

A fluffy white dog is lying on a light-colored wooden desk. In the foreground, a portion of a laptop is visible. The background is blurred, showing what appears to be a living room or office setting.

Developing secure software with GitHub

Laura Kokkarinen & Thomas Vochten

Who are we?



Software Architect, MVP

Laura Kokkarinen

Sulava, Finland

@LauraKokkarinen



Technology Evangelist, MVP

Thomas Vochten

De Cronos Groep, Belgium

@ThomasVochten

Agenda

Protect against what?

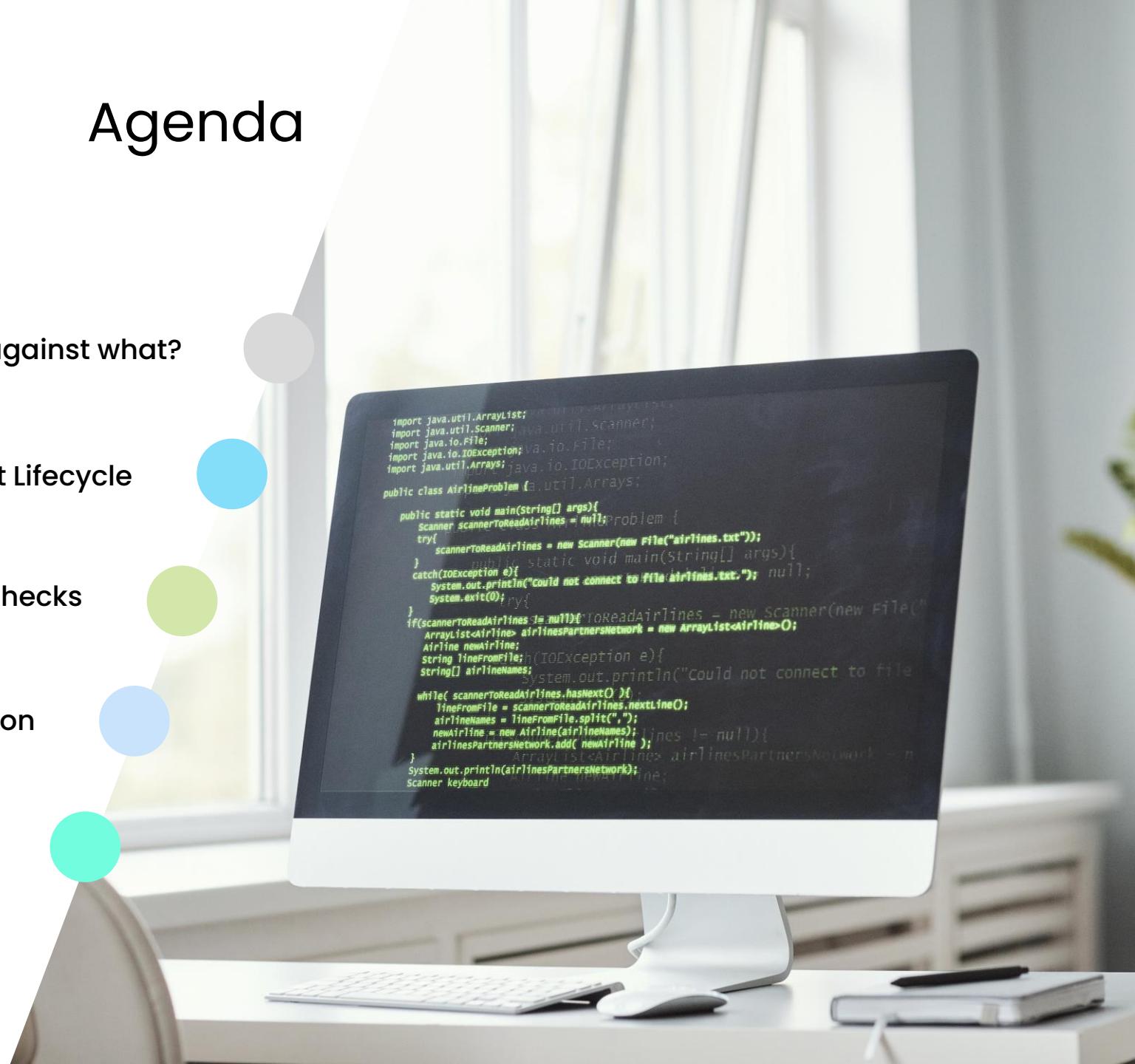
Secure Software Development Lifecycle

Automating code security checks

Defender for Cloud integration

What about Azure DevOps?

```
import java.util.ArrayList;
import java.util.Scanner;
import java.io.File;
import java.io.IOException;
import java.util.Arrays;
public class AirlineProblem {
    public static void main(String[] args){
        Scanner scannerToReadAirlines = null;
        try{
            scannerToReadAirlines = new Scanner(new File("airlines.txt"));
        } catch(IOException e){
            System.out.println("could not connect to file: " + e);
            System.exit(0);
        }
        if(scannerToReadAirlines != null){
            Scanner scannerFromfile = new Scanner(keyboard);
            ArrayList<Airline> airlinesPartnersNetwork = new ArrayList<Airline>();
            Airline newAirline;
            String lineFromFile;
            try{
                while( scannerToReadAirlines.hasNext() ){
                    lineFromFile = scannerToReadAirlines.nextLine();
                    airlineNames = lineFromFile.split(",");
                    newAirline = new Airline(airlineNames);
                    if(airlineNames != null){
                        airlinesPartnersNetwork.add(newAirline );
                    }
                }
            } catch(IOException e){
                System.out.println("could not read file: " + e);
            }
        }
    }
}
```

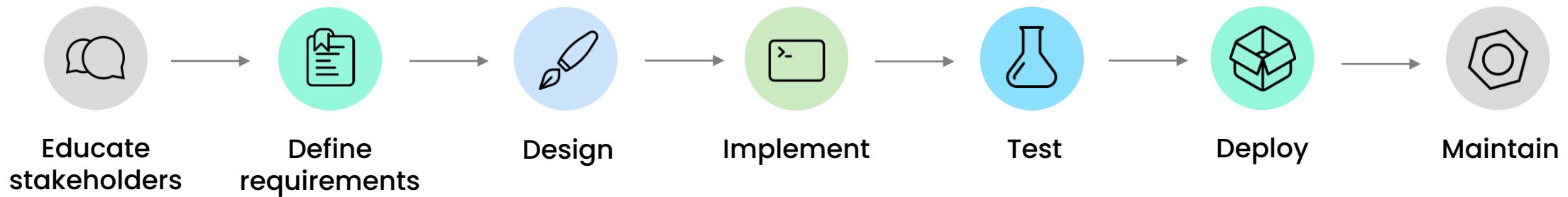


OWASP Top 10

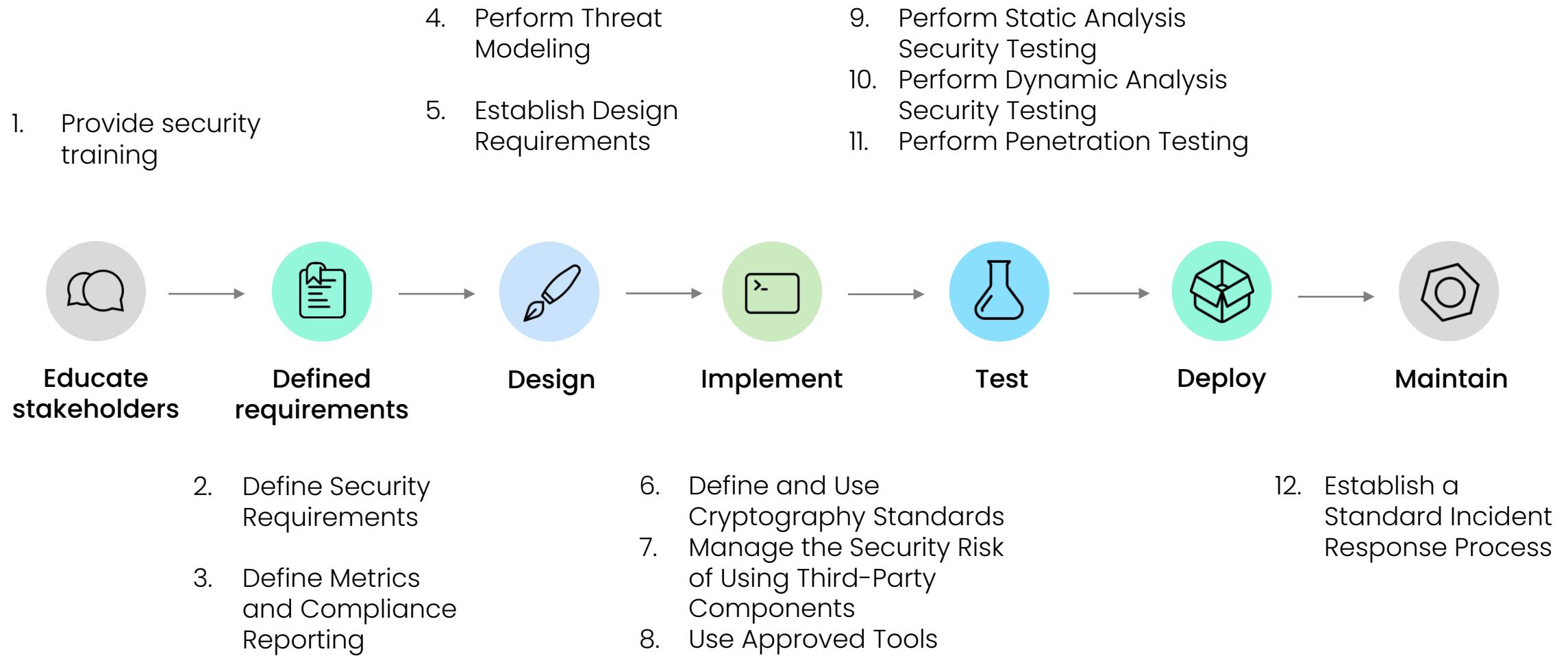


Secure Software Development Lifecycle (ssDLC)

- Software development process that prioritizes security at every stage
- Framework



Microsoft Security Development Lifecycle (SDL)



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?



GitHub to the rescue!



