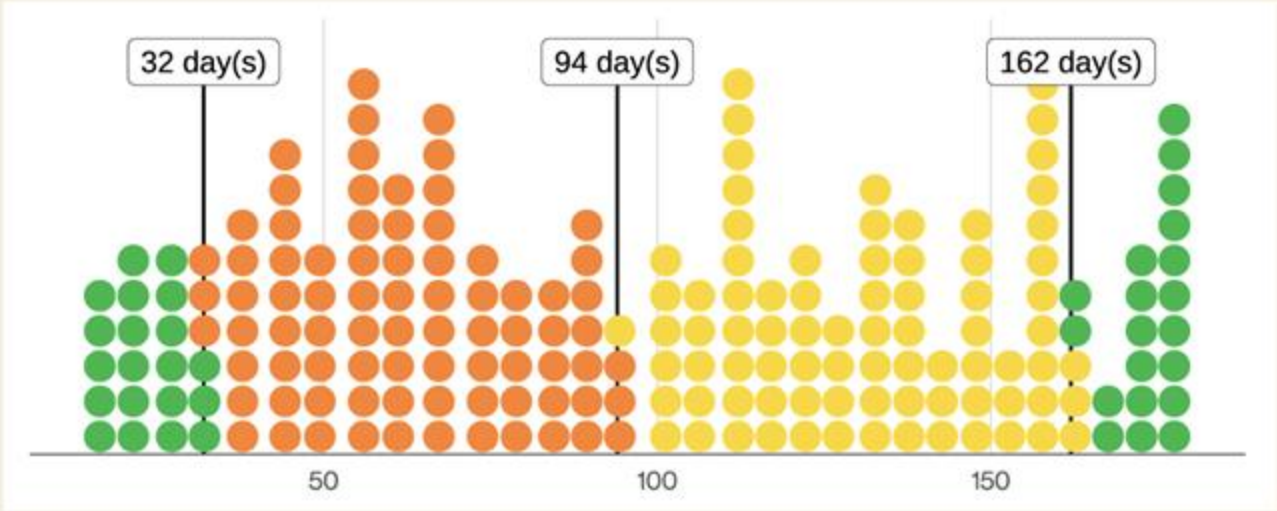



**Figure 13.** Distribution of days to remediate leaked secrets in git repositories (n=141 – each dot is 0.70 events)


@mdwayne-real.bsky.social


https://blog.gitguardian.com/voice-of-practitioners-2024/


@mdwayne-real.bsky.social

# The Confidence Gap

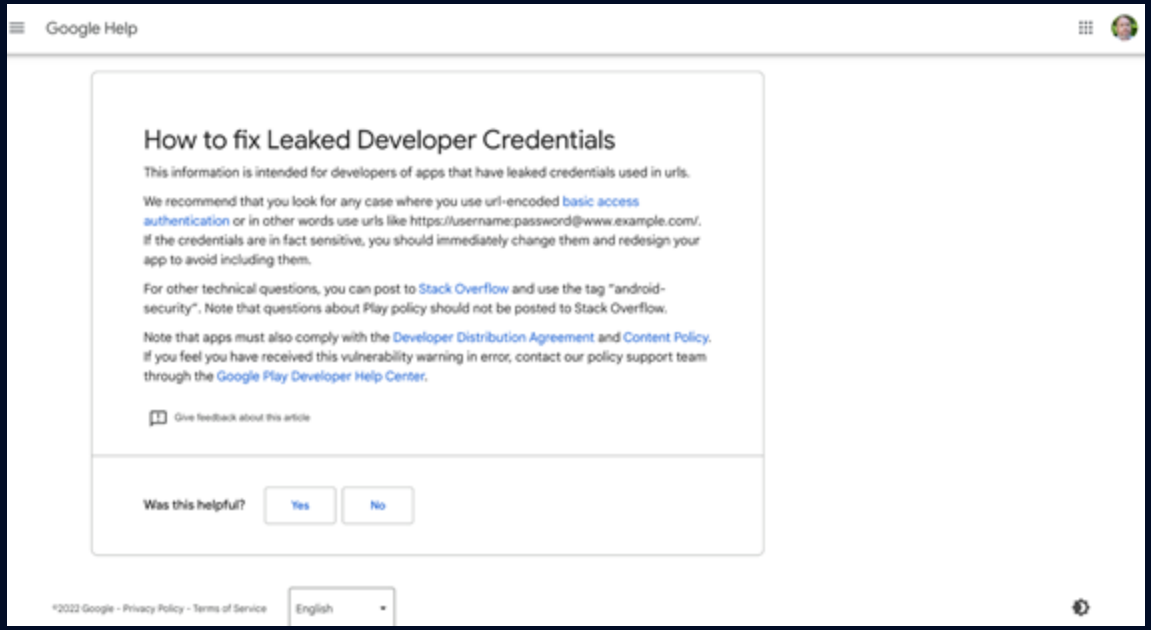
Despite heightened awareness and investment, a concerning gap exists between confidence and reality:

- 75% of respondents express strong confidence in their secrets management capabilities
- However, the average estimated time to remediate a leaked secret stands at 27 days
- Only 44% of developers are reported to follow security best practices
- Organizations maintain an average of 6 distinct secrets manager instances


@mdwayne-real.bsky.social

# What Can We Do?


@mdwayne-real.bsky.social




*"We recommend that you look for any case where you use url-encoded basic access authentication. If the credentials are in fact sensitive, you should **immediately change them** and redesign your app to avoid including them. " – Google Help*


@mdwayne-real.bsky.social

# Possible Solutions:

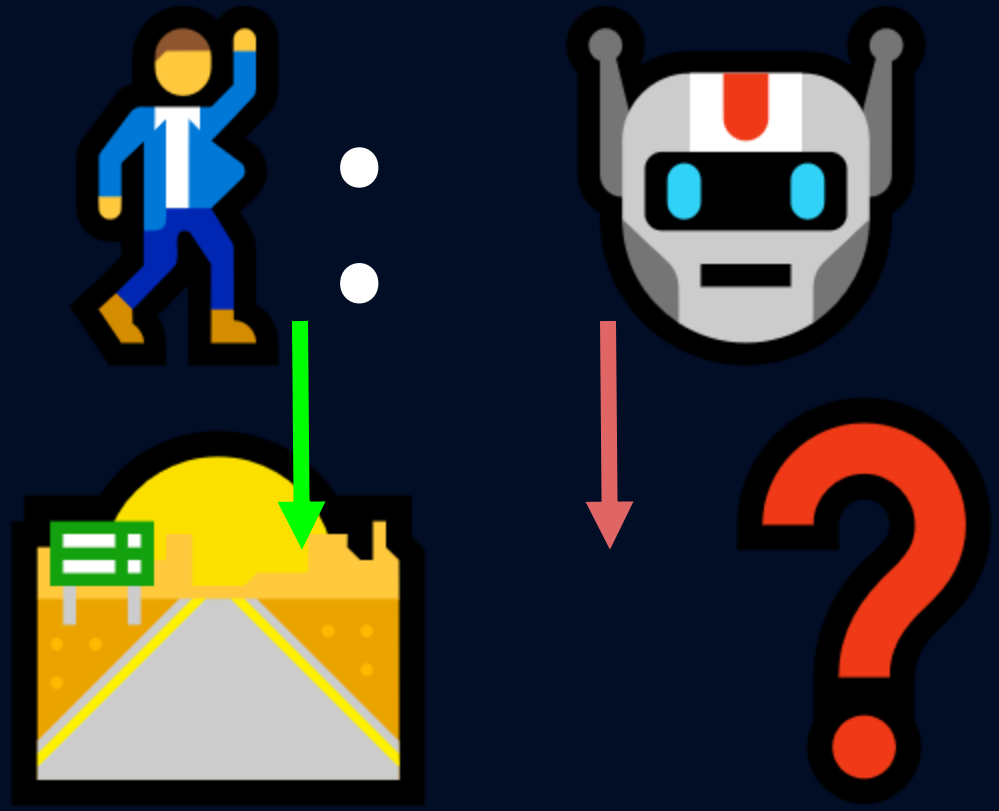
1. **When humans are involved, Phishing Resistant MFA (e.g. YubiKey with Biometrics)**

2. **Eliminate known credentials when possible (e.g. IAM roles, Passwordless, Password Managers)**

3. **Rotate credentials MUCH more often, using automation**

4. **Move to workload identity based authentication**


@mdwayne-real.bsky.social




@mdwayne-real.bsky.social

# Possible Solutions:

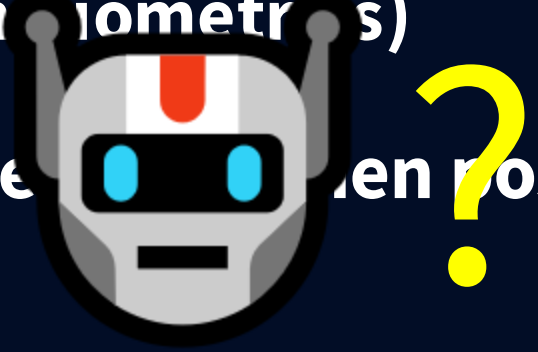
1. **hen humans are involved, Phishing Resistant FA (e.g. YubiKey with Biometrics)**
2. **Eliminate known credentials when possible (e.g. IAM roles)**
3. **Rotate credentials MUCH more often, using automation**
4. **Move to workload identity based authentication**

@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

# Possible Solutions:

1. When humans are involved, Phishing Resistant MFA (e.g. Yu[illegible]ey with [illegible]iometr[illegible]s)

2. Eliminate [illegible]wn crede[illegible]en possible (e.g. IAM r[illegible])



3. Rotate credentials MUCH more often, using automation

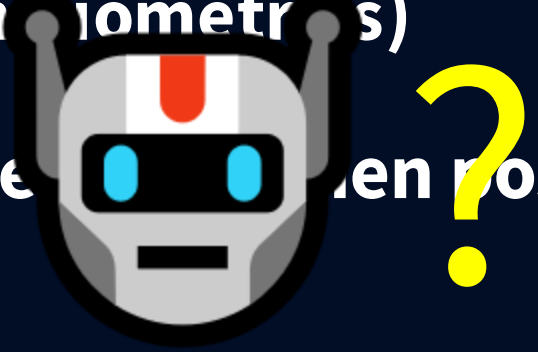
4. Move to workload identity based authentication


