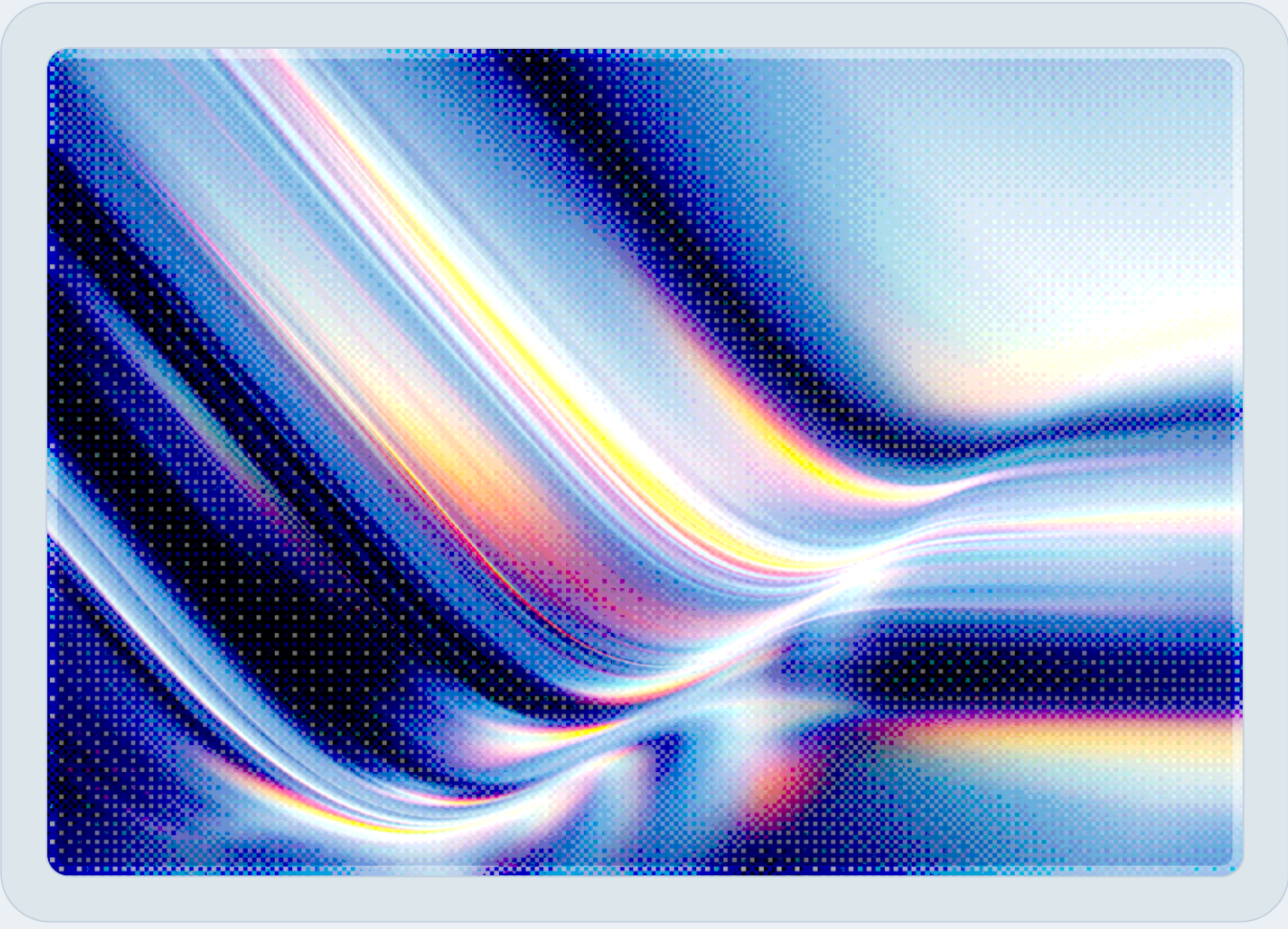
CSAI Foundation | Cloud Security Alliance

