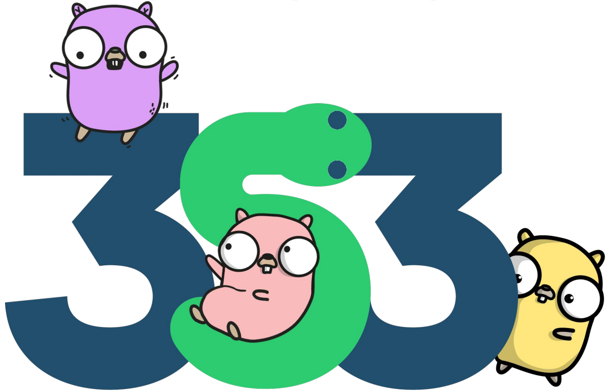
Writing Secure

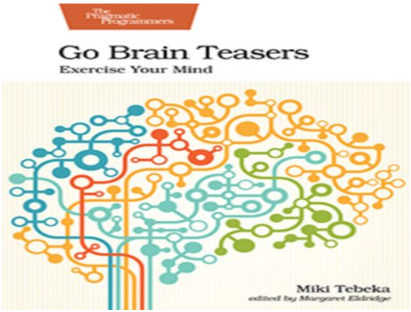
GO

Code

miki @tebeka

CEO, CTO, UFO ...





The Pragmatic Programmers

Go Brain Teasers

Exercise Your Mind

Miki Tebeka

edited by Margaret Eldridge

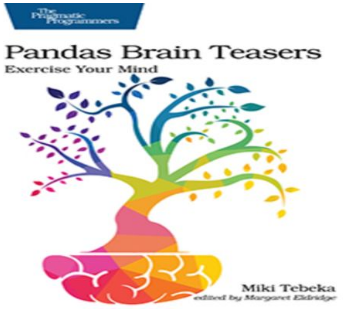
The Pragmatic Programmers

Python Brain Teasers

Exercise Your Mind

Miki Tebeka

edited by Margaret Eldridge



The Pragmatic Programmers

Pandas Brain Teasers

Exercise Your Mind

Miki Tebeka

edited by Margaret Eldridge

in Learning

First rule of computer security: **don't buy a computer**.

Second rule: if you buy one, **don't turn it on**.

