

## ManageEngine CVE-2022-47966 Vulnerability

<https://campus.barracuda.com/doc/98220256/>

**Severity:** 9.8 Critical | RCE | CVSS Attack Vector: Network

### Exploit

CVE-2022-47966 is an unauthenticated RCE vulnerability, and it affects Zoho's ManageEngine product portfolio. The vulnerability is a pre-authentication remote code execution (RCE). This CVE is exploitable based on the ManageEngine product and the state of SAML single-sign-on in the current or previous configuration state in certain conditions.

This happens due to the use of Apache "xmlsec" (aka XML Security for Java) 1.4.1. The exploitation is devised based on the vulnerable third-party dependency on Apache Santuario. In some cases, the system will only be vulnerable if SAML-based SSO is currently active.

As a best practice, follow the vendor advisory:

<https://www.manageengine.com/security/advisory/CVE/cve-2022-47966.html>

### Barracuda WAF Mitigation

The Barracuda Web Application Firewall protects against this attack with the help of the suggested configuration object.

### Action Required

1. Ensure **Enable Parameter Protection** is set to *Yes* on the **SECURITY POLICIES > Parameter Protection** page or the **Status** of Parameter Profiles is set to *On* on **WEBSITES > Website Profiles**.
2. Set **Base64 Decode Parameter Value** to *Yes*.
3. Ensure the **Blocked Attack Types** are selected, especially "OS command injection" on the **SECURITY POLICIES > Parameter Protection** page or the appropriate parameter class has the blocked attack types for parameter profiles.

#### Parameter Protection:

**Parameter Protection**Help

Enable Parameter Protection:

☒ Yes ☐ No

Denied Metacharacters:

Maximum Parameter Value Length:

Maximum Instances:

Base64 Decode Parameter Value:

☐ Yes ☒ No

Validate Parameter Name:

☐ Yes ☒ No

Allowed File Upload Type:

☒ Extensions ☐ Mime Types

File Upload Extensions:

Maximum Upload File Size:

Blocked Attack Types:

☐ Select/Deselect All

☒ SQL Injection

☒ OS Command Injection

☒ Directory Traversal

☒ Cross-Site Scripting

☒ Remote File Inclusion

☐ SQL Injection strict

☐ OS Command Injection strict

☐ Directory Traversal strict

☐ Cross-Site Scripting strict

☒ Remote File Inclusion strict

☒ LDAP Injection

☒ Python-PHP attacks

☒ HTTP Specific Injection

☒ Apache Struts attacks

☐ Apache struts attacks strict

☐ Spam URL List

Custom Blocked Attack Types:

N/A

Exception Patterns:

Checks individual parameter values in query strings and POST parameters. These settings are ignored when Parameter Profiles are used for validating the incoming requests. **Recommended:** Yes

The meta-characters to be denied in this parameter value. Meta-characters must be URL encoded. **Recommended:** %00%04%1b%08%7f

The maximum allowed length of a parameter value. **Recommended:** 1000

Specify the maximum number of times the parameter should be allowed in the request. Restricting this to 1 would help avoid a large class of HTTP Parameter Pollution attacks. Leave blank for unlimited instances. **Recommended:** 1

Set to Yes to apply base64 decoding to the parameter values.

Set to Yes to validate the parameter names in the request against the attack types selected in **Blocked Attack Types** and **Custom Blocked Attack Types**.

Select **Extensions** to allow the files uploaded with extensions specified in **File Upload Extensions**.  
Select **Mime Types** to identify the content in the files before allowing to be uploaded with the mime types specified in **File Upload Mime Types**.

Extensions that are allowed as uploaded files. Use a "." to indicate a file with no extension, and use a "\*" to indicate any kind of file extension. The setting is case insensitive, so "JPG" will allow .JPG as well as .jpg files. **Recommended:** Media files that may not contain scripts, like .JPG, .GIF, .PDF etc.

The maximum size (in KB) for an individual file that can be uploaded in a request. **Recommended:** 1024

Attack Types are malicious patterns that can be checked for in the parameter's value. **Note:** Each security policy is configured with default set of attack types that are applied to the matching requests. For more comprehensive validation, select other attack type patterns.

List of custom attack types as defined on the **ADVANCED > Libraries** page.

File Extensions	Actions
GIF	Delete
JPG	Delete
PDF	Delete

Exception Patterns	Actions
--------------------	---------

## Parameter Profile:

ManageEngine CVE-2022-47966 Vulnerability

2 / 4

Define a fixed set of strings to match against the parameter's value, if the parameter **Type** is set to **Global Choice**.

Parameter Class:

Select a parameter class to be compared to the parameters sent in the requests/responses.

**Parameter Class Details:**  
Parameter Class: multibyte  
Input Type Validation:  
Custom Input Type Validation:  
Denied Metacharacters: %00%01%04%1b%08%7f  
Blocked Attack Types: SQL Injection, Cross-Site Scripting, OS Command Injection, LDAP Injection, Python-PHP attacks, HTTP Specific Injection, Apache Struts attacks, Directory Traversal  
Custom Blocked Attack Types:

Custom Parameter Class:

Select the custom parameter class to be compared to the parameters sent in the requests/responses. This is applicable only when **Parameter Class** is set to **CUSTOM**.

Max Value Length:

Set the maximum allowable length for the value of the parameter. Example: The parameter "p2" set to 0, which means:  
p1=v1&p2=&p3=v2 : allowed  
p1=v1&p2=v&p3=v2 : not allowed  
No value indicates unlimited.

Required: ☒ No ☐ Yes

Set to **Yes** if the parameter must always be present in the request.

Ignore: ☒ No ☐ Yes

Set to **Yes** if the parameter must be ignored completely, that is, never validate the value of the parameter at all.

Validate Parameter Name: ☒ No ☐ Yes

Set to **Yes** to validate the parameter names in the request against the attack types selected in **Blocked Attack Types** and **Custom Blocked Attack Types**.

Maximum Instances:

Specify the maximum number of times the parameter should be allowed in the request. Restricting this to 1 would help avoid a large class of HTTP Parameter Pollution attacks. Leave blank for unlimited instances. **Recommended: 1**

Base64 Decode Parameter Value: ☐ No ☒ Yes

Set to **Yes** to apply base64 decoding to the parameter values.

**Related Articles:**

- <https://www.manageengine.com/security/advisory/CVE/cve-2022-47966.html>
- <https://github.com/projectdiscovery/nuclei-templates/pull/6564/files#diff-782cd8c119b60b37026742310fd381c791b4f2e6d2749b4da090e71ea9ba3693>
- <https://github.com/horizon3ai/CVE-2022-47966>

## Figures

1. Parameter\_Protection.png
2. Parameter Class.png

© Barracuda Networks Inc., 2024 The information contained within this document is confidential and proprietary to Barracuda Networks Inc. No portion of this document may be copied, distributed, publicized or used for other than internal documentary purposes without the written consent of an official representative of Barracuda Networks Inc. All specifications are subject to change without notice. Barracuda Networks Inc. assumes no responsibility for any inaccuracies in this document. Barracuda Networks Inc. reserves the right to change, modify, transfer, or otherwise revise this publication without notice.