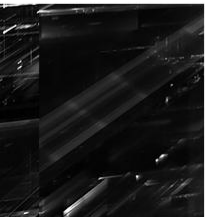


route-detect

Find authentication and authorization security bugs in web application routes



\$ whoami

- Matt Schwager
- Senior Product Security Engineer at Red Canary
- Background in Software Engineering and {App,Prod}Sec
- Interests include: tool development, security research, static analysis testing, fuzz testing, and other automated testing
- Route security research focus in Master's thesis at GA Tech

What's the problem?

- Insecure routes in web app code
- **Routes:** connect URL paths to app code responsible for handling that web request
- **Insecure:** improper authentication (authn) or authorization (authz) logic
- **Authn:** validate who you are
- **Authz:** validate what you can access
- **Roles:** access levels specifying what actions you may perform
 - Endpoint publicly available (no authn)
 - E.g. missing **@RequiresAuthentication** annotation
 - Endpoint accessible by guest accounts (improper authz)
 - E.g. using **@RolesAllowed(ROLE_GUEST)** instead of **ROLE_ADMIN**

Why is it a problem?

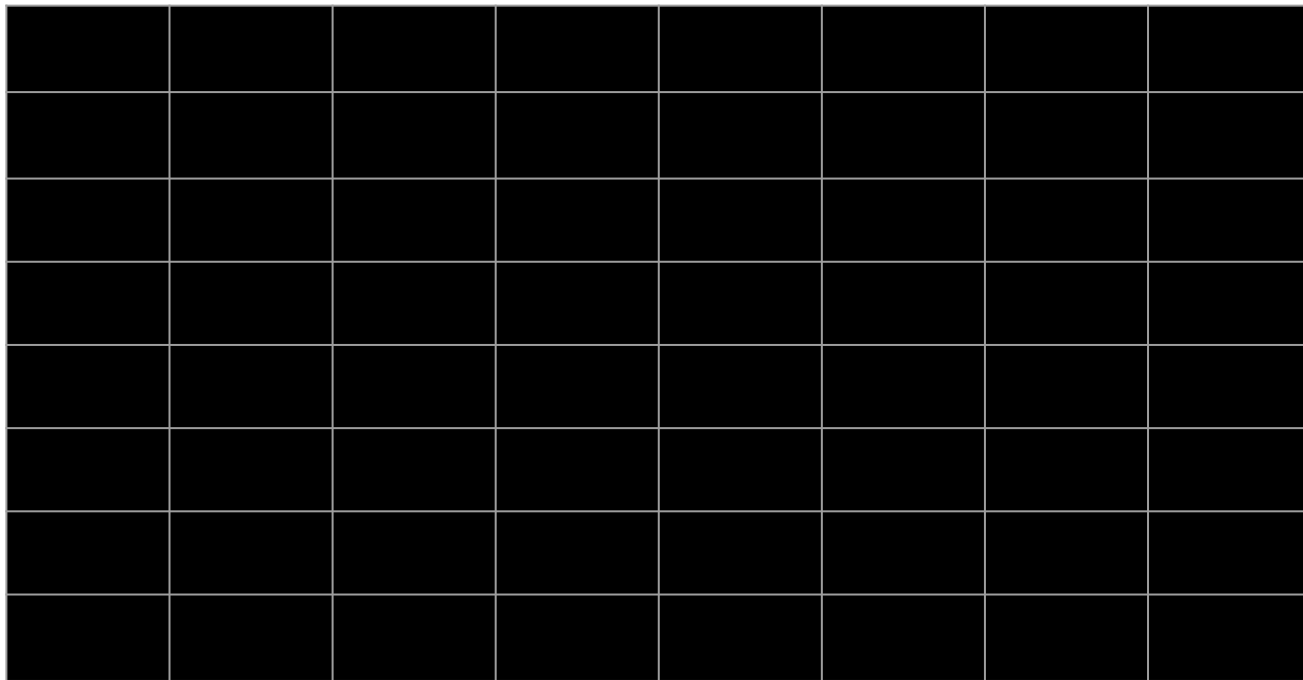
- Complexity
 - Modern web applications have **hundreds or thousands of routes**
 - Authz schemes with **dozens of user roles** or access controls
- Opt-in
 - Authn and authz are typically opt-in vs. opt-out
 - Does **not** follow **secure by default** property
 - Programmer error, forgetfulness, or unfamiliarity with codebase

Evidence

- 2021 OWASP Top 10
 - [#1](#) - Broken **Access Control**
 - [#7](#) - Identification and **Authentication** Failures (formerly Broken Authentication)
- 2023 OWASP API Top 10
 - [#1](#) - Broken Object Level **Authorization**
 - [#2](#) - Broken **Authentication**
 - [#5](#) - Broken Function Level **Authorization**
- 2023 CWE Top 25
 - #11 - [CWE-862](#): Missing **Authorization**
 - #13 - [CWE-287](#): Improper **Authentication**
 - #20 - [CWE-306](#): Missing **Authentication** for Critical Function
 - #24 - [CWE-863](#): Incorrect **Authorization**

