



Тодор Тодоров

Senior Software Engineer @ Payhawk

How .NET Apps Get Hacked?



PS /how-dotnet-apps-get-hacked> whoami

Payhawk



Senior Software Engineer



Chapter Leader

DEV.BG



Speaker

XAEP.BG



Community Supporter



OSS Contributor



Todor Todorov



OWASP
SOFIA, BULGARIA

Open
Worldwide
Application
Security
Project

A nonprofit foundation
that works to improve the
security of software.

OWASP Top 10:2025

Welcome to the OWASP Top 10:2025 Release.

The OWASP Top 10 is a standard awareness document for developers and web application security. It represents a broad consensus about the most critical security risks to web applications.

About This Release

This is the **2025** version of the OWASP Top 10. This version includes updates based on the latest data and security trends.

Main Project Page

The [main project page](#) has information about older versions and metadata about this project.

Getting Started

Start with the [Introduction](#) to learn about what's new in the 2025 version.

Top 10:2025 List

1. A01:2025 - Broken Access Control
2. A02:2025 - Security Misconfiguration
3. A03:2025 - Software Supply Chain Failures
4. A04:2025 - Cryptographic Failures
5. A05:2025 - Injection
6. A06:2025 - Insecure Design
7. A07:2025 - Authentication Failures
8. A08:2025 - Software or Data Integrity Failures
9. A09:2025 - Security Logging and Alerting Failures
10. A10:2025 - Mishandling of Exceptional Conditions



1. A01 - Broken Access Control
2. A02 - Security Misconfiguration
3. A04 - Cryptographic Failures
4. A05 - Injection
5. A06 - Insecure Design

A01 - Broken Access Control

DEMO



What **NOT** to do

© Assume HTTP request is checked before you (Microservices)



RE USE
CYCLE
PEAT



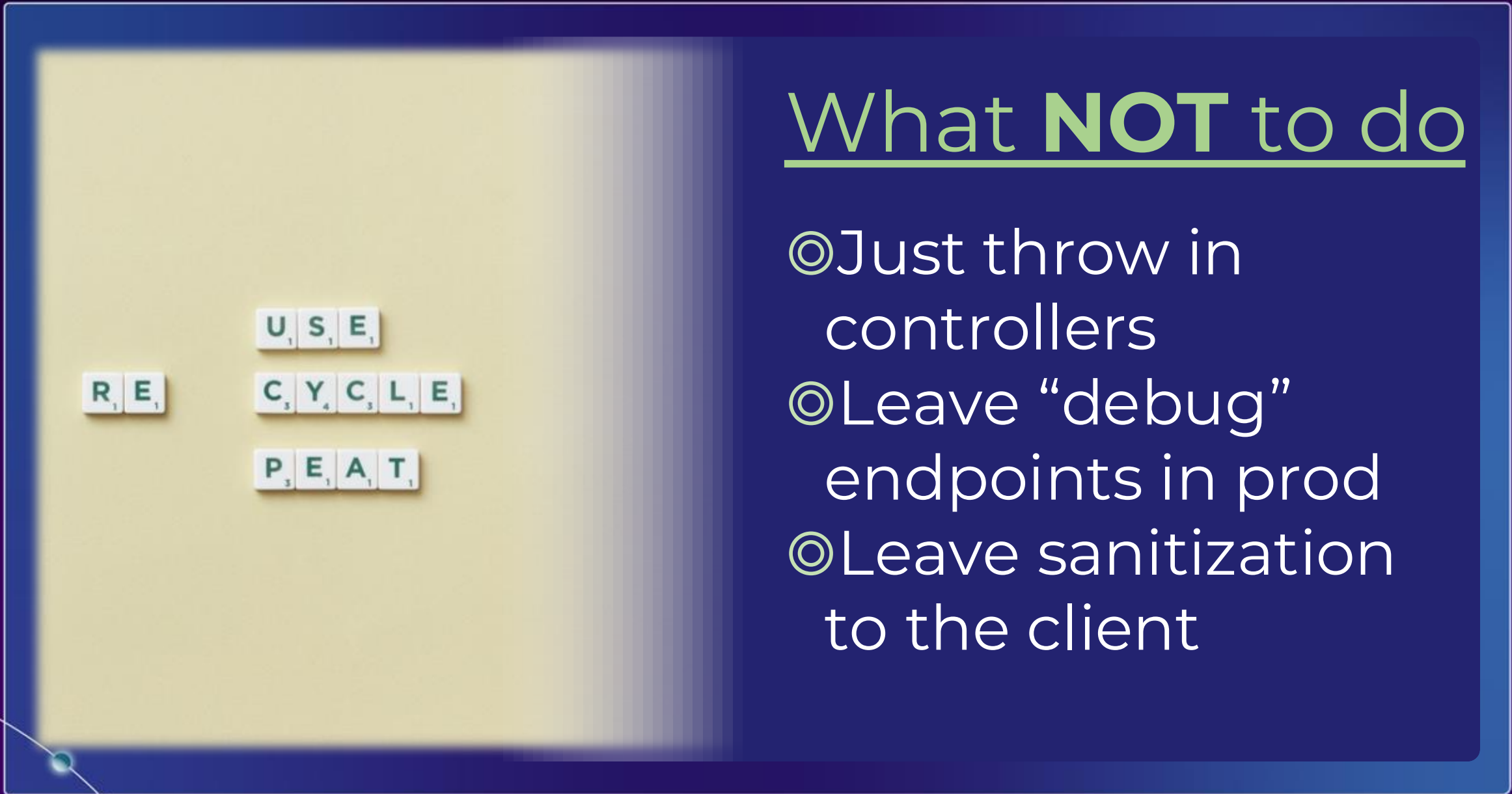
1. A01 - Broken Access Control
2. A02 - Security Misconfiguration
3. A04 - Cryptographic Failures
4. A05 - Injection
5. A06 - Insecure Design

