
{DevSecOps}
Application Security and DevSecOps
- End to end

Losses caused by cyber attacks reported to IC3

2021	2023
$ 6.900.000.000	$ 12.500.000.000

Loss caused by reported cyber crime (in billion USD)

Top 5 crime types:

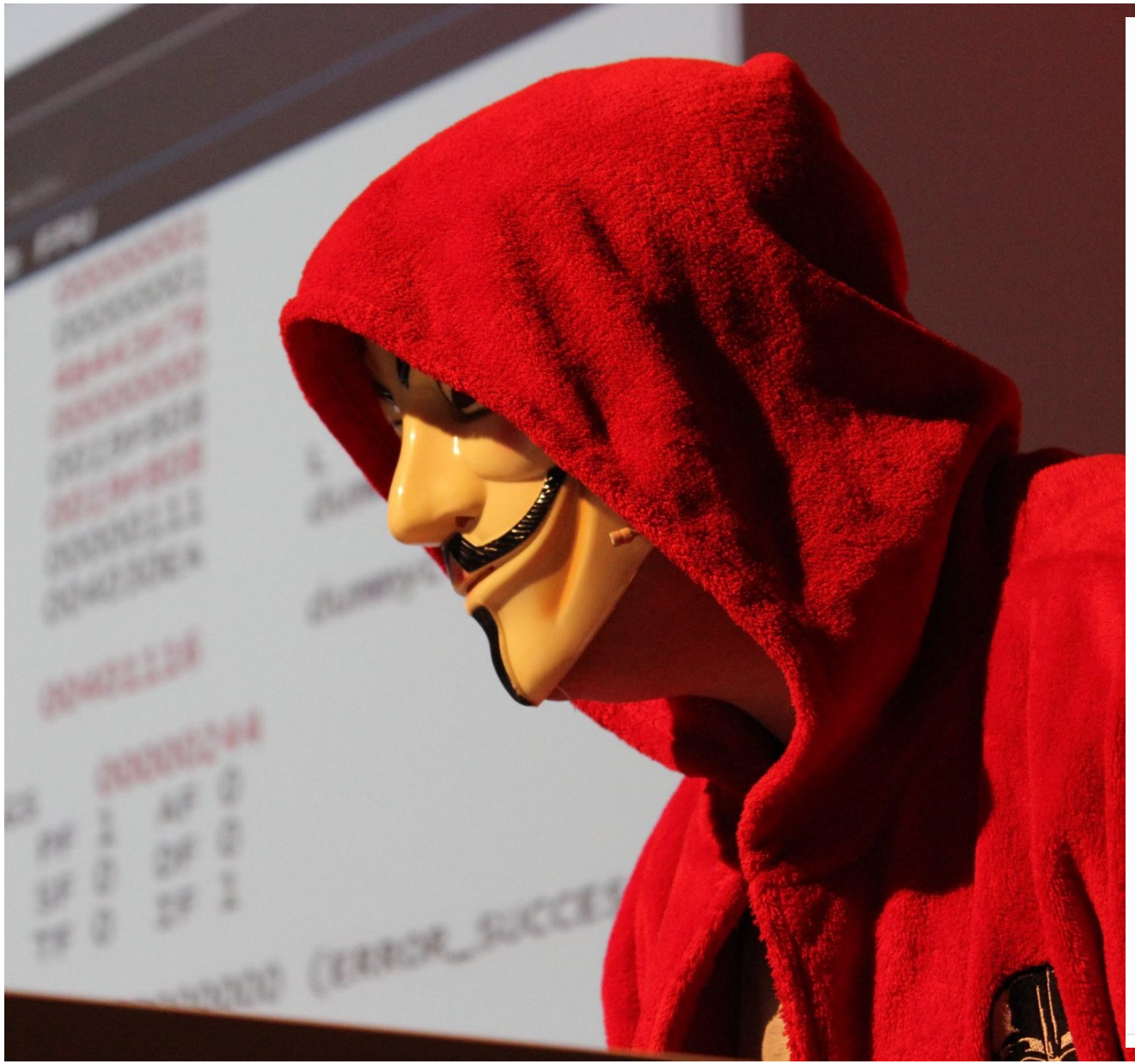
- Phishing
- Non-Payment / Delivery
- Data Breach
- Identity Theft
- Extortion

Trends

- Confidence fraud / Romance scams
- Cryptocurrency
- Ransomware
- Tech support fraud

https://www.ic3.gov/Media/PDF/AnnualReport/2023_IC3Report.pdf


Hackers in movies
vs
Image thanks to Gregor Biswanger @BFreakout
© 2021 Michael Kaufmann @mike_kaufmann



How people think they get hacked

How they really get hacked

Michael Kaufmann

Founder & Managing Director, Xebia Germany

@mike_kaufmann

@wulfland

https://writeabout.net

Microsoft REGIONAL DIRECTOR

MVP Microsoft® Most Valuable Professional

- 25 years software developer
- 15 years ALM & DevOps
- 10 years Git and GitHub
- Microsoft RD & MVP