@mdwayne-real.bsky.social

# Possible Solutions:

1. When humans are involved, Phishing Resistant MFA (e.g. Yu[illegible]ey with [illegible]iometr[illegible]s)

2. Eliminate [illegible]wn crede[illegible]en possible (e.g. IAM r[illegible])



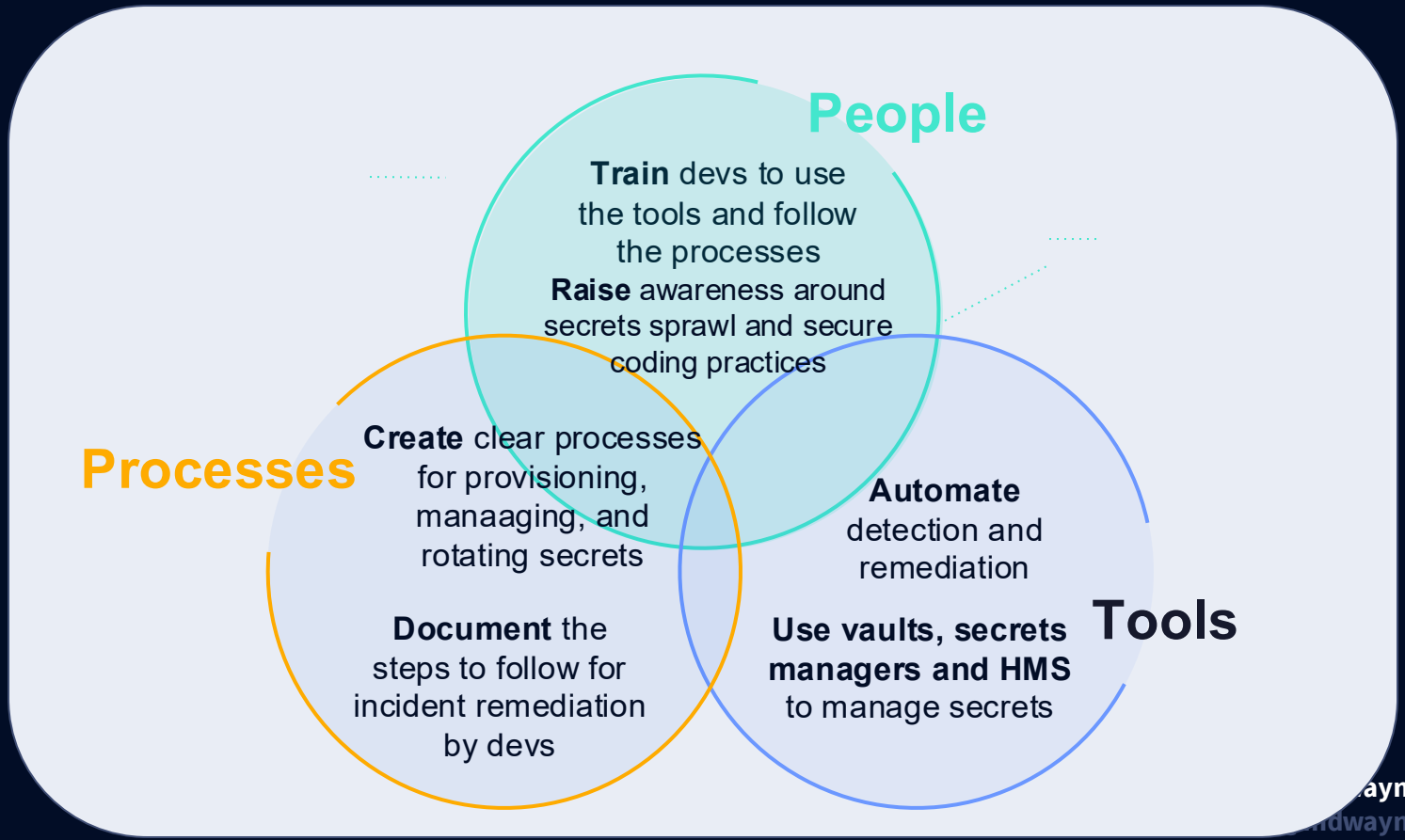
3. Rotate credentials MUCH more often, using automation

4. Move to workload identity based authentication

@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

```go
package main

import (
    "fmt"
    "os"
)

func main() {
    databaseName := "53CR3TD4T4B453"
    secretKey := "DIFFERENT SECRET"
    secretPhrase := "Always know where your towel is. - Douglas Adams, The Hitchhiker's Guide to the Galaxy"

    var dbName string
    var dbPass string

    fmt.Println("Please enter database name:")
    fmt.Scanf("%s", &dbName)

    fmt.Println("Please enter database password:")
    fmt.Scanf("%s", &dbPass)

    if dbName == databaseName && dbPass == secretKey {
        fmt.Println("Welcome to the database!")
        fmt.Println("Your secret phrase is: ", secretPhrase)
        os.Exit(0)
    }
    fmt.Println("Sorry, wrong database name or password")
}
```

# Three pillars of a secrets management program



# Who is responsible for fixing secrets sprawl?

# Who owns the risks?

Devs?

DevOps?
Operations?

Security?
CISO?

Exec Team?


@mdwayne-real.bsky.social

# Why not the IAM owner?

Why do we treat Non-Human Identities different than humans?

Does this person even exist in your org?


@mdwayne-real.bsky.social

# Who “owns” NHIs?

## Who owns the risks?

Devs

DevOps/
Operations

Security/
CISO

Exec Team

IAM/Identity

@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

Risks are NOT threats.
Risks are NOT vulns.
Risks are NOT exploits.

Risks are what you are set to lose
if things go bad.


– Walt Powell - Field CISO, CDW


@mdwayne-real.bsky.social

# A Formula For Measuring Security Risk

**Security Risk = Threat ✖ Exploitability ✖ Criticality**

- Where am I vulnerable?
- What is the likelihood of a successful attack?
- What would it cost the company or you?


@mdwayne-real.bsky.social

# Business Risks != Security Risks

"The Board does not know or care what a CVE is.
They care that something is going to make them lose money.

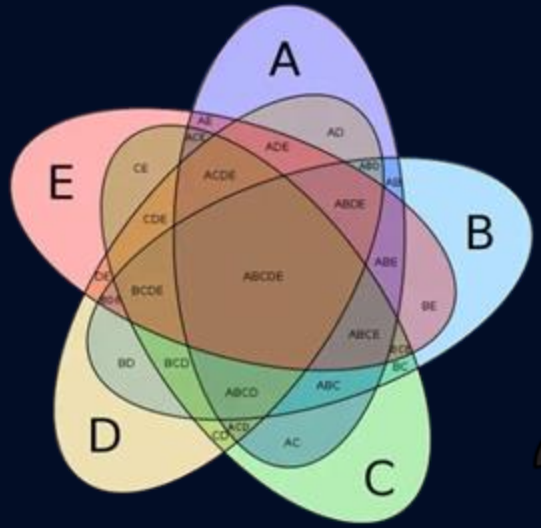
Period."