A02 - Security Misconfiguration

DEMO



@totollygeek



What **NOT** to do

- © Just throw in controllers
- © Leave “debug” endpoints in prod
- © Leave sanitization to the client



1. A01 - Broken Access Control
2. A02 - Security Misconfiguration
3. A04 - Cryptographic Failures
4. A05 - Injection
5. A06 - Insecure Design

A04 - Cryptographic Failures

DEMO



What **NOT** to do

- © Just return what the DB gave you
- © Store passwords with MD5



RE USE
CYCLE
PEAT



1. A01 - Broken Access Control
2. A02 - Security Misconfiguration
3. A04 - Cryptographic Failures
4. A05 - Injection
5. A06 - Insecure Design

A05 - Injection

DEMO



@totollygeek

What **NOT** to do

- © Take parameters sanitization for granted
- © Pass arguments to shell directly



RE USE
CYCLE
PEAT



1. A01 - Broken Access Control
2. A02 - Security Misconfiguration
3. A04 - Cryptographic Failures
4. A05 - Injection
5. A06 - Insecure Design

DEMO



What **NOT** to do

© Use your internal entities for requests



There is
more than
just your
code...



Malicious I Malware

CYBERATTACKS & DATA BREACHES

Ravie Lakshmanan

Supply Chain Attackers (Ar

The Shai-Hulud 2.0 campaign version will be exponential

December 22, 2025

Infosecuritiy Magazine Home » News » Researchers Uncover 454,000+ Malicious Open Source Packages

The Hacker News

zscaler REPORT ThreatLabz 2026 VPN Risk Report DOWNLOAD NOW

NEWS 28 January 2026

Research Open

Axios Supply Chain Attack Pushes Cross-Platform RAT via Compromised npm Account

Ravie Lakshmanan Mar 31, 2026

Open Source / Supply Chain Attack

State of the software supply chain

1,233,219



OPEN SOURCE MALWARE PACKAGES LOGGED BY SONATYPE SINCE 2019

9.8 TRILLION



DOWNLOADS ACROSS MAVEN CENTRAL, PYPI, NPM AND NUGET

27.76%



RECOMMENDED DEPENDENCY UPGRADE HALLUCINATION RATE OBSERVED WITH LEADING LLM

65%



OF OPEN SOURCE CVES WERE LEFT WITHOUT CVSS BY THE NVD

Search for packages...

