HUNTRESS

The Rise of Infostealers and Session Hijacking

Learn how infostealers and session hijacking open the door for stealthy enterprise-wide intrusions and how to protect yourself against unwanted access attempts.

Table of Contents

Introduction

Beyond the login: the next generation of credential theft compromise

What started as a shift in credential compromise tactics in 2020 has evolved into a far more dangerous identity threat: **session hijacking**. This stealthy initial access technique has overtaken traditional targeting of stolen usernames and passwords, letting hackers slip silently into authenticated stolen sessions.

Session hijacking isn't just a threat on the horizon—it's a reality in your environment now. Resetting a password no longer guarantees eviction of an attacker: many systems issue bearer tokens (session cookies, Json Web Tokens (JWT), or long-lived OAuth tokens) that remain valid until the service revokes them or the token expires. Token behaviors vary widely between services, so password resets alone are often insufficient unless the platform explicitly revokes sessions or refresh tokens. This puts your identities at risk, giving attackers an open window of ongoing unauthorized access. The modern authentication we rely on has been flipped into a security liability.

This means traditional defenses like MFA and perimeter security aren't enough. Organizations must treat session data as privileged access, implement short-lived tokens, enforce behavioral access controls, and beef up developer systems with the same priority as production environments.

In this ebook, we'll dive into the risks of infostealer malware and session hijacking attacks. By understanding this cybercriminal ecosystem, you'll be able to wreck these advanced cyber threats against your organization's endpoints and identities.

A Look Inside the Infostealer Ecosystem

Infostealer malware: the gateway to session hijacking

Threat actors use infostealer malware to collect credentials, financial information, and sensitive data. This data is sold on dark web underground marketplaces to buyers who target individuals or enterprises with session hijacking, ransomware, extortion, and more.

According to the Huntress 2025 Cyber Threat Report, infostealers were the most commonly observed threat category throughout 2024, accounting for 24% of all observed incidents across multiple industries.

Here's a look at the types of data infostealers collect:

Identity

- Email, autofill data, and credentials
- Cookies, SSO, sessions, and JWT
- 2FA, MFA, OTP keys

Financial

- Credit card or digital payment data
- Cryptocurrency wallets, recovery and seed phrases, digital currency

Corporate IT

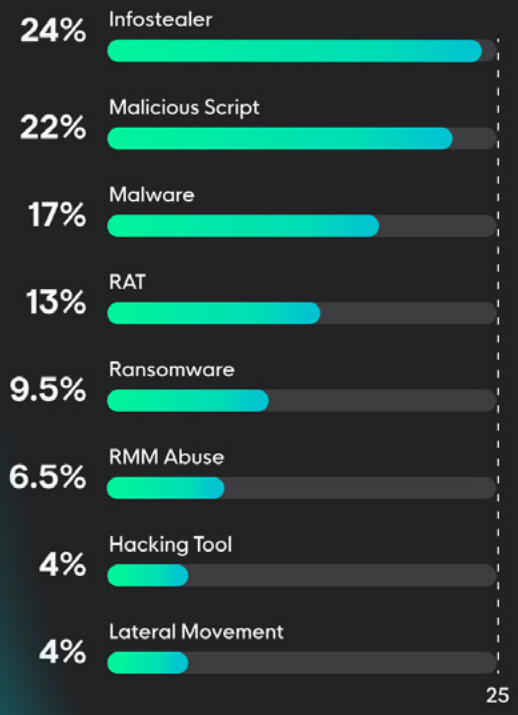
- SAML, SSO, VPN, SSH, RMM, RDP, Slack, and FTP credentials
- API keys and cloud service credentials

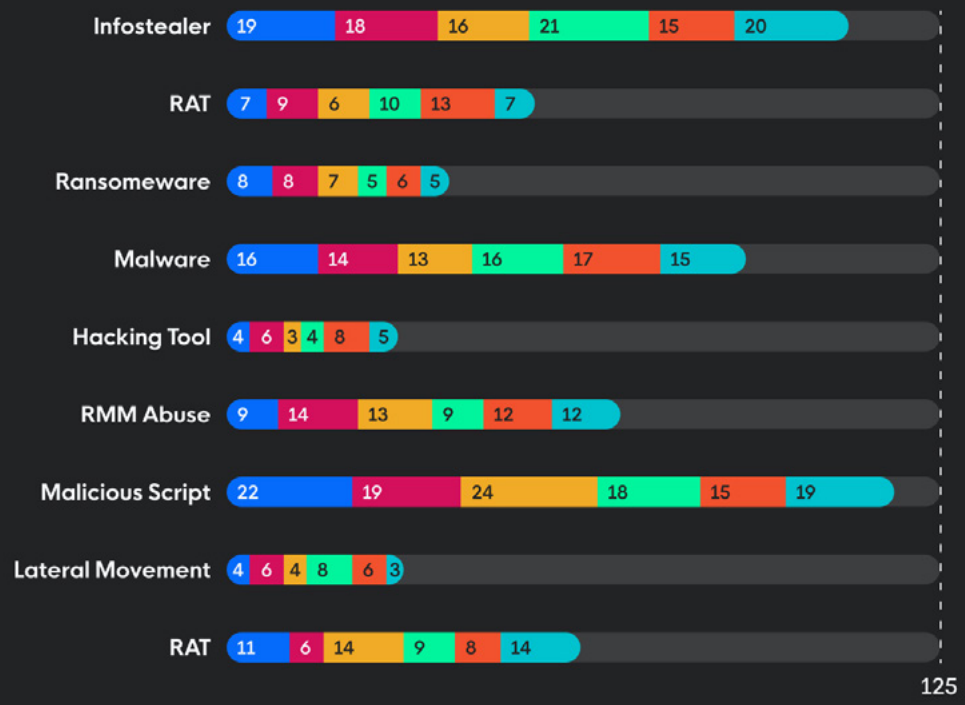
Other sensitive data

- Personally Identifiable Information (PII), Personal Health Information (PHI), geolocational data, and system fingerprints
- Installed applications, bookmarks, and active processes
- Chat history, photos, movies, documents, and user-created content

