Developing secure software with GitHub

Laura Kokkarinen & Thomas Vochten

Who are we?

Technology Evangelist, MVP

Thomas Vochten

De Cronos Groep, Belgium

@ThomasVochten

Software Architect, MVP

Laura Kokkarinen

Sulava, Finland

@LauraKokkarinen

Agenda

Protect against what?

Secure Software Development Lifecycle

Automating code security checks
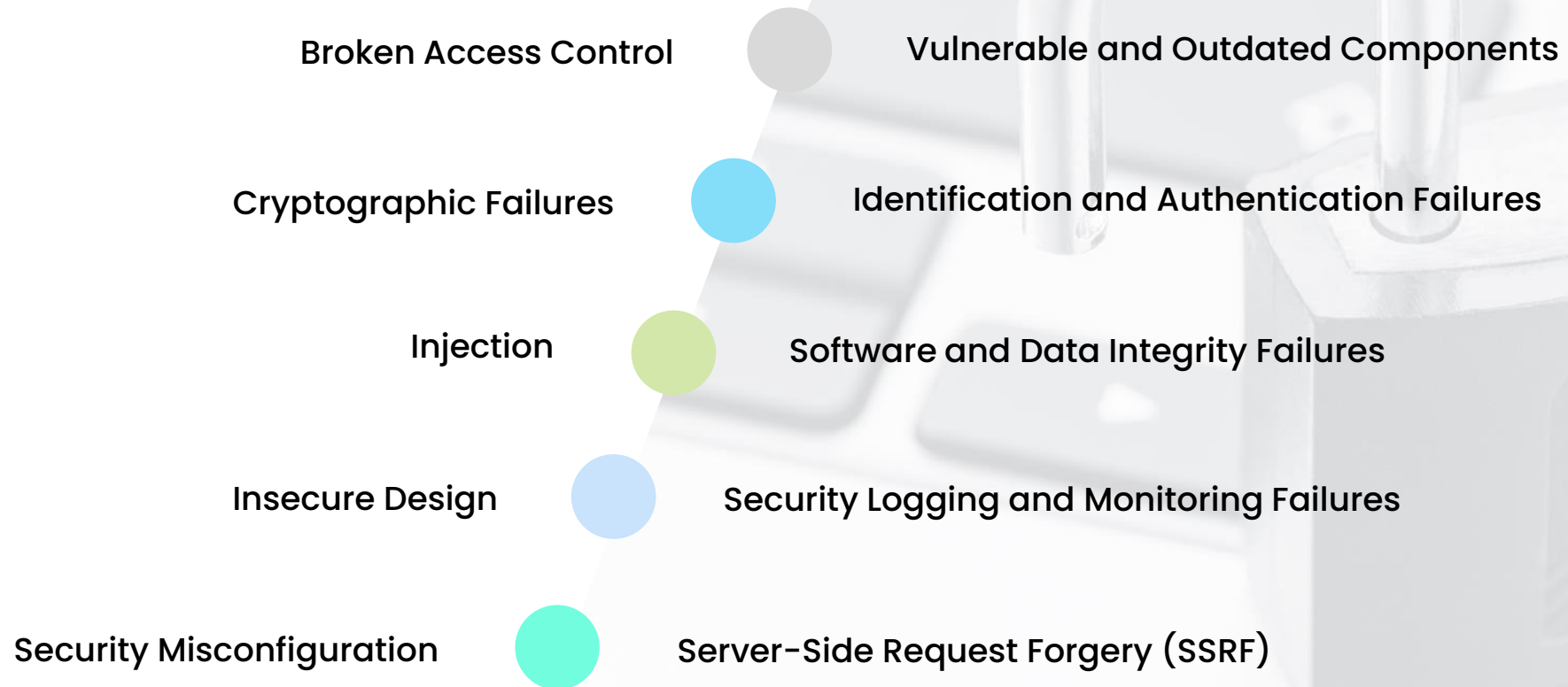
Protecting source code and repositories

Defender for Cloud integration

What about Azure DevOps?

OWASP Top 10

Broken Access Control

Vulnerable and Outdated Components

Cryptographic Failures

Identification and Authentication Failures

Injection

Software and Data Integrity Failures

Insecure Design

Security Logging and Monitoring Failures

Security Misconfiguration

Server-Side Request Forgery (SSRF)

