Electron Security

Making your Mac a worse place?

Mykola Grymalyuk - June 21st, 2024

MacDevOpsYVR 2023

$ '/usr/bin/whoami'

```
> "Mykola Grymalyuk"
```

- Security and Development Technician at RIPEDA Consulting.
- Project lead of OpenCore Legacy Patcher.
- Breaks macOS internals on my blog, khronokernel.com.

What is Electron?

What is Electron?

What is Electron?

1. Chromium

What is Electron?

1. Chromium

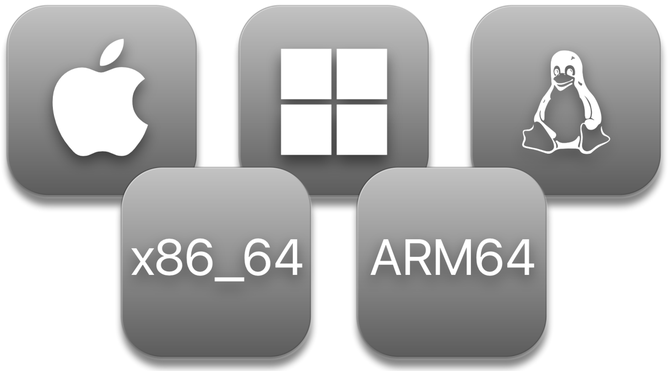
2. Node.JS

What is Electron?

1. Chromium

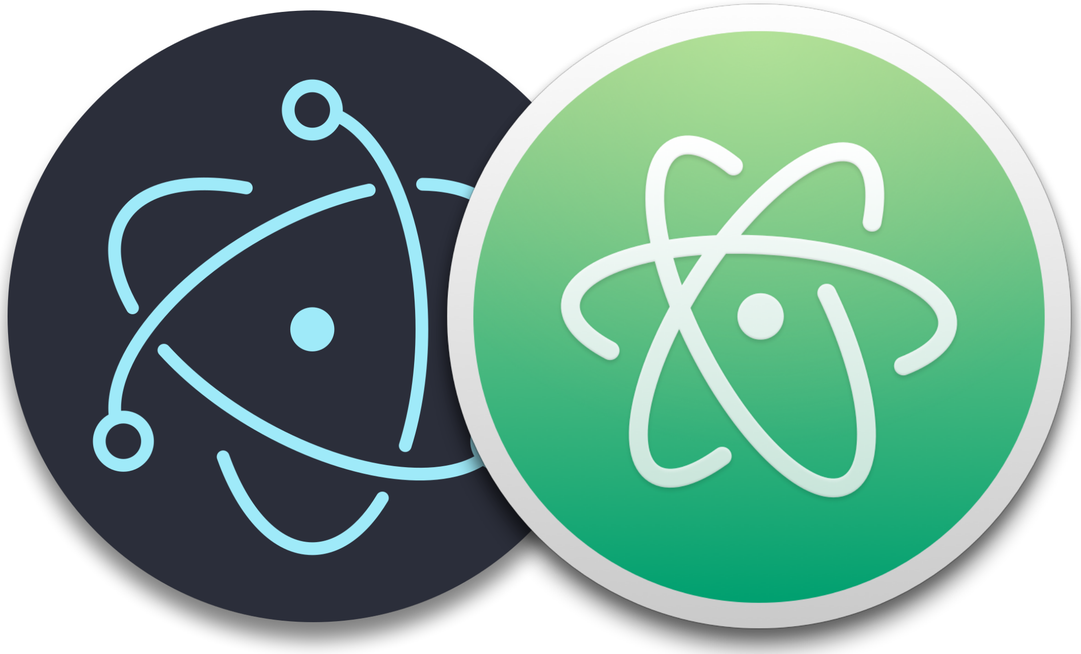
2. Node.JS

3. Multi-Platform



Electron

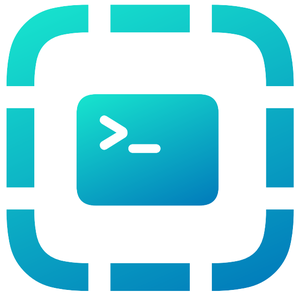
- Initial release in 2013.
- Designed originally for Atom.
- Other frameworks also exist, like nwjs (formerly node-webkit)
- Many applications use Electron:
  - Slack
  - Discord
  - Visual Studio Code
  - 1Password
  - OpenVPN



Electron Structure

Electron Structure

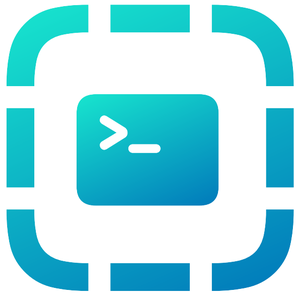
Electron Structure



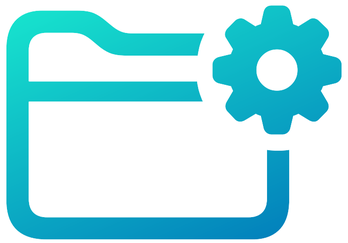
Entry Point

`Vendor.app/Contents/MacOS/Vendor`

Electron Structure



Entry Point

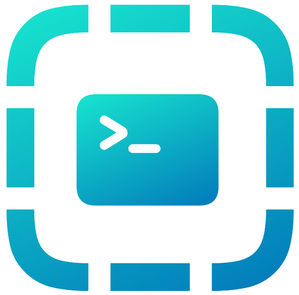


Electron Framework

`Vendor.app/Contents/MacOS/Vendor`

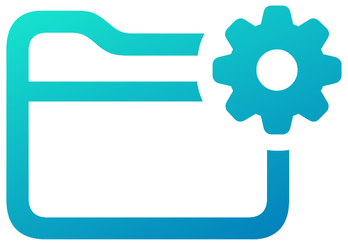
`./Contents/Frameworks/Electron Framework.framework`

Electron Structure



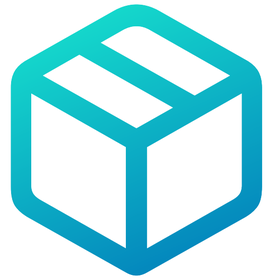
Entry Point

`Vendor.app/Contents/MacOS/Vendor`



Electron Framework

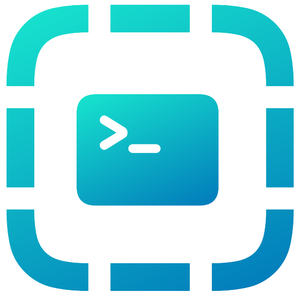
`./Contents/Frameworks/Electron Framework.framework`



