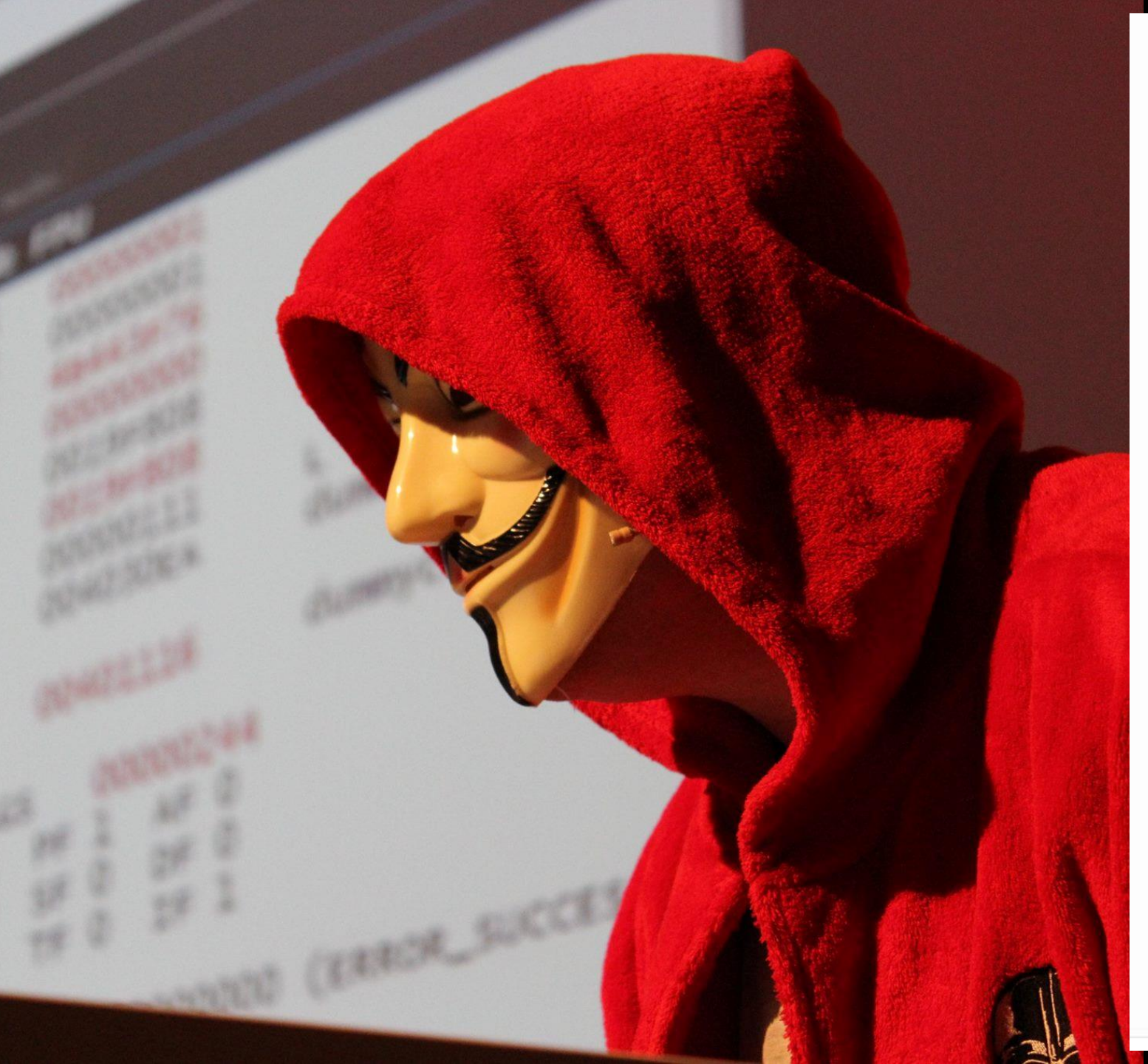
Secure DevOps

#DevSecOps

Seguridad en tiempos de despliegues diarios

Image thanks to Gregor Biswanger @BFreakout



How people think they get hacked

How they really get hacked

Cuando hablamos de seguridad…

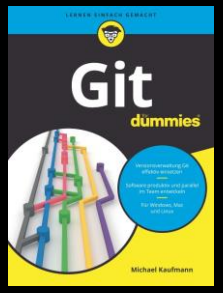
- “Hasta ahora nunca hemos tenido un incidente con la seguridad”
- “Nuestros desarrolladores son expertos”
- “Nuestra empresa no es tan importante como para ser atacada”
- “Estamos bien protegidos”
- “Nuestros cortafuegos son seguros”

Michael Kaufmann

DevOps Consultant & Trainer

@mike_kaufmann
@wulfland
https://writeabout.net

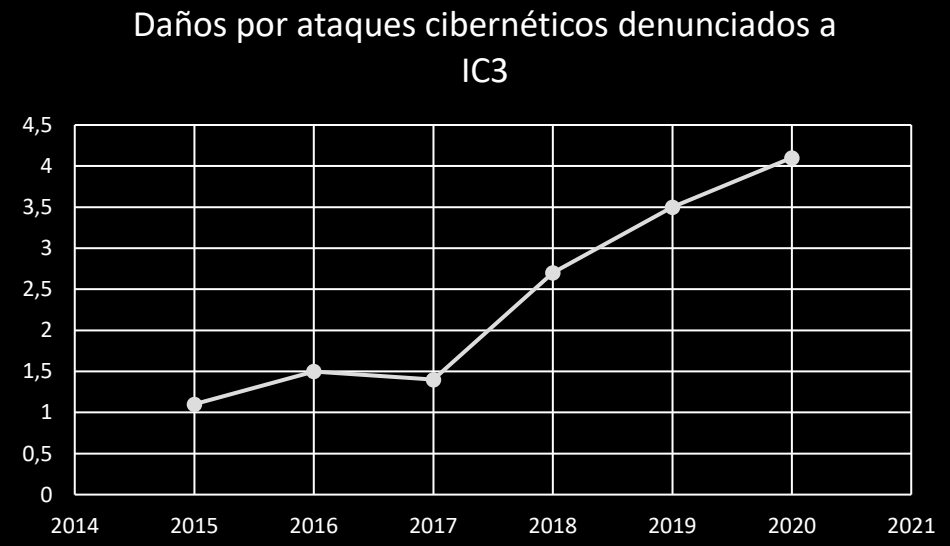
CEO Xpirit Germany GmbH
20 años Desarrollador / Informatico
15 años ALM & DevOps
Microsoft Regional Director & MVP



Git für Dummies, Michael Kaufmann, Wiley-VCH, 2021

Daños causados por los ataques cibernéticos denunciados al IC3:

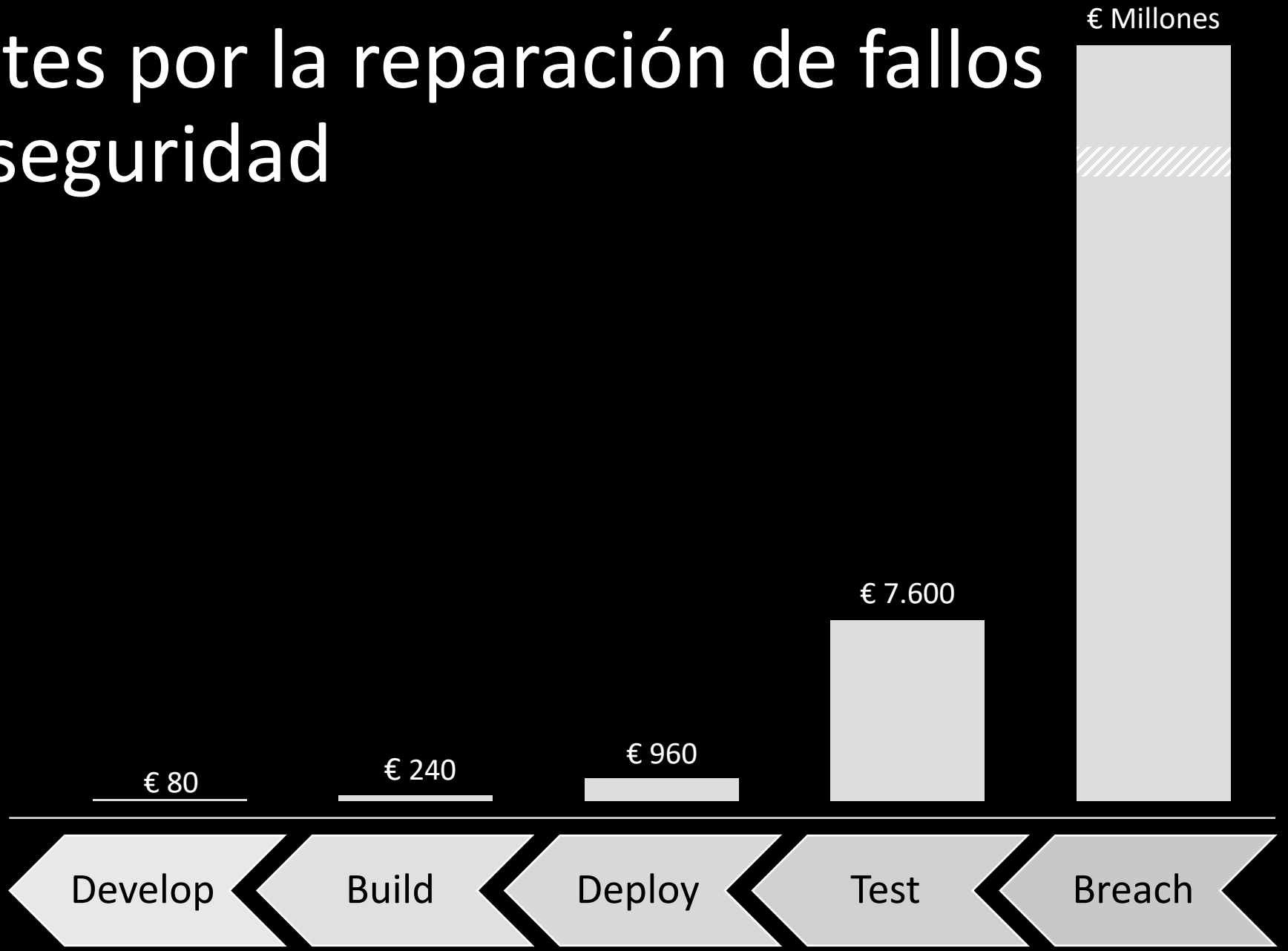
2019	2020
$ 3.500.000.000	$ 4.100.000.000



SolarWinds

March to June 2020: access to government and other systems through a compromised update to SolarWinds' **Orion** software

Costes por la reparación de fallos de seguridad



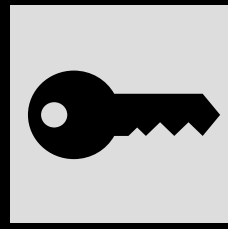
Assume-Breach-Paradigma

“Básicamente es así, si alguien quiere entrar, entra. Está bien. Acéptenlo.”

- Michael V. Hayden

Exgeneral de la Fuerza Aérea de EE.UU. y exdirector de la NSA y de la CIA)

De la prevención de fallos (Prevent-Breach) a la asunción de fallos (Assume-Breach)



Prevent-Breach

- Capas de seguridad (Firewall, VPN, DMZ)
- Análisis de riesgos
- Modelado de amenazas
- Revisiones de la arquitectura
- Revisiones del código
- Pruebas de seguridad

Assume-Breach

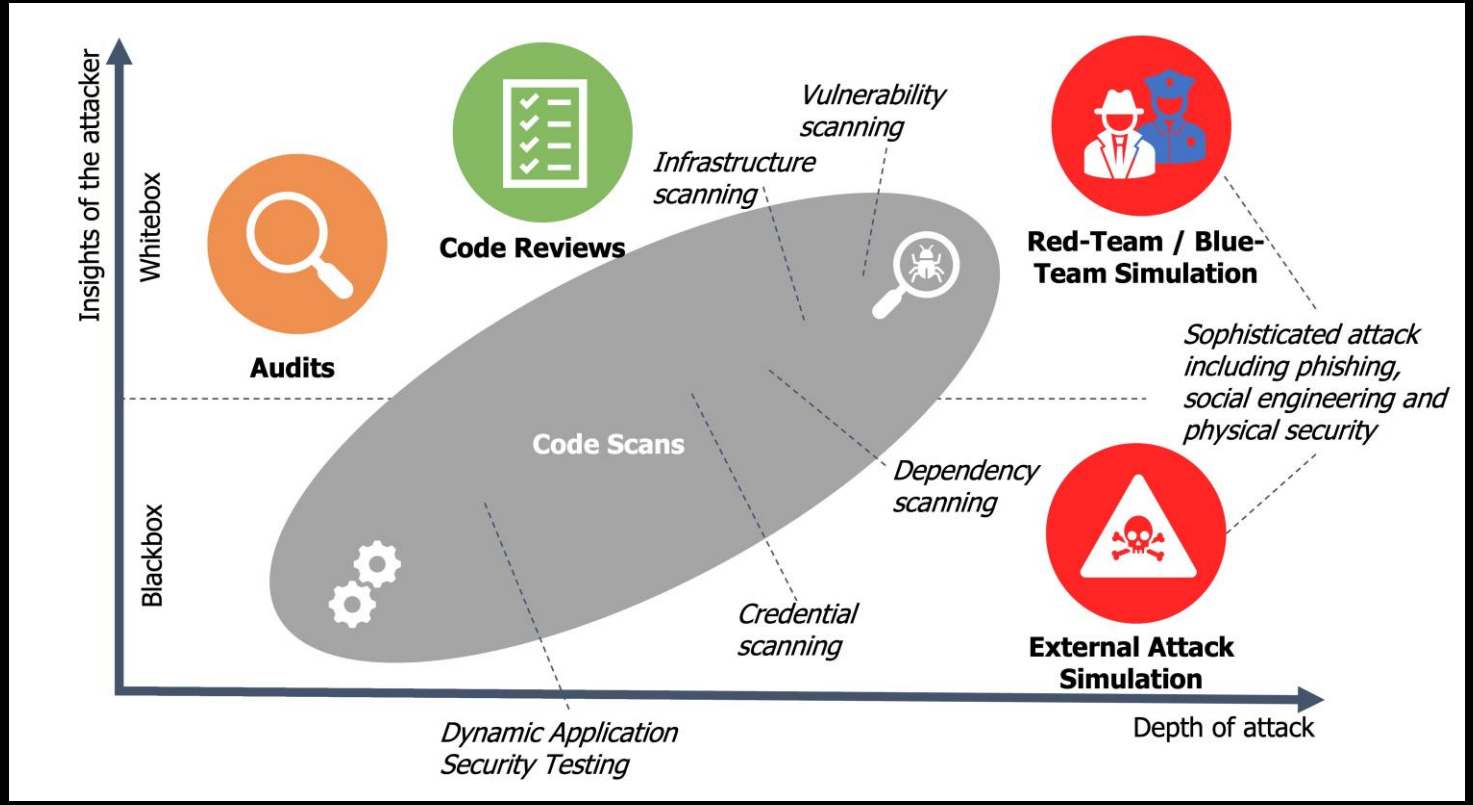
- Ejercicios Equipo Rojo | Equipo Azul (juegos de guerra)
- Monitorización central (SIEM)
- Pruebas de penetración en el entorno productivo
- Identificación de irregularidades

Zero-Trust-Policy

- Todos los sistemas, también los internos, deben asegurarse de tal modo como si fueran accesibles a través de Internet: MFA activado, encriptado HTTPS, sistema siempre parcheado, etc.
- Los empleados solo deben contar con los accesos absolutamente necesarios mientras sean necesarios. Deberán evitarse los derechos de administrador locales.
- Nadie debe tener acceso a los sistemas de producción con sus datos de accesos normales. Ni siquiera las cuentas de prueba que se utilizan en los entornos de desarrollo y de prueba deben tener acceso a la producción.