ServiceNow AI Platform Sandbox Escape Under Active Attack

2026-07-22

AI-assisted Rapid Research



CSAI

CSA cloud security alliance®

Key Takeaways

CVE-2026-6875 is a critical, CVSS 9.5 sandbox escape vulnerability in the ServiceNow AI Platform, the shared scripting and workflow layer beneath ServiceNow's ITSM, CSM, HRSD, and SecOps modules as well as its Now Assist and AI Agent Studio agentic capabilities [1]. An unauthenticated attacker with network access to a vulnerable instance can send a crafted HTTP POST request to the pre-authentication `/assessment_thanks.do` endpoint, escape the platform's server-side script sandbox, and execute arbitrary code, with no credentials, phishing, or prior foothold required [2][3]. Security researchers at Searchlight Cyber reported the flaw to ServiceNow on April 1, 2026, and ServiceNow patched its own hosted instances within a day; self-hosted and partner-managed customers received patches quietly over the following two months, and the vulnerability's existence and technical details were made public on July 13, 2026 [2]. Active exploitation began within days of that disclosure: threat intelligence firm Defused observed the earliest attack attempts in the days immediately following, with reporting dating the first activity to somewhere between July 17 and July 19, 2026, and confirmed that attackers were using a sandbox-escape gadget chain distinct from the one described in the published proof of concept, indicating independent vulnerability research or rapid variant development by the threat actors involved [2][3][4]. Organizations running self-hosted ServiceNow instances that have not yet applied the June and July 2026 patches should treat this as an emergency remediation priority, since successful exploitation grants complete compromise of the instance and any connected proxy servers used to bridge cloud and on-premises resources [1].

Background

ServiceNow markets its core product as the "AI Platform," reflecting the company's positioning of generative and agentic AI as integral to its workflow automation stack rather than a bolted-on module [1][5]. Features such as Now Assist, the AI Agent Studio, and the AI Agent Orchestrator all execute on top of the same underlying scripting engine that has long powered ServiceNow's business rules, script includes, and workflow logic, and that engine relies on a sandbox to prevent customer-authored and AI-generated scripts from accessing resources or executing operations beyond their intended scope [1][5]. This sandbox functions as a foundational trust boundary: ServiceNow instances are typically deployed

with broad integrations into enterprise identity providers, ticketing data, HR records, and IT infrastructure, so a script sandbox that can be escaped effectively collapses the separation between a low-privilege scripting context and the full authority of the platform itself.

CVE-2026-6875 breaks that boundary without requiring any authentication at all. The vulnerability is reachable through `/assessment_thanks.do`, a pre-authentication endpoint that ServiceNow exposes as part of its assessment and survey functionality, and which was apparently not intended to accept the kind of crafted input that ultimately reaches sandboxed script evaluation [2][3]. By submitting a specially constructed HTTP POST request to this endpoint, an attacker can trigger a chain of sandbox-escape gadgets that culminates in remote code execution on the underlying instance, all before any login prompt is ever presented [2][3][4]. Published reporting has not disclosed the precise mechanism by which input reaches the sandbox evaluator, but the pattern, an unauthenticated entry point feeding attacker-controlled data into a code-evaluation context that was assumed to be isolated, mirrors a similar failure seen recently in at least one other AI-adjacent workflow platform, Langflow's CVE-2026-5027, raising the question of whether this reflects an emerging pattern across the category rather than an established one [6].

Searchlight Cyber identified and reported CVE-2026-6875 to ServiceNow on April 1, 2026, giving the vendor more than three months of lead time before public disclosure [2]. ServiceNow's response followed a split remediation timeline: because ServiceNow directly operates the infrastructure for its cloud-hosted customers, it was able to deploy a fix across those environments within a single day of receiving the report [2]. Self-hosted and partner-managed deployments, which ServiceNow does not directly control, received patches quietly over the course of June 2026, ahead of any public acknowledgment that the vulnerability existed; the flaw itself, along with the full list of affected release trains, only became public on July 13, 2026, when ServiceNow disclosed the vulnerability and confirmed the fix across the Brazil, Australia, Zurich, and Yokohama release families, specifically Brazil EA and GA, Australia Patch 2, Zurich Patch 7b and 9, and Yokohama Patch 12 Hot Fix 1b and 13 [1][2]. This roughly three-and-a-half-month gap between private disclosure to the vendor and public disclosure of the vulnerability's existence reflects the operational reality that ServiceNow's largest customers run heavily customized, self-managed instances that cannot be patched unilaterally by the vendor; because the June patches were rolled out without public fanfare, organizations that do not track vendor patch notes independently of public vulnerability disclosures may not have prioritized applying them until the July 13 disclosure made the risk unmistakable, leaving a window during which a working exploit chain, once independently discovered, could be weaponized before defenders on self-hosted platforms fully appreciated the urgency.

