Identidad digital explicada

Qué son realmente los tokens de sesión, los tokens de actualización, las cookies, las claves de API y las claves de monedero, por qué los atacantes los tienen como objetivo, y cómo el motor de comportamiento a nivel de kernel de Cyber Crucible los protege sin leerlos jamás.

¿Recopilan algún dato de identidad?

La respuesta corta es no, no lo hacemos.

La respuesta más larga es, no, no lo hacemos, pero de dos maneras diferentes:

Primero, nunca transmitimos contraseñas, cookies, tokens de oauth, ni ninguna otra cosa que pudiera considerarse dato de identidad a nuestros servidores de Cyber Crucible. Eso es cierto ya sea que el servidor esté alojado por el cliente o alojado por Cyber Crucible.

Segundo, aprendimos a realizar análisis de comportamiento de la información de identidad, sin procesar realmente la información de identidad. Eso significa que nuestros análisis en realidad no procesan ni exponen datos sensibles como contraseñas reales, tokens de sesión, etc.

¿En Qué Se Diferencia Esto del Monitoreo de Identidad Normal?

La capacidad de Cyber Crucible frente al robo de identidad digital es proactiva y preventiva. Impide el acceso a la información necesaria para llevar a cabo operaciones de chantaje, extorsión y robo de identidad moderno.

La protección tradicional contra el robo de identidad se ocupa de las contraseñas y otros datos en manos de los estafadores, y del monitoreo de asuntos como el uso fraudulento de tarjetas de crédito, la obtención de préstamos fraudulentos o el hallazgo eventual de contraseñas en filtraciones de datos. Esto es muy reactivo y resulta difícil de gestionar para los clientes en su vida diaria. Además, abre la posibilidad, que Cyber Crucible ha observado, de chantajear a personas, e incluso coaccionar a empleados y ejecutivos para que ayuden a los atacantes a atacar a sus propios empleadores.

Existe un esfuerzo concertado por parte de los atacantes para obtener cookies, credenciales de inicio de sesión, plataformas de mensajería y otras formas de autenticación y datos privados que faciliten sus ataques.

Los análisis de Cyber Crucible enfocados en el robo de identidad previenen esto, de una manera que proporciona una toma de decisiones conductual autorizada sin necesidad de intervención del usuario, frente a las tácticas modernas de extorsión y los ataques cibercriminales sin archivos (file-less).

¿Qué datos de identidad se protegen?

Sistema operativo Windows principal

- NTDS (Active Directory)
- Función en fase de incubación que continuará mejorando

Credenciales de cliente VPN

- nombres de usuario
- contraseñas
- autenticación basada en claves

Navegadores web

- cookies
- tokens de actualización (tokens de autenticación “recordarme”)
- tokens de oAuth (tokens de sesión, tokens de acceso)
- contraseñas
- nombres de usuario
- historial de navegación

Aplicaciones de colaboración y mensajería

- contactos
- nombres de usuario
- historiales de chat
- contenidos de chat guardados
- archivos adjuntos de chat guardados
- claves de cifrado de mensajería privada
- contraseñas
- tokens de oauth (tokens de sesión, tokens de acceso, tokens de actualización)

Monederos de criptomonedas

- contenido de los monederos
- registros de transacciones
- contraseñas
- claves de cifrado

Juegos:

- En fase de incubación: actualmente solo Steam
- contraseñas
- compras
- credenciales
- nombres de usuario
- tokens de oAuth (tokens de sesión, tokens de acceso, tokens de actualización)

Compartición de archivos:

- En fase de incubación: actualmente solo FileZilla
- contraseñas
- compras
- credenciales
- nombres de usuario
- autenticación basada en claves
- servidores y configuraciones de servidor

¿Cómo se protegen los datos de identidad?

