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?

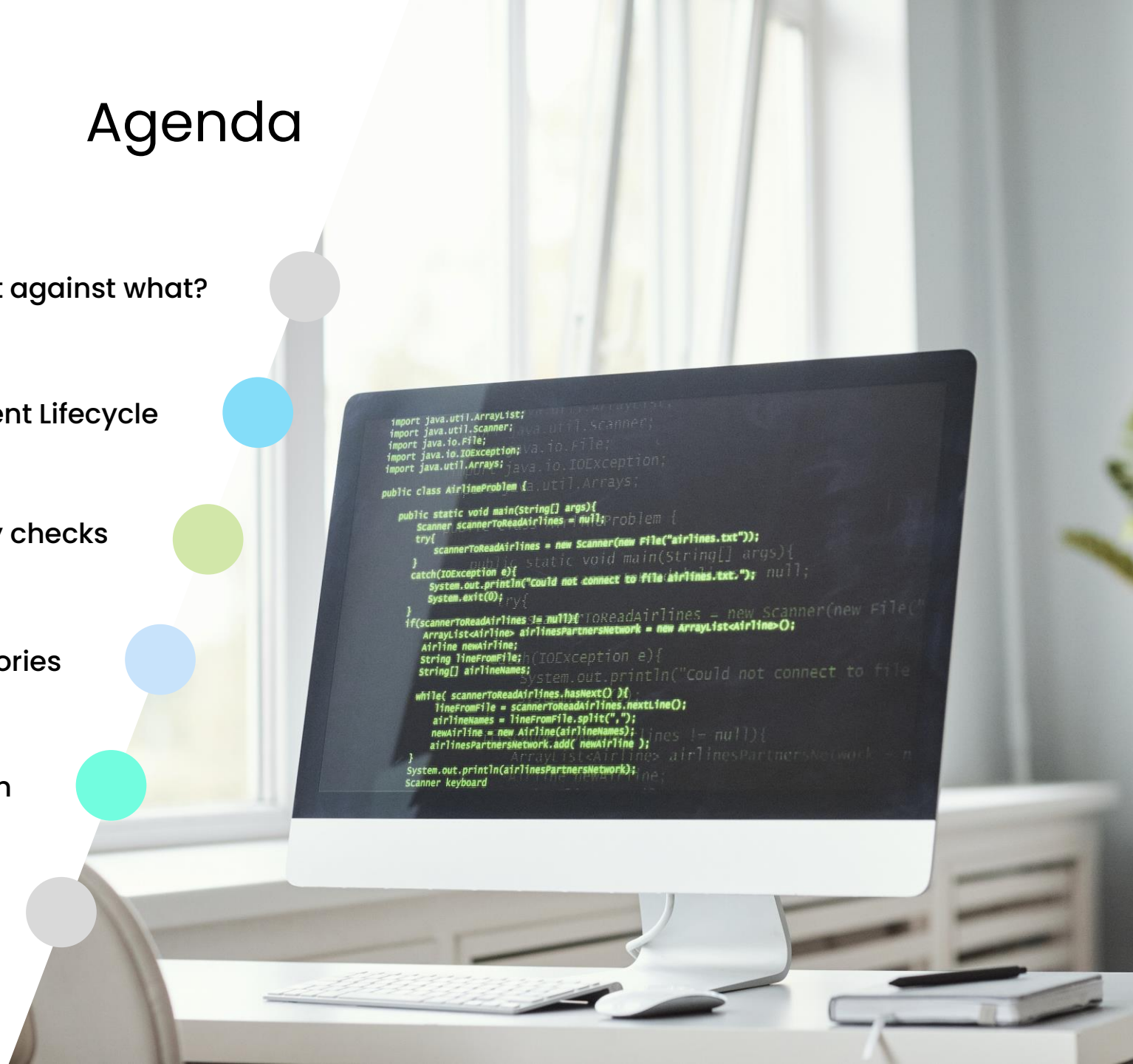
Secure Software Development Lifecycle

Automating code security checks

Protecting source code and repositories

Defender for Cloud integration

What about Azure DevOps?

