# Digitale Identität erklärt

Was Sitzungstoken, Aktualisierungstoken, Cookies, API-Schlüssel und Wallet-Schlüssel tatsächlich sind, warum Angreifer sie ins Visier nehmen und wie die kernelbasierte Verhaltensanalyse-Engine von Cyber Crucible sie schützt, ohne sie jemals auszulesen.

# Erfassen Sie Identitätsdaten?

Die kurze Antwort lautet: Nein, das tun wir nicht.

Die längere Antwort lautet ebenfalls Nein, jedoch aus zwei unterschiedlichen Gründen:

Erstens übertragen wir niemals Passwörter, Cookies, OAuth-Token oder sonstige Informationen, die als Identitätsdaten gelten würden, an unsere Cyber Crucible-Server. Das gilt unabhängig davon, ob der Server vom Kunden oder von Cyber Crucible gehostet wird.

Zweitens haben wir gelernt, Verhaltensanalysen von Identitätsinformationen durchzuführen, ohne diese Identitätsinformationen tatsächlich zu verarbeiten. Das bedeutet, dass unsere Analysen keine sensiblen Daten wie tatsächliche Passwörter, Sitzungstoken usw. verarbeiten oder offenlegen.

# Wie unterscheidet sich dies von einer normalen Identitätsüberwachung?

Die Fähigkeit von Cyber Crucible zum Schutz vor digitalem Identitätsdiebstahl ist proaktiv und präventiv. Sie verhindert den Zugriff auf die Informationen, die für Erpressung, Nötigung und moderne Identitätsdiebstahl-Operationen benötigt werden.

Herkömmlicher Identitätsdiebstahlschutz befasst sich mit Passwörtern und anderen Daten, die in die Hände von Betrügern gelangt sind, sowie mit der Überwachung von Vorgängen wie betrügerischer Kreditkartennutzung, der Aufnahme betrügerischer Kredite oder dem letztendlichen Auffinden von Passwörtern in Datenlecks. Dies ist sehr reaktiv und für Kunden im Alltag nur schwer zu handhaben. Zudem eröffnet dies die Möglichkeit – die Cyber Crucible beobachtet hat –, Einzelpersonen zu erpressen und sogar Mitarbeiter und Führungskräfte dazu zu nötigen, den Angreifern bei Angriffen auf ihre Arbeitgeber zu helfen.

Angreifer bemühen sich gezielt darum, an Cookies, Anmeldedaten, Nachrichtenplattformen und andere Formen der Authentifizierung sowie private Daten zu gelangen, um ihre Angriffe zu ermöglichen.

Die auf Identitätsdiebstahl fokussierten Analysen von Cyber Crucible verhindern dies auf eine Weise, die eine maßgebliche verhaltensbasierte Entscheidungsfindung ohne Nutzerbeteiligung ermöglicht – gegen moderne Erpressungstaktiken und dateilose Cyberangriffe.

# Welche Identitätsdaten werden geschützt?

## Windows-Kernbetriebssystem

- NTDS (Active Directory)
- Sich in der Entwicklung befindende Funktion, die weiter verbessert wird

## VPN-Client-Anmeldedaten

- Benutzernamen
- Passwörter
- schlüsselbasierte Authentifizierung

## Webbrowser

- Cookies
- Refresh-Token („Angemeldet bleiben“-Authentifizierungstoken)
- oAuth-Token (Sitzungstoken, Zugriffstoken)
- Passwörter
- Benutzernamen
- Browserverlauf

## Kollaborations- & Messaging-Anwendungen

- Kontakte
- Benutzernamen
- Chatverläufe
- gespeicherte Chatinhalte
- gespeicherte Chatanhänge
- Verschlüsselungsschlüssel für private Nachrichten
- Passwörter
- oAuth-Token (Sitzungstoken, Zugriffstoken, Refresh-Token)

## Krypto-Wallets

- Wallet-Inhalte
- Transaktionsaufzeichnungen
- Passwörter
- Verschlüsselungsschlüssel

# Gaming:

- Sich in der Entwicklung befindend - derzeit nur Steam
- Passwörter
- Käufe
- Anmeldedaten
- Benutzernamen
- oAuth-Token (Sitzungstoken, Zugriffstoken, Refresh-Token)

# Dateifreigabe:

- Sich in der Entwicklung befindend - derzeit nur FileZilla
- Passwörter
- Käufe
- Anmeldedaten
- Benutzernamen
- schlüsselbasierte Authentifizierung
- Server & Servereinstellungen