Respuesta breve: Cyber Crucible identifica las ubicaciones donde se almacenan los datos de identidad y protege esas ubicaciones a nivel de kernel. Cuando un programa intenta acceder a una de ellas, su acceso se rastrea y se analiza, **sin que las contraseñas, tokens o claves de cifrado en sí mismos sean accedidos ni recopilados nunca**. Con base en ese análisis, el programa es permitido, recibe datos falsos, es denegado o suspendido.

Proteger el acceso, no el secreto

La mayoría de las soluciones de protección de identidad actúan sobre el secreto: lo cifran, lo almacenan en una bóveda o vigilan si aparece en una filtración de datos. Cyber Crucible actúa sobre **el acto de intentar acceder a él**.

Los ataques automatizados son predecibles en un aspecto concreto: deben dirigirse a ubicaciones conocidas para encontrar credenciales. Los almacenes de cookies del navegador, las rutas de credenciales de VPN, el almacén de credenciales de Windows, los archivos de wallets y los datos de Active Directory se encuentran todos en lugares donde los atacantes ya saben buscar. Esa previsibilidad constituye la oportunidad defensiva.

Cyber Crucible identifica esos almacenes de datos de identidad y luego los protege. Cualquier programa que toque uno de ellos es evaluado, junto con sus procesos padre e hijo y las bibliotecas que ha cargado.

Los secretos nunca se leen

Esto es una restricción de diseño deliberada, no un efecto secundario: **el acceso se rastrea y se analiza sin que se acceda a la credencial en sí**. Cyber Crucible no lee, procesa, recopila ni transmite contraseñas, tokens de sesión, cookies ni claves de cifrado, a ningún servidor, ya sea alojado por Cyber Crucible o por el cliente.

Resulta que el análisis conductual del acceso a la identidad no requiere los datos de identidad en sí. Lo que importa es *qué programa está accediendo a dónde, en qué contexto, y qué ha hecho previamente*, nada de lo cual requiere abrir el secreto.

La consecuencia es que la protección en sí misma no puede convertirse en la brecha, y no existe un almacén central de sus credenciales que pueda ser objeto de una citación judicial, filtrarse o

perderse.

Qué evalúa el motor conductual

La decisión no es "¿está este programa en una lista?", sino "¿qué está haciendo realmente este código en este momento?". Las señales incluyen:

- **Estado y comportamiento de la memoria**: se rastrean a lo largo de todo el ciclo de vida del programa, incluidos los cambios en memoria que nunca llegan al disco. Esta es la capa de sensores fundamental de la plataforma, compartida con la protección de datos y FortressAI, en lugar de ser específica de la identidad.
- **Linaje del proceso**: qué lo inició y qué inició él a su vez.
- **Patrón de acceso**: ¿se trata de una aplicación normal que lee su propia credencial, o de un barrido simultáneo a través de múltiples ubicaciones de identidad?
- **Actividad de bibliotecas e inyección**: ¿se ha inyectado o alterado en memoria este proceso confiable?

La respuesta es gradual, no binaria

No todos los programas que intentan acceder a datos de identidad son atacantes, por lo que la respuesta depende de qué es el programa y cómo se está comportando:

Situación	Respuesta
Una aplicación conocida, no explotada, que accede a lo que legítimamente necesita	Permitido
Una aplicación conocida, no explotada, pero que **excede** su acceso apropiado	**Recibe datos falsos, o se le deniega el acceso** — regido por reglas conductuales de privacidad
Una aplicación **desconocida**	**Rechazada o suspendida**, según el motor conductual y su configuración
Una aplicación **comprometida** — proceso o bibliotecas explotados	**Rechazada o suspendida**, según el motor conductual y su configuración

El caso intermedio importa más de lo que parece a primera vista. Mucho software legítimo accede más allá de lo que necesita dentro de los almacenes de identidad. Suspenderlo arruinaría el día del usuario sin ninguna ganancia de seguridad, pero entregarle credenciales reales representa una exposición innecesaria. Devolver datos falsos satisface al programa mientras el secreto real permanece intacto.