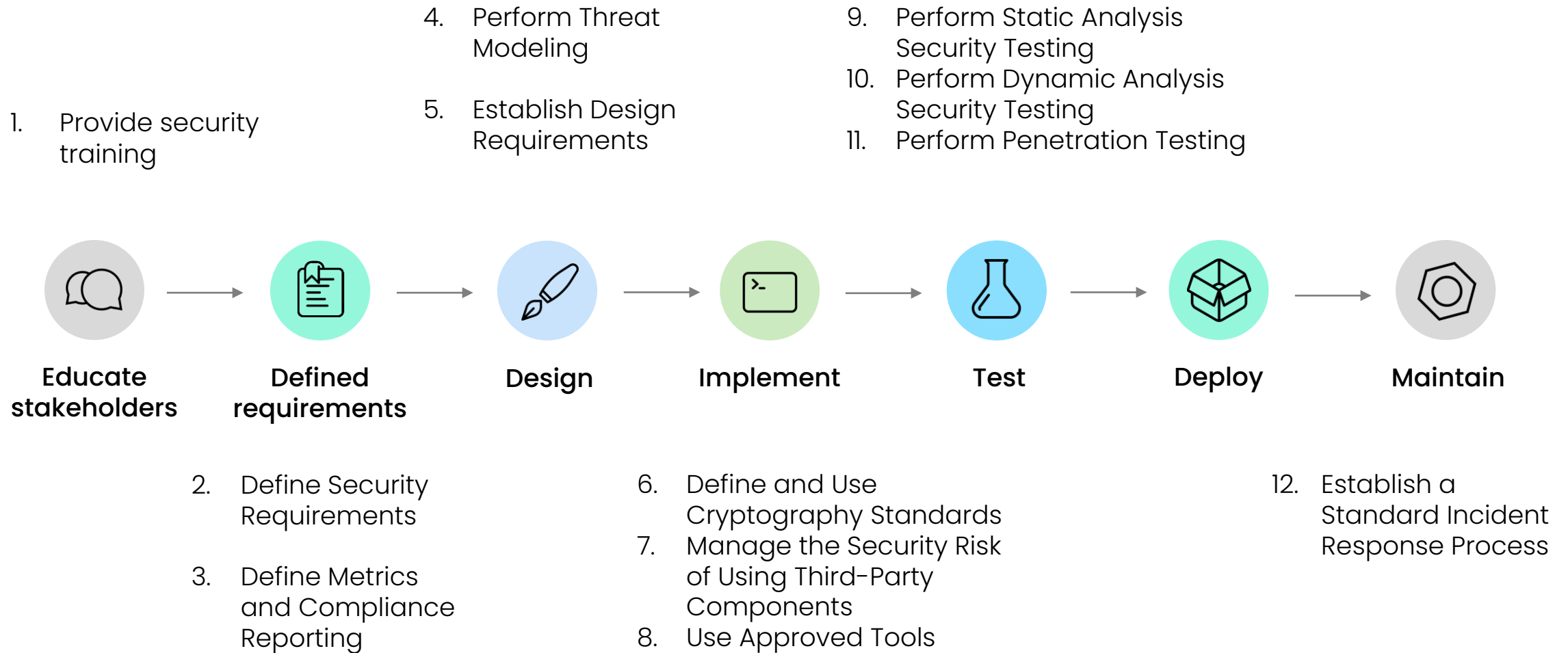# Secure Software Development Lifecycle (SDLC)

- Software development process that prioritizes security at every stage

- Framework

Educate stakeholders → **Define requirements** → **Design** → **Implement** → **Test** → **Deploy** → **Maintain**
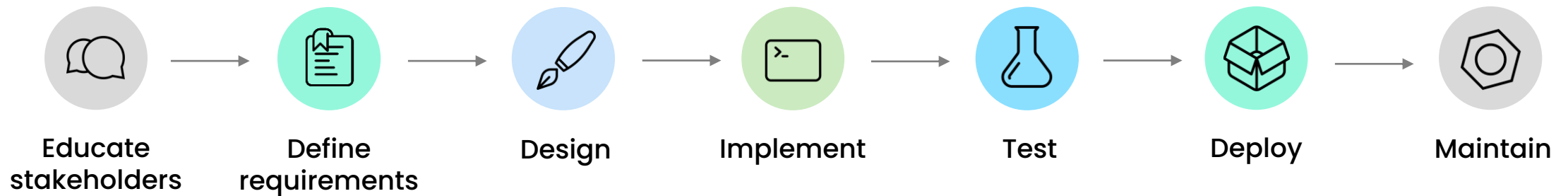
Microsoft Security Development Lifecycle (SDL)

1. Provide security training

4. Perform Threat Modeling

5. Establish Design Requirements

9. Perform Static Analysis Security Testing
10. Perform Dynamic Analysis Security Testing
11. Perform Penetration Testing

Educate stakeholders → **Defined requirements** → **Design** → **Implement** → **Test** → **Deploy** → **Maintain**

2. Define Security Requirements

3. Define Metrics and Compliance Reporting

6. Define and Use Cryptography Standards
7. Manage the Security Risk of Using Third-Party Components
8. Use Approved Tools

12. Establish a Standard Incident Response Process

Why is it important and what will you gain?

- Attacks targeting apps become ever more prevalent and sophisticated

- Initially requires additional resources but has ROI over long term

Reduced number of security vulnerabilities

Improved overall quality

Compliance with regulations and standards

Reduced development costs

Competitive advantage through reputation and customer trust

Improved customer satisfaction

When is it feasible to implement?

- Viable for all projects

- Ideally adopted from the very beginning

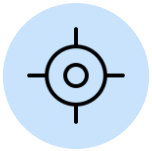**Business, enterprise or infrastructure environment**

Sensitive information

Communicates over a network

What steps can be automated?

- Often some manual work is combined with automation

 Threat modeling

 Code analysis

 Security testing

 Configuration scanning

 Continuous integration and deployment

 Incident response

UP NEXT

How can GitHub help us automate steps during SDLC?

Automated testing is key

• Provide a baseline quality check
• Avoid common mistakes or anti-patterns
• Awareness of vulnerabilities
• Consistency through CI/CD integration

Never trust

Always verify

Manual tests are still valuable

• Reviewing pull requests
• Validation of design decisions
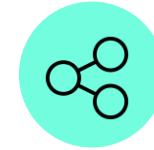• Applying common sense

GitHub to the rescue!

Secret
Scanning

Code
Scanning

Dependency
Scanning

Secret Scanning – What?