# Wie werden Identitätsdaten geschützt?

**Kurze Antwort:** Cyber Crucible identifiziert die Orte, an denen Identitätsdaten gespeichert sind, und schützt diese Orte auf Kernel-Ebene. Wenn ein Programm auf einen solchen Ort zugreift, wird dieser Zugriff verfolgt und analysiert — **ohne dass die Passwörter, Tokens oder Verschlüsselungsschlüssel selbst jemals abgerufen oder gesammelt werden**. Auf Basis dieser Analyse wird das Programm zugelassen, mit gefälschten Daten versorgt, abgelehnt oder ausgesetzt.

# Schutz des Zugriffs, nicht des Geheimnisses

Die meisten Ansätze zum Schutz von Identitäten setzen beim Geheimnis selbst an: verschlüsseln, in einem Tresor ablegen oder beobachten, ob es in einem Datenleck auftaucht. Cyber Crucible setzt beim **Akt des Zugreifens** an.

Automatisierte Angriffe sind in einem bestimmten Punkt vorhersehbar — sie müssen bekannte Orte aufsuchen, um Zugangsdaten zu finden. Browser-Cookie-Speicher, VPN-Zugangsdatenpfade, der Windows-Anmeldeinformationsspeicher, Wallet-Dateien und Active-Directory-Daten befinden sich alle dort, wo Angreifer bereits wissen, wo sie suchen müssen. Diese Vorhersehbarkeit ist die defensive Chance.

Cyber Crucible identifiziert diese Identitätsdatenspeicher und schützt sie anschließend. Jedes Programm, das einen davon berührt, wird bewertet — zusammen mit seinen übergeordneten und untergeordneten Prozessen sowie den von ihm geladenen Bibliotheken.

# Die Geheimnisse werden niemals gelesen

Dies ist eine bewusste Designentscheidung, kein Nebeneffekt: **Der Zugriff wird verfolgt und analysiert, ohne dass die Zugangsdaten selbst abgerufen werden.** Cyber Crucible liest, verarbeitet, sammelt oder überträgt keine Passwörter, Sitzungstokens, Cookies oder Verschlüsselungsschlüssel — an keinen Server, weder von Cyber Crucible noch vom Kunden gehostet.

Es stellt sich heraus, dass die Verhaltensanalyse des Identitätszugriffs die Identitätsdaten selbst nicht benötigt. Entscheidend ist, *welches Programm wo, in welchem Kontext und nach welchen vorherigen Aktionen zugreift* — nichts davon erfordert das Öffnen des Geheimnisses.

Die Folge ist, dass der Schutzmechanismus selbst nicht zur Sicherheitsverletzung werden kann und es keinen zentralen Speicher Ihrer Zugangsdaten gibt, der vorgeladen, geleakt oder verloren werden könnte.

# Was die Verhaltens-Engine bewertet

Die Entscheidung lautet nicht „Steht dieses Programm auf einer Liste?“, sondern „Was tut dieser Code gerade tatsächlich?“ Zu den Signalen gehören:

- **Speicherzustand und -verhalten** — über den gesamten Lebenszyklus des Programms verfolgt, einschließlich Änderungen im Arbeitsspeicher, die niemals auf die Festplatte gelangen. Dies ist die grundlegende Sensor-Ebene der Plattform, die mit dem Datenschutz und FortressAI gemeinsam genutzt wird, statt spezifisch für Identität zu sein.
- **Prozessabstammung** — was dies gestartet hat und was es wiederum gestartet hat.
- **Zugriffsmuster** — handelt es sich um eine normale Anwendung, die ihre eigenen Zugangsdaten liest, oder um ein gleichzeitiges Durchsuchen mehrerer Identitätsorte?
- **Bibliotheks- und Injektionsaktivität** — wurde in diesen vertrauenswürdigen Prozess injiziert oder wurde er im Speicher verändert?

