Los investigadores de ciberseguridad revelaron detalles sobre una nueva campaña automatizada denominada Megalodon, la cual introdujo 5,718 commits maliciosos en 5,561 repositorios de GitHub en un periodo de apenas seis horas.

"Mediante cuentas desechables y identidades de autor falsificadas (build-bot, auto-ci, ci-bot, pipeline-bot), el atacante insertó flujos de trabajo de GitHub Actions que contenían cargas útiles bash codificadas en Base64, capaces de extraer secretos de CI, credenciales en la nube, claves SSH, tokens OIDC y secretos del código fuente hacia un servidor C2 alojado en 216.126.225[.]129:8443", indicó SafeDep en un informe.

La lista completa de información recopilada por el malware incluye:

- Variables de entorno de CI, `/proc/*/environ` y entorno del PID 1
- Credenciales de Amazon Web Services (AWS)
- Tokens de acceso de Google Cloud
- Credenciales de roles de instancia obtenidas mediante consultas a AWS IMDSv2, metadatos de Google Cloud y endpoints de Microsoft Azure IMDS
- Claves privadas SSH
- Configuraciones de Docker y Kubernetes
- Tokens de Vault
- Credenciales de Terraform
- Historial de shell
- API keys, cadenas de conexión a bases de datos, JWTs, claves privadas PEM y tokens cloud que coinciden con más de 30 patrones de expresiones regulares de secretos
- URL de solicitud y token OIDC de GitHub Actions
- `GITHUB_TOKEN`, tokens de GitLab CI/CD y Bitbucket
- Archivos `.env`, `credentials.json`, `service-account.json` y otros archivos de configuración

Uno de los paquetes afectados fue @tiledesk/tiledesk-server, el cual incorporaba una carga bash codificada en Base64 dentro de un workflow de GitHub Actions. En total, se registraron 5,718 commits dirigidos a 5,561 repositorios distintos el 18 de mayo de 2026, entre las 11:36

a.m. y las 5:48 p.m. UTC.

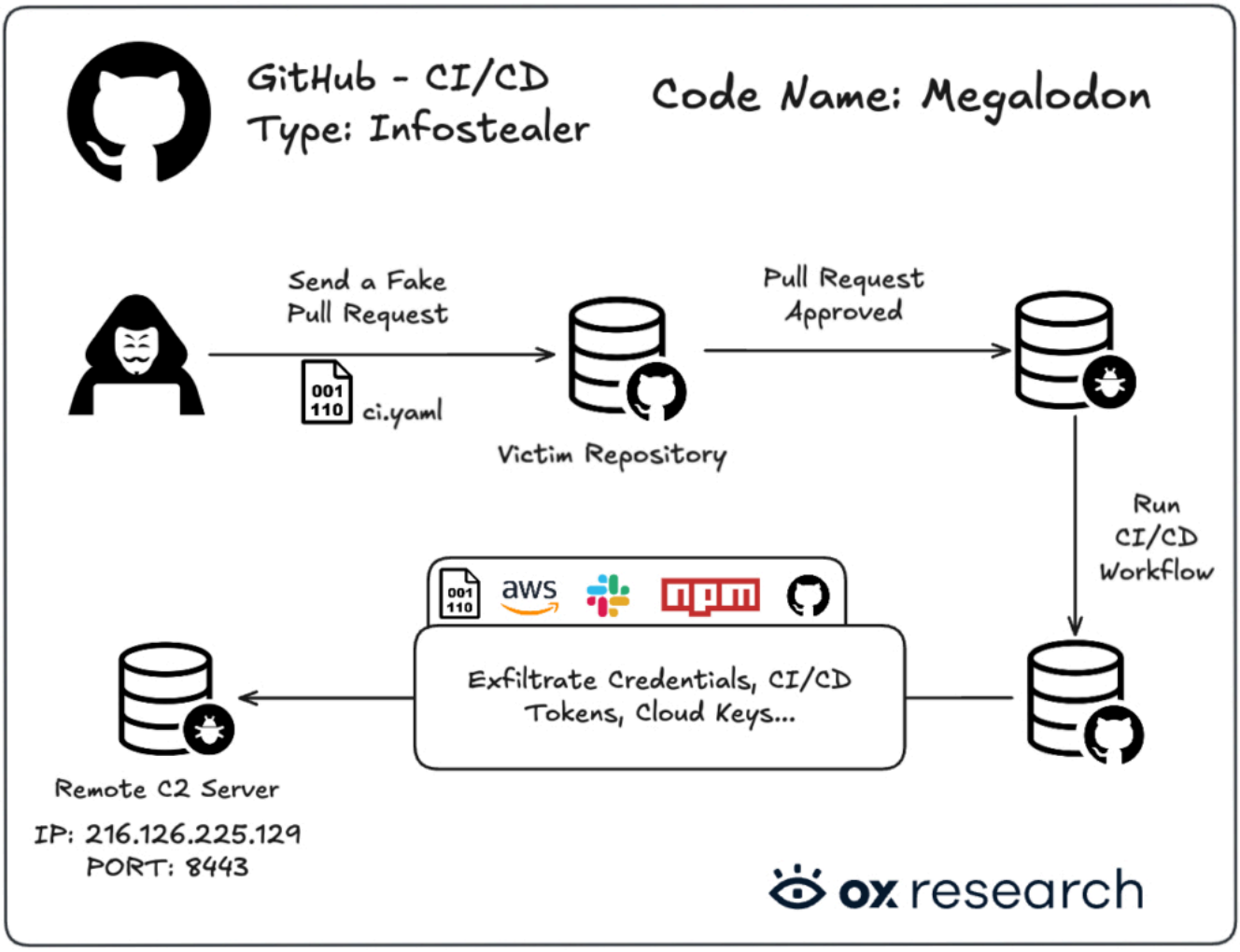
"El atacante alternó entre cuatro nombres de autor (build-bot, auto-ci, ci-bot y pipeline-bot) y siete mensajes de commit distintos, todos diseñados para parecer tareas rutinarias de mantenimiento CI", señaló SafeDep. *"También utilizó cuentas temporales de GitHub con nombres aleatorios de ocho caracteres (por ejemplo, rkb8el9r, bhlru9nr y lo6wt4t6), modificó la configuración de git para falsificar la identidad del autor y realizó los envíos utilizando PATs comprometidos o claves de despliegue."*