5

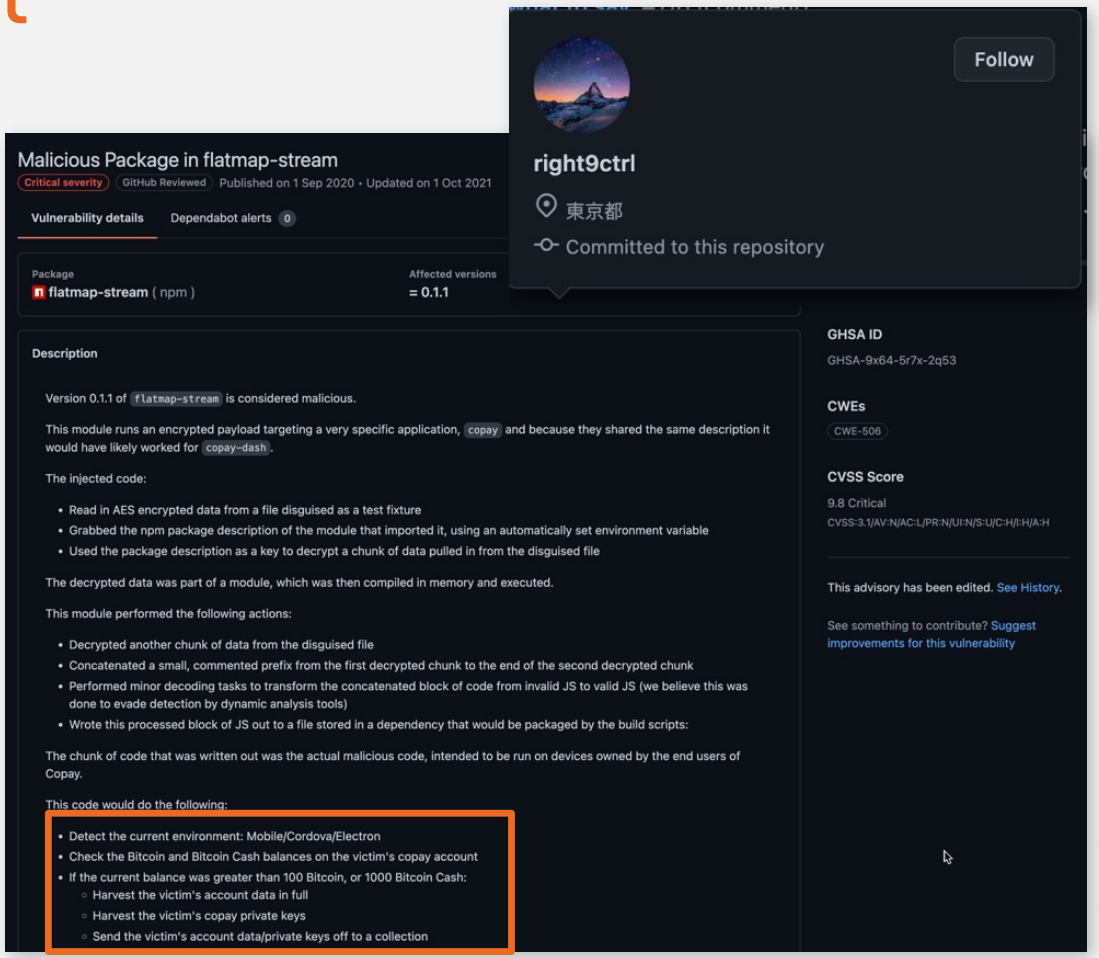
The event-stream incident

Social engineering attack

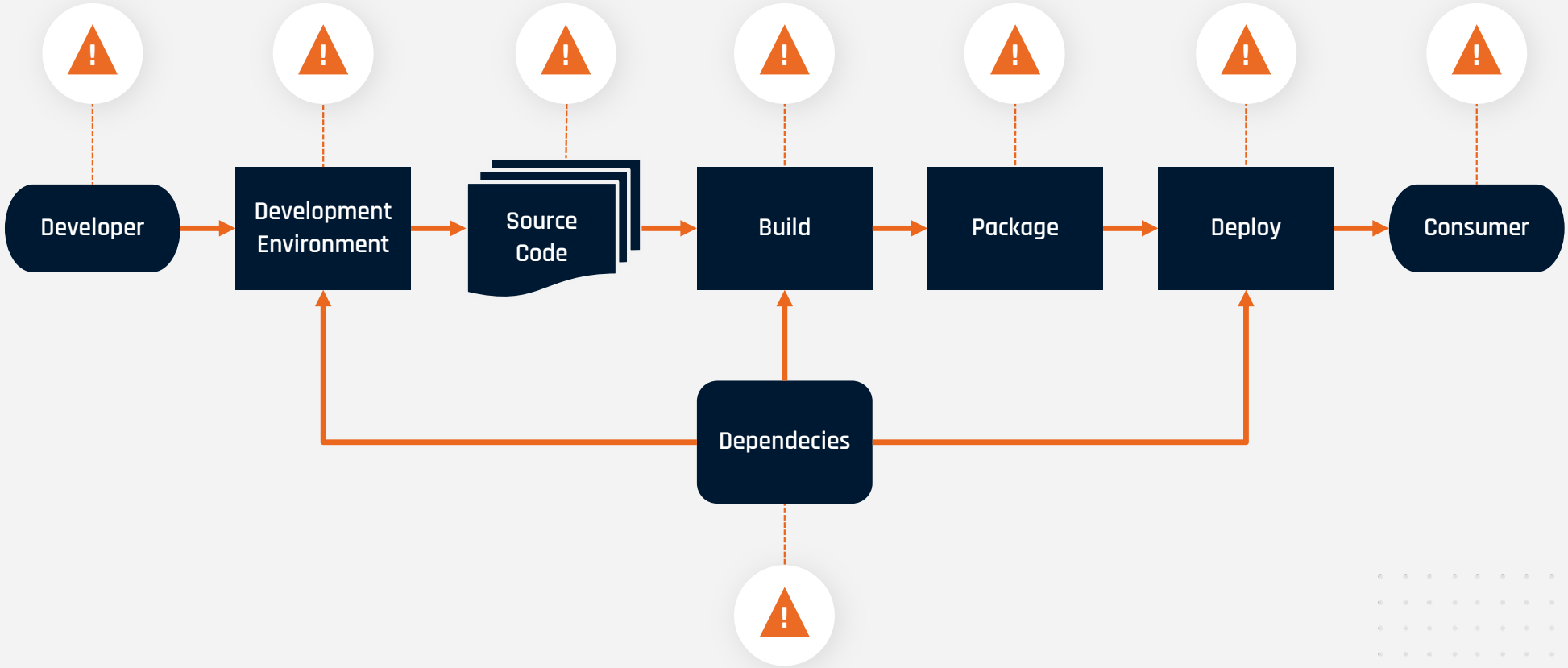
Supply chain attack:
event-stream@3.3.6 -> flatmap-stream@0.1.1

Code execution in build process targeting copay

Harvest the user's bitcoin and private keys

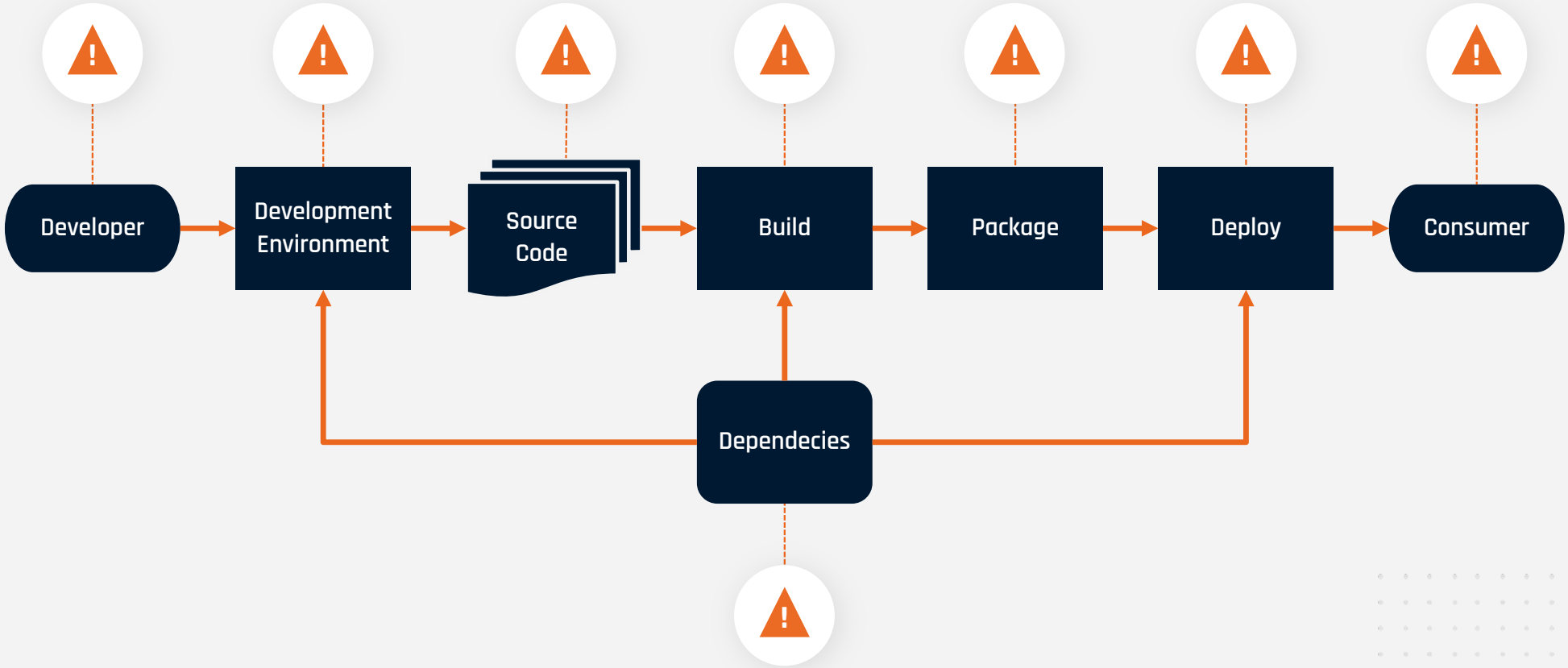


Attack vectors

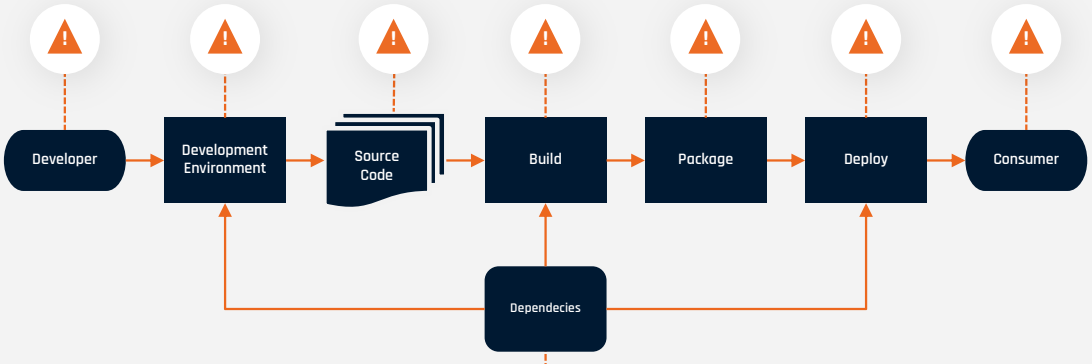



gatsby|
show
Remaining packages: 1
view source code
share to twitter
become a patron
NoteBook