ConsoleMagic 1.0.1

.NET Standard 2.0 .NET Framework 4.0

.NET CLI PMC PackageReference CPM Paket CLI Script & Interactive File-based Apps Cake

```
> dotnet add package ConsoleMagic --version 1.0.1
```

Copy

README Frameworks Dependencies Used By Versions Release Notes

Style your C# console output!

Downloads [Full stats →](#)

Total **69K**

Current version **1024**

Per day average **250**

About

Last updated 04/18/2026

[License Info](#)

[Download package](#) (231.39 KB)

[Open in NuGet Package Explorer](#)

[Open in NuGet Trends](#)

[Report package](#)

Owners [Contact owners →](#)

 totollylegit

Style **Styled** Output
Colourful Colourful Console
Command Line ASCII
Art FIGlet

© Legit AF. All rights Reserved.



A03 - Software Supply Chain Failures

DEMO





TELL ME HOW

**YOU DON'T USE
EXTERNAL DEPENDENCIES**

imgflip.com



Visual Studio Code




Visual Studio Code



ELECTRON

electron TS

27.0.3 • Public • Published 3 days ago

 [Readme](#)

 [Code](#) Beta

 3 Dependencies

 1,256 Dependents

 1,107 Versions



ELECTRON

 passing  passing  2285 online

 Available Translations:        . View these docs in other languages on our [Crowdin](#) project.

The Electron framework lets you write cross-platform desktop applications using JavaScript, HTML and CSS. It is based on [Node.js](#) and [Chromium](#) and is used by the [Atom editor](#) and many other [apps](#).

Follow [@electronjs](#) on Twitter for important announcements.