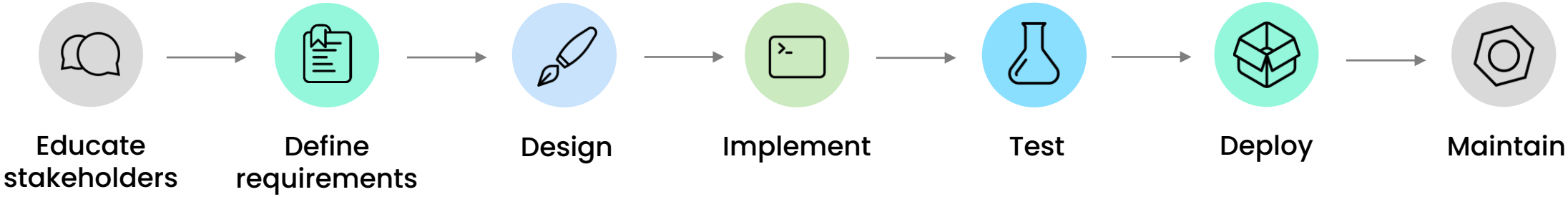


OWASP Top 10

- 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- Vulnerable and Outdated Components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



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?

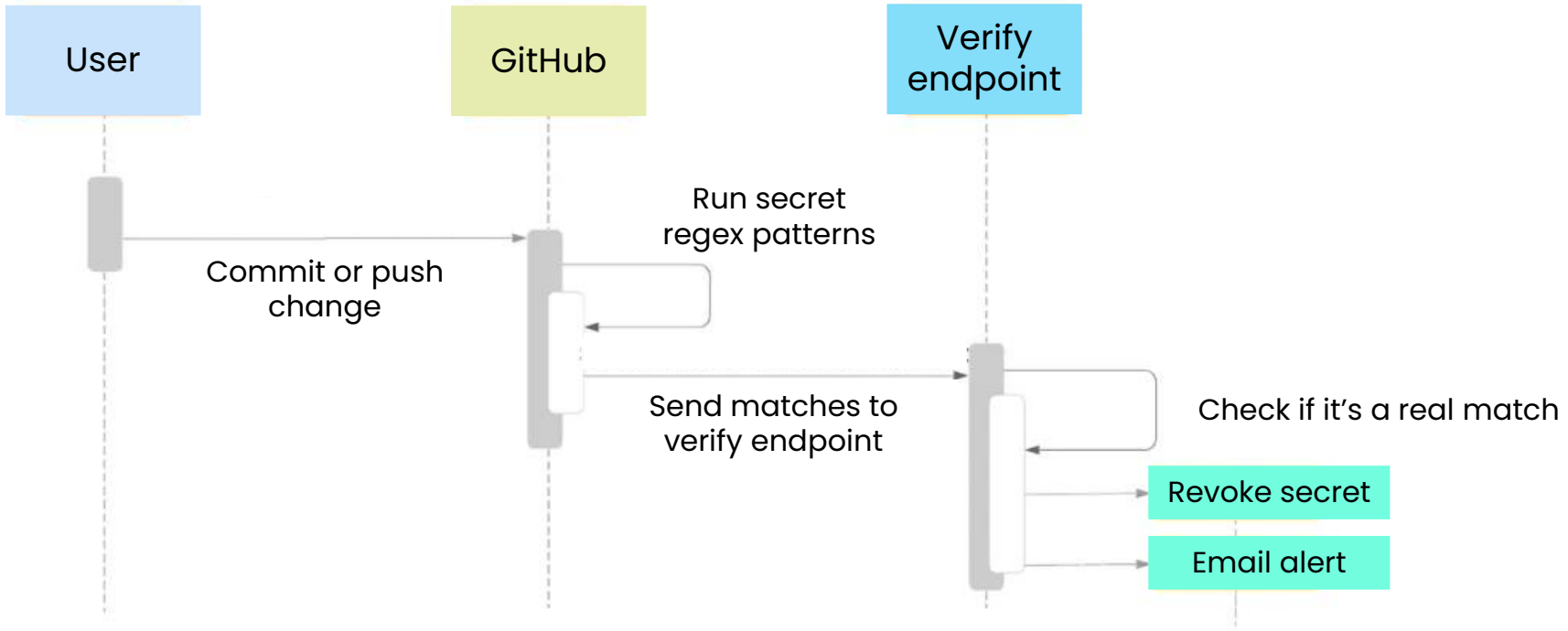
- Prevents exposing tokens, private keys or other secrets
- Scanning across all branches and git history
- Looks for patterns provided by the vendors
- Reported as alerts in the repository's Security tab
- Free for public repositories