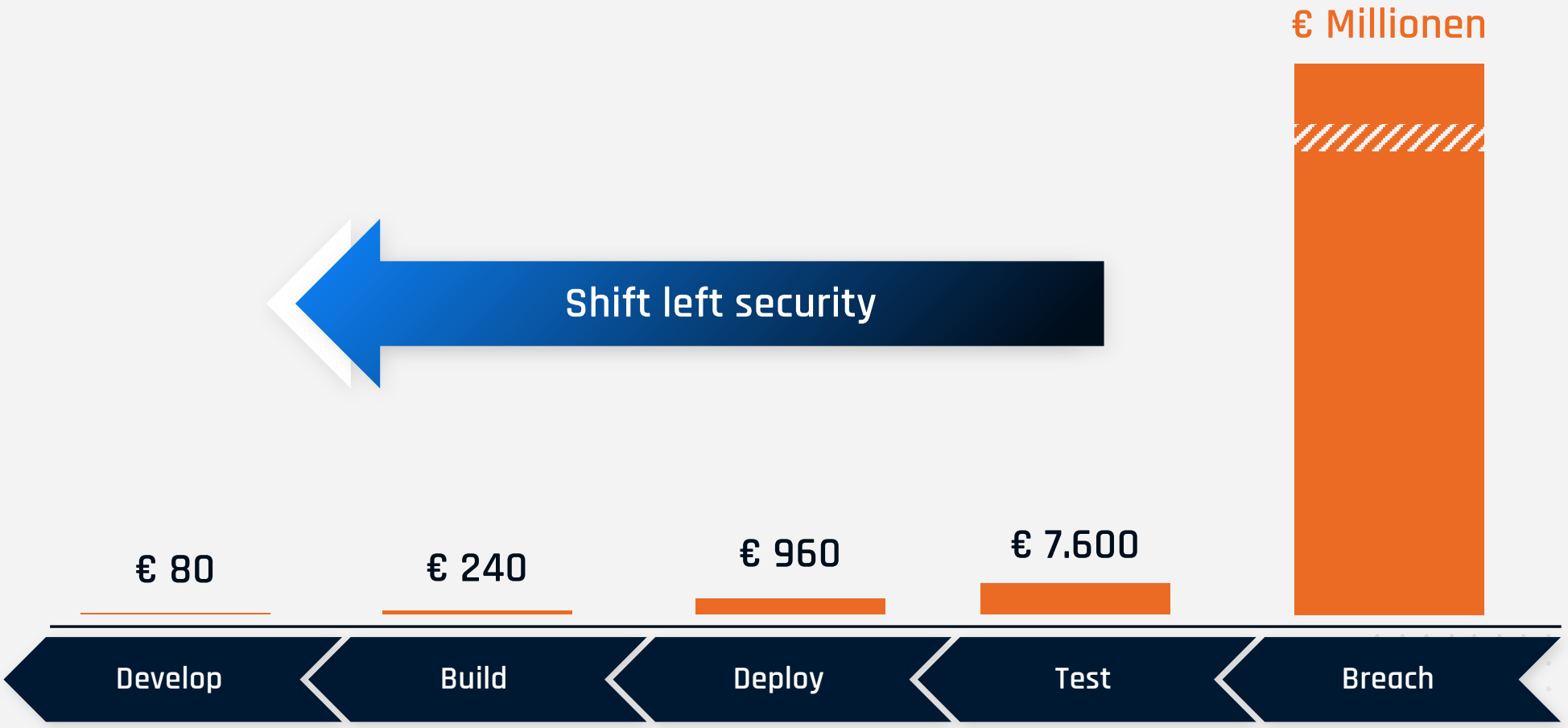
Attack vectors



Attack vectors



Costs for fixing a security vulnerability

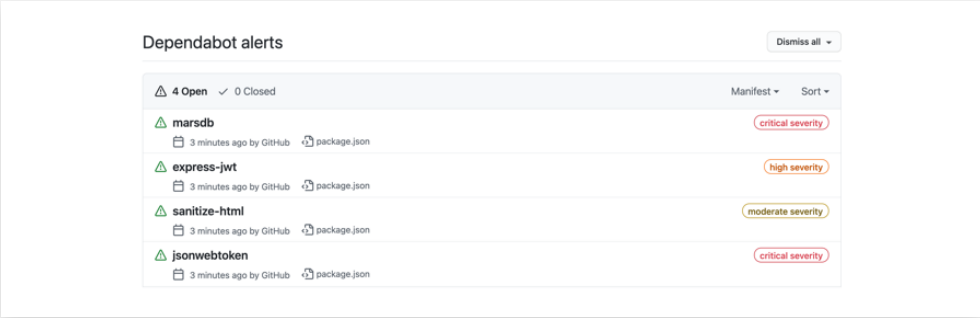


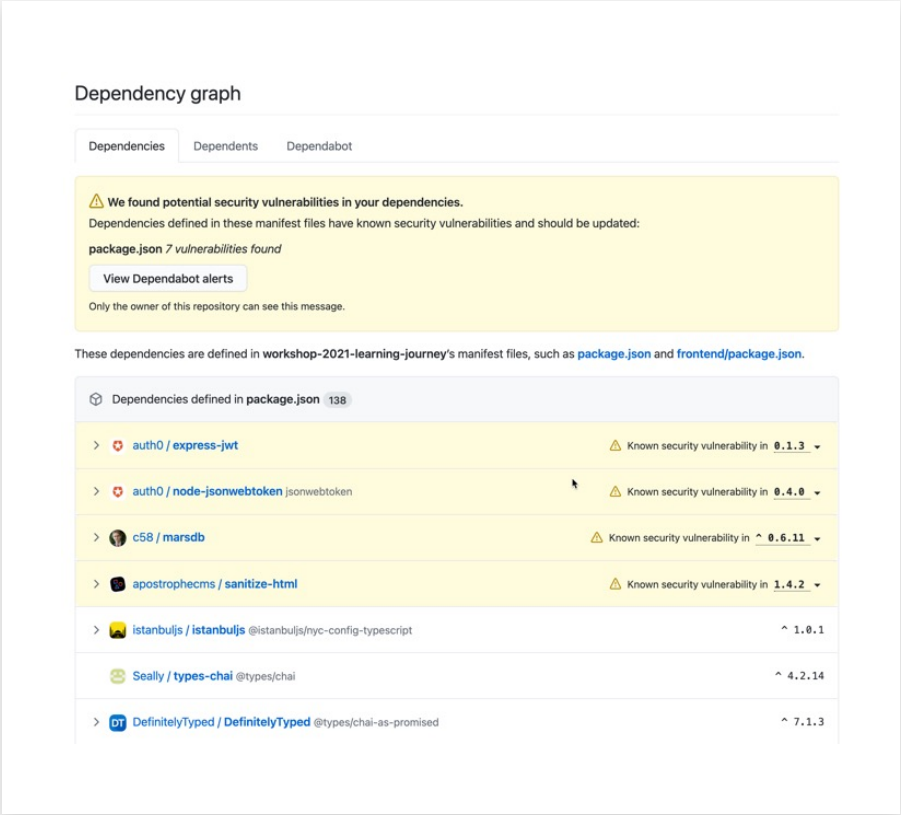
Source: Ponemon Institute Cost of a Data Breach 2020

Supply Chain Security

Software Composition Analysis (SCA)

- GitHub (Dependency-Graph/Dependabot)
- anchore (https://anchore.com/)
- Dependency-Track (https://dependencytrack.org/)

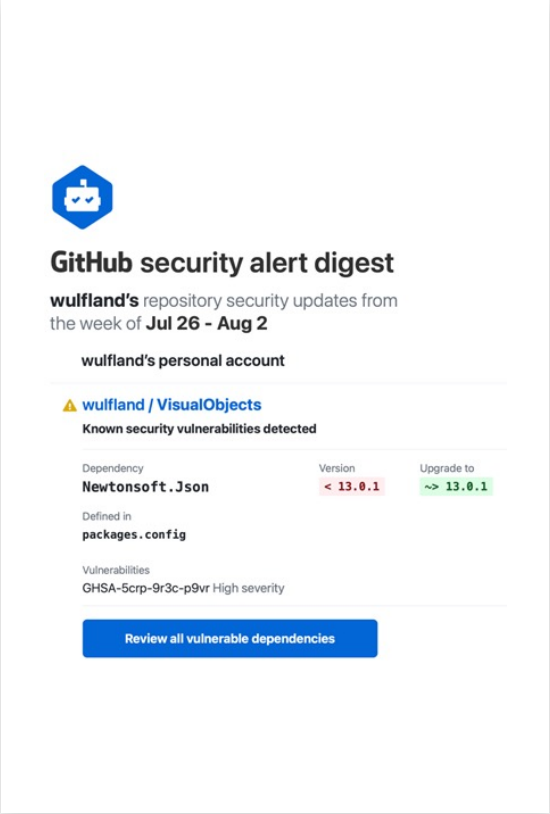


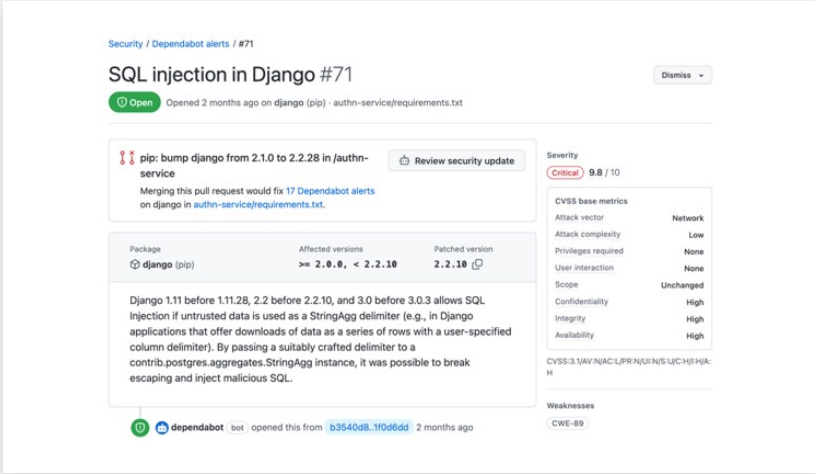


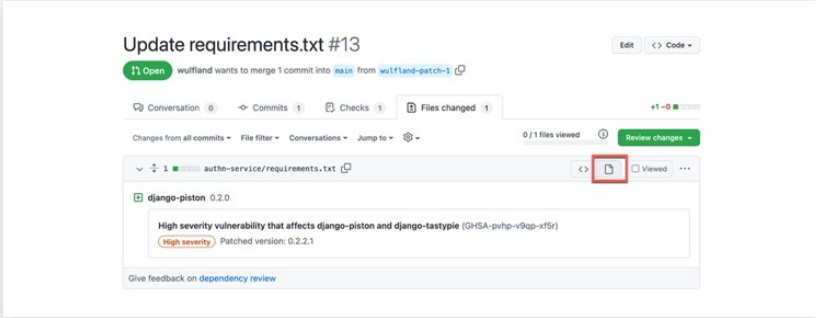
Dependency Management

GitHub Dependency graph

- Dependabot **alerts**
- Dependabot **security updates**
- Dependabot **version updates**