– again, thank you to Walt Powell - Field CISO, CDW


@mdwayne-real.bsky.social

Devs

DevOps/
Operations




Security/
CISO

Exec Team

IAM/Identity

@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

# We must get EVERYONE on board this same submarine

Devs

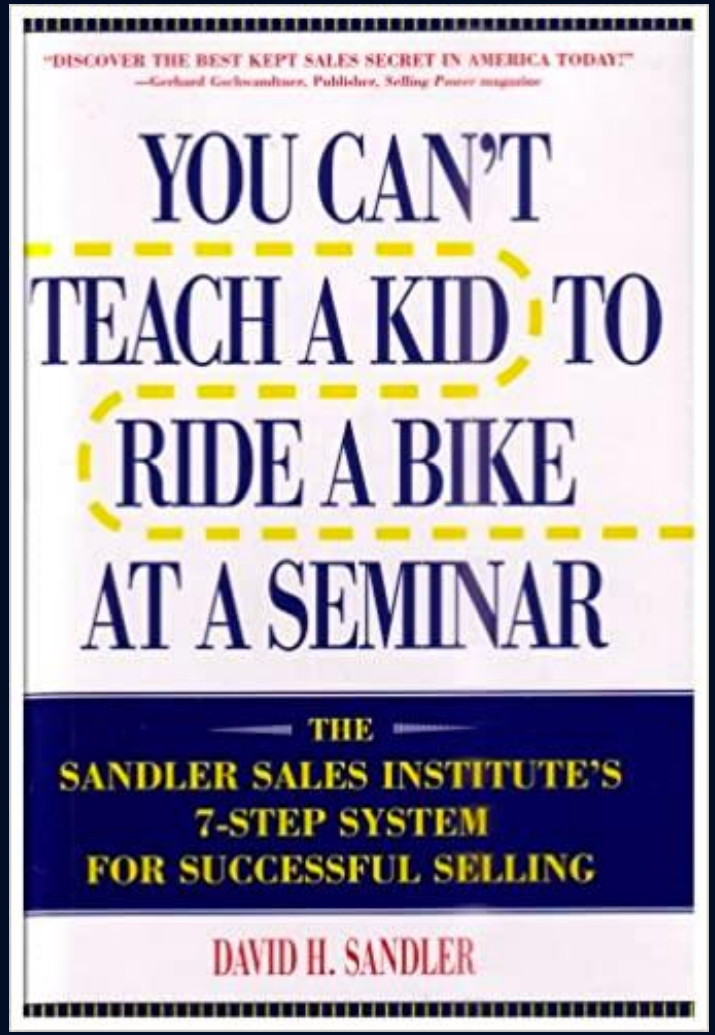
DevOps/
Operations

Security/
CISO

Exec Team

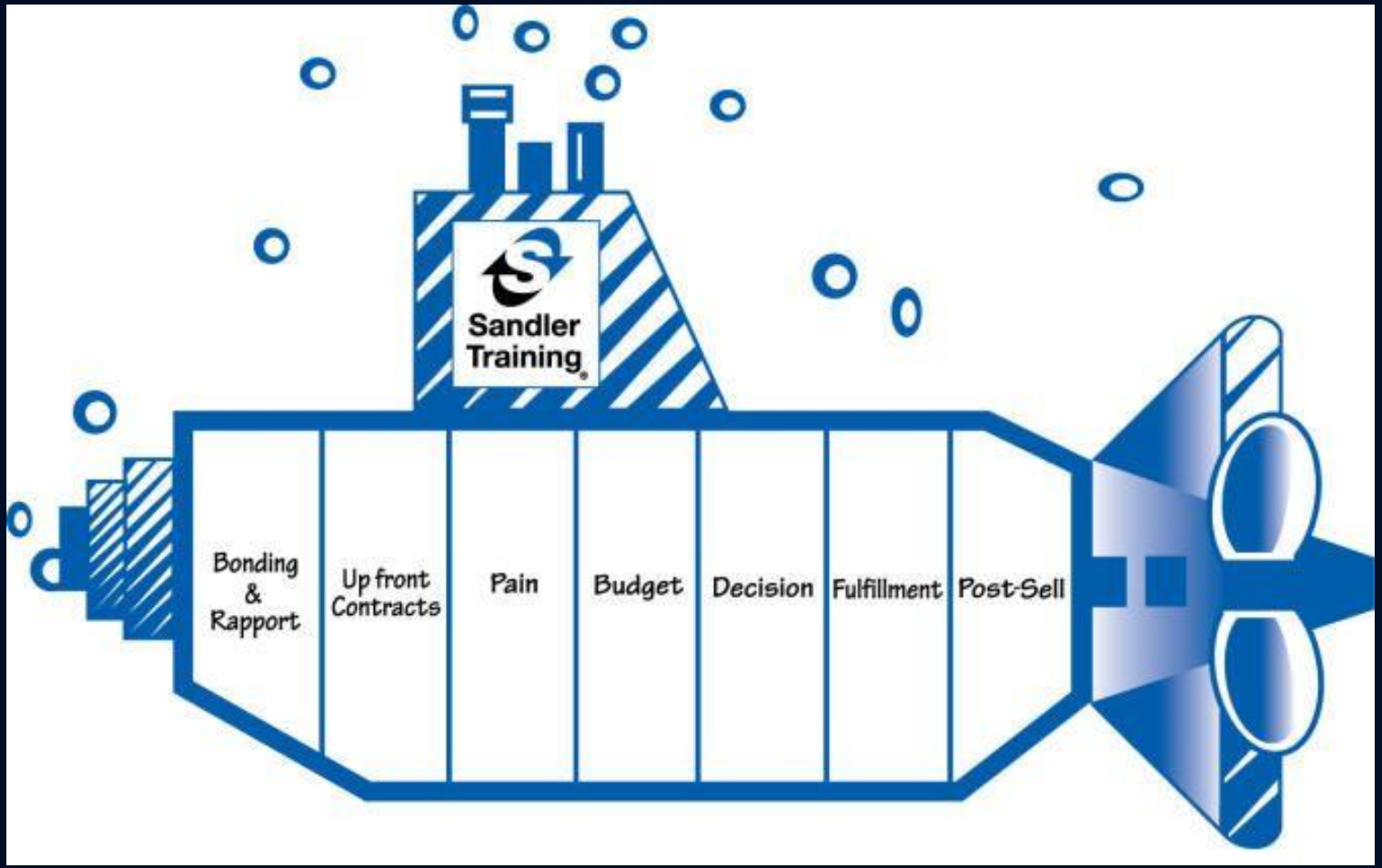
IAM/Identity

@mdwayne-real.bsky.social
@mdwayne-real.bsky.social





@mdwayne-real.bsky.social



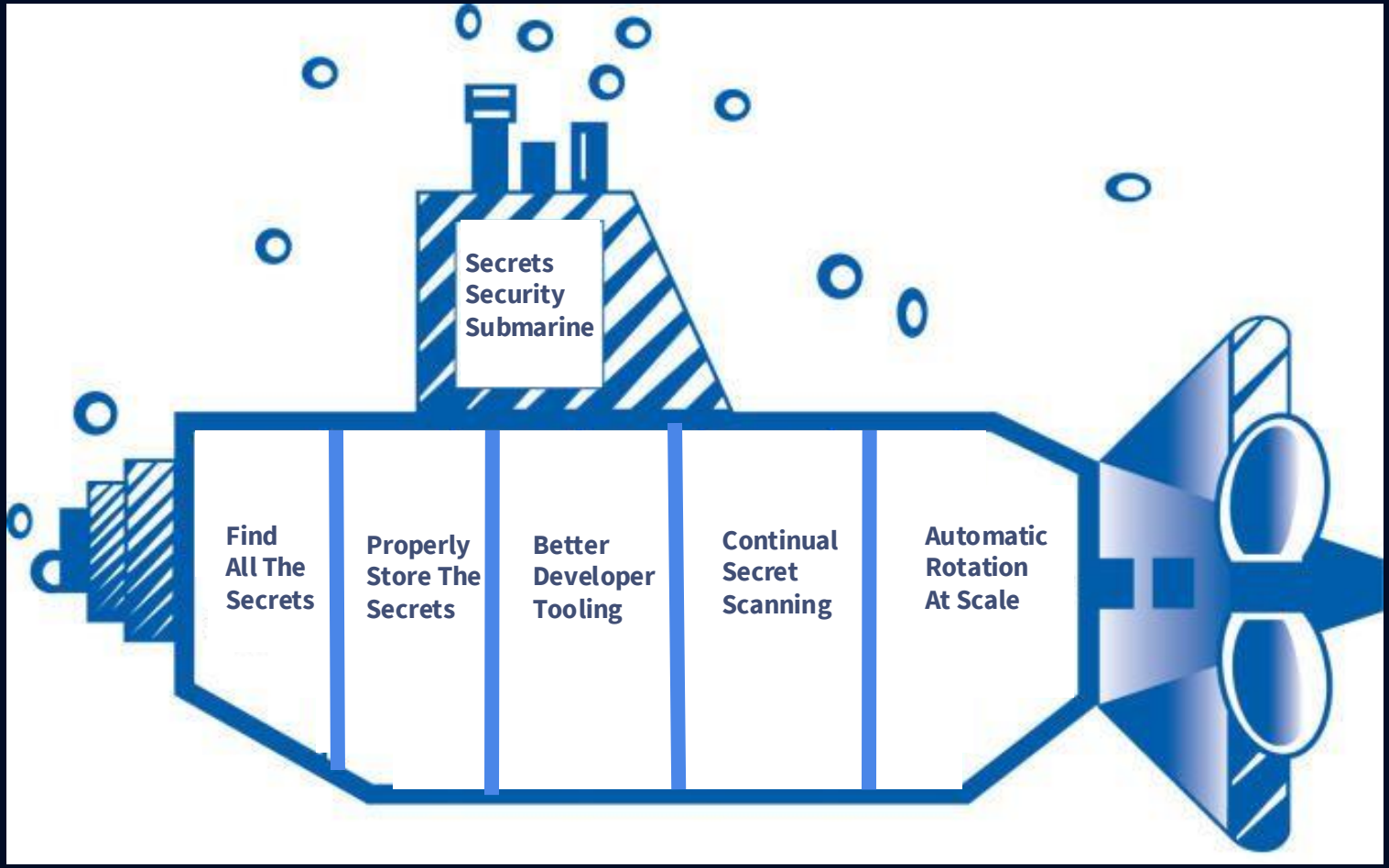


@mdwayne-real.bsky.social

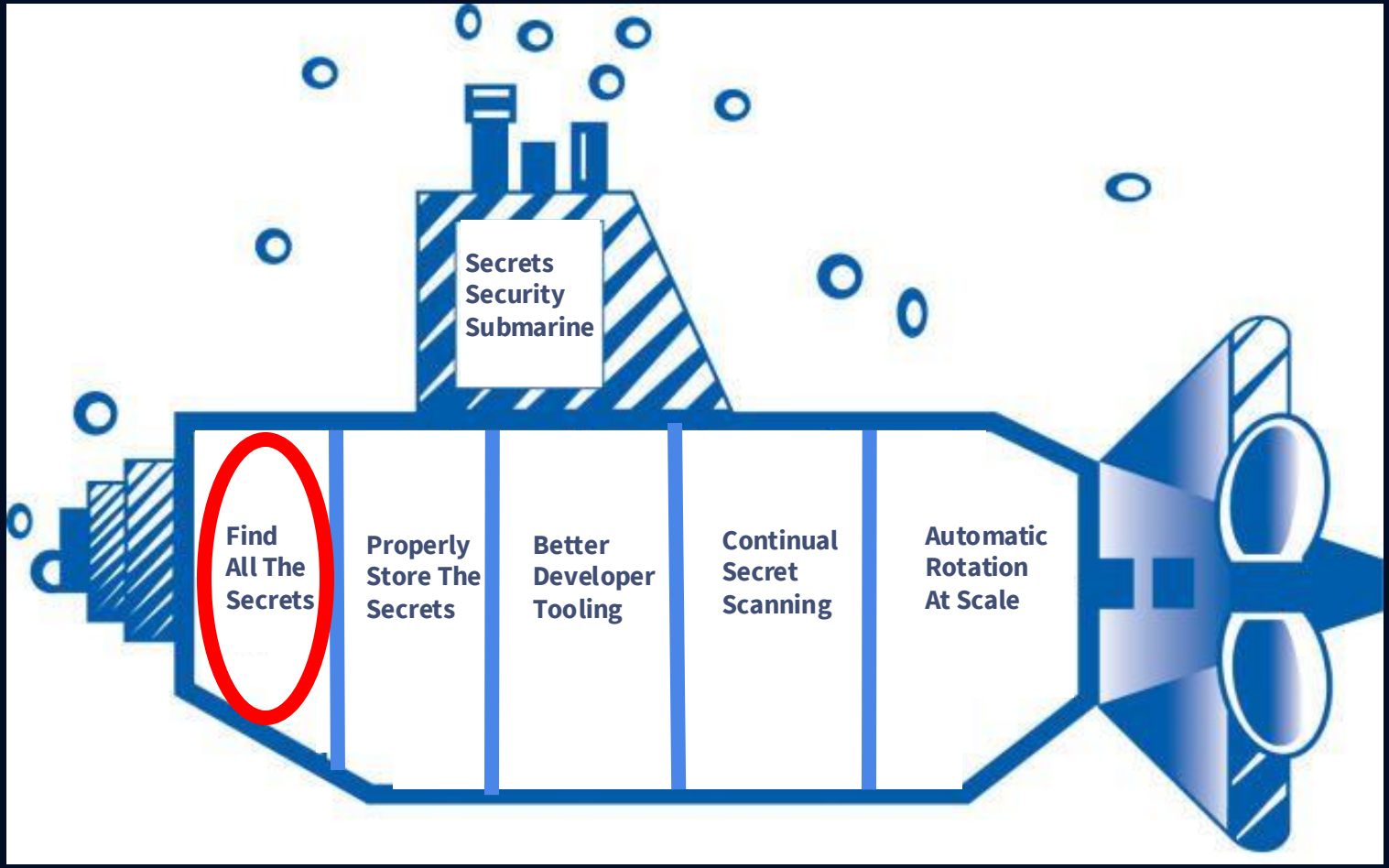
# Our new game plan:

- Find All The Secrets
- Properly Store The Secrets
- Adopt Better Developer Tooling
- Continual Secrets Scanning
- Automatic Rotation At Scale


@mdwayne-real.bsky.social




@mdwayne-real.bsky.social
@mdwayne-real.bsky.social





@mdwayne-real.bsky.social


**“Step 1: Scope your work**
**The first step in the threat modeling process is concerned with gaining an understanding of what you’re working on.”**

https://owasp.org/www-community/Threat_Modeling_Process


@mdwayne-real.bsky.social

# Hopefully everyone is keeping track…right?

…Right!?!?