ASAR

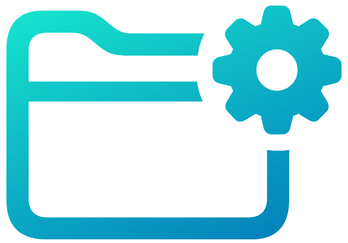
`./Contents/Resources/app.asar`

Electron Structure



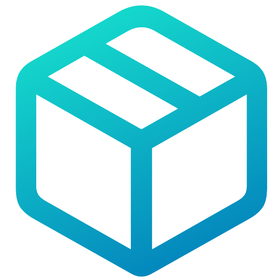
Entry Point

`Vendor.app/Contents/MacOS/Vendor`



Electron Framework

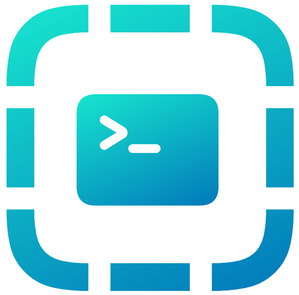
`./Contents/Frameworks/Electron Framework.framework`



ASAR

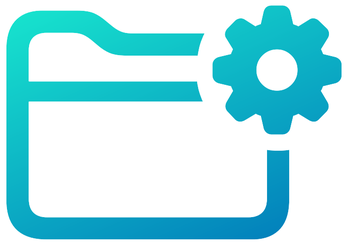
`./Contents/Resources/app.asar`