- Dark Avenger

GO

Go Security Policy

OWASP Top Ten

A01:2021-Broken Access Control

A02:2021-Cryptographic Failures

A03:2021-Injection

A04:2021-Insecure Design

A05:2021-Security Misconfiguration

A06:2021-Vulnerable and Outdated Components

A07:2021-Identification and Authentication Failures

A08:2021-Software and Data Integrity Failures

A09:2021-Security Logging and Monitoring Failures

A10:2021-Server-Side Request Forgery

Input

A03:2021-Injection

A08:2021-Software and Data Integrity Failures

A10:2021-Server-Side Request Forgery

Output

```
A02:2021-Cryptographic Failures
A03:2021-Injection
```

Authentication & Authorization

A01:2021-Broken Access Control

A07:2021-Identification and Authentication Failures

A05:2021-Security Misconfiguration

Infrastructure

A04:2021-Insecure Design

A06:2021-Vulnerable and Outdated Components

A09:2021-Security Logging and Monitoring Failures

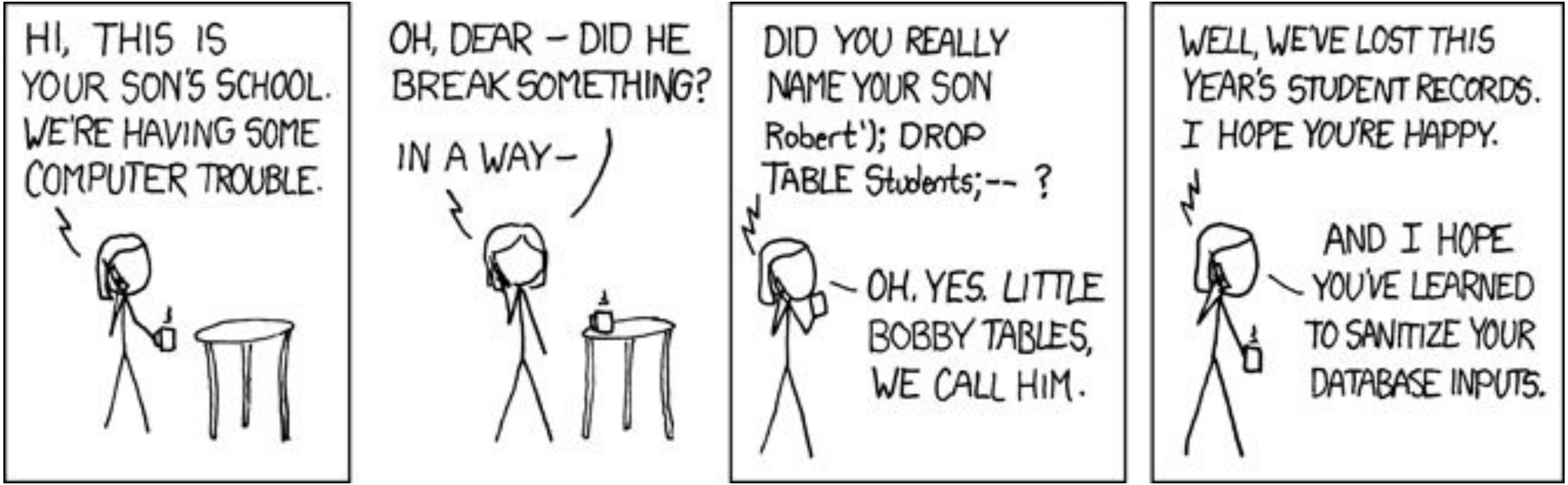
Input

> In general, it is best to assume that the network is filled with malevolent entities that will send in packets designed to have the worst possible effect.
>
> — RFC 1122

A03:2021-Injection

- User-supplied data is **not validated, filtered, or sanitized** by the application.
- **Dynamic queries** or non-parameterized calls without context-aware **escaping** are used directly in the interpreter.
- Hostile data is used within object-relational mapping (ORM) search parameters to **extract additional, sensitive records**.
- Hostile data is directly used or concatenated. The SQL or command contains the structure and **malicious data in dynamic queries, commands, or stored procedures**.

database/sql


HI, THIS IS YOUR SON'S SCHOOL. WE'RE HAVING SOME COMPUTER TROUBLE.
OH, DEAR – DID HE BREAK SOMETHING?
IN A WAY–
DID YOU REALLY NAME YOUR SON Robert'); DROP TABLE Students;-- ?
OH. YES. LITTLE BOBBY TABLES, WE CALL HIM.
WELL, WE'VE LOST THIS YEAR'S STUDENT RECORDS. I HOPE YOU'RE HAPPY.
AND I HOPE YOU'VE LEARNED TO SANITIZE YOUR DATABASE INPUTS.

https://xkcd.com/327/

```sql
INSERT INTO logs (
    time, level, message
) VALUES (
    @time, @level, @message
);
```

```go
_, err := d.db.ExecContext(
     ctx, addSQL,
     sql.Named("time", entry.Time),
     sql.Named("level", entry.Level),
     sql.Named("message", entry.Message),
)
```