Secret
Scanning



Code
Scanning



Dependency
Scanning

GitHub Advanced Security

- All features are free for public repositories
- Paying option for private repositories,
as add-on for GitHub Enterprise customers



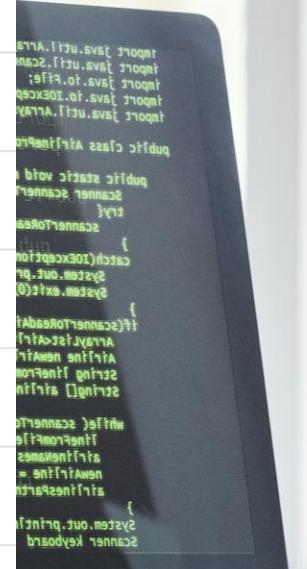
Secret Scanning – What?



- Prevents exposing tokens, private keys or other secrets
- Scanning across all branches and git history
- Looks for patterns provided by the vendors
- Doesn't only scan the code itself
- Reported as alerts in the repository's Security tab

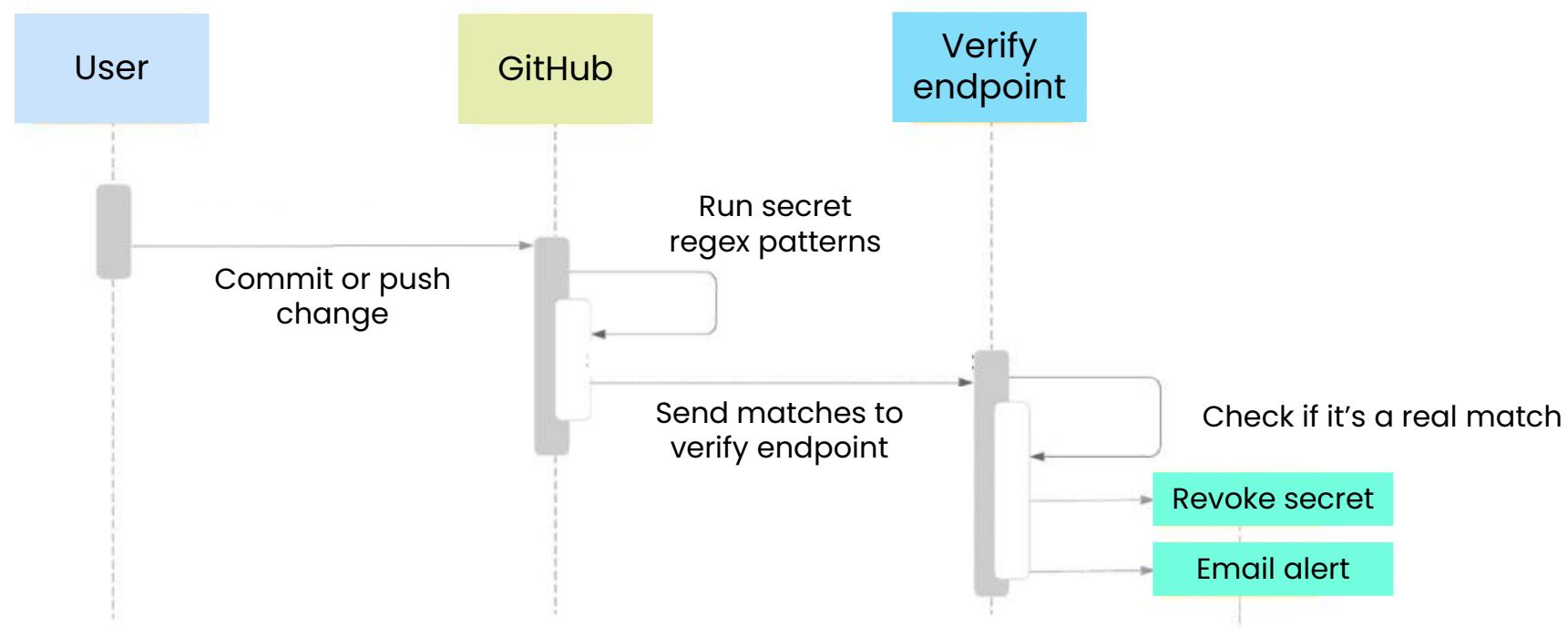
Secret Scanning Patterns

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

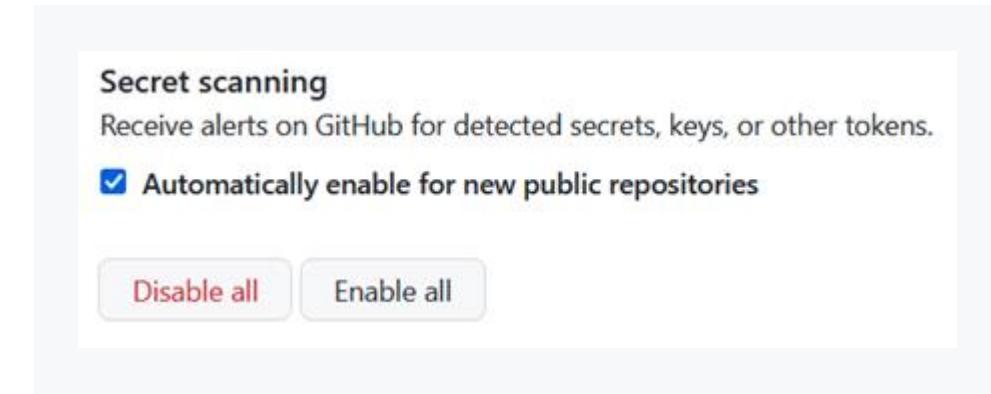
- Contributed by partners such as Microsoft
- Offers additional intelligence and automatic revoking of secrets (!)



Secret Scanning - Configuration

Scope

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



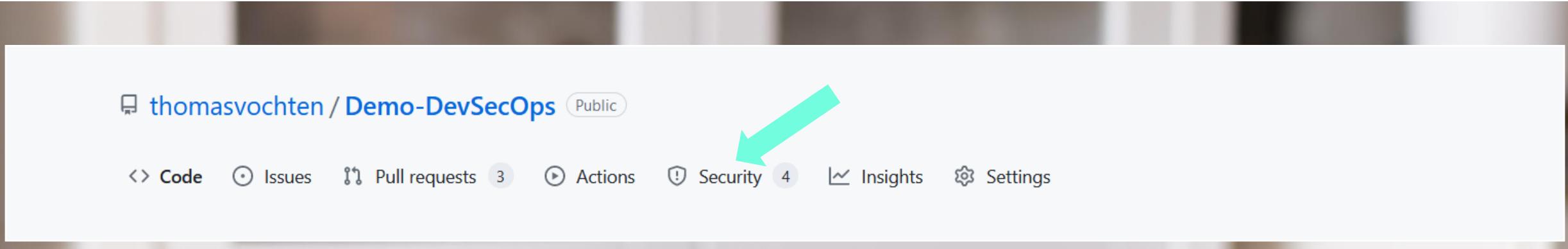
Exclusions

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

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

Secret Scanning – Alerting

- Alerts are sent to
 - Contributor who committed the secret
 - Repository administrators
 - Organization owners
- Only admins can dismiss secret scanning alerts



Secret Scanning – Alerting

Secret scanning alerts / #1

Azure Storage Account Access Key

Open GitHub detected a secret on 21 Apr 2023 Beta Give us feedback

Possibly active secret

```
+eAwigsvbNxLb7Zc0xvgrHjuXE0o1iYfNQb21rpeuFkENfUoJ/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=+eAwigsvbNxLb7Zc0xvgrHjuXE0o1iYfNQb21rpeuFkENfU
37     blobServiceClient.GetBlobContainerClient("demo-devsecops").GetBlobClient("demo-devsecops.txt").DownloadTo("demo-devsecops.txt");
38
39
```

