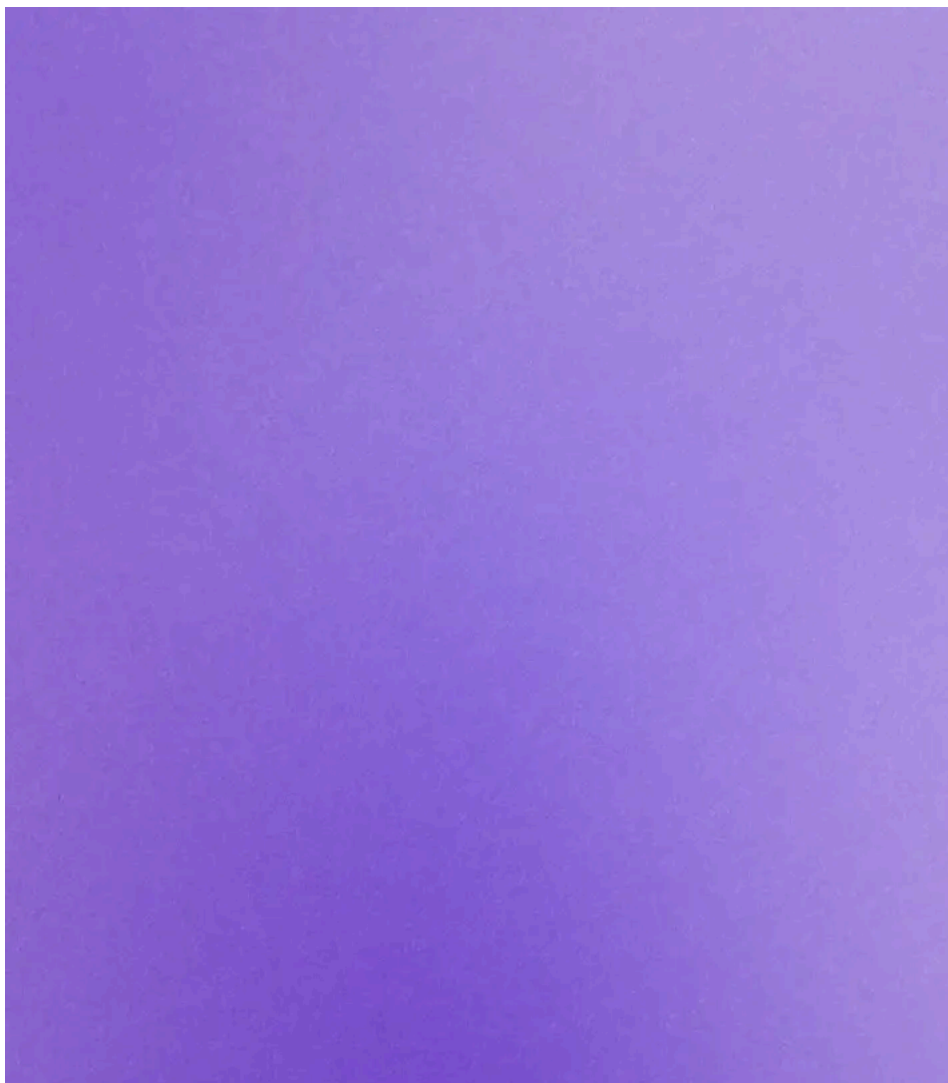


Lazarus Expands Malicious npm Campaign: 11 New Packages Add

...

Archived: 2026-04-05 13:28:02 UTC



Secure your dependencies with us

Socket proactively blocks malicious open source packages in your code.

[Install](#)

North Korean threat actors behind the [Contagious Interview](#) operation have expanded their presence in the npm ecosystem, publishing additional malicious packages that deliver the previously [identified](#) BeaverTail malware and introducing new packages with remote access trojan (RAT) loader functionality. These latest samples employ hexadecimal string encoding to evade automated detection systems and manual code audits, signaling a variation in the threat actors' obfuscation techniques.

The threat group's objectives remain unchanged: to compromise developer systems, steal sensitive credentials or financial assets, and maintain access to compromised environments. The Contagious Interview threat actors continue to create new npm accounts and deploy malicious code across platforms like the npm registry, GitHub, and Bitbucket, demonstrating their persistence and showing no signs of slowing down.