Electron Structure



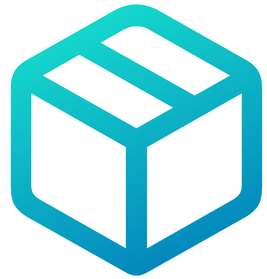
Entry Point

`Vendor.app/Contents/MacOS/Vendor`



Electron Framework

`./Contents/Frameworks/Electron Framework.framework`



ASAR

`./Contents/Resources/app.asar`

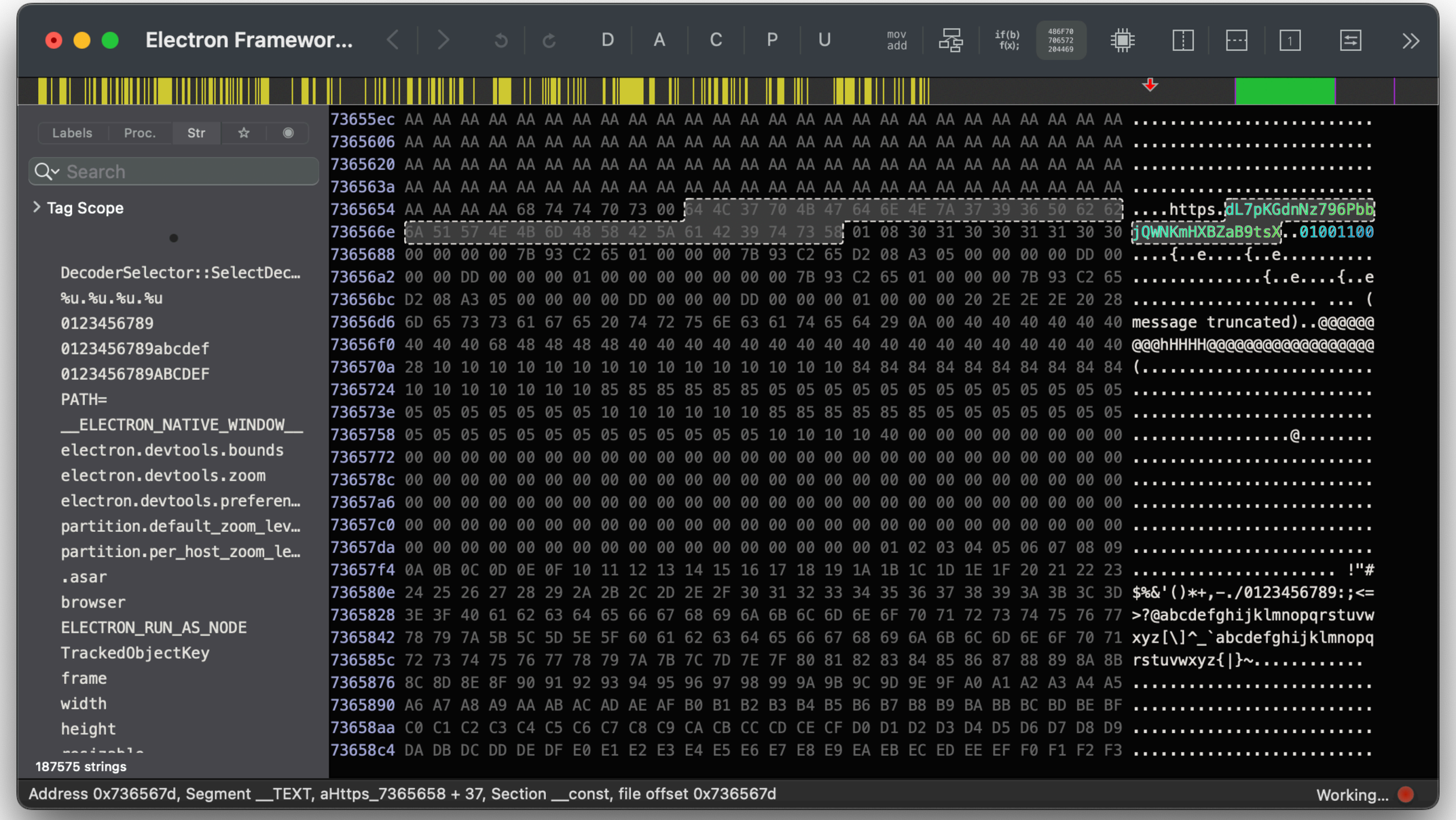
Electron Fuses

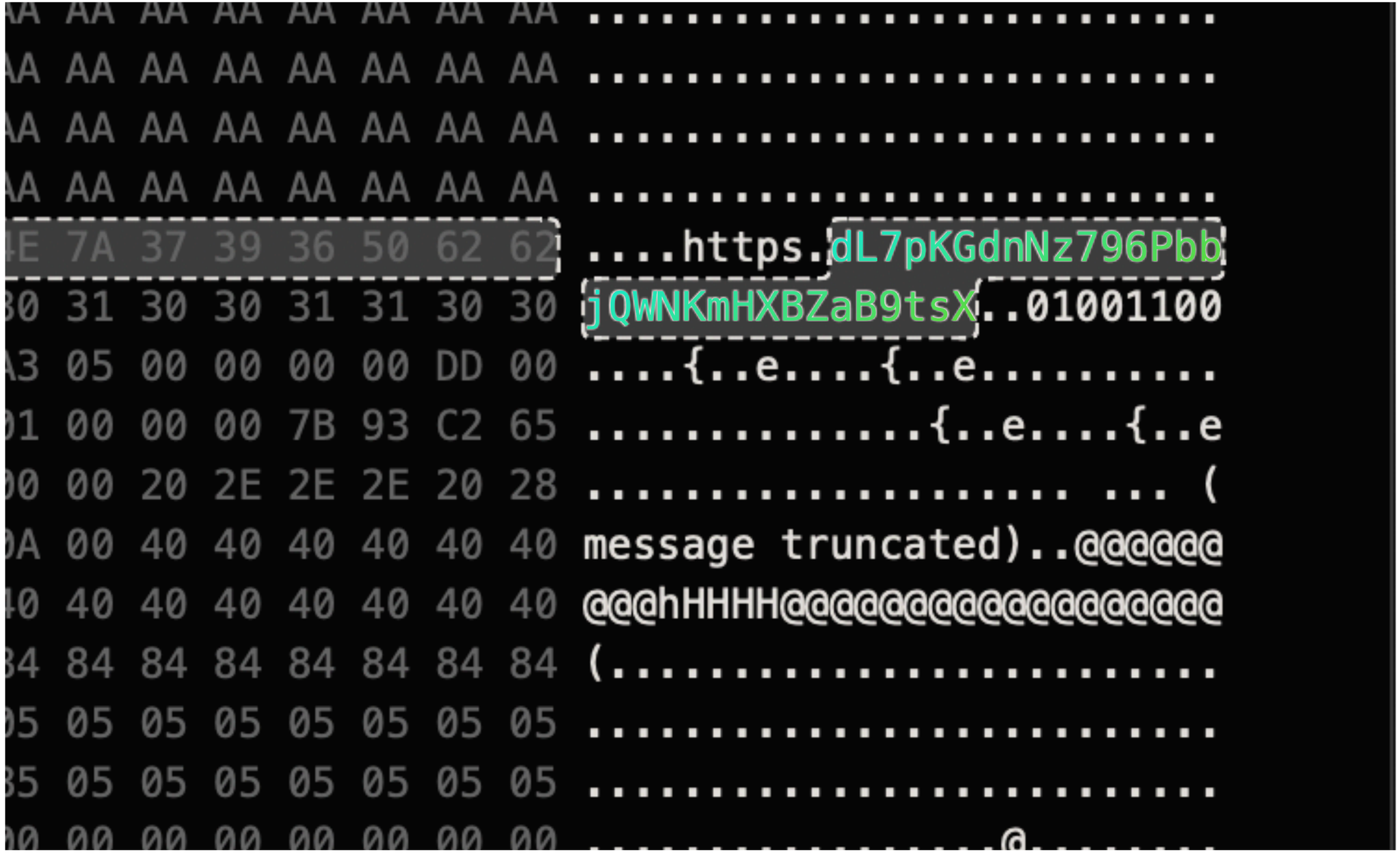
Electron Fuses

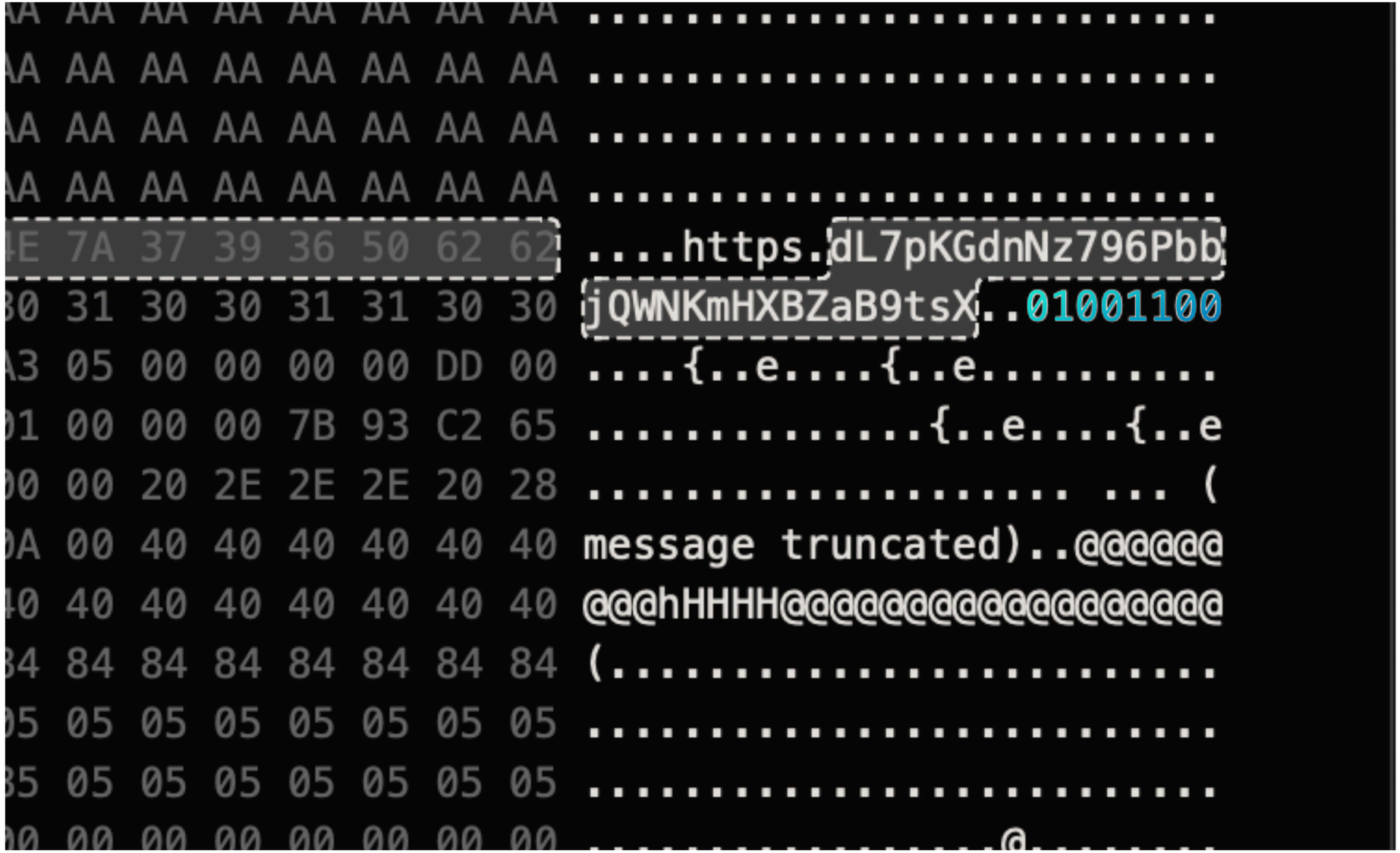
Debug options!