On a roll! 0e7a41d

on 21 Apr 2023

Select a close reason

Revoked
This secret has been revoked

Used in tests
This secret is not in production code

False positive
This alert is not valid

Won't fix
This alert is not relevant

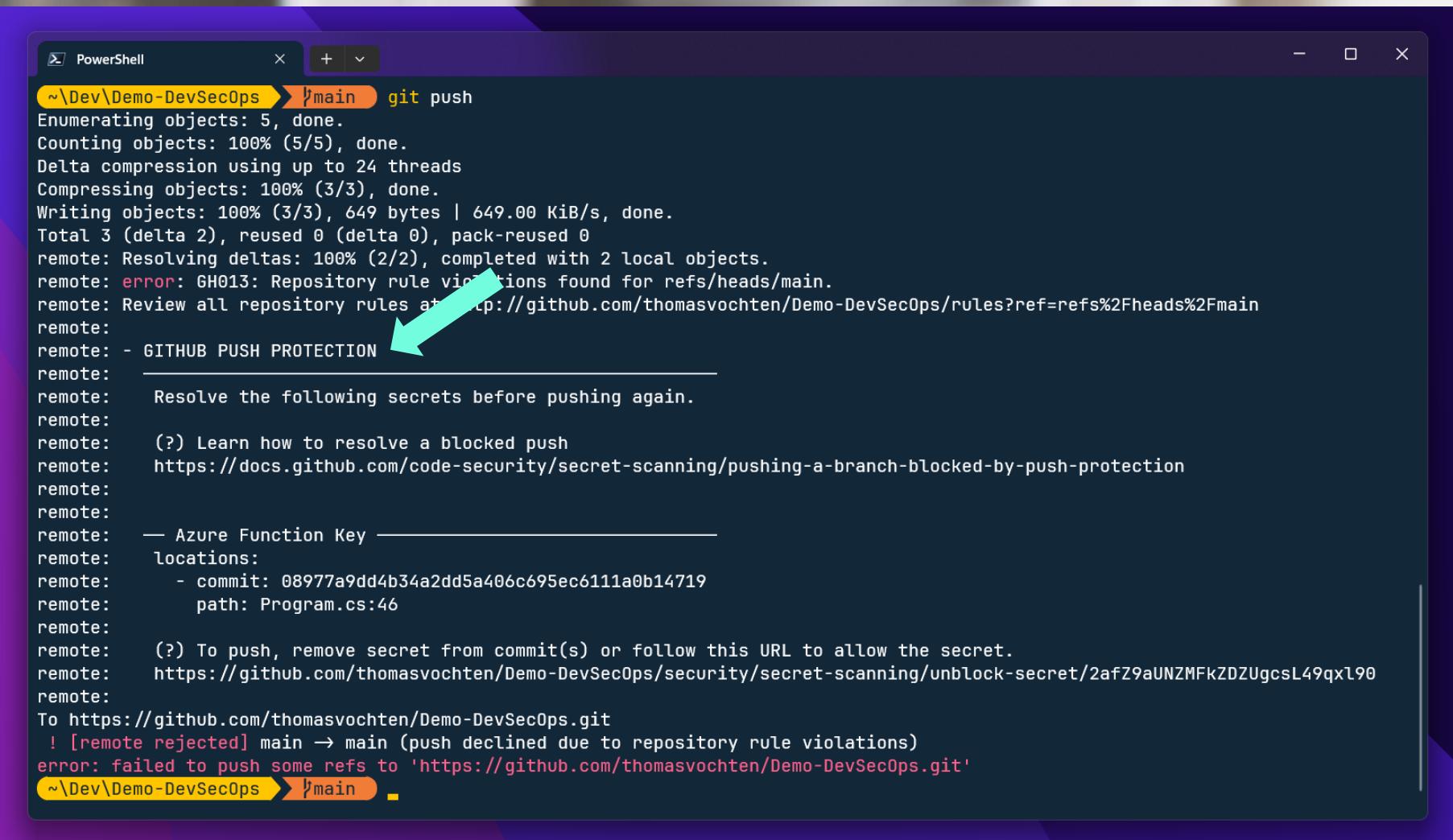
Close as ▾

Secret Scanning – What now?

- How do I remove secrets from git history?

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

Secret Scanning – Push Protection



```
PowerShell ~\Dev\Demo-DevSecOps > \main > git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 24 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 649 bytes | 649.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote: error: GH013: Repository rule violations found for refs/heads/main.
remote: Review all repository rules at https://github.com/thomasvochten/Demo-DevSecOps/rules?ref=refs%2Fheads%2Fmain
remote:
remote: - GITHUB PUSH PROTECTION
remote:
remote: Resolve the following secrets before pushing again.
remote:
remote: (?) Learn how to resolve a blocked push
remote: https://docs.github.com/code-security/secret-scanning/pushing-a-branch-blocked-by-push-protection
remote:
remote:
remote: — Azure Function Key —
remote: locations:
remote:   - commit: 08977a9dd4b34a2dd5a406c695ec6111a0b14719
remote:     path: Program.cs:46
remote:
remote:   (?) To push, remove secret from commit(s) or follow this URL to allow the secret.
remote: https://github.com/thomasvochten/Demo-DevSecOps/security/secret-scanning/unblock-secret/2afZ9aUNZMFkZDZUgcsL49qxI90
remote:
To https://github.com/thomasvochten/Demo-DevSecOps.git
! [remote rejected] main → main (push declined due to repository rule violations)
error: failed to push some refs to 'https://github.com/thomasvochten/Demo-DevSecOps.git'
~\Dev\Demo-DevSecOps > \main >
```

Secret Scanning – Push Protection override

The image shows a GitHub commit page with a large red box highlighting specific interactions related to bypassing push protection.

Left Panel (Commit Summary):

- Push blocked because a secret was detected**
- Secret scanning** found a Azure Function Key secret in your attempted push.
- Allowing this secret risks exposure. Instead, consider [removing the secret from your commit and commit history](#).
- Exposing this secret can allow someone to:
 - Verify the identity of this Azure Function Key secret.
 - Know which resources this secret can access
 - Act on behalf of the secret's owner
 - Push this secret to this repository without being blocked
- It's used in tests**: The secret poses no risk. If anyone finds it, they cannot do any damage or gain access to sensitive information.
- It's a false positive**: The detected string is not a secret.
- I'll fix it later**: The secret is real, I understand the risk, and I will need to revoke it. This will open a security alert and notify admins of this repository.
- Allow me to expose this secret**

Middle Panel (Alert Details):

Azure Function Key
Closed GitHub detected a secret 3 minutes ago Beta Give us feedback

Secret detected
muwk2LUAW75LekiiKMEu326hJzov9xA2JxkWBXVZEV6fAzFurd3D4Q==

Detected in 1 location

```
Program.cs
43     Console.WriteLine("SAS token is: " + sas_token);
44     Console.WriteLine("SAS URL is: " + sas_url);
45
46     var TheKeysToTheKingdom = "muwk2LUAW75LekiiKMEu326hJzov9xA2JxkWBXVZEV6fAzFurd3D4Q==";
47 }
48 }
```

Comments:

- Getting ready [faf1551](#)
- thomasvochten** bypassed push protection 3 minutes ago
- GitHub** opened this alert 3 minutes ago
- thomasvochten** closed this as used in tests 3 minutes ago

Right Panel (Email Summary):

Secret protection bypassed: User pushed a blocked secret in **thomasvochten/Demo-DevSecOps**

Push protection bypassed as a test case

Anyone with read access can view exposed secrets. If the secret is in use, consider rotating then revoking the secret to avoid unauthorized access.

Azure Function Key
Review secret detected in Program.cs#L46 • commit [faf15516](#)

Sign in to GitHub · Notification settings

You are receiving this email because GitHub blocked a commit with a detected secret, and a user bypassed the protection.

Secret Scanning – Generic secret detection

- Only for GitHub Enterprise customers
- AI-powered expansion of secret scanning
- Identifies unstructured secrets in your code using LLM

Secret scanning alerts / #15

Password

Open GitHub detected a secret 20 hours ago

Possibly active secret

```
sup3rcompl6xPa553%dThatN01WouLd5=Gu357!
```

This secret was [detected by AI](#). False positive rates will improve over time.

[Give feedback](#)

Detected in 1 location

src/config.yml

```
1 connection_string: "mongodb://backend-beat-bot.io:27017/back-end"
2 mongo_password: "sup3rcompl6xPa553%dThatN01WouLd5=Gu357!"
```

Update config.yml c6eb2f1 20 hours ago

 GitHub opened this alert 20 hours ago





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

Code Scanning – How?

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

The image shows a grid of 15 GitHub Actions cards, each representing a different tool for code scanning. The tools are arranged in a 5x3 grid. Each card includes the tool name, developer information, a brief description, and two buttons: 'Configure' and 'Code scanning'.

- CodeQL Analysis** By GitHub: Security analysis from GitHub for C, C++, C#, Go, Java, JavaScript, TypeScript, Python, Ruby and Kotlin developers. (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. (Code scanning)
- NowSecure** By NowSecure: The NowSecure Action delivers fast, accurate, automated security analysis of iOS and Android apps coded in any language. (Code scanning)
- Bandit Scan** By abirismyname: Bandit is free software designed to find common security issues in Python code, maintained by PyCQA. (Code scanning)
- Datre** By Datre: Detect misconfigurations in your Kubernetes manifests and present them in GitHub code scanning. (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. (Code scanning)
- Snyk Infrastructure as Code** By Snyk: Detect vulnerabilities in your infrastructure as code files and surface the issues in GitHub code scanning. (Code scanning)
- Detekt** By Detekt: Static code analysis for Kotlin. (Code scanning)
- Red Hat CodeReady Dependency Analytics** By Red Hat: Scan your project's dependencies with CodeReady Dependency Analytics. (Code scanning)
- OSSAR** By GitHub: Run multiple open source security static analysis tools without the added complexity with OSSAR (Open Source Static Analysis Runner). (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. (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. (Code scanning)
- Trivy** By Aqua Security: Scan Docker container images for vulnerabilities in OS packages and language dependencies with Trivy from Aqua Security. (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. (Code scanning)
- EthicalCheck** By APsec: EthicalCheck provides the industry's only free & automated API security testing service that uncovers security vulnerabilities using OWASP API list. (Code scanning)
- lintr** By GitHub Actions: lintr provides static code analysis for R. (Code scanning)
- CodeScan** By CodeScan Enterprises, LLC: CodeScan allows for better visibility on your code quality checks based on your custom rulesets. (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. (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

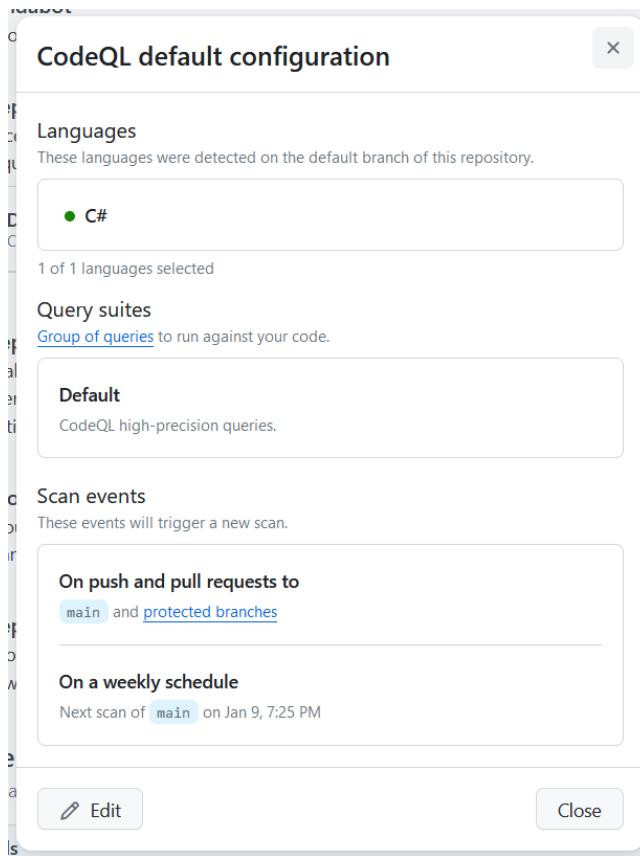
The query results are shown as code scanning alerts in GitHub when you use CodeQL with code scanning.