- Prevents exposing tokens, private keys or other secrets

- Scanning across all branches and git history

- Looks for patterns provided by the vendors

- Reported as alerts in the repository's Security tab

- Free for public repositories

Secret Scanning Patterns

Azure	azure_active_directory_application_secret
Azure	azure_batch_key_identifiable
Azure	azure_cache_for_redis_access_key
Azure	azure_cosmosdb_key_identifiable
Azure	azure_devops_personal_access_token
Azure	azure_function_key
Azure	azure_ml_web_service_classic_identifiable_key
Azure	azure_sas_token
Azure	azure_search_admin_key
Azure	azure_search_query_key
Azure	azure_management_certificate
Azure	azure_sql_connection_string
Azure	azure_storage_account_key
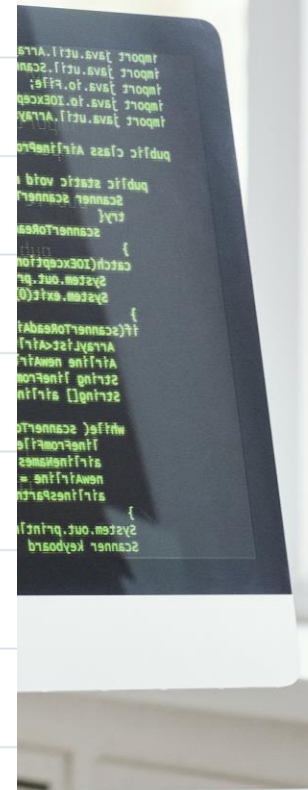
Secret Scanning – Partner Program

- Contributed by partners such as Microsoft

- Offers additional intelligence and automatic revoking of secrets (!)
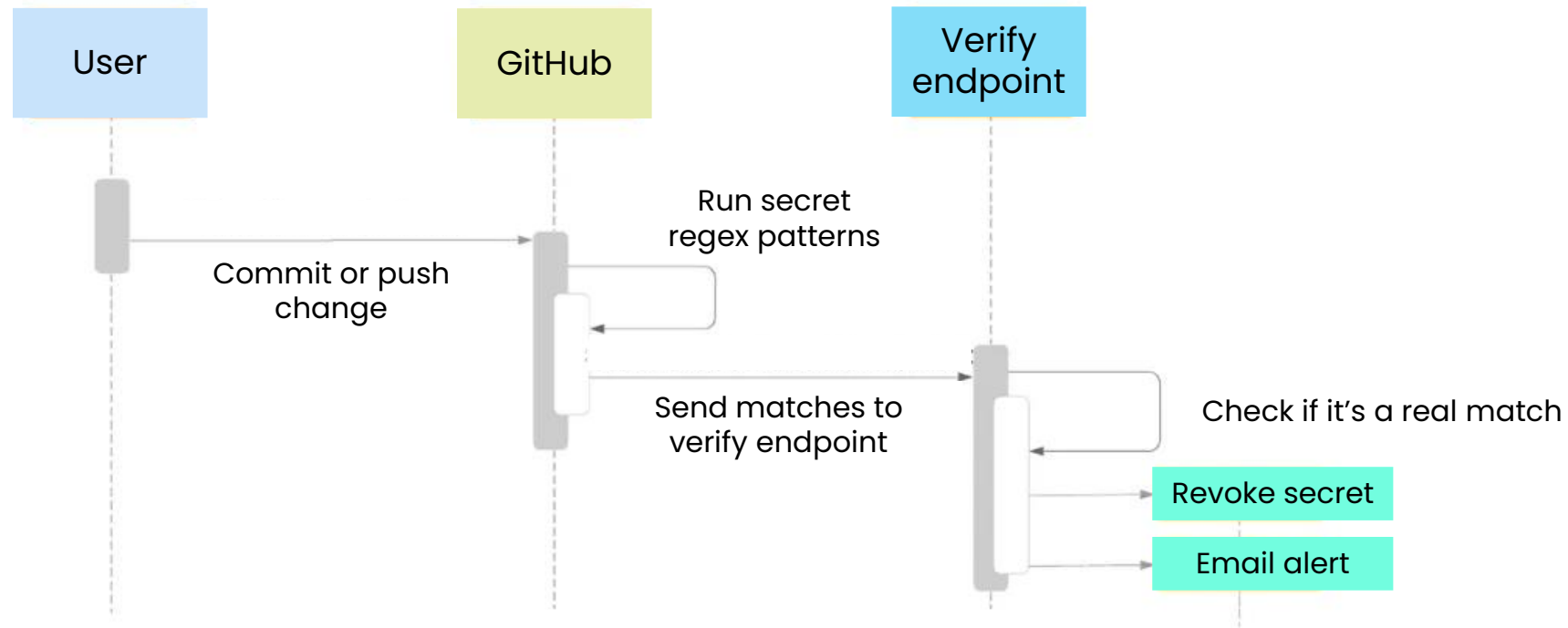
Secret Scanning - Configuration

🔵 **Scope**

Enable secret scanning
for a single repository,
or on an account level

🟢 **Exclusions**

You can exclude
directories through a
.github/secret_scanning
.yml file

Secret scanning
Receive alerts on GitHub for detected secrets, keys, or other tokens.