Malicious Campaign Proliferation#

The threat actors broadened their campaign by publishing new malicious npm packages under previously identified aliases – `alex­tucker0519`, `edan0831`, and `hottblaze` – as well as newly created accounts, including `taras_lakhai`, `mvitalii`, `wishorn`, and `crouch626`. Each package posed as a utility for arrays, logging, debugging, or event and API handling. As of this writing, the npm registry has suspended all accounts (and associated packages) besides `taras_lakhai`. We have reported this account and petitioned for its removal, along with all GitHub and Bitbucket repositories and user profiles associated with any of the identified accounts. In total, the 11 additional malicious packages identified in this expanded campaign have been downloaded over 5,600 times.

Before the account `alex­tucker0519` was suspended following the [discovery](#) of the malicious package `array-empty-validator`, it had published an additional malicious package, `empty-array-validator`, which communicated with a separate command and control (C2) server at `144.172.87[.]27` on port `1224`.

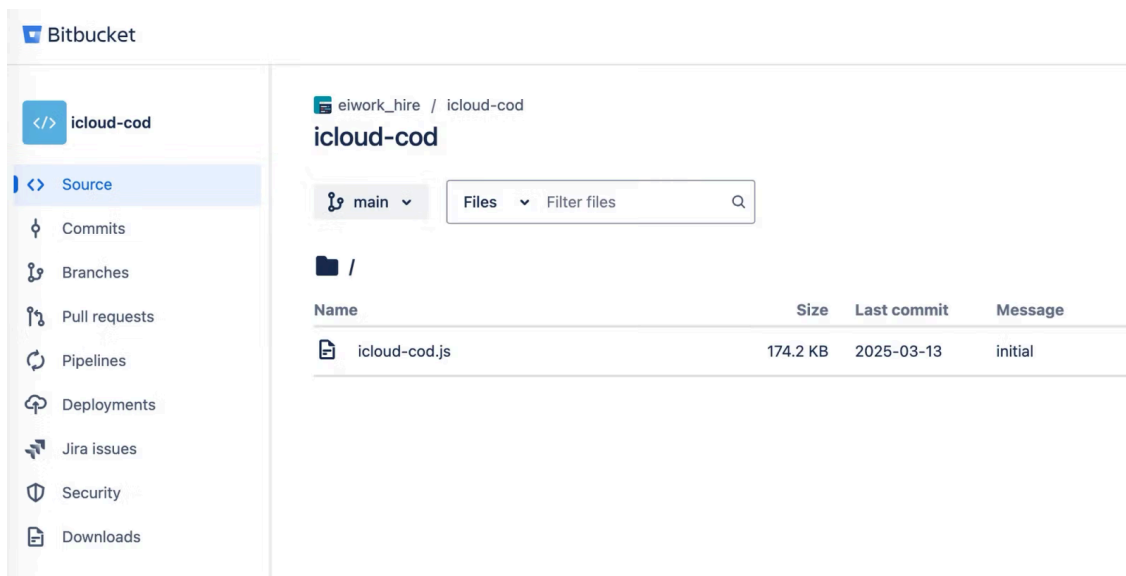
`taras_lakhai` and `mvitalii`, two newly identified accounts, use the same IP and port combination to connect to a C2 server at `45.61.151[.]71` on port `1224`. The shared infrastructure links these two accounts as part of the same threat activity. The `taras_lakhai` account published a malicious package `twitterapis`, which uses this endpoint — matching the infrastructure previously observed in malicious packages from the `mvitalii` account.

SecurityScorecard researchers [identified](#) infrastructure linked to the Lazarus Group during their investigation of the Contagious Interview operation. One of the packages published by `wishorn`, a newly created npm account, uses the same obfuscated C2 IP address — `185.153.182[.]241` on port `1224` — within the `dev-debugger-vite` package. This account also published two additional malicious packages, `snore-log` and `core-pino`, reinforcing its connection to the broader Lazarus-led campaign.

Beyond common infrastructure, the identified packages from newly created accounts also share structural hallmarks with previously attributed [Lazarus operations](#). These packages consistently implement tight loops that scan up to 200 browser profile directories for Brave, Chrome, and Opera, and attempt to extract private keys from Solana's `id.json` file. Exfiltration occurs silently through HTTP POST requests to C2 servers linked to known Lazarus infrastructure. The scripts exhibit hallmark traits of the group's tooling: layered obfuscation, multi-stage payload delivery, and repeated use of BeaverTail — an infostealer that targets browser data, macOS keychain, and cryptocurrency wallets. Several packages also reference InvisibleFerret as a second-stage backdoor, retrieved from the same C2 endpoints used in earlier campaigns.