Equipo Rojo | Equipo Azul

- Objetivo del juego
- Alineación del equipo
- Duración del juego



Equipo Rojo | Equipo Azul

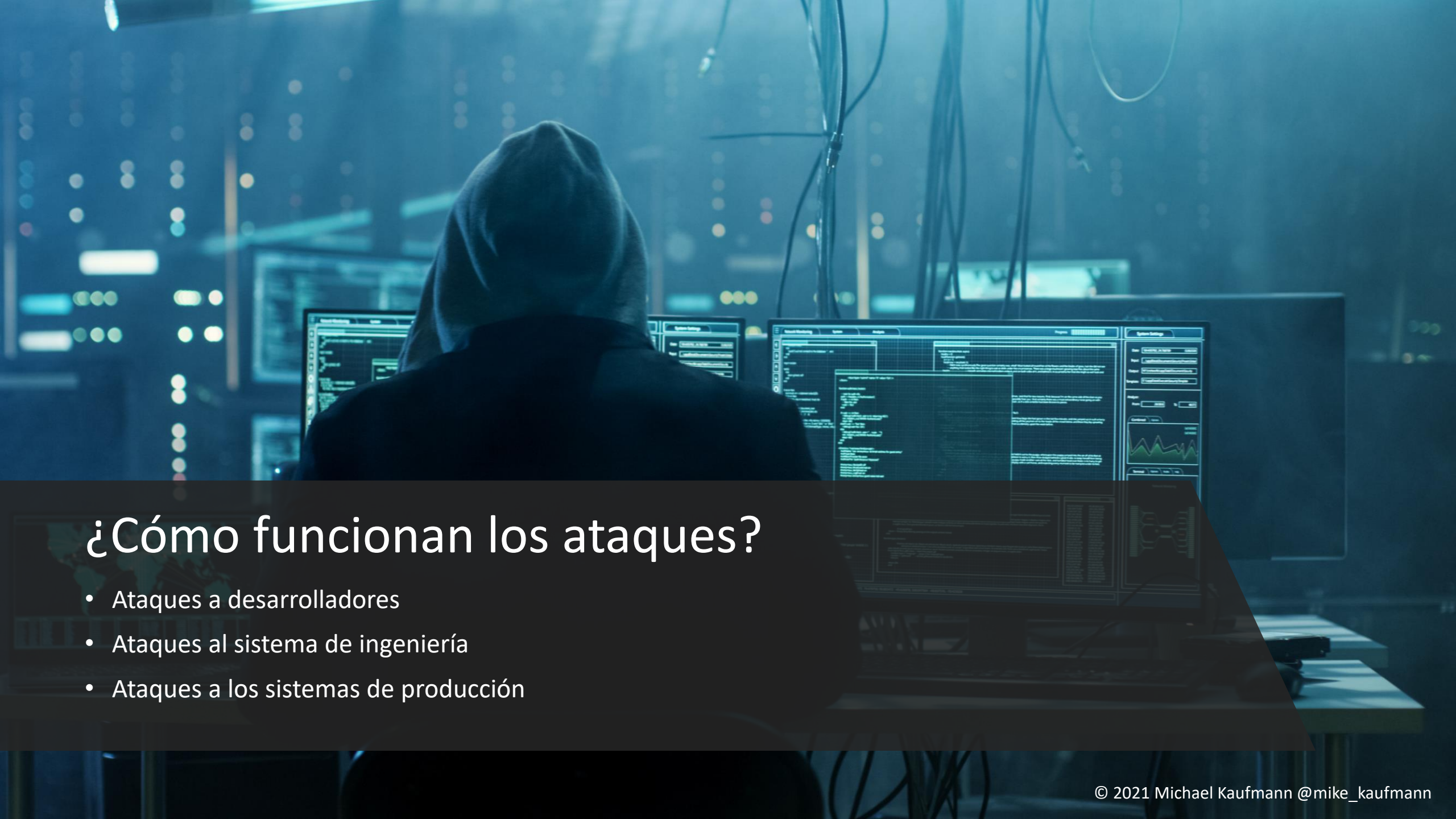
Código de comportamiento:

- No causar ningún daño real
- No hacer más de lo necesario para alcanzar el objetivo
- Ataques físicos dentro de lo normal (no amenazar a nadie, no robar o imprimir tarjetas de acceso, etc. “Tener sentido común”)
- No exponer nombres de personas que hayan sido comprometidas

Equipo Rojo | Equipo Azul

Reglas de juego:

- No debe perjudicarse la disponibilidad del sistema de producción ➔ el equipo rojo ha ganado
- No deben comprometerse los datos reales de los clientes (pero sí los datos de clientes internos)
- Los datos comprometidos (datos de acceso, datos personales, etc.) deben depositarse de manera segura
- La seguridad del sistema de producción no debe debilitarse excesivamente
- No deben ser ocasionados daños innecesarios
- ¡Prueba!



¿Cómo funcionan los ataques?

- Ataques a desarrolladores
- Ataques al sistema de ingeniería
- Ataques a los sistemas de producción

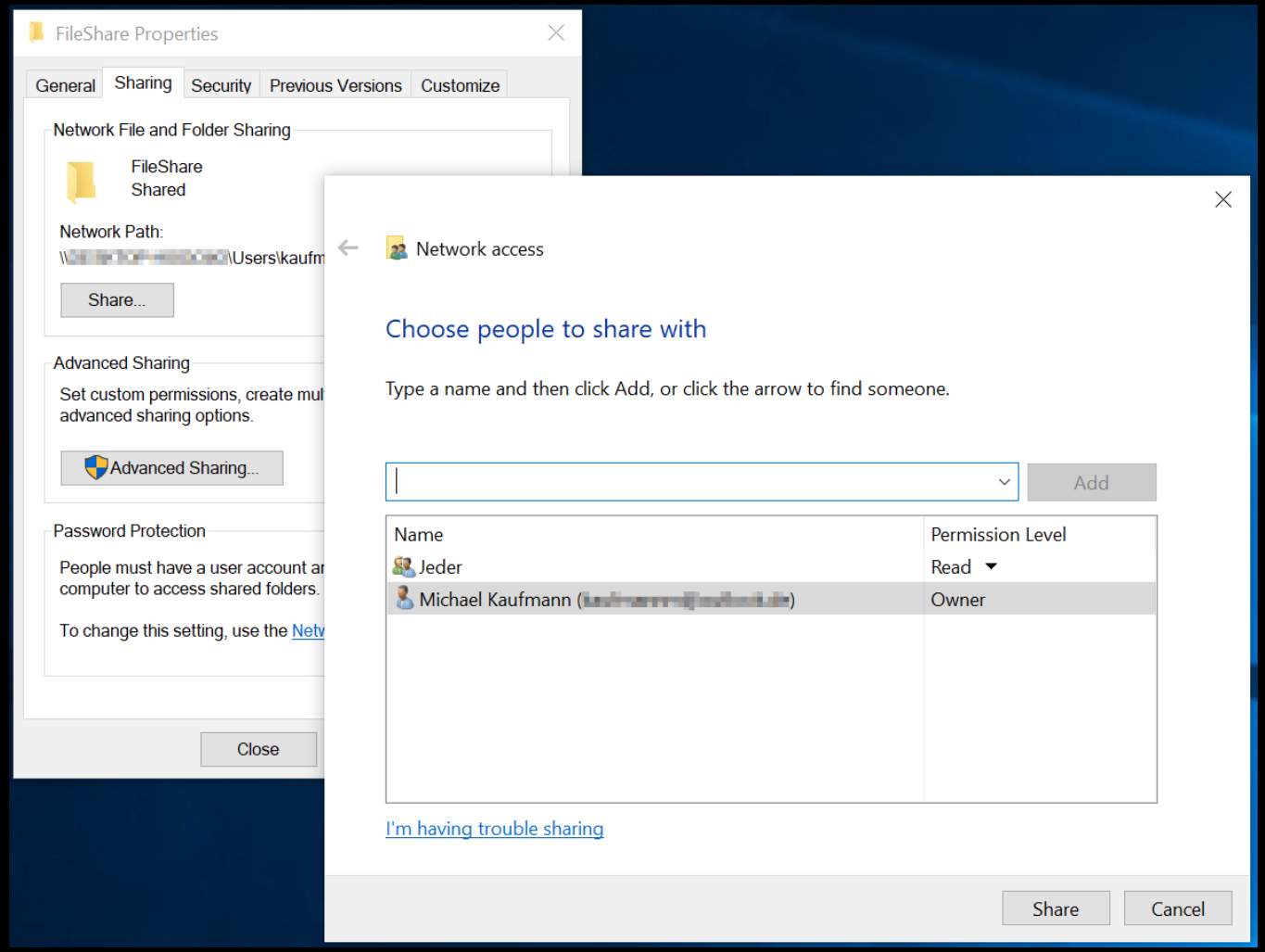
Ataques a desarrolladores

- Intercambios de archivos no protegidos
- Datos de acceso en archivos de texto o en el código
- Mimikatz
- Phishing

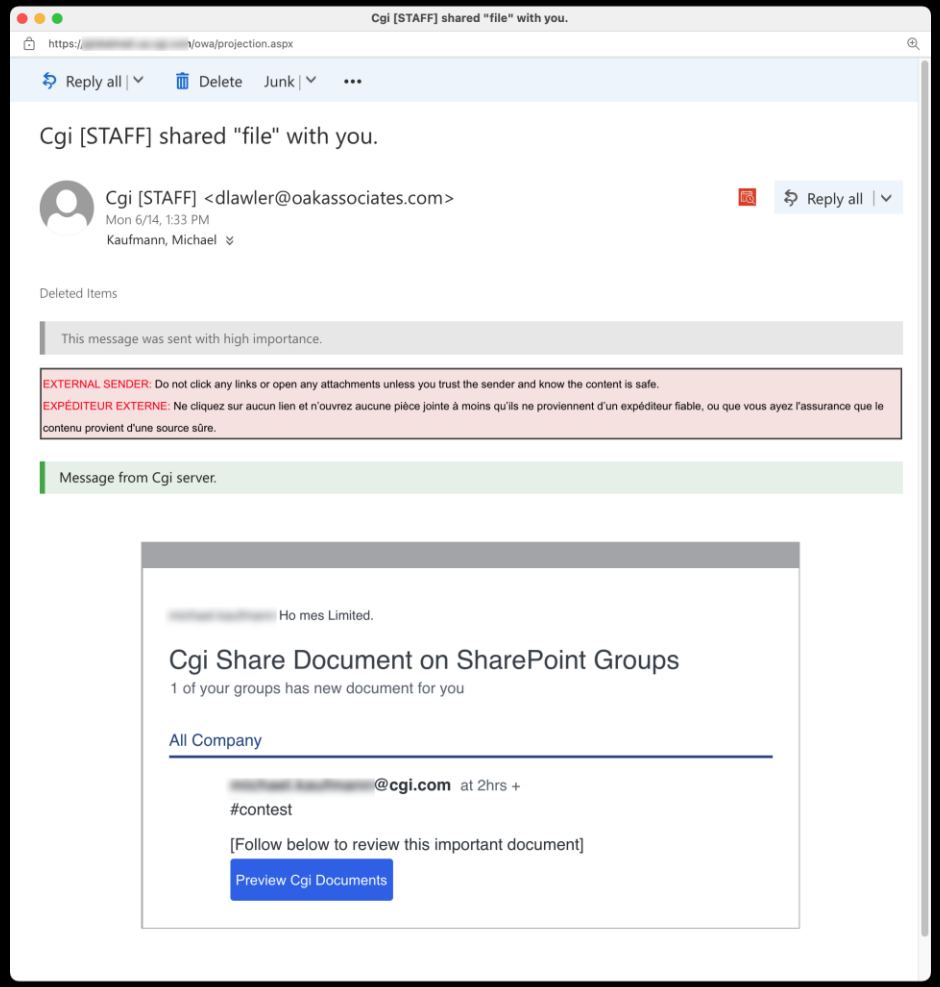
Ataques al sistema de ingeniería

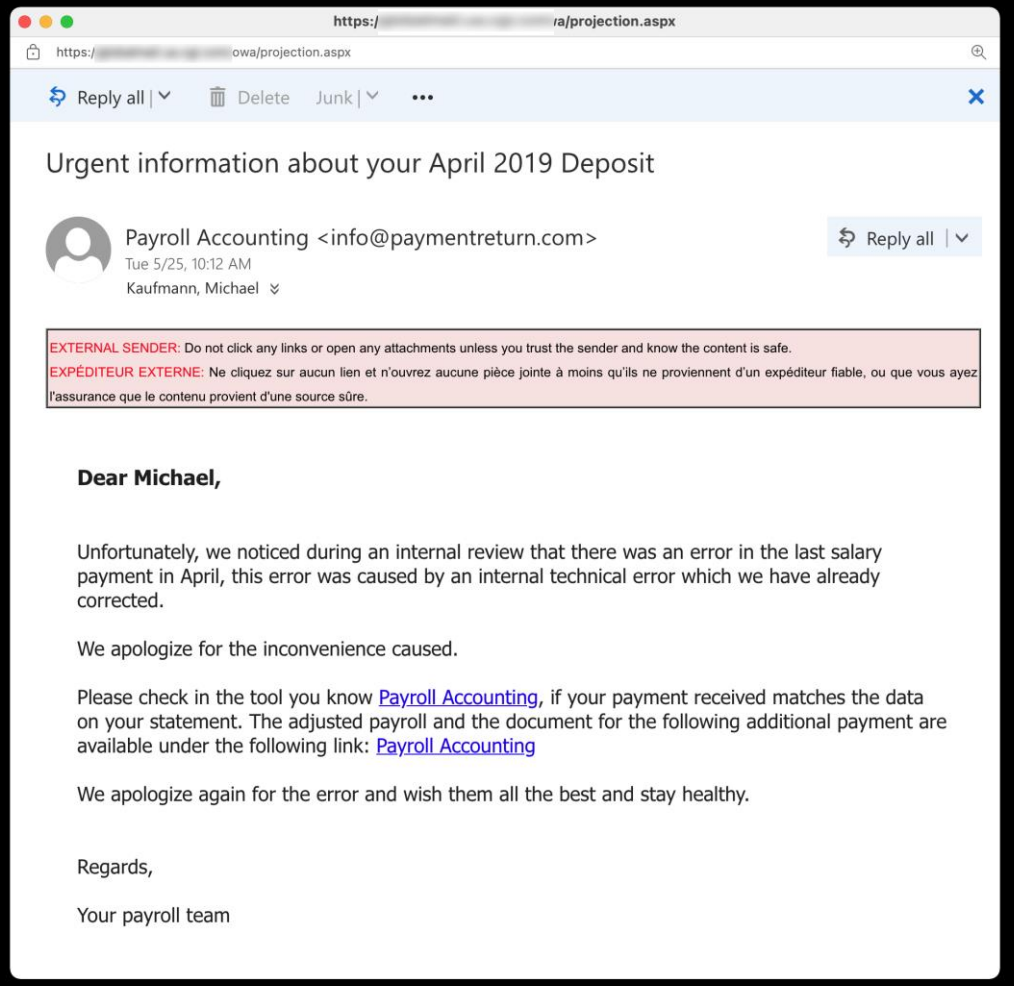
- Entornos de desarrollo
- Entornos de prueba / cuentas de prueba
- Ataques a las cadenas de suminitro de software
- Cambios de código (basta con una línea)
- Acceso a la pipeline (ejecución de código/scripts)