☑ **Automatically enable for new public repositories**
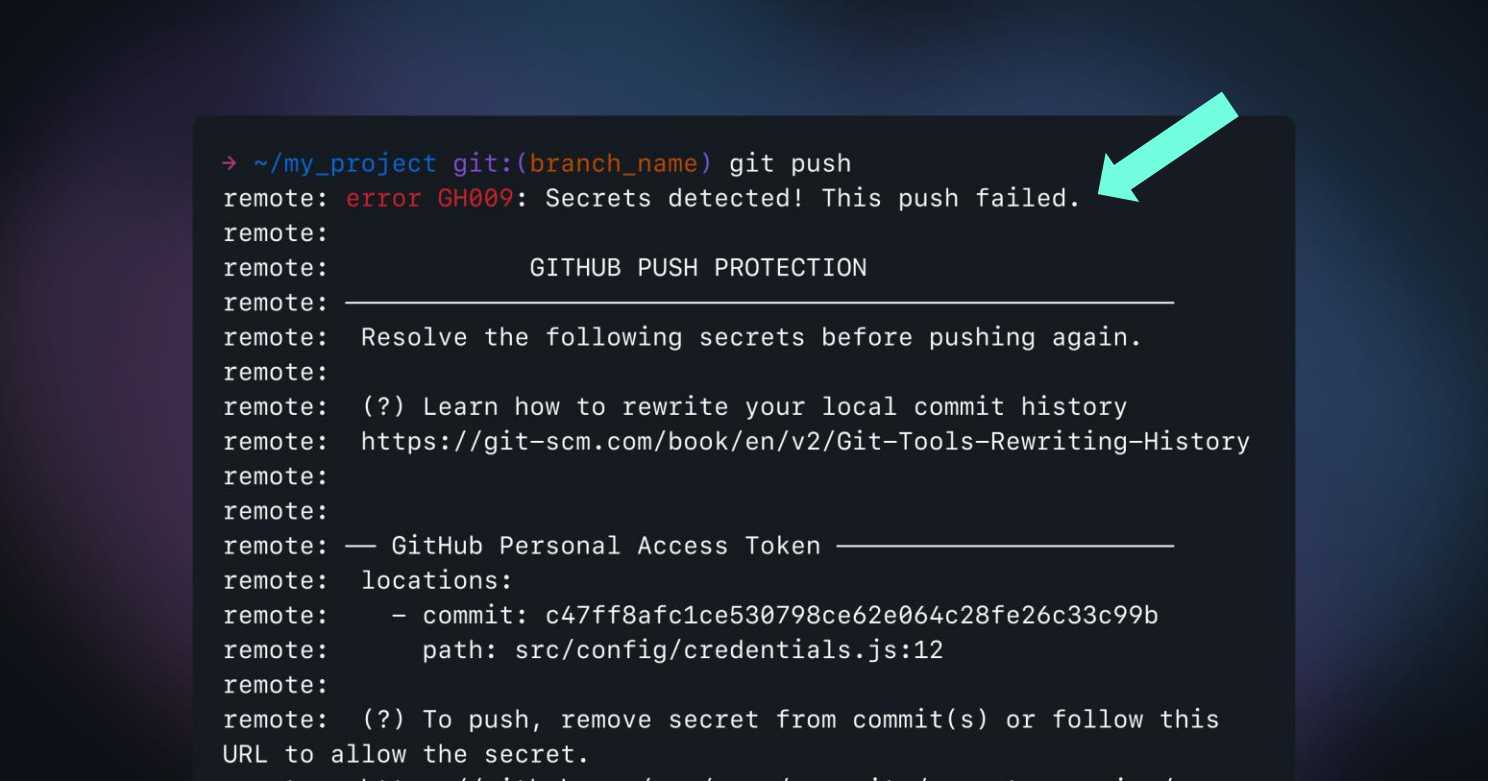
Disable all   Enable all

```
paths-ignore:
    - "foo/bar/*.js"
```

Secret Scanning – Push Protection

● Free for public repositories


```
→ ~/my_project git:(branch_name) git push
remote: error GH009: Secrets detected! This push failed.
remote:
remote:              GITHUB PUSH PROTECTION
remote: ————————————————————————————————————————————————————————
remote:  Resolve the following secrets before pushing again.
remote:
remote:  (?) Learn how to rewrite your local commit history
remote:  https://git-scm.com/book/en/v2/Git-Tools-Rewriting-History
remote:
remote:
remote: — GitHub Personal Access Token ————————————————————————
remote:  locations:
remote:    - commit: c47ff8afc1ce530798ce62e064c28fe26c33c99b
remote:      path: src/config/credentials.js:12
remote:
remote:  (?) To push, remove secret from commit(s) or follow this
URL to allow the secret.
```

Secret Scanning – Alerting

● Alerts are sent to

- Contributor who committed the secret
- Repository administrators
- Organization owners

● Only admins can dismiss secret scanning alerts
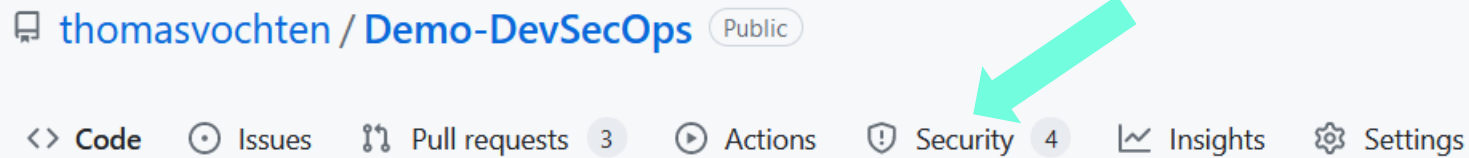
Secret Scanning – Alerting

Azure Storage Account Access Key

Close as ▾

Open **GitHub** detected a secret 1 minute ago (Beta) Give us feedback

Possibly active secret

+eAwigsvbNxLb7ZcOxvgrHjuXE0oliYfNQb21rpeuFkENfUoJ/abfDw1eOaSwIGOR1CIHWYGIXZx+AStJb4kqA==

Remediation steps
Follow the steps below before you close this alert.

