CAA-2026-0016 Threat Actor Publishes Malicious npm Packages Installing Dropper and RAT

This report is distributed as **TLP:CLEAR**. Recipients may share this information without restriction. Information is subject to standard copyright rules.

Disclaimer | CyberAlberta

Summary

On 30 March 2026, threat actors published two malicious npm packages named plain-crypto-js as well as two malicious axios npm packages to install an obfuscated dropper on victim devices. Axios is a popular HTTP client library, with over 100 million weekly downloads.

Details

Public reporting identified the exposure window for organizations was limited to 2 hours 54 minutes for `axios@1.14.1` and 2 hours and 15 minutes for `axios@0.30.4` on 31 March 2026.[1][2]

- 2026-03-30 05:57 (UTC): The decoy `plain-crypto-js@4.2.0` was published to npm.
- 2026-03-30 23:59 (UTC): The malicious `plain-crypto-js@4.2.1` was published containing an obfuscated dropper to npm.
- 2026-03-31 00:21 (UTC): The `axios@1.14.1` npm package was compromised to include `plain-crypto-js` version 4.2.1 as a runtime dependency.
- 2026-03-31 01:00 (UTC): The `axios@0.30.4` npm package was also compromised to include `plain-crypto-js@4.2.1` as a runtime dependency.
- 2026-03-31 03:15 (UTC): npm unpublishes the compromised `axios` packages.

The malicious packages include an obfuscated setup file (named `setup.js`) that retrieves a platform-specific second stage payload from `sfrclak[.]com`. The second stage payload is a minimal remote access trojan (RAT) capable of receiving and executing arbitrary commands. Wiz security researchers reported that all observed second stages have similar capabilities, despite their different implementations:

- **Linux Payload:** Delivered as a Python script named `ld.py`
- **MacOS Payload:** Delivered as a compiled Mach-O universal binary named `com.apple.act.mond`
- **Windows Payload:** Delivered as a PowerShell script named `6202033.ps1`

Analysts at socket.dev identified two additional packages distributing the same malware:[3]

- `@shadanai/openclaw (v2026.3.28-2, 2026.3.28-3, 2026.3.31-1, 2026.3.31-2)`
- `@qqbrowser/openclaw-qbot(v0.0.130)`

Recommendations

CyberAlberta Threat Intelligence recommends that organizations do the following:

[1] https://www.wiz.io/blog/axios-npm-compromised-in-supply-chain-attack
[2] https://www.stepsecurity.io/blog/axios-compromised-on-npm-malicious-versions-drop-remote-access-trojan
[3] https://socket.dev/blog/axios-npm-package-compromised

- Confirm the presence of any of the included IOCs and any of following npm packages:

```
- plain-crypto-js@4.2.1
- axios@0.30.4
- axios@1.14.1
- shadanai/openclaw@2026.3.28-2
- shadanai/openclaw@2026.3.28-3
- shadanai/openclaw@2026.3.31-1
- shadanai/openclaw@2026.3.31-2
- qqbrowser/openclaw-qbot@0.0.130
```

- If compromise is detected, remediate the impacted devices and rotate **all** credentials and secrets accessible by the infected system.
- Developers should consider pinning packages to known-good versions given the increasing prevalence of package compromises.

Detection Opportunities

The following KQL query can help detect usage of the compromised packages during the exposed timeline. These queries are dependent on the products being stored in their default folder paths.

```
let malicious_npm_packages = dynamic(['axios', 'plain-crypto-js', 'shadanai/openclaw',
'qqbrowser/openclaw-qbot']);
union isfuzzy=true DeviceEvents, DeviceFileEvents, DeviceNetworkEvents, DeviceProcessEvents
| where Timestamp >= datetime(2026-03-30T05:57:00.0000000Z)
| where
    InitiatingProcessCommandLine matches regex @"[\\/]+node_modules[\\/]+"
    or
    ProcessCommandLine matches regex @"[\\/]+node_modules[\\/]+"
    or
    FolderPath matches regex @"[\\/]+node_modules[\\/]+"
    or
    InitiatingProcessFolderPath matches regex @"[\\/]+node_modules[\\/]+"
| extend npm_Package =
    coalesce
    (
        extract(@"node_modules[\\/]+@?([^\\/]+)", 1, InitiatingProcessCommandLine),
        extract(@"node_modules[\\/]+@?([^\\/]+)", 1, ProcessCommandLine),
        extract(@"node_modules[\\/]+@?([^\\/]+)", 1, FolderPath),
        extract(@"node_modules[\\/]+@?([^\\/]+)", 1, InitiatingProcessFolderPath)
    )
| where npm_Package matches regex @"^[a-z0-9][a-z0-9._-]*$"
| where npm_Package has_any (malicious_npm_packages)
| summarize FirstSeen = min(Timestamp), LastSeen = max(Timestamp) by npm_Package, SHA256,
DeviceName, InitiatingProcessAccountName
| sort by LastSeen desc
```

Figure 1. KQL Query for Identifying Related Packages

Indicators of Compromise (IOCs)

The following IOCs characterize activity described in this report.

Description	Indicator

Payload Download, URL	http[:]//sfrclak[.]com:8000/6202033
Payload Download, IP	142.11.206[.]73
Compromised plain-crypto-js-4.2.1.tgz, SHA256 HASH	58401c195fe0a6204b42f5f90995ece5fab74ce7c69c67a24c61a057325af668
Compromised axios-0.30.4.tgz, SHA256 Hash	59336a964f110c25c112bcc5adca7090296b54ab33fa95c0744b94f8a0d80c0f
Compromised axios-1.14.1.tgz, SHA256 Hash	5bb67e88846096f1f8d42a0f0350c9c46260591567612ff9af46f98d1b7571cd
Dropper (setup.js), SHA256 Hash	e10b1fa84f1d6481625f741b69892780140d4e0e7769e7491e5f4d894c2e0e09
Batch Files (system.bat), SHA256 Hash	f7d335205b8d7b20208fb3ef93ee6dc817905dc3ae0c10a0b164f4e7d07121cd
	e49c2732fb9861548208a78e72996b9c3c470b6b562576924bcc3a9fb75bf9ff
Second Stage Linux (ld.py), SHA256 Hash	fcb81618bb15edfdedfb638b4c08a2af9cac9ecfa551af135a8402bf980375cf
Second Stage MacOS, SHA256 Hash	92ff08773995ebc8d55ec4b8e1a225d0d1e51efa4ef88b8849d0071230c9645a
Second Stage Windows (6202033.ps1), SHA256 Hash	617b67a8e1210e4fc87c92d1d1da45a2f311c08d26e89b12307cf583c900d101

Table 1. Indicators of Compromise