- Embedded inside Electron.framework.
- Introduced with Electron v12.0.0 in 2021.
- Found after a “`sentinel`”:
  - `dL7pKGdnNz796PbbjQWNKmHXBZaB9tsX`
- 8 fuses currently implemented as of fuses v1.8.0.
- On/Off switches.
  - Similar to macOS’ System Integrity Protection (SIP)

```
5   /**
6    * Maps config keys to their index in the fuse wire
7    */
8   export enum FuseV1Options {
9     RunAsNode = 0,
10    EnableCookieEncryption = 1,
11    EnableNodeOptionsEnvironmentVariable = 2,
12    EnableNodeCliInspectArguments = 3,
13    EnableEmbeddedAsarIntegrityValidation = 4,
14    OnlyLoadAppFromAsar = 5,
15    LoadBrowserProcessSpecificV8Snapshot = 6,
16    GrantFileProtocolExtraPrivileges = 7,
17  }
18
```


Electron Framewor...
D
A
C
P
U
mov
add
if(b)
f(x);
Labels
Proc.
Str
Search
Tag Scope
DecoderSelector::SelectDec…
%u.%u.%u.%u
0123456789
0123456789abcdef
0123456789ABCDEF
PATH=
__ELECTRON_NATIVE_WINDOW__
electron.devtools.bounds
electron.devtools.zoom
electron.devtools.preferen…
partition.default_zoom_lev…
partition.per_host_zoom_le…
.asar
browser
ELECTRON_RUN_AS_NODE
TrackedObjectKey
frame
width
height
187575 strings
73655ec AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA ....
7365606 AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA ....
7365620 AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA ....
736563a AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA ....
7365654 AA AA AA AA 68 74 74 70 73 00 64 4C 37 70 4B 47 64 6E 4E 7A 37 39 36 50 62 62 ....https.dL7pKGdnNz796Pbb
736566e 6A 51 57 4E 4B 6D 48 58 42 5A 61 42 39 74 73 58 01 08 30 31 30 30 31 31 30 30 jQWNKmHXBZaB9tsX..01001100
7365688 00 00 00 00 7B 93 C2 65 01 00 00 00 7B 93 C2 65 D2 08 A3 05 00 00 00 00 DD 00 ....{..e....{..e..........
73656a2 00 00 DD 00 00 00 01 00 00 00 00 00 00 00 7B 93 C2 65 01 00 00 00 7B 93 C2 65 ....{..e....{..e
73656bc D2 08 A3 05 00 00 00 00 DD 00 00 00 DD 00 00 00 01 00 00 00 20 2E 2E 2E 20 28 .... (
73656d6 6D 65 73 73 61 67 65 20 74 72 75 6E 63 61 74 65 64 29 0A 00 40 40 40 40 40 40 message truncated)..@@@@@@
73656f0 40 40 40 68 48 48 48 48 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 @@@hHHHH@@@@@@@@@@@@@@@@@@@@
736570a 28 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 84 84 84 84 84 84 84 84 84 84 (..........................
7365724 10 10 10 10 10 10 10 85 85 85 85 85 85 05 05 05 05 05 05 05 05 05 05 05 05 05 ....
736573e 05 05 05 05 05 05 05 10 10 10 10 10 10 85 85 85 85 85 85 05 05 05 05 05 05 05 ....
7365758 05 05 05 05 05 05 05 05 05 05 05 05 05 10 10 10 10 40 00 00 00 00 00 00 00 00 ....@........
7365772 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
736578c 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
73657a6 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
73657c0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
73657da 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 02 03 04 05 06 07 08 09 ....
73657f4 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F 20 21 22 23 .... !"#
736580e 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F 30 31 32 33 34 35 36 37 38 39 3A 3B 3C 3D $%&'()*+,-./0123456789:;<=
7365828 3E 3F 40 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 >?@abcdefghijklmnopqrstuvw
7365842 78 79 7A 5B 5C 5D 5E 5F 60 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 xyz[\]^_`abcdefghijklmnopq
736585c 72 73 74 75 76 77 78 79 7A 7B 7C 7D 7E 7F 80 81 82 83 84 85 86 87 88 89 8A 8B rstuvwxyz{|}~............
7365876 8C 8D 8E 8F 90 91 92 93 94 95 96 97 98 99 9A 9B 9C 9D 9E 9F A0 A1 A2 A3 A4 A5 ....
7365890 A6 A7 A8 A9 AA AB AC AD AE AF B0 B1 B2 B3 B4 B5 B6 B7 B8 B9 BA BB BC BD BE BF ....
73658aa C0 C1 C2 C3 C4 C5 C6 C7 C8 C9 CA CB CC CD CE CF D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 ....
73658c4 DA DB DC DD DE DF E0 E1 E2 E3 E4 E5 E6 E7 E8 E9 EA EB EC ED EE EF F0 F1 F2 F3 ....
Address 0x736567d, Segment __TEXT, aHttps_7365658 + 37, Section __const, file offset 0x736567d
Working...


