National Security Agency | Cybersecurity Advisory

Patch Critical Cryptographic Vulnerability in Microsoft Windows Clients and Servers

Summary

NSA has discovered a critical vulnerability (CVE-2020-0601) affecting Microsoft Windows®[1] cryptographic functionality. The certificate validation vulnerability allows an attacker to undermine how Windows verifies cryptographic trust and can enable remote code execution. The vulnerability affects Windows 10 and Windows Server 2016/2019 as well as applications that rely on Windows for trust functionality. Exploitation of the vulnerability allows attackers to defeat trusted network connections and deliver executable code while appearing as legitimately trusted entities. Examples where validation of trust may be impacted include:

- HTTPS connections
- Signed files and emails
- Signed executable code launched as user-mode processes

The vulnerability places Windows endpoints at risk to a broad range of exploitation vectors. NSA assesses the vulnerability to be severe and that sophisticated cyber actors will understand the underlying flaw very quickly and, if exploited, would render the previously mentioned platforms as fundamentally vulnerable. The consequences of not patching the vulnerability are severe and widespread. Remote exploitation tools will likely be made quickly and widely available. Rapid adoption of the patch is the only known mitigation at this time and should be the primary focus for all network owners.

Mitigation Actions

NSA recommends installing all January 2020 Patch Tuesday patches as soon as possible to effectively mitigate the vulnerability on all Windows 10 and Windows Server 2016/2019 systems. In the event that enterprise-wide, automated patching is not possible, NSA recommends system owners prioritize patching endpoints that provide essential or broadly replied-upon services. Examples include:

- Windows-based web appliances, web servers, or proxies that perform TLS validation.
- Endpoints that host critical infrastructure (e.g. domain controllers, DNS servers, update servers, VPN servers, IPSec negotiation).

Prioritization should also be given to endpoints that have a high risk of exploitation. Examples include:

- Endpoints directly exposed to the internet.
- Endpoints regularly used by privileged users.

Administrators should be prepared to conduct remediation activities since unpatched endpoints may be compromised. Applying patches to all affected endpoints is recommended, when possible, over prioritizing specific classes of endpoints. Other actions can be taken to protect endpoints in addition to installing patches. Network devices and endpoint logging features may prevent or detect some methods of exploitation, but installing all patches is the most effective mitigation.

Network Prevention and Detection

Some enterprises route traffic through existing proxy devices that perform TLS inspection, but do not use Windows for certificate validation. The devices can help isolate vulnerable endpoints behind the proxies while the endpoints are being

[1] Microsoft Windows is a registered trademark of Microsoft Corporation.

patched. Properly configured and managed TLS inspection proxies independently validate TLS certificates from external entities and will reject invalid or untrusted certificates, protecting endpoints from certificates that attempt to exploit the vulnerabilities. Ensure that certificate validation is enabled for TLS proxies to limit exposure to this class of vulnerabilities and review logs for signs of exploitation.

Packet capture analysis tools such as Wireshark can be used to parse and extract certificates from network protocol data for additional analysis. Software utilities such as OpenSSL and Windows certutil can be used to perform in-depth analysis of certificates to check for malicious properties.

Certutil can be used to examine an X509 certificate by running the following command:

- certutil –asn <certificate_filename>

OpenSSL can be used to examine an X509 certificate by running the following command:

- openssl asn1parse –inform DER –in <certificate_filename> –i –dump

or

- openssl x509 –inform DER –in <certificate_filename> –text

The commands parse and display the ASN.1 objects within a specified DER encoded certificate file. Review the results for elliptic curve objects with suspicious properties. Certificates with named elliptic curves, manifested by explicit curve OID values, can be ruled benign. For example, the curve OID value for standard curve nistP384 is 1.3.132.0.34. Certificates with explicitly-defined parameters (e.g., prime, a, b, base, order, and cofactor) which fully-match those of a standard curve can similarly be ruled benign.

Certutil can be used to list registered elliptic curves and view their parameters by running the following commands:

- certutil –displayEccCurve
- certutil –displayEccCurve <curve_name>

OpenSSL can be used to view standard curves enabled/compiled into OpenSSL by running the following commands:

- openssl ecparam –list_curves
- openssl ecparam –name <curve_name> –param_enc explicit –text

Certificates containing explicitly-defined elliptic curve parameters which only partially match a standard curve are suspicious, especially if they include the public key for a trusted certificate, and may represent bona fide exploitation attempts.

Disclaimer of Endorsement

The information and opinions contained in this document are provided "as is" and without any warranties or guarantees. Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement, recommendation, or favoring by the United States Government, and this guidance shall not be used for advertising or product endorsement purposes.

Contact

Cybersecurity Requirements Center: 410-854-4200, Cybersecurity_Requests@nsa.gov

Media inquiries: Press Desk, 443-634-0721, MediaRelations@nsa.gov