# Die Reaktion ist abgestuft, nicht binär

Nicht jedes Programm, das auf Identitätsdaten zugreift, ist ein Angreifer. Daher hängt die Reaktion davon ab, um welches Programm es sich handelt und wie es sich verhält:

| Situation | Reaktion |
|---|---|
| Eine bekannte Anwendung, nicht kompromittiert, greift auf das zu, was sie legitim benötigt | Zugelassen |
| Eine bekannte Anwendung, nicht kompromittiert, greift jedoch über ihren angemessenen Zugriff hinaus | **Erhält gefälschte Daten oder wird der Zugriff verweigert** — gesteuert durch Verhaltensregeln zum Datenschutz |
| Eine **unbekannte** Anwendung | **Abgelehnt oder ausgesetzt**, gemäß der Verhaltens-Engine und Ihren Einstellungen |
| Eine **kompromittierte** Anwendung — Prozess oder Bibliotheken ausgenutzt | **Abgelehnt oder ausgesetzt**, gemäß der Verhaltens-Engine und Ihren Einstellungen |

Der mittlere Fall ist wichtiger, als es zunächst scheint. Viele legitime Softwareprogramme greifen weiter in Identitätsspeicher hinein, als sie eigentlich benötigen. Sie auszusetzen würde den Arbeitsalltag des Nutzers ohne jeglichen Sicherheitsgewinn stören — ihnen jedoch echte

Zugangsdaten zu übergeben, wäre eine unnötige Gefährdung. Die Rückgabe gefälschter Daten befriedigt das Programm, während das echte Geheimnis an seinem Ort verbleibt.

Die Unterscheidung, die die Engine trifft, liegt zwischen einem vertrauenswürdigen Programm, das *übergreift*, und einem Programm, das unbekannt ist oder *übernommen* wurde. Nur Letzteres wird als Angriff behandelt.

# Warum dies auf dem Endpunkt geschieht

Ein automatisierter Infostealer kann innerhalb von Sekunden eindringen, Daten sammeln und seine eigenen speicherresidenten Werkzeuge löschen. Telemetriedaten an einen Cloud-Dienst zu senden und auf ein Urteil zu warten, kann dagegen nicht bestehen. Der vollständige Detect-Decide-Respond-Zyklus läuft daher lokal im Kernel — dies ist auch der Grund, warum er völlig ohne Konnektivität funktioniert.

> Siehe auch: *„Sammeln Sie Identitätsdaten?“*, *„Welche Identitätsdaten werden geschützt?“* und *„Was passiert, wenn ein Programm versucht, auf Identitätsdaten zuzugreifen, auf die es keinen Zugriff haben sollte?“*

# Welche Kryptowährungs-Wallets werden geschützt?

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

# Was passiert, wenn ein Programm versucht, auf Identitätsdaten zuzugreifen, auf die es keinen Zugriff haben sollte?

**Kurze Antwort:** Es hängt davon ab, ob das Programm vertrauenswürdig ist und ob es kompromittiert wurde. Eine legitime Anwendung, die lediglich über ihre Befugnisse hinausgeht, erhält gefälschte Daten oder wird stillschweigend abgewiesen. Eine unbekannte oder kompromittierte Anwendung wird abgelehnt oder suspendiert. Die echten Anmeldedaten werden in keinem dieser Fälle preisgegeben.

## Warum eine einzige Reaktion falsch wäre

Der naheliegende Ansatz lautet: „Alles blockieren, was einen Credential Store berührt.“ In der Praxis scheitert dies sofort, denn legitime Software greift ständig auf solche Speicher zu — Browser, Passwort-Manager, VPN-Clients, Sync-Tools und Backup-Agenten haben allesamt echte Gründe, dort präsent zu sein.

Alles zu blockieren würde das System lahmlegen. Alles Vertrauenswürdige zuzulassen bedeutet, dass ein ausgenutzter Browser bekommt, was er anfordert. Die entscheidende Frage ist enger gefasst: **Verhält sich dieses Programm so, wie es sollte?**

## Die vier möglichen Ergebnisse

**1. Zugelassen.** Eine bekannte, nicht kompromittierte Anwendung greift auf das zu, was sie legitimerweise benötigt. Es ändert sich nichts.

**2. Gefälschte Daten werden zurückgegeben.** Eine bekannte, nicht kompromittierte Anwendung überschreitet ihren zulässigen Zugriffsbereich. Es gelten die Verhaltensregeln zum Datenschutz: Das Programm erhält plausible, aber falsche Daten. Es läuft normal weiter, und die

echten Anmeldedaten verlassen niemals den Speicher. Aus Sicht des Programms ist nichts fehlgeschlagen — genau deshalb funktioniert dies, ohne Arbeitsabläufe zu stören.

**3. Zugriff verweigert.** Dieselbe Situation, in der die Rückgabe gefälschter Daten nicht angemessen ist. Die Anfrage wird abgelehnt; die Anwendung läuft weiter.