Devs

DevOps/
Operations

IAM/Identity


@mdwayne-real.bsky.social



@mdwayne-real.bsky.social

# Credentials can appear in plaintext in:

- Code
- Config files
- Jira
- Slack/Teams
- Confluence
- ~/secrets.txt
- Other terrifying places


@mdwayne-real.bsky.social

# GitGuardian

Yelp/**detect-secrets**

aws **git-secrets**

trufflesecurity/trufflehog

`git grep -E <pattern>`

PingSafe™

gitLeaks


@mdwayne-real.bsky.social

# Now we all know the scope of the issue…now what?

Devs

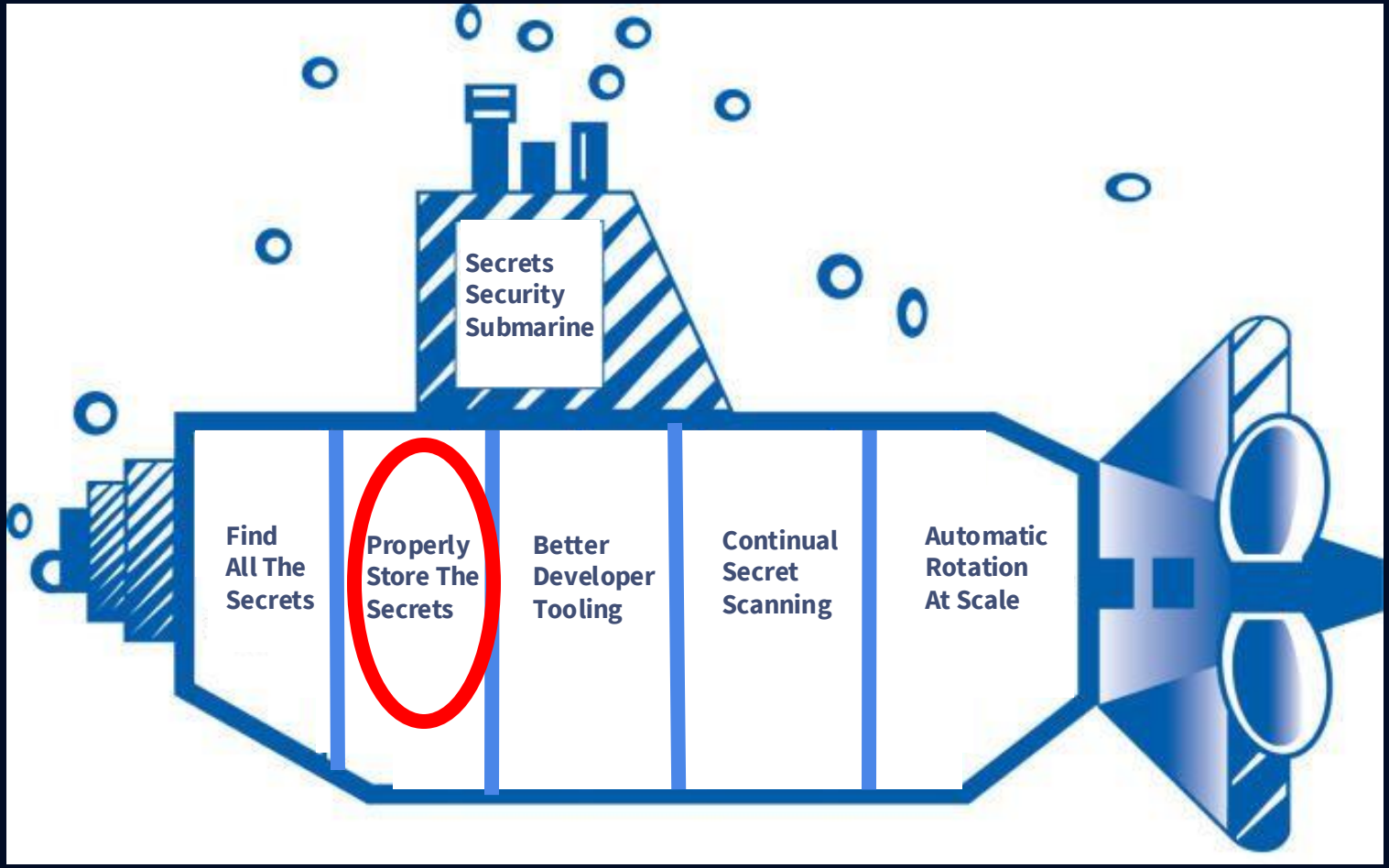
DevOps/
Operations

Security/
CISO

Exec Team

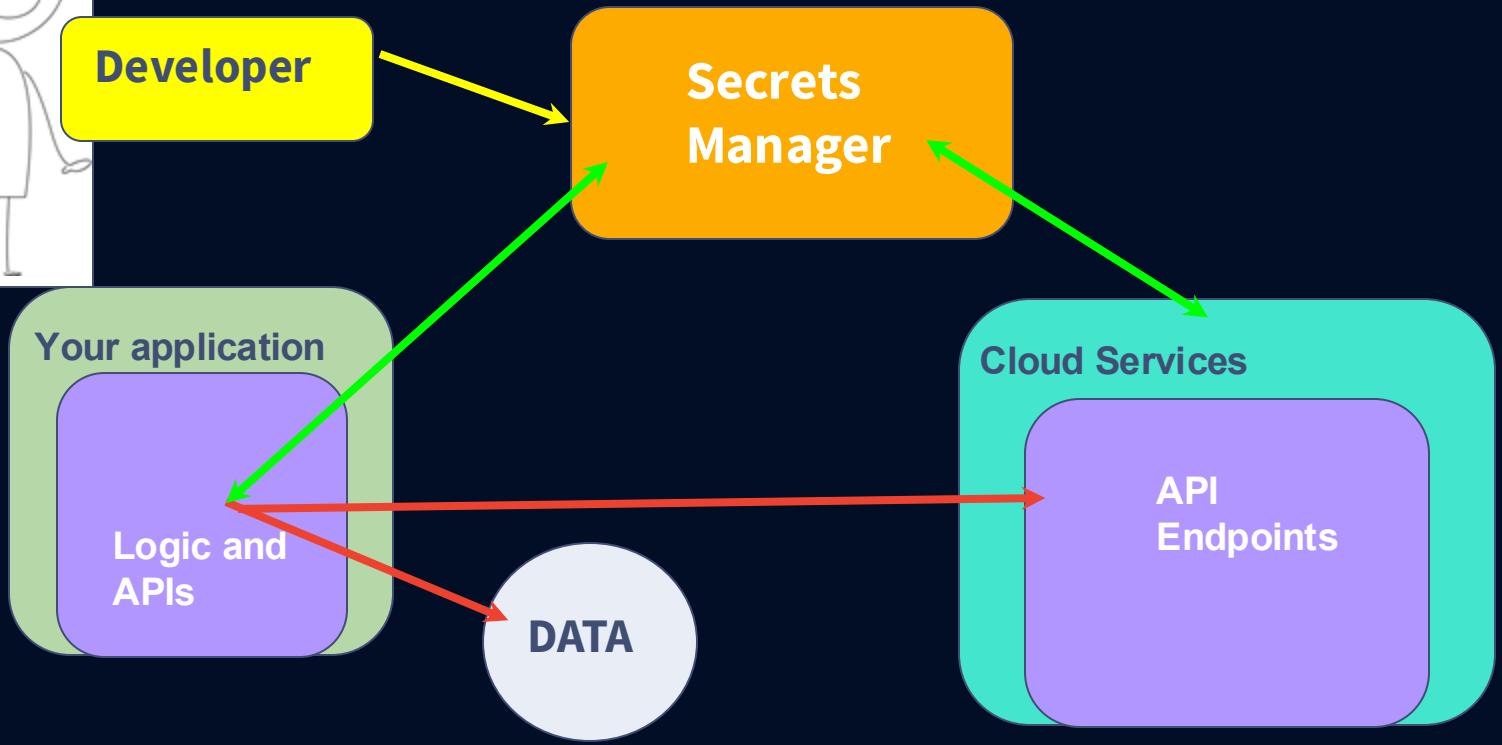
IAM/Identity


@mdwayne-real.bsky.social




@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

# Basic Secret Manager Architecture





@mcdwayne

# Basic needs for a secrets manager

1. Encrypts secrets at rest and in transit
2. Available across all environments
3. Centralized reporting
4. Easy enough to use for development