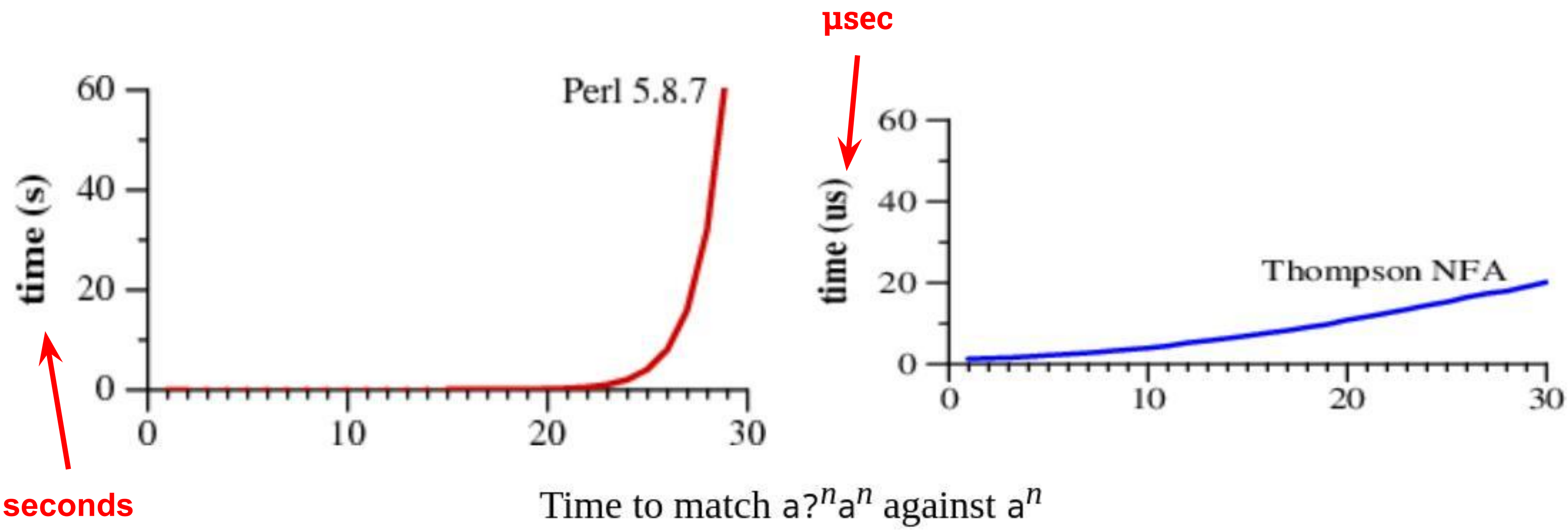
os/exec

```
cmd := exec.Command(
    "bash", "-c", "echo pwned > /etc/passwd"
)
```

```
cmd := exec.Command(
    "echo", "pwned > /etc/passwd",
)
```

regexp

> Unfortunately, last Tuesday’s update contained a **regular expression** that backtracked enormously and **exhausted CPU** used for HTTP/HTTPS serving.
>
> — Cloudflare 27 minute outage



µsec

seconds

Regular Expression Matching Can Be Simple And Fast by Russ Cox go1.17.8 (released 2022-03-03) includes a security fix to the `regexp/syntax` package, as well as …

- 1.17.8 release notes

A08:2021-Software and Data Integrity Failures

… **updates are downloaded** without sufficient **integrity verification** and applied to the previously trusted application. …

Another example is where objects or data are **encoded or serialized** into a structure that an attacker can see and modify is vulnerable to i**nsecure deserialization.**

```
<?xml version="1.0"?>
<!DOCTYPE lolz [
 <!ENTITY lol "lol">
 <!ELEMENT lolz (#PCDATA)>
 <!ENTITY lol1 "&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;">
 <!ENTITY lol2 "&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;">
 <!ENTITY lol3 "&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;">
 <!ENTITY lol4 "&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;">
 <!ENTITY lol5 "&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;">
 <!ENTITY lol6 "&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;">
 <!ENTITY lol7 "&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;">
 <!ENTITY lol8 "&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;">
 <!ENTITY lol9 "&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;">
]>
<lolz>&lol9;</lolz>
```

