C

Credential Harvesting: Phishing, Token Theft, Warning Signs, and Response

Learn how attackers harvest passwords, session cookies, tokens, and MFA approvals, how to recognize affected accounts, and what to revoke after exposure.

PUBLISHER
Gridinsoft Support

UPDATED
Oct 8, 2026

READ TIME
3 min

CONTENTS

In this guide

01 What is credential harvesting?
02 Common harvesting methods
03 Password theft versus session theft
04 Warning signs
05 Immediate response
06 Investigate the scope
07 Prevention
08 Related attacks

Credential harvesting is the collection of information that can be used to access an account: usernames, passwords, session cookies, API tokens, recovery codes, authentication secrets, or fraudulent MFA approvals. Attackers use harvested access for fraud, data theft, business email compromise, and movement into additional systems.

01 What is credential harvesting?

Credential harvesting describes the attacker's collection objective, not only one delivery method. A fake login page can capture credentials that a victim enters, while an infostealer can extract passwords and session tokens already stored on a device. Phishing is a common path, but breached databases, keylogging, malicious browser extensions, and adversary-in-the-middle proxies can produce the same result.

02 Common harvesting methods

- Fake sign-in pages reached through phishing email, text messages, QR codes, ads, or search results.
- Infostealer malware that extracts browser passwords, cookies, tokens, and cryptocurrency data.
- Keylogging, clipboard monitoring, browser extensions, or remote-access tools.
- Adversary-in-the-middle phishing that relays a real sign-in and captures the resulting session.
- LLMNR or name-resolution poisoning on a local network.
- Compromised databases, application logs, source repositories, and cloud storage.
- Social engineering that convinces a user to disclose a code or approve an MFA prompt.

03 Password theft versus session theft

Changing a password can stop future password authentication, but it may not invalidate existing sessions, application passwords, OAuth grants, refresh tokens, or API keys. A stolen session cookie can sometimes let an attacker act as the user without entering the password or completing MFA again.

Effective response therefore includes session revocation and review of every authentication and recovery mechanism, not only a password reset.

04 Warning signs

- Sign-ins from unexpected devices, locations, networks, or times.
- New mailbox forwarding rules, delegates, filters, or deleted security messages.
- Unknown MFA methods, recovery email addresses, phone numbers, passkeys, or trusted devices.
- OAuth applications, API tokens, or sessions the user does not recognize.
- Password-reset or one-time-code messages the user did not request.
- Contacts receiving messages the real user did not send.

05 Immediate response

1. Use a known-clean device and open the service from a saved bookmark or manually entered official address.
2. Change the password to a unique value and reset reused passwords on other services.
3. Revoke all active sessions, refresh tokens, API keys, and unfamiliar application access.
4. Remove unauthorized MFA and recovery methods, then enable phishing-resistant MFA where available.
5. Inspect the original device for infostealers before using replacement credentials on it.

6. Notify the account owner, administrator, financial institution, or incident-response team as appropriate.

06 Investigate the scope

Preserve the lure URL, sender, message headers, QR code, attachment, time, browser history, device alerts, and relevant sign-in logs. Review successful activity as well as failures. Determine which credentials, tokens, accounts, and devices were exposed and what the attacker accessed or changed.

For work accounts, inspect mailbox rules, cloud downloads, administrative changes, new applications, lateral movement, and financial requests. Do not revisit a malicious page simply to capture another screenshot.

07 Prevention

- Use a password manager, unique passwords, and phishing-resistant MFA such as security keys or properly implemented passkeys.
- Disable legacy authentication and unused application-password mechanisms.
- Restrict token lifetime and require reauthentication for high-risk actions.
- Monitor new MFA methods, OAuth consent, impossible travel, and suspicious session behavior.
- Keep browsers and endpoints updated and control unapproved extensions.
- Train users to verify domains and unexpected requests rather than relying on branding or display names.

08 Related attacks

Harvesting obtains credentials; credential stuffing tests previously stolen password pairs against other services. Phishing is one common delivery method, while spoofing makes an identity or destination appear trustworthy. One incident can use all three techniques.

NEED MORE HELP?

We are here when the guide is not enough.

Visit the Gridinsoft Support Center for current product guidance or submit a support request with the details of your case.

Open this guide online at support.gridinsoft.com