Intercambios de archivos no protegidos



Phishing

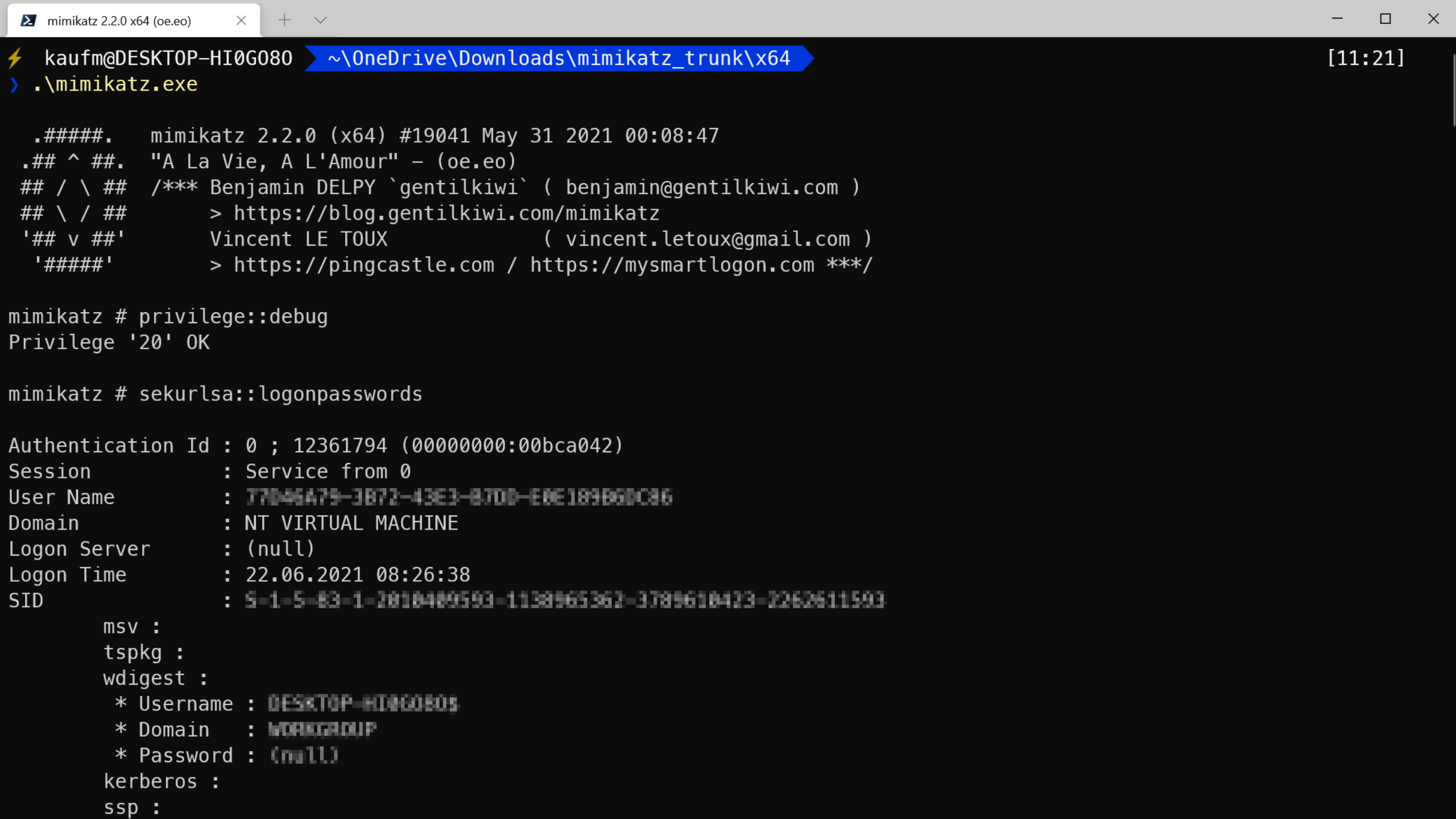




Datos de acceso en archivos de texto

TestCredentials.xml — Bearbeitet

```
<TestAccounts>
    <TestAccount name="domain\testUser1" password="testP@ssw0rd1!" />
    <TestAccount name="domain\testUser2" password="testP@ssw0rd2!" />
    <TestAccount name="domain\testUser3" password="testP@ssw0rd3!" />
</TestAccounts>
```


mimikatz 2.2.0 x64 (oe.eo)
kaufm@DESKTOP-HI0GO8O ~\OneDrive\Downloads\mimikatz_trunk\x64 [11:21]
.\mimikatz.exe
.#####. mimikatz 2.2.0 (x64) #19041 May 31 2021 00:08:47
.## ^ ##. "A La Vie, A L'Amour" - (oe.eo)
/ \ ## /*** Benjamin DELPY `gentilkiwi` (benjamin@gentilkiwi.com)
\ / ## > https://blog.gentilkiwi.com/mimikatz
'## v ##' Vincent LE TOUX (vincent.letoux@gmail.com)
'#####' > https://pingcastle.com / https://mysmartlogon.com ***/
mimikatz # privilege::debug
Privilege '20' OK
mimikatz # sekurlsa::logonpasswords
Authentication Id : 0 ; 12361794 (00000000:00bca042)
Session : Service from 0
User Name : 77046A79-3B72-43E3-87DD-E0E189B6DC86
Domain : NT VIRTUAL MACHINE
Logon Server : (null)
Logon Time : 22.06.2021 08:26:38
SID : S-1-5-83-1-2010409593-1138965362-3789618423-2262611593
msv :
tspkg :
wdigest :
* Username : DESKTOP-HI0GO8O$
* Domain : WORKGROUP
* Password : (null)
kerberos :
ssp :

mimikatz 2.2.0 x64 (oe.eo)

```
         * Username : kaufmann@outlook.de
         * Domain   : MicrosoftAccount
         * Password : (null)
        ssp :
        credman :
         [00000000]
         * Username : mk@writeabout.net
         * Domain   : outlook.office365.com
         * Password : [redacted]
         [00000001]
         * Username : (null)
         * Domain   : MicrosoftOffice16_Data:SSPI:mk@writeabout.net
         * Password : [redacted]
         [00000002]
         * Username : github.auth
         * Domain   : vscodevscode.github-authentication/github.auth
         * Password : [redacted]
64 4b 63 6c 43 46 37 4c 4f 72 4e 37 67 66 6a 44 55 36 4b 6a 51 36 65 6b 64 45 30 33 2f 0d 0a 35 61 78 57 42 4e 58 44 71
42 43 4a 52 44 74 42 39 47 63 34 45 4a 59 5a 69 43 36 4b 4a 2f 53 49 68 39 6e 55 67 6f 4e 6e 53 57 67 71 4a 47 6d 33 58
4e 30 34 34 75 4c 6e 4c 52 42 2f 50 32 43 51 0d 0a 45 41 45 41 2f 34 5a 4e 66 2b 47 54 6b 53 56 4d 65 75 52 79 49 74 4e
56 76 5a 5a 6f 35 42 68 4e 75 75 58 61 30 2f 4f 6a 50 5a 38 62 7a 7a 53 71 50 74 61 62 43 78 74 31 6b 6f 36 35 59 74 6d
[redacted]
64 4b 63 6c 43 46 37 4c 4f 72 4e 37 67 66 6a 44 55 36 4b 6a 51 36 65 6b 64 45 30 33 2f 0d 0a 35 61 78 57 42 4e 58 44 71
42 43 4a 52 44 74 42 39 47 63 34 45 4a 59 5a 69 43 36 4b 4a 2f 53 49 68 39 6e 55 67 6f 4e 6e 53 57 67 71 4a 47 6d 33 58
4e 30 34 34 75 4c 6e 4c 52 42 2f 50 32 43 51 0d 0a 45 41 45 41 2f 34 5a 4e 66 2b 47 54 6b 53 56 4d 65 75 52 79 49 74 4e
56 76 5a 5a 6f 35 42 68 4e 75 75 58 61 30 2f 4f 6a 50 5a 38 62 7a 7a 53 71 50 74 61 62 43 78 74 31 6b 6f 36 35 59 74 6d
72 0d 0a 53 30 2b 72 39 6c 69 47 46 34 42 55 32 48 6a 70 4f 5a 73 43 47 6a 6c 46 63 37 6d 72 46 6f 4c 70 64 30 6a 37 59
64 43 2b 47 48 5a 4a 37 68 79 2b 70 48 6f 4d 48 7a 32 51 76 6d 2f 78 5a 6d 30 75 0d 0a 38 55 49 6c 47 30 52 2b 71 37 47
4d 55 74 44 6f 72 43 7a 69 4a 78 4a 47 31 59 67 49 51 34 51 45 79 61 7a 55 4e 49 67 4f 70 63 68 74 47 4f 57 44 41 6e 35
5a 53 0d 0a 41 4c 51 37 38 6b 37 53 6a 0d 0a 64 5a 43 34 74 6c 65 6e 73 31 77 68 65 35 59 59 60 2f 66 33 56 32 70 39 5a
```

mimikatz 2.2.0 x64 (oe.eo)

```
[redacted]
         [00000003]
         * Username : wulfland
         * Domain   : GitHub - https://api.github.com/wulfland
         * Password : [redacted]
         [00000004]
         * Username : [redacted]
         * Domain   : https://gitlab.com
         * Password : [redacted]
        cloudap :
             Cachedir : [redacted]
             Key GUID : [redacted]
             PRT      : □
             DPAPI Key: [redacted]

Authentication Id : 0 ; 195058 (00000000:0002f9f2)
Session           : Interactive from 1
User Name         : DWM-1
Domain            : Window Manager
Logon Server      : (null)
Logon Time        : 21.06.2021 22:24:46
SID               : [redacted]
        msv :
        tspkg :
        wdigest :
         * Username : [redacted]
         * Domain   : [redacted]
         * Password : [redacted]
```

Ejemplo


Intercambio de archivo abierto
Datos de acceso cuenta de prueba en texto sin codificar
Máquina Dev con cuenta de prueba como administrador local
Mimikatz
Contraseña desarrollador

Datos de acceso del desarrollador

- E-Mail → Phishing de compañeros
- Acceso a máquinas → Logon, Mimikatz
- Código fuente → Insertar/modificar código
- Acceso Pipeline → Ejecutar scripts
- Acceso entorno de prueba → Prueba Prod?
- Acceso Prod → Game Over

```
{
  "/api/*": {
    "changeOrigin": true,
    "target": "https://api.project-demo.de",
    "auth": "[redacted]"
  }
}
```

```
const id_token_string = '[redacted]';

app.use(
  '/api/test',
  createProxyMiddleware({
    target: 'https://api.test.[redacted]com/',
    changeOrigin: true,
    pathRewrite: {
      '^/api/test': '/test',
    },
    headers: {
      'Content-Type': 'application/json',
      'Credentials': true,
      'Cookie': `id_token=${id_token_string}`
    },
  })
);
```

Schema: https://json.schemastore.org/appsettings.json

```
{
  "AppSettings": {
    "ScheduleExecutionIntervalInMinutes": 60
  },
  "MailSettings": {
    "Url": "Outlook.office365.com",
    "User": "[redacted]",
    "Password": "[redacted]",
    "ProcessedFolderName": "[redacted]",
    "ErrorFolderName" : "[redacted]",
    "OutOfToleranceFolderName": "[redacted]"
  }
}
```

27 lines (27 sloc) | 1.45 KB

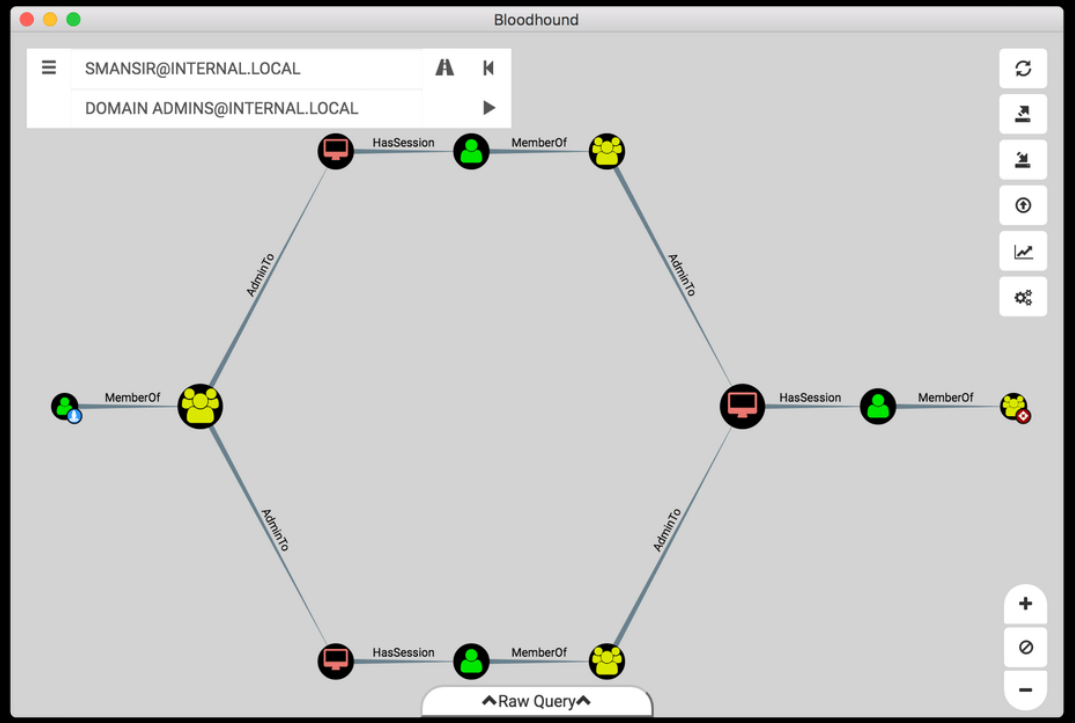
```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
    <configSections>
        <sectionGroup name="applicationSettings" type="System.Configuration.ApplicationSettingsG
            <section name="SPOEmulators.Tests.Properties.Settings" type="System.Configuration.Cl
        </sectionGroup>
    </configSections>
    <applicationSettings>
        <SPOEmulators.Tests.Properties.Settings>
            <setting name="OnPremUrl" serializeAs="String">
                <!-- Enter thr url to a on prem site for integration testing. -->
                <value>https://localhost/sites/dev</value>
            </setting>
            <!-- Enter thr url to a O365 site for integration testing. -->
            <setting name="O365Url" serializeAs="String">
                <value>http://xxxx.sharepoint.com</value>
            </setting>
            <!-- Enter credentials to connect to the site (O365 or on prem if neccessary) -->
            <setting name="O365User" serializeAs="String">
                <value>user@tenant.onmicrosoft.com</value>
            </setting>
            <setting name="O365Password" serializeAs="String">
                <value>****</value>
            </setting>
        </SPOEmulators.Tests.Properties.Settings>
    </applicationSettings>
</configuration>
```

De Dev a Prod

Bloodhound: https://github.com/adaptivethreat/Bloodhound

grpmgr test01 Administrators /enum

Otras posibilidades de conseguir datos de acceso:
- Phishing
- Responder (https://github.com/lgandx/Responder)
- Pineapple
- …
- Contraseñas no seguras



TestAccounts.txt
dom\test01

→ mkadev01
dom\mka
dom\test01

→ testsrv01
dom\mka
dom\admlisa

→ prod01
dom\admlisa

Ataques a los sistemas de producción

- SQL-Injection
- XSS (Cross-Site-Scripting)

SQL Injection

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

105; DROP TABLE Suppliers

```
11  module.exports = function searchProducts () {
12    return (req, res, next) => {
13      let criteria = req.query.q === 'undefined' ? '' : req.query.q || ''
14      criteria = (criteria.length <= 200) ? criteria : criteria.substring(0, 200)
15      models.sequelize.query(`SELECT * FROM Products WHERE ((name LIKE '%${criteria}%' OR description LIKE '%${criteria}%') AND deletedAt IS NULL) ORDER BY name`)
16        .then(([products]) => {
17          const dataString = JSON.stringify(products)
18          if (utils.notSolved(challenges.unionSqlInjectionChallenge)) { // vuln-code-snippet hide-start
19            let solved = true
20            models.User.findAll().then(data => {
21              const users = utils.queryResultToJson(data)
22              if (users.data?.length) {
23                for (let i = 0; i < users.data.length; i++) {
24                  solved = solved && utils.containsOrEscaped(dataString, users.data[i].email) && utils.contains(dataString, users.data[i].password)
25                  if (!solved) {
26                    break
27                  }
28                }
29                if (solved) {
30                  utils.solve(challenges.unionSqlInjectionChallenge)
31                }
```

XSS (Cross-Site-Scripting)

```
143   // vuln-code-snippet start localXssChallenge xssBonusChallenge
144   filterTable () {
145     let queryParam: string = this.route.snapshot.queryParams.q
146     if (queryParam) {
147       queryParam = queryParam.trim()
148       this.ngZone.runOutsideAngular(() => { // vuln-code-snippet hide-start
149         this.io.socket().emit('verifyLocalXssChallenge', queryParam)
150       }) // vuln-code-snippet hide-end
151       this.dataSource.filter = queryParam.toLowerCase()
152       this.searchValue = this.sanitizer.bypassSecurityTrustHtml(queryParam)
153       this.gridDataSource.subscribe((result: any) => {
154         if (result.length === 0) {
155           this.emptyState = true
156         } else {
157           this.emptyState = false
158         }
159       })
160     } else {
161       this.dataSource.filter = ''
162       this.searchValue = undefined
163       this.emptyState = false
164     }
165   }
```