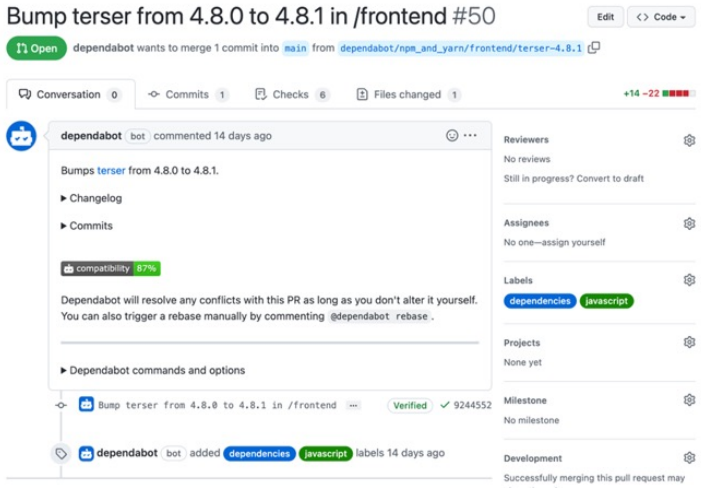
Dependency Management

GitHub Dependency graph

- Dependabot **alerts**
- Dependabot **security updates**
- Dependabot **version updates**

```yaml
jobs:
  dependabot:
    runs-on: ubuntu-latest
    if: ${{ github.actor == 'dependabot[bot]' }}
    steps:
      - name: Dependabot metadata
        id: dependabot-metadata
        uses: dependabot/fetch-metadata@v1.1.1
        with:
          github-token: "${{ secrets.GITHUB_TOKEN }}"

      - name: Enable auto-merge for all patch versions
        if: ${{steps.metadata.outputs.update-type == 'version-update:semver-patch'}}
        run: gh pr merge --auto --merge "$PR_URL"
        env:
          PR_URL: ${{github.event.pull_request.html_url}}
          GITHUB_TOKEN: ${{secrets.GITHUB_TOKEN}}
```



Artifact attestations to establish provenance for builds

actions/attest-sbom: Action for generating SBOM attestations for workflow artifacts (github.com)

actions/attest: Action for generating attestations for workflow artifacts (github.com)

Using artifact attestations to establish provenance for builds - GitHub Docs

The SLSA framework is an industry standard used to evaluate supply chain security. It is organized into levels. Each level represents an increasing degree of security and trustworthiness for a software supply chain. Artifact attestations provides SLSA v1.0 Build Level 2.

Demo: Demo: Dependabot

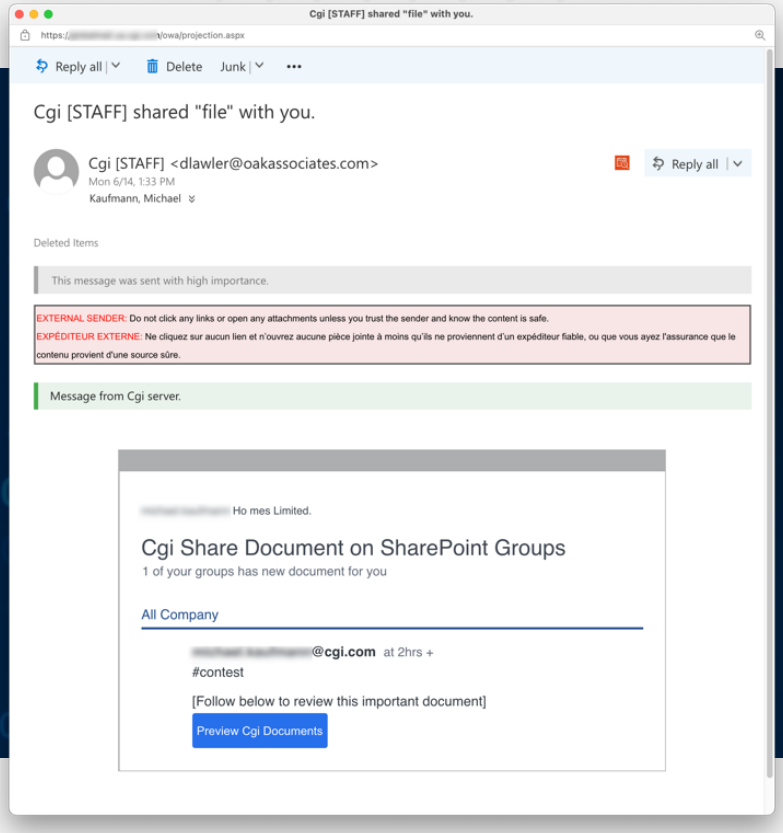
Developer Security

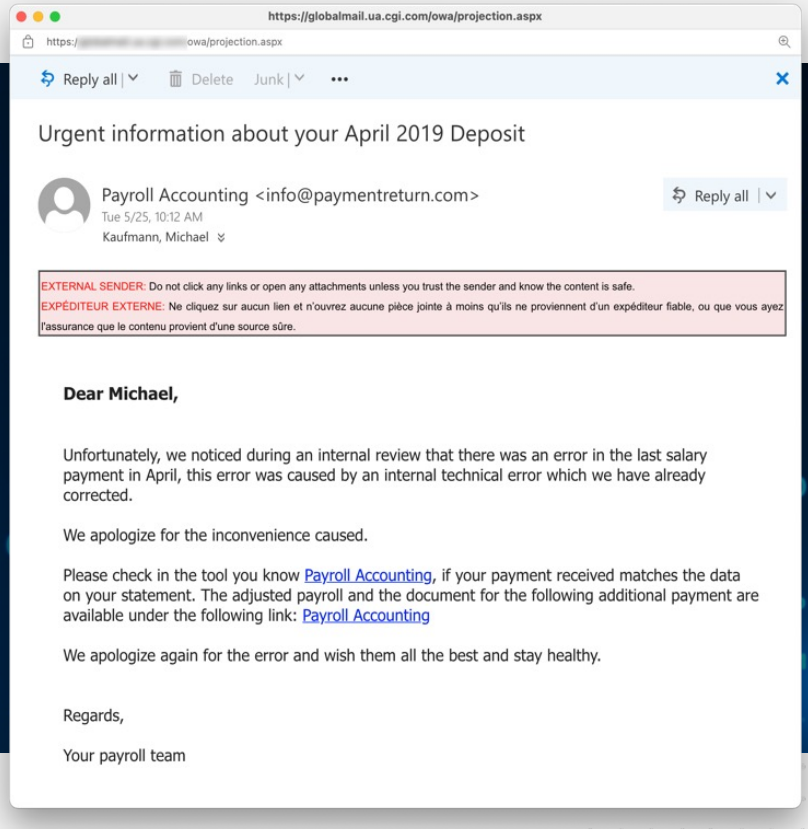
Attacking developers

- Phishing / Spear Phishing
- Social engineering
- Unsecured connections to test systems

"A developer is just a normal employee – that works as local admin, can push and execute code on various systems in minutes, and often runs unsecured web servers."

Phishing


Cgi [STAFF] shared "file" with you.
Reply all | Delete Junk |
Cgi [STAFF] shared "file" with you.
Cgi [STAFF] <dlawler@oakassociates.com>
Mon 6/14, 1:33 PM
Kaufmann, Michael
Reply all
Deleted Items
This message was sent with high importance.
EXTERNAL SENDER: Do not click any links or open any attachments unless you trust the sender and know the content is safe.
EXPÉDITEUR EXTERNE: Ne cliquez sur aucun lien et n'ouvrez aucune pièce jointe à moins qu'ils ne proviennent d'un expéditeur fiable, ou que vous ayez l'assurance que le contenu provient d'une source sûre.
Message from Cgi server.
Ho mes Limited.
Cgi Share Document on SharePoint Groups
1 of your groups has new document for you
All Company
@cgi.com at 2hrs +
#contest
[Follow below to review this important document]
Preview Cgi Documents


https://globalmail.ua.cgi.com/owa/projection.aspx
Reply all | Delete Junk |
Urgent information about your April 2019 Deposit
Payroll Accounting <info@paymentreturn.com>
Tue 5/25, 10:12 AM
Kaufmann, Michael
Reply all
EXTERNAL SENDER: Do not click any links or open any attachments unless you trust the sender and know the content is safe.
EXPÉDITEUR EXTERNE: Ne cliquez sur aucun lien et n'ouvrez aucune pièce jointe à moins qu'ils ne proviennent d'un expéditeur fiable, ou que vous ayez l'assurance que le contenu provient d'une source sûre.
Dear Michael,
Unfortunately, we noticed during an internal review that there was an error in the last salary payment in April, this error was caused by an internal technical error which we have already corrected.
We apologize for the inconvenience caused.
Please check in the tool you know Payroll Accounting, if your payment received matches the data on your statement. The adjusted payroll and the document for the following additional payment are available under the following link: Payroll Accounting
We apologize again for the error and wish them all the best and stay healthy.
Regards,
Your payroll team