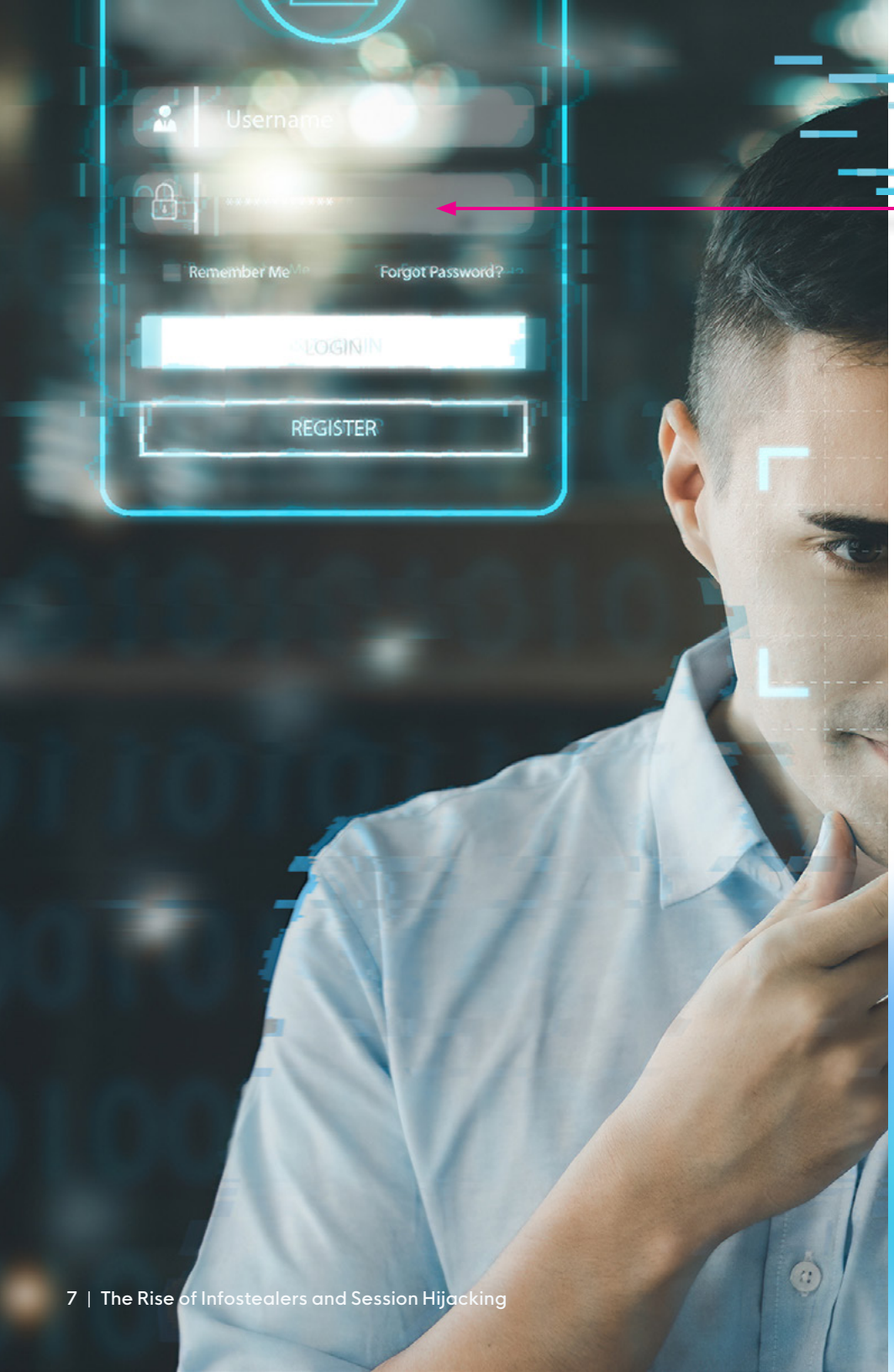
HUNTRESS

Frequency Of Threats Overall For 2024





HUNTRESS



Infostealer malware plays a key role in session hijacking attacks. It's the supply line for stolen session tokens or cookies, which attackers use to trick targeted servers into giving them unauthorized access.

Here's what this means for defenders:

A stolen session token is functionally equivalent to holding an active key to the victim's account:

- The attacker doesn't need the original password after authentication
- MFA isn't re-prompted, and no login alerts are triggered because session cookies are treated as valid proof of identity

Even if a user resets their password, many session tokens are still valid unless they're explicitly revoked or expired by security policies. This gives attackers a dangerous window of persistence.

HUNTRESS

How Infostealers Went Mainstream

How infostealers went mainstream

The demand for stolen credentials from ransomware affiliates, initial access brokers (IAB), and corporate espionage surged in 2020, prompting infostealer developers to quickly level up their business plans, tapping into new illicit cyber opportunities. They shifted their focus from crude key loggers to modular malware-as-a-service platforms like Raccoon, Vidar, and Lumma, capable of extracting browser-stored credentials, session tokens, auto-fill data, and even MFA seeds.

Malware deployment methods have evolved, too. Instead of predictable phishing attacks, infostealer malware upped its game to hide in cracked software, YouTube tutorials, SEO-poisoned downloads, and fake updates. This shift fed the rise of a thriving black market economy.

During the rapid shift to work-from-home (WFH) and bring-your-own-device policies in 2020, coupled with a lack of endpoint detection and response (EDR) tools, businesses struggled to protect themselves against the growing infostealer malware threat.

- **2020-2021:**
  The WFH infostealer gold rush
- **2022-2023:**
  SaaS infiltration at scale
- **2024:**
  The developer pipeline breach

• 2020-2021
The WFH infostealer gold rush:

The WFH era, fueled by the COVID-19 pandemic, increased the supply and demand for stolen credentials and offered new collection and distribution opportunities for infostealer developers and operators.

Collection:

- VPN clients (e.g., Cisco AnyConnect, FortiClient), RDP clients, and remote meeting apps (Zoom, Microsoft Teams)

- Discord token theft, which was exploited to pivot inside development teams, using social engineering within chat platforms

Distribution:

- Decentralized and encrypted communication channels, like Telegram, ramped up infostealer log distribution. As supply and demand suddenly increased, it became faster and harder to trace logs being bought and sold in these channels.

2022-2023
SaaS infiltration at scale:

As enterprises migrated to Software-as-a-Service (SaaS) platforms, infostealer developers and operators found a new targeting opportunity: **persistent session hijacking.**

The transition away from traditional password theft started to show up in attacks against identity management platforms like Okta, OneLogin, and Ping Identity—gateway access points to a wide range of corporate services. Instead of focusing solely on login credentials, attackers began targeting the underlying mechanisms that maintain active sessions across services' session tokens, authentication cookies, and other dynamic credential artifacts.

This means targeting local files like .json, .sqlite, .ldb, and .dat, which are found in the user directories of Chromium-based browsers (Chrome, Brave, and Edge) and communication platforms like Slack, Discord, and Microsoft Teams. These files contain everything from refresh tokens to cookie stores, and even partially encrypted session variables.

Identity targeting was moving at warp speed and quickly outpacing traditional defenses.

.json	Frequently used by browsers and extensions (like Chromium-based profiles) to store configuration, login state, tokens, and sometimes autofill data in a structured format. Infostealers parse these files for saved logins and cookies.
.sqlite	SQLite databases are the default storage format for many applications, including browsers (e.g., Firefox's logins.sqlite) and messaging apps. They often store usernames, passwords (sometimes encrypted), and message history.
.ldb	LevelDB files are used by Chromium-based browsers, Metamask, and Discord to store local session tokens and cached login states. Infostealers targeting .ldb files hijack active sessions without credentials.
.dat	A generic extension often used by applications, like Telegram's map0.dat, to store encrypted or raw binary data. Infostealers scrape .dat files for app state, tokens, or cached login sessions.

2024
The developer pipeline breach:

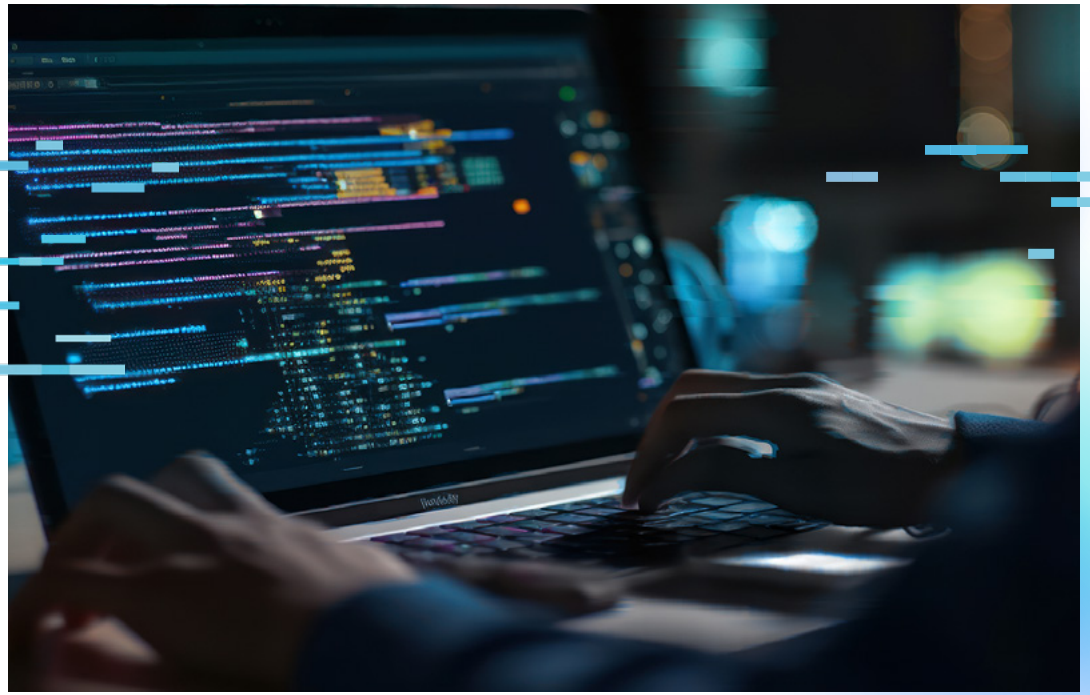
The battleground moved upstream. Your perimeter wasn't the only thing at risk—your code was too. By 2024, infostealer malware reached a new frontier: the developer pipeline.

This new cyber threat trend was a pivot in attackers' targeting focus:

- From consumer applications to corporate infrastructure environments
- From static passwords to dynamic, powerful, authenticated sessions

What started as opportunistic credential theft in consumer applications grew into a targeted, methodical extraction of CI/CD tokens, GitHub credentials, and AWS Identity and Access Management (IAM) sessions, which are all found in developer logs, build artifacts, and misconfigured environment files.

Infostealers began scanning for artifacts generated by automation tools like Jenkins, GitLab Runners, CircleCI, and GitHub Actions, because these tools often include persistent API tokens, access keys, and temporary AWS credentials issued via IAM roles. After exfiltration, these stolen identity secrets give attackers immediate and deep access to a targeted environment, bypassing MFA entirely.