Como parte de esta campaña masiva se identificaron dos variantes de carga maliciosa:

- SysDiag, una variante masiva que agrega un nuevo workflow ejecutado automáticamente en cada push y pull request.
- Optimize-Build, una variante más selectiva que solo se activa mediante `workflow_dispatch`, un disparador de GitHub Actions que permite ejecutar workflows manualmente bajo demanda.

En el caso de Tiledesk, el enfoque selectivo fue utilizado específicamente para comprometer runners CI/CD y no durante la instalación del paquete npm.

"La diferencia principal es el alcance: usar `on: push` garantiza la ejecución en cada commit hacia master, alcanzando más objetivos sin intervención", agregó SafeDep. *"`workflow_dispatch` sacrifica alcance a cambio de mayor seguridad operativa. Con más de 5,700 repositorios comprometidos, incluso una pequeña fracción de `GITHUB_TOKEN` válidos proporciona suficientes objetivos para ejecuciones bajo demanda."*



El resultado es que, una vez que el propietario del repositorio fusiona el commit malicioso, el malware se ejecuta dentro de sus pipelines CI/CD y continúa propagándose, facilitando el robo masivo de credenciales y secretos.

"Hemos entrado en una nueva era de ataques a la cadena de suministro; la intrusión de TeamPCP en GitHub fue solo el comienzo", afirmó Moshe Siman Tov Bustan, de OX Security. *"Lo que viene después será una ola interminable, un auténtico tsunami de ataques cibernéticos dirigidos a desarrolladores de todo el mundo."*

Este desarrollo ocurre mientras TeamPCP ha convertido la cadena de suministro de software interconectada en un arma para comprometer cientos de herramientas de código abierto, infiltrándose en múltiples ecosistemas y, en algunos casos, extorsionando a las víctimas con fines económicos. GitHub, propiedad de Microsoft, se convirtió en la víctima más reciente de un listado que también incluye a TanStack, Grafana Labs, OpenAI y Mistral AI.

Las operaciones de TeamPCP han impulsado un ciclo continuo de explotación de proyectos open source populares, donde cada compromiso facilita el siguiente, permitiendo que el malware se expanda rápidamente con un comportamiento similar al de un gusano informático. El grupo también parece actuar motivado por beneficios económicos y habría establecido vínculos con BreachForums y otros grupos de extorsión como LAPSUS$ y VECT.

Además, existen indicios de motivaciones geopolíticas, evidenciadas por el despliegue de malware destructivo tipo wiper al detectar sistemas ubicados en Irán e Israel.

Las consecuencias de la ola de ataques de TeamPCP y del gusano Mini Shai-Hulud llevaron a npm a invalidar tokens de acceso granular con permisos de escritura que evitaban la autenticación multifactor (2FA). Asimismo, npm recomendó a los usuarios adoptar Trusted Publishing para reducir la dependencia de este tipo de credenciales.

"Al invalidar todos los tokens que eluden el 2FA en la plataforma, npm corta el acceso a las credenciales que el gusano ya había recopilado", explicó la firma de seguridad Socket. *"Los mantenedores generan nuevos tokens y el gusano, que sigue activo, vuelve a recolectarlos. El reinicio brinda tiempo, pero no soluciona la vulnerabilidad principal."*

Actividades como Megalodon y TeamPCP se basan en comprometer paquetes legítimos para distribuir malware. En contraste, una cuenta temporal denominada "polymarketdev" fue detectada publicando nueve paquetes npm maliciosos que se hacían pasar por herramientas CLI de trading de Polymarket en un intervalo de apenas 30 segundos, con el objetivo de robar claves privadas de Ethereum y Polygon mediante un script `postinstall`.

Al momento de redactarse este informe, los paquetes seguían disponibles para descarga en

npm. Sus nombres son:

- polymarket-trading-cli
- polymarket-terminal
- polymarket-trade
- polymarket-auto-trade
- polymarket-copy-trading
- polymarket-bot
- polymarket-claude-code
- polymarket-ai-agent
- polymarket-trader

"Durante la instalación, un script `postinstall` muestra un falso proceso de configuración de billetera solicitando al usuario que pegue su clave privada, asegurando que 'permanece cifrada'", explicó SafeDep. *"El script envía la clave sin cifrar mediante una petición POST a un Cloudflare Worker alojado en hxxps://polymarketbot.polymarketdev.workers[.]dev/v1/wallets/keys."*

"El atacante desarrolló una CLI de trading completamente funcional alrededor de una operación de robo de credenciales. La ingeniería social es el elemento central del ataque: el mensaje `postinstall` parece un procedimiento legítimo de configuración de wallet, el enmascaramiento imita una entrada segura y el repositorio en GitHub aporta una falsa sensación de legitimidad."