From GitHub to Bitbucket#

Unlike earlier Lazarus-linked packages that occasionally referenced GitHub repositories, the packages [events-utils](#), [icloud-cod](#) (published by the [mvitalii](#) account), and the previously [identified](#) [react-event-dependency](#) (published by the [elondavid](#) account) were linked to Bitbucket repositories. Based on the observed timelines, the threat actors create these repositories before publishing their corresponding malicious npm packages. The threat actor behind the [alexstucker0519](#) account uploaded the malicious code to their GitHub repository on March 11, 2025, and published the [empty-array-validator](#) npm package containing the same code the following day, on March 12, 2025. This sequencing likely serves to establish a façade of legitimacy – giving unsuspecting developers the impression that the package is actively maintained simply because it links to a live code repository.



Malicious JavaScript file `icloud-cod.js` hosted on Bitbucket

The malicious package `icloud-cod` linked to a Bitbucket repository hosted within a directory named `eiwork_hire` – a detail that may reflect further efforts to legitimize the threat actors’ Contagious Interview operations, which lure unsuspecting developers with fake job offers to gain access to their systems for financial and other illicit objectives.

Expanded Payloads and Obfuscation#

The npm account `crouch626` published four malicious packages: `cln-logger`, `node-clog`, `consolidate-log`, and `consolidate-logger`. The first two packages, `cln-logger` and `node-clog`, featured distinct code structures and employed a different obfuscation technique from the others. In contrast, `consolidate-log` and `consolidate-logger` both communicated with the same C2 server at `144.172.87[.]27` on port `1224`, consistent with earlier findings. This divergence indicates that the threat actors are deploying multiple malware variants within the same broader campaign, potentially to diversify payload delivery and evade detection.

Below are defanged code [snippets](#) from the `cln-logger` package demonstrating the new obfuscation and malicious code with inline comments explaining key functions and objectives.

```
// Decodes obfuscated hex-encoded strings into readable text
// (e.g., function names, URLs)
// Used to hide malicious strings from static analysis tools and manual review
function g(h) {
  return h.replace(/../g, match => String.fromCharCode(parseInt(match, 16)));
}
```

The code defines a helper function `g(h)` that replaces every two hexadecimal characters with their ASCII equivalents. This mechanism conceals key strings such as `require`, `axios`, `get`, and a remote URL, making them less obvious during an inspection.

```
const hl = [
  g('72657175697265'), // "require" - dynamic module import
  g('6178696f73'),    // "axios" - HTTP client
  g('676574'),        // "get" - HTTP GET method
  g('68747470...613662'), // C2 URL - hxxps://mocki[.]io/v1/32f16c80-602a-4c80-80af-32a9b8220a6b
  g('7468656e'),      // "then" - handles async response
];
```

This array decodes critical JavaScript keywords and a remote URL hidden in hex to evade detection. Once resolved, it enables the script to fetch and execute code from a C2 server.

Unlike `cln-logger`, which connects to `mocki[.]io`, the `node-clog` package references `m21gk[.]wiremockapi[.]cloud/g/api/880`, decoded from the obfuscated string `68747470733a2f2f6d3231676b2e776972656d6f636b6170692e636c6f75642f672f6170692f383830`.

This obfuscation tactic is also used by another identified account, `wishorn`, in its `snore-log` and `core-pino` packages. The former references the endpoint `ip-api-server[.]vercel[.]app/api/ipcheck/703`, while the latter uses `ip-check-api[.]vercel[.]app/api/ipcheck/703`. This suggests the threat actors are rotating staging infrastructure or maintaining multiple redundant C2 endpoints while reusing the same loader pattern.

Consistently throughout these identified packages, the malicious code functions as a remote access trojan (RAT) loader, relying on obfuscation and dynamic payload execution to evade detection and deliver second-stage malware. The scripts encode critical strings in hexadecimal and decode them with `String.fromCharCode`, effectively concealing module names and C2 URLs. This obfuscation technique undermines both automated scanners and manual code audits, masking the true functionality and intent of the malware.