How Attackers Abuse Developer Credential Access

Unauthorized GitHub access	Unauthorized AWS IAM Sessions
Plant back doors in code	Exfiltrate data in S3 buckets
Steal intellectual property	Impersonate legitimate services
Poison packages	Hijack resources for cryptomining or password cracking
Hijack internal tools	

Instead of phishing emails to developers or brute-forcing login pages, attackers replayed these stolen tokens using tools like OpenBullet or StealyBot to sneak into code repositories, cloud instances, and ticketing systems like Jira, Confluence, and ServiceNow, completely under the radar.

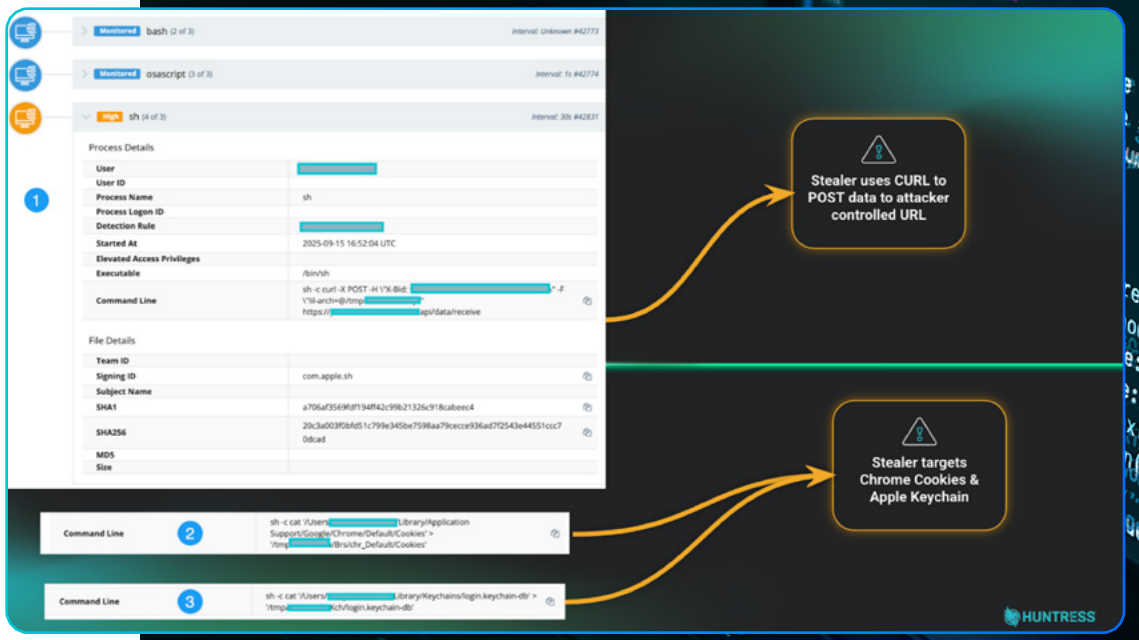
This subtle-but-powerful technique camouflages attackers as they move laterally across a company's most sensitive infrastructure. It gives them an initial foothold through the **code layer itself**, not IT or end-user devices, making it much tougher to spot.

Once this happens, the damage spirals fast, especially when secrets are shared across microservices or CI/CD stages without strict privilege separation. For many, the fallout is financial—delivered in an oversized cloud invoice.

During this critical era, leaked 1Password vault data intended for internal credential management also surfaced on dark web black markets. These types of breaches give attackers unauthorized administrative control over internal systems, including the ability to access audit logs, reset user credentials, and create persistent accounts.

Dubbed in some security circles, "The Developer Pipeline Breach," this period highlights an uncomfortable reality: while MFA greatly reduces risk for credential-based logins, it does not protect against theft of bearer tokens or secrets extracted from developer environments. When session tokens or bearer credentials are stolen directly from logs, temp files, or browser caches, the second factor is never invoked. The security model fails silently.

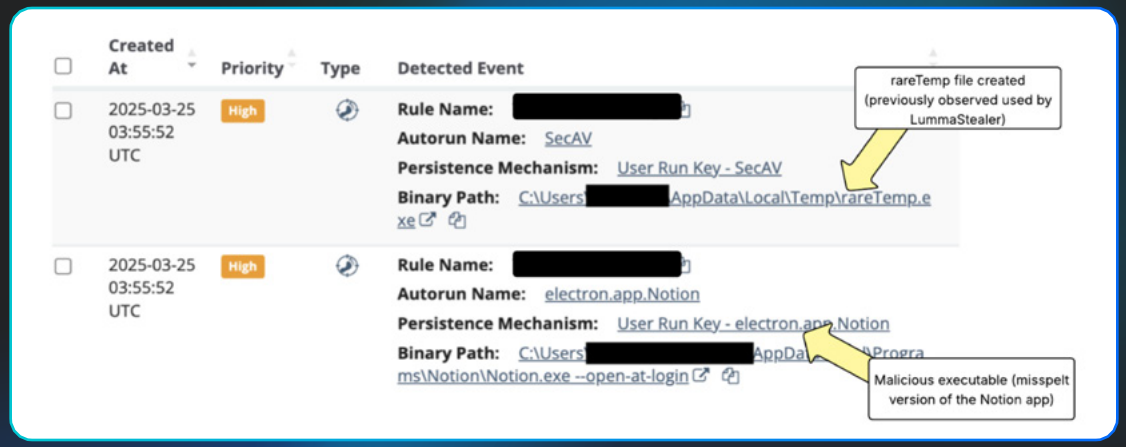
1Password vault data
- VPN keys
- Database logins
- Slack tokens
- Encryption passphrases



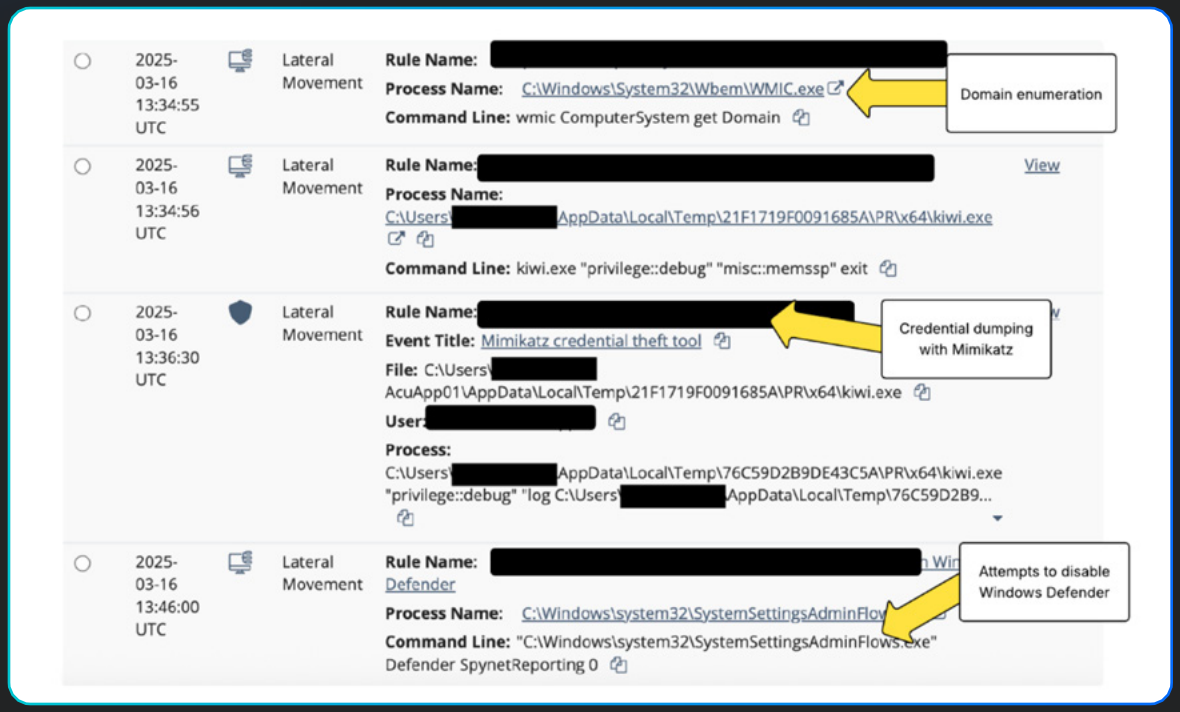
Tradecraft example of MacSync macOS infostealer

The takeaway is clear: To protect against infostealer malware, endpoints must be hardened as rigorously as production systems.