**4. Abgelehnt oder suspendiert.** Das Programm ist unbekannt, oder ein bekanntes Programm zeigt in seinem Prozess oder seinen Bibliotheken Anzeichen einer Kompromittierung. Hier geht es nicht um Überschreitung von Befugnissen, sondern darum, dass das Programm nicht mehr wie es selbst agiert. Ob es abgelehnt oder suspendiert wird, hängt von der Einschätzung der Verhaltens-Engine und Ihren konfigurierten Einstellungen ab.

# Warum Täuschung statt Blockierung

Die Rückgabe gefälschter Daten ist eine bewusste Entscheidung. Eine blockierte Anfrage signalisiert einem Angreifer, dass er entdeckt wurde, und veranlasst ihn, einen anderen Weg zu versuchen. Gefälschte Daten, die echt aussehen, tun dies nicht — das eingesetzte Tool arbeitet mit Anmeldedaten weiter, die nichts entsperren, und der Angreifer bemerkt möglicherweise über längere Zeit nicht, dass etwas nicht stimmt.

Dies ist dasselbe Prinzip, das auch Canary-Dateien in der Ransomware-Abwehr zugrunde liegt: dem Angreifer etwas Überzeugendes zum Ergreifen geben, das Sie selbst nichts kostet.

# Konfigurierbarkeit

Die Grenze zwischen Ablehnung und Suspendierung ist einstellbar. Umgebungen mit geringer Toleranz gegenüber Unterbrechungen — klinische Systeme, Produktionslinien — können stärker auf Verweigerung und gefälschte Daten anstelle von Suspendierung setzen, während hochsichere Umgebungen aggressiver vorgehen können.

# Was ist ein Sitzungstoken, und warum ist der Diebstahl eines solchen schlimmer als der Diebstahl eines Passworts?

**Kurze Antwort:** Ein Sitzungstoken (Session Token) ist der Berechtigungsnachweis, den Ihr Browser oder Ihre App *nach* der Anmeldung besitzt und der beweist, dass Sie bereits authentifiziert sind. Der Diebstahl eines solchen Tokens ist oft schlimmer als der Diebstahl eines Passworts, da er in der Regel sowohl das Passwort als auch die Multi-Faktor-Authentifizierung umgeht — der Angreifer setzt einfach Ihre bestehende Sitzung fort.

## Warum es das gibt

Bei jeder Anfrage das Passwort erneut einzugeben, wäre nicht praktikabel. Deshalb stellt der Dienst nach einer erfolgreichen Anmeldung ein Token aus. Jede weitere Anfrage trägt dieses Token anstelle Ihrer Anmeldedaten. Der Dienst vertraut darauf als Nachweis, dass Sie sich bereits authentifiziert haben.

## Warum Angreifer es bevorzugen

- **Es umgeht MFA.** MFA wird bei der Anmeldung geprüft. Ein Token, das *nach* dieser Prüfung ausgestellt wird, repräsentiert eine bereits erfolgte Authentifizierung, sodass das erneute Verwenden dieses Tokens in der Regel nicht erneut den zweiten Faktor auslöst.
- **Es wirkt legitim.** Für den Dienst ist ein gültiges Token ein gültiger Benutzer. Es gibt kein Signal für fehlgeschlagene Anmeldungen und nichts Auffälliges, das einen Alarm auslösen würde.
- **Es ist übertragbar.** Viele Token funktionieren von einem anderen Rechner, Netzwerk oder Land aus.

## Warum sich dadurch das Verhalten von Angreifern verändert hat

Angreifer vermeiden es zunehmend, sich lange in einem Netzwerk aufzuhalten. Ein längerer Verbleib erhöht die Wahrscheinlichkeit einer Entdeckung. Stattdessen entwenden sie Identitätsmaterial und verschwinden — um anschließend jederzeit als authentifizierter Benutzer zurückzukehren. Das Token ist eine legitim aussehende Hintertür, die vom Authentifizierungssystem selbst ausgestellt wurde.

# Wie Cyber Crucible dem begegnet

Token-Diebstahl wird im Moment des Zugriffs gestoppt. Wenn ein Programm auf den Browser- oder Anwendungsspeicher zugreift, in dem Token abgelegt sind, wird dieser Zugriff in unter 200 Millisekunden bewertet. Ein übergriffiges, aber legitimes Programm erhält gefälschte Daten oder wird abgewiesen; ein unbekanntes oder kompromittiertes Programm wird abgelehnt oder ausgesetzt. In beiden Fällen wird das echte Token niemals preisgegeben.

