Vulnerability Analysis CVE-2026-55040 Microsoft SharePoint JWT Token Authentication Bypass

Author: Synthex **Site:** denizhalil.com CVSS 9.1 HIGH **Date:** August 2026

INTRODUCTION

In recent enterprise security developments, a severe authentication bypass vulnerability designated as **CVE-2026-55040** was disclosed in Microsoft SharePoint Server. Rated with a CVSS v3.1 base score of **9.1 (Critical)**, this flaw exposes on-premises SharePoint deployments to unauthenticated remote exploitation. Originally discovered by security researcher Stephen Fewer at Rapid7 Labs, the vulnerability lies deep within SharePoint's identity handling and JSON Web Token (JWT) validation infrastructure. Because SharePoint Server often acts as the central repository for sensitive corporate intelligence, document management, and internal workflows, an authentication bypass at this level grants malicious actors unrestricted access to organizational data assets. Furthermore, when combined with secondary vector chainings, CVE-2026-55040 allows complete environment takeover, posing immediate systemic risks to affected organizations that fail to rapidly deploy available patches.

LEARNING OBJECTIVES

By reviewing this vulnerability analysis report, cyber security professionals and system administrators will achieve the following objectives:

1. **Understand the Root Cause:** Comprehend the architectural defects in SharePoint's server-to-server (S2S) authentication and JWT validation mechanics.
2. **Analyze Attack Mechanics:** Trace how an unauthenticated remote attacker crafts forged tokens to impersonate legitimate users and administrators without credentials.
3. **Recognize Exploit Chaining Risks:** Evaluate how authentication bypass vulnerabilities act as initial primitives for unauthenticated Remote Code Execution (RCE).
4. **Implement Defense-in-Depth:** Deploy official Microsoft patches, perform post-installation configuration steps, and enforce log monitoring strategies for active threat hunting.

WHAT IS MICROSOFT SHAREPOINT JWT TOKEN AUTHENTICATION BYPASS (CVE-2026-55040)

CVE-2026-55040 represents a critical architectural weakness categorized under *CWE-1390: Weak Authentication*, severely impacting the core identity pipeline of Microsoft SharePoint Server. It specifically targets SharePoint's internal Server-to-Server (S2S) authentication framework, which relies

on JSON Web Tokens (JWT) to establish trusted communications and pass security context between SharePoint nodes, search crawlers, and integrated enterprise applications such as Microsoft Exchange or Workflow Manager. Under standard operational design, when a request reaches SharePoint presenting an OAuth 2.0 Bearer token, the system enforces strict validation sequences. The token handler parses the incoming structure to verify cryptographic signatures, issuer authenticity, audience targets, and identity claims—such as Active Directory Security Identifiers (SID) or User Principal Names (UPN). However, CVE-2026-55040 introduces a catastrophic failure in this sequence, allowing crafted tokens with invalid or omitted signatures to bypass validation routines while still allowing claim extraction.

As a consequence of this logical parsing error, remote attackers can craft specialized JWT payloads containing arbitrary administrative identities. Because the backend token handler processes the identity claims without ensuring signature integrity, the application impersonates high-privilege accounts automatically. This grants unauthenticated actors complete read, write, and administrative access across affected SharePoint web applications and tenant databases.

- **Unauthenticated Identity Impersonation:** Attackers can inject arbitrary user claims (such as Farm Administrator SIDs) into forged Bearer tokens without possessing valid secret keys or credentials.
- **Bypass of Cryptographic Checks:** Flaws in the underlying `Microsoft.SharePoint.IdentityModel` token handler cause signature verification checks to be skipped or ignored during S2S token evaluation.
- **Zero User Interaction Required:** Exploitation occurs strictly over the network at the protocol level, requiring no social engineering or user intervention.
- **High-Impact Chaining Potential:** Once authenticated as an administrator via token forge, attackers can access privileged administrative endpoints to achieve full Remote Code Execution (RCE) on host systems.

TECHNICAL DETAIL: HOW THE VULNERABILITY WORKS

The architectural defect resides within the `Microsoft.SharePoint.IdentityModel` namespace, specifically implemented in `SPJsonWebSecurityTokenHandlerV2` and its base class `SPJsonWebSecurityBaseTokenHandlerV2`. These components handle the parsing, validation, and identity claims extraction for Server-to-Server (S2S) OAuth 2.0 / JWT Bearer tokens. Under normal conditions, these classes verify the signature of both the outer token and an embedded "actor token" against trusted certificate stores before establishing user contexts. When SharePoint receives an S2S request, the HTTP `Authorization: Bearer <token>` header is passed directly to `SPApplicationAuthenticationModuleV2.TryExtractAndValidateToken()`, which delegates the logic to `SPJsonWebSecurityTokenHandlerV2.ValidateToken()`. The handler initializes `TokenValidationParameters` to evaluate token structure and key resolvers. However, critical implementation flaws within this method cause the handler to disable cryptographic enforcement while continuing execution.

The fundamental breakdown occurs because SharePoint configures the JWT validator to ignore missing signatures, allowing unsigned tokens with `alg: none` headers to be processed as valid requests. By supplying the SHA-1 thumbprint (`x5t`) of SharePoint's publicly readable Security Token Service (STS) certificate within an inner actor token, an attacker forces the server to resolve its own signing key locally. The handler then extracts identity claims directly from the unverified payload and synthesizes an administrative `SPUserToken`, fully impersonating the specified target identity.