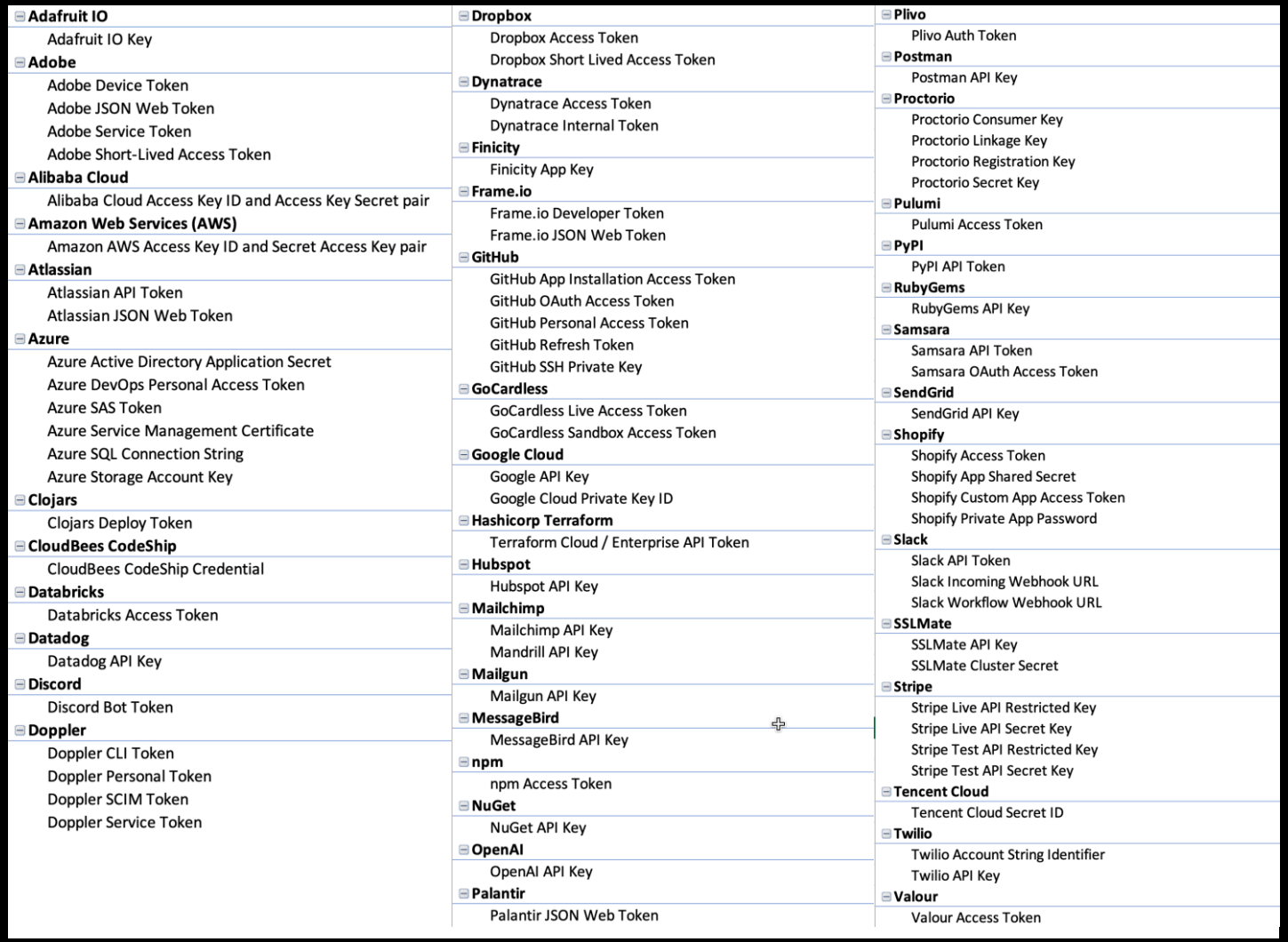
```
var Affix = function (element, options) {
  this.options = $.extend({}, Affix.DEFAULTS, options)

  this.$target = $(this.options.target)
    .on('scroll.bs.affix.data-api', $.proxy(this.checkPosition, this))
    .on('click.bs.affix.data-api',  $.proxy(this.checkPositionWithEventLoop, this))

  this.$element     = $(element)
  this.affixed      = null
  this.unpin        = null
  this.pinnedOffset = null

  this.checkPosition()
}
```

¿Qué se puede hacer (Equipo Azul)?

- MFA para todos los sistemas
- Retirar todos los roles de administrador local
- Estricta separación Test/Prod
- Key-Vault para todos los Secrets
  - Separación tiempo de ejecución/despliegue
  - Test/Prod
  - Rotación automática
- Suscripciones separadas
- Codespaces / VDI para Development
- SIEM (Azure Security Center, Splunk)
- Credential Scanning (Code/Fileshares)

How to shift left security?

Secret Scanning



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

Code

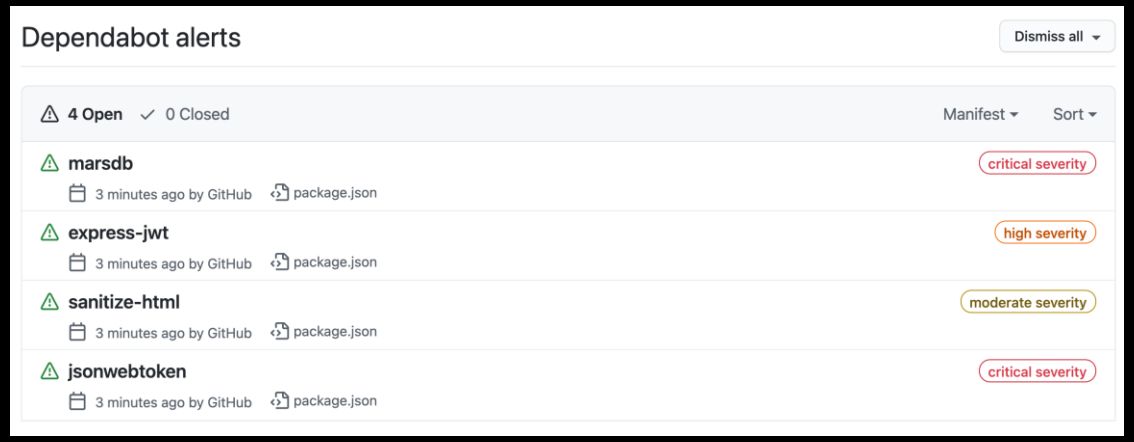
- **GitHub (private!)**
- gitLeaks.
- SpectralOps
- Git-Secrets
- Whispers
- Gittyleaks
- Git-all-secrets
- **…**

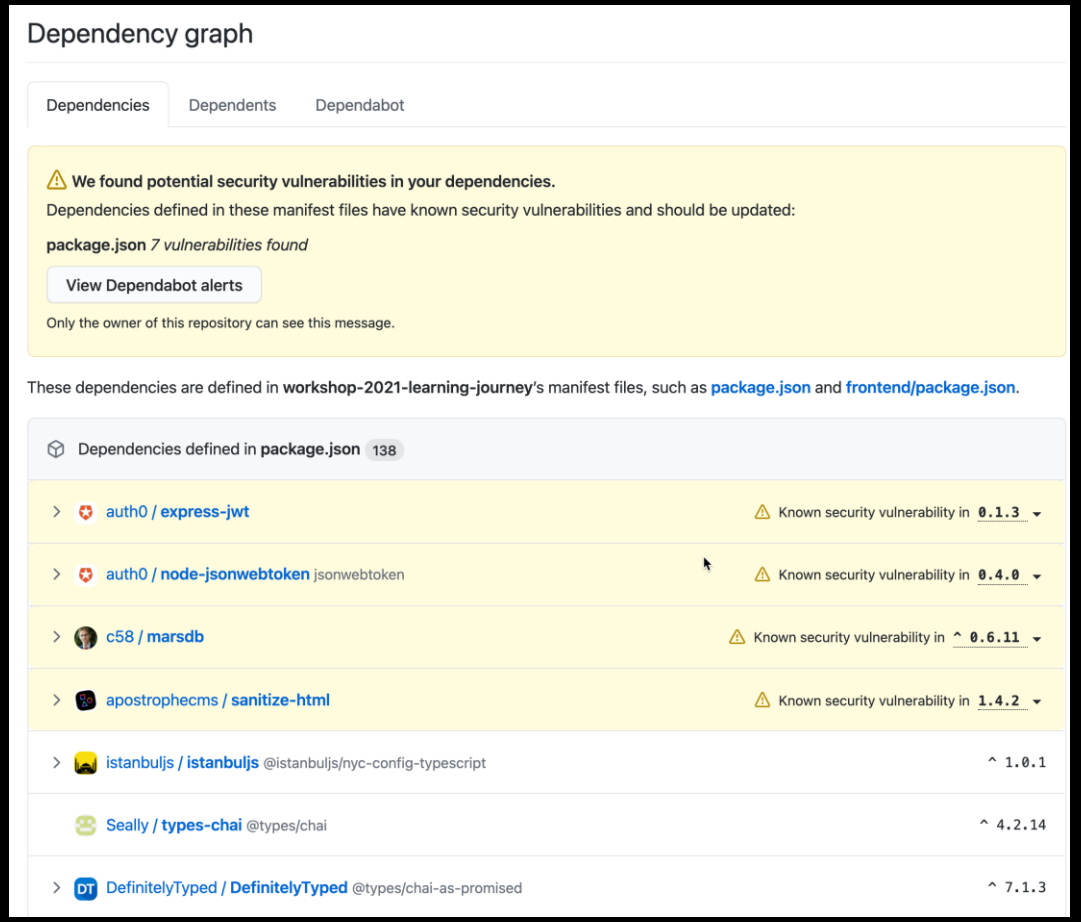
Fileshare

- Bash/PowerShell
- Dumpster

Software Composition Analysis (SCA)

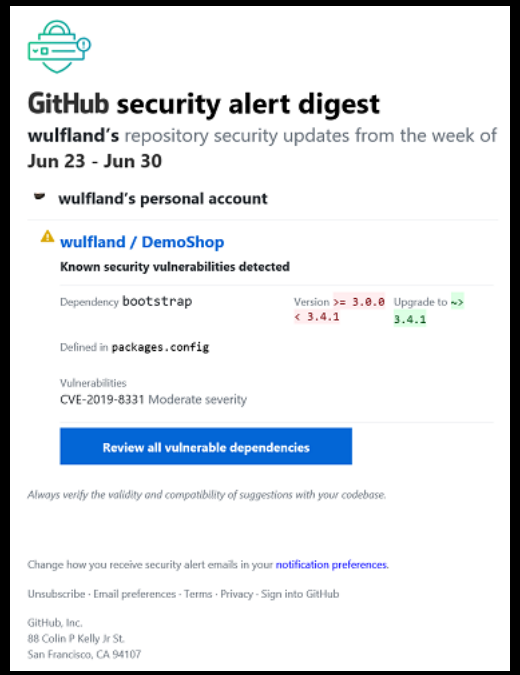
GitHub (Dependency-Graph/Dependabot)
anchore (https://anchore.com/) o
Dependency-Track (https://dependencytrack.org/)

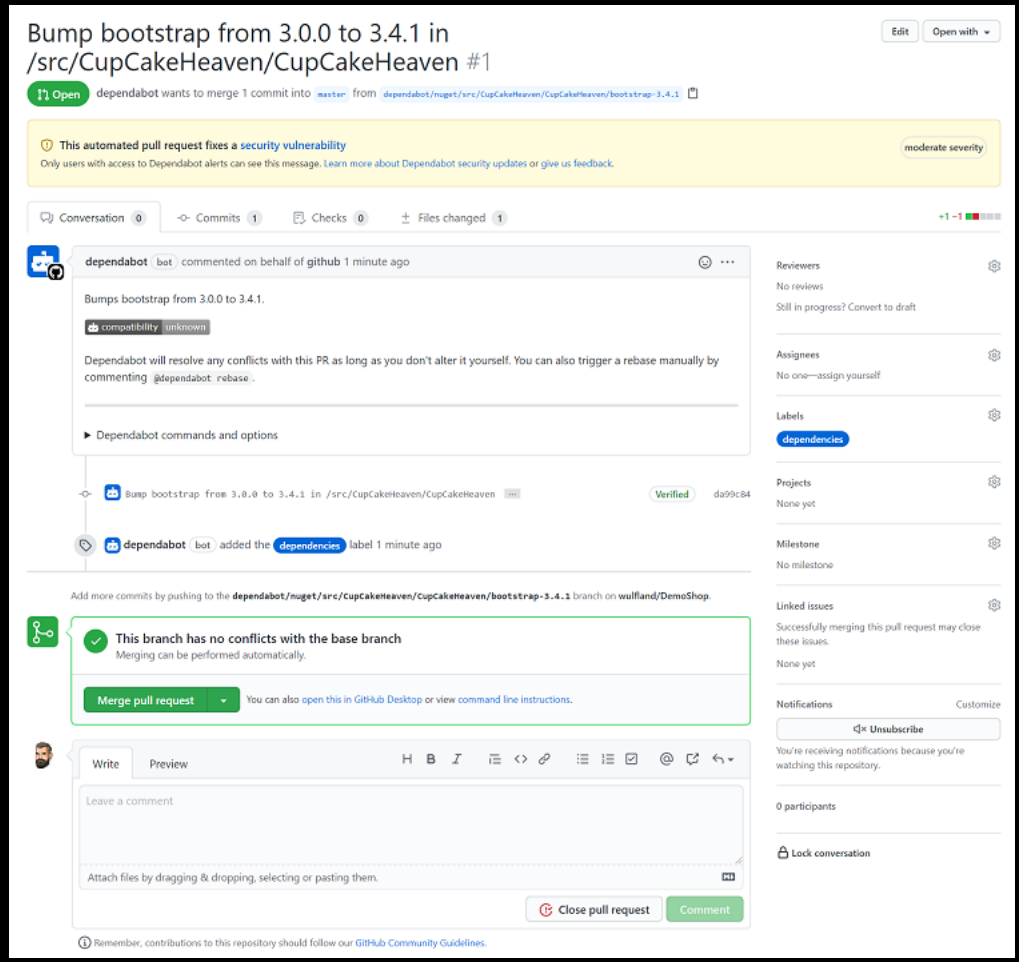




Software Composition Analysis (SCA)

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

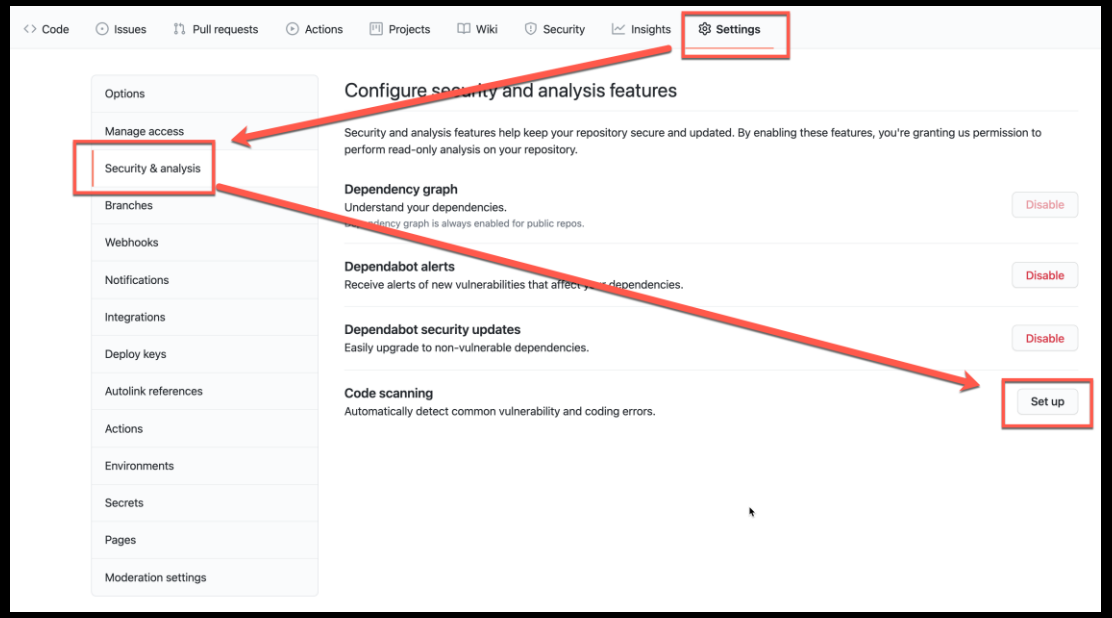


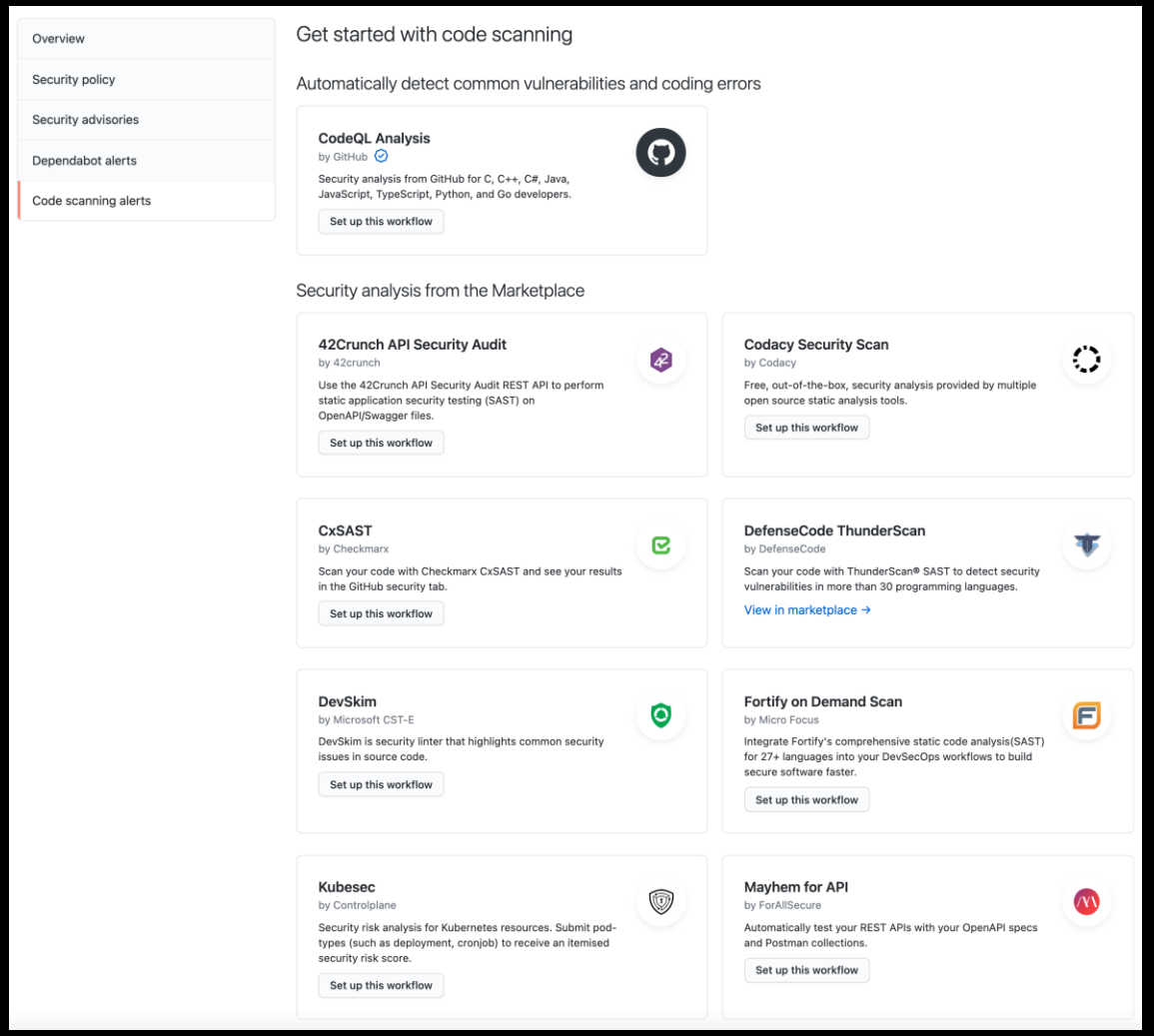


Static Application Security Testing (SAST)