1 Rotate the secret if it's in use to prevent breaking workflows.

2 Revoke this Azure Storage Account Access Key through Azure to prevent unauthorized access. Learn more about Azure tokens.

3 Check security logs for potential breaches.

4 Close the alert as revoked.

Detected in 1 location

⌄ Program.cs ⧉   ···

```
33          }
34
35          // Deliberately hardcoded secrets
36          BlobServiceClient blobServiceClient = new BlobServiceClient("DefaultEndpointsProtocol=https;AccountName=thisivulnerablestorage;AccountKey=+eAwigsvbNxLb7ZcOxvgrHjuXE0oliYfNQb21rpeuFkENfU
37          blobServiceClient.GetBlobContainerClient("demo-devsecops").GetBlobClient("demo-devsecops.txt").DownloadTo("demo-devsecops.txt");
38
39
```

On a roll! 0e7a41d   1 minute ago

🛡 ○ **GitHub** opened this alert 1 minute ago

Secret Scanning – What now?

- How do I remove secrets from git history?

- Examples:

```
$ bfg --delete-files YOUR-FILE-WITH-SENSITIVE-DATA
$ bfg --replace-text passwords.txt

$ git push --force
```

Code Scanning – What?

- Identify and fix security vulnerabilities and coding errors

- Scheduled scans or trigger on certain events (push)

- Creates an alert (and closes it automatically)

- Uses GitHub Actions

- Free for public repositories

Code Scanning – How?

- CodeQL

- Or through a 3rd party tool that supports the Static Analysis Results Interchange Format (SARIF)

CodeQL Analysis
By GitHub

Security analysis from GitHub for C, C++, C#, Go, Java, JavaScript, TypeScript, Python, Ruby and Kotlin developers.

Configure   Code scanning ●

zScan
By Zimperium

The zimperium-zscan GitHub action scans your mobile app binary (iOS or Android) and identifies security, privacy, and compliance-related vulnerabilities.

Configure   Code scanning ●

NowSecure
By NowSecure

The NowSecure Action delivers fast, accurate, automated security analysis of iOS and Android apps coded in any language

Configure   Code scanning ●

Bandit Scan
By abirismyname

Bandit is free software designed to find common security issues in Python code, maintained by PyCQA

Configure   Code scanning ●

Datree
By Datree

Detect misconfigurations in your Kubernetes manifests and present them in Github code scanning

Configure   Code scanning ●

Fortify on Demand Scan
By Micro Focus

Integrate Fortify's comprehensive static code analysis (SAST) for 27+ languages into your DevSecOps workflows to build secure software faster.

Configure   Code scanning ●

Snyk Infrastructure as Code
By Snyk

Detect vulnerabilities in your infrastructure as code files and surface the issues in GitHub code scanning.

Configure   Code scanning ●

Detekt
By Detekt

Static code analysis for Kotlin

Configure   Code scanning ●

Red Hat CodeReady Dependency Analytics
By Red Hat

Scan your project's dependencies with CodeReady Dependency Analytics.

Configure   Code scanning ●

OSSAR
By GitHub

Run multiple open source security static analysis tools without the added complexity with OSSAR (Open Source Static Analysis Runner).

Configure   Code scanning ●

Semgrep
By Returntocorp

Continuously run Semgrep to find bugs and enforce secure code standards. Start with 1k+ community rules or write your own in a few minutes.

Configure   Code scanning ●

Veracode Static Analysis
By Veracode

Get fast feedback on flaws with Veracode Static Analysis and the pipeline scan. Break the build based on flaw severity and CWE category.

Configure   Code scanning ●

Trivy
By Aqua Security

Scan Docker container images for vulnerabilities in OS packages and language dependencies with Trivy from Aqua Security.

Configure   Code scanning ●

Frogbot Scan Pull Request
By JFrog

Automatically scans new pull requests for security vulnerabilities. Uses JFrog Xray to scan the project. Included as part of JFrog's free subscription.

Configure   Code scanning ●

EthicalCheck
By APIsec

EthicalCheck provides the industry's only free & automated API security testing service that uncovers security vulnerabilities using OWASP API list.

Configure   Code scanning ●

lintr
By GitHub Actions

lintr provides static code analysis for R.

Configure   Code scanning ●

CodeScan
By CodeScan Enterprises, LLC

CodeScan allows for better visibility on your code quality checks based on your custom rulesets.

Configure   Code scanning ●

SOOS DAST Scan
By SOOS

SOOS DAST is the easy-to-integrate no-limit web vulnerability scanner. Integrate SOOS DAST with your CI pipeline to find vulnerabilities by scanning a web app or APIs.

Configure   Code scanning ●

Code Scanning – CodeQL

CodeQL is a code analysis engine to automate security checks.

Database

You generate a CodeQL database to represent your codebase

Queries

You run CodeQL queries on that database to identify problems in the codebase

Alerts

The query results are shown as code scanning alerts in GitHub when you use CodeQL with code scanning.