AA AA AA AA AA AA AA AA ....
AA AA AA AA AA AA AA AA ....
AA AA AA AA AA AA AA AA ....
AA AA AA AA AA AA AA AA ....
4E 7A 37 39 36 50 62 62 ....https.dL7pKGdnNz796Pbb
30 31 30 30 31 31 30 30 jQWNKmHXBZaB9tsX..01001100
A3 05 00 00 00 00 DD 00 ....{..e....{..e..........
01 00 00 00 7B 93 C2 65 ....{..e....{..e
00 00 20 2E 2E 2E 20 28 .... (
0A 00 40 40 40 40 40 40 message truncated)..@@@@@@
40 40 40 40 40 40 40 40 @@@hHHHH@@@@@@@@@@@@@@@@
84 84 84 84 84 84 84 84 (.........................
05 05 05 05 05 05 05 05 ....
85 05 05 05 05 05 05 05 ....
00 00 00 00 00 00 00 00 ....@.....


AA AA AA AA AA AA AA AA ....
AA AA AA AA AA AA AA AA ....
AA AA AA AA AA AA AA AA ....
AA AA AA AA AA AA AA AA ....
4E 7A 37 39 36 50 62 62 ....https.dL7pKGdnNz796Pbb
30 31 30 30 31 31 30 30 jQWNKmHXBZaB9tsX..01001100
A3 05 00 00 00 00 DD 00 ....{..e....{..e........
01 00 00 00 7B 93 C2 65 ....{..e....{..e
00 00 20 2E 2E 2E 20 28 .... (
0A 00 40 40 40 40 40 40 message truncated)..@@@@@@
40 40 40 40 40 40 40 40 @@@hHHHH@@@@@@@@@@@@@@@@
84 84 84 84 84 84 84 84 (.......................
05 05 05 05 05 05 05 05 ....
35 05 05 05 05 05 05 05 ....
00 00 00 00 00 00 00 00 ....@.....

Slack's Electron Fuses

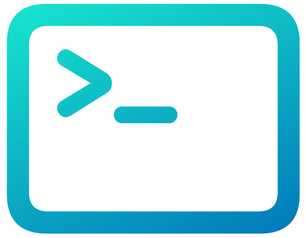
```
0 → 0x0 RunAsNode
1 → 0x1 EnableCookieEncryption
0 → 0x2 EnableNodeOptionsEnvironmentVariable
0 → 0x3 EnableNodeCliInspectArguments
1 → 0x4 EnableEmbeddedAsarIntegrityValidation
1 → 0x5 OnlyLoadAppFromAsar
0 → 0x6 LoadBrowserProcessSpecificV8Snapshot
0 → 0x7 GrantFileProtocolExtraPrivileges
```

Apple's System Integrity Protection

`0x803 -> 1000 0000 0011 -> 1100 0000 0001`

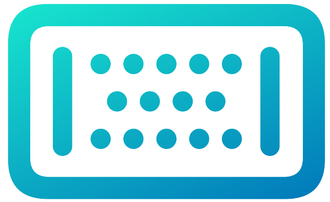
```
1 → 0x1   CSR_ALLOW_UNTRUSTED_KEXTS
1 → 0x2   CSR_ALLOW_UNRESTRICTED_FS
0 → 0x4   CSR_ALLOW_TASK_FOR_PID
0 → 0x8   CSR_ALLOW_KERNEL_DEBUGGER
0 → 0x10  CSR_ALLOW_APPLE_INTERNAL
0 → 0x20  CSR_ALLOW_UNRESTRICTED_DTRACE
0 → 0x40  CSR_ALLOW_UNRESTRICTED_NVRAM
0 → 0x80  CSR_ALLOW_DEVICE_CONFIGURATION
0 → 0x100 CSR_ALLOW_ANY_RECOVERY_OS
0 → 0x200 CSR_ALLOW_UNAPPROVED_KEXTS
0 → 0x400 CSR_ALLOW_EXECUTABLE_POLICY_OVERRIDE
1 → 0x800 CSR_ALLOW_UNAUTHENTICATED_ROOT
```

The fun electron fuses



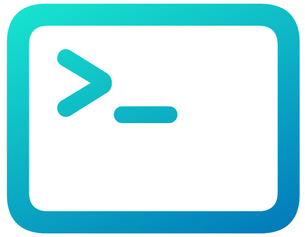
RunAsNode

- Introduced in Electron v0.35.2
- Environment Variable:
  - `RUN_AS_NODE`
- Free node.js shell.



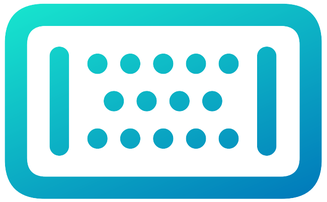
EnableNodeCliInspectArguments

- Introduced with Electron v2.0.0
- Argument:
  - `--inspect={port}`
- WebSocket for communication.



RunAsNode

```
$ RUN_AS_NODE=1 Vendor.app/Contents/MacOS/Vendor hello.js
```



EnableNodeCliInspectArguments

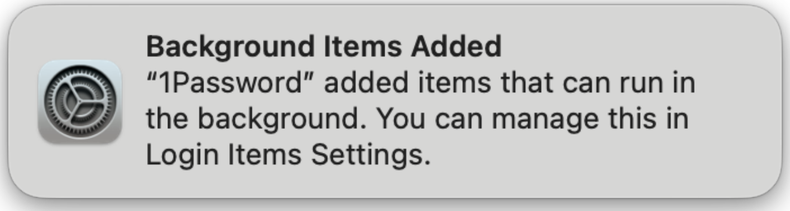
```
$ Vendor.app/Contents/MacOS/Vendor --inspect=4096

> http://127.0.0.1:4096/json/ -> webSocketDebuggerUrl

> ws://127.0.0.1:26711/e580f9b3-0905-4c49-8a37-e3d17ea04410:
{“params”: {
  “expression”: “eval(‘hello world’)”
  }}
```

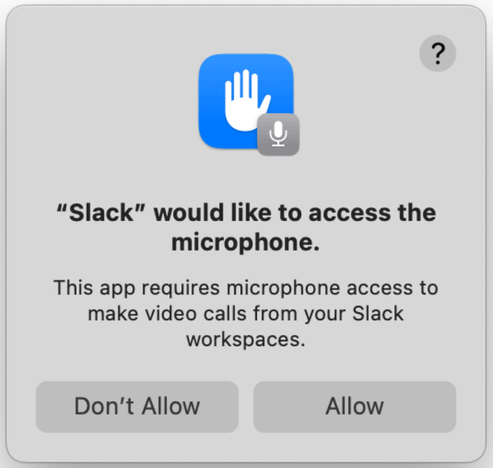
What does this mean for us?