Whitebox-Testing

- GitHub CodeQL
- SonarQube
- Semgrep (https://semgrep.dev/)
- Mobile-Security-Framework (MobSF)
  (https://github.com/MobSF/Mobile-Security-Framework-MobSF)

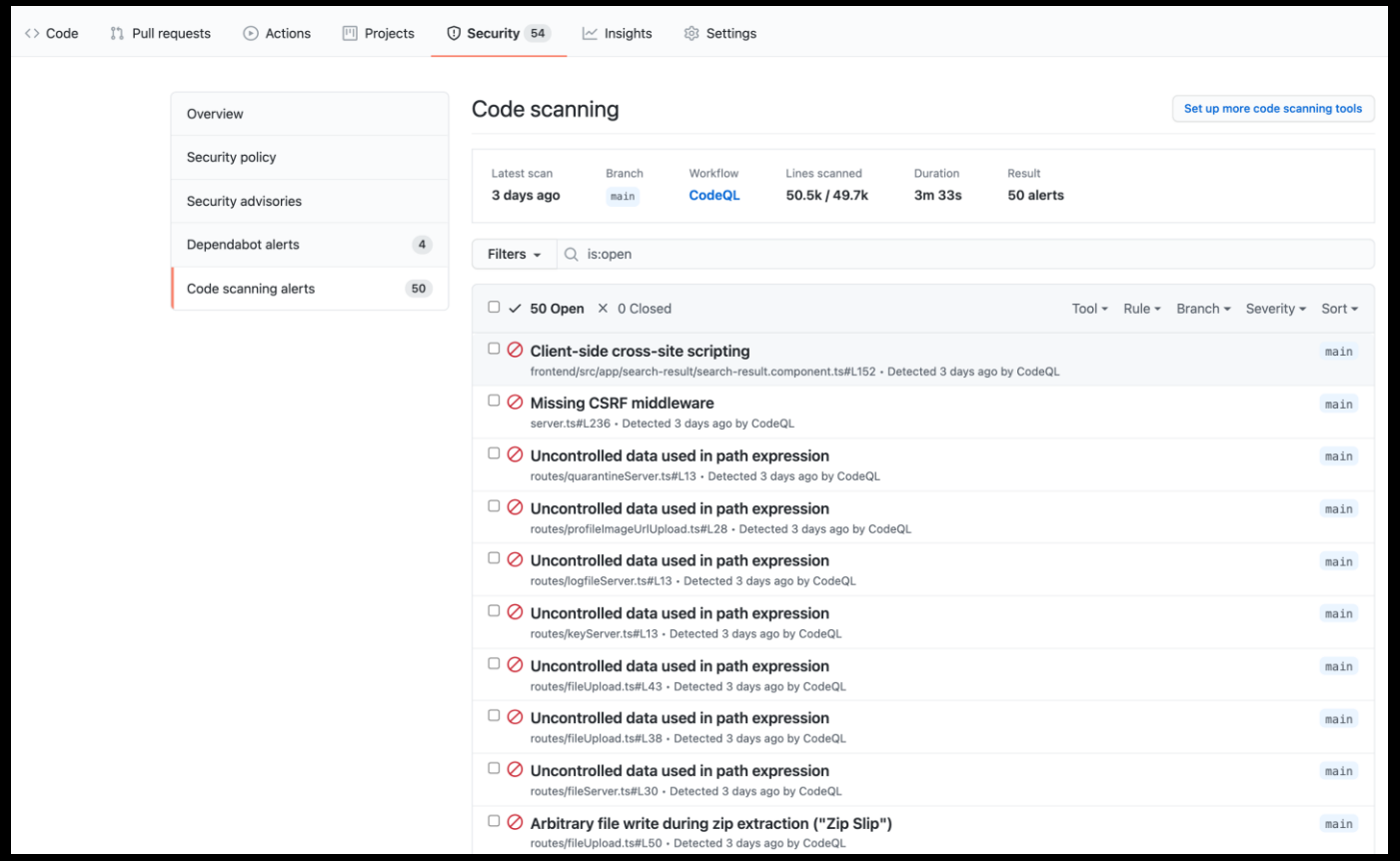




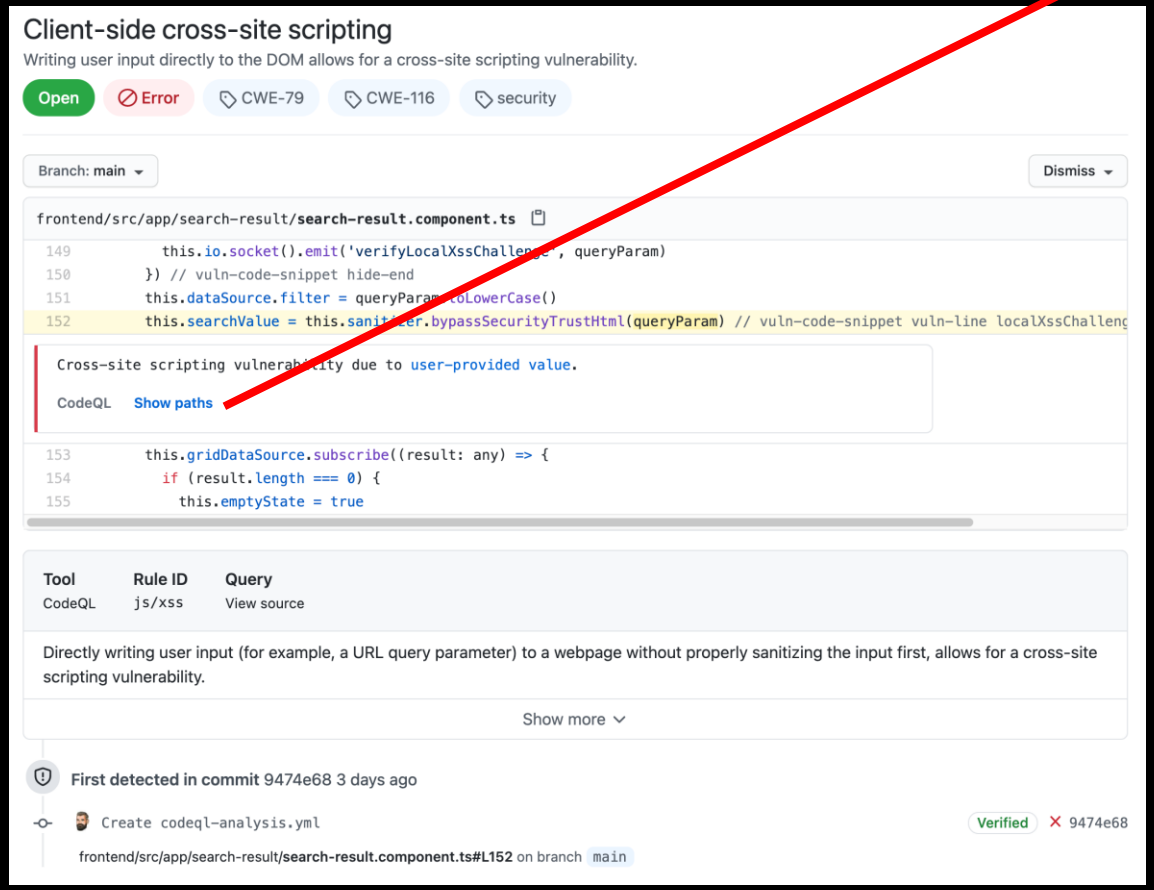
Code Scanning (CodeQL)

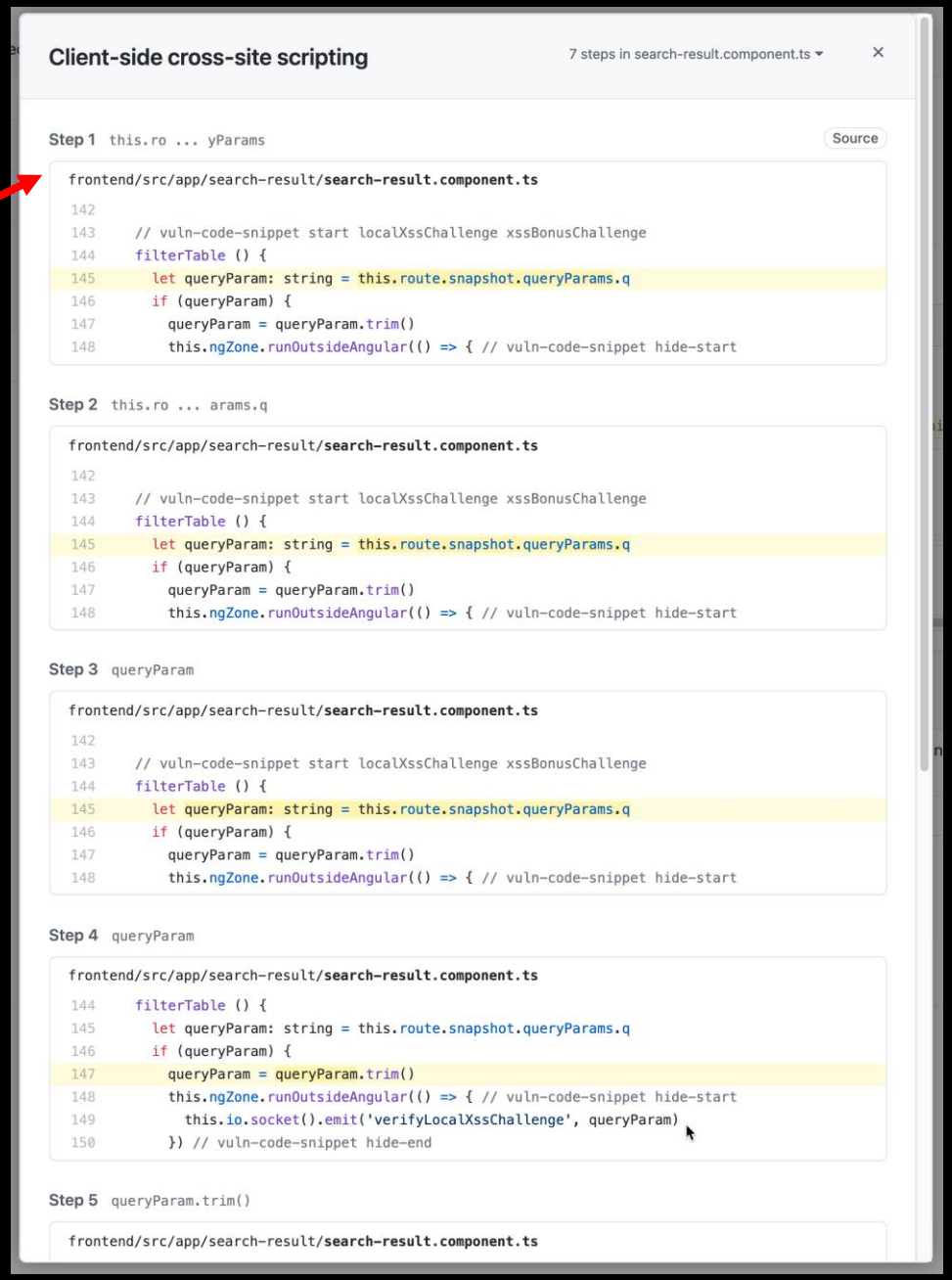
Fork: https://github.com/advanced-security/workshop-2021-learning-journey

```yaml
name: "CodeQL"

on:
  push:
    branches: [ master ]
  pull_request:
    # The branches below must be a subset of the branches above
    branches: [ master ]

jobs:
  analyze:
    name: Analyze
    runs-on: ubuntu-latest
    permissions:
      actions: read
      contents: read
      security-events: write

    strategy:
      fail-fast: false
      matrix:
        language: [ 'javascript', 'java' ]
        # CodeQL supports [ 'cpp', 'csharp', 'go', 'java', 'javascript', 'python' ]

    steps:
    - name: Checkout repository
      uses: actions/checkout@v2

    # Initializes the CodeQL tools for scanning.
    - name: Initialize CodeQL
      uses: github/codeql-action/init@v1
      with:
        languages: ${{ matrix.language }}
        # If you wish to specify custom queries, you can do so here or in a config file.
        # By default, queries listed here will override any specified in a config file.
        # Prefix the list here with "+" to use these queries and those in the config file.
        # queries: ./path/to/local/query, your-org/your-repo/queries@main

    # Autobuild attempts to build any compiled languages  (C/C++, C#, or Java).
    # If this step fails, then you should remove it and run the build manually (see below)
    - name: Autobuild
      uses: github/codeql-action/autobuild@v1
```