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	✓	✓	✗

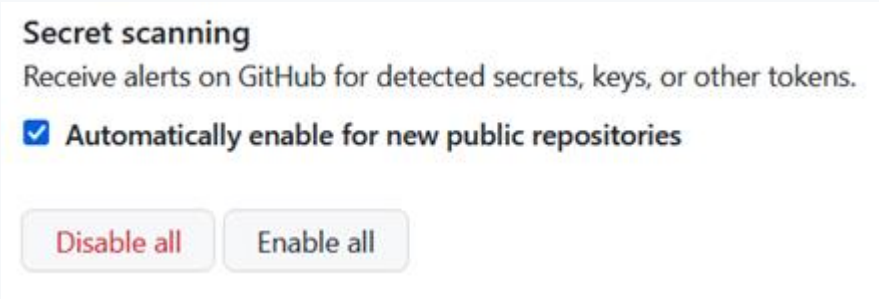
Secret Scanning – Partner Program

- 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

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

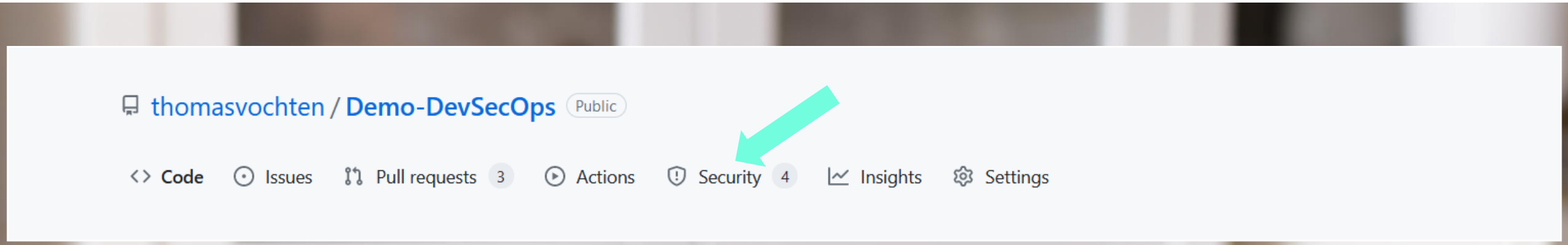
Secret Scanning – Push Protection NEW

- 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:

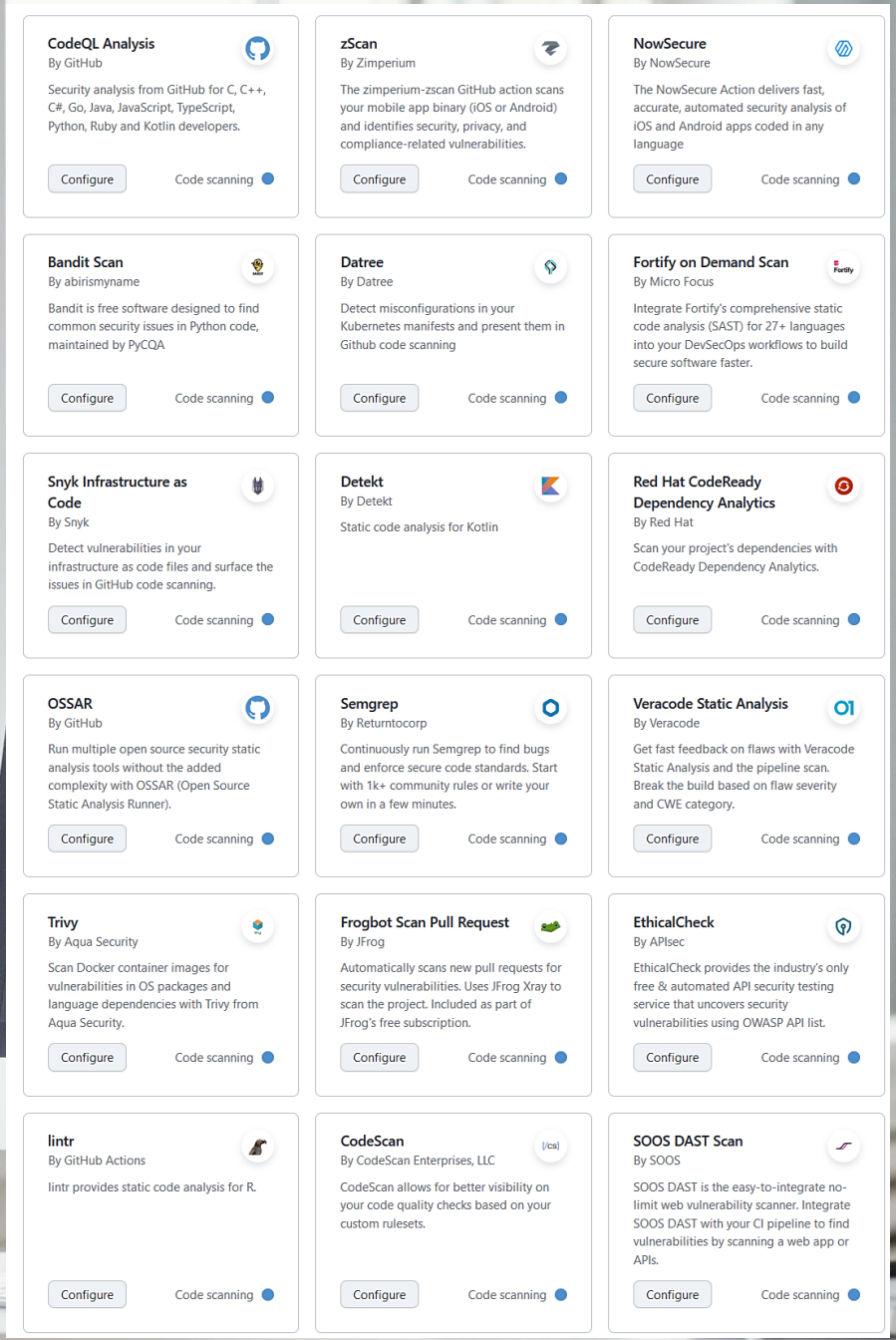
```
$ bfg --delete-files YOUR-FILE-WITH-SENSITIVE-DATA
$ bfg --replace-text passwords.txt

$ 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)



Code Scanning – CodeQL

- CodeQL is the code analysis engine developed by GitHub to automate security checks.
- CodeQL treats code like data, allowing you to find potential vulnerabilities in your code with greater confidence than traditional static analyzers.

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

Automatically detect common vulnerabilities and coding errors.

Tools

CodeQL analysis
Identify vulnerabilities and errors with CodeQL for eligible repositories.

Last scan 1 hour ago | Set up | •••

Default
Languages detected in this repository are not compatible with this setup type at this time. Use the advanced setup instead.

Advanced
Customize your CodeQL configuration via a YAML file checked into the repository.

Other tools
Add any third-party code scanning tool.

lows

Protection rules

Pull request check failure