# Was ist ein Refresh-Token, und warum ist der Verlust eines solchen besonders schädlich?

**Kurzantwort:** Ein Refresh-Token ist ein langlebiges Zugangsdatum, das verwendet wird, um im Hintergrund neue Access-Token zu erhalten, ohne dass sich der Benutzer erneut anmelden muss. Darauf beruht die Funktion „Angemeldet bleiben“. Da damit potenziell unbegrenzt neuer Zugriff erzeugt werden kann, kann ein gestohlenes Refresh-Token einem Angreifer dauerhaften Zugriff verschaffen, weit über das Ende Ihrer Sitzung hinaus.

## Access-Token vs. Refresh-Token

- **Access-Token (Sitzungs-Token):** kurzlebig, wird bei jeder Anfrage verwendet, läuft innerhalb von Minuten oder Stunden ab.
- **Refresh-Token:** langlebig, wird nur verwendet, um neue Access-Token anzufordern. Teilweise wochen- oder monatelang gültig.

Ein gestohlenes Access-Token läuft irgendwann ab. Ein gestohlenes Refresh-Token kann wiederholt verwendet werden, um neue Access-Token zu erzeugen – wodurch aus einem einmaligen Diebstahl dauerhafter Zugriff wird.

## Warum „Angemeldet bleiben“ ein Kompromiss ist

Um angemeldet zu bleiben, muss etwas Dauerhaftes auf dem Gerät gespeichert werden. Genau diese Bequemlichkeit macht das gespeicherte Token für einen Angreifer wertvoll. Browser- und App-Zugangsdatenspeicher befinden sich an standardisierten, vorhersehbaren Orten, sodass automatisierte Tools direkt darauf zugreifen.

## Schadensbegrenzung

Das Widerrufen eines Refresh-Tokens macht den damit erzeugbaren Zugriff ungültig – deshalb sind „Von allen Geräten abmelden“ und der Widerruf von Token nach jedem Verdacht auf eine

Kompromittierung so wichtig. Ein Widerruf ist jedoch eine Reaktion, die *nach* dem Diebstahl erfolgt.

Der Ansatz von Cyber Crucible besteht darin, das Auslesen von vornherein zu verhindern. Der Zugangsdatenspeicher ist ein geschützter Ort: Ein legitimes Programm, das dort unbefugt zugreifen will, erhält gefälschte Daten oder wird abgewiesen, während ein unbekanntes oder kompromittiertes Programm blockiert oder gesperrt wird. Das Refresh-Token selbst wird niemals preisgegeben.

# Was sind Browser-Cookies, und wie missbrauchen Angreifer sie?

**Kurze Antwort:** Cookies sind kleine Datenstücke, die eine Website in Ihrem Browser speichert, und einige davon sind Authentifizierungs-Cookies, die Sie eingeloggt halten. Deren Diebstahl ist funktional gleichbedeutend mit dem Diebstahl einer aktiven Sitzung — der Angreifer kann sie in seinen eigenen Browser laden und dabei als Sie erscheinen.

## Nicht alle Cookies sind gleich wichtig

- **Präferenz-Cookies** — Sprache, Design, Layout. Geringer Wert.
- **Analyse-Cookies** — Nutzungsmessung. Geringer Wert.
- **Authentifizierungs-/Sitzungs-Cookies** — belegen, dass Sie eingeloggt sind. **Hoher Wert.**

Angreifer interessieren sich fast ausschließlich für die dritte Kategorie.

## Warum Cookie-Diebstahl wirksam ist

Ein gestohlenes Authentifizierungs-Cookie löst in der Regel kein Login-Ereignis, keine Passwortabfrage und keine MFA-Aufforderung aus. Die Sitzung existiert bereits; der Angreifer legt lediglich einen Nachweis dafür vor. Aus Sicht der Anwendung sieht dies wie gewöhnliche fortlaufende Aktivität aus.

Browser speichern Cookies auf jedem Betriebssystem an bekannten, vorhersehbaren Speicherorten, sodass automatisierte Tools sie finden können, ohne etwas über den konkreten Rechner zu wissen.