Code Scanning – CodeQL

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

Language	Variants	Compilers	Extensions
C/C++	C89, C99, C11, C17, C++98, C++03, C++11, C++14, C++17, C++20 [1] [2]	Clang (including clang-cl [3] and armclang) extensions (up to Clang 12.0), GNU extensions (up to GCC 11.1), Microsoft extensions (up to VS 2019), Arm Compiler 5 [4]	.cpp, .c++, .cxx, .hpp, .hh, .h++, .hxx, .c, .cc, .h
C#	C# up to 11	Microsoft Visual Studio up to 2019 with .NET up to 4.8, .NET Core up to 3.1 .NET 5, .NET 6, .NET 7	.sln, .csproj, .cs, .cshtml, .xaml
Go (aka Golang)	Go up to 1.21	Go 1.11 or more recent	.go
Java	Java 7 to 20 [5]	javac (OpenJDK and Oracle JDK), Eclipse compiler for Java (ECJ) [6]	.java
Kotlin [7]	Kotlin 1.5.0 to 1.9.20	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 [8]
Python [9]	2.7, 3.5, 3.6, 3.7, 3.8, 3.9, 3.10, 3.11, 3.12	Not applicable	.py
Ruby [10]	up to 3.2	Not applicable	.rb, .erb, .gemspec, Gemfile
Swift [11]	Swift 5.4-5.9.1	Swift compiler	.swift
TypeScript [12]	2.6-5.3	Standard TypeScript compiler	.ts, .tsx, .mts, .cts

Code Scanning – CodeQL



Choose the Default or Advanced configuration

uses GitHub Action:

github/codeql-action/analyze@v2

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 – Alerting (CodeQL)

Code scanning alerts / #958

Uncontrolled data used in path expression

[Open](#) in `main` 5 days ago

spec-main/api-session-spec.ts:940

```
937     const downloadfilePath = path.join/fixtures, 'logo.png');
938     const rangeServer = http.createServer((req, res) => {
939         const options = { root: fixtures };
940         send(req, req.url!, options)
This path depends on a user-provided value.
```

CodeQL [Show paths](#)

```
941             .on('error', (error: any) => { throw error; }).pipe(res);
942         );
943         try {
```

Severity [High](#)

Affected branches [main](#), [octocat-patch-1](#)

Tags [security](#)

Weaknesses [CWE-22](#), [CWE-23](#), [CWE-36](#), [CWE-73](#), [CWE-99](#)

Tool CodeQL Rule ID js/path-injection Query View source

Accessing files using paths constructed from user-controlled data can allow an attacker to access unexpected resources. This can result in sensitive information being revealed or deleted, or an attacker being able to influence behavior by modifying unexpected files.

Show more ▾

First detected in commit on Apr 3, 2023

Merge branch 'main' of github.com:octo-org/octo-repo a08159f

spec-main/api-session-spec.ts:828 on branch `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

No description provided.



More silly stuff 😎

Owner ...

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](#)

Code Scanning - Autofix

- Only for GitHub Enterprise customers
 - AI-powered expansion of code scanning
 - Currently only support JavaScript/TypeScript

github-advanced-security bot

The vulnerability in the code is due to the fact that user-provided input (`req.query.name`) is directly used in the HTTP response without any sanitization. This can lead to a cross-site scripting (XSS) attack if the user input includes malicious scripts.

To fix this vulnerability, we need to sanitize the user input before using it in the HTTP response. One way to do this is by using the `escape-html` library, which can escape any special characters in the user input that can be interpreted as part of an HTML document.

The changes need to be made in the file `index-2023-10-30.js`. Specifically, we need to add an import statement at the top of the file to import the `escape-html` library. Then, we need to use the `escape` function from this library to sanitize `req.query.name` before using it in the HTTP response.

Suggested fix generated by CodeQL and AI (Beta)

```
... ... @@ -1,4 +1,5 @@  
1 1 const express = require('express');  
2 + const escape = require('escape-html');  
3 3 const app = express();  
4 - app.get('/', (req, res) => res.send(`Hello, ${req.query.name}!`));  
5 + app.get('/', (req, res) => res.send(`Hello, ${escape(req.query.name)}!`));  
  
package.json Outside changed files  
... ... @@ -9,3 +9,4 @@  
9 9 "dependencies": {  
10 - "express": "^4.17.1"  
10 + "express": "^4.17.1",  
11 + "escape-html": "^1.0.3"  
11 12
```

Dismiss suggestion Edit Commit fix

This feature may produce inaccurate results. Double-check the suggested code change and make any necessary adjustments.

GitHub Advisory Database

15,693 advisories		Severity ▾	CWE ▾	Sort ▾
view_component	Cross-site Scripting vulnerability Moderate			
CVE-2024-21636 was published for view_component (RubyGems) 17 hours ago				
Duplicate Advisory: Malicious URL drafting attack against iodines static file server may allow path traversal Low				
GHSA-qwf7-rv77-fcr3 was published for iodine (RubyGems) 18 hours ago • withdrawn				
Duplicate Advisory: Integer overflow in cmark-gfm table parsing extension leads to heap memory corruption High				
GHSA-c2v4-chx5-vff6 was published for commonmarker (RubyGems) 18 hours ago • withdrawn				
Duplicate Advisory: encoded_id-rails potential DOS vulnerability due to URLs with extremely long encoded IDs High				
GHSA-4553-hq82-8654 was published for encoded_id-rails (RubyGems) 18 hours ago • withdrawn				
Duplicate Advisory: govuk_tech_docs vulnerable to unescaped HTML on search results page Low				
GHSA-4mvm-xh8j-fv27 was published for govuk_tech_docs (RubyGems) 18 hours ago • withdrawn				
Duplicate Advisory: Race Condition leading to logging errors Low				
GHSA-v444-jggx-6v7f was published for audited (RubyGems) 18 hours ago • withdrawn				
class.upload.php allows cross-site scripting attacks via uploaded files Moderate				
CVE-2023-6551 was published for verot/class.upload.php (Composer) yesterday				
Froxlor username/surname AND company field Bypass High				
CVE-2023-50256 was published for froxlor/froxlor (Composer) yesterday				



<https://github.com/advisories>

Dependency Graph

The screenshot shows a dependency graph interface with the following details:

- Dependencies** tab selected.
- Dependabot** tab available.
- Export SBOM** button.
- Search all dependencies** input field.
- Azure.Identity 1.8.2**: Detected automatically on Nov 28, 2023 (NuGet) · DevSecOpsDemo.csproj · MIT. Alert status: 1 high.
- System.Data.SqlClient 4.8.0**: Detected automatically on Nov 28, 2023 (NuGet) · DevSecOpsDemo.csproj · MIT. Alert status: 1 moderate.
- actions/checkout 3**: Detected automatically on Mar 23, 2023 (GitHub Actions) · .github/workflows/sonarcloud.yaml
- actions/setup-dotnet 3**: Detected automatically on Mar 23, 2023 (GitHub Actions) · .github/workflows/sonarcloud.yaml
- actions/setup-java 3**: Detected automatically on Mar 23, 2023 (GitHub Actions) · .github/workflows/sonarcloud.yaml
- Azure.Storage.Blobs 12.16.0**: Detected automatically on Nov 28, 2023 (NuGet) · DevSecOpsDemo.csproj · MIT
- actions/checkout 3**: Detected automatically on Mar 23, 2023 (GitHub Actions) · .github/workflows/codeql.yml
- github/codeql-action/analyze 2**

A modal window is open for **System.Data.SqlClient**, showing:

- View Dependabot alert**: 1 moderate · 1 total
- Update to v4.8.5**: #1
- DevSecOpsDemo.csproj update suggested:** `System.Data.SqlClient ~> ...`
- Note:** Always verify the validity and compatibility of suggestions with your codebase.

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

GitHub security alert digest

thomasvochten's repository security updates from the week of **Nov 21 - Nov 28**

💻 thomasvochten's personal account

⚠️ thomasvochten / Demo-DevSecOps

Known security vulnerabilities detected

Dependency **System.Data.SqlClient** Version `<= 4.8.4` Upgrade to `~>`
4.8.5

Defined in **DevSecOpsDemo.csproj**

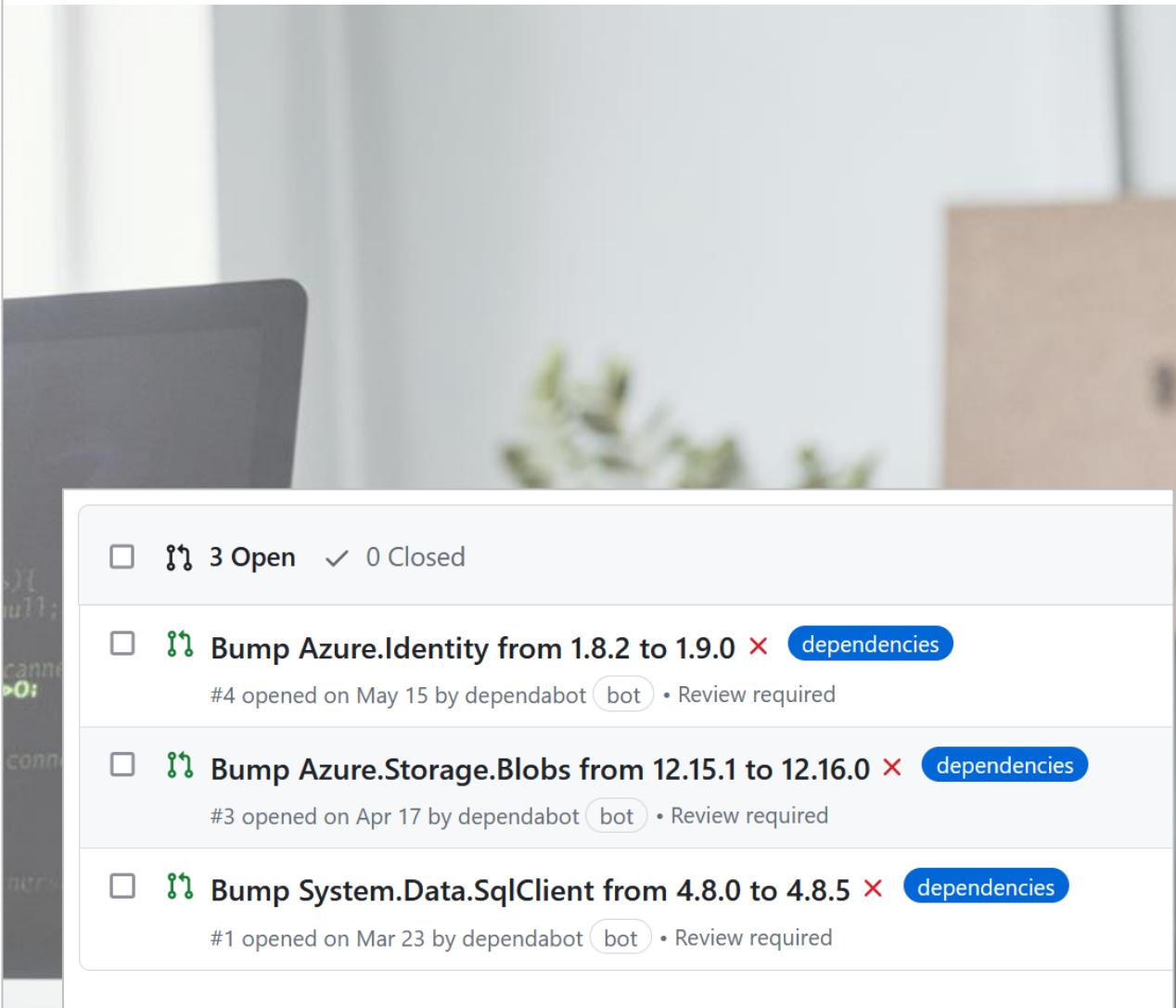
Suggested update
[#1](#)