HUNTRESS



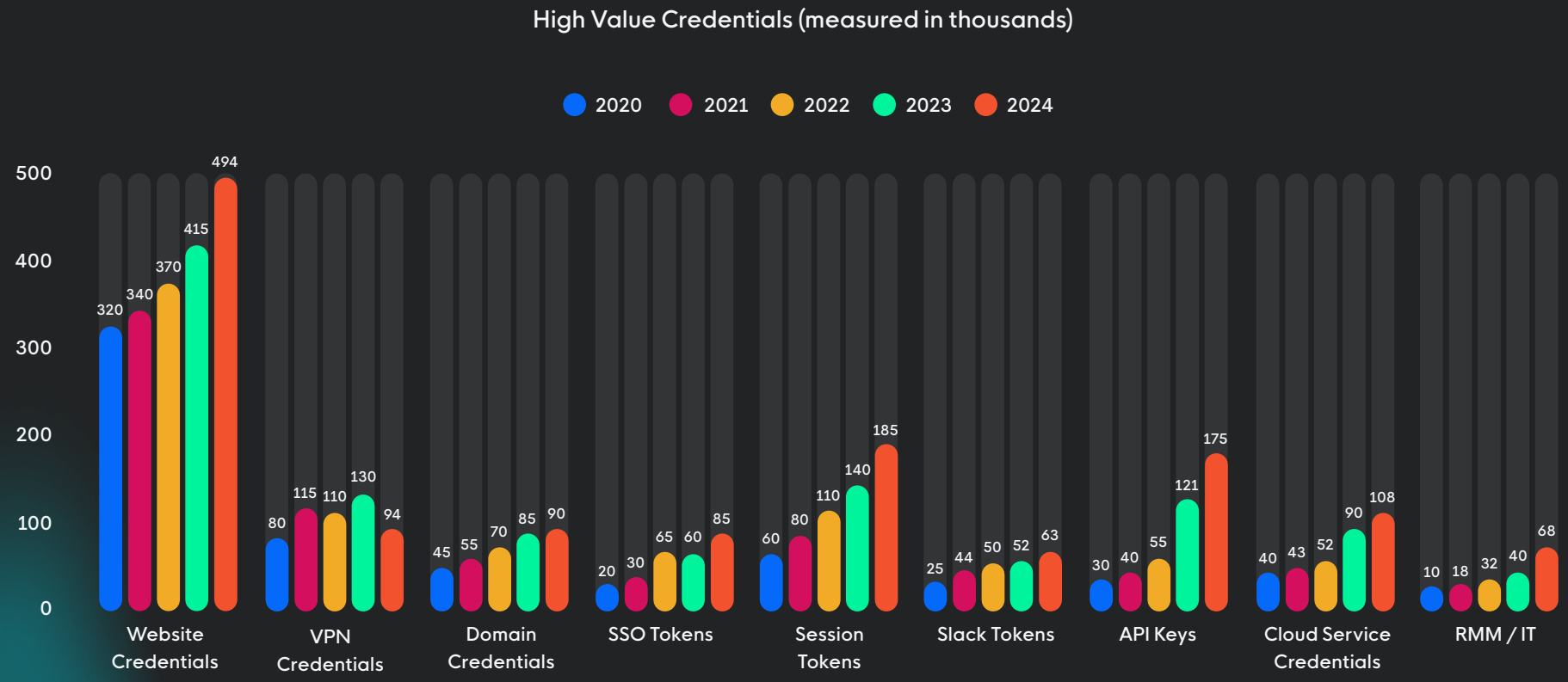
Luma Stealer tradecraft example



Amadey infostealer tradecraft example

HUNTRESS

Types of Stolen Credentials on Dark Web Marketplaces



This chart breaks down the different types of high-value stolen credentials on dark web marketplaces from 2020-2024.

HUNTRESS

Time Period	Key Targets and Attacker Techniques	Impact on Targeted Environments
WFH Infostealer Gold Rush: 2020-2021	VPN clients, RDP clients, remote meeting apps (e.g., Zoom, Microsoft Teams)	Increased risk to remote access and collaboration tools
	Spike in Discord token theft, exploited for development team pivoting	Enabled lateral movement within development environments
	Rise in Telegram-based exfiltration for criminal distribution	Facilitated rapid log dissemination to buyers
SaaS Infiltration at Scale: 2022-2023	Okta, OneLogin, PingIdentity cookies and tokens	Direct access to identity management systems
	Extraction of targeted files from Chromium browsers	Shift to persistent sessions, bypassing passwords
	Attacks moved from passwords to session replay	Enhanced ability to bypass MFA and EDR visibility
Developer Pipeline Breach: 2024	CI/CD tokens, GitHub credentials, AWS IAM sessions in logs	Enabled lateral movement into code repos, cloud infrastructure, and ticketing systems
	MFA-bypassed access via stolen sessions	Increased risk of supply chain attacks and data breaches
	Corporate 1Password exports surfaced in resale markets	Exposed sensitive credential management data

The infostealer marketplace: a one-stop shop for unauthorized access

Session hijacking requires an upfront investment, but the ROI can be big, depending on how an attacker plays their cards.

Cybercriminals who use session hijacking techniques often buy infostealer logs from IABs. This gives them a better (and faster) chance of successfully getting initial access to the targeted environment.

Let's break down how the infostealer market works.

Typical infostealer logs are $5 to $25, with several factors affecting the price:

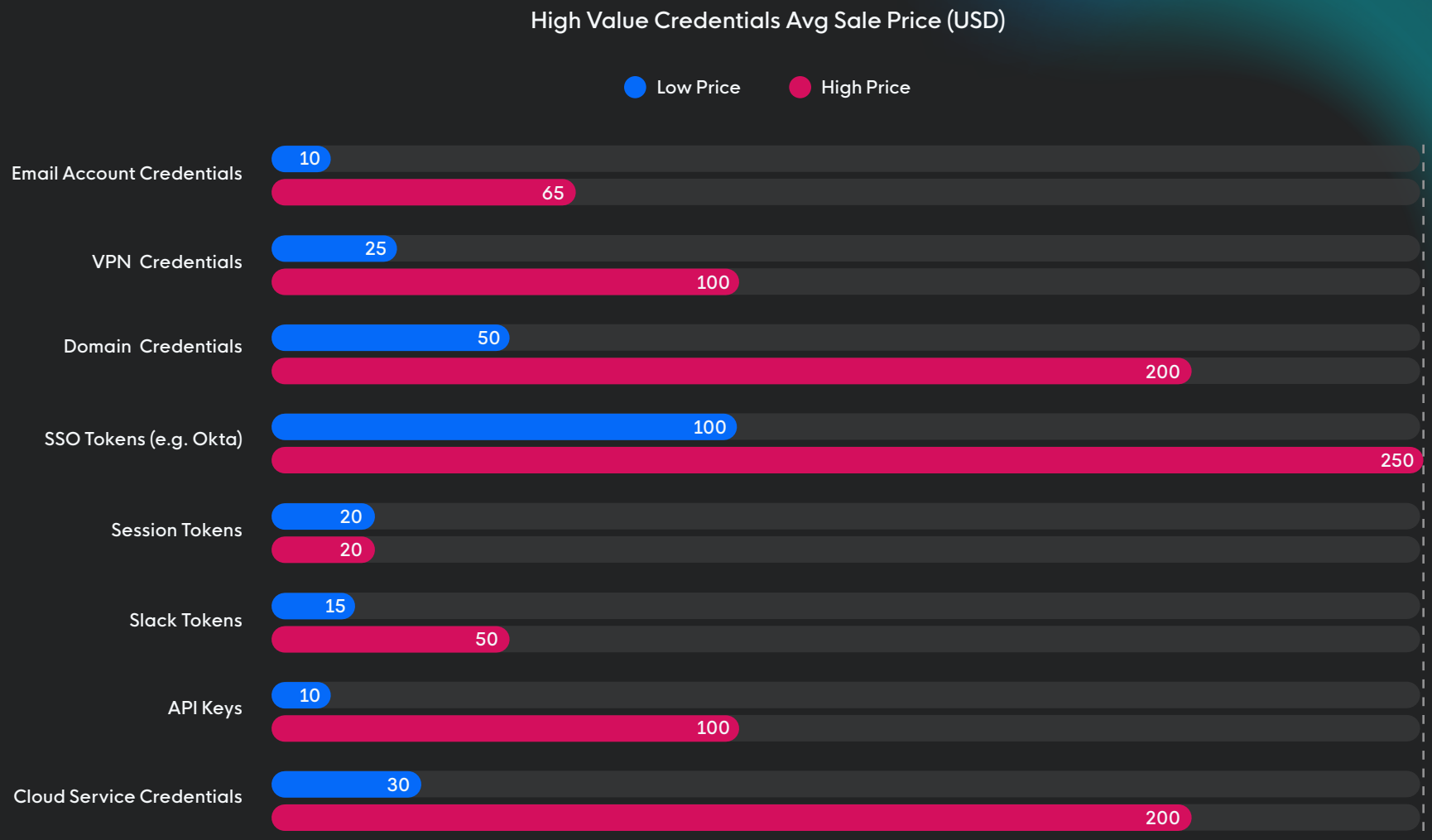
- Data quality, with newer data selling for a premium
- Geolocation of the victim
- Data type: VPN, admin panels, and cloud content demand a higher price

Infostealer logs with Fortune 500 domain credentials, valid Microsoft 365 sessions, Slack or Okta tokens, or access to developer tools (GitHub, Jira, AWS CLI sessions) are more valuable, ranging from $100 to more than $500, depending on exclusivity. Slack tokens are often the most prized credentials and have their own marketplaces, as attackers glamorized them in 2023 for compromising major companies such as Riot, EA Games, and several casinos.