Code Scanning – CodeQL

- CodeQL code scanning automatically detects code written in the supported languages:

Language	Variants	Compilers	Extensions
C/C++	C89, C99, C11, C18, C++98, C++03, C++11, C++14, C++17, C++20 [1]	Clang (and clang-cl [2]) extensions (up to Clang 12.0), GNU extensions (up to GCC 11.1), Microsoft extensions (up to VS 2019), Arm Compiler 5 [3]	`.cpp, .c++, .cxx, .hpp, .hh, .h++, .hxx, .c, .cc, .h`
C#	C# up to 10.0	Microsoft Visual Studio up to 2019 with .NET up to 4.8, .NET Core up to 3.1, .NET 5, .NET 6	`.sln, .csproj, .cs, .cshtml, .xaml`
Go (aka Golang)	Go up to 1.20	Go 1.11 or more recent	`.go`
Java	Java 7 to 20 [4]	javac (OpenJDK and Oracle JDK), Eclipse compiler for Java (ECJ) [5]	`.java`
Kotlin [6]	Kotlin 1.5.0 to 1.8.20	kotlinc	`.kt`
JavaScript	ECMAScript 2022 or lower	Not applicable	`.js, .jsx, .mjs, .es, .es6, .htm, .html, .xhtm, .xhtml, .vue, .hbs, .ejs, .njk, .json, .yaml, .yml, .raml, .xml` [7]
Python [8]	2.7, 3.5, 3.6, 3.7, 3.8, 3.9, 3.10, 3.11	Not applicable	`.py`
Ruby [9]	up to 3.2	Not applicable	`.rb, .erb, .gemspec, Gemfile`
TypeScript [10]	2.6-4.9	Standard TypeScript compiler	`.ts, .tsx, .mts, .cts`

Code Scanning – CodeQL

Code Scanning – Alerting

Code scanning

✅ All tools are working as expected   🔧 Tool status 2   + Add tool

🔍 is:open branch:main

☐ 🛡 **3 Open**   ✓ 1 Closed   Tool ▾   Branch ▾   Rule ▾   Severity ▾   Sort ▾

☐ 🛡 **Azure Storage Account Keys should not be disclosed** (Critical)   `main`
     #2 opened 1 hour ago • Detected by **SonarCloud** in Program.cs:34

☐ 🛡 **Constant condition** ⚠ Warning   `main`
     #4 opened 1 hour ago • Detected by **CodeQL** in Program.cs:29

☐ 🛡 **Useless assignment to local variable** ⚠ Warning   `main`
     #3 opened 1 hour ago • Detected by **CodeQL** in Program.cs:34

Code Scanning

Pull request integration

Dependabot – What?

Alerts
GitHub creates alerts when a vulnerable dependency or malware is detected

Scans
Scans when a new advisory is published or when you change the dependencies of your project

Fixes
Dependabot can fix vulnerable dependencies for you by raising pull requests with security updates.

Updates
You can use Dependabot to keep the packages you use updated to the latest versions.

PR integration
Can also integrate as a pull request (PR) check

Free
Free for all repositories

Dependabot – Alerting

.NET Information Disclosure Vulnerability #1

🛡 Open · Opened last month on **System.Data.SqlClient** (NuGet) · DevSecOpsDemo.csproj

Dismiss alert ▾

⑂ Bump System.Data.SqlClient from 4.8.0 to 4.8.5

Merging this pull request would fix 1 Dependabot alert on System.Data.SqlClient in DevSecOpsDemo.csproj.

🔒 Review security update

Package	Affected versions	Patched version
⬡ **System.Data.SqlClient** (NuGet)	<= 4.8.4	4.8.5 ⎘

Microsoft is releasing this security advisory to provide information about a vulnerability in .NET, .NET Core and .NET Framework's System.Data.SqlClient and Microsoft.Data.SqlClient NuGet Packages.

A vulnerability exists in System.Data.SqlClient and Microsoft.Data.SqlClient libraries where a timeout occurring under high load can cause incorrect data to be returned as the result of an asynchronously executed query.

Mitigation factors

If you are not talking to Microsoft SQL Server from your application you are not affected by this vulnerability.

How do I know if I am affected?

.NET has two types of dependencies: direct and transitive. Direct dependencies are dependencies where you specifically add a package to your project, transitive dependencies occur when you add a package to your project that in turn relies on another package.

Severity

Moderate 5.8 / 10

CVSS base metrics

Attack vector	Adjacent
Attack complexity	High
Privileges required	Low
User interaction	None
Scope	Changed
Confidentiality	High
Integrity	None
Availability	None

CVSS:3.1/AV:A/AC:H/PR:L/UI:N/S:C/C:H/I:N/A:N

Tags

Direct dependency   Patch available

Weaknesses
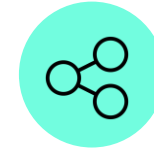
No CWEs

CVE ID

CVE-2022-41064

GitHub to the rescue!