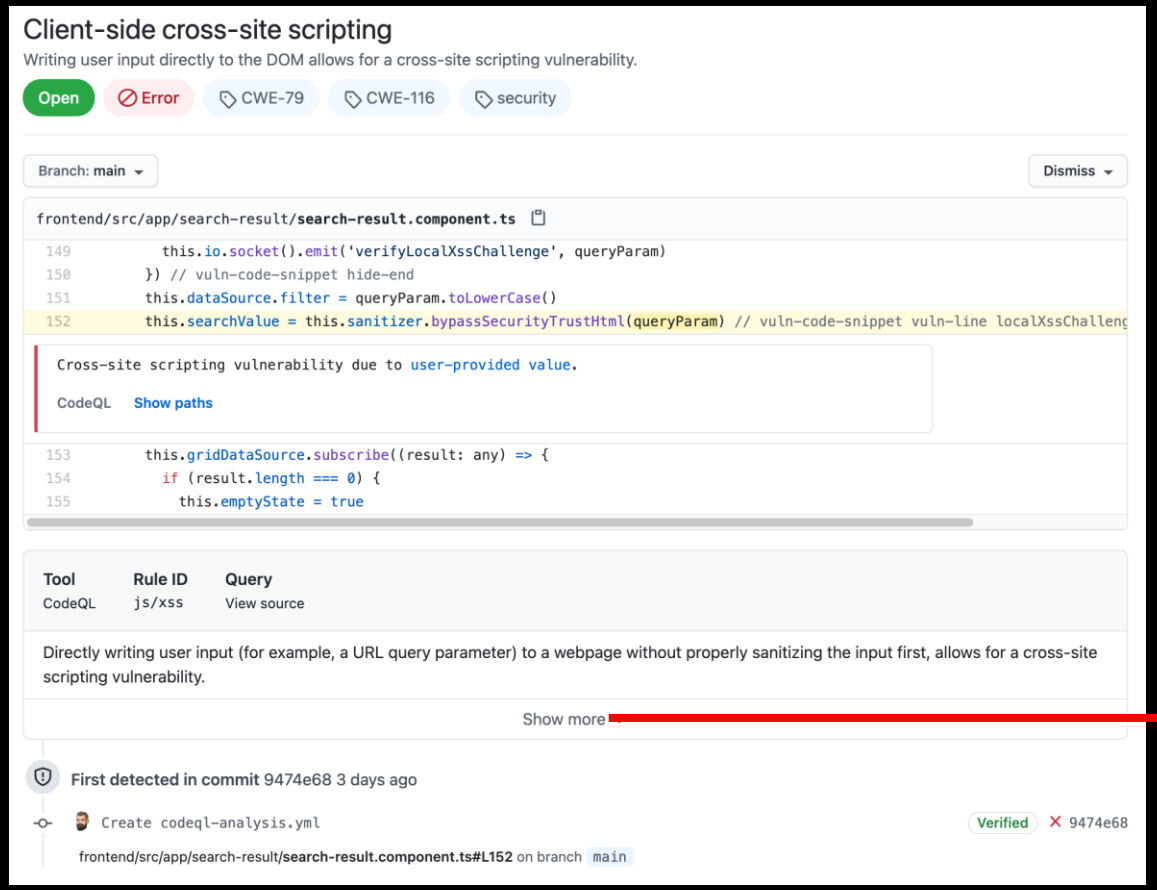


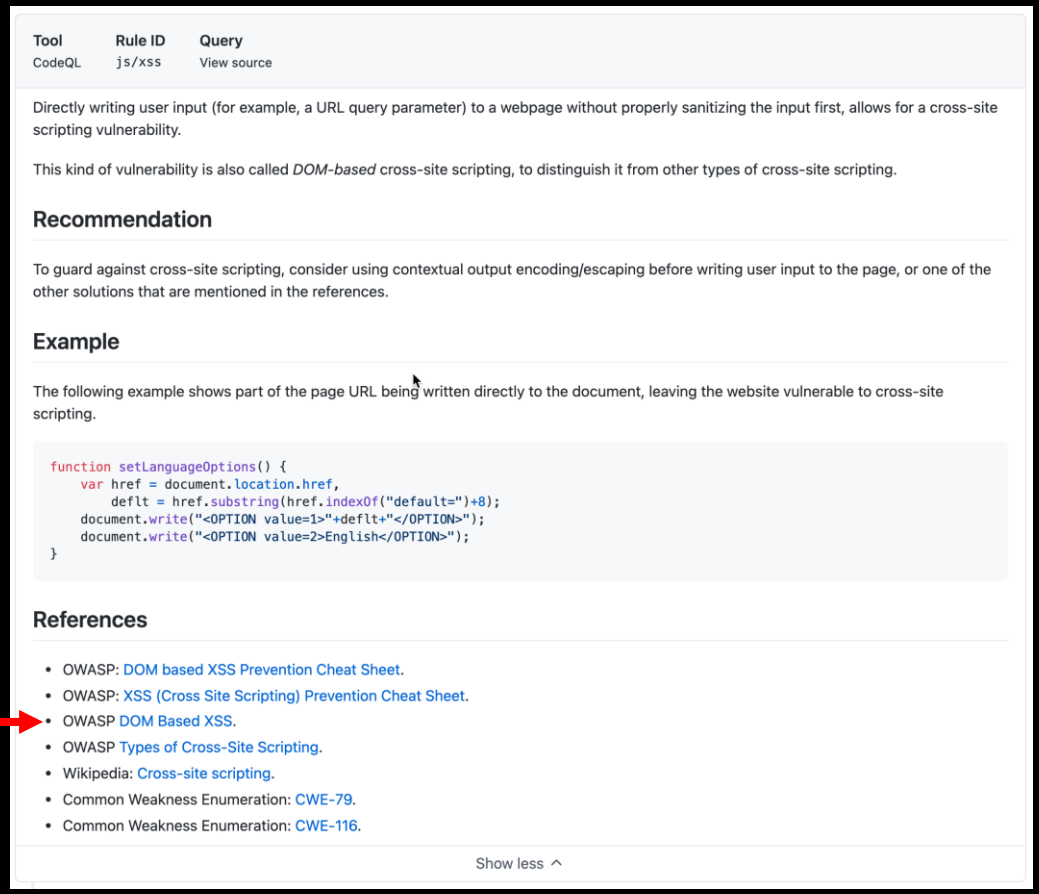
Code Scanning (CodeQL)



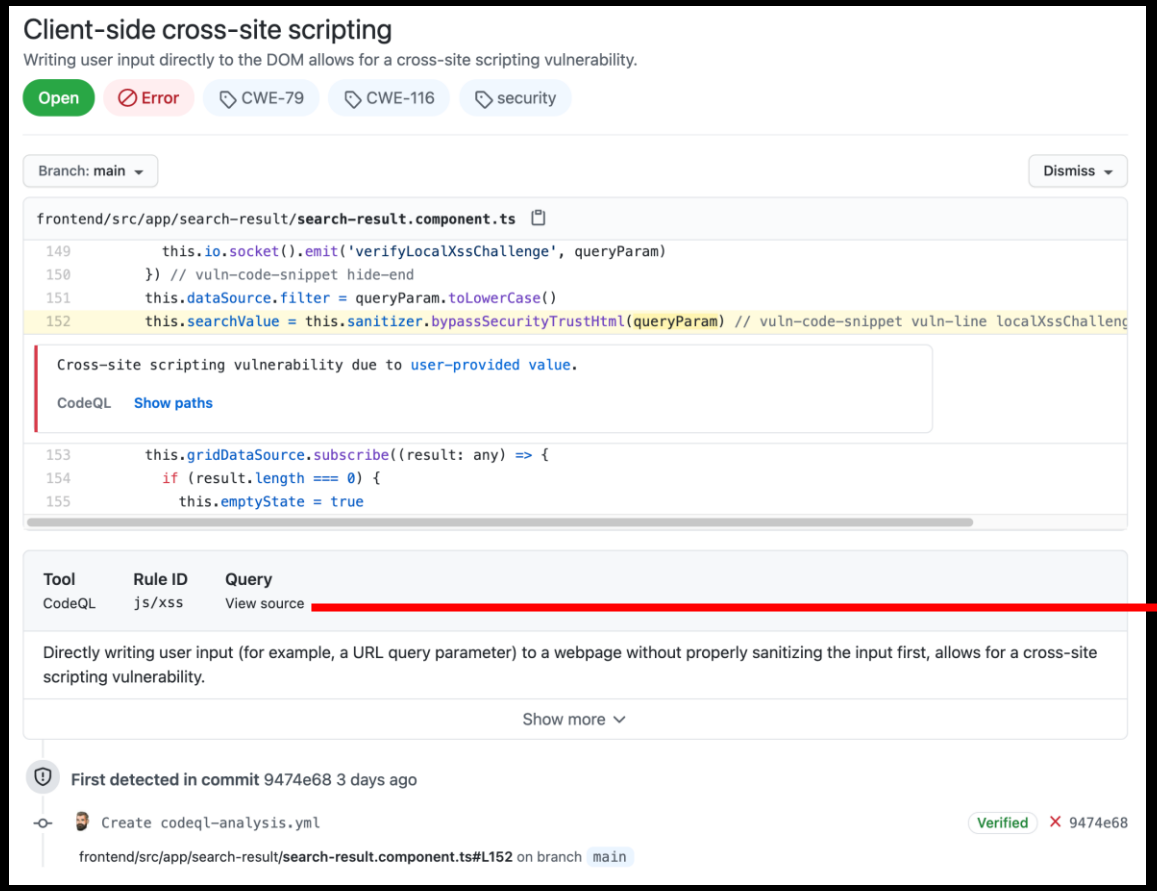


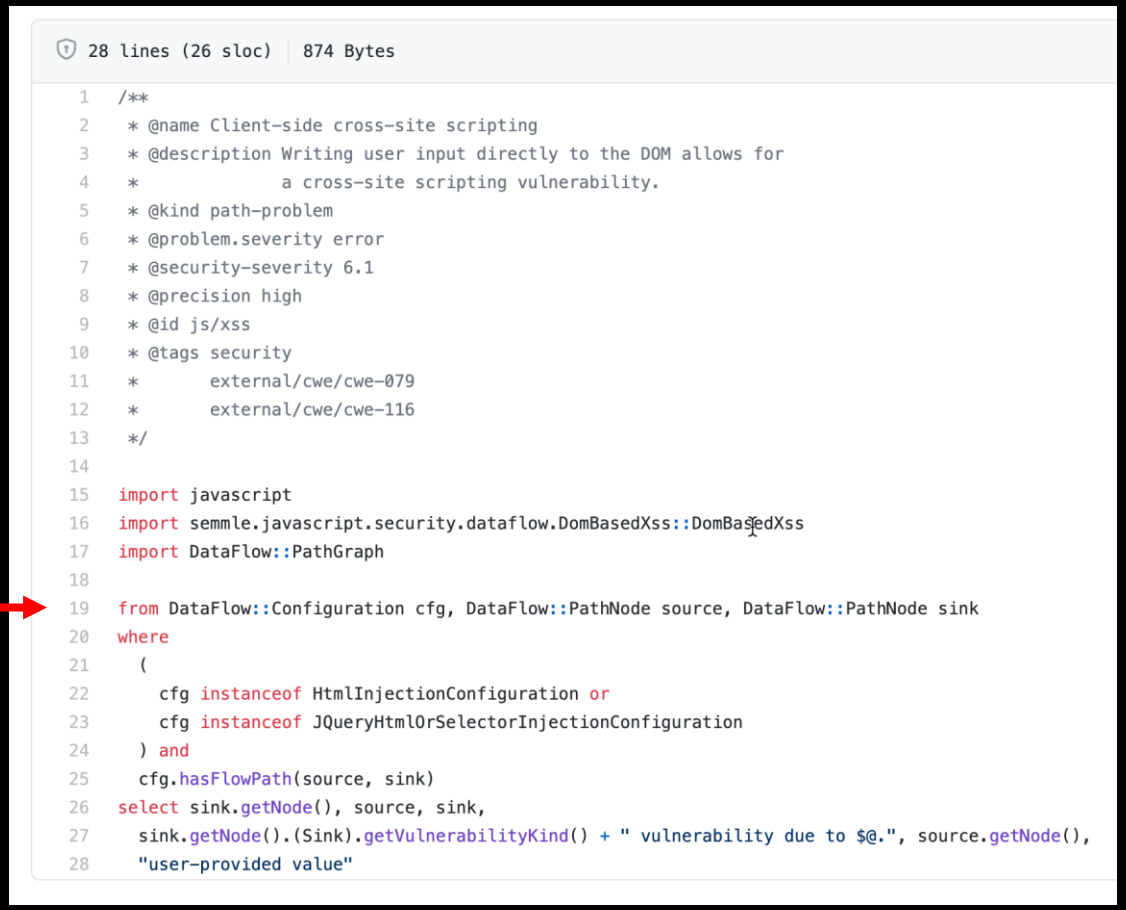
Code Scanning (CodeQL)



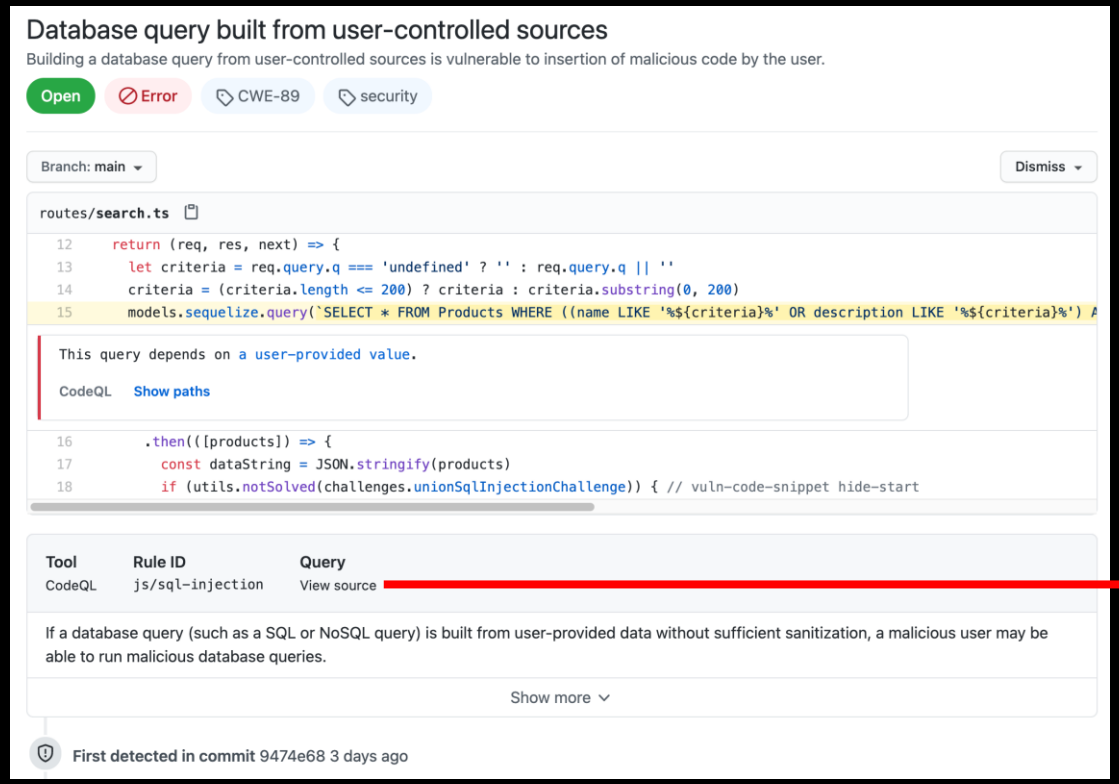


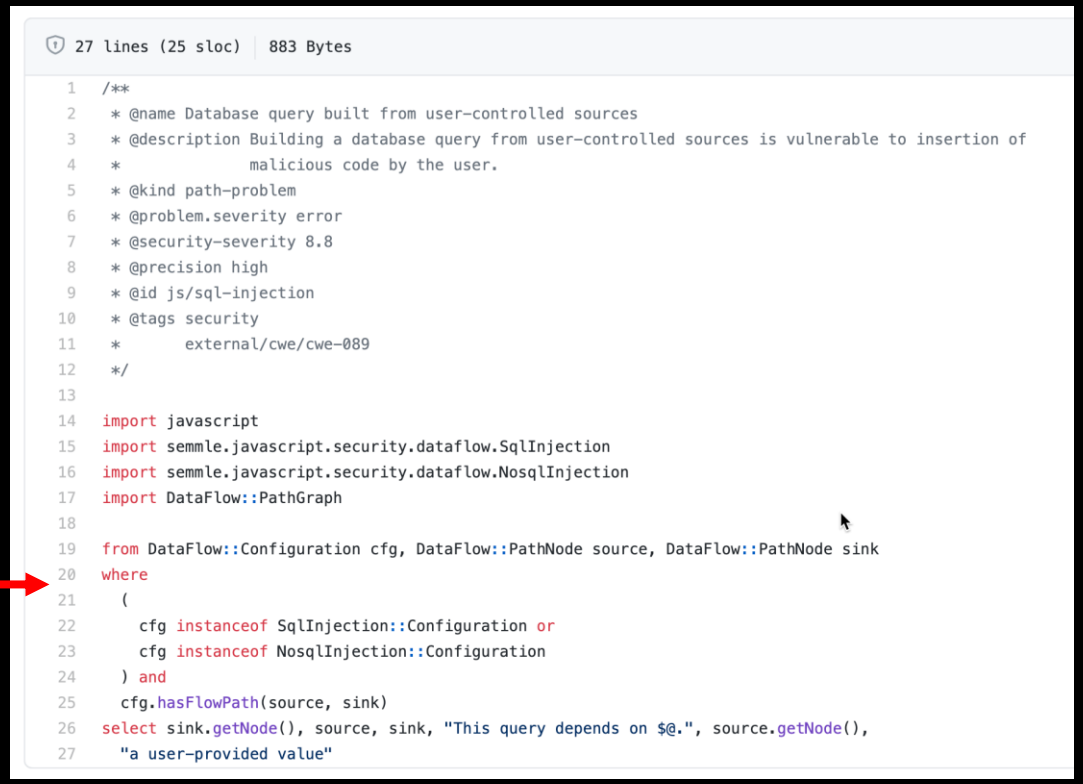
Code Scanning (CodeQL)