La distinción que traza el motor es entre un programa confiable que *se extralimita* y un programa que es desconocido o que ha sido *tomado por control externo*. Solo este último se trata como un ataque.

Por qué esto ocurre en el endpoint

Un infostealer automatizado puede infiltrarse, recopilar datos y eliminar sus propias herramientas en memoria en cuestión de segundos. Enviar telemetría a un servicio en la nube y esperar un veredicto no puede competir con esa velocidad. Por ello, el ciclo completo de detectar-decidir-responder se ejecuta localmente en el kernel, lo cual también explica por qué funciona sin ninguna conectividad.

> Vea también: "*¿Recopilan algún dato de identidad?*", "*¿Qué datos de identidad se protegen?*" y "*¿Qué ocurre cuando un programa intenta acceder a datos de identidad que no debería?*"

¿Qué carteras de criptomonedas están protegidas?

- Coinomi
- Armory (“Bitcoin Armory”)
- Electrum
- Exodus
- Guarda

¿Qué ocurre cuando un programa intenta acceder a datos de identidad que no debería?

Respuesta breve: Depende de si el programa es de confianza y de si ha sido comprometido. A una aplicación legítima que simplemente se excede en sus permisos se le entregan datos falsos o se le deniega el acceso discretamente. Una aplicación desconocida o comprometida es rechazada o suspendida. La credencial real nunca se entrega en ninguno de estos casos.

Por qué una única respuesta sería incorrecta

El diseño obvio es "bloquear todo lo que toque un almacén de credenciales". Falla de inmediato en la práctica, porque el software legítimo interactúa constantemente con esos almacenes: navegadores, gestores de contraseñas, clientes VPN, herramientas de sincronización y agentes de copia de seguridad tienen razones reales para estar ahí.

Bloquear todo rompe el equipo. Permitir todo lo que es de confianza significa que un navegador explotado obtiene lo que solicita. La pregunta útil es más específica: **¿este programa se está comportando como debería?**

Los cuatro resultados posibles

1. Permitido. Una aplicación conocida y no explotada que accede a lo que legítimamente necesita. No cambia nada.

2. Se devuelven datos falsos. Una aplicación conocida y no explotada que excede su acceso apropiado. Se aplican las reglas de comportamiento de privacidad: el programa recibe datos plausibles pero falsos. Continúa ejecutándose con normalidad, y la credencial real nunca sale del almacenamiento. Desde la perspectiva del programa, nada falló, y precisamente por eso esto funciona sin romper los flujos de trabajo.

3. Acceso denegado. La misma situación, cuando devolver datos falsos no es apropiado. La solicitud se rechaza; la aplicación continúa ejecutándose.

4. Rechazado o suspendido. El programa es desconocido, o es un programa conocido cuyo proceso o bibliotecas muestran signos de explotación. Aquí la preocupación no es el exceso de alcance, sino que el programa ya no actúa como sí mismo. Que sea rechazado o suspendido depende de la evaluación del motor de comportamiento y de la configuración establecida.

Por qué el engaño en lugar del bloqueo

Devolver datos falsos es una decisión deliberada. Una solicitud bloqueada le indica a un atacante que ha sido detectado y lo impulsa a intentar otra vía. Los datos falsos que parecen reales no lo hacen: la herramienta del atacante continúa con credenciales que no desbloquean nada, y el operador puede no enterarse de que algo anda mal durante algún tiempo.

Este es el mismo principio detrás de los archivos señuelo (canary files) en la defensa contra ransomware: darle al atacante algo convincente que capturar y que no le cuesta nada a usted.

Configurabilidad

El límite entre el rechazo y la suspensión es ajustable. Los entornos con baja tolerancia a las interrupciones —sistemas clínicos, líneas de producción— pueden inclinarse hacia la denegación y los datos falsos en lugar de la suspensión, mientras que los entornos de alta seguridad pueden ser más agresivos.

