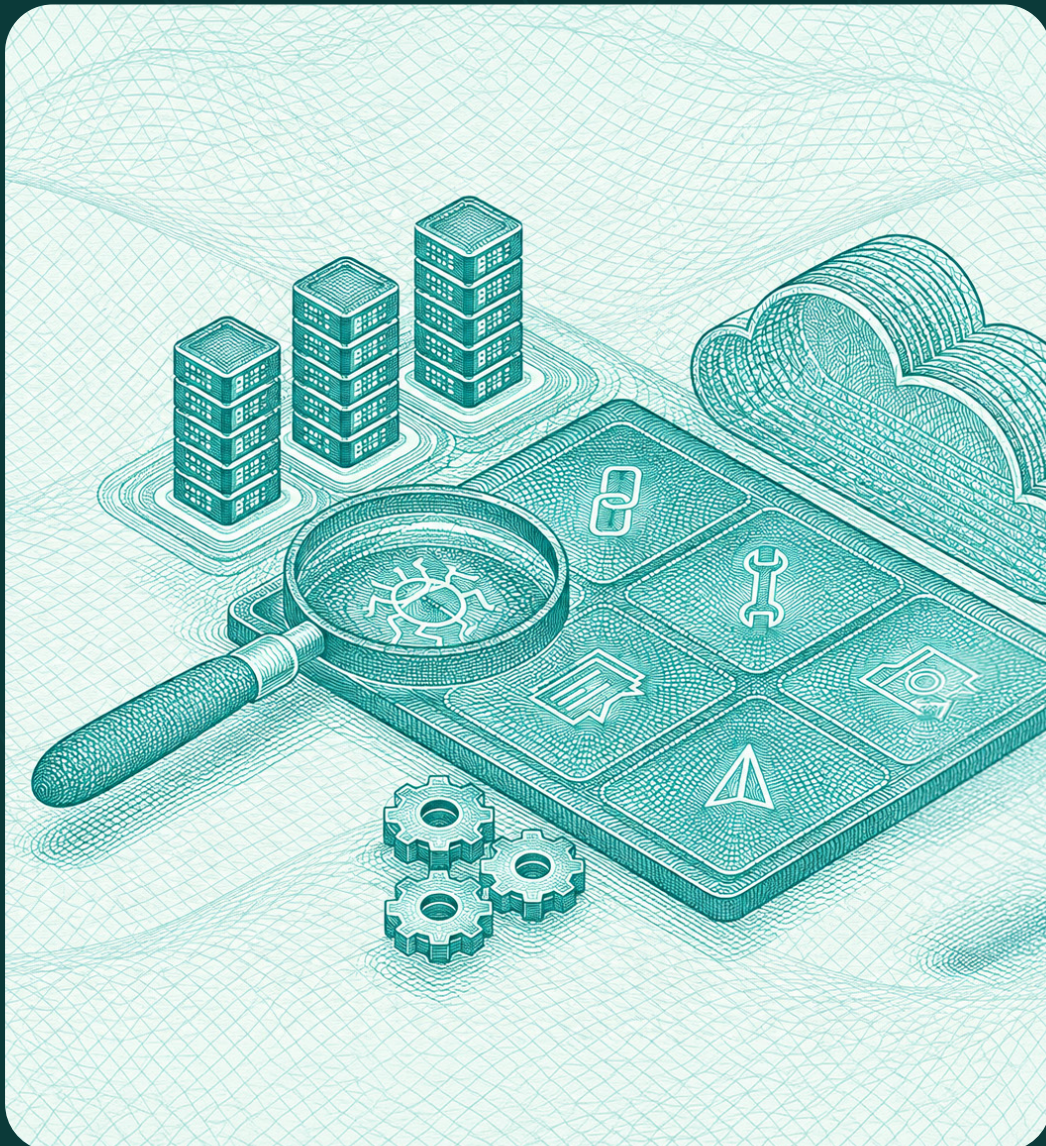




Customer Success Story

How SISA Helped a Global BPM Service
Provider Close a Critical SSRF Vulnerability
Through Forensics-Driven Application
Security Testing

About The Customer

The customer is a global digital engineering and business process management (BPM) company, headquartered in the United States, with delivery centers across India, North America, and Asia. With over two decades of experience, the organization partners with innovative, growth-oriented enterprises to solve complex challenges in digital transformation, product engineering, and operational efficiency. It delivers technology-driven solutions across key industries including Financial Services, Manufacturing, Aerospace, Education, and Emerging Markets.

Its core offerings include cloud modernization, intelligent automation, data engineering, enterprise platforms, and managed services. Backed by deep domain expertise and a collaborative approach, It enables clients to enhance agility, improve customer experiences, and achieve strategic business outcomes in a constantly evolving digital landscape.

Business Challenge



During a security assessment of a web application, a Server-Side Request Forgery (SSRF) vulnerability was identified. The application included a feature that accepted user-supplied URLs to fetch internal resources. Direct attempts to abuse this functionality using a Burp Collaborator URL were blocked by the application's input-validation mechanism, giving the impression that the feature was adequately protected.