## Warum der Browser ein zentraler Angriffspunkt ist

Der Browser bewahrt Cookies, gespeicherte Passwörter, OAuth-Token und den Verlauf an einem Ort auf, was ihn zum reichhaltigsten Identitätsziel auf den meisten Endpunkten macht. Cyber Crucible hat dies direkt beobachtet: Ab dem vierten Quartal 2023 stiegen automatisierte

Reaktionen auf Chrome-, Edge- und Chromium-Aktivitäten von null auf Tausende, mit dazugehörigen CVEs, die später von Google und Microsoft veröffentlicht wurden.

# Schutz

Der Browser-Anmeldedatenspeicher ist einer der geschützten Identitätsorte. Ein Prozess, der versucht, darauf zuzugreifen, wird im jeweiligen Kontext bewertet — ein kompromittierter Browser wird ganz anders behandelt als ein unversehrter, und in keinem der beiden Fälle erhält eine übergriffige Anfrage das echte Cookie.

# Was ist ein API-Schlüssel, und was passiert, wenn er gestohlen wird?

**Kurzantwort:** Ein API-Schlüssel ist eine geheime Zeichenfolge, die ein Programm – und nicht eine Person – identifiziert und autorisiert, wenn es einen Dienst aufruft. Ein gestohlener Schlüssel ermöglicht es einem Angreifer, Anfragen im Namen Ihrer Anwendung zu stellen, oft ohne Benutzerinteraktion, ohne MFA und ohne offensichtliches Anzeichen dafür, dass etwas nicht stimmt.

## Warum sie attraktive Ziele sind

- **Kein Mensch beteiligt.** Schlüssel sind für automatisierte, unbeaufsichtigte Nutzung konzipiert, sodass keine MFA-Aufforderung stört.
- **Häufig weit gefasste Berechtigungen.** Schlüssel werden oft mit mehr Rechten ausgestellt, als für die Aufgabe erforderlich ist.
- **Langlebig.** Viele werden nie rotiert, sodass ein heute gestohlener Schlüssel möglicherweise noch Monate später funktioniert.
- **An vorhersehbaren Orten gespeichert.** Konfigurationsdateien, Umgebungsvariablen und Zugangsdatenspeicher – allesamt Orte, die Automatisierung durchsuchen kann.

## Typische Folgen

Je nachdem, wofür der Schlüssel autorisiert: Lesen oder Exportieren von Kundendaten, Versenden von Nachrichten im Namen Ihrer Organisation, Ausgeben von Geld über ein Cloud-Konto oder das Vordringen in andere verbundene Systeme.

Da die Anfragen ordnungsgemäß authentifiziert sind, erscheinen sie in Protokollen meist als normaler Anwendungsverkehr. Die Entdeckung erfolgt in der Regel erst deutlich später, etwa durch eine Rechnung oder ein Audit.

## Verringerung der Angriffsfläche

Beschränken Sie Schlüssel eng auf ihren Zweck, rotieren Sie sie regelmäßig, und halten Sie sie aus der Versionsverwaltung fern. Dies sind sinnvolle Maßnahmen, die jedoch den möglichen Schaden

begrenzen, statt den Diebstahl zu verhindern.

Cyber Crucible setzt beim Diebstahl selbst an. Speicherorte für Zugangsdaten und Konfigurationen werden geschützt, und ein Programm, das darauf zugreift, wird im jeweiligen Kontext bewertet: gefälschte Daten oder Verweigerung bei legitimer, aber übermäßiger Zugriffsanforderung, Ablehnung oder Aussetzung bei unbekannten oder kompromittierten Prozessen. Der Schlüssel wird in beiden Fällen nicht preisgegeben.

# Wo speichert Windows Passwörter und Anmeldeinformationen?

**Kurze Antwort:** An mehreren vorhersehbaren Orten – im Windows-Anmeldeinformationsverwaltung (Credential Manager), in den Passwortspeichern der Browser, in der VPN-Client-Konfiguration und, auf Domänencontrollern, in der Active-Directory-Datenbank (NTDS). „Vorhersehbar" ist dabei das entscheidende Wort: Automatisierte Angriffe verlassen sich darauf, dass diese Orte auf jedem Rechner gleich sind.