¿Qué es un token de sesión y por qué robarlo es peor que robar una contraseña?

Respuesta breve: Un token de sesión es la credencial que tu navegador o aplicación conserva *después* de iniciar sesión, y que demuestra que ya estás autenticado. Robar uno suele ser peor que robar una contraseña porque normalmente evita tanto la contraseña como la autenticación multifactor: el atacante simplemente retoma tu sesión existente.

Por qué existe

Volver a introducir la contraseña en cada solicitud sería inviable, por lo que, tras un inicio de sesión exitoso, el servicio emite un token. Cada solicitud posterior lleva ese token en lugar de tus credenciales. El servicio confía en él como prueba de que ya te autenticaste.

Por qué los atacantes lo prefieren

- **Evita la MFA.** La MFA se verifica en el inicio de sesión. Un token emitido *después* de esa verificación representa una autenticación ya superada, por lo que reutilizarlo normalmente no vuelve a activar el segundo factor.
- **Parece legítimo.** Para el servicio, un token válido equivale a un usuario válido. No hay señal de inicio de sesión fallido ni nada anómalo que genere una alerta.
- **Es portátil.** Muchos tokens funcionan desde una máquina, una red o un país distintos.

Por qué esto cambió el comportamiento de los atacantes

Los atacantes evitan cada vez más permanecer en una red. Quedarse residente aumenta las probabilidades de detección. En cambio, se llevan el material de identidad y se marchan, para luego regresar cuando quieran como un usuario autenticado. El token es una puerta trasera de apariencia legítima que el propio sistema de autenticación emitió.

Cómo lo aborda Cyber Crucible

El robo de tokens se detiene en el momento del acceso. Cuando un programa intenta acceder al almacenamiento del navegador o de la aplicación donde residen los tokens, su acceso se evalúa en menos de 200 milisegundos. A un programa legítimo pero que se extralimita se le entregan datos falsos o se le deniega el acceso; uno desconocido o comprometido es rechazado o suspendido. En cualquier caso, el token real nunca se entrega.

¿Qué es un token de actualización (refresh token) y por qué su pérdida es especialmente perjudicial?

Respuesta breve: Un token de actualización (refresh token) es una credencial de larga duración que se usa para obtener silenciosamente nuevos tokens de acceso sin que el usuario tenga que iniciar sesión de nuevo. Es en lo que se basa la opción "recordarme". Debido a que puede seguir generando accesos nuevos indefinidamente, un token de actualización robado puede otorgar a un atacante acceso duradero mucho después de que finalice su sesión.

Token de acceso frente a token de actualización

- **Token de acceso (token de sesión):** de corta duración, se utiliza en cada solicitud y expira en minutos u horas.
- **Token de actualización:** de larga duración, se utiliza únicamente para solicitar nuevos tokens de acceso. A veces es válido durante semanas o meses.

Un token de acceso robado eventualmente expira. Un token de actualización robado puede utilizarse para generar nuevos tokens de acceso repetidamente, lo que convierte un robo puntual en un acceso persistente.

Por qué "recordarme" implica una compensación

Permanecer conectado requiere almacenar algo duradero en el dispositivo. Precisamente esa comodidad es lo que hace que el token almacenado resulte valioso para un atacante. Los almacenes de credenciales de navegadores y aplicaciones son ubicaciones estándar y predecibles, por lo que las herramientas automatizadas van directamente a ellos.

Cómo reducir el daño

Revocar un token de actualización invalida el acceso que este puede generar, razón por la cual "cerrar sesión en todos los dispositivos" y la revocación de tokens son importantes tras cualquier sospecha de compromiso. Sin embargo, la revocación es una respuesta *posterior* al robo.

El enfoque de Cyber Crucible consiste en impedir la lectura. El almacén de credenciales es una ubicación protegida: un programa legítimo que intenta acceder de forma indebida recibe datos falsos o se le deniega el acceso, mientras que uno desconocido o comprometido es rechazado o suspendido. El token de actualización en sí nunca se entrega.