Vulnerabilities
CVE-2022-41064 Moderate severity

Dependency **Azure.Identity** Version `< 1.10.2` Upgrade to `~>`
1.10.2

Defined in **DevSecOpsDemo.csproj**

Vulnerabilities
CVE-2023-36414 High severity



Dependabot dashboard showing four open pull requests:

- 3 Open ✓ 0 Closed
- Bump Azure.Identity from 1.8.2 to 1.9.0 X dependencies
#4 opened on May 15 by dependabot bot • Review required
- Bump Azure.Storage.Blobs from 12.15.1 to 12.16.0 X dependencies
#3 opened on Apr 17 by dependabot bot • Review required
- Bump System.Data.SqlClient from 4.8.0 to 4.8.5 X dependencies
#1 opened on Mar 23 by dependabot bot • Review required

Dependabot – Alerting

.NET Information Disclosure Vulnerability #1

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

Fix 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 View

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

Dependabot – Pull Request

Bump Azure.Identity from 1.8.2 to 1.10.2 #5

[Edit](#) [Code](#)

[Open](#) dependabot wants to merge 1 commit into [main](#) from [dependabot/nuget/Azure.Identity-1.10.2](#)

Conversation 0 Commits 1 Checks 1 Files changed 1 +1 -1

dependabot (bot) commented on behalf of github on Nov 28, 2023

Bumps [Azure.Identity](#) from 1.8.2 to 1.10.2.

► Commits

compatibility unknown

Dependabot will resolve any conflicts with this PR as long as you don't alter it yourself. You can also trigger a rebase manually by commenting `@dependabot rebase`.

► Dependabot commands and options

dependabot (bot) added the [dependencies](#) label on Nov 28, 2023

dependabot (bot) mentioned this pull request on Nov 28, 2023

[Closed](#)

Reviewers: thomasvochten Request: At least 1 approving review is required to merge this pull request.

Still in progress? [Convert to draft](#)

Assignees: No one—[assign yourself](#)

Labels: [dependencies](#)

Projects: None yet

Milestone: No milestone

Development: Successfully merging this pull request may close these

Dependabot – Notable other options

.github/dependabot.yml

```
version: 2
updates:
  - package-ecosystem: "nuget" # See documentation for possible values
    directory: "/" # Location of package manifests
    schedule:
      interval: "weekly"
```

Code security and analysis / Dependabot rules Beta Give feedback

New rule

GitHub presets

Managed by GitHub

Dismiss low-impact alerts for development-scoped dependencies

In a developer (non-production or runtime) environment, these alerts are unlikely to be exploitable or have limited effect like slow builds or long-running tests. [Learn more about this methodology.](#)

Enabled





GitHub to the rescue!



Secret
Scanning



Code
Scanning



Dependency
Scanning

Defender for Cloud integration

Severity ↑↓	Description	Status ↑↓
High	GitHub repositories should have secret scanning enabled	● Healthy
High	GitHub repositories should have secret scanning findings resolved	● Unhealthy
Medium	GitHub repositories should have dependency vulnerability scanning findings resolved	● Unhealthy
Medium	GitHub repositories should have code scanning enabled	● Healthy
Medium	GitHub repositories should have infrastructure as code scanning findings resolved	● Healthy
Medium	GitHub repositories should have code scanning findings resolved	● Unhealthy
Medium	GitHub repositories should have Dependabot scanning enabled	● Healthy
Medium	GitHub repositories should have API security testing findings resolved	● Preview Healthy

Defender for Cloud integration

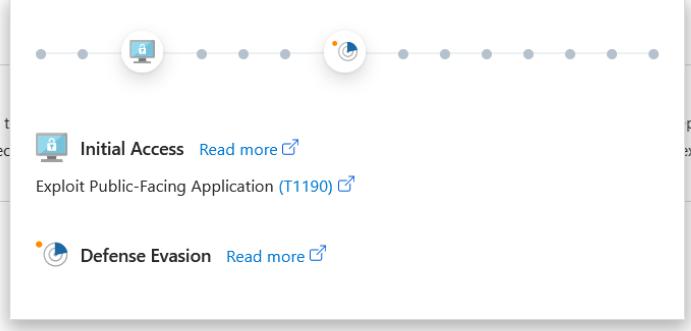
GitHub repositories should have secret scanning enabled ... X

Exempt Open query

Severity	Freshness interval	Tactics and techniques
High	60 Min	 Initial Access +1

Description
GitHub scans repositories for known types of secrets, to prevent fraudulent use of secrets that were accidentally committed to a repository. Examples of secrets are tokens and private keys that a service provider can issue for authentication. If a secret is checked into a repository, GitHub scans it for known types of secrets. If a secret is found, GitHub prevents those privileges. Secrets should be stored in a dedicated, secure location outside the repository for the project.

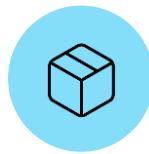
Remediation steps
Manual remediation:
1. On GitHub.com, navigate to the main page of the repository.
2. Under your repository name, click Settings.
3. In the left sidebar, click Security and analysis.
4. If Advanced Security is not already enabled for the repository, to the right of GitHub Advanced Security, click Enable.
5. Review the impact of enabling Advanced Security, then click Enable GitHub Advanced Security for this repository.
6. When you enable Advanced Security, secret scanning may automatically be enabled for the repository due to the organization's settings. If Secret scanning is shown with an Enable button, you still need to enable secret scanning by clicking Enable. If you see a Disable button, secret scanning is already enabled


Initial Access [Read more ↗](#)
Exploit Public-Facing Application (T1190) [↗](#)
Defense Evasion [Read more ↗](#)

Take action Trigger logic app Exempt Assign owner



What about Azure DevOps?



Similar
features?



Something
missing?



Something
more?

GitHub Advanced Security for Azure DevOps

- GA on September 20, 2023
- Secret, code and dependency scanning
- Enable for individual repositories under Project Settings -> Repositories
- Project Collection Admin

The screenshot shows the GitHub repository settings for 'Fabrikam'. A red box highlights the 'Advanced Security' section, which is currently turned off. Below it, the 'Repository Settings' section is shown, containing several other toggle switches:

- Forks**: Off (Allows users to create forks from this repository)
- Commit mention linking**: Off (Automatically creates links for work items mentioned in a commit comment)
- Commit mention work item resolution**: Off (Allows mentions in commit comments to close work items (e.g. "Fixes #123"))
- Work item transition preferences**: Off (Remembers user preferences for completing work items with pull requests)
- Permissions management**: Off (Allows users to manage permissions for the branches they created)
- Strict Vote Mode**: Off (Enables Strict Vote Mode for repository which requires Contribute permission to vote in Pull Requests)

Secret scanning

- Automatically enabled with push protection(!)

The screenshot shows the Microsoft CloudHub interface for managing repositories. On the left, there's a sidebar with project settings for 'Contoso'. The main area displays 'All Repositories' with three listed: 'AdventureWorks', 'Fabrikam' (which is selected), and 'TailSpin'. The 'Fabrikam' repository page is shown in detail, featuring tabs for 'Settings', 'Policies', and 'Security'. A red box highlights the 'Advanced Security' section, which contains a toggle switch set to 'On', a description about protecting repositories with security features, and a checked checkbox for 'Block secrets on push'. Below this, another red box highlights the 'Repository Settings' section, which includes toggles for 'Forks' (off) and 'Commit mention linking' (off). The top right of the interface includes a search bar, navigation icons, and a user profile.

CloudHub / Contoso / Settings / Repositories

Search

Project Settings

Contoso

General

- Overview
- Teams
- Permissions
- Notifications
- Service hooks
- Dashboards

Boards

- Project configuration
- Team configuration
- GitHub connections

Pipelines

All Repositories

Filter by keywords

- AdventureWorks
- Fabrikam
- TailSpin

Fabrikam

Browse Rename

Settings Policies Security

Advanced Security

Protect your repositories with security and analysis features like dependency scanning, code scanning, and secret scanning. [View billing](#) | [Learn more](#)

Block secrets on push

Scan all pushes to the repository and block pushes containing secrets.

Repository Settings

Forks

Allow users to create forks from this repository.

Commit mention linking

Automatically create links for work items mentioned in a commit comment.

Secret scanning

- GitHub Advanced Security team maintains the default secret scanning patterns

The screenshot shows the Azure DevOps interface for the Contoso organization under the Advanced Security section. The 'Secrets' tab is selected. A search bar at the top right contains the word 'Search'. On the left, a sidebar lists various repository-related options: Overview, Boards, Repos (which is selected), Files, Commits, Pushes, Branches, Tags, Pull requests, Advanced Security (which is also selected), and Pipelines. In the main content area, the title 'Advanced Security' is displayed above three tabs: Dependencies, Code scanning, and Secrets. Below these tabs is a search bar with the placeholder 'Filter by keywords' and dropdown filters for 'State: Open' and 'Type'. A single alert is listed: 'Alert' for 'Azure DevOps personal access token (PAT) ...uo4kta' was introduced 'Just now'. The alert is labeled 'Critical' and is located in file '#146 in src/secrets.txt:1'.