Top-tier IABs are the high-end middlemen of the cybercrime black market economy. They purchase high-value infostealer logs or do targeted intrusions themselves, then resell verified, curated access to other threat actors, most often ransomware affiliates, data extortion crews, or industrial espionage clients. These exclusive brokers often charge several thousand dollars for exclusive top-tier one-off, high-stakes unauthorized access.

HUNTRESS

Average Prices of Stolen Credentials*

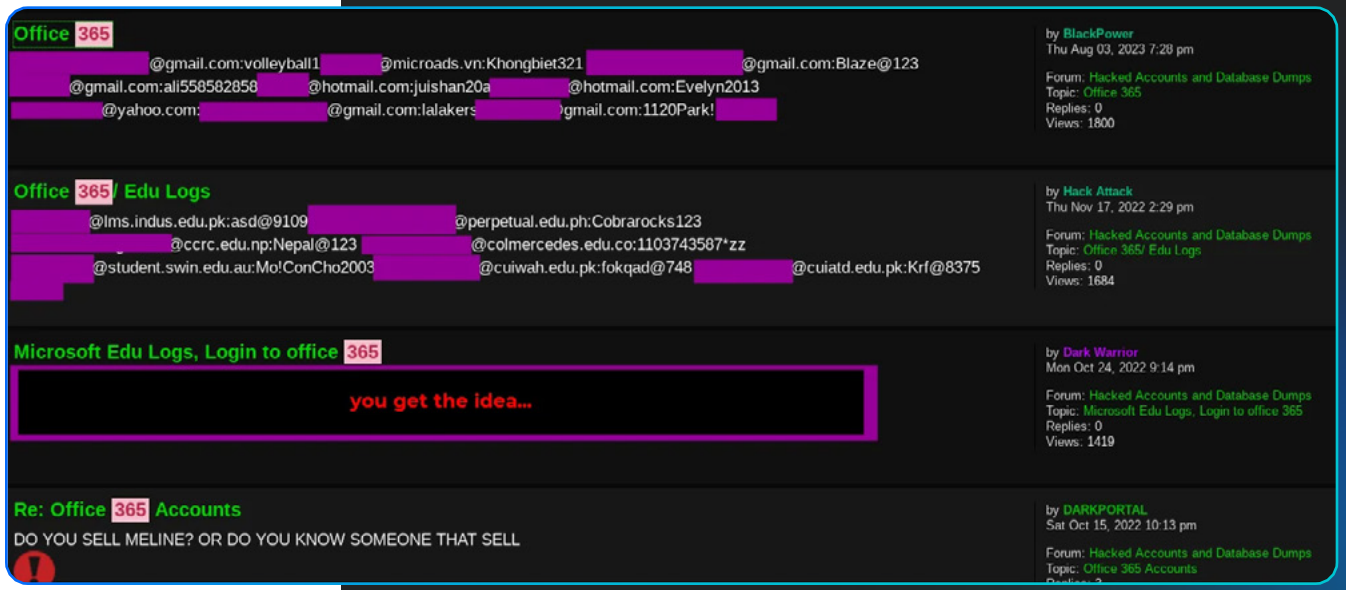


*High Value Confirmed Credentials Averages Across Multiple Markets

HUNTRESS

The infostealer ADDON market

The illicit infostealer economy has grown into a powerful ecosystem of modular, upsell-ready tools and data packs designed to maximize an attacker's payday for each individual breach. After an infostealer log or compromised machine is collected, the sellers bolt on additional services, tools, or specialized data dumps to scale their operations, deepen access, or tailor attacks to high-value targets.



Example of dark web forum selling stolen credentials

The following types of add-ons are available:

Typical add-ons:

- Discord Tokens: $5-$20 (depending on Nitro status or mod/admin role)
- Slack/Mattermost Tokens: $25-$75
- Google Workspace / Microsoft 365 Cookies: $50-$200+
- GitHub Personal Access Tokens (PATs: $50-$300
- AWS IAM Session Tokens: $100-$500
- Cloudflare / Okta / PingIdentity Session Keys: $100-$800+
- Browser fingerprint bundles to replay sessions without triggering security challenges (Price varies based on data)

HUNTRESS

Developer/DevOps environment extract add-ons:

- .env dumps from Node.js or React apps: $25 per file
- Jenkins credential files: $100+
- .npmrc and .pypirc (with publish tokens): $50-$100
- .git-credentials, .aws/config, SSH private keys: $50-$300
- Full .git folders (entire repo + commit history): $100+

Password Manager Vault add-ons:

- 1Password export JSONs: $300-$1,000
- Bitwarden vaults: $200-$700
- KeePass databases (.kdbx): $100-$500|
- Browser vault exports (Chrome, Edge): $25-$75

It doesn't stop there. Attackers also monetize automation and verification services:

- Log Checkers (RedLine/Stealy validators): $100-$300
- RDP Scanner Bots (auto-test credentials across IP ranges): $50/month
- OpenBullet Configs (pre-built for Shopify, AWS, GitHub, etc.): $20-$150 each
- Stealer deployment panels and encrypting services: $200-$600 monthly
- Telegram bots that sort logs into access types: ~$100
- Company Lookups (Clearbit-style): tags logs with domain reputation or industry
- Geo-IP Enrichments: locates the target geography
- Credential Health Checks: flags MFA/2FA protection or recent login timestamps
- Dark Web Cross-Reference Tools: show if the target appears in other breaches

The infostealer upsell market

IABs are always looking for new ways to maximize their data offerings and profits. A typical IAB upsell flow looks like this:

- Buy a raw infostealer log for about $10. It's an unsorted and unverified dump with Chromium browser history, cookies, saved passwords, localStorage data, autofill, and clipboard contents.

- Run the log through an automated or semi-automated tool to parse valuable data like:
  - Valid Slack tokens
  - .env file with PostgreSQL + Stripe keys
  - GitHub PAT (Personal Access Token)
  - .aws/credentials file with active IAM role
  - Session cookies for Google Workspace and Jira

At this point, the IAB might sell the parsed credentials for around $200 to $400, or they can hack into the targeted company with the parsed data, because this can lead to more valuable information that can be bundled for a bigger price tag.

Let's say they get lucky and find a 1Password Export Vault with credentials for the following:

- CRM (like HubSpot)
- SFTP server
- Corporate email

HUNTRESS

With a little extra time and hacking, the IAB now sells a "Developer Access Bundle" for $1,000 to ransomware affiliates, extortion crews, and phishing-as-a-service groups, with the following credential accesses:

- GitHub token
- AWS session
- CRM + email creds
- Vault export
- Valid Slack token

What started as a $10 investment led to a 10,000% ROI for a bad guy on a dark web forum!

Step	Cost	Upsell Value
Initial Log Purchase	$10	N/A
Token Access	Free (included)	$200 to $400
Vault Extract	Free (from log)	$300 to $800
Resale Bundle	Time	$1,000+
Total ROI	$1,000+	10,000%+

Session Hijacking: Intrusions Have Never Been Easier (or Cheaper)

What is session hijacking?

When you log in to a website, your computer and the site need to remember who you are as you click around. They do this with a 'session' that tracks your activity with cookies or tokens.

Session hijacking happens when an attacker steals a session and uses it for unauthorized access. The attacker uses stolen session information to pretend to be you and access your private data or online accounts without passwords or triggering MFA—exactly what makes session hijacking a serious threat.

Several high-profile incidents have been reported to involve stolen sessions or infostealer-sourced artifacts. Public reporting suggests these techniques played a role in the following cases:

Breach	Date	Summary
EA Games	June 2021	Attackers purchased a Slack cookie from an infostealer logs market, replayed the session, and pivoted into EA's code repositories.
Twilio	August 2022	A social engineering attack led to credential theft. The attacker reused session data for Okta and other SSO providers, likely sourced from infostealer logs.
LastPass	August-November 2022	The initial compromise was traced to a developer's stolen credentials and session data, exfiltrated via infostealer malware.