@mcdwayne

# Are you “all in” on one Cloud provider?







@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

# Multicloud, Or On Prem, Or A Mix?

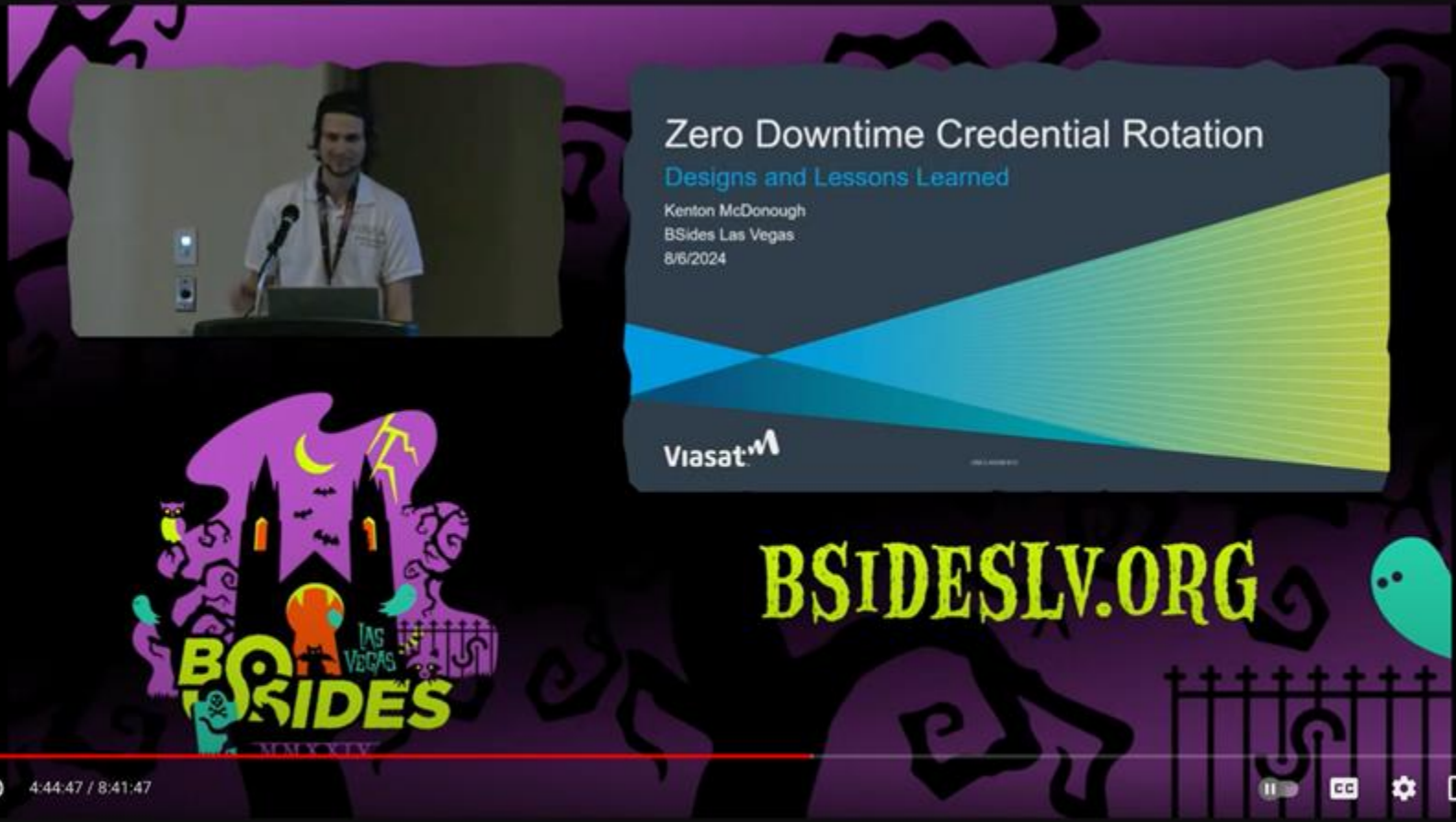




@mdwayne-real.bsky.social
@mdwayne-real.bsky.social


```python
# Reading a secret
read_response = client.secrets.kv.read_secret_version(path='my-secret-password')

password = read_response['data']['data']['password']

if password != 'Hashi123':
    sys.exit('unexpected password')

print('Access granted!')
```


@mcdwayne




# BsidesLV 2024 - PasswordsCon - Tuesday

BSidesLV
4.25K subscribers

Subscribe

4 | Share | Download | Clip | ...

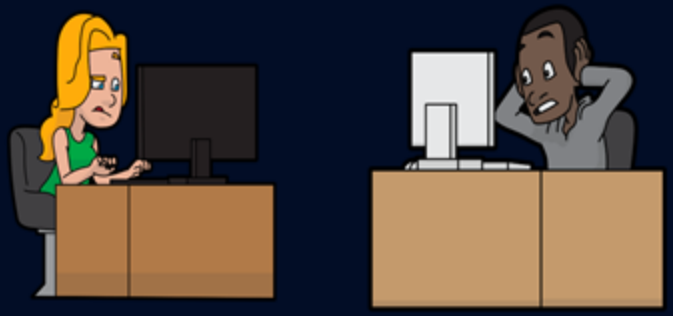
Live chat replay was turned off for this video.