Secret Scanning

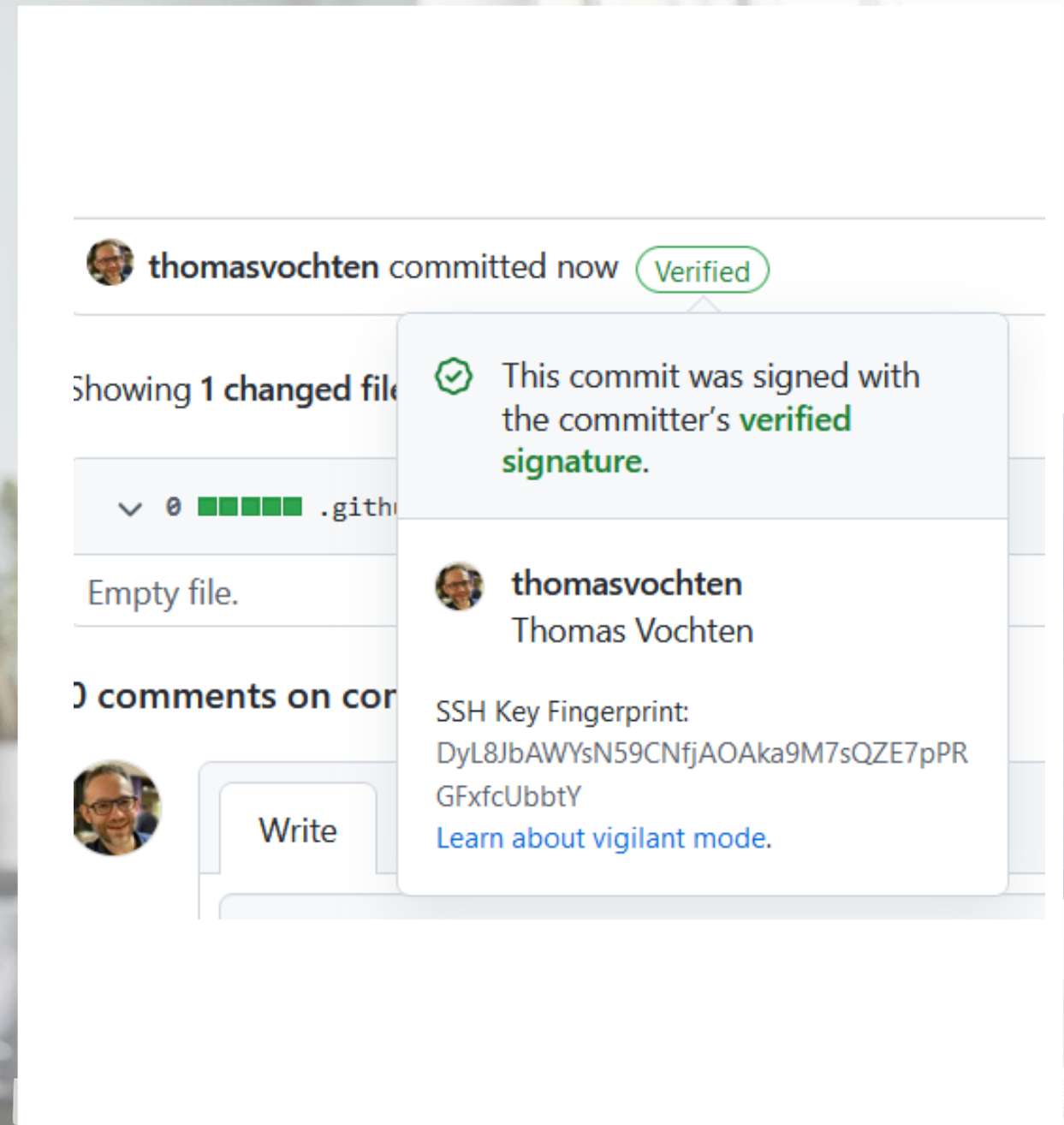**Code Scanning**

Dependency Scanning

GitHub Account Security

- Username/Password with MFA

  - Time-based one-time password (TOTP)
  - GitHub Mobile
  - Security Key
  - Passkey

- Personal Access Token (PAT)

- SSH Key

Verified Commits

Using GPG, SSH, or S/MIME, you can sign tags and commits locally. These tags or commits are marked as verified on GitHub so other people can be confident that the changes come from a trusted source.

Defender for Cloud integration

Defender for Cloud integration
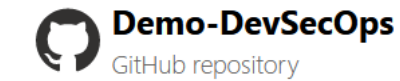
Code repositories should have secret scanning findings resolved ···

⟜ Open query

Severity
| High

Freshness interval
🕐 60 Min

Tactics and techniques
🔒 **Initial Access** +1

⌄ Description

Secrets have been found in code repositories. This should be remediated immediately to prevent a security breach. Secrets found in repositories can be leaked or discovered by adversaries, leading to compromise of an application or service. For Azure DevOps, the Microsoft Security DevOps cre
Therefore, results may not reflect the complete status of secrets in your repositories.

⌄ Remediation steps

Manual remediation:

To resolve discovered secrets:

1. Review the secret found by the scan.
2. Click on each finding to view details.
3. Invalidate the secret, tokens, and/or passwords.
4. Navigate to the repository using the Html URL or Build URL.
5. Refactor your code to remove the secret.
6. Check-in the remediated project.
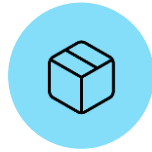7. Review scan results for the repository to verify the secret no longer exists.

⌄ Security checks

Findings

🔍 Search to filter items...

ID	Security check	Category	Severity
2da861e0-2b22-a473-cc3c-36646028e104	azure_storage_account_key	Secrets	🔴 High

What about Azure DevOps?

Similar features?

Something missing?

Something more?

Secret scanning

- No native GA feature available (yet)

- The next best thing: **Microsoft Defender for DevOps (preview)**

Requires more effort to setup
- Microsoft Defender for DevOps
- Microsoft Security DevOps task
- Every repo/branch

Result display
- Azure Portal
- Build results
- Pull requests (if enabled)

Alerting
- Recommendation on Microsoft Defender for Cloud
- Action via Azure Logic App
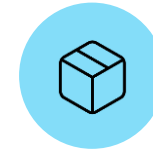
Preview
- False positives
- Future cost?

git-secrets
Git pre-commit hook

Future
GitHub Advanced Security for Azure DevOps (preview)

Code scanning

No native GA feature available (yet)

The next best thing

- Use a third-party solution, e.g., SonarCloud
- Free for OSS, 10$+ /month for private
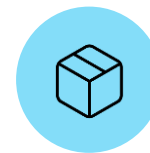- Requires build pipeline configuration

Future options

- GitHub Advanced Security for Azure DevOps (preview)
- Access needs to be requested from Microsoft
- 49$/month per active commiter