Install

```
> npm i electron
```

Repository

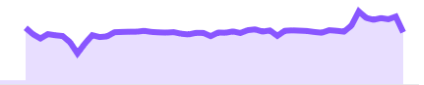
 github.com/electron/electron

Homepage

 github.com/electron/electron#readme

Weekly Downloads

635,911



Version


27.0.3

License

MIT

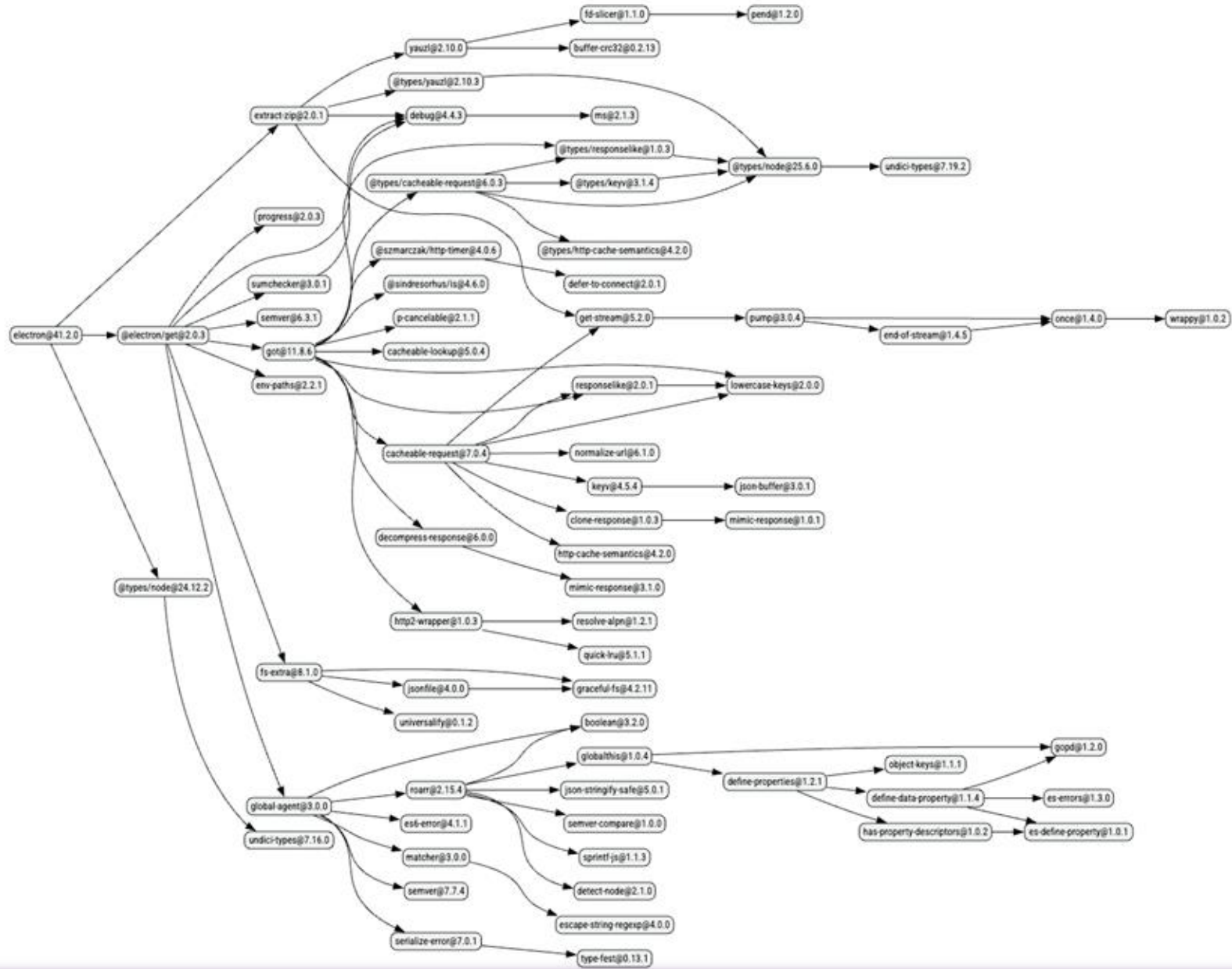
electron TS

27.0.3 • Public • Published 3 days ago



































































































 [Readme](#)

 [Code](#) Beta

 [3 Dependencies](#)

