MWR LABS
Security Advisory

FingerTec and ZKTeco Unauthenticated Remote Enrollment and Disclosure Vulnerability

10/01/2017

Hardware	ZKTeco based Fingerprint Access Control Devices
Affected Versions	FingerTec AC900, M2, R2 have been tested.
CVE Reference	N/A
Author	Daniel Lawson
Severity	High
Vendor	ZKTeco, FingerTec
Vendor Response	Issue reported fix, contact vendor

Description:

ZKTeco based FingerTec access control devices allow remote unauthenticated enrollment by any malicious user who has network access to the device. Additionally, the devices disclose the user IDs, PINs, RFID numbers, and names of any enrolled users by default without authentication.

Impact:

An attacker with network can grant themselves the ability to bypass the physical security implemented by the FingerTec systems.

Cause:

While the device can use authentication, it is disabled by default, and relatively weak. The device does not use any sort of encryption. Additionally, the user manual for the AC900s and R2/M2 readers located at:

http://www.fingertec.com/customer/download/postsales/HUM-AC900R2-E.pdf

Explicitly recommend setting the COMM Key to 0. The documentation for Ingress located at:

http://www.fingertec.com/customer/download/postsales/SUM-Ingress-E.pdf

Recommends setting a weak key (5 character).

Interim Workaround:

The devices should be either disconnected from the network, or segregated so that nothing may access them except for the Ingress Server. The devices should also have the COMM Key set to a 6 digit number. This in and of itself is not sufficient to protect the device though. For an AC900, the entire keyspace can be brute forced in under 3 days.

Solution:

While there is authentication available for the device, there are no countermeasures in place to prevent brute force attacks, and the key space is relatively small (only 999,999 entries). Ideally, when the device is activated with either an Ingress server or a TCMS server, the server should set a random 24-bit password. This would still work with the current password hashing scheme in use by the API. The device should provide some sort of brute force protection, such as stepping off the response times after X number of failed authentication attempts, or locking out an IP for a certain amount of time after a set number of failed authentication attempts. Additionally, the devices should use strong encryption when communicating with the server.

Technical details

The command 1501 creates an entry in the user.dat file, putting exactly what is positioned on the command line into the database.

The command 1503 gets the user.dat sent back in its entirety. This data is trivially to parse and extract out sensitive information. Below is the output from the tool built as a proof of concept:

```
17:25:14 ~/src/fingertec$ ./fingertec-tool.py create_user 192.168.100.200
Connecting...
User created successfully.
User ID:1337
PIN:1337
RFID:0
User Name:haxx0r

 17:26:10 ~/src/fingertec$ ./fingerpick.py list_users 192.168.100.200
```

```
Connecting...
| ID         | Privilege | PIN    | RFID        | Username |
|-----------+-----------+-------+------------+----------|
| 1004       | 6         | 1234   | 2347918     | FakeUser |
| 1337       | 6         | 1337   | 0           | haxx0r   |
```

Detailed Timeline

Date	Summary
2016-07-07	Vulnerability discovered
2016-07-08	FingerTec contacted
2016-07-20	Advisory updated with additional authentication information
2016-08-02	ZKTeco Contacted
2016-08-28	Fixed released to FingerTec Engineers
2017-01-10	Advisory Published