## Die wichtigsten Speicherorte

- **Windows-Anmeldeinformationsverwaltung (Credential Manager)** — gespeicherte Anmeldedaten für Netzwerkressourcen und Anwendungen.
- **Passwortspeicher der Browser** — Chrome, Edge und Firefox speichern gespeicherte Anmeldeinformationen jeweils an bekannten Profilpfaden.
- **VPN-Client-Anmeldeinformationen** — Benutzernamen, Passwörter und schlüsselbasiertes Authentifizierungsmaterial.
- **NTDS (Active Directory)** — auf einem Domänencontroller die Anmeldeinformationsdaten für die gesamte Domäne. Das wertvollste Ziel im Netzwerk.
- **Anwendungsspezifische Speicher** — Messaging- und Collaboration-Tools, Dateiübertragungs-Clients und Gaming-Plattformen unterhalten jeweils eigene Speicher.

## Warum Vorhersehbarkeit der entscheidende Punkt ist

Angreifer wissen nichts über Ihre spezifische Umgebung. Ihre Automatisierung ist für Pfade geschrieben, die überall existieren — `C:\Users\[Username]`, Standardordner für Anwendungsdaten, Standard-Laufwerksbuchstaben. Ein Skript muss Ihr Netzwerk nicht verstehen; es durchläuft einfach bekannte Pfade.

Das ist eine Schwäche in ihrer Methode. Wenn genau diese Orte überwacht werden, wird die eigene Zuverlässigkeitsanforderung des Angreifers zu dem Auslöser, der ihn entlarvt.

# Was Cyber Crucible damit macht

Diese Speicherorte sind die überwachten Einstiegspunkte für Identitätsdiebstahl. Bei einem Programm, das darauf zugreift, wird die Absicht in weniger als 200 Millisekunden bewertet — und es wird angehalten, falls diese Absicht bösartig ist, noch bevor eine Anmeldeinformation entwendet werden kann.

# Was sind Kryptowährungs-Wallet-Schlüssel, und warum ist ihr Diebstahl unumkehrbar?

**Kurze Antwort:** Ein Wallet-Schlüssel ist der private kryptografische Schlüssel, der Ausgaben von einer Kryptowährungsadresse autorisiert. Wer ihn besitzt, kontrolliert die Gelder. Der Diebstahl ist praktisch unumkehrbar, da Blockchain-Transaktionen nicht rückgängig gemacht werden können und es keine Institution gibt, an die man sich wenden könnte.

## Warum sich dies von einer Bank unterscheidet

Eine betrügerische Kartenbelastung kann angefochten und von einem Kartenaussteller rückgängig gemacht werden. Eine mit Ihrem privaten Schlüssel signierte Kryptowährungstransaktion ist per Definition gültig — das Netzwerk kann einen Diebstahl nicht von einer legitimen Überweisung unterscheiden, und es gibt keine zentrale Instanz, die sie rückgängig machen könnte.

Das macht Wallet-Schlüssel ungewöhnlich attraktiv: Der Gewinn ist sofort verfügbar, endgültig und schwer nachzuverfolgen.

## Worauf Angreifer es abgesehen haben

- **Private Schlüssel und Seed-Phrasen**, die in Anwendungsdateien der Wallet gespeichert sind
- **Wallet-Inhalte und Transaktionsaufzeichnungen**, die dazu dienen, die wertvollsten Opfer zu identifizieren
- **Passwörter, die die Wallet schützen**, oft auch anderswo wiederverwendet

Desktop-Wallet-Software speichert diese Daten an vorhersehbaren Speicherorten der Anwendungsdaten — dasselbe Muster, das jeden anderen Anmeldedatenspeicher für Automatisierung zugänglich macht.

## Schutz

Wallet-Dateien gehören zu den Identitätsspeicherorten, die Cyber Crucible überwacht. Da die Reaktion in der Sperrung im Moment des bösartigen Zugriffs besteht, wird der Diebstahl verhindert statt erst nachträglich erkannt — was hier mehr als fast überall sonst von Bedeutung ist, da es keinen Wiederherstellungsweg gibt, sobald Gelder transferiert wurden.

> Siehe auch: "*Welche Kryptowährungs-Wallets sind geschützt?*" für die spezifischen, derzeit abgedeckten Anwendungen.

# Was ist schlüsselbasierte Authentifizierung, und warum verschaffen gestohlene Schlüssel dauerhaften Zugriff?

**Kurzantwort:** Schlüsselbasierte Authentifizierung weist die Identität mittels eines kryptografischen privaten Schlüssels statt eines Passworts nach — verwendet von SSH, vielen VPNs und Dateiübertragungs-Clients. Ein gestohlener privater Schlüssel gewährt Zugriff ohne jegliche Passwort- oder MFA-Abfrage, und da Schlüssel nur selten rotiert werden, kann dieser Zugriff über sehr lange Zeit bestehen bleiben.