Billion laughs attack

Java Hangs When Converting 2.2250738585072012e-308

Exploring Binary

```
Accept-Language: en-US,en;q=0.5
```

Valid JSON/XML/…

<<

Valid Data

- https://cuelang.org/
- https://github.com/go-playground/validator
- …

```
$ curl -I https://bit.ly/3lTQ9fF
HTTP/1.1 200 OK
x-amz-request-id: PA6K4665EFZQ5846
Date: Mon, 30 May 2022 14:40:52 GMT
Accept-Ranges: bytes
Server: AmazonS3
Content-Length: 110439634
```

```
const maxSize = 1 << 20 // MB
lr := io.LimitReader(r.Body, maxSize)
dec := json.NewDecoder(lr)
```

Output

A03:2021-Injection

- User-supplied data is not validated, filtered, or sanitized by the application.
- Dynamic queries or non-parameterized calls without context-aware escaping are used directly in the interpreter.
- **Hostile data is used within object-relational mapping (ORM) search parameters to extract additional, sensitive records.**
- Hostile data is directly used or concatenated. The SQL or command contains the structure and malicious data in dynamic queries, commands, or stored procedures.

~~text/template~~

html/template

```
enc := json.NewEncoder(os.Stdout)
msg := map[string]interface{}{
    "error":   nil,
    "message": `<script>alert(\"Hi\")</script>`,
}
enc.Encode(msg)
```

```
{
  "error": null,
  "message":
"\u003cscript\u003ealert(\\\"Hi\\\")\u003c/scriptu003e"
}
```

A02:2021-Cryptographic Failures

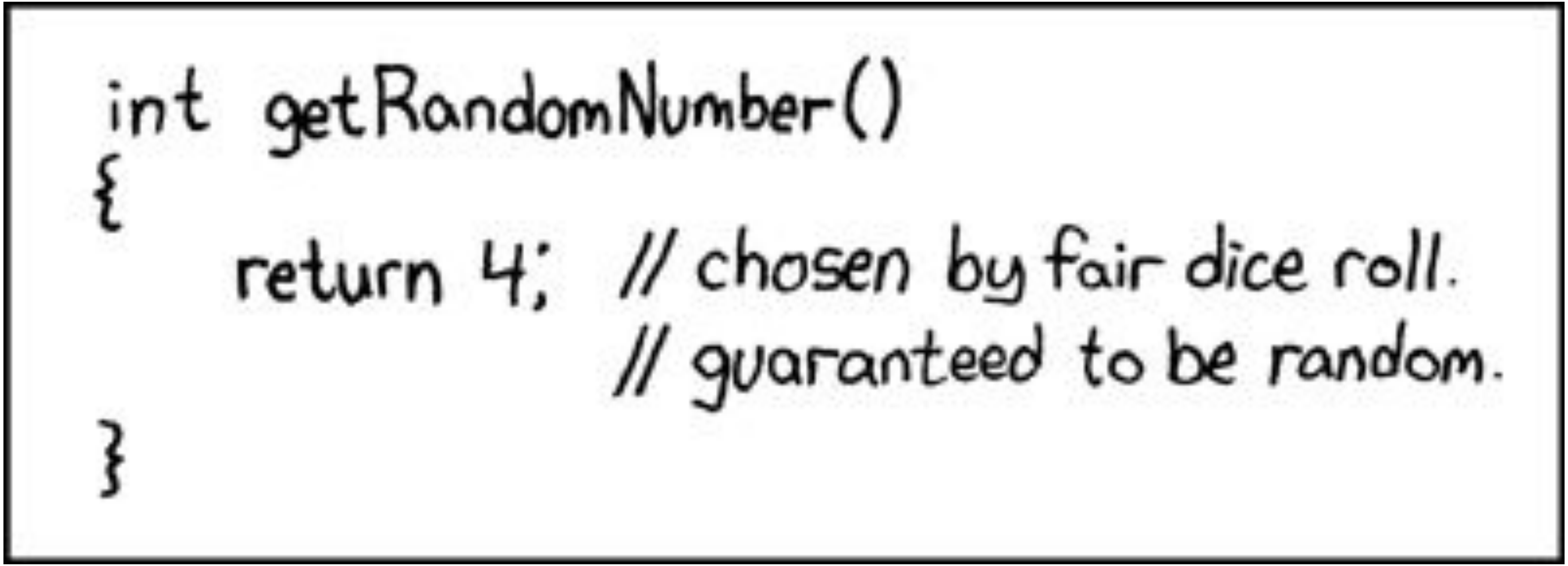
The first thing is to determine the **protection needs of data in transit and at rest**. For example, passwords, credit card numbers, health records, personal information, and business secrets require extra protection, mainly if that data falls under **privacy laws**, ...

```go
package main

import (
	"fmt"
	"math/rand"
)

func main() {
	fmt.Println(rand.Intn(100))
}
```

```
$ for i in $(seq 3); do go run rnd.go; done
81
81
81
```



```
int getRandomNumber()
{
    return 4;  // chosen by fair dice roll.
               // guaranteed to be random.
}
```