Define which code scanning alert severity should cause a pull request c
applies to analysis results uploaded via the API.

rs

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

#2 opened 1 hour ago • Detected by SonarCloud in Program.cs:34

main

Constant condition Warning

#4 opened 1 hour ago • Detected by CodeQL in Program.cs:29

main

Useless assignment to local variable Warning

#3 opened 1 hour ago • Detected by CodeQL in Program.cs:34

main

Code Scanning

Pull request integration

More silly stuff #2

Open thomasvochten wants to merge 1 commit into `main` from `testbranch`

Conversation 1 | Commits 1 | Checks 3 | Files changed 1

thomasvochten commented last month — Owner

No description provided.

More silly stuff — Verified — 1d6a69a

sonarcloud bot commented last month

SonarCloud Quality Gate failed. Failed

- 0 Bugs
- 0 Vulnerabilities
- 1 Security Hotspot
- 1 Code Smell

0.0% Coverage
0.0% Duplication

Add more commits by pushing to the `testbranch` branch on **thomasvochten/Demo-DevSecOps**.

Review required
At least 1 approving review is required by reviewers with write access. Learn more.

Some checks were not successful — Hide all checks
2 successful and 1 failing checks

- DevSecOpsDemo - Sonarcloud / SonarCloud (pull_request) Successful in 1m — Details
- SonarCloud Code Analysis Failing after 31s — Quality Gate failed — Details
- Code scanning results / SonarCloud Successful in 2s — No new alerts — Details

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

Dismiss alert

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

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

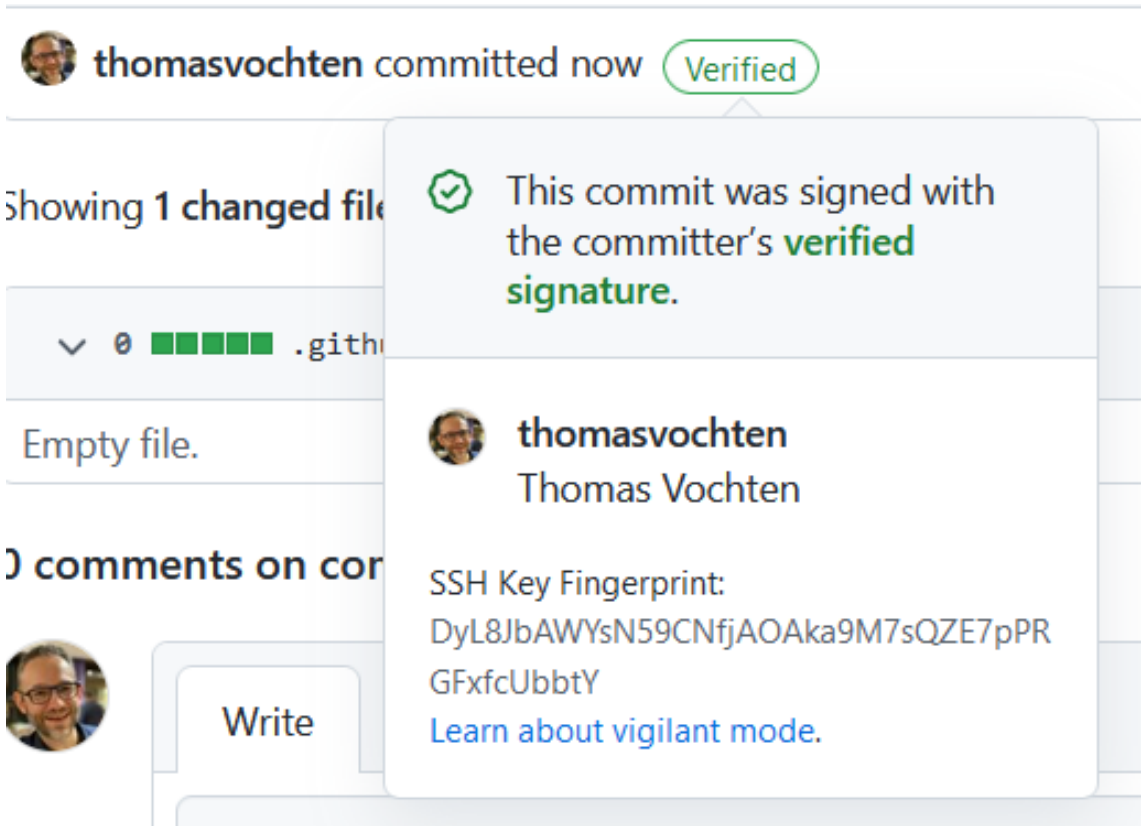
Dependency Scanning

GitHub Account Security

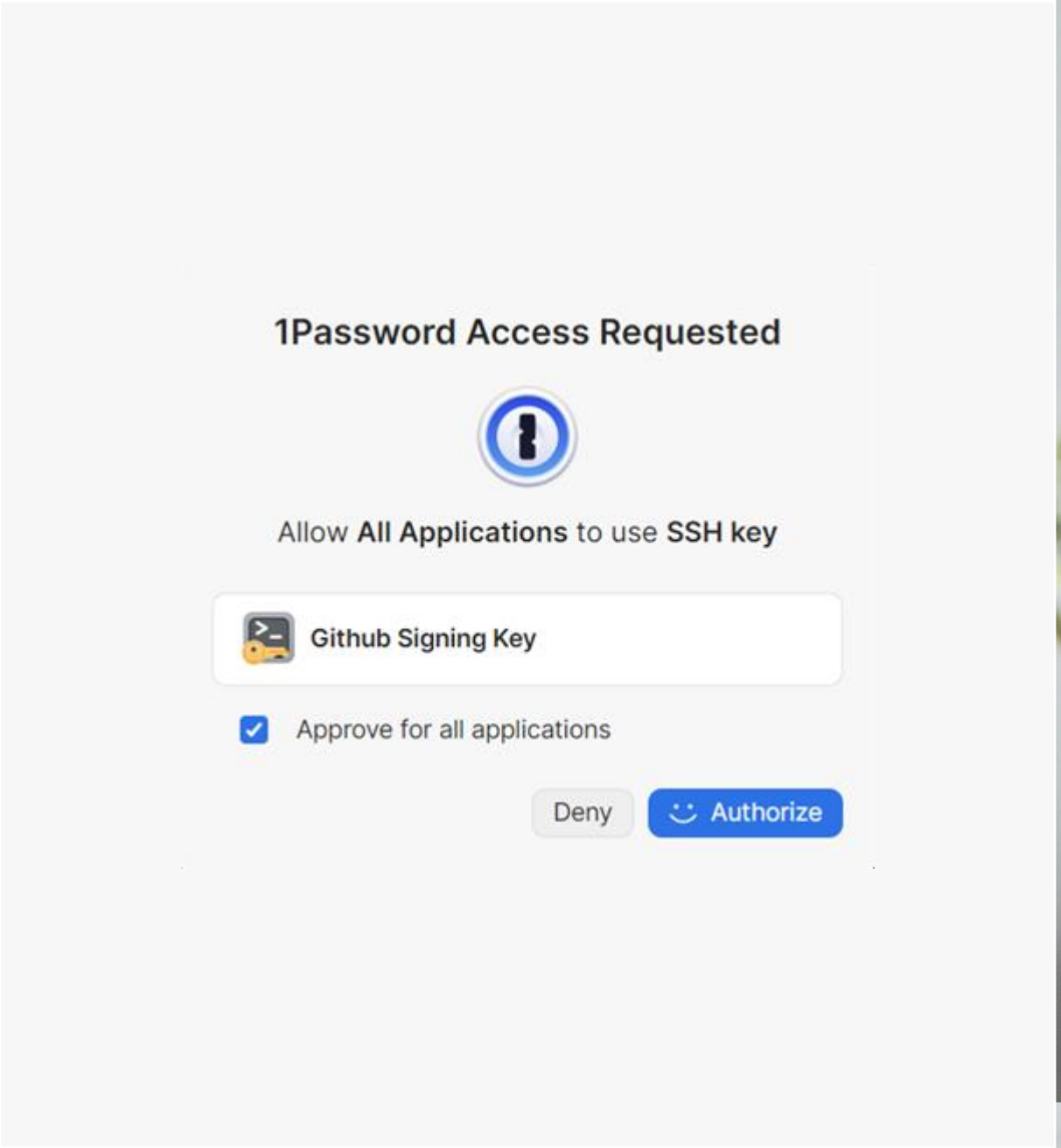
- Username/Password with MFA
  - Time-based one-time password (TOTP)
  - GitHub Mobile
  - Security Key
- Personal Access Token (PAT)
- SSH Key

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.



Verified commits



Defender for Cloud integration

Demo-DevSecOps
GitHub repository

3
Active recommendations

0
Active alerts

Resource information

Subscription
Visual Studio Ultimate wit...

