R

RAM Scraping: How Memory-Scraping Malware Steals Data

RAM scraping malware searches a running process's memory for sensitive data during the brief time it appears in readable form. Learn how POS attacks work and how to detect, contain, and prevent them.

PUBLISHER
Gridinsoft Support

UPDATED
Oct 8, 2026

READ TIME
4 min

CONTENTS

In this guide

01 How a POS RAM-scraping attack works

02 Why encryption does not always stop it

03 RAM scraping vs. related techniques

04 Possible warning signs

05 What to do after suspected RAM scraping

06 How to reduce the risk

07 Frequently asked questions

RAM scraping, or **memory scraping**, is a technique in which malware searches a running process's memory for sensitive data. Information protected while stored or transmitted may briefly exist in readable form while an application processes it. Attackers try to capture it during that window.

The best-known targets are point-of-sale (POS) systems processing payment-card data, but memory-scraping techniques can also target credentials, browser sessions, authentication tokens, cryptographic material, or other secrets in desktop and server processes.

01 How a POS RAM-scraping attack works

1. **Initial compromise:** attackers gain access through stolen remote-access credentials, phishing, an exposed service, a vulnerable application, or another infected system.
2. **Privilege and persistence:** malware obtains enough access to inspect the memory of the payment application and remain on the endpoint.
3. **Memory search:** the scraper repeatedly scans selected processes for patterns that resemble payment track data or account numbers.
4. **Validation and staging:** matching data is filtered and temporarily collected. The malware may compress or encrypt it to hide the contents.
5. **Exfiltration:** the staged data leaves through web traffic, file transfer, remote-access infrastructure, or another compromised host.

Implementation varies. Some malware reads another process directly, while other samples inject code, dump memory, or use privileged access. A pattern match is not automatically a valid payment record, so scrapers often filter results before theft.

02 Why encryption does not always stop it

Encryption protects data only while it remains encrypted. In a traditional payment flow, card data may be decrypted or read by the POS application before the next protected transmission. Disk encryption and TLS do not protect that brief processing state inside RAM.

A validated point-to-point encryption (P2PE) solution reduces this exposure by encrypting account data in a secure interaction device and keeping it protected until it reaches the approved decryption environment. Tokenization can reduce stored account data, but it is not a substitute for protecting the initial capture path.

03 RAM scraping vs. related techniques

Technique	Target	Difference
RAM scraping	Readable data in process memory	Searches memory while an application is using the data
Keylogging	Keyboard input	Captures keystrokes before or as an application receives them
Form grabbing	Data submitted through forms	Intercepts application or browser form values
File theft	Saved databases, logs, or exports	Targets data already written to storage

04 Possible warning signs

- Unknown processes repeatedly opening POS or browser processes with memory-read access.
- Code injection, unexpected memory dumps, credential-access alerts, or unsigned services.
- New scheduled tasks, startup entries, remote tools, or administrator accounts on payment systems.
- POS endpoints making outbound connections that are not required for payment processing.
- Compressed or encrypted staging files and repeated transfers at quiet times.
- Fraud reports clustering around one store, terminal group, or processing period.

Individual signs are not conclusive. Correlate endpoint telemetry, authentication, process access, DNS, firewall, payment, and fraud data.

05 What to do after suspected RAM scraping

1. Isolate affected endpoints and payment segments while maintaining evidence required by the incident plan.
2. Contact the acquiring bank, payment processor, forensic provider, legal counsel, and other required parties according to contracts and applicable breach rules.
3. Identify the access method, malware, affected processes, account-data window, staging location, and exfiltration route.
4. Disable compromised accounts, rotate exposed credentials and keys, and remove unauthorized remote access.
5. Rebuild affected POS systems from trusted images. Do not assume deleting one executable restores integrity.
6. Monitor replacement systems and payment activity for recurrence.

06 How to reduce the risk

Use validated P2PE where appropriate, minimize account-data handling, and never store prohibited sensitive authentication data after authorization. Segment payment systems from office and guest networks, restrict outbound traffic, use multifactor authentication for remote access, remove default accounts, and apply least privilege. Application allowlisting, endpoint monitoring, centralized logs, timely patching, file-integrity controls, and alerts for process-memory access can make installation and collection harder to hide.

07 Frequently asked questions

Is RAM scraping the same as fileless malware?

No. RAM scraping describes a data-theft technique. The malware performing it may use files, memory-only components, or both.

Does rebooting solve a RAM-scraping infection?

A reboot clears ordinary memory but does not remove malware persistence, stolen data, compromised credentials, or the original access path.

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