Talos Vulnerability Report

TALOS-2020-1073

OS4Ed openSIS email parameter SQL injection vulnerability

AUGUST 31, 2020

CVE NUMBER

CVE-2020-6123, CVE-2020-6124

Summary

An exploitable sql injection vulnerability exists in the email parameter functionality of OS4Ed openSIS 7.3. A specially crafted HTTP request can lead to SQL injection. An attacker can make an authenticated HTTP request to trigger this vulnerability.

Tested Versions

OS4Ed openSIS 7.3

Product URLs

https://opensis.com/

CVSSv3 Score

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

CWE

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

Details

openSIS is a student information system and school management system. It is available in commercial and open-source versions. It allows schools to create schedules and track attendance, grades and transcripts.

CVE-2020-6123 - EmailCheck.php

The `email` parameter in the page `EmailCheck.php` is vulnerable to SQL injection.

Below is an example request that will trigger the vulnerability:

```
GET /opensis/EmailCheck.php?email=1[SQLINJECTION]&p_id=0 HTTP/1.1
Host: [IP]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:74.0) Gecko/20100101 Firefox/74.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-GB,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 0
Origin: http://[IP]
DNT: 1
Connection: close
Referer: http://[IP]/opensis/Modules.php?modname=eligibility/Student.php&modfunc=add&start_date=
Cookie: miniSidebar=0; PHPSESSID=6chg16qcanbg3adrqlq6sm6fa3
Upgrade-Insecure-Requests: 1
```

The vulnerable code for this parameter is at lines 36-44:

```
 34             if($_REQUEST['p_id']==0)
 35             {
 36                 $result=DBGet(DBQuery('SELECT STAFF_ID FROM people WHERE EMAIL=\''.$_REQUEST['email'].'\''));
 37                 $res_stf=DBGet(DBQuery('SELECT STAFF_ID FROM staff WHERE EMAIL=\''.$_REQUEST['email'].'\''));
 38                 $res_stu=DBGet(DBQuery('SELECT STUDENT_ID FROM students WHERE EMAIL=\''.$_REQUEST['email'].'\''));
 39             }
 40             else
 41             {
 42                 $result=DBGet(DBQuery('SELECT STAFF_ID FROM people WHERE EMAIL=\''.$_REQUEST['email'].'\' AND 
STAFF_ID!='.$_REQUEST['p_id']));
 43                 $res_stf=DBGet(DBQuery('SELECT STAFF_ID FROM staff WHERE EMAIL=\''.$_REQUEST['email'].'\''));
 44                 $res_stu=DBGet(DBQuery('SELECT STUDENT_ID FROM students WHERE EMAIL=\''.$_REQUEST['email'].'\''));
 45             }
```

CVE-2020-6124 - EmailCheckOthers.php

The `email` parameter in the page `EmailCheckOthers.php` is vulnerable to SQL injection.

Below is an example request that will trigger the vulnerability:

```
GET /opensis/EmailCheckOthers.php?email=1[SQLINJECTION]&id=0&type=3 HTTP/1.1
Host: [IP]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:74.0) Gecko/20100101 Firefox/74.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-GB,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 0
Origin: http://[IP]
DNT: 1
Connection: close
Referer: http://[IP]/opensis/Modules.php?modname=eligibility/Student.php&modfunc=add&start_date=
Cookie: miniSidebar=0; PHPSESSID=6chg16qcanbg3adrqlq6sm6fa3
Upgrade-Insecure-Requests: 1
```

The vulnerable code for this parameter is at line 36-42:

```
 36                 if($_REQUEST['id']==0)
 37                 $result_stu=DBGet(DBQuery('SELECT COUNT(1) as EMAIL_EX FROM students WHERE EMAIL=\''.$_REQUEST['email'].'\''));
 38                 else
 39                 $result_stu=DBGet(DBQuery('SELECT COUNT(1) as EMAIL_EX FROM students WHERE EMAIL=\''.$_REQUEST['email'].'\' AND
STUDENT_ID!='.$_REQUEST['id']));
 40
 41                 $result_pe=DBGet(DBQuery('SELECT COUNT(1) as EMAIL_EX FROM people WHERE EMAIL=\''.$_REQUEST['email'].'\''));
 42                 $result_stf=DBGet(DBQuery('SELECT COUNT(1) as EMAIL_EX FROM staff WHERE EMAIL=\''.$_REQUEST['email'].'\''));
```

Timeline

2020-06-02 - Vendor Disclosure

2020-08-13 - Vendor provided patch to Talos for testing

2020-08-17 - Talos confirmed patch resolved issue

2020-08-31 - Public Release

CREDIT

Discovered by Yuri Kramarz of Cisco Talos.

VULNERABILITY REPORTS

PREVIOUS REPORT

NEXT REPORT

TALOS-2020-1072

TALOS-2020-1074