Resource Group
githubdefender

Environment
Azure

Connector
githubdefender

Resource type
GitHub repository

Recommendations | Alerts

Search

More (2)

Severity	Description
High	Code repositories should have secret scanning findings resolved Preview
Medium	GitHub repositories should have code scanning enabled Preview
Medium	GitHub repositories should have Dependabot scanning enabled Preview
Medium	Code repositories should have infrastructure as code scanning findings resolved Preview
Medium	Code repositories should have code scanning findings resolved Preview
Medium	Code repositories should have dependency vulnerability scanning findings resolved Preview

< Previous Page 1 of 1 Next >

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 cr
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)

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)

Authentication

Azure AD login

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

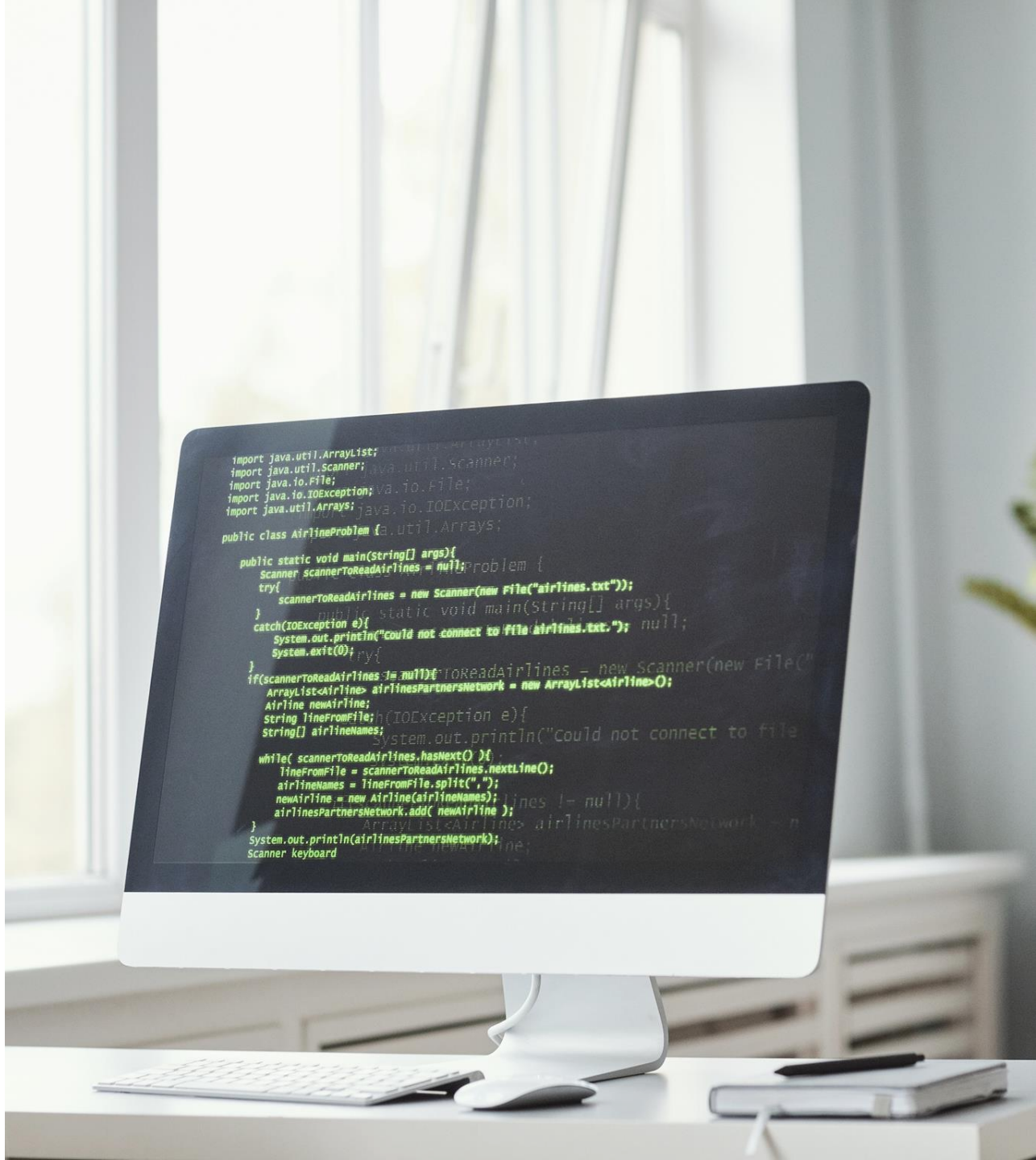
PAT & SSH

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?