https://xkcd.com/221/

```go
package main

import (
   "fmt"
   "math/rand"
)

func main() {
   rand.Seed(time.Now().UnixNano())
   fmt.Println(rand.Intn(100))
}
```

```
$ for i in $(seq 3); do go run rnd_seed.go; done
17
64
0
```

How We Learned to Cheat at Online Poker: A Study in Software Security

- cigital

~~math/rand~~

crypto/rand

```
type error interface {
  Error() string
}
```

```go
type AuthError struct {
  Session string
  Reason  string
  User    *User
}

func (ae *AuthError) Error() string {
  return ae.Reason
}
```

```
reply := map[string]interface{}{
   "error": err,
}
w.WriteHeader(http.StatusUnauthorized)
w.Header().Set("Content-Type", "application/json")
json.NewEncoder(w).Encode(reply)
```

```
{
  "error": {
    "Session": "a3223ab91f3b42c3a587647e34708b04",
    "Reason": "Bad password",
    "User": {
      "Login": "Elliot"
    }
  }
}
```

Authentication

A07:2021- Identification and Authentication Failures

Confirmation of the **user's identity**, authentication, and **session management** is critical to protect against authentication-related attacks. There may be authentication weaknesses if the application…

- Basic
- Oauth2
- JWT
- OICD
- ...

If having a coffee in the morning doesn't wake you up, try deleting a table in a production database instead.

- Juozas Kaziukenas

A01:2021-Broken Access Control

Access control enforces policy such that users cannot act outside of their intended permissions. Failures typically lead to unauthorized information disclosure, modification, or destruction of all data or performing a business function outside the user's limits. Common access control vulnerabilities include...