Breach	Date	Summary
Okta (via Lapsus$)	March 2022	Lapsus$ used stolen credentials and active sessions from a third-party contractor's endpoint, likely acquired via stealer logs.
Riot Games	January 2023	Internal systems were breached after infostealer logs with valid sessions gave access to development tools and Slack.
CircleCI	January 2023	Developer credentials were stolen via an infostealer. The attackers accessed customer environment variables and secrets.
Nvidia	February 2022	Lapsus$ used employee credentials, likely from infostealer logs, to exfiltrate gigabytes of sensitive data.
Telefónica	January 2025	Telefónica was breached by the HELLCAT ransomware group, which used infostealer logs from late 2024 to gain access to the provider's Jira system
Jaguar Land Rover	March 2025	The HELLCAT ransomware group used compromised credentials from a third-party contractor's Jira/VPN login credentials by an infostealer to later use to gain access to Jaguar Land Rover, causing major operational disruptions
Samsung	March 2025	Samsung Germany's ticketing system was breached, likely due to third-party provider Spectos, which had been previously compromised by infostealer malware.
Royal Mail	April 2025	A data dump of customers' PII and mailing lists originating from the Royal Mail compromise has been attributed to the third-party Spectos infostealer dump

How attackers hijack sessions

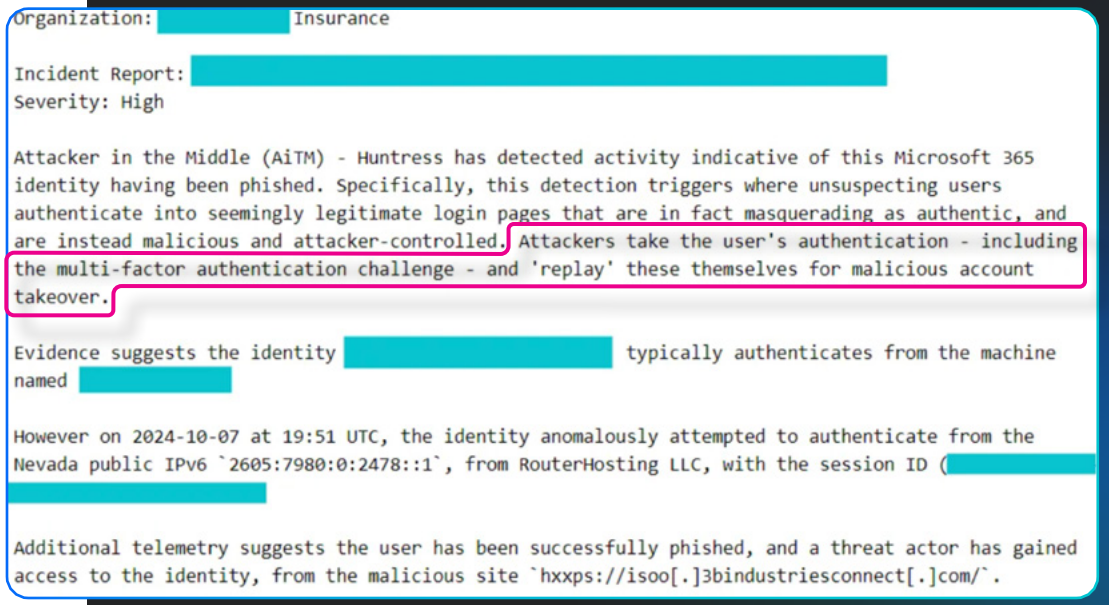
Thanks to the booming infostealer ecosystem, attackers can buy initial access instead of hacking for it. Let's get into how they do it.

Attackers buy infostealer logs (both processed and unprocessed) from dark web marketplaces:

- Processed logs: These have readily available session information, aka instant unauthorized account access
- Unprocessed logs: Require a little attacker elbow grease with special tools to extract things like stolen session tokens, MFA-bypassed cookies, device fingerprints, and internal URLs

With stolen sessions at their fingertips and keyboards, attackers are ready for session replay. In this technique, attackers use tools to simulate an access request to a targeted server, replace their own token with the stolen one, and bypass logins and authentication in targeted accounts. Since this is the legitimate user's session, the server accepts the access request as normal.

In less than an hour, attackers have initial access to systems and networks, including enterprise-level systems, opening the door to ransomware or data theft.

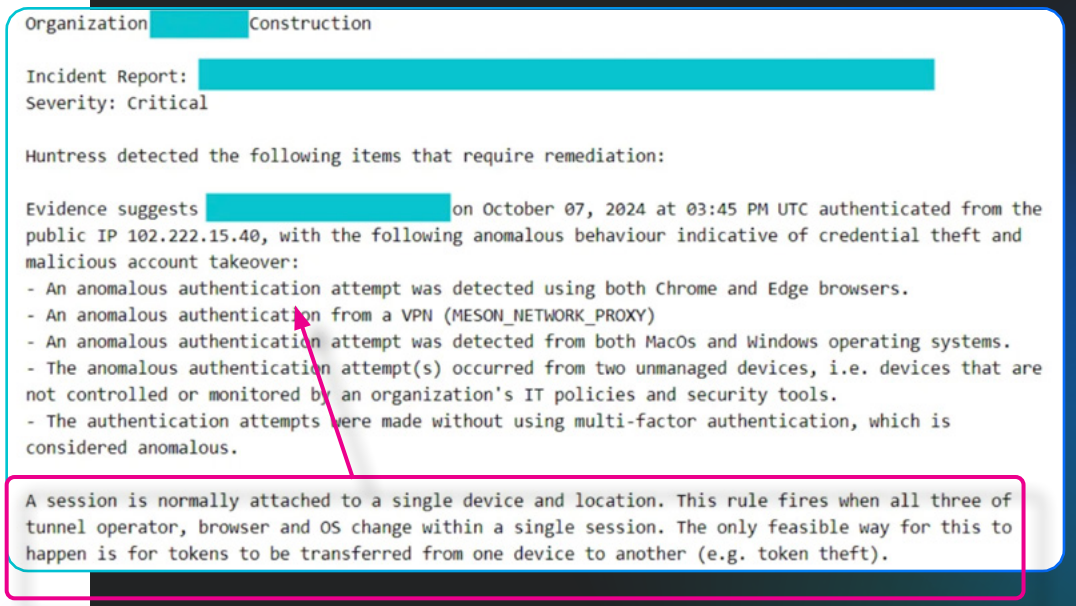


HUNTRESS

Session hijacking is fast and cheap, making it a go-to method for attackers looking for a quick way in. Here's a typical session hijacking attack path that doesn't require phishing or exploits:

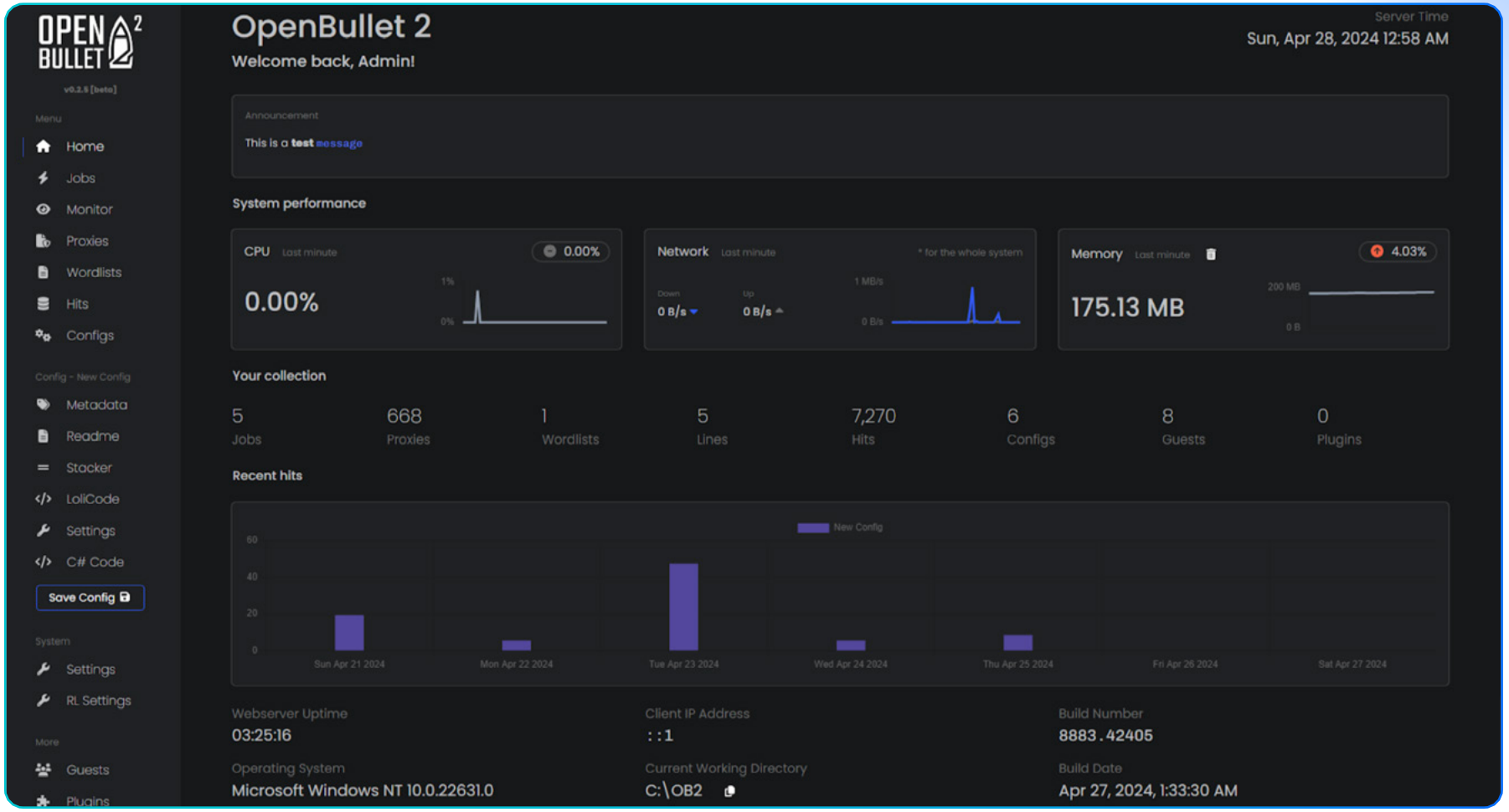
- Buy a log with credentials of a targeted organization
- Connect to a VPN or proxy system in the same global region as the victim
- Run a replay session via automated tools
- Bypass MFA (In many cases, MFA can be sidestepped due to how sessions are handled)
- Browse internal systems or drop malware for persistence
- Escalate to ransomware, extortion, or IP theft