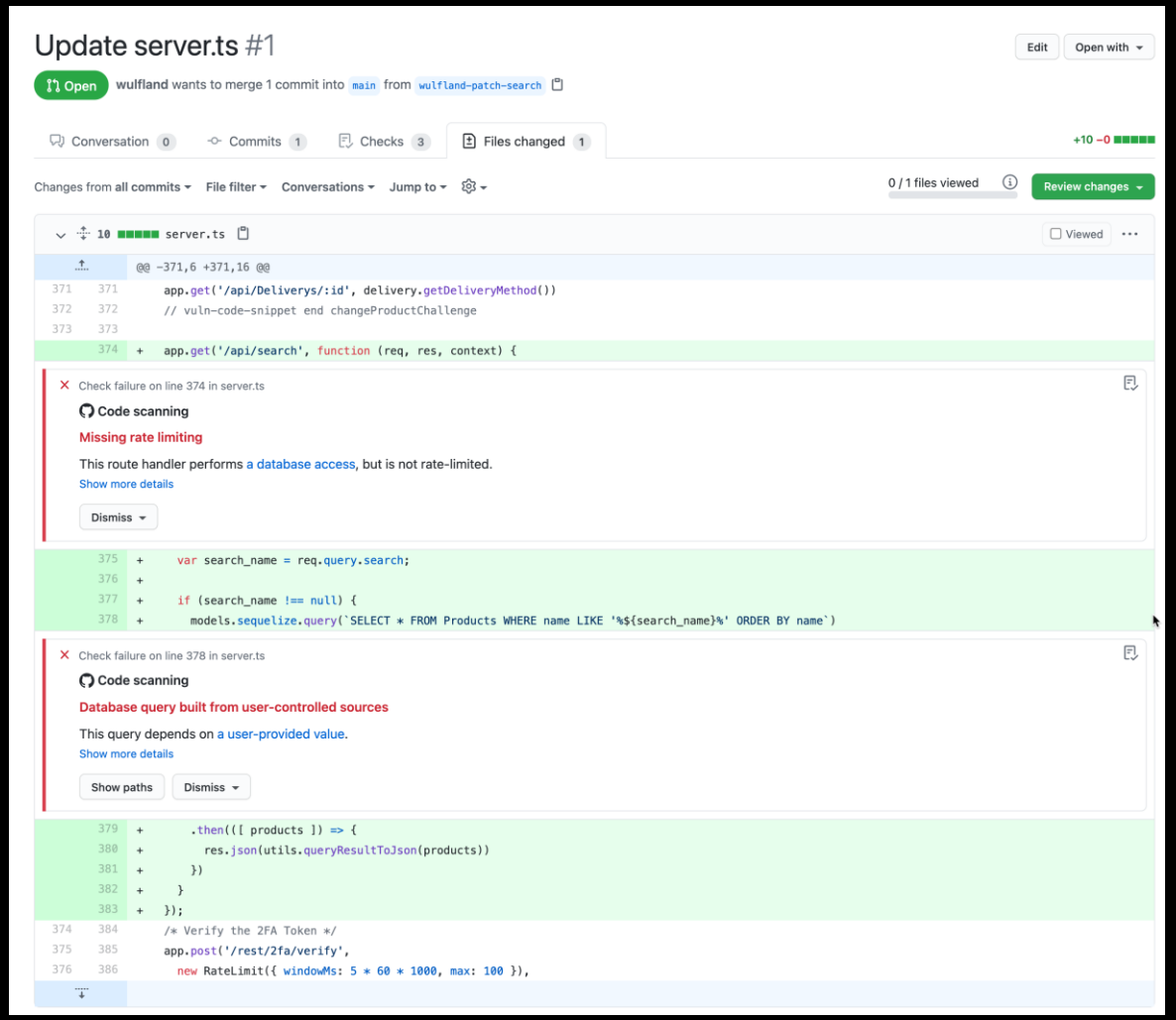


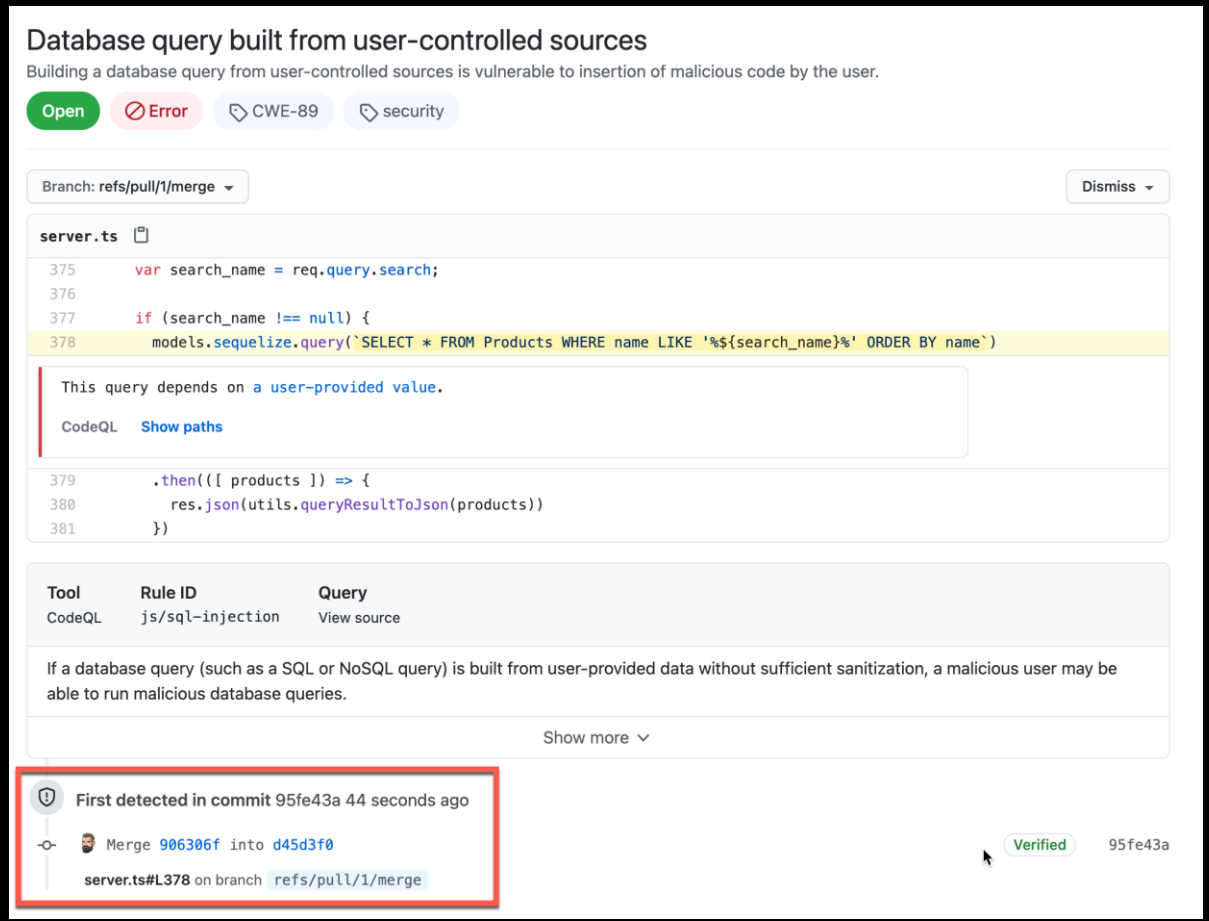
Code Scanning (CodeQL)





CodeQL: Pull-Request Integration





CodeQL: Config

```yaml
# Initializes the CodeQL tools for scanning.
- name: Initialize CodeQL
  uses: github/codeql-action/init@v1
  with:
    languages: ${{ matrix.language }}
    # If you wish to specify custom queries, you can do so here or in a config file.
    # By default, queries listed here will override any specified in a config file.
    # Prefix the list here with "+" to use these queries and those in the config file.
    # queries: ./path/to/local/query, your-org/your-repo/queries@main
    queries: security-extended
```

CodeQL: Config

```yaml
# Initializes the CodeQL tools for scanning.
- name: Initialize CodeQL
  uses: github/codeql-action/init@v1
  with:
    languages: ${{ matrix.language }}
    # If you wish to specify custom queries, you can do so here or in a config file.
    # By default, queries listed here will override any specified in a config file.
    # Prefix the list here with "+" to use these queries and those in the config file.
    # queries: ./path/to/local/query, your-org/your-repo/queries@main
    # queries: security-extended
    config-file: ./.github/codeql-config.yml
```

codeql-config.yml

Raw

```yaml
name: "Custom CodeQL Configuration"

# Disabling the default queries is critical part of this.
# Nothing will be disabled if the default queries still run.
# - https://docs.github.com/en/code-security/secure-coding/configuring-code-scanning#disabling-the-default-queries
disable-default-queries: true

queries:
  # Point to the custom query suite that needs to be used
  - uses: ./.github/codeql/custom-javascript.qls

# Ignore some of the paths as these are some open source componants.
# This only works for Interpreted Languages (JavaScript / TypeScript / Python)
# - https://docs.github.com/en/code-security/secure-coding/automatically-scanning-your-code-for-vulnerabilities-and-errors/conf
paths-ignore:
  - '**/node_modules'
  # Ignoring our test folders
  - '**/test'
```

CodeQL: Config

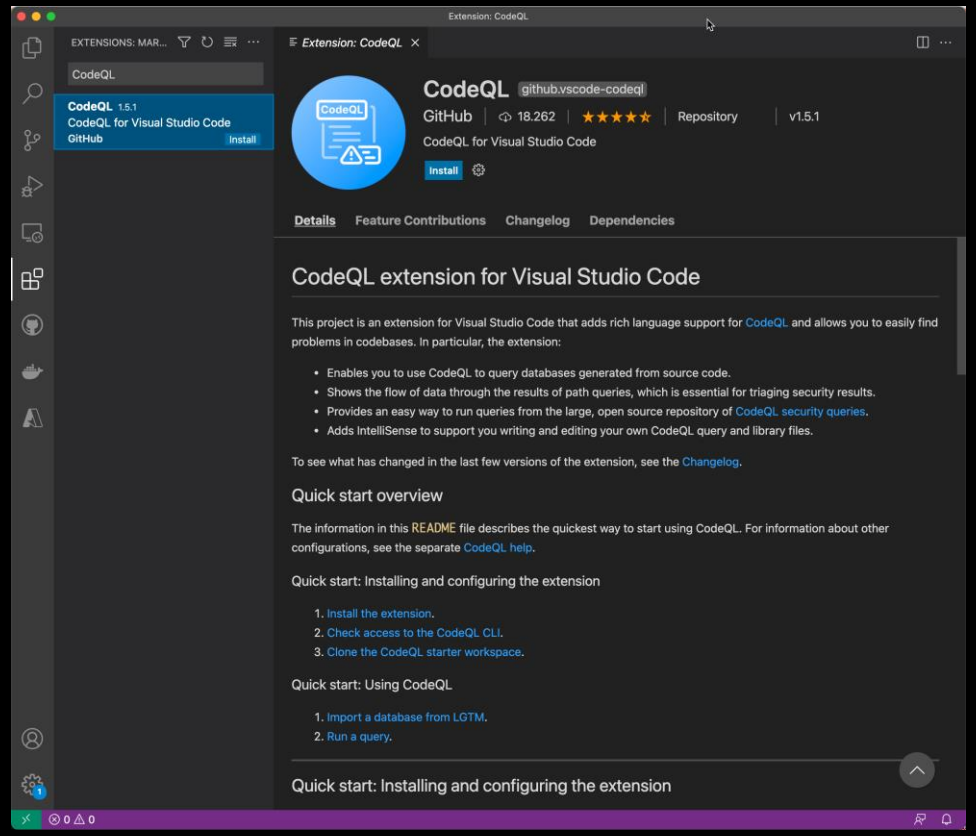
custom-javascript.qls

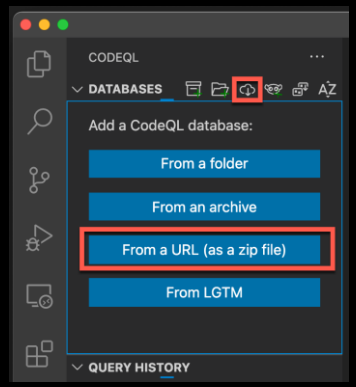
Raw

```
# Documentation
# - https://codeql.github.com/docs/codeql-cli/creating-codeql-query-suites/
# - https://github.com/github/codeql/actions/workflows/query-list.yml
# - https://github.com/github/codeql/blob/main/misc/suite-helpers/security-extended-selectors.yml
# - https://codeql.github.com/docs/codeql-cli/creating-codeql-query-suites/#examples

- description: "Custom JavaScript Suite"

# Import/Extend from an existing CodeQL suite
- import: codeql-suites/javascript-security-extended.qls
  from: codeql-javascript


# Exclude queries that are not relevant for my organisation or have a
# higher False Positive rate than we are confident with.
- exclude:
    id:
      # https://github.com/github/codeql/blob/main/javascript/ql/src/Security/CWE-770/MissingRateLimiting.ql
      - js/missing-rate-limiting
```

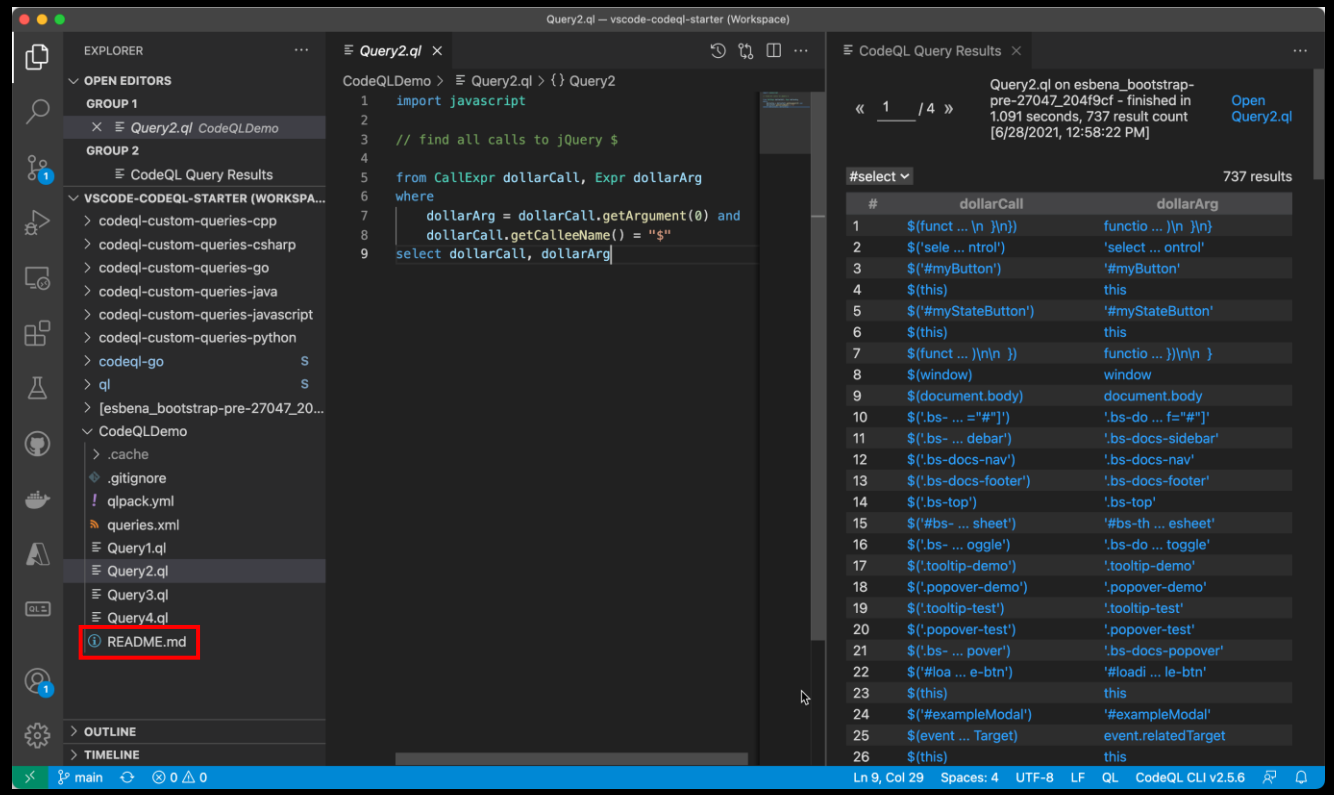
CodeQL Queries propios

https://github.com/wulfland/CodeQLDemo





https://github.com/githubsatelliteworkshops/codeql/releases/download/v1.0/esbena_bootstrap-pre-27047_javascript.zip