Secret scanning

- Close the alert after revoking the secret, accepting the risk or if false positive

Azure DevOps CloudHub / Contoso / Repos / Advanced Security / Fabrikam

Search

Contoso

Overview

Boards

Repos

Files

Commits

Pushes

Branches

Tags

Pull requests

Advanced Security

Pipelines

#146 [Open](#) in 41b9a93b • detected Just now

← Azure DevOps personal access token (PAT)

Location

src/secrets.txt:1 @ 41b9a93b

Reason

Revoked
The secret has been revoked.

Risk accepted
Risk is tolerable or irrelevant (e.g., only used in tests or not exploitable in your implementation).

False positive
This alert is inaccurate or incorrect.

Comment (optional)

Cancel Close

Push protection

- Push protection is available on both the command line and the web interface

The screenshot shows the Azure DevOps web interface for a repository named 'Fabrikam'. The left sidebar includes links for Contoso, Overview, Boards, Repos (selected), Files, Commits, Pushes, Branches, Tags, Pull requests, Advanced Security, and Pipelines. The main area displays the repository structure under 'main': 'src' (containing 'app.js' and 'index.html'), 'secrets.txt', '.gitignore', 'azure-pipelines.yml', 'package-lock.json', 'package.json', and 'README.md'. A tooltip over 'secrets.txt' indicates it contains secrets. A 'Commit' modal is open, showing a warning message: 'VS403654: The push was rejected because it contains one or more secrets. Resolve the following secrets before pushing again. For help, see https://aka.ms/advancedsecurity/secret-scanning/push-protection. Secrets: commit: 1dcf26a75cbcd0f189f76af2e9aefff1daee24fc paths: /src/secrets.txt (1,1-53) : SEC101/102 : AdoPat'. The modal also has fields for 'Comment' (containing 'Added secrets.txt'), 'Branch name' (set to 'main'), and 'Work items to link' (with a placeholder 'Search work items by ID or title').

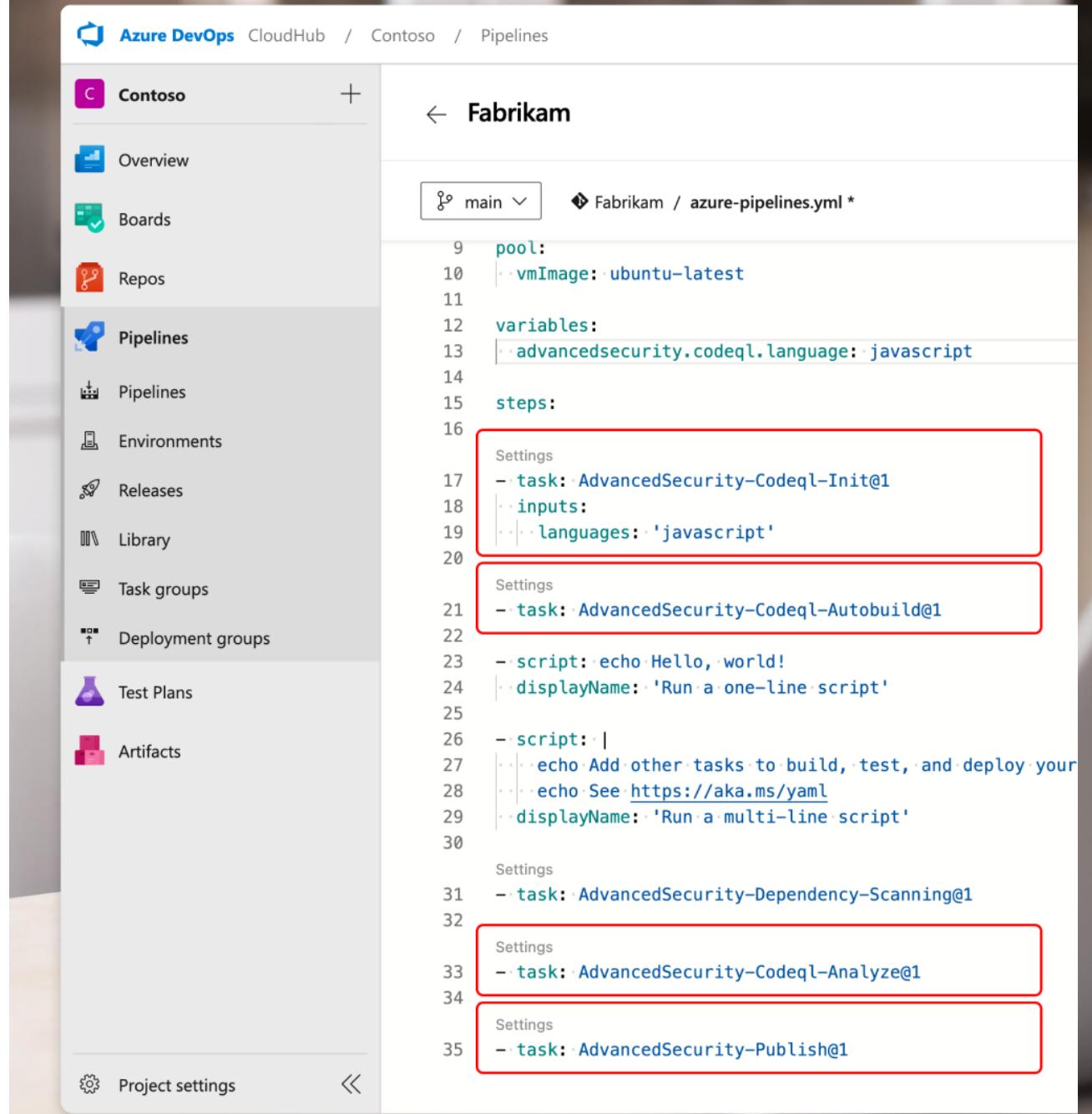
Code scanning

- CodeQL identifies security vulnerabilities
- Code analysis engine
- Queries are open source
 - C/C++
 - C#
 - Go
 - Java,
 - JavaScript/TypeScript
 - Kotlin (beta)
 - Python
 - Ruby



Code scanning

- Add the required actions to the pull request build pipeline
- Pros:
 - Easy to set up
- Cons:
 - Does not find as many issues (esp. quality) as some third-party products



The screenshot shows the Azure DevOps Pipelines interface. On the left, a sidebar lists various project management features: Overview, Boards, Repos, Pipelines (which is selected), Pipelines, Environments, Releases, Library, Task groups, Deployment groups, Test Plans, and Artifacts. At the bottom of the sidebar are Project settings and a back arrow. The main area displays the contents of the `azure-pipelines.yml` file for the `Fabrikam` project. The file defines a pipeline with several steps, some of which are highlighted with red boxes:

```
pool:
vmImage: ubuntu-latest

variables:
advancedsecurity.codeql.language: javascript

steps:
- task: AdvancedSecurity-Codeql-Init@1
  inputs:
    languages: 'javascript'

- task: AdvancedSecurity-Codeql-Autobuild@1
  inputs:
    script: echo Hello, world!
    displayName: 'Run a one-line script'

- task: AdvancedSecurity-Dependency-Scanning@1
  inputs:
    task: AdvancedSecurity-Codeql-Analyze@1

- task: AdvancedSecurity-Publish@1
```

Code scanning

- Automatically closed when no longer detected; can also be accepted or marked as FP

The screenshot shows a detailed view of a code scanning alert in the Azure DevOps interface. The alert is for a vulnerability titled "DOM text reinterpreted as HTML (js/xss-through-dom)" in repository "Fabrikam".

Alert Details:

- #145 (Open) in main • detected Today at 11:22 AM
- Reason: False positive (selected)
- Comment (optional): [Empty box]

Location:

- src/index.html:41 @ 258d2fd1

Description:

Extracting text from a DOM node and interpreting it as HTML can lead to a cross-site scripting vulnerability.

A webpage with this vulnerability reads text from the DOM, and afterwards adds the text as HTML to the DOM. Using text from the DOM as HTML effectively unescapes the text, and thereby invalidates any escaping done on the text. If an attacker is able to control the safe sanitized text, then this vulnerability can be exploited to perform a cross-site scripting attack.

Recommendation:

To guard against cross-site scripting, consider using contextual output encoding/escaping before writing text to the page, or one of the other solutions that are mentioned in the References section below.

Tags: js/xss-through-dom

Weaknesses: CWE-079, CWE-116

Code scanning

- Closed alerts can be viewed via the filtering options

The screenshot shows the 'Advanced Security' page in the Azure DevOps interface. The left sidebar is visible with items like 'Overview', 'Boards', 'Repos', 'Files', 'Commits', 'Pushes', 'Branches', 'Tags', 'Pull requests', 'Advanced Security' (which is selected and highlighted in blue), 'Pipelines', and 'Test Plans'. The main content area has tabs for 'Dependencies', 'Code scanning', and 'Secrets', with 'Dependencies' selected. A search bar at the top says 'Filter by keywords'. To the right of the search bar are filters for 'Branch: main', 'State: Closed', 'Pipeline', 'Package', and 'Severity'. A dropdown menu is open over the 'State: Closed' filter, showing options: 'Open' (disabled), 'All closed' (selected and highlighted in blue), 'Risk accepted', 'False positive', and 'Fixed'. Below the dropdown, there are several rows of security findings:

Alert	First detected
Regular Expression Denial of Service in ms (CVE-2015-8315) High #7318931	9m ago
Prototype Pollution Protection Bypass in qs (CVE-2017-1000048) High #7318932	9m ago
qs vulnerable to Prototype Pollution (CVE-2022-24999) High #7318933	9m ago
Prototype Pollution Protection Bypass in qs (CVE-2017-1000048) High #7318936	9m ago
qs vulnerable to Prototype Pollution (CVE-2022-24999) High #7318937	9m ago
qs vulnerable to Prototype Pollution (CVE-2022-24999) High #7318938	9m ago