Outlook and Recommendations#

The recent expanded malicious campaign tied to the Lazarus Group demonstrates a sustained and adaptable threat to software supply chains. Far from slowing down, the advanced persistent threat (APT) group is diversifying its tactics — publishing new malware under fresh aliases, hosting payloads in both GitHub and Bitbucket repositories, and reusing core components like BeaverTail and InvisibleFerret alongside newly observed RAT/loader variant. These packages exhibit not only reused infrastructure and targeting logic but also redundant

C2 endpoints and varied obfuscation styles, underscoring the threat group’s intent to ensure resilience and evade automated detection and manual audits.

Organizations must assume that targeted infiltration campaigns like Contagious Interview will persist and continue to evolve. Developers — particularly those working in open source, DevOps, and infrastructure engineering roles — remain high-value targets due to their access and trust within broader environments. As such, proactive defense must become foundational to software development practices.

To mitigate these risks, we recommend embedding multiple layers of supply chain security throughout the development lifecycle. This includes automated dependency audits, contextual scanning of third-party packages, and close scrutiny of packages with limited download history or unverifiable maintainers. Monitoring for unusual dependency changes and blocking outbound traffic to known or suspicious C2 endpoints can help contain threats before they escalate.

Socket’s security tooling is purpose-built to address these challenges. The [Socket GitHub App](#) provides real-time scanning of pull requests, flagging suspicious or malicious packages before they are merged. The [Socket CLI](#) tool surfaces red flags during npm installations, helping teams catch dangerous code early. Meanwhile, the [Socket browser extension](#) alerts users to suspicious packages upon download or viewing. Integrating these tools into development pipelines empowers organizations to detect and prevent malware proactively, reducing exposure to Contagious Interview-style supply chain attacks.

Indicators of Compromise (IOCs)#

Malicious npm Packages and Download Count

- empty-array-validator (129)
- twitterapis (102)
- dev-debugger-vite (1,606)
- snore-log (1,904)
- core-pino (483)
- events-utils (133)
- icloud-cod (145)
- cln-logger (308)
- node-clog (213)
- consolidate-log (297)
- consolidate-logger (291)

Threat Actor Identifiers

- **npm Aliases and Email Addresses:**
 - taras_lakhai — kevintracy516@gmail[.]com
 - mvitalii — mvitalii206@gmail[.]com
 - wishorn — starlancer555@gmail[.]com
 - crouch626 — crouchtomy@gmail[.]com

- **GitHub Accounts:**
 - lukobogdan47
 - austin-a3
- **Bitbucket Accounts:**
 - Ezra Walmsley
 - Raymundo Curiel

Malicious GitHub Repositories

- <https://github.com/lukobogdan47/empty-array-validator>
- <https://github.com/austin-a3/twitterapis>

Malicious Bitbucket Repositories

- <https://bitbucket.org/events-utils/launch-events-utils/src/master/>

Command and Control (C2) Endpoints

- 144.172.87[.]27
- 45.61.151[.]71
- 185.153.182[.]241
- mocki[.]io/v1/32f16c80-602a-4c80-80af-32a9b8220a6b
- m21gk[.]wiremockapi[.]cloud/g/api/880
- ip-api-server[.]vercel[.]app/api/ipcheck/703
- ip-check-api[.]vercel[.]app/api/ipcheck/703

MITRE ATT&CK Techniques

- T1195.002 — Supply Chain Compromise: Compromise Software Supply Chain
- T1608.001 — Stage Capabilities: Upload Malware
- T1204.002 — User Execution: Malicious File
- T1059.007 — Command and Scripting Interpreter: JavaScript
- T1027.013 — Obfuscated Files or Information: Encrypted/Encoded File
- T1546.016 — Event Triggered Execution: Installer Packages
- T1005 — Data from Local System
- T1082 — System Information Discovery
- T1083 — File and Directory Discovery
- T1217 — Browser Information Discovery
- T1555.003 — Credentials from Password Stores: Credentials from Web Browsers
- T1555.001 — Credentials from Password Stores: Keychain
- T1041 — Exfiltration Over C2 Channel
- T1105 — Ingress Tool Transfer
- T1119 — Automated Collection
- T1657 — Financial Theft

Source: <https://socket.dev/blog/lazarus-expands-malicious-npm-campaign-11-new-packages-add-malware-loaders-and-bitbucket>