- ACL
- RBAC
- …

Infrastructure

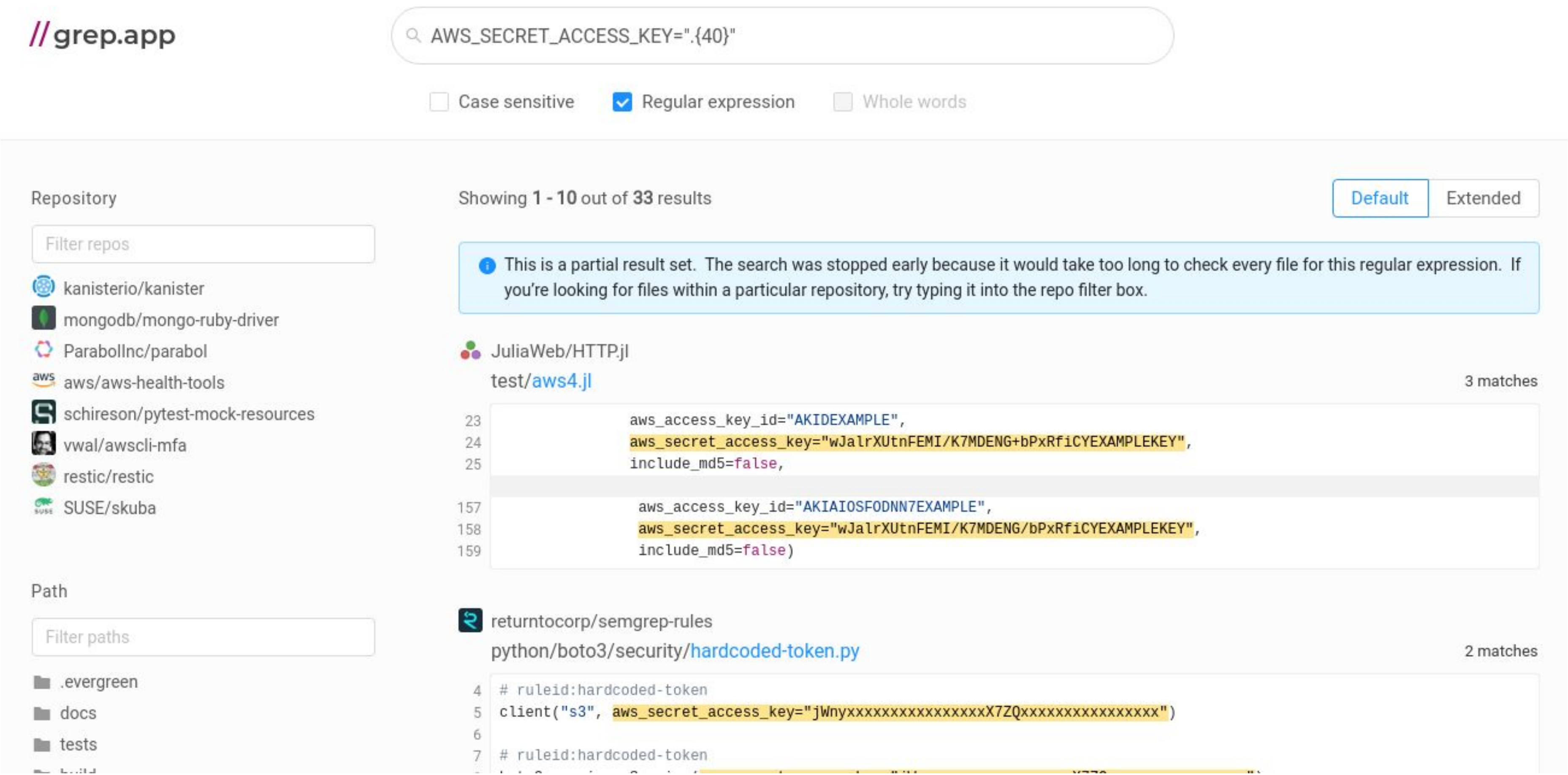
A05:2021-Security Misconfiguration

- Missing appropriate **security hardening** …
- Unnecessary features are enabled or installed (e.g., unnecessary ports, services, pages, accounts, or **privileges**).
- **Default accounts** and their passwords are still enabled and unchanged.
- **Error handling reveals** stack traces or other **overly informative** error messages to users.
- For upgraded systems, the **latest security features** are **disabled** or **not configured** securely.
- The **security settings** in the application servers, application frameworks … are **not set to secure values**.
- The server does not send **security headers** or directives, or they are not set to secure values.
- The software is **out of date** or vulnerable …

Datensparsamkeit

Datensparsamkeit is a German word that's difficult to translate properly into English. It's an **attitude** to how we capture and store data, saying that **we should only handle data that we really need.**

- Martin Fowler



//grep.app

AWS_SECRET_ACCESS_KEY=".{40}"

□ Case sensitive ☑ Regular expression □ Whole words

Repository

Filter repos

- kanisterio/kanister
- mongodb/mongo-ruby-driver
- ParabolInc/parabol
- aws/aws-health-tools
- schireson/pytest-mock-resources
- vwal/awscli-mfa
- restic/restic
- SUSE/skuba

Path

Filter paths

- .evergreen
- docs
- tests
- build

Showing **1 - 10** out of **33** results

This is a partial result set. The search was stopped early because it would take too long to check every file for this regular expression. If you're looking for files within a particular repository, try typing it into the repo filter box.