Security Analysis

The technical severity of CVE-2026-6875 rests on the convergence of three factors that security teams should not treat as independent risks. First, the vulnerability requires no authentication whatsoever, which removes the most common barrier to exploitation and makes the instance's exposure to the public internet, rather than any credential-based control, the primary determinant of risk. Second, the flaw grants remote code execution rather than a lesser outcome such as information disclosure or denial of service, meaning a successful attacker gains the ability to run arbitrary commands with the privileges of the ServiceNow application process. Third, because ServiceNow instances routinely operate as an integration hub, connecting to on-premises resources through MID Server proxies and to other cloud services through configured integrations, the blast radius of a compromised instance extends well beyond the ServiceNow platform itself [1]. ServiceNow's own advisory acknowledges that a successful exploit can result in "complete compromise of the ServiceNow instance as well as all connected proxy servers," language that signals lateral movement into an organization's broader on-premises and cloud environment is a realistic outcome, not a theoretical worst case [1].

The active exploitation evidence reported by Defused adds an important nuance for defenders relying solely on signatures or detections built from the published proof of concept. Defused researchers found that the payloads observed in the wild hit the same pre-authentication endpoint described in ServiceNow's advisory, but that the sandbox-escape gadget chain used to achieve code execution took a different route than the one documented in the public PoC [2][3]. That divergence suggests attackers either conducted their own independent analysis of the patch to reconstruct an alternate exploitation path, a technique known as patch diffing, or discovered a parallel escape mechanism that was not captured in ServiceNow's original disclosure. Either explanation means that detections narrowly tuned to the published gadget chain are unlikely to catch every exploitation attempt, and defenders should prioritize monitoring the vulnerable endpoint itself, along with post-exploitation indicators such as unexpected process spawning or outbound connections from the ServiceNow application tier, rather than relying exclusively on payload-specific signatures.

ServiceNow has stated that, based on its investigation to date, it has not observed evidence that the exploitation activity reported by outside researchers is related to instances that ServiceNow itself hosts [1]. This distinction matters operationally: it implies that the currently observed attacks are concentrated against self-hosted and partner-managed instances that received patches later and on a timeline the vendor could not directly enforce, reinforcing the pattern already visible in the disclosure timeline. In parallel with the emergency patches, ServiceNow introduced a longer-term architectural mitigation called Guarded Script, which restricts the types of code that can execute within sandbox contexts across

the platform [2]. Guarded Script appears to function as a defense-in-depth response rather than a one-time patch, in that it narrows the sandbox's effective capability surface so that future gadget chains, including variants not yet discovered, have fewer primitives available to escalate into code execution.

Recommendations

Immediate Actions

Security teams operating self-hosted or partner-managed ServiceNow instances on the Brazil, Australia, Zurich, or Yokohama release families should confirm patch status against the specific versions named in ServiceNow's advisory (Brazil EA/GA, Australia Patch 2, Zurich Patch 7b/9, Yokohama Patch 12 Hot Fix 1b/13), which were rolled out quietly in June 2026 and publicly confirmed on July 13, 2026, and should apply them without delay if they have not already done so [1][2]. Given confirmed in-the-wild exploitation, unpatched instances reachable from the internet should be treated as a live incident-response scenario rather than a routine patching backlog item; teams should review web server access logs for POST requests to `/assessment_thanks.do` and treat any unexpected traffic to that endpoint as a potential compromise indicator warranting immediate investigation.

Short-Term Mitigations

Organizations that cannot immediately patch should restrict network exposure of the affected instance, using firewall rules, IP allowlisting, or a web application firewall to limit which sources can reach the ServiceNow instance until patching is complete, while recognizing that WAF rules alone are an imperfect substitute for the underlying fix given the demonstrated existence of at least two distinct exploitation paths. Instances should also be audited for signs of prior compromise, including unfamiliar scheduled jobs, unexpected outbound network connections from the application tier, and any anomalous activity on MID Servers or other proxy infrastructure connected to the instance, since an attacker who gained access before detection could have already pivoted into connected systems.