¿Qué son las cookies del navegador y cómo las explotan los atacantes?

Respuesta breve: Las cookies son pequeños fragmentos de datos que un sitio web almacena en su navegador, y algunas de ellas son cookies de autenticación que mantienen su sesión iniciada. Robar esas es funcionalmente equivalente a robar una sesión activa: el atacante puede cargarlas en su propio navegador y aparentar ser usted.

No todas las cookies importan por igual

- **Cookies de preferencias** — idioma, tema, diseño. Valor bajo.
- **Cookies de análisis** — medición de uso. Valor bajo.
- **Cookies de autenticación / sesión** — demuestran que ha iniciado sesión. **Valor alto.**

Los atacantes se interesan casi exclusivamente en la tercera categoría.

Por qué el robo de cookies es efectivo

Una cookie de autenticación robada normalmente no desencadena un evento de inicio de sesión, una solicitud de contraseña ni un desafío de MFA. La sesión ya existe; el atacante simplemente presenta una prueba de ella. Desde la perspectiva de la aplicación, esto parece actividad continua ordinaria.

Los navegadores almacenan las cookies en ubicaciones de archivo conocidas y predecibles en todos los sistemas operativos, por lo que las herramientas automatizadas pueden localizarlas sin saber nada específico sobre la máquina en cuestión.

Por qué el navegador es un punto focal

El navegador aloja cookies, contraseñas guardadas, tokens OAuth e historial en un solo lugar, lo que lo convierte en el objetivo de identidad más rico de la mayoría de los endpoints. Cyber Crucible ha observado esto directamente: desde el cuarto trimestre de 2023 en adelante, las respuestas automatizadas contra actividad de Chrome, Edge y Chromium aumentaron de cero a miles, con

CVE relacionados publicados posteriormente por Google y Microsoft.

Protección

El almacenamiento de credenciales del navegador es una de las ubicaciones de identidad protegidas. Un proceso que intenta leerlo se evalúa en contexto: un navegador explotado se trata de forma muy distinta a uno sano, y en ninguno de los dos casos una solicitud excesiva recibe la cookie real.

¿Qué es una clave de API y qué ocurre si es robada?

Respuesta breve: Una clave de API es una cadena secreta que identifica y autoriza a un programa —en lugar de a una persona— cuando llama a un servicio. Una clave robada permite a un atacante realizar solicitudes en nombre de su aplicación, a menudo sin interacción del usuario, sin MFA y sin ninguna señal evidente de que algo esté mal.

Por qué son objetivos atractivos

- **No hay un humano en el ciclo.** Las claves están diseñadas para uso automatizado y desatendido, por lo que no hay ninguna solicitud de MFA que pueda interferir.
- **A menudo con un alcance demasiado amplio.** Las claves se emiten con frecuencia con más permisos de los que la tarea requiere.
- **De larga duración.** Muchas nunca se rotan, por lo que una clave robada hoy puede seguir funcionando meses después.
- **Almacenadas en lugares predecibles.** Archivos de configuración, variables de entorno y almacenes de credenciales: todos ellos ubicaciones que la automatización puede enumerar.

Consecuencias típicas

Según lo que autorice la clave: leer o exportar datos de clientes, enviar mensajes en nombre de su organización, gastar dinero contra una cuenta en la nube, o desplazarse hacia otros sistemas conectados.

Dado que las solicitudes están correctamente autenticadas, suelen aparecer en los registros como tráfico normal de la aplicación. La detección suele producirse mucho después de los hechos, mediante una factura o una auditoría.

Reducción de la exposición

Limite el alcance de las claves, rótelas regularmente y manténgalas fuera del control de versiones. Son buenas prácticas, pero limitan el radio de impacto en lugar de prevenir el robo.