This apparent safety, however, rested on a fragile control. If the validation logic could be circumvented, the same feature would allow an attacker to reach internal services, retrieve sensitive metadata, or pivot deeper into the internal network - and, by chaining SSRF with other weaknesses, escalate the impact significantly. The customer needed certainty about whether the control genuinely held, and precisely what was at stake if it did not.

Solution Offered by SISA

SISA carried out a forensic-driven security assessment of the web application, going beyond surface-level detection to reproduce the complete exploitation path and establish its real business impact. Rather than recording the SSRF as an isolated finding, the team studied how the application's input-validation control behaved under attack, engineered a working bypass, and traced exactly what an attacker could reach once that control was defeated. The engagement followed a deliberate, evidence-led sequence.

Phase 1



Functionality Mapping and Input-Flow Review

SISA began by examining the feature that accepted user-supplied URLs to fetch internal resources, tracing how each URL parameter was received, validated, and processed on the server. This mapped the exact point at which untrusted user input crossed into server-side request handling and marked the URL-fetch capability as a high-value target for request forgery.



Phase 2



Baseline Testing and Control Fingerprinting

To characterize the application's defenses, the team first attempted direct exploitation by submitting a Burp Collaborator URL designed to trigger an out-of-band callback. The request was rejected by the input-validation filter. Instead of treating that rejection as evidence of safety, SISA used it to fingerprint the control - determining how the filter inspected untrusted URLs, what it rejected, and where its logic was likely to be incomplete.



Phase 3



Filter Bypass and Controlled Exploitation

SISA established that the filter reasoned about the URL as submitted, without resolving where it ultimately pointed. Exploiting this gap, the team obfuscated the malicious payload behind a TinyURL shortened link, which the validation logic accepted as a benign address. When the application processed the request, it followed the shortened URL, which resolved to the attacker-controlled server and produced the expected out-of-band interaction - confirming a working SSRF that fully circumvented the input filter.

Phase 4



Forensic Impact Tracing and Attack-Path Analysis

With exploitation confirmed, SISA traced the vulnerability's downstream reach rather than stopping at proof of concept. The team assessed whether the same technique could be used to access internal-only services, internal network resources, and other non-routable assets normally shielded from external access. It also examined how internal responses could be combined with other weaknesses to perform network reconnaissance and potentially escalate from a single request-forgery flaw into broader network compromise.



Phase 5



Root-Cause Analysis and Defense-in-Depth Remediation

SISA attributed the issue to input validation that trusted the URL at face value, combined with an application permitted to make unrestricted outbound requests. The team delivered evidence, reproduction steps, and layered, prioritized remediation guidance.

SISA's 4-Point Remediation Guidance

01

Restrict outbound requests: Disable the URL-fetch feature where it is not essential, or limit it to an explicitly validated allowlist of trusted domains, preventing arbitrary calls to internal or sensitive IP ranges.

02

Validate URLs by destination, not appearance: Parse user-supplied URLs and resolve them server-side before use, blocking IP addresses, localhost references, and obfuscated or shortened links (such as those created with TinyURL) so the final destination is inspected rather than the submitted string.

03

Contain the blast radius: Apply outbound firewall rules and network segmentation so the application server cannot reach internal systems or metadata services, limiting the damage of any future SSRF attempt.

04

Disable automatic redirection and response rendering: Stop the application from silently following redirects or returning fetched content, closing avenues for data leakage and onward exploitation.

Business Impact

By identifying and fully validating the SSRF vulnerability before it could be abused, SISA helped the customer close a critical, exploitable gap in a web application. Left unaddressed, the flaw could have allowed an attacker to access internal services, extract sensitive data, or pivot deeper into the organization's infrastructure. It could have exposed instance metadata and credentials, opening the door to full environment compromise.

Key benefits included:



Prevented a high-impact security incident

Closing the SSRF path removed a realistic route to internal-service access, data exposure, and lateral movement before it could be exploited in the wild.



Protected internal network resources

Remediation and network-level restrictions reduced the risk of attackers abusing the application as a proxy to reach internal services and network resources that are otherwise inaccessible from the external network.



Contained future risk through defense-in-depth

Destination-based URL validation, segmentation, and disabled redirection ensured that even a partial control failure could no longer lead to full exploitation.



Strengthened secure-development practices

The root-cause analysis gave the team a reusable model for validating untrusted input by destination and governing outbound requests across its wider application portfolio.

Conclusion

Through this forensic-driven application security assessment, SISA moved beyond flagging a vulnerability to demonstrating precisely how it could be exploited and how far an attacker could go. By reproducing the filter bypass and tracing the full attack path, the team gave the organization both the evidence and the layered remediation needed to close the gap decisively. The result is a more resilient web application - one with input validation that inspects true destinations, tightly governed outbound access, and containment controls that limit the impact of any single flaw.