SharePoint ToolShell | Zero-Day Exploited in-the-Wild Targets Enterprise Servers

By SentinelOne

Published: 2025-07-21 · Archived: 2026-04-06 01:00:00 UTC

On July 19th, Microsoft confirmed that a 0-day vulnerability impacting on-premises Microsoft SharePoint Servers, dubbed "**ToolShell**" (by researcher Khoa Dinh @_l0gg), was being actively exploited in the wild. This flaw has since been assigned the identifier CVE-2025-53770, along with an accompanying bypass tracked as CVE-2025-53771. These two new CVEs are being used alongside the previously patched CVEs (49704/49706) which were patched on July 8th, with PoC code surfacing by July 14th.

The advisory also confirmed emergency patches for on-prem SharePoint Subscription Edition and SharePoint Server 2019, with updates scheduled for version 2016 as well. We strongly recommend immediate patching, and following Microsoft's recommendations of enabling AMSI detection, rotating ASP.NET machine keys, and isolating public-facing SharePoint servers until defenses are in place.

SentinelOne first observed ToolShell exploitation on **July 17th, ahead of official Microsoft advisories**. Since then, we've identified three distinct attack clusters, each with unique tradecraft and objectives. In this blog, we unpack the timeline, explore these clusters, and equip defenders with best-practice mitigation strategies. At this time, we provide no attribution beyond this early clustering as research is ongoing.

Observed Targets

We have observed initial ToolShell exploitation against high value organizations, with victims primarily in technology consulting, manufacturing, critical infrastructure, and professional services tied to sensitive architecture and engineering organizations. The early targets suggest that the activity was initially carefully selective, aimed at organizations with strategic value or elevated access.

The attacks that we describe in this report were targeted in nature and occurred before public disclosure of the vulnerability spurred mass exploitation efforts from a wider set of actors. We expect broader exploitation attempts to accelerate, driven by both state-linked and financially motivated actors seeking to capitalize on unpatched systems.

SentinelOne has observed multiple state-aligned threat actors, unrelated to the first wave of exploitation, beginning to engage in reconnaissance and early-stage exploitation activities. Additionally, we've also identified actors possibly standing up decoy honeypot environments to collect and test exploit implementations , as well as sharing tooling and tradecraft across known sharing platforms. As awareness spreads within these communities, we expect further weaponization and sustained targeting of vulnerable SharePoint infrastructure.

Technical Overview

Both previously patched CVEs (49704/49706) were first disclosed at Pwn2Own Berlin. It was later discovered that these two flaws could be paired together to produce the full RCE 'ToolShell' attack chain. The name 'ToolShell' refers to the initial abuse of SharePoint's / `ToolPane.aspx` (CVE-2025-49704), a system page used for website configuration and management.

This vulnerability chain enables unauthenticated remote code execution by sending a crafted POST request to the URI `/layouts/15/ToolPane.aspx?DisplayMode=Edit`, exploiting a logic flaw in the Referer header validation. This bypass allows attackers to access SharePoint's ToolPane functionality without authentication, ultimately leading to code execution via uploaded or in-memory web components.

xxx.aspx

On July 18th, 2025 at 09:58 GMT, SentinelOne observed a single exploitation attempt where the attacker dropped a custom password-protected ASPX webshell named `xxx.aspx`. This activity appears to be hands-on and exploratory in nature, likely performed by a human operator rather than an automated script.

The webshell was written to the following path:

```
C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\16\TEMPLATE\LAYOUTS\xxx.aspx
```

This webshell provides a basic HTML interface allowing three primary functions:

1. Authentication via an embedded form that sets a cookie.
2. Command Execution by submitting commands through the `GTaRkhJ9wz` parameter, which are run via `cmd.exe` and returned to the client.
3. File Upload via a multipart form using fields `0z3H8H8atO` (file) and `7KAjlfecWF` (destination path).

The shell leverages basic obfuscation and validation mechanisms, including cookie-based authentication and a hardcoded SHA512 hash to restrict access. The password check logic suggests the actor anticipated repeated or remote usage of the shell.

After the webshell was dropped, the attacker issued the following commands:

```
cmd.exe /c whoami > c:\progra~1\common~1\micros~1\webser~1\16\template\layouts\info.js
```

The first attempt to redirect the `whoami` output failed due to a typo (`\templa`), indicating the activity was likely manual and exploratory. The corrected second command successfully writes the output of `whoami` into a web-accessible `.js` file, a common tactic for validating command execution and potentially retrieving output through a browser.

While this activity was limited to a single observed instance, the customized tooling and interactive behavior suggest a deliberate post-exploitation attempt by a threat actor testing or preparing for broader operations.

spinstall0.aspx

SentinelOne observed two distinct waves of activity involving a consistent final payload, `spinstall0.aspx`, dropped across SharePoint environments from different attacker infrastructure on July 18 and 19, 2025. While the

initial dropper scripts varied slightly between waves, both resulted in deployment of the same webshell, designed to extract and expose sensitive cryptographic material from the host.