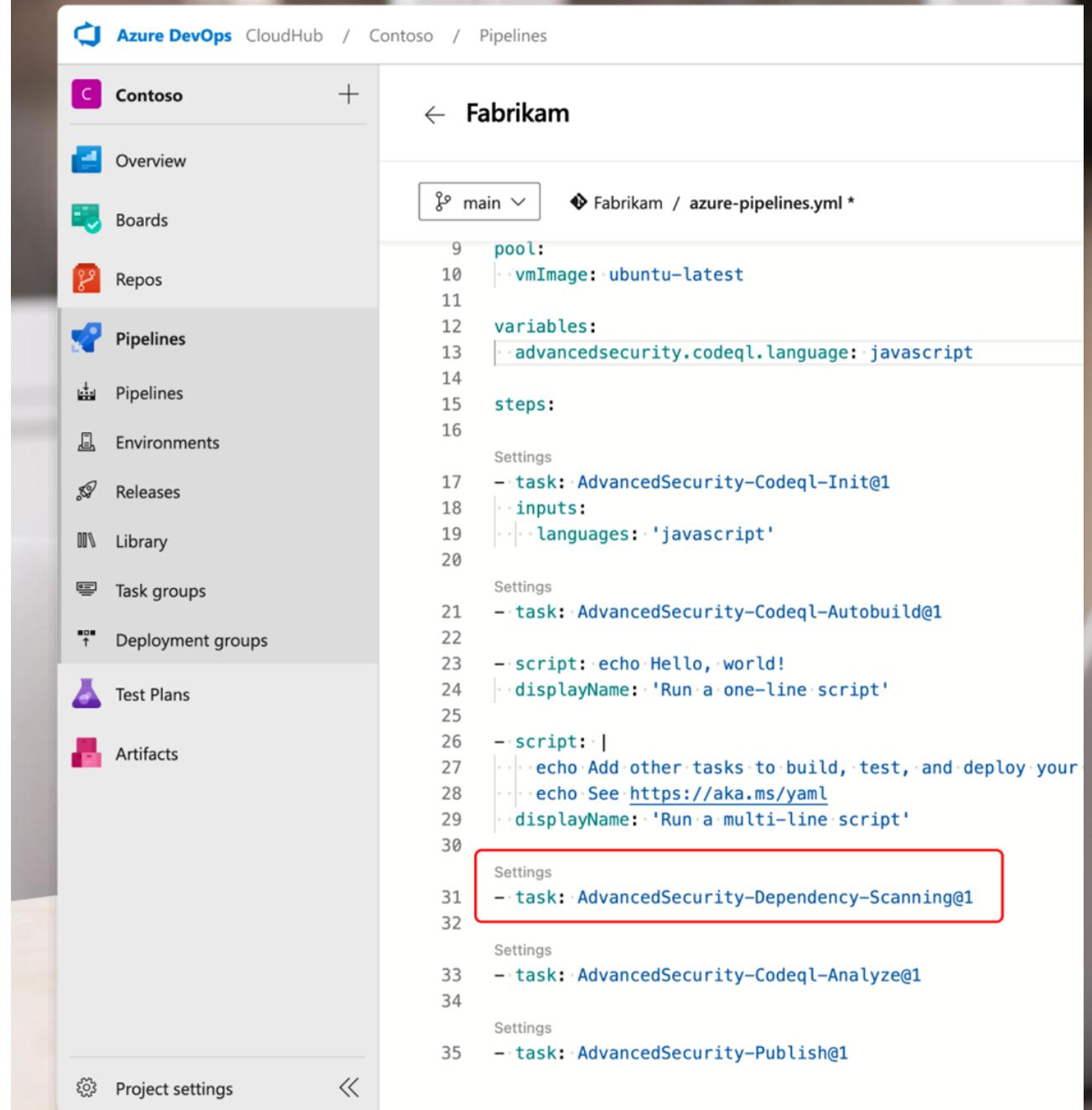
Dependency scanning

- Checks if open-source components used in your source code have associated vulnerabilities
- Direct and transitive
- GitHub Advisory Database



Dependency scanning

- Add the required action to the build pipeline
- Pros:
 - Seems to find more issues than many competitors
- Cons:
 - No inheritance path
 - No email alerts
 - No break build option



The screenshot shows the Azure DevOps Pipelines interface. On the left, there's a sidebar with options like Overview, Boards, Repos, Pipelines (which is selected), Pipelines, Environments, Releases, Library, Task groups, Deployment groups, Test Plans, and Artifacts. At the bottom of the sidebar is a 'Project settings' link. The main area shows a YAML configuration file for a pipeline named 'main'. The file includes sections for pool, variables, steps, and tasks. A specific task at line 31, 'AdvancedSecurity-Dependency-Scanning@1', is highlighted with a red rectangle. The code snippet is as follows:

```
9 pool:
10   vmImage: ubuntu-latest
11
12 variables:
13   advancedsecurity.codeql.language: javascript
14
15 steps:
16
17   - task: AdvancedSecurity-Codeql-Init@1
18     inputs:
19       languages: 'javascript'
20
21   - task: AdvancedSecurity-Codeql-Autobuild@1
22
23     - script: echo Hello, world!
24       displayName: 'Run a one-line script'
25
26     - script: |
27       echo Add other tasks to build, test, and deploy your
28       echo See https://aka.ms/yaml
29       displayName: 'Run a multi-line script'
30
31   - task: AdvancedSecurity-Dependency-Scanning@1
32
33   - task: AdvancedSecurity-Codeql-Analyze@1
34
35   - task: AdvancedSecurity-Publish@1
```

Price and billing

- 49\$ per *active committer* per month
- Billed via the Azure sub connected to the ADO org
- If the same sub has multiple orgs, committers are deduplicated

The screenshot shows the Azure DevOps CloudHub Settings / Billing page. The left sidebar lists various organization settings categories: General (Overview, Projects, Users, Billing), Security (Policies, Permissions), Boards, Repos, Pipelines, and Parallel jobs. The 'Billing' section is highlighted. The main content area displays the Azure Subscription ID (12345678-abcd-abcd-abcd-12345678abcd) and a table of usage metrics. A red box highlights the 'Advanced Security' row, which shows 'Used' under the 'Status' column and 'Unique active committers' with a value of '123'. A note below states: 'Advanced Security is billed based on the number of unique active committers in repository. Active committers are users that have committed to an Advanced Security enabled repository in the last 90 days.' with a 'Learn more' link.

Pipelines for private projects	Free	Paid parallel jobs
MS Hosted CI/CD	10	
Self-Hosted CI/CD	1	10
Visit parallel jobs for full details on free pipelines and public concurrency		
Boards, Repos and Test Plans	Free	
Basic users	5	
Basic + Test Plans		
This organization is enabled for user assignment based billing and daily pro-rated charges, instead of monthly committed purchases. Learn more		
Advanced Security	Used	
Unique active committers	123	
Advanced Security is billed based on the number of unique active committers in repository. Active committers are users that have committed to an Advanced Security enabled repository in the last 90 days. Learn more		

Secret scanning – no longer

- Microsoft Defender for DevOps recommends enabling GASfADO

Microsoft Defender for Cloud | DevOps Security (Preview) Showing 2 subscriptions | PREVIEW

Search Add environment Refresh DevOps workbook Guides and Feedback Getting Started Configure

General

- Overview
- Getting started
- Recommendations
- Security alerts
- Inventory
- Cloud Security Explorer (Preview)
- Workbooks
- Community
- Diagnose and solve problems

Cloud Security

- Security posture
- Regulatory compliance
- Workload protections
- Firewall Manager
- DevOps Security (Preview)

Management

- Environment settings
- Security solutions
- Workflows

Security Overview

DevOps security vulnerabilities 234 VULNERABILITIES

High	Medium	Low
39	195	0

DevOps security results

169 Code scanning vulnerabilities	18 Exposed Secrets
31 OSS vulnerabilities	28 Recommendations

DevOps coverage

1 Github Connectors	1 Azure DevOps Connectors
30 Total	
Github repositories 27	Azure DevOps repositories 3

Subscription: Contoso Hotels Tenant - Production, CyberSecurity

Resource Types: Github Repository, Azure DevOps Repository

Name	Pull request status	Total exposed secrets	OSS vulnerabilities	Total code scanning vulnerabilities
ASE_SG_Demo	N/A	Unhealthy (1)	1	65
RS_ramontest	N/A	Unhealthy (1)	0	65
DfDDemo	N/A	Unhealthy (4)	17	16
Toy-Website	N/A	Unhealthy (2)	0	0
Contoso Hotels	On	Unhealthy (1)	N/A	0
RepositoriesSampleContent	N/A	Healthy	0	0
Toy-Website	On	Healthy	N/A	0
DfD Demo	On	Healthy	N/A	0