Arbitrary code execution 🎉

Arbitrary Code Execution

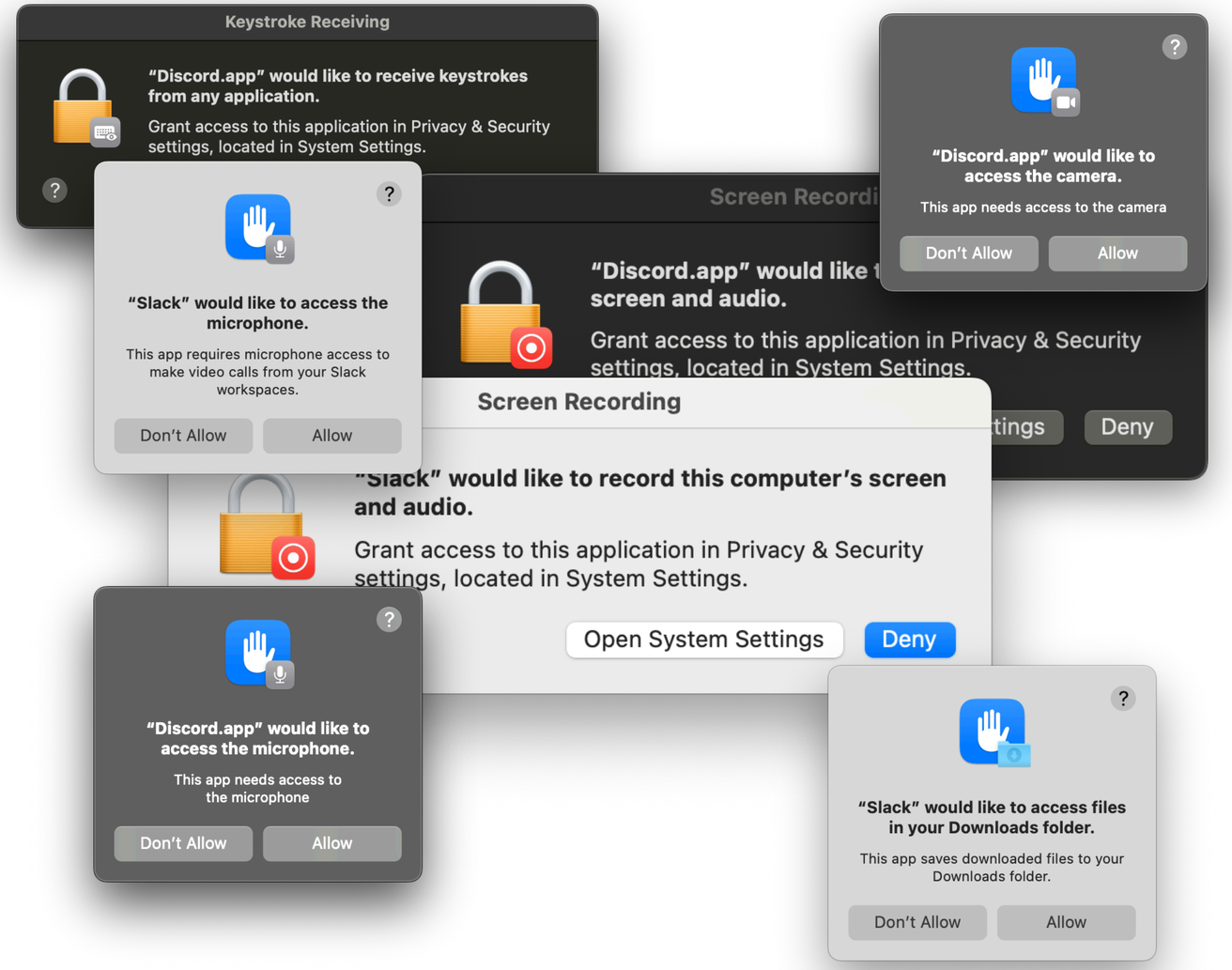
- Ability to run random code of your choosing.
- Code Signature Inheritance.
  - Launch Service Prompts.
  - TCC Inheritance.





TCC

- Transparency, Consent, and Control.
- Yeah those annoying prompts.
- Inheritance for child processes.
- Important to Apple.



Attack Scenario #1 in Enterprise

Malicious software in Enterprise

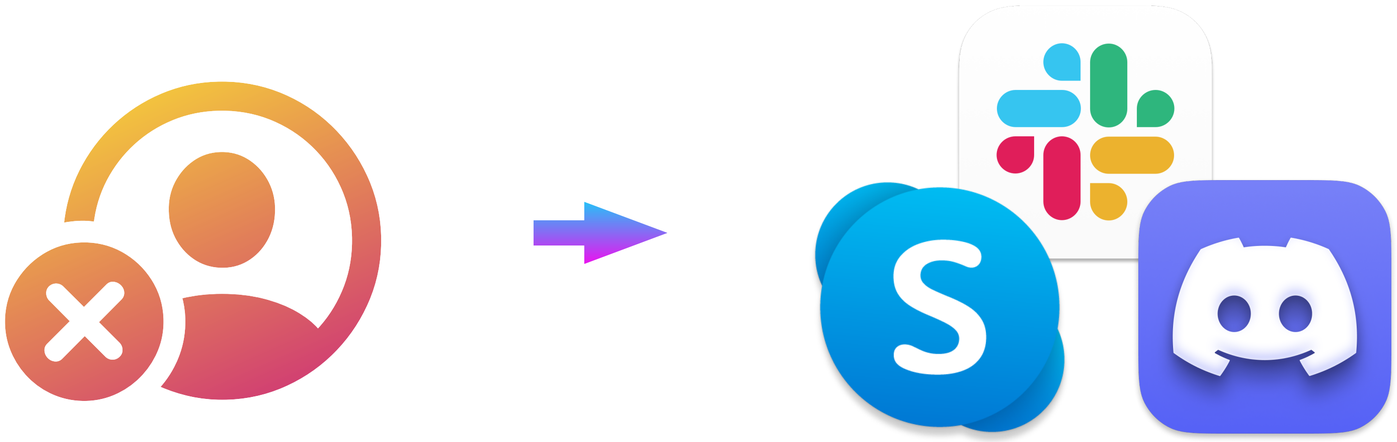
Malicious software in Enterprise

Malicious software in Enterprise

Attack Scenario #2 in Enterprise

Malicious user in Enterprise

Malicious user in Enterprise



Malicious user in Enterprise

Malicious user in Enterprise

com.apple.TCC.configuration-profile-policy

Services

ScreenCapture

AllowStandardUserToSetSystemService

So how do you find misconfigured apps?