Attacking developers

Typo squatting

Namespace shadowing

Typo squatting

```
$ npm install crossenv
```

Steals all your environment variables

```
$ npm install cross-env
```

Normal package

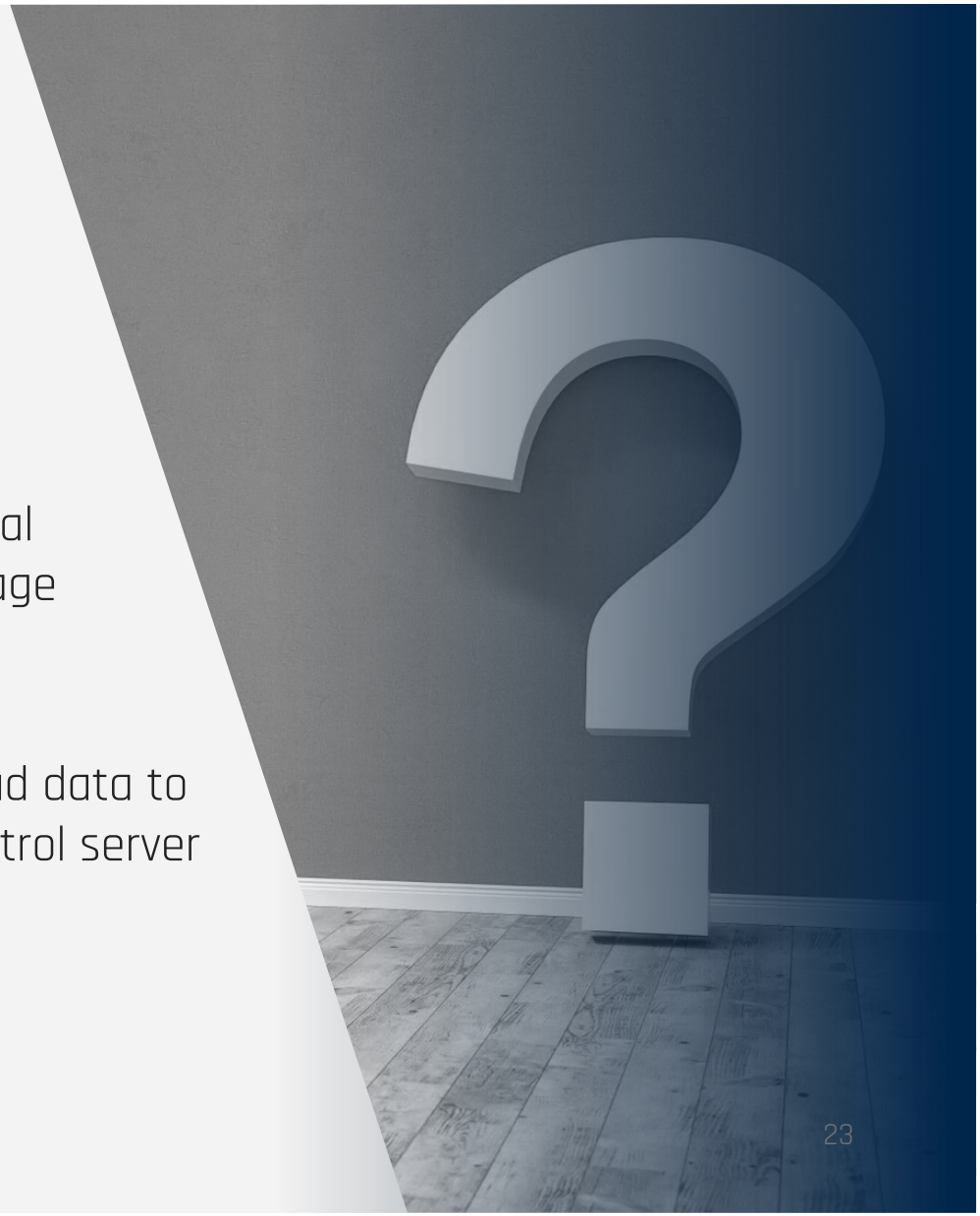
Namespace shadowing

```
$ npm install @azure/core-tracing
```

Normal package

```
$ npm install core-tracing
```

Upload data to a control server



Credentials Developer

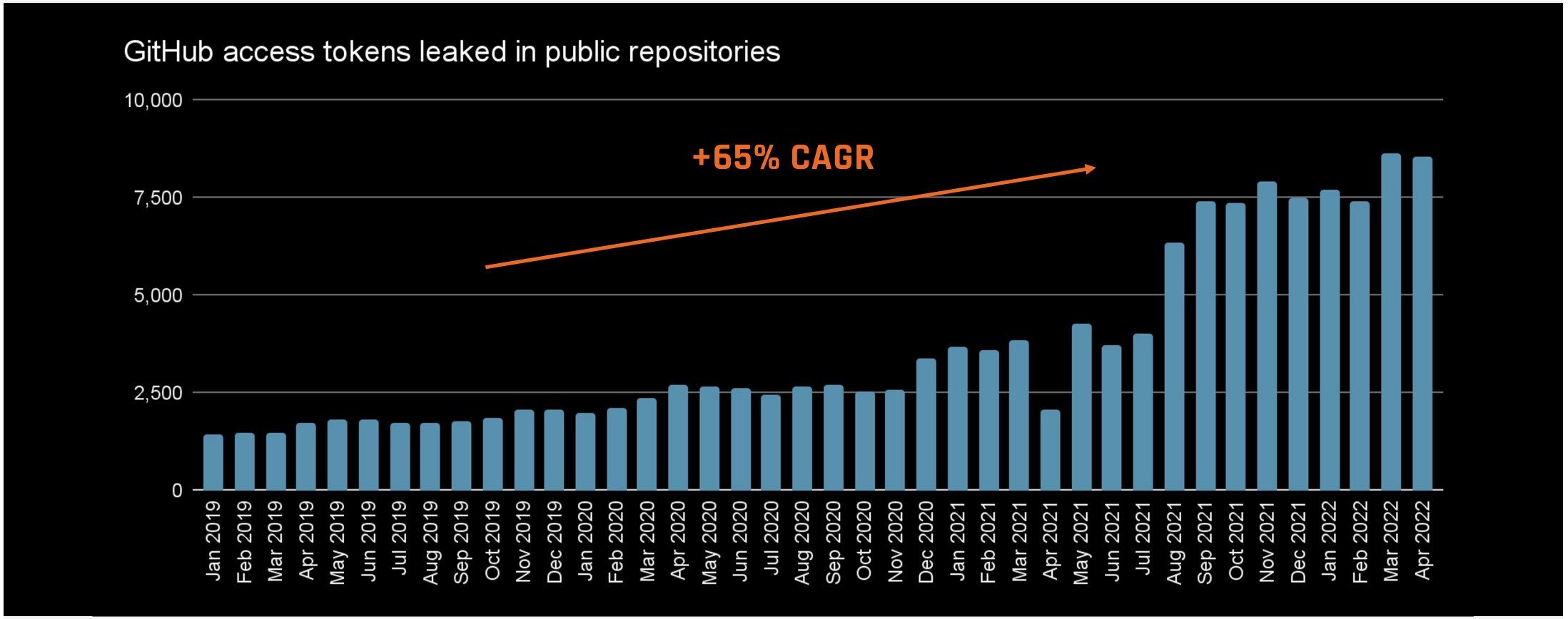
E-Mail	Spear phishing
Access machines	Log on, Mimikatz
Source	Inject code
Pipeline	Execute code / scripts
Access test environment	Test against prod?
Access prod?	

Virtual DevEnv

- Virtual development environments
- Specific for project
- No local admin rights
- Codespaces / DevBox
- Secret scanning

Secret Scanning

We're seeing more credential leaks than ever



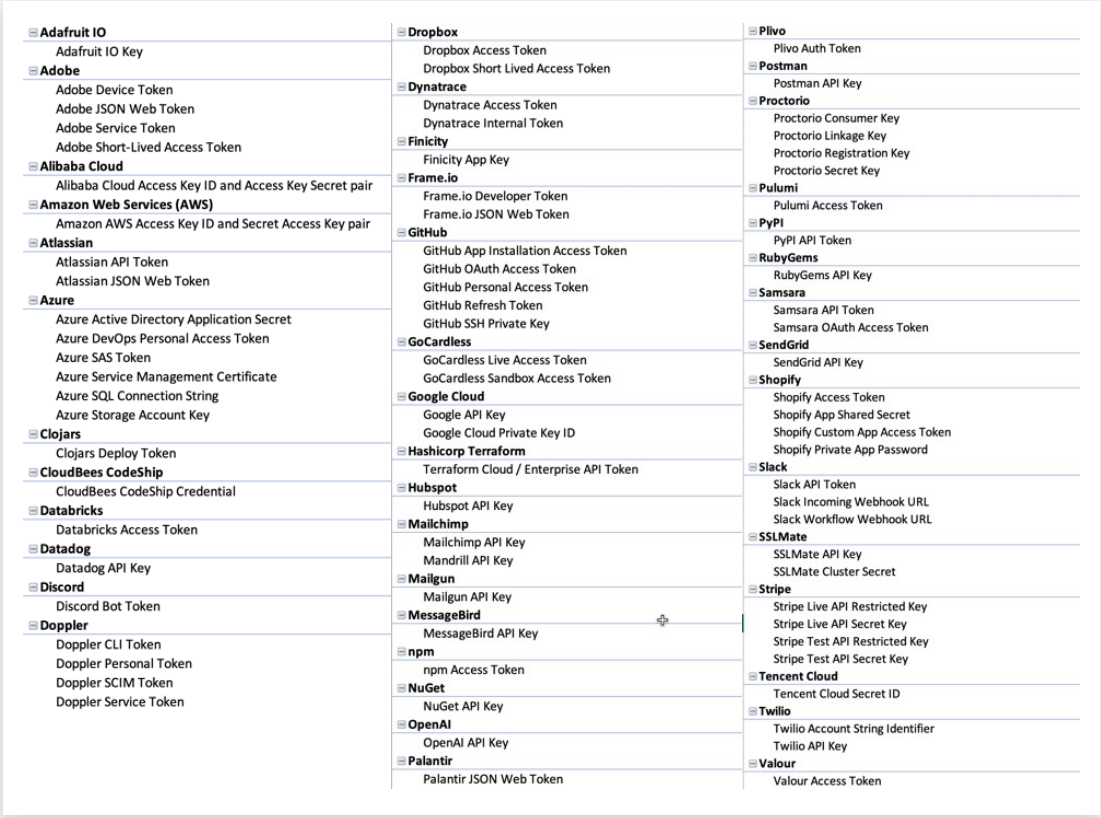
Secret Scanning

Code

- GitHub Secret Scanning
- gitLeaks
- SpectralOps
- Git-Secrets
- Whispers
- Gittyleaks
- Git-all-secrets
- ...