# Prioritizing secrets to add

1. New secrets
2. Production/Critical secrets
3. Legacy secrets
4. Zombie secrets

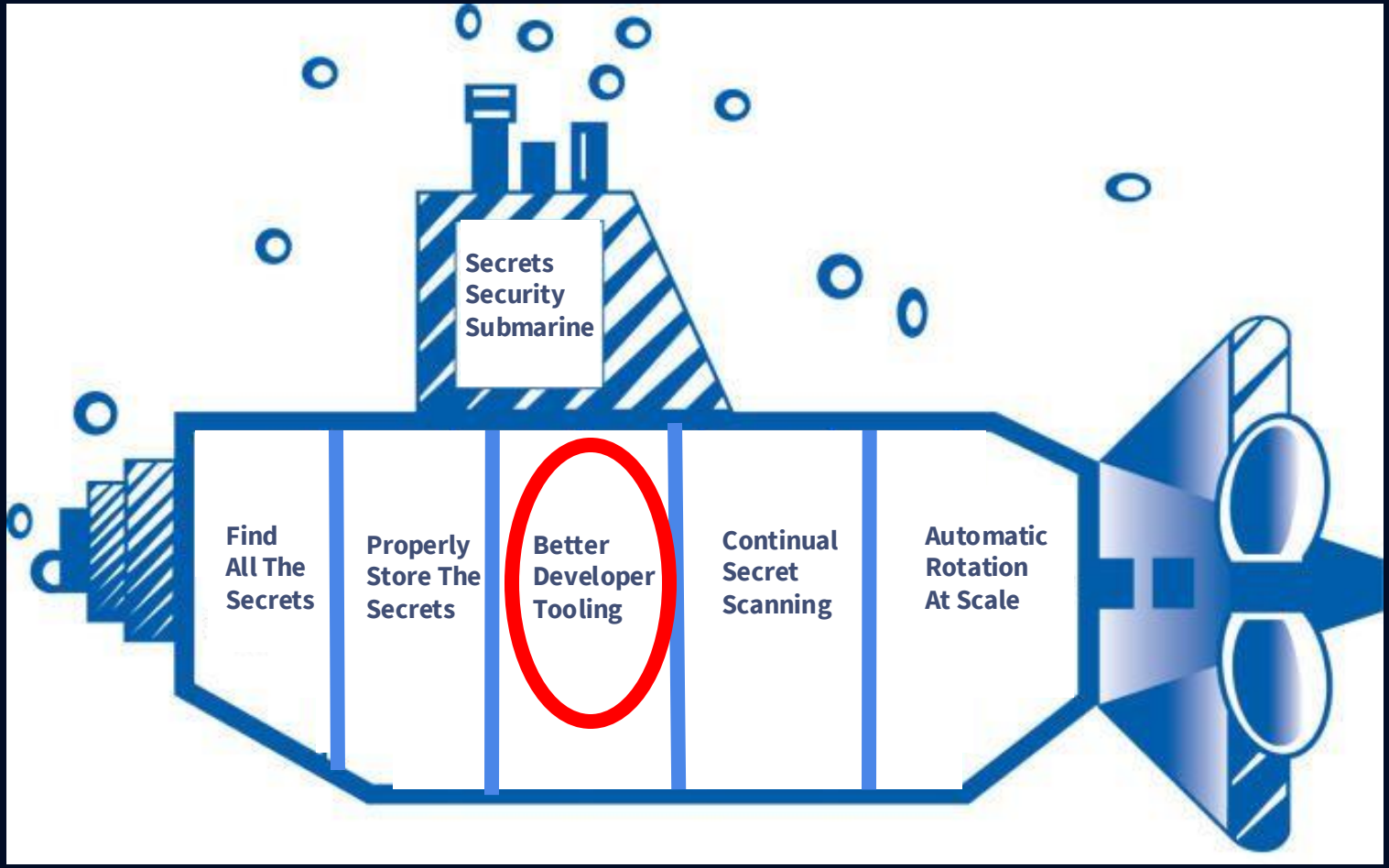

@mcdwayne

# And how do we get Devs/DevOps to use it?

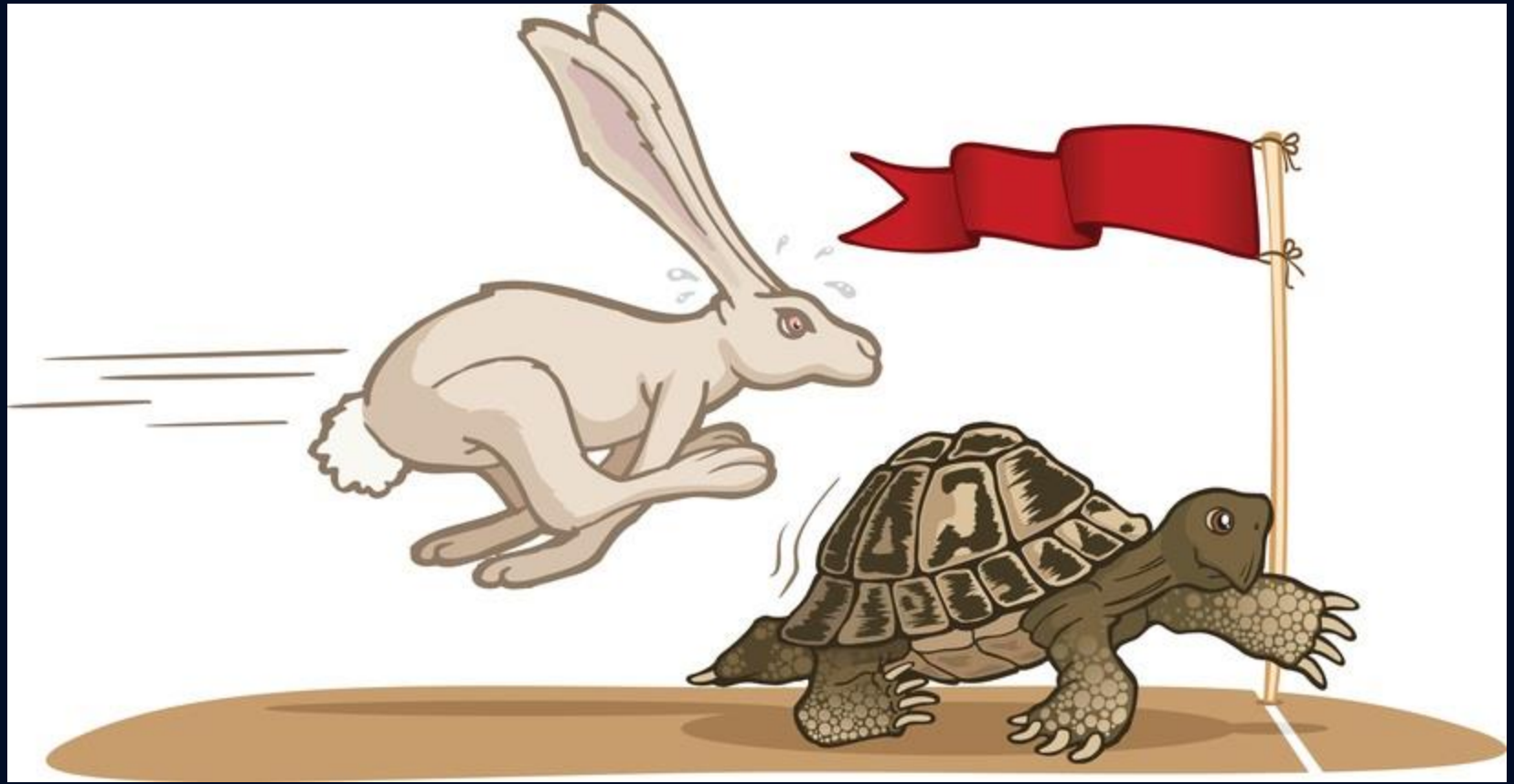


Devs DevOps


@mdwayne-real.bsky.social




@mdwayne-real.bsky.social
@mdwayne-real.bsky.social




@mdwayne-real.bsky.social

# Security Engineering


@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

# Security Engineering


@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

# Git hooks or IDE extension based tool logic :

1. If a plaintext secret is detected before commit - DO NOT COMMIT IT
2. Check if it is already in your vault(s)
   -If so, return the call to the vault in the IDE
   -else, store it in the vault
1. Replace the secret with the vault call
2. Upon successful commit call to rotate the secret

