ethanhunnt / IoT_vulnerabilities Public

Code Issues 1 Pull requests Actions Projects Security

master

IoT_vulnerabilities / Depstech Microscope Smart Kid Toy.pdf

ethanhunnt New 2020 CVEs uploaded ... History

1 contributor

325 KB

Depstech Wireless Microsocope Smart Toy

- ## Introduction

  According to Wikipedia[1], “A smart toy is a toy which effectively has its own intelligence by virtue of on-board electronics. These enable it to learn, behave according to pattern, and alter its actions depending upon environmental stimuli. Typically, it can adjust to the abilities of the player. A modern smart toy has electronics consisting of one or more microprocessors or microcontrollers, volatile and/or non-volatile memory, storage devices, and various forms of input–output devices.[1] It may be networked together with other smart toys or a personal computer in order to enhance its play value or educational features.[2][3] Generally, the smart toy may be controlled by software which is embedded in firmware or else loaded from an input device such as a USB flash drive, Memory Stick or CD-ROM.[4] Smart toys frequently have extensive multimedia capabilities, and these can be utilized to produce a realistic, animated, simulated personality for the toy. Some commercial examples of smart toys are Amazing Amanda, Furby and iDog”

  In this article, we are going to discuss the methods used by attackers as well as other security researchers to find security issues in Depstech Wireless Microscope. The chapter will give a quick introduction on the techniques used to exploit this toy[2]. The chapter will talk about how the security researcher obtained the device’s firmware, how the firmware was dissected to obtain necessary files and finally a quick introduction on some of the vulnerabilities that were identified in this process. We started our research by looking at smart toys on the Internet. A wireless microscope by Depstech caught our attention. We were surprised by the number of these devices being sold on Amazon for prices that vary from 10$ to 50$. This drove our attention to this specific device. And the rest you can find out for yourself!!

[1] https://en.wikipedia.org/wiki/Smart_toy
[2] https://www.amazon.com/gp/product/B07JN17KP2

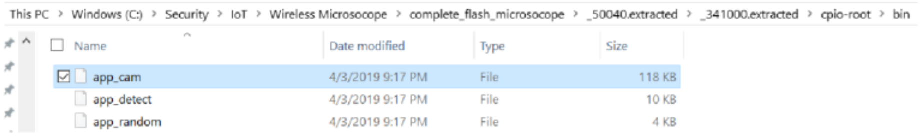
- ## Depstech Exploits

  In the next section we are going to work towards various issues that were identified in the Shekar borescope.

  - ### Network Daemon (App Camera)

    As a part of security mapping earlier, we decided to focus on network daemons that the device was exposing on its own wireless network. After performing NMAP[3] scans we identified that the device exposed a telnet server. We focused first on the telnet server binary that manages the telnet daemon but soon found out that we were dealing with standard busybox implementation. We could have focused on that specifically but being opensource busybox has gone through numerous security audits and hence we would have less chance of success with that.

    Next, we identified that there was A UDP daemon "app_camera" running on port "50000". So, we decided to focus on that.



This PC > Windows (C:) > Security > IoT > Wireless Microsocope > complete_flash_microsocope > _50040.extracted > _341000.extracted > cpio-root > bin

Name	Date modified	Type	Size
app_cam	4/3/2019 9:17 PM	File	118 KB
app_detect	4/3/2019 9:17 PM	File	10 KB
app_random	4/3/2019 9:17 PM	File	4 KB

app_cam Server binary

We decided to focus on identifying common security issues that plague the network daemons in case of embedded devices and would allow attacker to take complete control of the device. We identified this specific component suffered from: