Talos Vulnerability Report

TALOS-2022-1569

Abode Systems, Inc. iota All-In-One Security Kit telnet hard-coded password vulnerability

OCTOBER 20, 2022

CVE NUMBER

CVE-2022-29889

SUMMARY

A hard-coded password vulnerability exists in the telnet functionality of Abode Systems, Inc. iota All-In-One Security Kit 6.9Z. Use of a hard-coded root password can lead to arbitrary command execution. An attacker can authenticate with hard-coded credentials 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 6.9Z

PRODUCT URLS

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

CVSSV3 SCORE

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

CWE

CWE-798 - Use of Hard-coded Credentials

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.

Version 6.9Z of the `iota` firmware exposes a telnet service on TCP/55023. The password for the root user of each device is derived from the MAC address of the device. This derivation is easily repeatable and can be conducted off-device in order to recover the password of an arbitrary device.

The business-logic of password-derivation and updates to `/etc/shadow` occur within the `/root/hpgw` binary, in a function that we will refer to in this report as `update_root_password`. In version 6.9Z this is located at offset 0xA7BA0 and in 6.9X at offset 0xA7ADC. The actual calculation occurs within a function we will refer to in this report as `keygen` [6.9Z: 0xA7A48, 6.9X: 0xA7984].

```
unsigned int __fastcall keygen(
    char *ciphertext,
    const char *mac_addr,
    int prefix,
    int postfix,
    int map_1,
    int map_2,
    int map_3,
    int map_4,
    int map_5,
    int map_6,
    int map_7,
    int map_8)
{
    uint8_t sha256_result[32];
    char plaintext[64];

    vsnprintf_nullterm(plaintext, 0x3Fu, "%d%s%d", prefix, mac_addr, postfix);
    sha256(plaintext, 20, sha256_result, 0);
    ciphertext[0] = g_lookup_table[sha256_result[map_1] % 0x30u];
    ciphertext[1] = g_lookup_table[sha256_result[map_2] % 0x30u];
    ciphertext[2] = g_lookup_table[sha256_result[map_3] % 0x30u];
    ciphertext[3] = g_lookup_table[sha256_result[map_4] % 0x30u];
    ciphertext[4] = g_lookup_table[sha256_result[map_5] % 0x30u];
    ciphertext[5] = g_lookup_table[sha256_result[map_6] % 0x30u];
    ciphertext[6] = g_lookup_table[sha256_result[map_7] % 0x30u];
    ciphertext[7] = g_lookup_table[sha256_result[map_8] % 0x30u];
    ciphertext[8] = 0;
}
```

The result of this calculation is stored in to the `ciphertext` variable, which the `update_root_password` function will use to calculate the md5crypt and update the `/etc/shadow` file contents with the modified password. This appears to occur once on every execution of the `hpgw` binary.

While this password generation functionality existed within version 6.9X, the telnet server had been disabled and the root password was not used anywhere else. With the release of 6.9Z, the telnet server has been re-enabled and this vulnerability became exploitable again.

Knowledge of the hard-coded `prefix`, `postfix`, `map_#`, and `g_lookup_table` values and the MAC address of the device is all it takes to calculate the root password for the device and successfully authenticate via telnet.

TIMELINE

2022-07-13 - Initial Vendor Contact
2022-07-14 - Vendor Disclosure
2022-10-20 - Public Release

CREDIT

Discovered by Matt Wiseman of Cisco Talos.