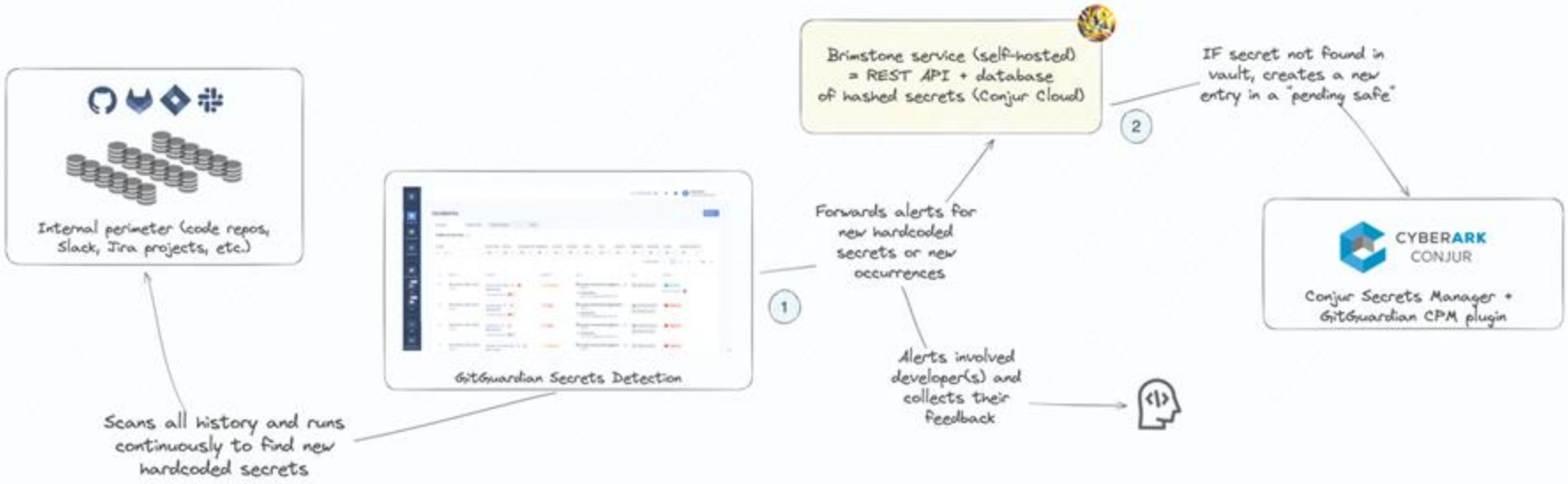

@mdwayne-real.bsky.social

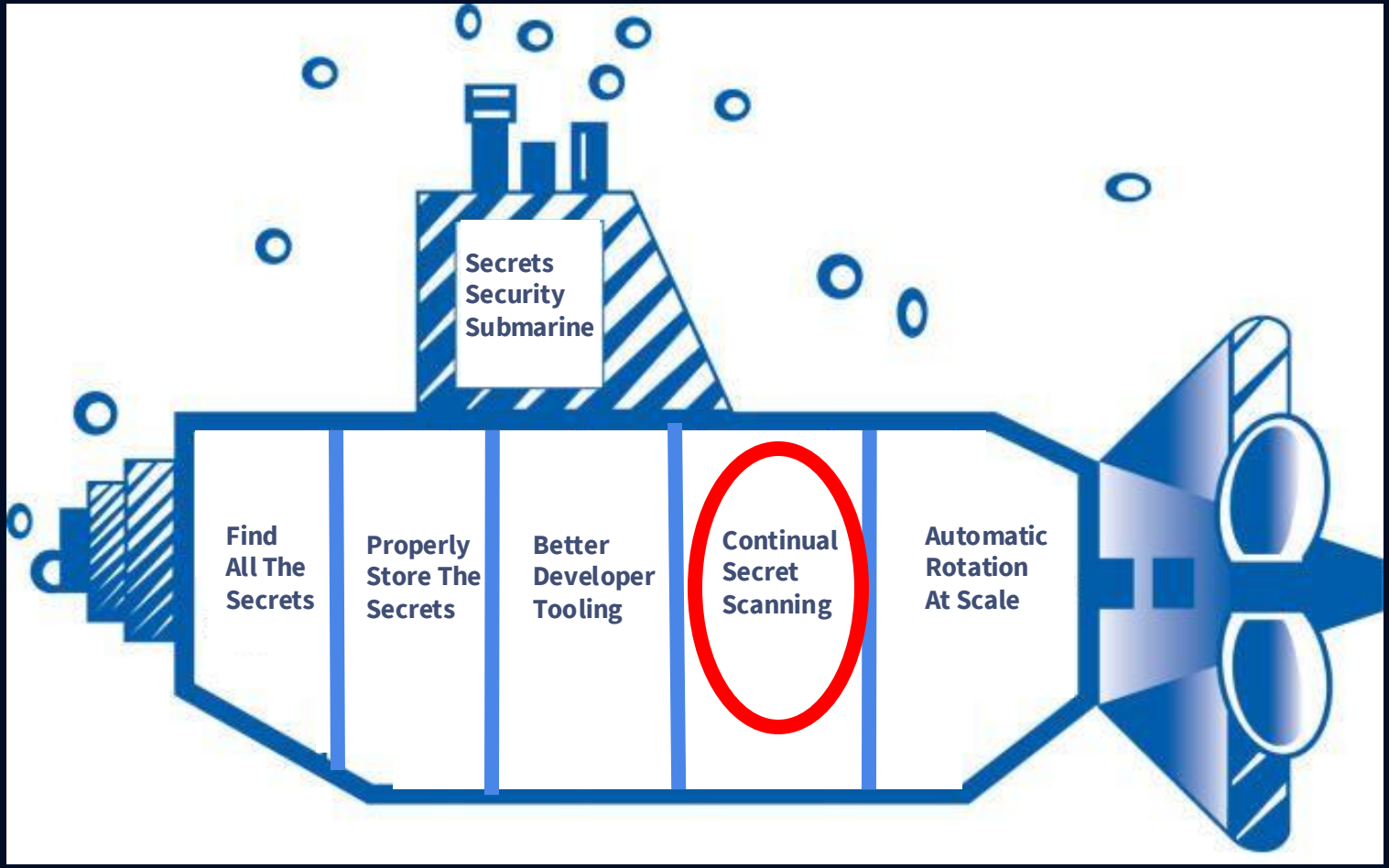
CYBERARK

GitGuardian

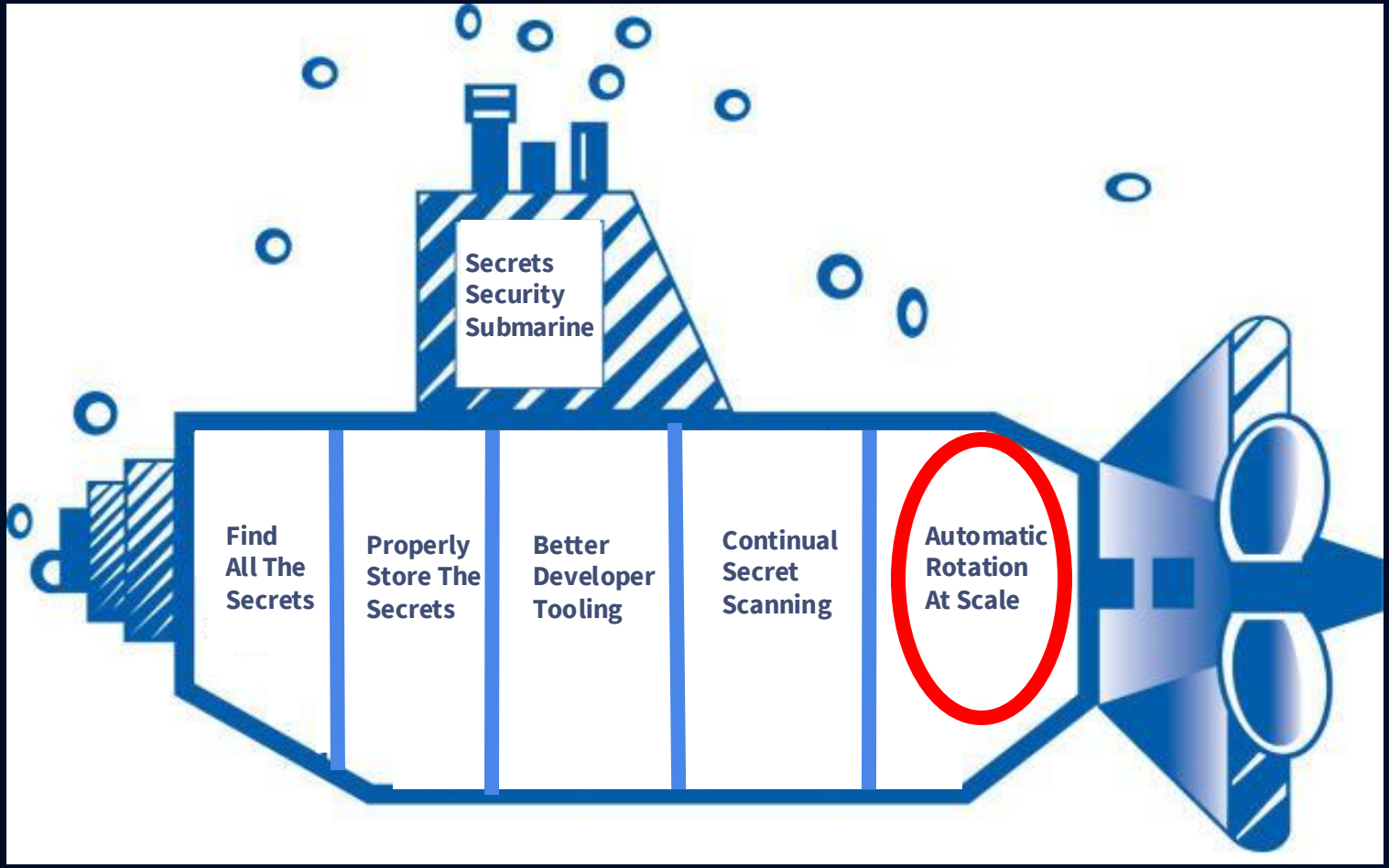
## Use case 2 – Enforcing vault usage with secrets discovery

GitGuardian helps security teams find "untracked" or "out-of-band" secrets to store them back in Conjur Cloud Secrets Manager.






@mdwayne-real.bsky.social
@mdwayne-real.bsky.social




@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

# Script Logic Needed:

1. Create new secret for service in question

   1.5. Test the new secret
1. Swap in the new secret for the existing one
2. Test to make sure this did not break access
3. Clean up step for internal labeling


@mdwayne-real.bsky.social

# aws-secrets-manager-rotation-lambdas

Code | Blame | 174 lines (123 loc) · 6.85 KB | Raw

```python
# Copyright 2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0

import boto3
import logging
import os

logger = logging.getLogger()
logger.setLevel(logging.INFO)


def lambda_handler(event, context):
    """Secrets Manager Rotation Template

    This is a template for creating an AWS Secrets Manager rotation lambda

    Args:
        event (dict): Lambda dictionary of event parameters. These keys must include the following:
            - SecretId: The secret ARN or identifier
            - ClientRequestToken: The ClientRequestToken of the secret version
            - Step: The rotation step (one of createSecret, setSecret, testSecret, or finishSecret)

        context (LambdaContext): The Lambda runtime information

    Raises:
        ResourceNotFoundException: If the secret with the specified arn and stage does not exist

        ValueError: If the secret is not properly configured for rotation

        KeyError: If the event parameters do not contain the expected keys

    """
```

@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

# Are you all in on one Cloud provider?

Set up automatic rotation for AWS Secrets Manager secrets using the AWS CLI


Configuring Automatic Key Rotation in GCP for Existing Keys

You can set up automatic key rotation for existing keys by using either the GCP Console or the gCloud CLI.


Configure cryptographic key auto-rotation in Azure Key Vault



@mdwayne-real.bsky.social

# Multicloud, Or On Prem, Or A Mix?

**Can you rotate the secret through an API/CLI call?**


@mdwayne-real.bsky.social


API

```
def rotate
    conjur_url =
Conjur.configuration.appliance_url
    account = Conjur.configuration.account
    var_name = params['id']
    token =
Base64.strict_encode64(api.token.to_json)
    RestClient.post(
      "#{conjur_url}/secrets/#
{account}/variable/#{var_name}?expirations",
      '',
      :Authorization => "Token token=\"#
{token}\""
    )
```

## Enable rotation

This section describes how to enable rotation in Conjur.

circleci

```
curl --request PUT \
  --url https://circleci.com/api/v2/context/%7Bcontext-id%7D/environment-variable/POSTGRES_USER \
  --header 'authorization: Basic REPLACE_BASIC_AUTH' \
  --header 'content-type: application/json' \
  --data '{"value":"some-secret-value"}'
```

slack

```
POST /api/oauth.v2.access
HOST slack.com
Content-Type: application/x-www-form-urlencoded

client_id=60503450.61416
client_secret=8bc5fc53901afc11c
grant_type=refresh_token
refresh_token=xoxe-1-...
```

@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

# Is there another option?


@mdwayne-real.bsky.social

We accept the reality that traditional

Long-lived credential based approaches

are a

TERRIBLE

idea for workload/machine identities


@mdwayne-real.bsky.social

# Possible Solutions:

1. When humans are involved, Phishing Resistant MFA (e.g. YubiKey with Biometrics)
2. Eliminate known credentials when possible (e.g. IAM roles, Passwordless)
3. Rotate credentials MUCH more often, using automation

# 4. Move to workload identity based authentication


@mdwayne-real.bsky.social

# Universal identity control plane for distributed systems

SPIFFE and SPIRE provide strongly attested, cryptographic identities to workloads across a wide variety of platforms


@mdwayne-real.bsky.social

SPIFFE, the Secure Production Identity Framework For Everyone (SPIFFE) Project defines a framework and set of standards for identifying and securing communications between application services.

SPIRE (the SPIFFE Runtime Environment) is a toolchain of APIs for establishing trust between software systems across a wide variety of hosting platforms.


@mdwayne-real.bsky.social

## Key Use Cases

Secure microservices communication automatically with Envoy, X.509 PKI, or JWT

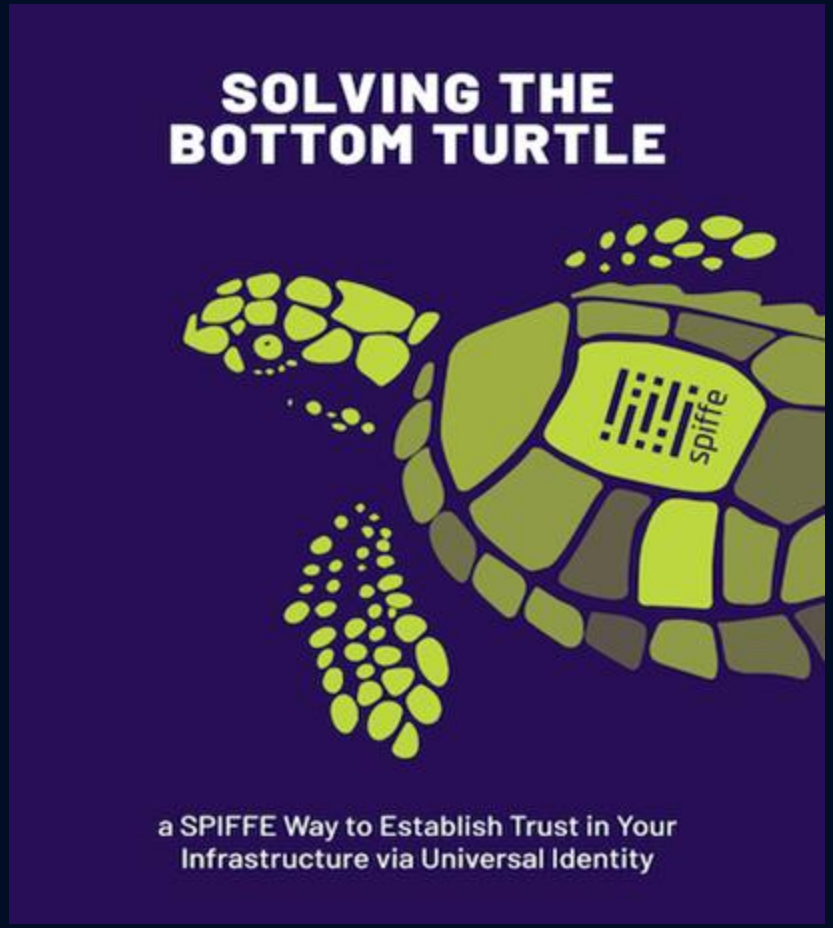
Authenticate securely to common databases or platforms without passwords or API keys