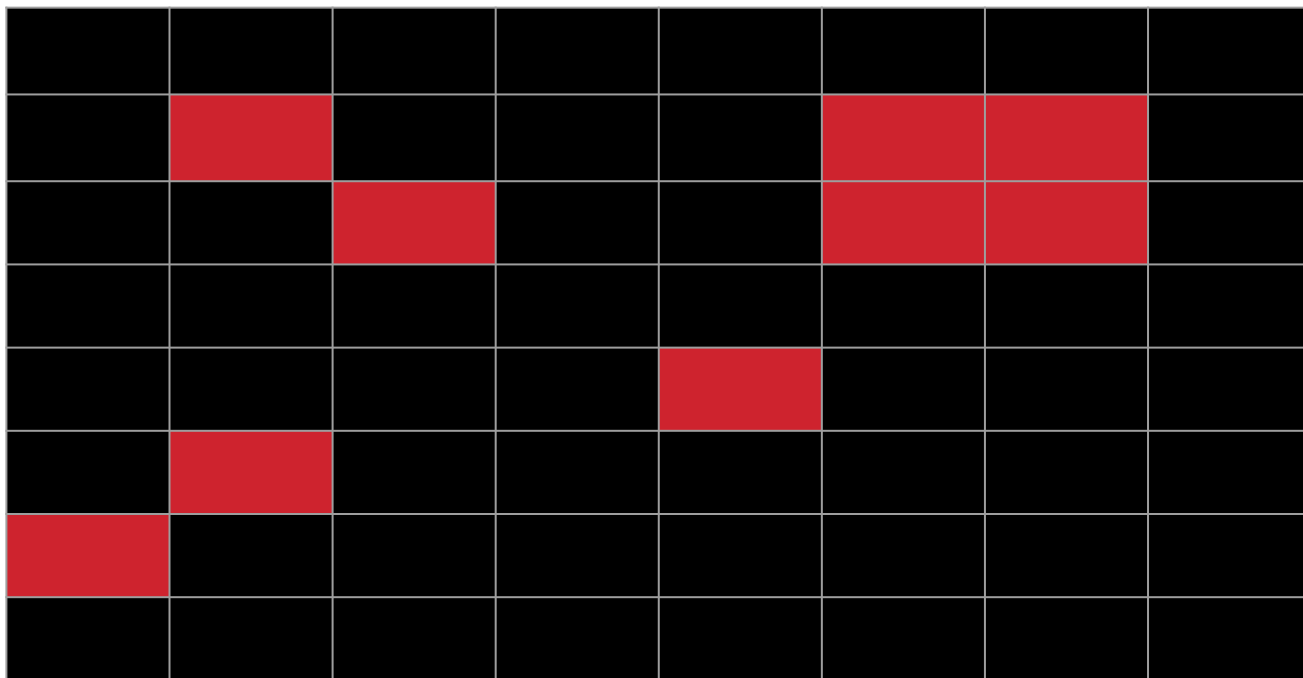
Needle in a haystack problem

Find the insecure route:

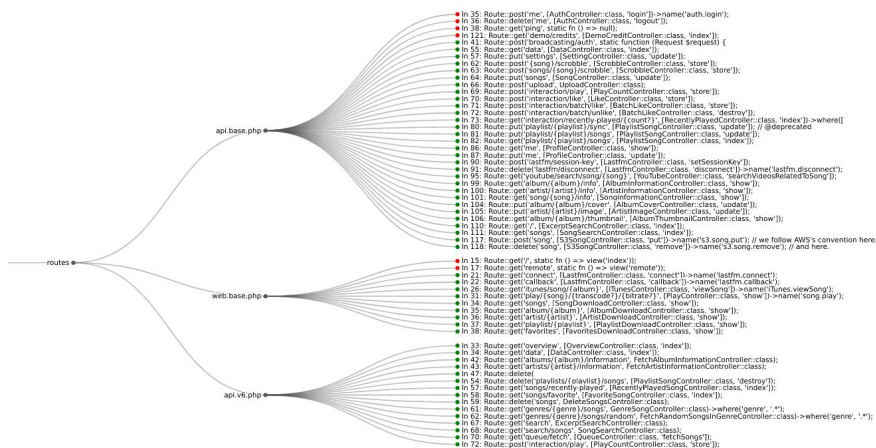


What if needles **glowed** in the dark?