Fileshare

- Bash/PowerShell
- Dumpster



Adafruit IO	**Dropbox**	**Plivo**
Adafruit IO Key	Dropbox Access Token	Plivo Auth Token
Adobe	Dropbox Short Lived Access Token	**Postman**
Adobe Device Token	**Dynatrace**	Postman API Key
Adobe JSON Web Token	Dynatrace Access Token	**Proctorio**
Adobe Service Token	Dynatrace Internal Token	Proctorio Consumer Key
Adobe Short-Lived Access Token	**Finicity**	Proctorio Linkage Key
Alibaba Cloud	Finicity App Key	Proctorio Registration Key
Alibaba Cloud Access Key ID and Access Key Secret pair	**Frame.io**	Proctorio Secret Key
Amazon Web Services (AWS)	Frame.io Developer Token	**Pulumi**
Amazon AWS Access Key ID and Secret Access Key pair	Frame.io JSON Web Token	Pulumi Access Token
Atlassian	**GitHub**	**PyPI**
Atlassian API Token	GitHub App Installation Access Token	PyPI API Token
Atlassian JSON Web Token	GitHub OAuth Access Token	**RubyGems**
Azure	GitHub Personal Access Token	RubyGems API Key
Azure Active Directory Application Secret	GitHub Refresh Token	**Samsara**
Azure DevOps Personal Access Token	GitHub SSH Private Key	Samsara API Token
Azure SAS Token	**GoCardless**	Samsara OAuth Access Token
Azure Service Management Certificate	GoCardless Live Access Token	**SendGrid**
Azure SQL Connection String	GoCardless Sandbox Access Token	SendGrid API Key
Azure Storage Account Key	**Google Cloud**	**Shopify**
Clojars	Google API Key	Shopify Access Token
Clojars Deploy Token	Google Cloud Private Key ID	Shopify App Shared Secret
CloudBees CodeShip	**Hashicorp Terraform**	Shopify Custom App Access Token
CloudBees CodeShip Credential	Terraform Cloud / Enterprise API Token	Shopify Private App Password
Databricks	**Hubspot**	**Slack**
Databricks Access Token	Hubspot API Key	Slack API Token
Datadog	**Mailchimp**	Slack Incoming Webhook URL
Datadog API Key	Mailchimp API Key	Slack Workflow Webhook URL
Discord	Mandrill API Key	**SSLMate**
Discord Bot Token	**Mailgun**	SSLMate API Key
Doppler	Mailgun API Key	SSLMate Cluster Secret
Doppler CLI Token	**MessageBird**	**Stripe**
Doppler Personal Token	MessageBird API Key	Stripe Live API Restricted Key
Doppler SCIM Token	**npm**	Stripe Live API Secret Key
Doppler Service Token	npm Access Token	Stripe Test API Restricted Key
	NuGet	Stripe Test API Secret Key
	NuGet API Key	**Tencent Cloud**
	OpenAI	Tencent Cloud Secret ID
	OpenAI API Key	**Twilio**
	Palantir	Twilio Account String Identifier
	Palantir JSON Web Token	Twilio API Key
		Valour
		Valour Access Token

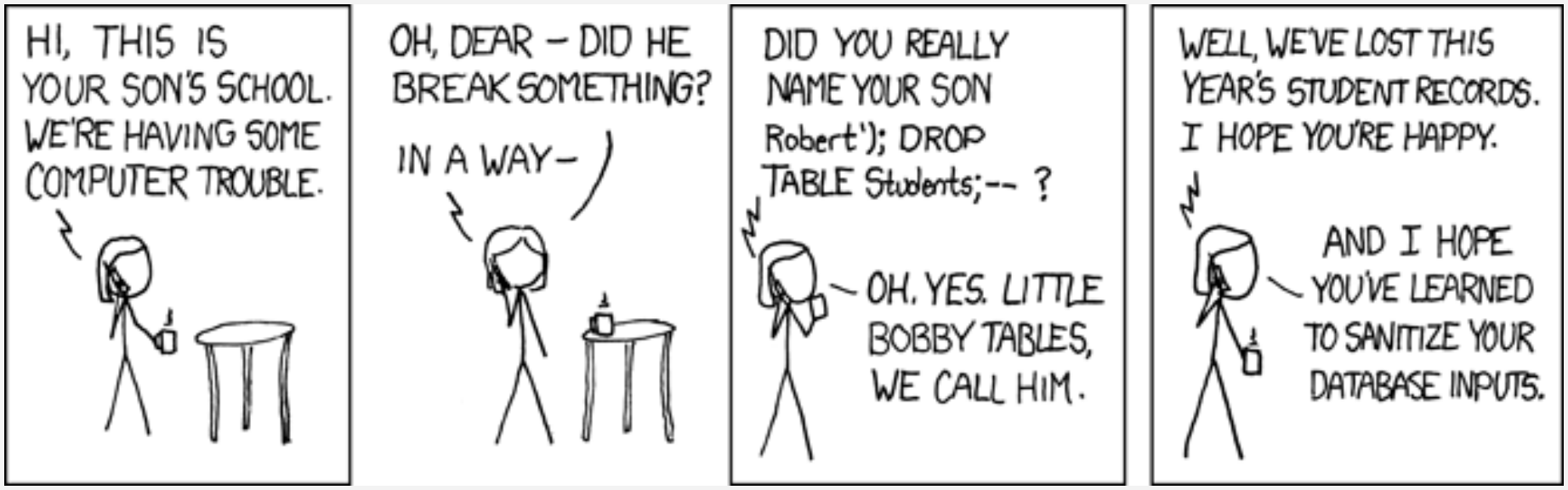
Demo: Secret Scanning

Code Security

SQL Injection

```
txtUserId = getRequestString("UserId");
txtSQL = "SELECT * FROM Users WHERE UserId = " + txtUserId;
```

User ID: 105; DROP TABLE Suppliers



(Source: https://xkcd.com/327/)

4 years

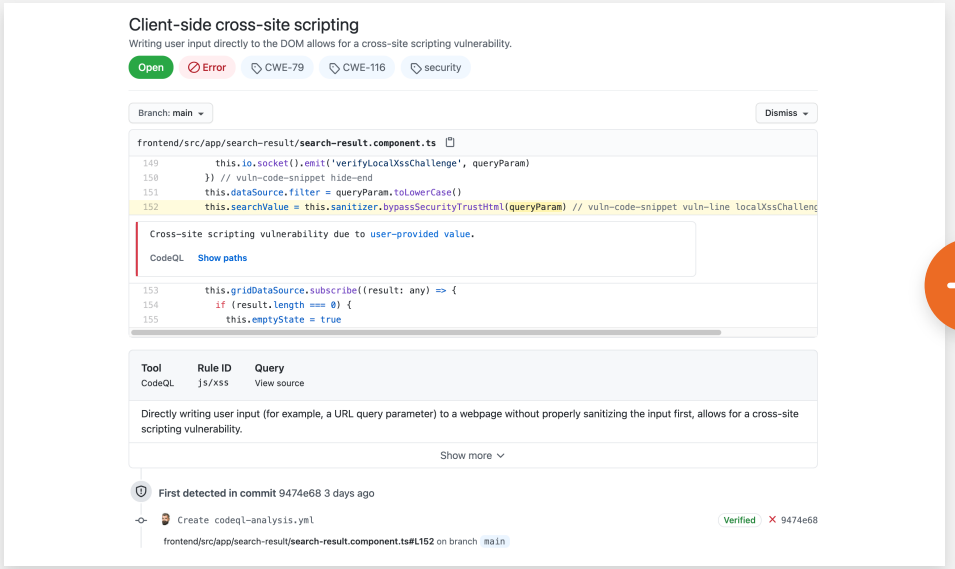
On average, vulnerabilities go undetected for four years before being identified.
Sometimes, even longer than that - Log4j was vulnerable for ~7 years