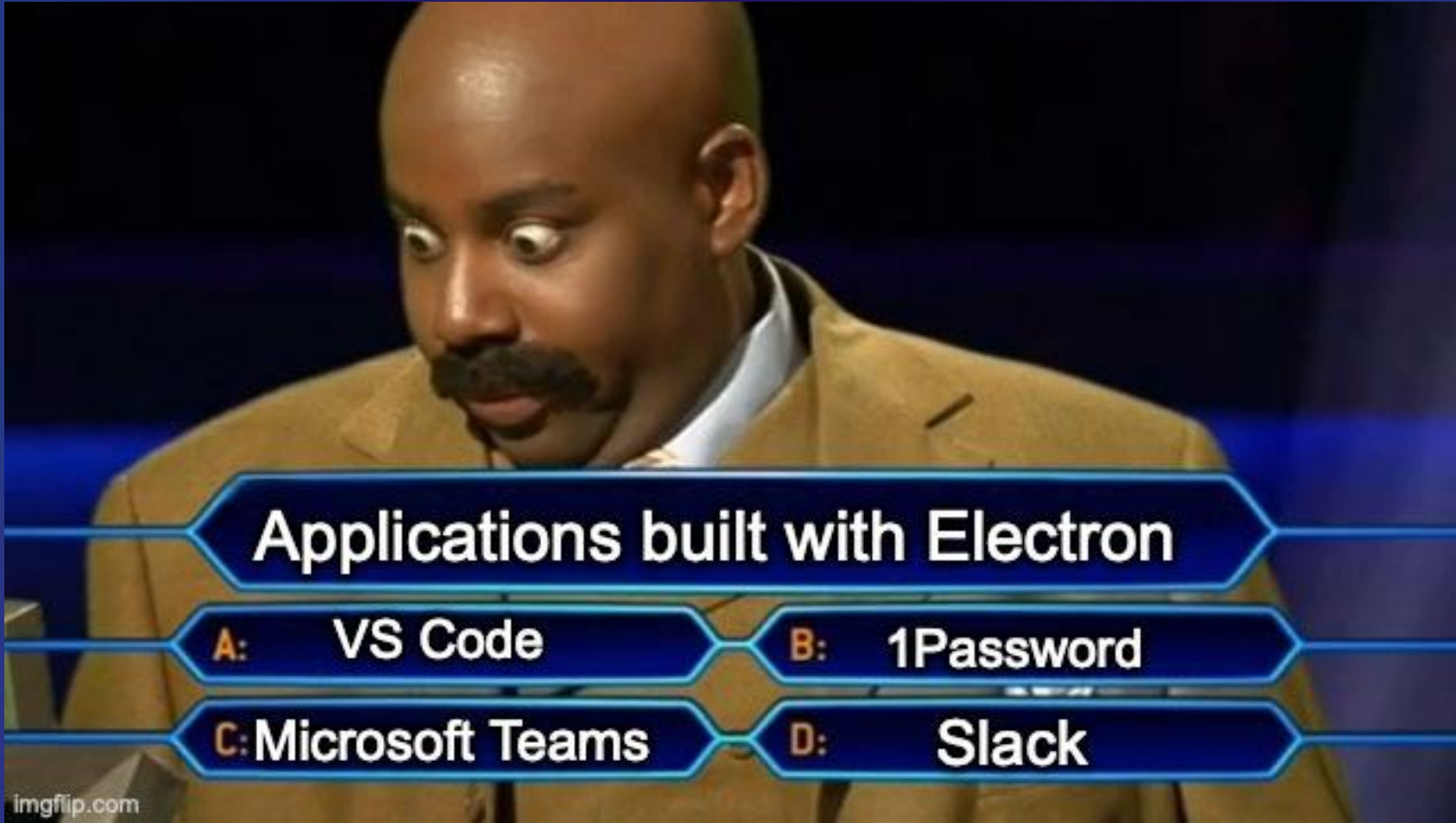


▼ 98 Maintainers

-  alexaltea
-  alexei
-  anatrajkovska
-  andybitz
-  arunoda
-  arzafran
-  atcastle
-  b3nnyl
-  bjy
-  brianloveswords
-  caarlos0
-  codetheory
-  coetry
-  dav-is
-  dominictarr
-  dotkuro
-  electron-cfa
-  electron-nightly
-  electronhq
-  ethan_arrowood
-  fivepointseven
-  fritzy (2)
-  gajus (2)
-  gar (2)
-  goloroden
-  guybedford
-  hharnisc
-  huvik
-  iamevilrabbit
-  igorklopov
-  iijk
-  iliakan
-  isaacs (6)
-  janicklas-ralph
-  jaredwray (2)
-  javivelasco
-  joecohens
-  jprichardson (2)
-  juancampa
-  kevva
-  kornel
-  leo
-  lfades
-  ljharb (11)
-  lucleray
-  lukechilds (4)
-  lukekarrys (2)
-  mafintosh (2)
-  malept (2)
-  manidlou
-  manovotny
-  marcosnils
-  matheuss
-  matteo.collina
-  maxogden
-  mfix22
-  mglagola
-  moll
-  msweeneydev
-  nhummel
-  nkzawa
-  nlf (2)
-  npm-cli-ops (2)
-  olliv
-  paco
-  paulogdm
-  prezjordan
-  qix
-  quietshu
-  rabaut
-  radubrehar
-  ragojose
-  rauchg
-  raynos
-  ryanzim (3)
-  saquibkhan (2)
-  sarupbanskota
-  sindresorhus (19)
-  skillcrn
-  sophearak
-  styfle
-  substack
-  superjoe (3)
-  szmarczak (7)
-  thebigredgeek (2)
-  thejameskyle
-  thejoshwolfe (2)
-  thenativeweb-ad...
-  timer
-  timneutkens
-  tjholowaychuk (2)
-  tootallnate (2)
-  types (7)
-  umegaya
-  williamli
-  yeldir
-  zeit-bot
-  zkat



iamevilrabbit



Applications built with Electron

A: VS Code

B: 1Password

C: Microsoft Teams

D: Slack

imgflip.com

Packages



RE USE
CYCLE
PEAT

What you should do

- © Use strict versions
- © Know your dependencies
- © Don't use packages when possible
- © Use tooling for scanning
- © Update regularly

Educate
your
developers!



Who should be
responsible for security?



Thank you!

Sources



- [Malicious NuGet Packages Stole ASP.NET Data](#)
- [Axios Supply Chain Attack Pushes Cross-Platform RAT](#)
- [Supply Chain Worms in 2026](#)
- [Researchers Uncover 454,000+ Malicious Packages](#)
- [State of the software supply chain](#)
- [Images](#)
- [OWASP Top 10](#)
- [NPM Graph](#)