Cyber Crucible se enfoca en el propio robo. Las ubicaciones de credenciales y configuración están protegidas, y un programa que las lee se evalúa en contexto: datos falsos o denegación ante una extralimitación legítima, rechazo o suspensión ante un proceso desconocido o comprometido. En ninguno de los dos casos se entrega la clave.

¿Dónde almacena Windows las contraseñas y las credenciales?

Respuesta breve: En varios lugares predecibles: el Administrador de credenciales de Windows, los almacenes de contraseñas del navegador, la configuración del cliente VPN y, en los controladores de dominio, la base de datos de Active Directory (NTDS). "Predecible" es la palabra clave: los ataques automatizados dependen de que esas ubicaciones sean las mismas en todas las máquinas.

Los principales almacenes

- **Administrador de credenciales de Windows** — inicios de sesión guardados para recursos de red y aplicaciones.
- **Almacenes de contraseñas del navegador** — Chrome, Edge y Firefox conservan cada uno las credenciales guardadas en rutas de perfil conocidas.
- **Credenciales del cliente VPN** — nombres de usuario, contraseñas y material de autenticación basado en claves.
- **NTDS (Active Directory)** — en un controlador de dominio, los datos de credenciales de todo el dominio. El objetivo de mayor valor en la red.
- **Almacenes específicos de aplicaciones** — las herramientas de mensajería y colaboración, los clientes de transferencia de archivos y las plataformas de juegos mantienen cada uno los suyos propios.

Por qué la previsibilidad es todo el asunto

Los atacantes no saben nada acerca de su entorno específico. Su automatización está escrita para rutas que existen en todas partes: `C:\Users\[Username]`, carpetas estándar de datos de aplicaciones, letras de unidad estándar. Un script no necesita entender su red; simplemente recorre rutas conocidas.

Esa es una debilidad de su método. Si esas ubicaciones exactas son las que se están vigilando, el propio requisito de fiabilidad del atacante se convierte en el disparador que lo atrapa.

Qué hace Cyber Crucible al respecto

Estos almacenes son los puntos de entrada de robo de identidad que se monitorean. La intención de un programa que los lee se evalúa en menos de 200 milisegundos, y se suspende si esa intención es maliciosa, antes de que se sustraiga cualquier credencial.

¿Qué son las claves de las carteras de criptomonedas y por qué su robo es irreversible?

Respuesta breve: una clave de cartera es la clave criptográfica privada que autoriza el gasto desde una dirección de criptomoneda. Quien la posea controla los fondos. El robo es prácticamente irreversible porque las transacciones en blockchain no se pueden revertir y no existe una institución ante la cual apelar.

Por qué esto difiere de un banco

Un cargo fraudulento con tarjeta puede impugnarse y ser revertido por el emisor. Una transacción de criptomoneda firmada con su clave privada es válida por definición: la red no puede distinguir un robo de una transferencia legítima, y no existe una autoridad central para deshacerla.

Esto hace que las claves de cartera sean especialmente atractivas: la ganancia es inmediata, definitiva y difícil de rastrear.

Qué buscan los atacantes

- **Claves privadas y frases semilla** almacenadas en los archivos de la aplicación de la cartera
- **Contenido de la cartera y registros de transacciones**, útiles para identificar a las víctimas de mayor valor
- **Contraseñas que protegen la cartera**, a menudo reutilizadas en otros lugares

El software de cartera de escritorio almacena este material en ubicaciones predecibles de datos de aplicación, el mismo patrón que hace que cualquier otro almacén de credenciales sea accesible mediante automatización.

Protección

Los archivos de cartera se encuentran entre las ubicaciones de identidad que Cyber Crucible supervisa. Dado que la respuesta consiste en la suspensión en el momento del acceso malicioso, el robo se previene en lugar de detectarse posteriormente, lo cual es más importante aquí que en

casi cualquier otro caso, ya que no existe una vía de recuperación una vez que los fondos se transfieren.