JuliaWeb/HTTP.jl

test/aws4.jl 3 matches

```
                aws_access_key_id="AKIDEXAMPLE",
                aws_secret_access_key="wJalrXUtnFEMI/K7MDENG+bPxRfiCYEXAMPLEKEY",
                include_md5=false,

                 aws_access_key_id="AKIAIOSFODNN7EXAMPLE",
                 aws_secret_access_key="wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY",
                 include_md5=false)
```

returntocorp/semgrep-rules

python/boto3/security/hardcoded-token.py 2 matches

```
# ruleid:hardcoded-token
client("s3", aws_secret_access_key="jWnyxxxxxxxxxxxxxxxxX7ZQxxxxxxxxxxxxxxxx")

# ruleid:hardcoded-token
```

```
err := srv.ListenAndServeTLS("cert.pem","key.pem")
if err != nil {
   log.Fatal(err)
}
```

x/crypto/acme/autocert

```go
srv := &http.Server{
   Addr:              ":" + httpsPort,
   ReadTimeout:       1 * time.Second,
   WriteTimeout:      1 * time.Second,
   IdleTimeout:       30 * time.Second,
   ReadHeaderTimeout: 2 * time.Second,
   Handler:           r,
}
```

github.com/gorilla/websocket

A06:2021-Vulnerable and Outdated Components

- If you do not **know the versions of all components you use** … This includes components you directly use as well as **nested dependencies**.
- If the software is **vulnerable, unsupported, or out of date**. …
- If **you do not scan for vulnerabilities regularly** and **subscribe** to security bulletins related to the components you use.
- If you do not **fix or upgrade** the underlying platform, frameworks, and dependencies in a risk-based, timely fashion. …
- If software developers do not **test the compatibility** of updated, upgraded, or patched libraries.
- If you do not secure the components' configurations…

```
$ go version
go version go1.19.2 linux/amd64
```

Our Software Dependency Problem

- Russ Cox

… every time I add the sqlite amalgamation — sqlite.c and sqlite.h — to my @PlatformIO_Org project, **it adds Doom as one of my dependencies.**

- Joe Castillo

```
$ cat go.mod
module github.com/tebeka/flags

go 1.18
```

- Go CVE List
- golang-announce
- golang/x/vuln

A09:2021-Security Logging and Monitoring Failures

- **Auditable events**, such as **logins**, failed logins, and high-value transactions, are not logged.
- Warnings and errors generate no, inadequate, or **unclear log messages**.
- Logs of applications and APIs are not **monitored** for suspicious activity.
- Logs are only **stored locally**.
- **Appropriate alerting** thresholds and response escalation processes are not in place or effective.
- **Penetration testing and scans** by dynamic application security testing (DAST) tools (such as OWASP ZAP) **do not trigger alerts**.
- The application **cannot detect**, escalate, or alert for active **attacks** in real-time or near real-time.

1. Time stamp from a trusted system component
2. Severity rating for each event
3. Tagging of security relevant events, if they are mixed with other log entries
4. Identity of the account/user that caused the event
5. Source IP address associated with the request
6. Event outcome (success or failure)
7. Description of the event

OWASP Secure Coding Practices - Quick Reference Guide

- `log`
- `go.uber.org/zap`
- ...

- expvar
- prometheus
- ...

The Security Mindset

Bruce Schneier

Your 80's band name is a combination of the street you were born in and your mother's maiden name.

Culture > Process

Questions?

Thank You!

@tebeka
miki@353solutions.com

353SOLUTIONS

LEARN FROM THE EXPERTS