Specialized tools are key for attackers to extract and exploit infostealer log data. Each tool has unique capabilities that help attackers sneak in under the radar.



HUNTRESS

Tool/Script	Main Use Case	Targeted Protocols	Features	How does it help attackers?
OpenBullet/ OpenBullet2	Credential stuffing, session replay	HTTP, JWT, SAML	Configurable workflows, CAPTCHA bypass, and session replay capabilities	Widely used for automating credential attacks and session hijacking scenarios
StealyBot	Automated session hijacking via messaging apps	Web sessions (cookies)	Extracts session tokens, automates hijacking through messaging platforms	Facilitates quick session hijacking through platforms like Telegram
SAML Raider	SAML token manipulation and testing	SAML	Burp Suite extension allows editing and replaying SAML assertions	Essential for testing SAML implementations and identifying vulnerabilities
JWT Tool	JWT analysis and manipulation	JWT	Decodes, modifies, and re-signs JWTs	Useful for testing JWT token security and identifying potential weaknesses
Session Hijacking Script	Manual session hijacking	HTTP, Cookies	Custom scripts to capture and reuse session tokens	Requires manual setup and understanding of session management mechanisms



OpenBullet2 session hijacking tool

HUNTRESS

How To Defend Your Business Against Advanced Credential Theft

Your Go-To Advanced Credential Theft Defense Checklist

Attackers don't just want your passwords anymore. They want **your sessions**—and they're paying for them. The good news? This checklist helps you build stronger defenses against stealthy credential theft tactics to keep your identities secure.

- [ ] **Watch for suspicious identity activity**
  Monitor for sketchy patterns like logins from unexpected locations, browsers, or devices. Be ready to force logouts or re-authentication when you spot them.

- [ ] **Use multi-factor authentication (MFA)**
  MFA isn't a silver bullet, but it adds a critical verification layer that makes life harder for attackers, even when they have valid session tokens

- [ ] **Reduce session lifetimes**
  Cut down how long sessions stay active, so stolen tokens expire before attackers can use them

- [ ] **Use secure cookies**
  Set the Secure flag (HTTPS only) and HttpOnly flag (blocks scripts) to stop cookie theft via insecure channels or cross-site scripting (XSS) attacks

- [ ] **Fortify developer environments**
  Dev tools *will* be compromised. Build safeguards like session replay detection, log sanitization, and short-lived tokens to catch tampering early.

- [ ] **Think like an attacker to spot one**
  Your team is often your first line of defense. Train them to spot phishing tactics, so they don't click malicious links or download infected files that lead to session hijacking attacks.

- [ ] **Rely on Canary credentials**
  Plant fake credentials in your systems to trigger alerts when attackers test or use them. It's a smart early warning system to stop breaches early.

HUNTRESS

Session hijacking is sneaky, but not impossible to spot with the right defenses. Real security is about stacking strong managed layers backed by a 24/7 human-led, AI-assisted Security Operations Center (SOC) to monitor and detect infostealer activity on endpoints, and session hijacking for unauthorized, stealthy access.

Here's how you can stay safe online:

Monitor for suspicious identity activity

Why it works: Spotting unusual patterns, like attempted logins from unexpected locations, browsers, VPNs, OSs, or multiple failed logins, flags potential session hijacking. This is a call for immediate action, like forcing a logout or requiring re-authentication.

What it defends against: Credential stuffing patterns, unusual device fingerprint mismatches, and custom scripts by flagging scripted login attempts.

Use multi-factor authentication (MFA)

Why it works: Yes, MFA can be bypassed with the right attacker techniques, but it always adds another verification layer on top of your username and password. A mobile code or biometric data authentication requirement makes it tougher for attackers to get in, even with valid session tokens. This is crucial against tools that bypass passwords.

What it defends against: Stolen tokens, emulated fingerprints, and custom replay scripts.

Token Theft Detection Triggers

- 7.2% Location
- 27.8% Browser
- 28.9% VPN
- 36.1% OS

HUNTRESS

Reduce session lifetimes

Why it works: Reducing session durations limits the time an attacker has to use a stolen session token. Even if the attacker captures the token (e.g., via OpenBullet or StealyBot), when the session expires quickly, it might already be invalid by the time they try to replay it.

What it defends against: Session replay, stolen browser profiles, and custom scripts by shortening the effective period for replay attacks.

Use secure cookies

Why it works: Setting the Secure flag ensures cookies are only sent over HTTPS, preventing interception over unencrypted connections. The HttpOnly flag blocks client-side scripts from accessing cookies, mitigating XSS attacks.

What it defends against: Cookie theft via insecure channels, cookie theft through client-side exploitation, and custom scripts by reducing the ability to extract cookies insecurely.

Fortify developer environments

Why it works: By assuming developer access tools will be compromised, you can put effective safeguards in place, like session replay detection, including behavioral anomaly tracking and device fingerprinting at the CI/CD and SaaS level. Also, log sanitization helps shut down the storage of tokens, secrets, or credentials in plaintext logs. Using short-lived tokens and scoped permissions replaces vulnerable static secrets that attackers like to hijack. By including layered defenses, robust logging, and multi-tiered validation systems at multiple phases of your development pipeline, you can see instances of unwanted tampering more clearly.

What it defends against: CI/CD compromises, internal supply chain attacks, and development hijacks where downstream victims are the primary goal.

Think like an attacker to spot one

Why it works: Savvy users trained to spot phishing tactics are less likely to fall victim to attacks that lead to session hijacking, like clicking malicious links or downloading infected attachments.

What it defends against: Initial compromise attempts through social engineering tactics, which are often the first step in obtaining session tokens or credentials.

Rely on Canary credentials

Why it works: Canary credentials are fake or decoy credentials placed in systems to detect unauthorized access. If an attacker uses these credentials, it triggers an alert, indicating a breach. This is particularly effective against infostealer marketplaces, where stolen credentials are tested or sold. It gives you solid early detection warnings, especially in environments with high credential exposure risks.

What it defends against: Attacker credential testing, unauthorized profile use, and custom scripts by identifying scripted attempts with canary data.

Defense Strategy	✔ Pros	✘ Cons
Multi-Factor Authentication (MFA)	A foundational control that helps protect initial logins	It's not effective once a session is established. Most infostealers bypass MFA by replaying tokens after it's been validated.
Short Session Lifetimes	Great at disrupting token resale timelines and essential for high-risk apps, like admin panels and developer consoles	This degrades the user experience. Attackers can still act quickly post-infection if the session is fresh. Needs to be paired with device binding.
Secure & Http Only Cookies	Blocks passive theft and XSS-based access. Follows standard OWASP guidance.	Ineffective if the info stealer has full system access (e.g., browser files, memory dumps) and against session replay with cookies already stolen
Anomaly Detection	Crucial for live attacks and detecting anomalies	Heavily dependent on tuning and baselines. Skilled attackers blend in with business hours or spoof device fingerprints to fly under the radar.
User Education	Reduces exposure to phishing attacks and malware infections	It doesn't protect against silent stealer infections (malvertising, cracked software, poisoned packages)
Canary Credentials	Key for early threat alerts and credential resale monitoring	It needs to be carefully placed in the environment. It doesn't trigger alerts for session-only theft unless attackers use fake login credentials.