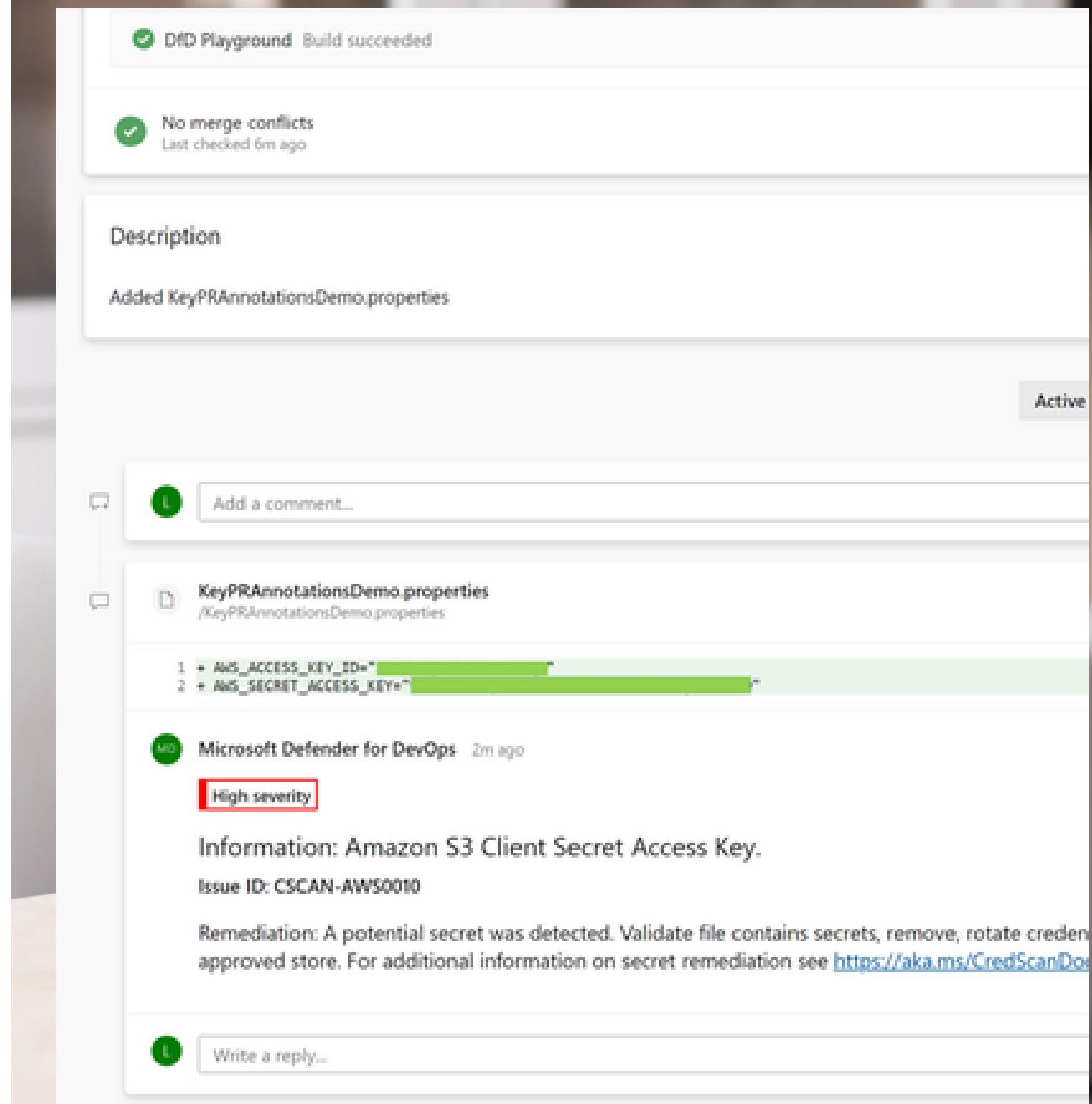
IaC scanning

- Enable Microsoft Defender for DevOps on Azure Portal
- Add Microsoft Security DevOps task to pipeline
- Needs to be run for every repo/branch you want to scan

```
6   trigger:
7     - main
8
9   pool:
10    vmImage: ubuntu-latest
11
12  steps:
13    Settings
14    - task: NodeTool@0
15      inputs:
16        versionSpec: '16.x'
17        displayName: 'Install Node.js'
18
19    - script: |
20      npm install
21      npm run build
22      displayName: 'npm install and build'
23
24    Settings
25    - task: MicrosoftSecurityDevOps@1
26      inputs:
27        categories: 'secrets, code, artifacts'
28        languages: 'javascript, typescript'
29        tools: 'credscan, eslint'
30
31    Settings
32    - task: Npm@1
33      displayName: 'Publish package to feed'
34      continueOnError: true
35      inputs:
36        command: 'publish'
37        publishRegistry: 'useFeed'
38        publishFeed: '3236f8cd-1ab2-47cd-9e49-f395927f206b/13954e39-73a7'
```

IaC scanning

- Results show in Azure Portal, build results, pull requests (if enabled)
- Combine pull request annotations with a branch policy



Code scanning

- Many third-party tools available
- SonarCloud
- 10\$/month
- Requires build pipeline configuration

```
1 trigger: none
2
3 pool:
4   - vmImage: ubuntu-latest
5
6 steps:
7   - checkout: self
8   - fetchDepth: 0
9   Settings
10  - task: NodeTool@0
11    inputs:
12      - versionSpec: '16.x'
13      - displayName: 'Install Node.js'
14    Settings
15  - task: SonarCloudPrepare@1
16    inputs:
17      - SonarCloud: 'SonarCloud'
18      - organization: 'sulava'
19      - scannerMode: 'CLI'
20      - configMode: 'manual'
21      - cliProjectKey: 'Sulava.Common'
22      - cliProjectName: 'Sulava.Common'
23      - cliSources: '.'
24    - script: |
25        - npm install
26        - npm run build
27    - displayName: 'npm install and build'
28    Settings
29  - task: SonarCloudAnalyze@1
30    Settings
31  - task: SonarCloudPublish@1
32    inputs:
33      - pollingTimeoutSec: '300'
```

Code scanning

- Require build pipeline execution and resolving comments via branch policies

Laura Kokkarinen completed the pull request May 27

TS May 27

```
^ 248 250      : null}
249 251      { editMode ?
250 252          <div>
251 -          <PrimaryButton className={styles.saveButton} text={strings.Save} onClick={async () => await saveItem()} />
253 +          <PrimaryButton className={styles.saveButton} text={strings.Save} onClick={() => saveItem()} />
252 254          <DefaultButton className={styles.cancelButton} text={strings.Cancel} onClick={() => cancelEdit()} />
253 255      </div>
254 256      : null }
```

KB Kimmo Bergius May 27 Active

Bug: Promise-returning function provided to attribute where a void return was expected. ([typescript:S6544](#))
[See it in SonarCloud](#)

Write a reply... Resolve

Dependency scanning

- Many third-party tools available
- Snyk (free)
- Requires build pipeline configuration
- Config when to fail the build

```
6 trigger:
7 - main
8
9 pool:
10 | .vmImage: ubuntu-latest
11
12 steps:
13 Settings
14 --task: NodeTool@0
15 | .inputs:
16 | | .versionSpec: '16.x'
17 | | .displayName: 'Install Node.js'
18 | .script: |
19 | | .npm install
20 | | .npm run build
21 | | .displayName: 'npm install and build'
22
23 Settings
24 --task: SnykSecurityScan@1
25 | .inputs:
26 | | .serviceConnectionEndpoint: 'Snyk'
27 | | .testType: 'app'
28 | | .monitorWhen: 'always'
29 | | .failOnIssues: true
30 | | .additionalArguments: '--fail-on-all'
31
32 Settings
33 --task: Npm@1
34 | .displayName: 'Publish package to feed'
35 | .continueOnError: true
36 | .inputs:
37 | | .command: 'publish'
38 | | .publishRegistry: 'useFeed'
39 | | .publishFeed: '3236f8cd-1ab2-47cd-9e49-f395927f206b/13954e39-73a7-4799'
```

Dependency scanning

- Direct and transitive
- Detailed report
- Sends out email alerts about discovered vulnerabilities

The screenshot shows a software interface for dependency scanning. On the left, there's a vertical toolbar with icons for Reports, Issues, Vulnerabilities, and others. The main area is titled "Reports:" and shows a summary for a "Snyk Test for npm (report-2023-05-27 16:04:31) | Found 5 issues". A specific issue is highlighted for the "validator@8.2.0" package. The "Overview" section states that "validator" is a library of string validators and sanitizers, mentioning a vulnerability related to Regular Expression Denial of Service (ReDoS) via the `isSlug` function. Below this, a "PoC" (Proof of Concept) code snippet is provided:

```
var validator = require("validator")
function build_attack(n) {
  var ret = "111"
  for (var i = 0; i < n; i++) {
    ret += "a"
  }
  return ret+"_";
}
for(var i = 1; i <= 50000; i++) {
  if (i % 10000 == 0) {
    var time = Date.now();
    var attack_str = build_attack(i)
    validator.isSlug(attack_str)
    var time_cost = Date.now() - time;
    console.log("attack_str.length: " + attack_str.length + ":" + time_cost+" ms")
  }
}
```

The "Details" section defines Denial of Service (DoS) as a family of attacks aimed at making a system inaccessible to its original and legitimate users, with many types ranging from network clogging to generating large volumes of traffic.

Dependency scanning

- How to override transitive vulnerable “acorn” package in a Node project:

```
{  
  "name": "npm-overrides",  
  "version": "1.0.0",  
  "license": "MIT",  
  "dependencies": {  
    "axios": "0.19.2",  
    "eslint  },  
  "overrides": {  
    "eslint      "espree": {  
        "acorn": "6.4.1" // patched, non-vulnerable version  
      }  
    }  
  }  
}
```

Dependency scanning

- How to override transitive vulnerable “Http.Connections” package in a .NET project:

```
<Project Sdk="Microsoft.NET.Sdk.Web">
  <PropertyGroup>
    <TargetFramework>netcoreapp3.1</TargetFramework>
    <RootNamespace>NuGet.Dependencies</RootNamespace>
  </PropertyGroup>
  <ItemGroup>
    <PackageReference Include="Microsoft.AspNetCore.App" Version="2.2.8" />
  </ItemGroup>

  <ItemGroup Label="Dependency Resolutions">
    <!-- Microsoft.AspNetCore.App -->
    <PackageReference Include="Microsoft.AspNetCore.Http.Connections"
Version="[1.0.15,1.1.0)" />
  </ItemGroup>
</Project>
```

Which way to go?

- Ideally, use both
- Costs



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

Defender for Cloud integration

What about Azure DevOps?

```
import java.util.ArrayList;
import java.util.Scanner;
import java.io.File;
import java.io.IOException;
import java.util.Arrays;
public class AirlineProblem {
    public static void main(String[] args){
        Scanner scannerToReadAirlines = null;
        try{
            scannerToReadAirlines = new Scanner(new File("airlines.txt"));
        } catch(IOException e){
            System.out.println("could not connect to file: " + e);
            System.exit(0);
        }
        if(scannerToReadAirlines != null){
            Scanner scannerToReadAirlinePartnersNetwork = new Scanner(new File("airlinePartnersNetwork.txt"));
            ArrayList<Airline> airlinesPartnersNetwork = new ArrayList<Airline>();
            Airline newAirline;
            String lineFromfile;
            try{
                while( scannerToReadAirlines.hasNext() ){
                    lineFromfile = scannerToReadAirlines.nextLine();
                    airlineNames = lineFromfile.split(",");
                    newAirline = new Airline(airlineNames);
                    if(airlineNames != null){
                        airlinesPartnersNetwork.add(newAirline);
                    }
                }
            } catch (IOException e) {
                System.out.println("could not connect to file: " + e);
            }
        }
    }
}
```





Thank you!