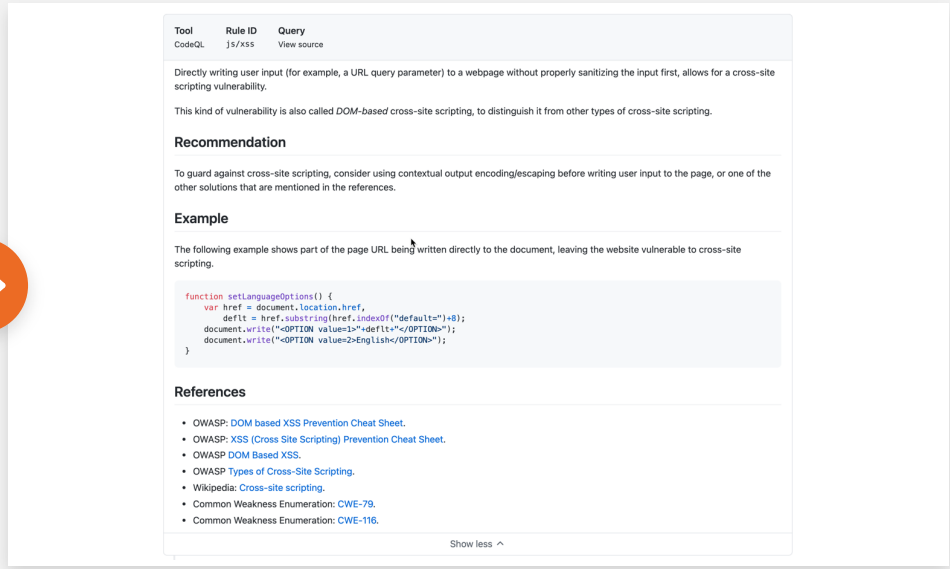
Static Application Security Testing (SAST)

Analyze source code, bytecode, or binary code to identify security vulnerabilities in the application's codebase.

Whitebox-Testing

- GitHub Code Analysis
- SonarQube, Fortify, Checkmarx, and Veracode
- Semgrep (https://semgrep.dev/)
- Mobile-Security-Framework (MobSF) (https://github.com/MobSF/Mobile-Security-Framework-MobSF)



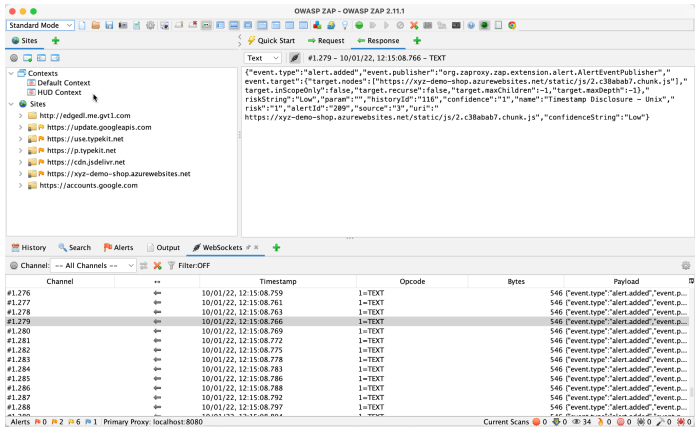


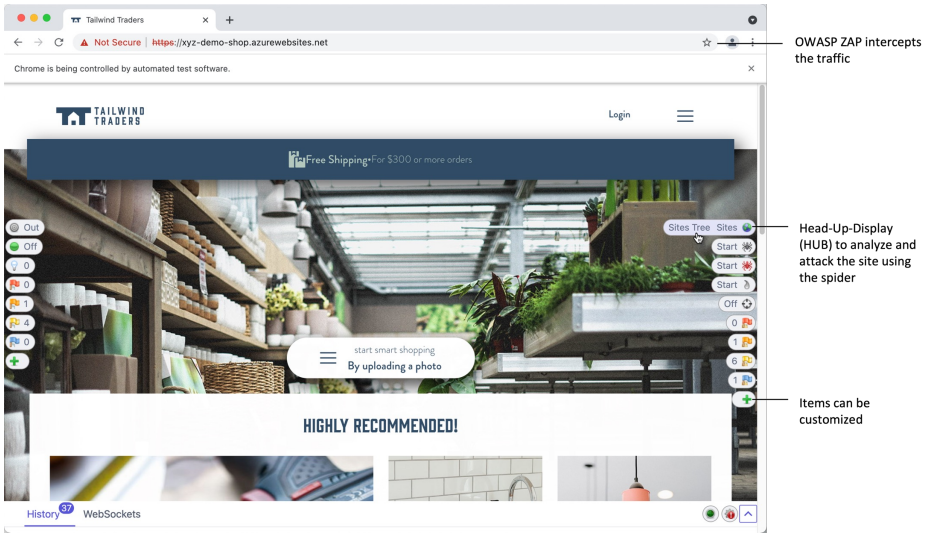
Dynamic Application Security Testing (DAST)

Scans a running application to identify vulnerabilities by sending input requests and analyzing responses.

Blackbox-Testing

- OWASP ZAP (Zed Attack Proxy, https://owasp.org/www-project-zap)
- Burp Suite von PortSwigger (https://portswigger.net/burp)



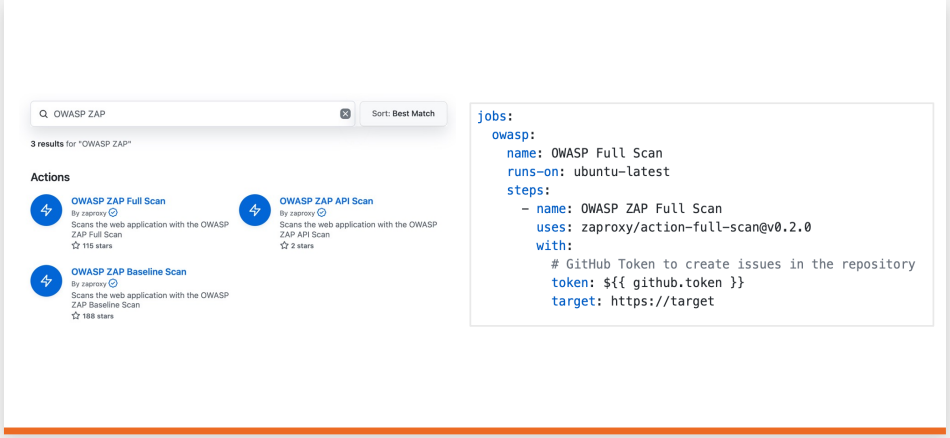


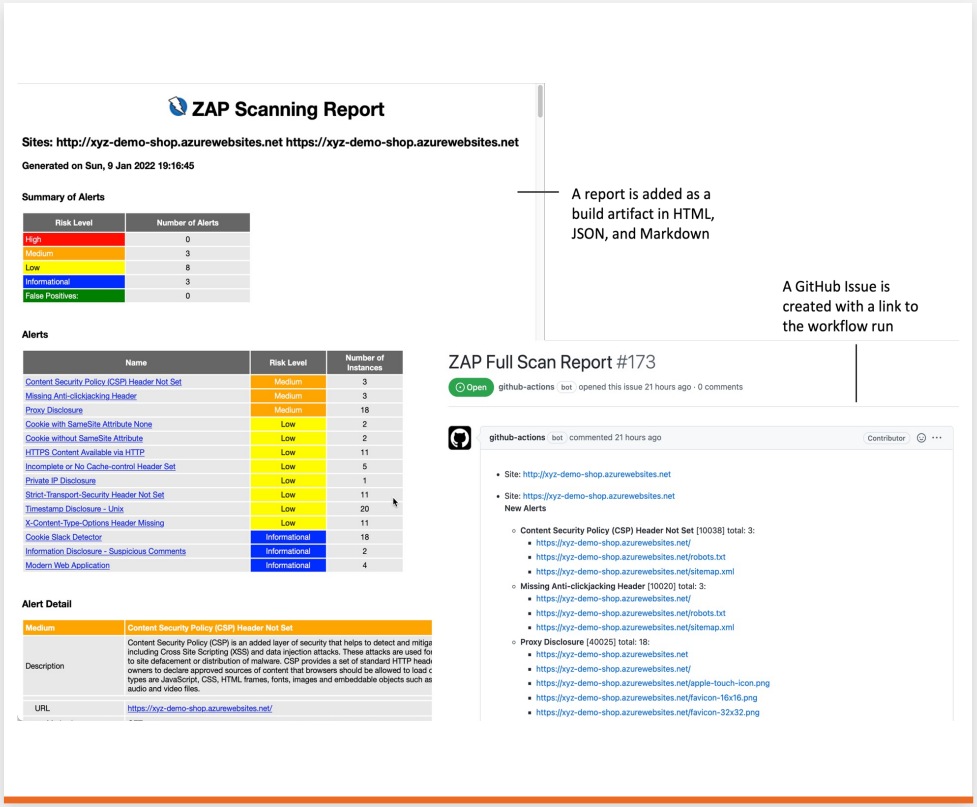
Dynamic Application Security Testing (DAST)

Blackbox-Testing

› OWASP ZAP (Zed Attack Proxy, https://owasp.org/www-project-zap)

› Burp Suite von PortSwigger (https://portswigger.net/burp)