Stop session hijacking before it starts

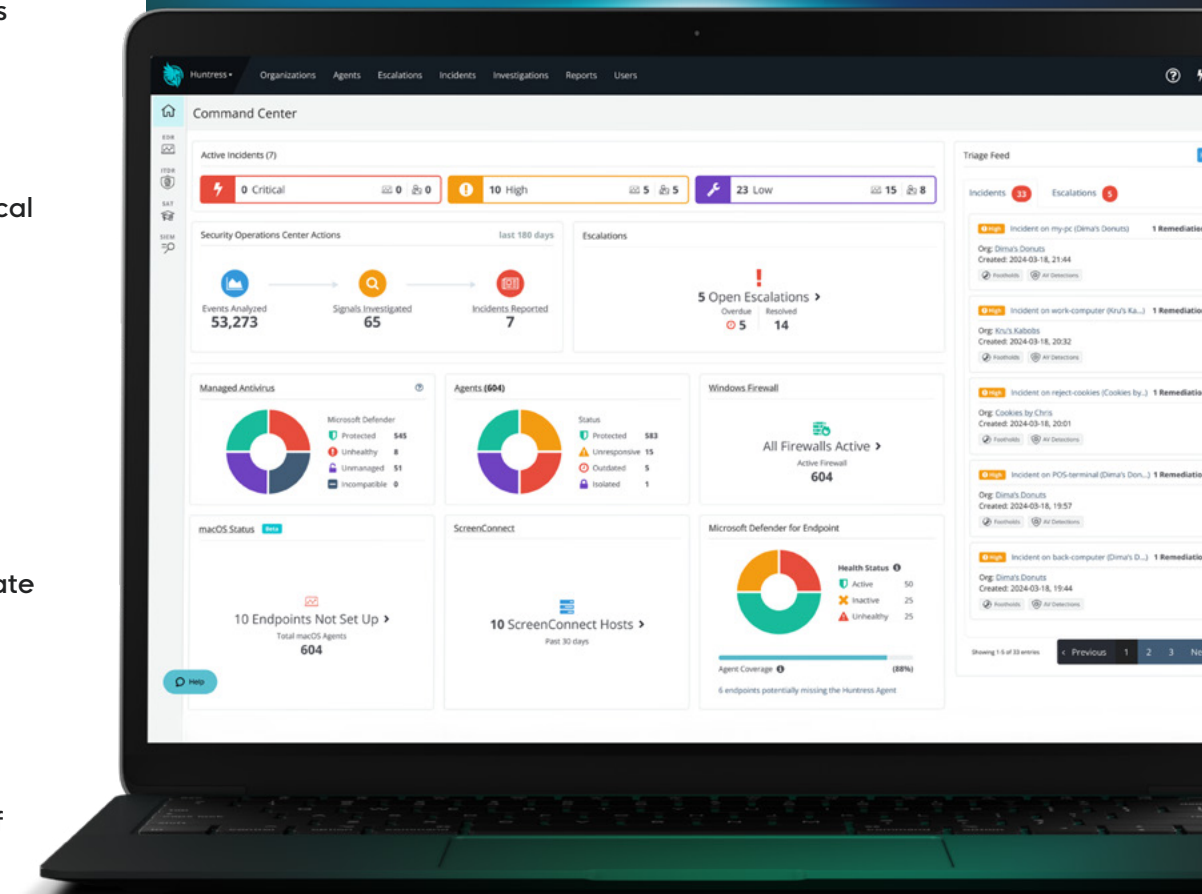
The rapid growth of the infostealer and session hijacking criminal economy should be a clear warning sign for all businesses and industries. Attackers pay for unauthorized access that bypasses trusted credentials and MFA requirements, making unwanted access easier, faster, and cheaper than ever.

As attackers continue to chase tactics that make intrusions stealthier and more lucrative, fast detection is increasingly critical and time-sensitive for your business.

That's where Huntress comes in

At Huntress, we don't rely on off-the-shelf solutions. We design and build our own technology, giving our 24/7 AI-assisted SOC experts the precision and agility to detect, analyze, and eliminate threats in real time. From proprietary endpoint and identity protection to an advanced SIEM and engaging awareness training, our solutions are all managed by peer-recognized cybersecurity leaders.

Huntress does more than wreck hackers—we give you peace of mind. With a team of experts monitoring your systems around the clock for attacks like session hijacking, you'll be confident knowing your business is protected from threat actors slipping past your defenses.

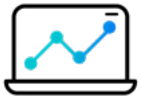


HUNTRESS

We manage threats so you don't have to

Threats don't follow a schedule, but you do. That's why it's so important to have cybersecurity that's fully managed and monitored by a 24/7 AI-assisted Security Operations Center (SOC).

Huntress has fully managed solutions for ALL businesses to detect and protect against threats:



Managed Endpoint Detection and Response (EDR)
An EDR backed by real cybersecurity experts can pinpoint activity indicative of attackers in your systems and networks, isolate the compromised endpoints, and neutralize threats before they can do any damage.

Managed Identity Threat Detection and Response (ITDR)
An ITDR fully managed by a people-powered SOC helps stop attacks by preventing identities from being compromised and abused, e.g., through login hijacks, session theft, and other sneaky attempts to break in.

Managed Security Information and Event Management (SIEM)
SIEM helps spot threat actor activities and attacks early in the attack chain, gives visibility on systems where EDR and ITDR can't be used, provides critical insights to speed up investigations, response, and recovery, and meets compliance requirements.

Managed Security Awareness Training (SAT)
An engaging SAT program developed by real experts can teach your employees to outsmart potential attacks by building sharper instincts and smarter habits.

HUNTRESS

Terminology

Terminology

Access token: a credential used in token-based authentication that grants a client application permission to access a specific resource on behalf of a user

Bearer token: a type of token used for authentication and authorization. A bearer token is used in web applications and APIs to hold user credentials and indicate authorization for requests and access

Cookie: a small piece of data a website stores on a user's computer

Identity and Access Management (IAM) session: a period of time during which an authenticated user or application has temporary, dynamically generated security credentials to access resources

JWT: a secure way to send information between a client and a server, used in web applications and APIs to verify users and prevent unauthorized access

OpenID Connect (OIDC): an identity authentication protocol that is an extension of open authorization (OAuth) 2.0 to standardize the process for authenticating and authorizing users when they sign in to access digital services

Refresh token: a long-lived, special token that allows a client to obtain new, short-lived access tokens without requiring the user to re-enter their credentials

Session replay: a cyberattack where an attacker intercepts and reuses a valid data transmission from a previous session, often a session ID or authentication token, to impersonate a legitimate user

Session token: a unique, encrypted string that identifies a user's session. This lets a server recognize the user across multiple requests without requiring them to log in repeatedly

Single Sign-On (SSO): an authentication method that allows users to log in to multiple applications with a single set of credentials, such as one username and password

Token binding: lets applications and services cryptographically bind their security tokens to the TLS layer to mitigate token theft and replay attacks

Token introspection: the process of validating an OAuth 2.0 access token and retrieving its metadata by sending the token to an authorization server's introspection endpoint

About Huntress

Huntress is a global cybersecurity company on a mission to make enterprise-grade products accessible to all businesses. Purpose-built from the ground up, Huntress' technology is specifically designed to continuously address the unique needs of security and IT teams of all sizes. From Endpoint Detection and Response (EDR) and Identity Threat Detection and Response (ITDR) to Security Information and Event Management (SIEM) tools and Security Awareness Training (SAT), the platform provides targeted protection for endpoints, identities, data, and employees, delivering trusted outcomes and valuable peace of mind.

Our 24/7, AI-assisted Security Operations Center (SOC) is powered by a team of world-renowned engineers, researchers, and security analysts, dedicated to stopping cyber threats before they can cause harm. Huntress is often the first to respond to major hacks and incidents, with its expert security team sharing real-time tradecraft analysis and actionable advisories with the community. Currently safeguarding over 4 million endpoints and 8+ million identities, Huntress empowers security teams, IT departments, and Managed Service Providers (MSPs) worldwide to protect their businesses with enterprise-grade security accessible to everyone.

As long as hackers keep hacking, Huntress keeps hunting. Join the hunt at www.huntress.com and follow us on X, Instagram, Facebook, and LinkedIn.

HUNTRESS