- **Explicit Signature Bypass:** The handler explicitly sets `RequireSignedTokens = false` during token parameter initialization, allowing tokens with `alg: none` headers to pass validation without any signature check.
- **Unauthenticated Key Resolution:** The inner actor token leverages the public `x5t` certificate thumbprint obtained from endpoints like `/_layouts/15/metadata/json/1` to trick SharePoint into binding local signing keys without validating authority trust.
- **Direct Claim Extraction:** Subject claims (such as `nameid`, `upn`, or user SIDs) are parsed directly from the untrusted JWT payload and supplied to internal user-resolution routines.
- **Identity Thread Switching:** The authentication pipeline constructs a valid `IClaimsPrincipal` and calls `ConstructIClaimsPrincipalAndSetThreadIdentity()`, elevating the current HTTP execution context to administrative status.

```
// Decompiled View of SPJsonWebSecurityTokenHandlerV2 &
SPJsonWebSecurityBaseTokenHandlerV2
public override ReadOnlyCollection<ClaimsIdentity> ValidateToken(SecurityToken token)
{
    SPJwtSecurityToken sPJwtSecurityToken = token as SPJwtSecurityToken;
    TokenValidationParameters validationParameters = this.GetValidationParameters();
```

```
    // CRITICAL WEAKNESS: Cryptographic signature checks are explicitly disabled
    validationParameters.RequireSignedTokens = false;

    // Token claims are parsed without valid signature enforcement
    ClaimsIdentityCollection claimsIdentities =
base.ValidateToken(sPJwtSecurityToken, validationParameters);

    // Identity is synthesized directly from unverified actor token claims (e.g.,
nameid / user SID)
    string userSid = sPJwtSecurityToken.GetClaim("nameid");
    return IdentityProvider.CreateIdentityFromSid(userSid);
}
```

MITIGATION & DETECTION

Protecting enterprise Microsoft SharePoint deployments against CVE-2026-55040 requires an urgent combination of official patch application, post-installation database alignment, and active network defense strategies. Because this vulnerability acts as the initial authentication-bypass phase in chains that can lead to remote code execution (RCE), simply monitoring for successful logins is insufficient. Security operations teams must combine official vendor hotfixes with proactive threat hunting across Web Application log files, Windows Event logs, and Unified Logging System (ULS) traces to ensure early detection and comprehensive remediation.

- **Deploy Vendor Security Updates:** Immediately install the relevant Microsoft Security Updates issued during the July 2026 Patch Tuesday cycle for SharePoint 2016, 2019, or Subscription Edition.
- **Execute Product Configuration Wizard:** Run `psconfig.exe -cmd autoupdates-exec -cmd upgrade -inplace b2b -wait` across all farm nodes post-patching to properly apply binary modifications and database updates.
- **Inspect IIS Web Access Logs:** Audit web logs for unauthorized external requests to S2S endpoints (such as `/_vti_bin/client.svc` or `/_layouts/15/metadata/json/1`) containing Bearer tokens.
- **Filter Unauthenticated S2S Requests:** Implement perimeter Layer-7 reverse proxy filters or WAF rules to inspect HTTP `Authorization: Bearer` headers and block malformed JWT payloads containing `alg: none`.
- **Monitor SharePoint ULS Traces:** Query ULS logs for diagnostic messages from `Microsoft.SharePoint.IdentityModel` involving `SPJsonWebSecurityTokenHandlerV2` to identify anomalous claim extractions.
- **Enable AMSI Request Body Scanning:** Activate Antimalware Scan Interface (AMSI) integration in "Full Mode" for all SharePoint web applications to inspect incoming payloads for post-authentication exploit chaining.

CONCLUSION

CVE-2026-55040 serves as a stark reminder of the critical risks inherent in identity validation failures within modern enterprise infrastructure. As organizations increasingly rely on centralized platforms like Microsoft SharePoint to manage core data workflows, flaws in low-level authentication handlers become high-value targets for threat actors. This vulnerability highlights how small logical flaws in cryptographic verification routines can inadvertently dismantle entire perimeter defenses.

The severity of this issue is further amplified by its role as an initial access vector. By allowing unauthenticated actors to easily forge administrative JSON Web Tokens, CVE-2026-55040 provides a reliable entry point that can be chained with secondary administrative features. Consequently, an attacker can transition from simple token manipulation to full remote code execution, granting them persistent control over the host operating system and underlying corporate network.

Safeguarding infrastructure against this flaw requires immediate, decisive operational action. Applying Microsoft's July 2026 security updates across all SharePoint farms is only the first step. System administrators must also execute the SharePoint Products Configuration Wizard (`PSConfig`) on every farm node to ensure that binary patches, database schemas, and service dependencies are fully upgraded and properly bound.

Ultimately, addressing CVE-2026-55040 demands a comprehensive defense-in-depth posture. Beyond rapid patching, security teams must proactively audit IIS and ULS logs for anomalous S2S Bearer token activity, enforce strict perimeter filtering, and ensure robust runtime protections are active. Only through continuous monitoring and total patch compliance can organizations effectively neutralize active exploitation attempts and maintain the integrity of their enterprise data assets.