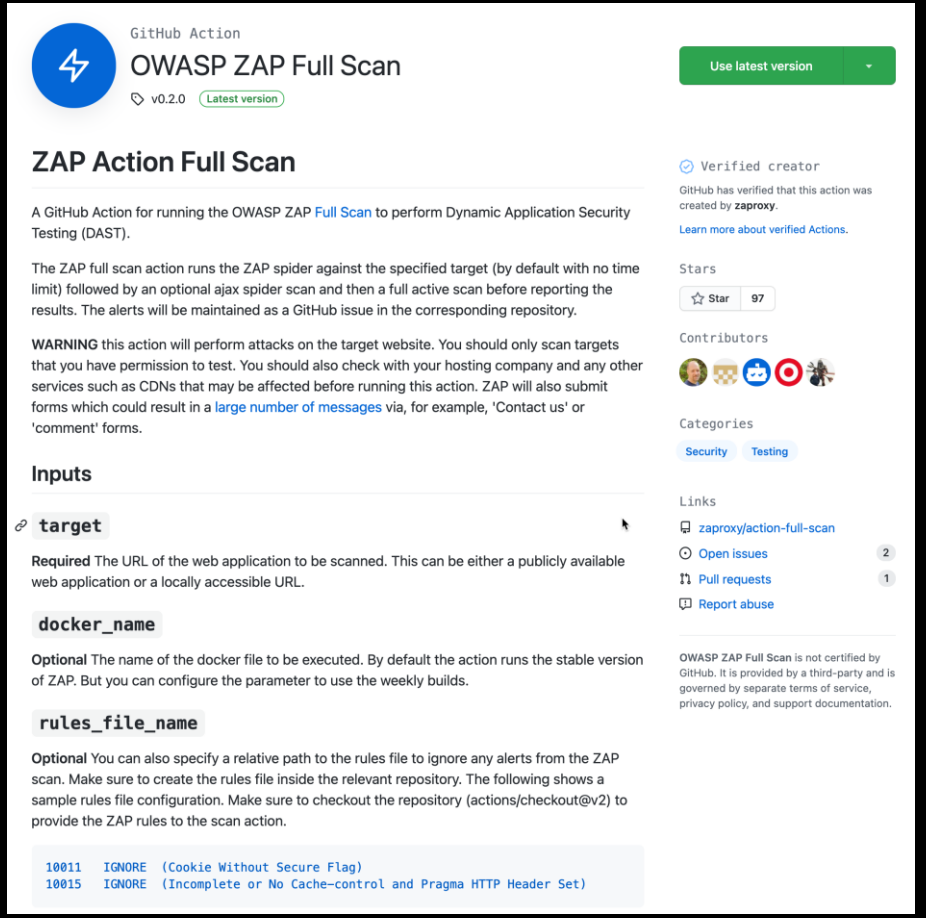


Dynamic Application Security Testing (DAST)

Blackbox-Testing

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

```yaml
jobs:
  owasp:
    name: OWASP Full Scan
    runs-on: ubuntu-latest
    steps:
      - name: OWASP ZAP Full Scan
        uses: zaproxy/action-full-scan@v0.2.0
        with:
          # GitHub Token to create issues in the repository
          token: ${{ github.token }}
          target: https://target
```



Infrastructure Scanning

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

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

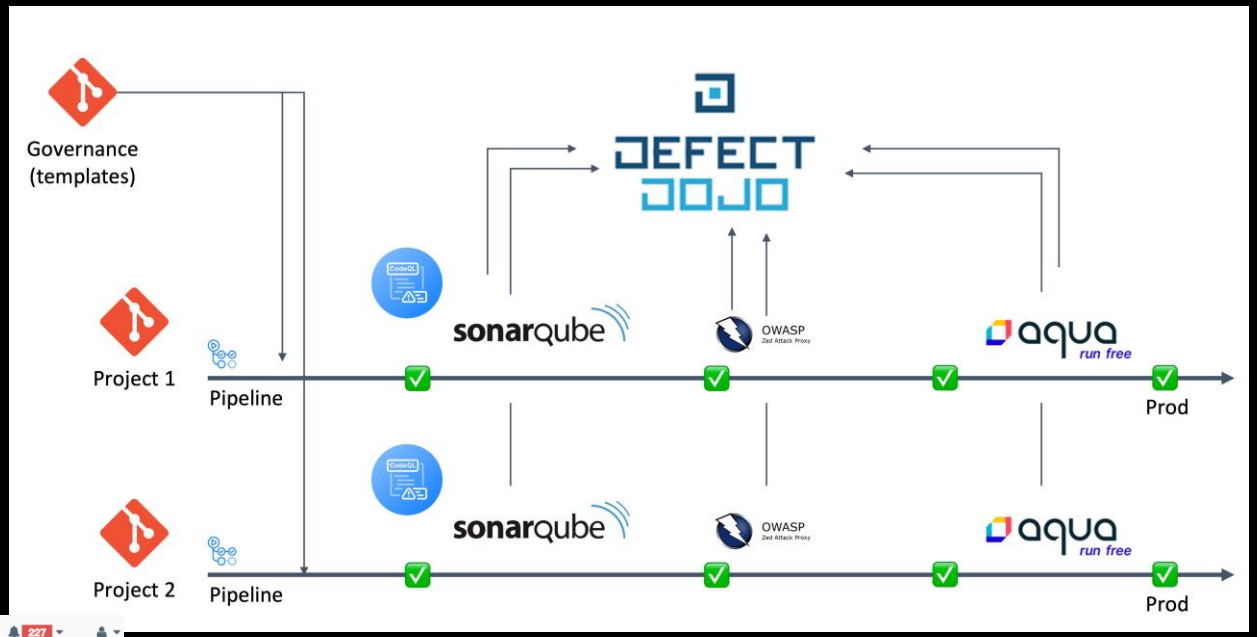
Otra infraestructura:

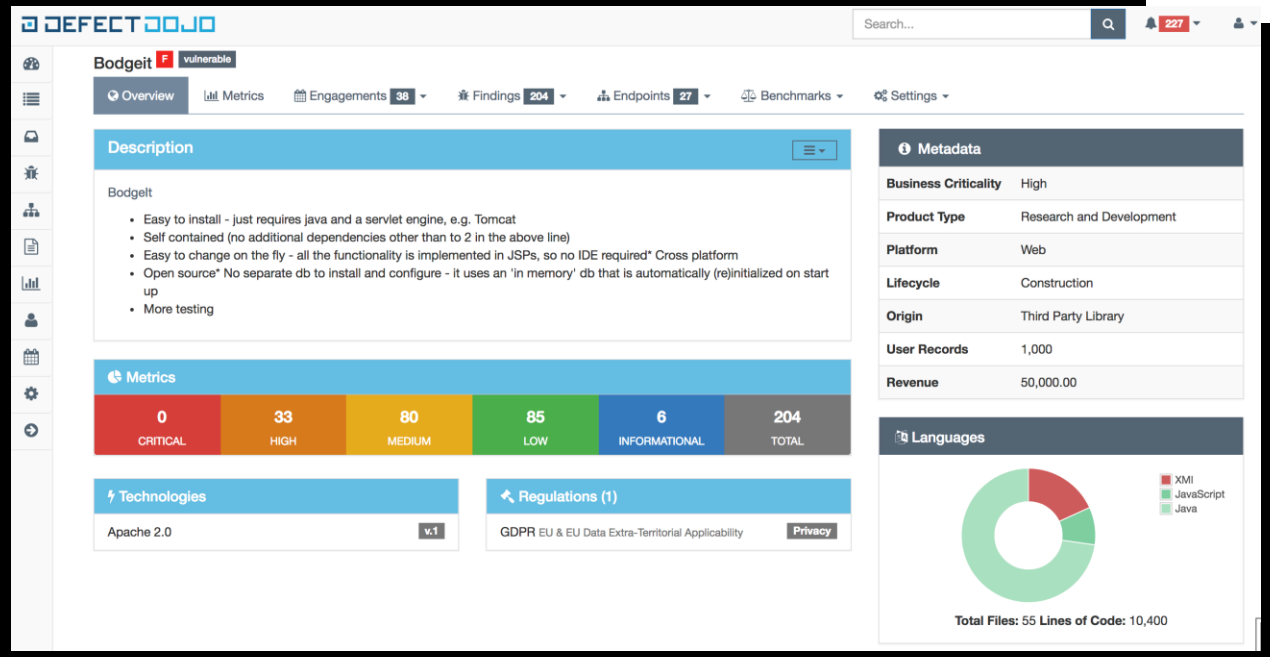
- Azure Policy
  https://docs.microsoft.com/de-de/azure/governance/policy/
- Checkov
  https://www.aquasec.com/products/container-security/
- OpenVAS
  https://www.openvas.org/

```yaml
name: build
on:
  push:
    branches:
      - master
  pull_request:
jobs:
  build:
    name: Build
    runs-on: ubuntu-18.04
    steps:
      - name: Checkout code
        uses: actions/checkout@v2

      - name: Run Trivy vulnerability scanner in repo mode
        uses: aquasecurity/trivy-action@master
        with:
          scan-type: 'fs'
          ignore-unfixed: true
          format: 'template'
          template: '@/contrib/sarif.tpl'
          output: 'trivy-results.sarif'
          severity: 'CRITICAL'

      - name: Upload Trivy scan results to GitHub Security tab
        uses: github/codeql-action/upload-sarif@v1
        with:
          sarif_file: 'trivy-results.sarif'
```

DefectDojo

https://github.com/DefectDojo/django-DefectDojo
Demo: demo.defectdojo.org
Log in: admin / defectdojo@demo#appsec



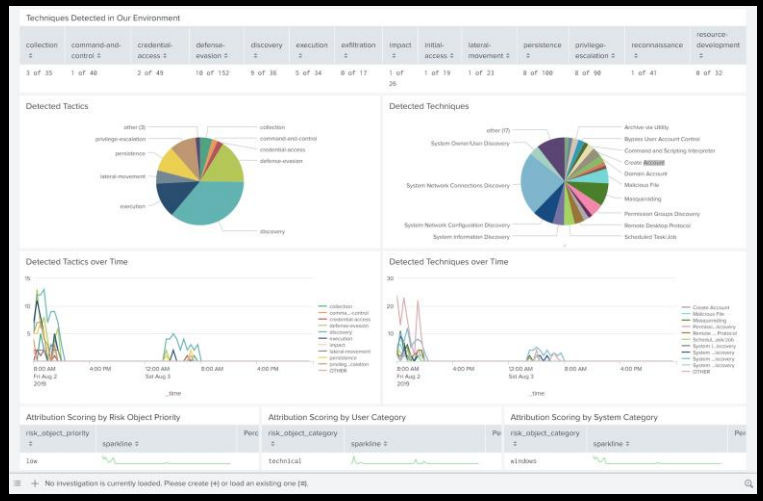


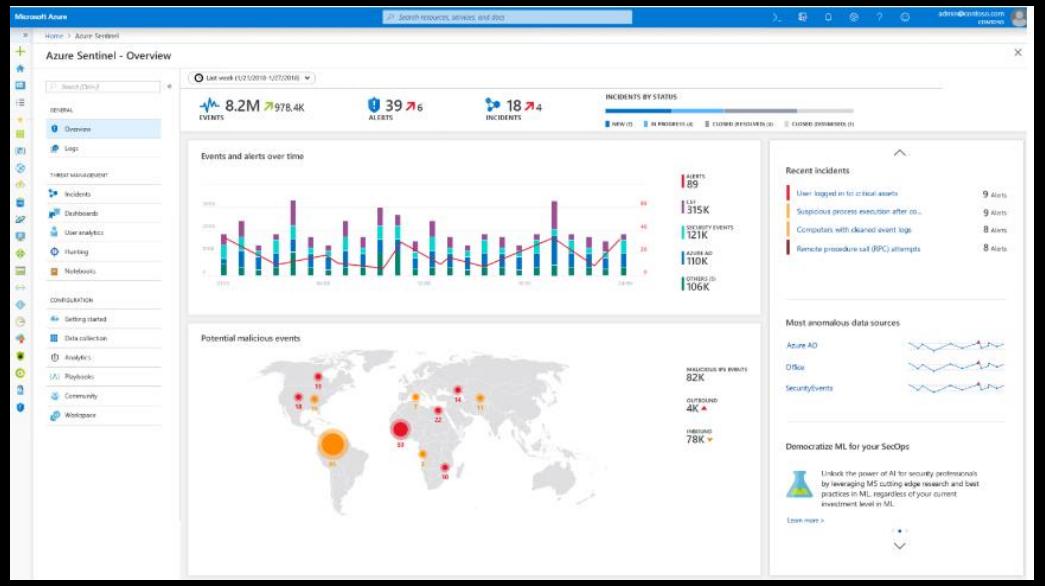
Security Information & Event Management (SIEM)

Ejemplo:

- Azure Sentinel
- Splunk

- Un log central
- Multicloud/Hybrid
- Detección de anomalías (ML)
- Avisos en tiempo real





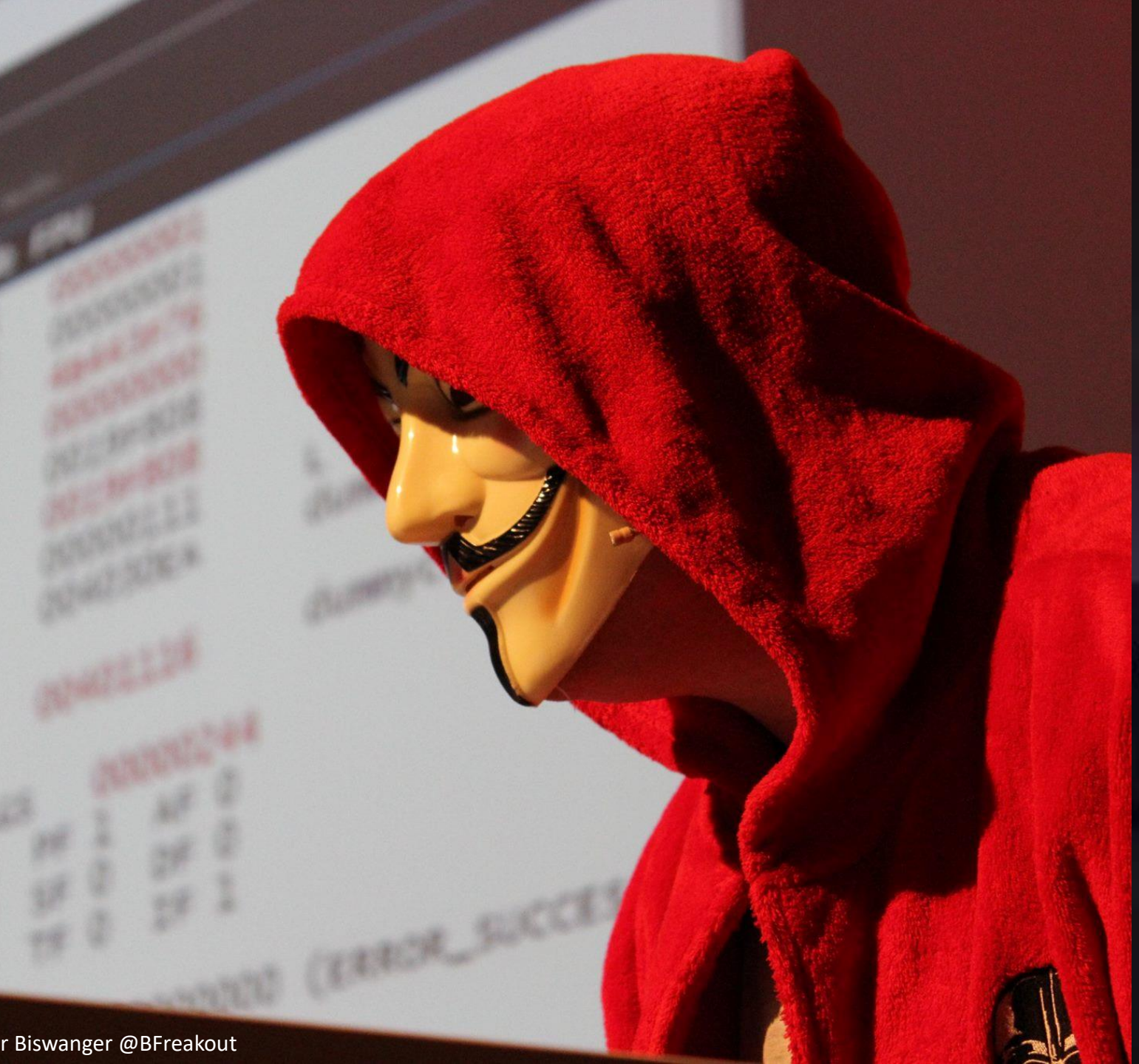
6 tips to integrate security into your DevOps practices

1. **Build a security-first culture across the business**
2. **Integrate security in the early stages of the development lifecycle**
3. **Monitor and observe continuously with purpose**
4. **Embrace everything-as-code**
5. **Realize compliancy with policy automation**
6. **Secure and visualize your software supply chain**

https://azure.microsoft.com/en-us/resources/6-tips-to-integrate-security-into-your-devops-practices/

Cuando hablamos de seguridad…

- ~~“Hasta ahora nunca hemos tenido un incidente con la seguridad”~~
- ~~“Nuestros desarrolladores son expertos”~~
- ~~“Nuestra empresa no es tan importante como para ser atacada”~~
- ~~“Estamos bien protegidos”~~
- ~~“Nuestros cortafuegos son seguros”~~
- “¡Tenemos que hacer algo!”
- “¡La seguridad tiene máxima prioridad!”
- “¡Si hemos tenido un incidente con la seguridad, es que es demasiado tarde!”



Muchas gracias

Blog: https://writeabout.net

Twitter: @mike_kaufmann

GitHub: @wulfland

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