Strategic Considerations

Looking beyond this specific CVE, security leaders should recognize that the convergence of workflow automation platforms with embedded generative and agentic AI capabilities, of which ServiceNow's AI Platform is a leading example, is expanding the attack surface of systems that were historically treated as internal business tools rather than internet-facing, security-critical infrastructure. As vendors add AI

Agent Studio-style capabilities that let customers and AI agents author and execute custom logic, the underlying sandboxing and code-execution controls that isolate that logic become as security-critical as the authentication layer itself, and should be inventoried, monitored, and included in vulnerability management programs with the same rigor traditionally reserved for internet-facing web applications. Organizations should also build patch-timeline awareness into their vendor risk assessments: the gap between private disclosure, vendor-hosted remediation, and self-hosted patch availability, three and a half months in this case, is a pattern worth watching for at other SaaS platforms with a similar hosted/self-hosted split, and should inform how quickly self-hosted or partner-managed deployments are expected to apply vendor security updates once patches become available.

CSA Resource Alignment

CSA's rapid research note on Langflow Path Traversal: Unauthenticated RCE Actively Exploited analyzed CVE-2026-5027, a path traversal flaw in the Langflow AI development platform that, like CVE-2026-6875, allowed unauthenticated attackers to reach remote code execution through a platform feature that was assumed to isolate untrusted input from the underlying system [7]. Both incidents illustrate the same structural risk in AI-adjacent workflow and development platforms: a pre-authentication endpoint or upload path feeds attacker-controlled data into a context, whether a file storage service or a script sandbox, that was not hardened against adversarial input, and the resulting compromise extends to every credential and integration the platform holds. The Langflow note's recommendation to treat AI development and workflow platforms as first-class, internet-facing attack surface, rather than internal tooling exempt from the patch cadence applied to customer-facing applications, applies directly to ServiceNow's AI Platform given its role as an enterprise-wide integration hub.

The vulnerability and sandbox-isolation failure at the center of CVE-2026-6875 also falls squarely within the scope of CSA's AI Controls Matrix (AICM) v1.1, whose Threat and Vulnerability Management and Application and Interface Security domains address exactly this class of risk: ensuring that AI-enabled platforms maintain rigorous vulnerability management processes and that code-execution boundaries such as script sandboxes are treated as security controls requiring their own validation and monitoring, not implementation details to be assumed secure [8]. Because ServiceNow's AI Platform increasingly hosts autonomous, agentic workflows through capabilities like AI Agent Studio and the AI Agent Orchestrator, CSA's MAESTRO agentic AI threat modeling framework is also relevant background for organizations assessing this incident, since MAESTRO's layered approach to agentic infrastructure explicitly calls out the deployment and infrastructure layer, including sandboxing and isolation guarantees, as a distinct threat surface that agentic capabilities can inherit and amplify when the underlying platform's isolation controls fail [9].

References

[1] ServiceNow. "CVE-2026-6875 - Sandbox Escape in ServiceNow AI Platform." Now Support Portal, July 13, 2026.

[2] Help Net Security. "ServiceNow pre-auth RCE exploited in the wild (CVE-2026-6875)." Help Net Security, July 20, 2026.

[3] The Hacker News. "Critical ServiceNow AI Platform Flaw Exploited for Unauthenticated Code Execution." The Hacker News, July 2026.

[4] BleepingComputer. "Critical ServiceNow code execution flaw now exploited in attacks." BleepingComputer, July 2026.

[5] Kellton. "ServiceNow AI Agents and Agentic Workflow Automation: Complete Guide." Kellton, 2026.

[6] The Hacker News. "Langflow Vulnerability CVE-2026-5027 Exploited for Unauthenticated RCE." The Hacker News, June 2026.

[7] Cloud Security Alliance AI Safety Initiative. "Langflow Path Traversal: Unauthenticated RCE Actively Exploited." CSA Lab Space, June 12, 2026.

[8] Cloud Security Alliance. "AI Controls Matrix (AICM) v1.1." Cloud Security Alliance, June 22, 2026.

[9] Cloud Security Alliance. "Agentic AI Threat Modeling Framework: MAESTRO." Cloud Security Alliance, February 6, 2025.