First Wave – July 18, 2025 (14:54–18:44 GMT)

Source IP: `107.191.58[.]76`

This initial wave involved PowerShell-based payload delivery. A base64-encoded blob was decoded and written to the SharePoint `LAYOUTS` directory:

```
$base64String = [REDACTED]
$destinationFile = "C:\PROGRA~1\COMMON~1\MICROS~1\WEBSER~1\16\TEMPLATE\LAYOUTS\spinstall0.aspx"
$decodedBytes = [System.Convert]::FromBase64String($base64String)
$decodedContent = [System.Text.Encoding]::UTF8.GetString($decodedBytes)
$decodedContent | Set-Content -Path $destinationFile -ErrorAction Stop
```

The resulting file, `spinstall0.aspx`, is not a traditional command webshell but rather a reconnaissance and persistence utility:

```
<%@ Import Namespace="System.Diagnostics" %>
<%@ Import Namespace="System.IO" %>
```

This code extracts and prints the host's `MachineKey` values, including the `ValidationKey`, `DecryptionKey`, and cryptographic mode settings—information critical for attackers seeking to maintain persistent access across load-balanced SharePoint environments or to forge authentication tokens.

Second Wave – July 19, 2025 (03:06–07:59 GMT)

Source IP: `104.238.159[.]149`

Roughly 12 hours later, a second wave used nearly identical logic to deliver the same `spinstall0.aspx` payload. The key difference was in the PowerShell staging script:

```
$b = [REDACTED]
$c = "C:\PROGRA~1\COMMON~1\MICROS~1\WEBSER~1\15\TEMPLATE\LAYOUTS\spinstall0.aspx"
$d = [System.Convert]::FromBase64String($b)
$e = [System.Text.Encoding]::UTF8.GetString($d)
$e | Set-Content -Path $c -ErrorAction Stop
Start-Sleep -s 3
```

While the encoded payload was marginally different in form, it decoded to the same `spinstall0.aspx` shell. The change in target directory, from `16\TEMPLATE` to `15\TEMPLATE`, may reflect testing across different SharePoint versions or environments.

Unlike more interactive webshells observed in this campaign, `spinstall0.aspx` does not support command execution or file upload. Instead, its singular purpose appears to be information gathering, specifically targeting cryptographic secrets that could be reused to forge authentication or session tokens across SharePoint instances.

Given the uniqueness and strategic value of the `MachineKey` data harvested by this shell, we assess this cluster to be part of a broader effort to establish durable access into high-value SharePoint deployments.

“no shell”

This activity cluster, tracked as `“no shell”`, represents a more advanced and stealthy approach compared to others in this campaign. SentinelOne observed this cluster operating between July 17, 2025 10:35:04 GMT and July 18, 2025 03:51:29 GMT, making it our earliest known exploitation of CVE-2025-53770 in the wild.

Unlike the other clusters, no persistent webshells were written to disk. Instead, telemetry and behavioral indicators suggest the attackers relied on in-memory .NET module execution, avoiding traditional file-based artifacts entirely. This approach significantly complicates detection and forensic recovery, underscoring the threat posed by fileless post-exploitation techniques.

All observed activity in this cluster originated from a single IP address: `96.9.125[.]147`. Despite the lack of file system artifacts, compromised hosts exhibited patterns consistent with SharePoint exploitation, followed by encoded payload delivery and dynamic assembly loading via PowerShell or native .NET reflection.

Given the timing, just days after public proof-of-concept chatter began, and the sophistication of the fileless execution chain, we assess this cluster to be either a skilled red team emulation exercise or the work of a capable threat actor with a focus on evasive access and credential harvesting.

Defenders should be especially vigilant for memory-resident activity following SharePoint exploitation attempts and should employ EDR solutions capable of detecting anomalous .NET execution patterns and assembly loading.

Conclusion

Modern threat actors are maximizing gains from patch diffing, n-day adoption, and iterative development of exploits through fast adoption. SharePoint servers are attractive to threat actors for the high likelihood that they store sensitive organizational data. Beyond their value as a knowledge store, vulnerable SharePoint servers can be used to stage and deliver additional attack components to the victim organization for internal watering hole attacks. The ease of exploitation and potential value of the data hosted on these servers make ‘ToolShell’ a potent and dangerous attack chain.

As of this writing, SharePoint Online for Microsoft 0365 is not impacted. Our research teams have provided out-of-the-box Platform Detection rules and Hunting Queries to assist in discovering and isolating related behavior. We recommend that vulnerable organizations apply the available security updates released by Microsoft (released July 21, 2025) to mitigate the related vulnerabilities as soon as possible. SentinelOne is actively monitoring its customer base for impact and is notifying those affected as they are identified.

Indicators of Compromise

SHA-1

f5b60a8ead96703080e73a1f79c3e70ff44df271 - spinstall0.aspx webshell
fe3a3042890c1f11361368aeb2cc12647a6fdae1 - xxx.aspx webshell
76746b48a78a3828b64924f4aedca2e4c49b6735 - App_Web_spinstall0.aspx.9c9699a8.avz5nq6f.dll, a compiled version of spinstall0.aspx

IP Addresses

96.9.125[.]147 - attacker IP from “no shell” cluster
107.191.58[.]76 - attacker IP used in 1st wave of spinstall0.aspx cluster
104.238.159[.]149 - attacker IP used in 2nd wave of spinstall0.aspx cluster

New SentinelOne Platform Detection Rules

- Web Shell Creation in LAYOUTS Directory
- Web Shell File Detected in LAYOUTS Directory
- Suspicious Process Spawned by SharePoint IIS Worker Process

SentinelOne Platform Hunting Queries

```
dataSource.name = 'SentinelOne' and endpoint.os = "windows" and event.type = "Process Creation" and s
```

```
dataSource.name = 'SentinelOne' and endpoint.os = "windows" and event.type = "Process Creation" and s
```

Disclaimer