Build, bridge, and extend service mesh across organizations without sharing keys

Cross-service authentication for zero trust security model

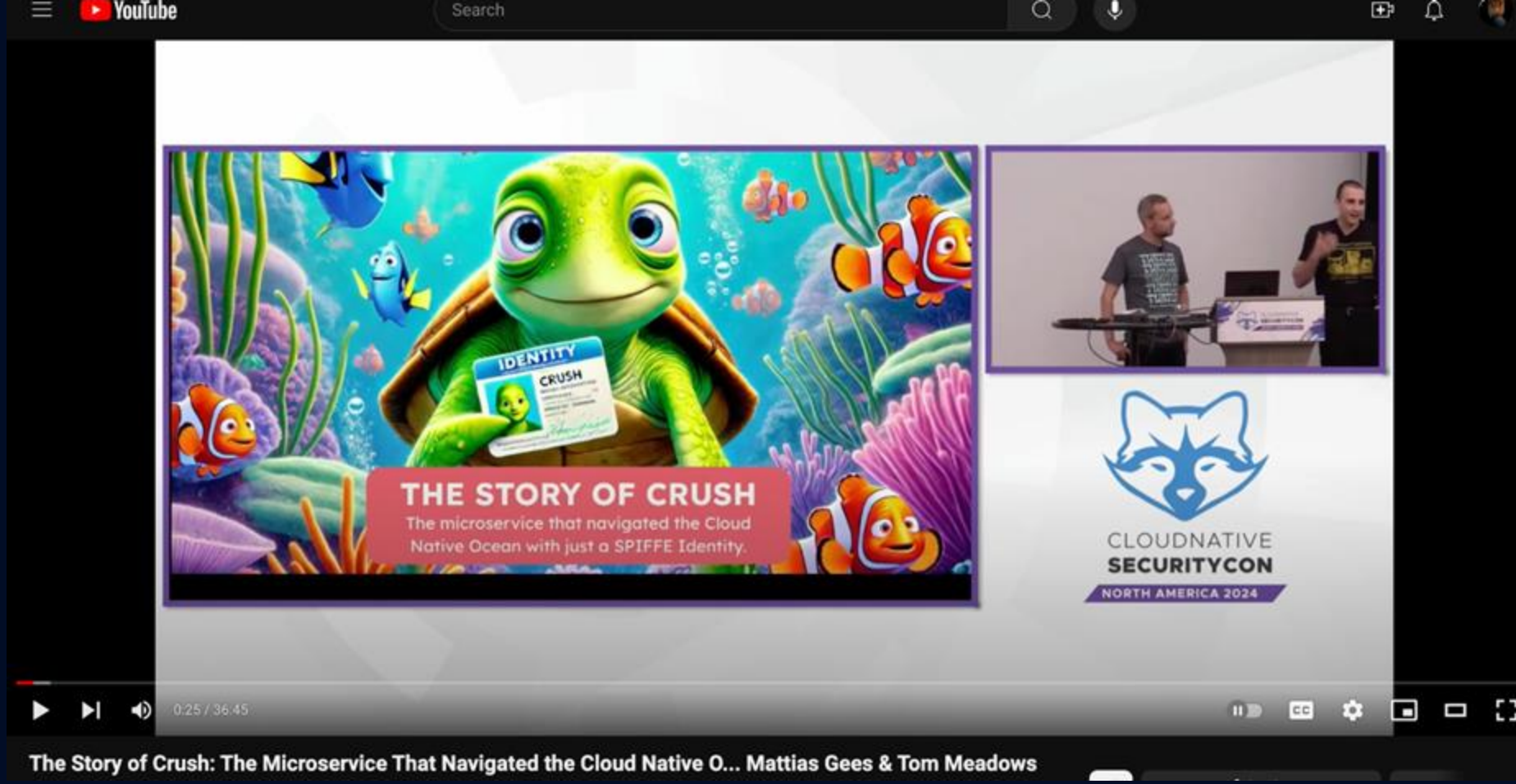
Bridging the gap between Kubernetes and other platforms


@mdwayne-real.bsky.social




https://spiffe.io/book/

@mdwayne-real.bsky.social
@mdwayne-real.bsky.social




https://youtu.be/4HHvEamsxjs?si=KZhaCL_51QKAMuG3


@mdwayne-real.bsky.social

# WIMSE IETF Working Group

## Workload Identity in Multi-System Environments

The IETF Workload Identity in Multi-System Environments (WIMSE) working group addresses authentication and identity challenges in distributed microservices architectures. WIMSE focuses on defining standardized solutions for managing workload identities and secure communication between services.

### Key Objectives:

- Develop a unified architecture for securely managing workload identities.
- Create token-based solutions to authenticate and secure REST/HTTP communications between workloads.
- Ensure interoperability across diverse platforms.
- Collaborate with other IETF working groups and external organizations to ensure conhesive standards.

Most collaboration in the working group is via the mailing list.

For more information about IETF, visit the official IETF website.


@mdwayne-real.bsky.social


# Workload Identity in a Multi System Environment (WIMSE) Architecture

- Workload

A workload is a running instance of software executing for a specific purpose. Workload typically interacts with other parts of a larger system. A workload may exist for a very short durations of time (fraction of a second) and run for a specific purpose such as to provide a response to an API request. Other kinds of workloads may execute for a very long duration, such as months or years. Examples include database services and machine learning training jobs.

https://datatracker.ietf.org/doc/draft-ietf-wimse-arch/

@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

# Kubernetes v1.33: From Secrets to Service Accounts: Kubernetes Image Pulls Evolved

By **Anish Ramasekar (Microsoft)** | Wednesday, May 07, 2025


@mdwayne-real.bsky.social

# How it works

## 1. Service Account tokens for credential providers

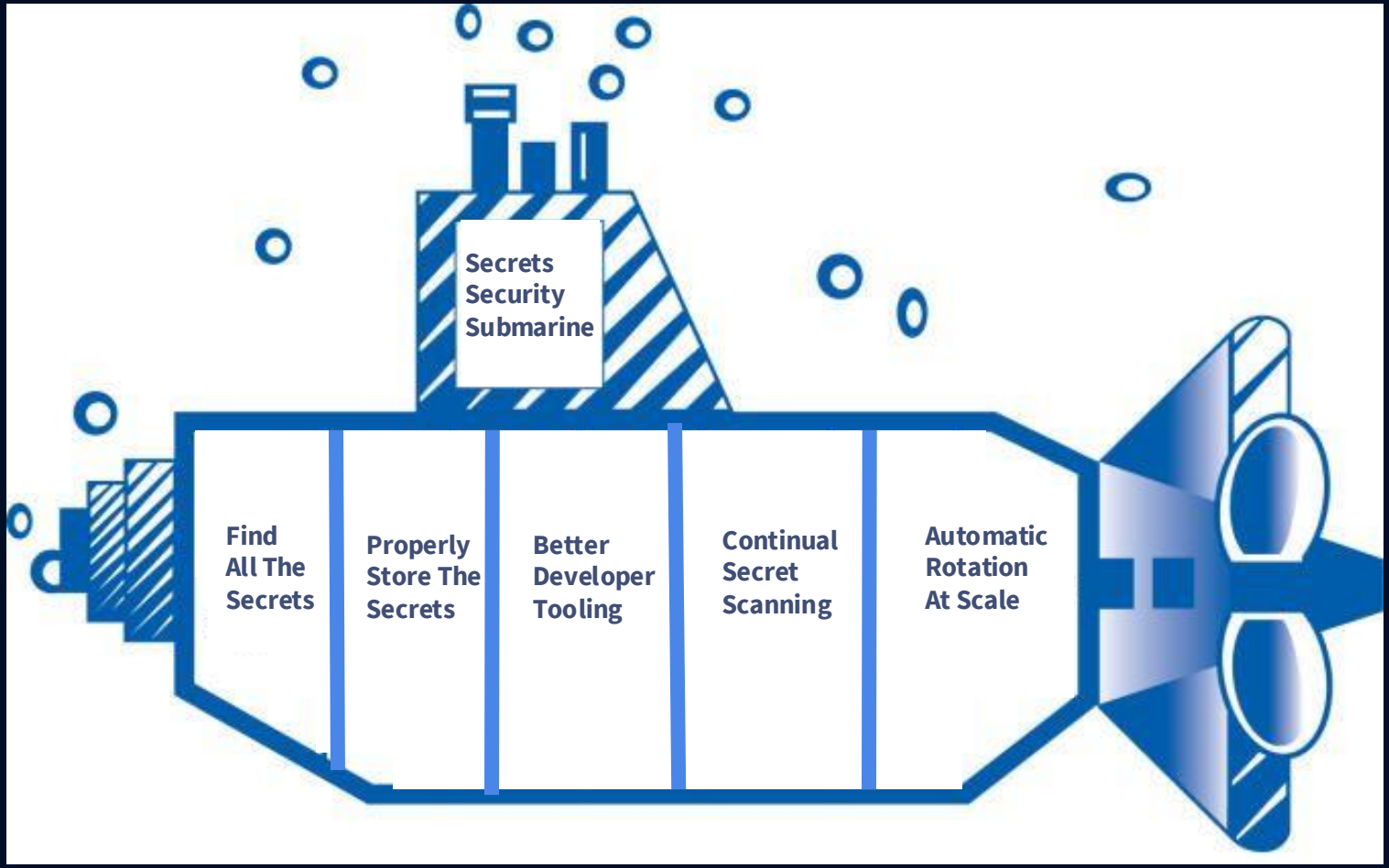
Kubelet generates **short-lived, automatically rotated** tokens for service accounts if the credential provider it communicates with has opted into receiving a service account token for image pulls. These tokens conform to OIDC ID token semantics and are provided to the credential provider as part of the `CredentialProviderRequest`. The credential provider can then use this token to authenticate with an external service.


@mdwayne-real.bsky.social

**But you can’t get to approaches like SPIFFE or WIMSE at scale until…**

**You first understand what machine identities and secrets you even have**


@mdwayne-real.bsky.social





@mdwayne-real.bsky.social


@mdwayne-real.bsky.social

# Hi. I'm Dwayne.

Dwayne McDaniel
Senior Developer Advocate
dwayne.mcdaniel@gitguardian.com

{

"Hometown" : "Chicago",

- "Mission" : "Help people figure stuff out",
- "Developer-advocate-since" : "2014",
- "Host" : "The Security Repo Podcast",
- "Socials" : {

"mcdwayne@mastodon.social",

"www.linkedin.com/in/dwaynemcdaniel" },

"Other-interests" : {"crochet", "karaoke",

"rock and roll concerts", "music in general"}

}

@mdwayne-real.bsky.social
@mdwayne-real.bsky.social

# About GitGuardian

**GitGuardian is an enterprise platform helping teams solve Non-Human Identity security crisis**

- NHI Governance
- Public Monitoring of GitHub
- Secrets Detection and Remediation Platform
- Developer Tooling for Prevention
- Honeytokens

https://tinyurl.com/dwayne-corn

# Secrets Security End-To-End


@mdwayne-real.bsky.social
@mdwayne-real.bsky.social