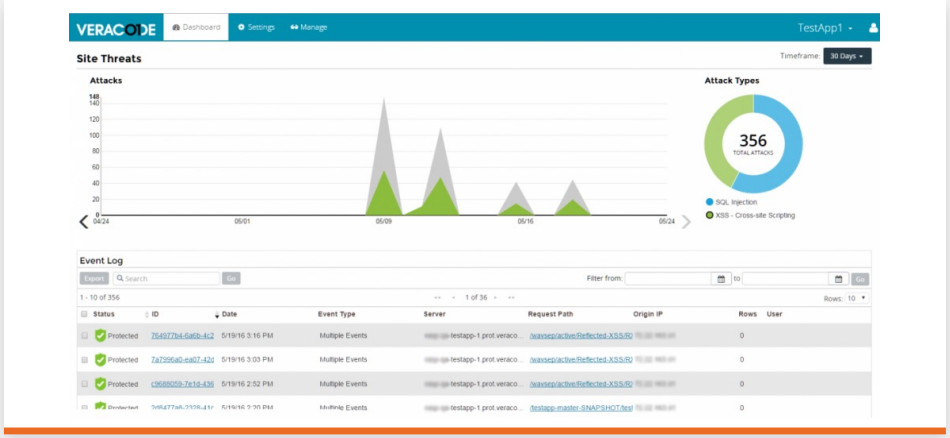


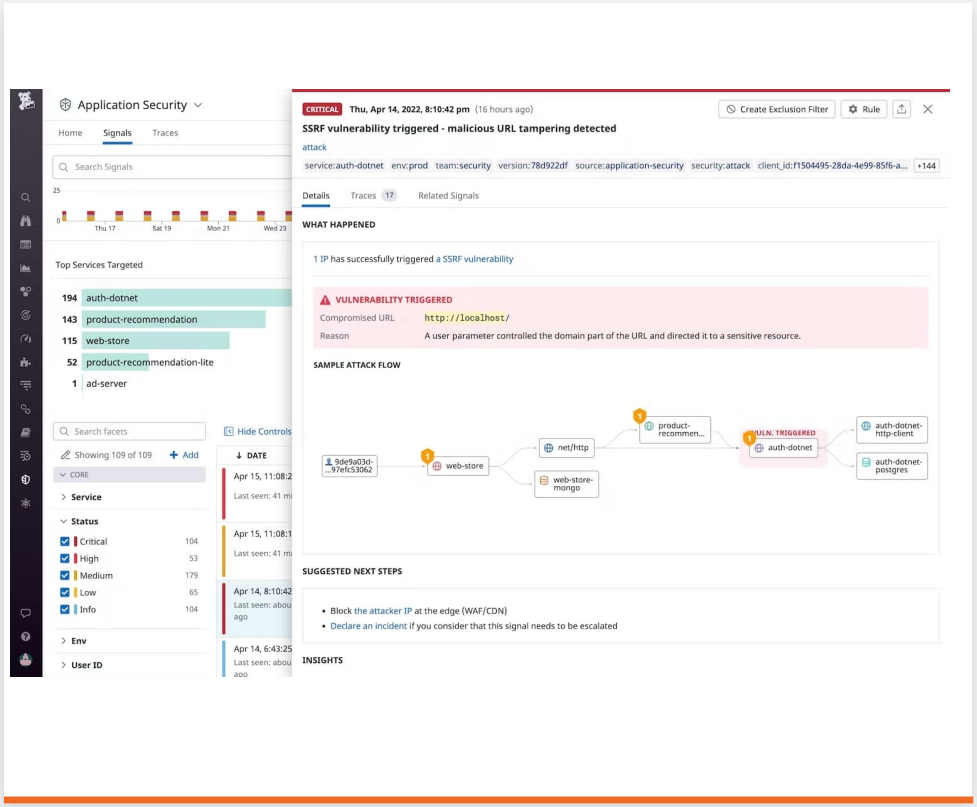
Runtime Application Self-Protection (RASP)

Monitors and protects applications in real-time, detecting and responding to security threats as they occur.

Blackbox-Testing

- Veracode Runtime Protection
- F5 Advanced WAF with RASP
- Datadog Application Security Management





Infrastructure Scanning

Container Vulnerability Analysis (CVA) / Container Security Analysis (CSA)

Open source:

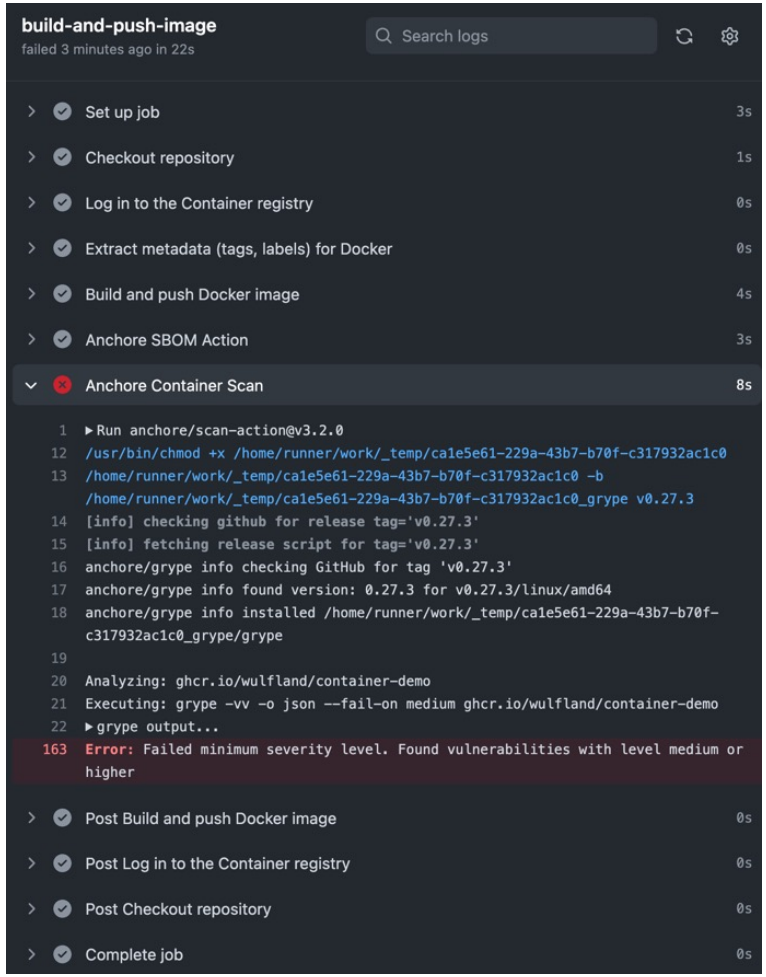
- Anchore gryp
  https://github.com/anchore/grype/
- Clair
  https://quay.github.io/clair/

Commercial:

- WhiteSource
  https://www.whitesourcesoftware.com/solution-for-containers/
- Aqua
  https://www.aquasec.com/products/container-security/

```yaml
- name: Anchore Container Scan
  uses: anchore/scan-action@v3.2.0
  with:
    image: ${{ env.REGISTRY }}/${{ env.IMAGE_NAME }}
    debug: true
```

https://github.com/wulfland/container-demo/actions/runs/2179243137



Infrastructure Scanning

Infrastructure policies

Open source:

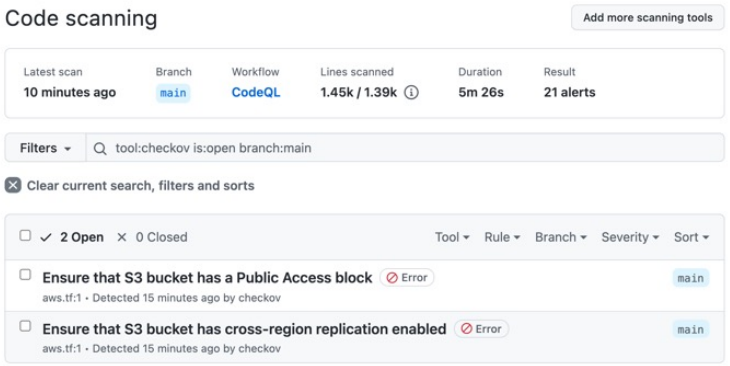
- Checkov
  https://www.aquasec.com/products/container-security/
- OpenVAS

Commercial:

- Defender for Cloud
  https://azure.microsoft.com/en-us/services/defender-for-cloud
- Azure Policy
  https://docs.microsoft.com/de-de/azure/governance/policy/

```yaml
- name: Checkov GitHub Action
  uses: bridgecrewio/checkov-action@master
  with:
    directory: ch15_sec/
    output_format: sarif

- name: Upload SARIF file
  uses: github/codeql-action/upload-sarif@v1
  with:
    sarif_file: results.sarif
  if: always()
```



CodeQL: a revolutionary semantic code engine

- Advanced code analysis engine based on 13 years of research by a 30 person team from Oxford University
- Allows you to query your code's logic to find vulnerabilities
- Queries can be quickly customized to adapt to your specific threat topology
- Community-driven query set powers every project with a world-class security team

Demo: Code analysis (Auto-Fix)


Hackers in movies
vs
Image thanks to Gregor Biswanger @BFreakout
© 2021 Michael Kaufmann @mike_kaufmann

Thank you


Blog : https://writeabout.net


Twitter : @mike_kaufmann


GitHub : @wulfland


LinkedIn : https://www.linkedin.com/in/mikaufmann/