Python to the rescue 🐍

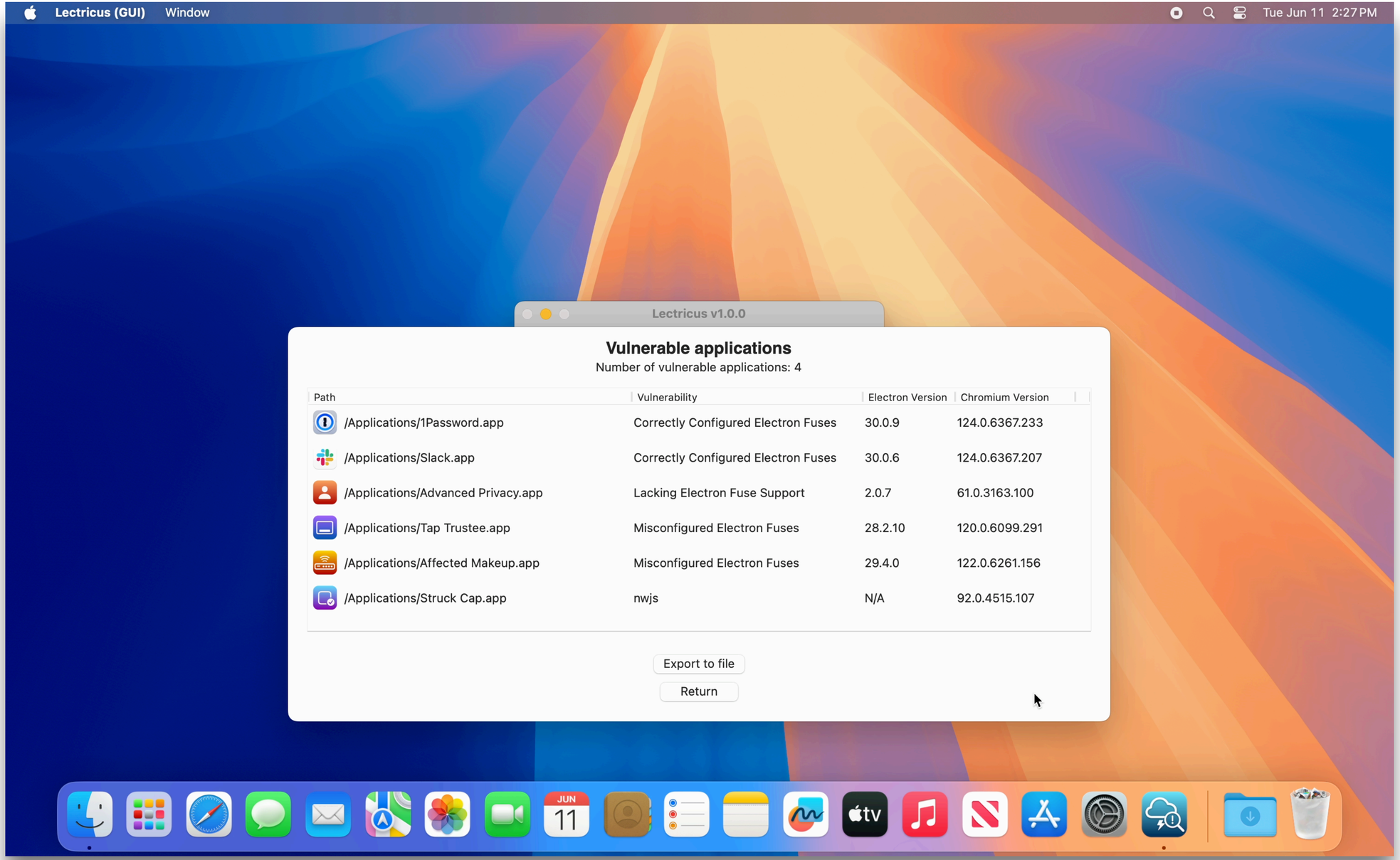
Lectricus

- Python-based library for finding vulnerable electron apps.
- Multi-platform.
- Available as standalone CLI or GUI.
  - No dependancy hell.
- Supports PLIST, XML, JSON and CSV exports.
- Open source on GitHub.