Dependency scanning

No native GA feature available (yet)

The next best thing

- Use a third-party solution, e.g., Snyk
- Requires build pipeline configuration
- Reports inheritance
- Config when to fail
- Sends email alerts on new vulns
- Free

Future options

- GitHub Advanced Security for Azure DevOps (preview)
- Access needs to be requested from Microsoft
- 49$/month per active commiter

Authentication

Azure AD login

- SSO
- MFA, conditional access etc.
- For an enterprise

PAT & SSH

Azure AD identities

- Service principals & managed identities
- Preview feature
- Alternative for PATs

Verified commits

- Git feature

- Not similarly supported on Azure DevOps

- Possible to validate commit signatures in a build pipeline
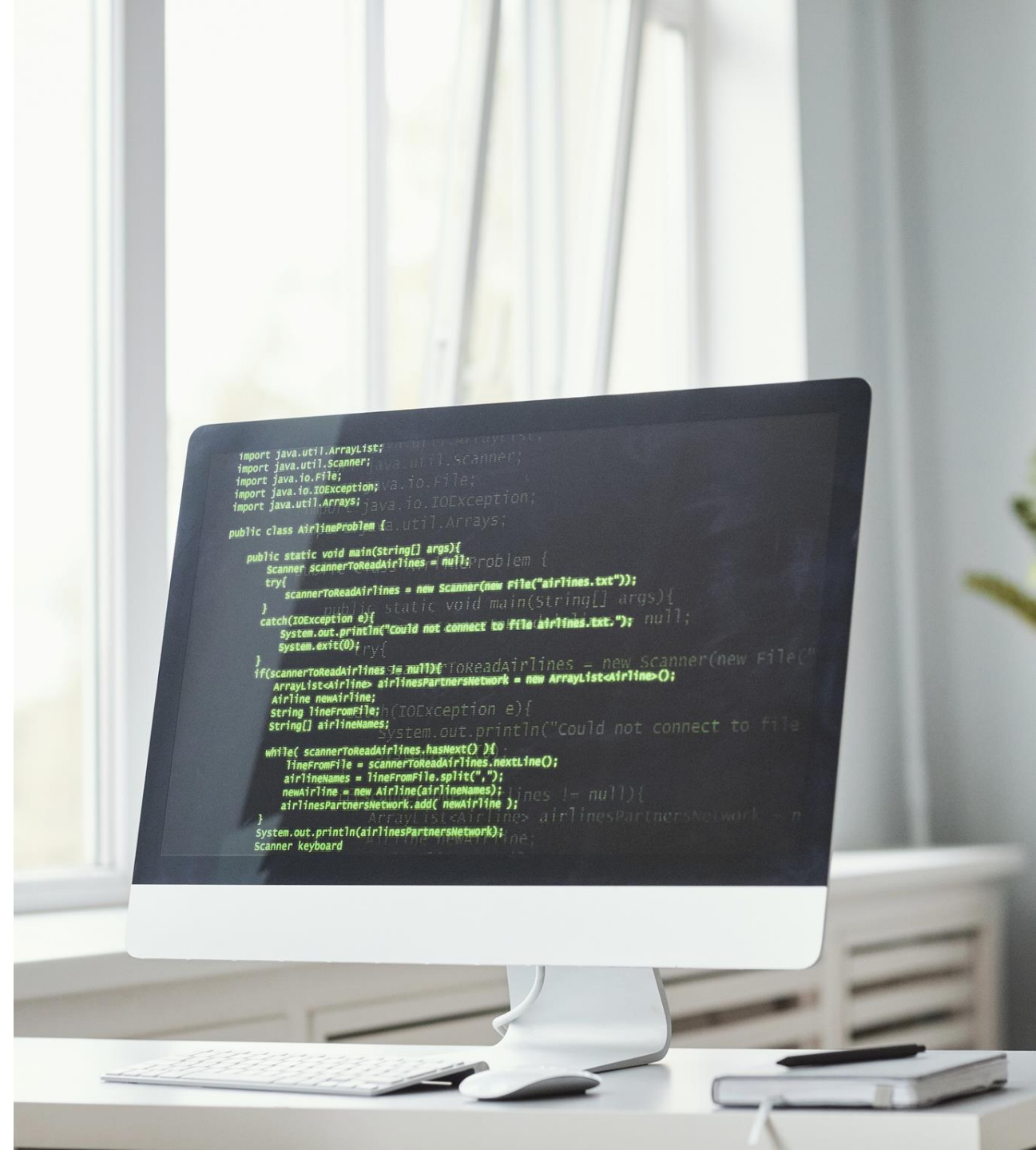
Azure DevOps vs GitHub

- GitHub for open source, Azure DevOps for the enterprise

- What about GitHub Enterprise? What is keeping people on Azure DevOps?

Project management and collaboration

- Planning
- Collaboration
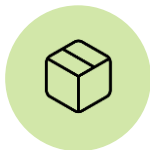- Analytics and reports

Tracing and auditing

- Work item links

More granular access control management

- Levels
- Features
- Permissions

Flexible licensing

- Stakeholder
- Basic
- Basic + Test Plans

More mature pipelines

- Microsoft
- Release pipeline features

Other features

- Test Plans
- Printable Wikis
- Etc.

Future

- What is the point of having two products for the same purpose?

- Migration from Azure DevOps to GitHub Enterprise – eventually

Recap

Protect against what?

Secure Software Development Lifecycle

Automating code security checks

Protecting source code and repositories

Defender for Cloud integration

What about Azure DevOps?

Questions?