## Funktionsweise

Sie besitzen einen privaten Schlüssel; der Server besitzt den dazugehörigen öffentlichen Schlüssel. Beim Verbindungsaufbau stellt der Server eine Challenge, die nur der private Schlüssel beantworten kann. Der Schlüssel selbst überquert das Netzwerk niemals.

Dies ist tatsächlich stärker als Passwörter — es widersteht Erraten, Phishing und Wiederverwendung. Seine Schwäche ist anders gelagert: **Alles hängt davon ab, dass die private Schlüsseldatei geheim bleibt.**

## Warum ein gestohlener Schlüssel so dauerhaft ist

- **Keine Passwortabfrage.** Der Schlüssel *ist* die Anmeldeinformation.
- **Häufig keine MFA.** Schlüsselbasierte Authentifizierung steht oft für sich allein, insbesondere bei automatisierten Systemen.
- **Selten rotiert.** Passwörter laufen nach einem festen Zeitplan ab; Schlüssel bestehen häufig über Jahre.
- **Vorhersehbare Speicherorte.** Standardverzeichnisse und Konfigurationspfade von VPN-Clients.

Ein gestohlener Schlüssel ist eine stille, dauerhafte Hintertür, die keine fehlgeschlagenen Anmeldungen erzeugt und vollständig legitim erscheint.

# Schutz

VPN- und schlüsselbasiertes Authentifizierungsmaterial gehören ausdrücklich zu den geschützten Identitätskategorien. Der Zugriff wird auf Kernel-Ebene bewertet, sodass ein Programm, das Schlüsseldateien abgreift, gestoppt wird — abgewiesen oder mit gefälschten Daten versorgt, wenn es sich um ein vertrauenswürdiges Programm handelt, das seine Berechtigungen überschreitet, zurückgewiesen oder ausgesetzt, wenn es unbekannt oder kompromittiert ist — bevor der Schlüssel den Rechner verlässt.

# Warum erfordert der Schutz von Identitätsdaten Zugriff auf Kernel-Ebene?

**Kurze Antwort:** Weil der Diebstahl in dem Bereich zwischen der Anforderung von Anmeldedaten durch ein Programm und deren Bereitstellung durch das Betriebssystem stattfindet. Nur eine Verteidigung, die auf Kernel-Ebene arbeitet — unterhalb der Anwendungen und Bibliotheken, die ein Angreifer manipulieren kann — kann diese Anfrage erkennen, ihre Absicht beurteilen und sie rechtzeitig stoppen.

## Die drei Anforderungen

**1. Den Zugriff erkennen, während er geschieht.** Identitätsdiebstahl äußert sich nicht dadurch, dass eine Datei auf der Festplatte erscheint; es handelt sich um einen Lesevorgang. Tools, die nach schädlichen Dateien suchen, sehen nichts, da bei einem modernen Infostealer nichts geschrieben wird. Der zu erfassende Vorgang ist der Zugriffsversuch selbst.

**2. Schnell genug entscheiden.** Automatisierte Tools können innerhalb von Sekunden eindringen, Daten sammeln und sich selbst löschen. Jede Architektur, die Telemetriedaten an einen Cloud-Dienst sendet und auf ein Urteil wartet, hat das Rennen bereits verloren. Die Entscheidung muss lokal erfolgen, innerhalb von Millisekunden.

**3. Nicht von dem abhängen, was der Angreifer kontrolliert.** Sicherheitstools, die auf Betriebssystembibliotheken angewiesen sind, können blind werden, wenn ein Angreifer diese Bibliotheken im Arbeitsspeicher kompromittiert — der Agent läuft weiter, meldet jedoch nichts. Cyber Crucible wurde bewusst von den Windows-Bibliotheken entkoppelt, sodass ein kompromittiertes Betriebssystem es nicht zum Schweigen bringen kann.

## Die Konsequenz

Der Betrieb auf Kernel-Ebene ist es, der alle drei Anforderungen gleichzeitig ermöglicht: Sichtbarkeit des Zugriffsversuchs, eine lokale Entscheidung in unter 200 Millisekunden und Unabhängigkeit von einem möglicherweise kompromittierten Betriebssystem.

Deshalb funktioniert der Schutz auch ohne jegliche Konnektivität — abgeschottet (air-gapped), offline oder auf See —, da nichts das Gerät verlassen muss, damit eine Entscheidung getroffen werden kann.