macOS Demo


Finder File Edit View Go Window Help
Tue Jun 11 2:25 PM
JUN
11
tv

macOS Demo

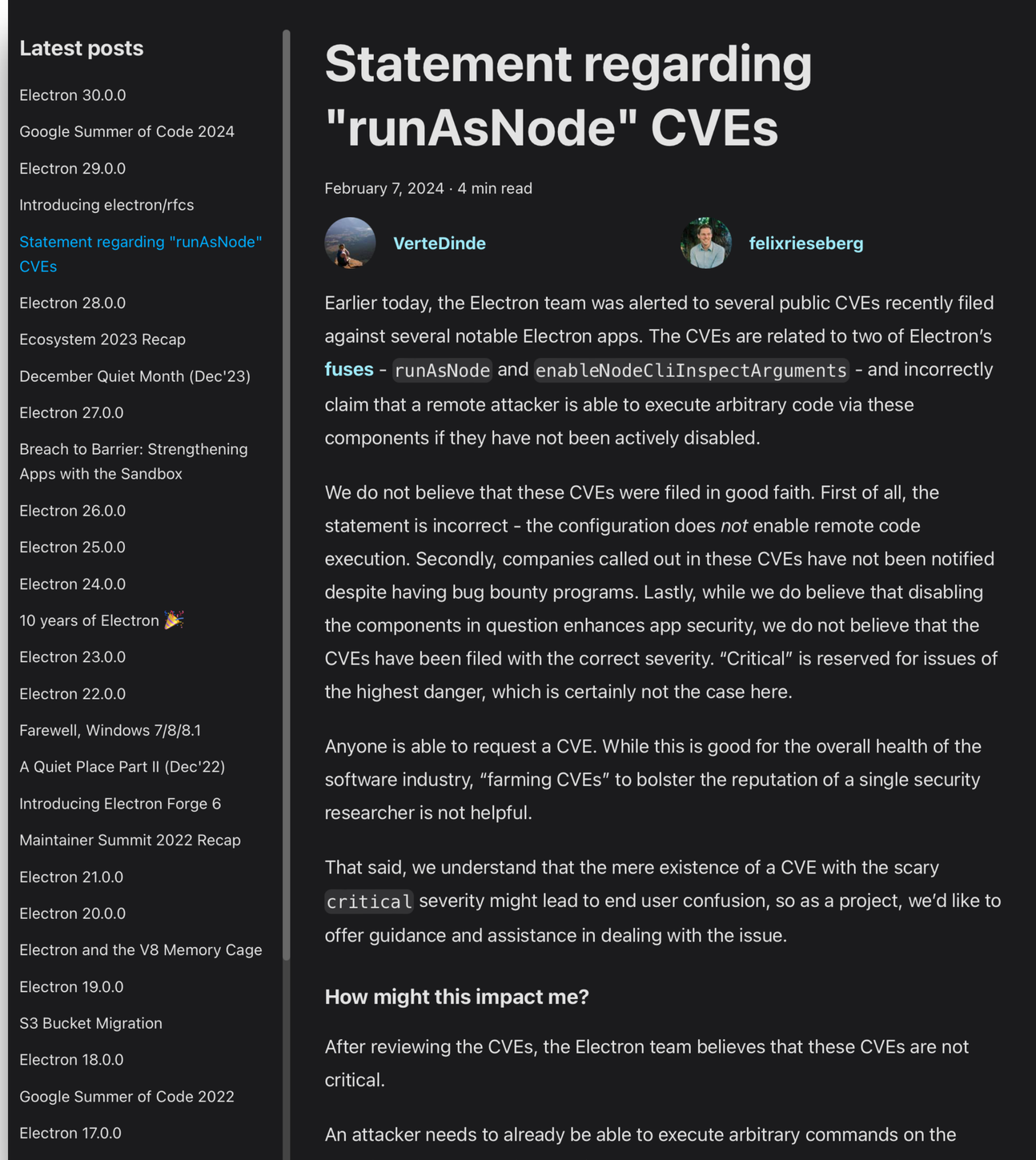

Lectricus (GUI)
Window
Tue Jun 11 2:27 PM
Lectricus v1.0.0
Vulnerable applications
Number of vulnerable applications: 4
Path
Vulnerability
Electron Version
Chromium Version
/Applications/1Password.app
Correctly Configured Electron Fuses
30.0.9
124.0.6367.233
/Applications/Slack.app
Correctly Configured Electron Fuses
30.0.6
124.0.6367.207
/Applications/Advanced Privacy.app
Lacking Electron Fuse Support
2.0.7
61.0.3163.100
/Applications/Tap Trustee.app
Misconfigured Electron Fuses
28.2.10
120.0.6099.291
/Applications/Affected Makeup.app
Misconfigured Electron Fuses
29.4.0
122.0.6261.156
/Applications/Struck Cap.app
nwjs
N/A
92.0.4515.107
Export to file
Return

What's Electron's response to this?

Statement regarding "runAsNode" CVEs

Posted February 7th, 2024

- In response to CVEs filed in bad faith.
- Is a valid vulnerability.
- Chrome Security Model.
- No TCC bypasses mentioned.



Latest posts

Electron 30.0.0
Google Summer of Code 2024
Electron 29.0.0
Introducing electron/rfcs
Statement regarding "runAsNode" CVEs
Electron 28.0.0
Ecosystem 2023 Recap
December Quiet Month (Dec'23)
Electron 27.0.0
Breach to Barrier: Strengthening Apps with the Sandbox
Electron 26.0.0
Electron 25.0.0
Electron 24.0.0
10 years of Electron 🎉
Electron 23.0.0
Electron 22.0.0
Farewell, Windows 7/8/8.1
A Quiet Place Part II (Dec'22)
Introducing Electron Forge 6
Maintainer Summit 2022 Recap
Electron 21.0.0
Electron 20.0.0
Electron and the V8 Memory Cage
Electron 19.0.0
S3 Bucket Migration
Electron 18.0.0
Google Summer of Code 2022
Electron 17.0.0

Statement regarding "runAsNode" CVEs

February 7, 2024 · 4 min read

VerteDinde felixrieseberg

Earlier today, the Electron team was alerted to several public CVEs recently filed against several notable Electron apps. The CVEs are related to two of Electron's **fuses** - `runAsNode` and `enableNodeCliInspectArguments` - and incorrectly claim that a remote attacker is able to execute arbitrary code via these components if they have not been actively disabled.

We do not believe that these CVEs were filed in good faith. First of all, the statement is incorrect - the configuration does *not* enable remote code execution. Secondly, companies called out in these CVEs have not been notified despite having bug bounty programs. Lastly, while we do believe that disabling the components in question enhances app security, we do not believe that the CVEs have been filed with the correct severity. "Critical" is reserved for issues of the highest danger, which is certainly not the case here.

Anyone is able to request a CVE. While this is good for the overall health of the software industry, "farming CVEs" to bolster the reputation of a single security researcher is not helpful.

That said, we understand that the mere existence of a CVE with the scary `critical` severity might lead to end user confusion, so as a project, we'd like to offer guidance and assistance in dealing with the issue.

How might this impact me?

After reviewing the CVEs, the Electron team believes that these CVEs are not critical.

An attacker needs to already be able to execute arbitrary commands on the

What’s the end goal with Lectricus?

What's the end goal with Lectricus?

Educate

Take action

Ways to take action

- Developers:
  - Review fuses in your electron Apps.
  - Not just the two talked about, many more dangerous fuses.
- Users:
  - Find and report misconfigured fuses in app you use.
- Admins:
  - Same as users.
  - Remove TCC permissions from apps that are vulnerable.
  - `$ sudo tccutil reset All com.bad.app`

Thanks for listening to my rambles!

Mirrored on khronokernel.com

Links and shoutouts!

- Lectricus:
  - https://github.com/ripeda/lectricus
- TsunekOh's CVE-2023-32546:
  - What kicked off this idea for Electron querying!
  - Chatwork Desktop Application.
  - Part of their SIP bypass talk at CODE BLUE 2023.
- Wojciech Reguła's electroniz3r:
  - Didn't know it existed when I started Lectricus...
  - But made me go further with Lectricus!
  - https://github.com/r3ggi/electroniz3r
- Electron:
  - https://www.electronjs.org/docs/latest/tutorial/fuses
  - https://www.electronjs.org/blog/statement-run-as-node-cves