> Consulte también: "*¿Qué carteras de criptomonedas están protegidas?*" para conocer las aplicaciones específicas actualmente cubiertas.

¿Qué es la autenticación basada en claves y por qué las claves robadas otorgan acceso duradero?

Respuesta breve: La autenticación basada en claves demuestra la identidad mediante una clave privada criptográfica en lugar de una contraseña, y es utilizada por SSH, muchas VPN y clientes de transferencia de archivos. Una clave privada robada otorga acceso sin necesidad de contraseña ni de una solicitud de MFA, y como las claves rara vez se rotan, ese acceso puede persistir durante mucho tiempo.

Cómo funciona

Usted posee una clave privada; el servidor posee la clave pública correspondiente. Al momento de la conexión, el servidor emite un desafío que solo la clave privada puede responder. La clave en sí nunca atraviesa la red.

Esto es genuinamente más sólido que las contraseñas: resiste la adivinación, el phishing y la reutilización. Su debilidad es distinta: **todo depende de que el archivo de la clave privada se mantenga privado.**

Por qué una clave robada es tan duradera

- **Sin solicitud de contraseña.** La clave *es* la credencial.
- **Frecuentemente sin MFA.** La autenticación por clave a menudo funciona sola, particularmente en sistemas automatizados.
- **Rara vez se rota.** Las contraseñas caducan según un calendario; las claves suelen durar años.
- **Almacenamiento predecible.** Directorios estándar y rutas de configuración de clientes VPN.

Una clave robada es una puerta trasera silenciosa y duradera que no genera inicios de sesión fallidos y que parece completamente legítima.

Protección

El material de autenticación basado en claves y de VPN se encuentra explícitamente entre las categorías de identidad protegidas. El acceso se evalúa a nivel del kernel, de modo que un programa que recolecta archivos de claves es detenido —denegado o alimentado con datos falsos si se trata de un programa de confianza que se extralimita, rechazado o suspendido si es desconocido o está comprometido— antes de que la clave salga del equipo.

¿Por qué la protección de los datos de identidad requiere acceso a nivel de kernel?

Respuesta breve: Porque el robo ocurre en el espacio entre el momento en que un programa solicita datos de credenciales y el momento en que el sistema operativo los entrega. Solo una defensa que opera en el kernel —por debajo de las aplicaciones y bibliotecas que un atacante puede manipular— puede ver esa solicitud, evaluar su intención y detenerla a tiempo.

Los tres requisitos

1. Ver el acceso en el momento en que ocurre. El robo de identidad no es un archivo que aparece en el disco; es una lectura. Las herramientas que buscan archivos maliciosos no ven nada, porque en un infostealer moderno no se escribe nada. El evento que hay que detectar es el intento de acceso en sí.

2. Decidir con la suficiente rapidez. Las herramientas automatizadas pueden infiltrarse, recopilar datos y autoeliminarse en cuestión de segundos. Cualquier arquitectura que envíe telemetría a un servicio en la nube y espere un veredicto ya ha perdido la carrera. La decisión debe tomarse localmente, en milisegundos.

3. No depender de lo que el atacante controla. Las herramientas de seguridad que dependen de bibliotecas del sistema operativo pueden quedar ciegas cuando un atacante compromete esas bibliotecas en memoria: el agente sigue ejecutándose y no informa nada. Cyber Crucible fue deliberadamente desacoplado de las bibliotecas de Windows para que un sistema operativo comprometido no pueda silenciarlo.

La consecuencia

El funcionamiento a nivel de kernel es lo que permite las tres cosas a la vez: visibilidad sobre el intento de acceso, una decisión local en menos de 200 milisegundos, e independencia de un sistema operativo potencialmente comprometido.

Esto también explica por qué la protección funciona sin conectividad —en entornos con air gap, sin conexión, o en el mar— ya que no es necesario que nada salga del dispositivo para que se tome una decisión.