Find the insecure route:



Introducing: route-detect



Routes from *koel* streaming server

- Uses **static analysis** to find web application routes and their authn and authz properties
- Enables security researchers and engineers to quickly analyze and diagnose codebases for route security misconfigurations
- Supports **6 programming languages, 17 web application frameworks**, and **61 authn/authz libraries**
- Favors breadth over depth

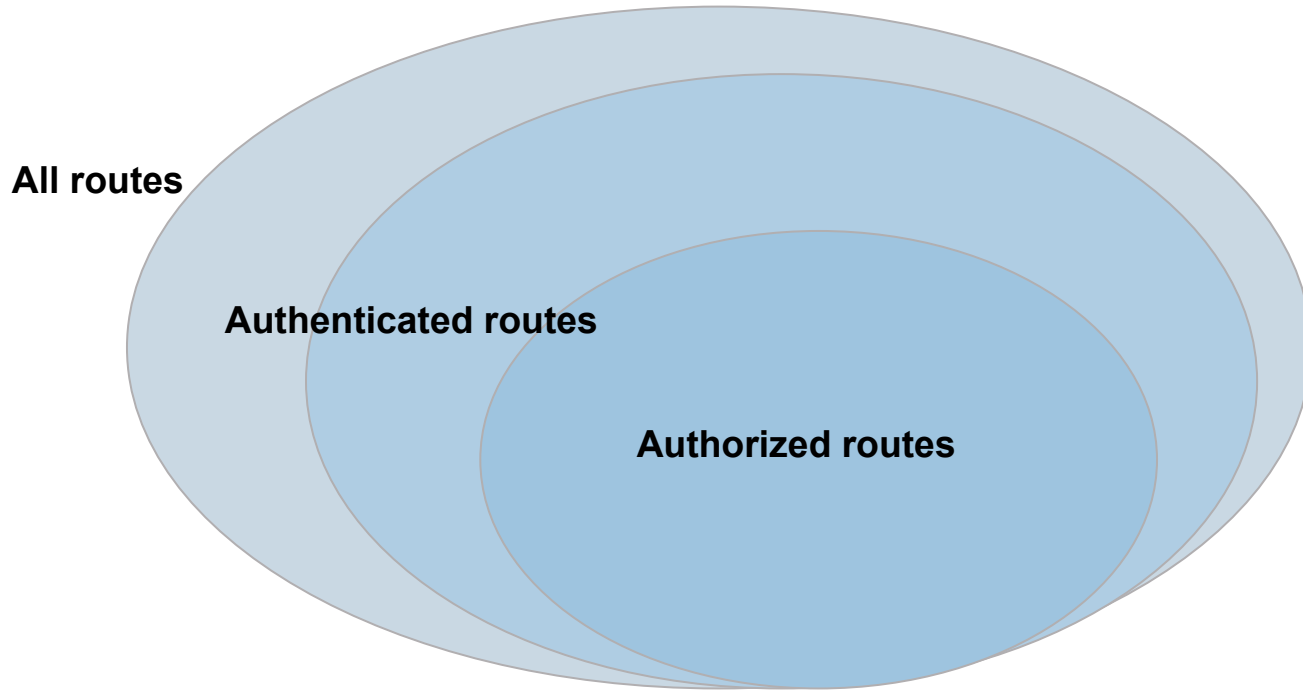
Demo



How does it work?

- Provides an **easily installable, open source, CLI application**
- Builds on Semgrep for code analysis and CLI findings
- Builds on D3.js and local HTML files for visualizations
- Uses Python for "glue" code and application interface
- Heavy use of **automated testing to prevent false positives**
 - E.g. create test code like real findings, ensure route-detect finds it
- Cross-reference **basic regex code search to minimize false negatives**
 - E.g. search for "route", "path", etc, and improve route-detect rules

Finding routes and authn/authz



E.g. Python Flask route authn

rules:

- id: flask-route-unauthenticated

patterns:

- pattern: |
 @\$APP.route(\$PATH, ...)
 def \$FUNC(...):
 ...
- pattern-not: |
 @\$APP.route(\$PATH, ...)
 @login_required(...)
 def \$FUNC(...):
 ...

message: Found unauthenticated Flask route

languages: [python]

severity: INFO

rules:

- id: flask-route-authenticated

pattern: |

 @\$APP.route(\$PATH, ...)
 @login_required(...)
 def \$FUNC(...):
 ...

message: Found authenticated Flask route

languages: [python]

severity: INFO

Limitations

- [Convention over configuration](#), i.e. **implicit** code relationships
 - Ruby Rails
- **Interprocedural** authn/authz information
 - Route information is logically far from authn/authz information
 - Python Django, Ruby Rails
- Middleware-based authn/authz information
 - Explosion in number of ways authn/authz may be specified
 - Golang Gin, Golang Gorilla

What's next?

- Expand horizontally
 - Support more languages, frameworks, and authn/authz libraries
- Expand vertically
 - Deeper analysis, reduce false positives, address limitations, etc.
- Anomaly detection
 - What if all routes in a source code file are authn except one?
 - What if all routes in a directory have the same authz role except one?

— Q&A

Questions, comments, rants?



<https://github.com/mschwager/route-detect>

