Virtual Honeypots for Infusion Pump Systems

Client & SME: Matthew Wilkes, INOVA

Anthony Tate, David Nguyen, Randy Maysaud, Ronan Roque, Salma Almaz

Volgenau School of Engineering, Cyber Security Engineering, George Mason University

INTRODUCTION

- Infusion pumps are medical devices used to deliver fluids to a patient's body
- Previously standalone devices, now communicate in the network
- The increased connectivity also increased the attack surface for cyber attacks
- There is a need to defend these medical devices
- One form of defense are *honeypots* which are used to divert attackers from harming the legitimate systems

Figure 1: Baxter SIGMA Spectrum Infusion Pump

APPROACH

- Developed honeypot to mimic Baxter SIGMA Spectrum Infusion Pump
- Uses open source software with minimal cost to the organization
- Pre-existing vulnerabilities such as weak authentication and hard-coded passwords were previously found on the infusion pump
- Raspberry Pi placed as a server on INOVA's network to deploy virtual honeypots
- Information generated from the honeypots will be logged and sent to processing by the Risk Management team
- After successful installation, the organization will be able to deter attackers from legitimate systems and identify malicious behavior on their network

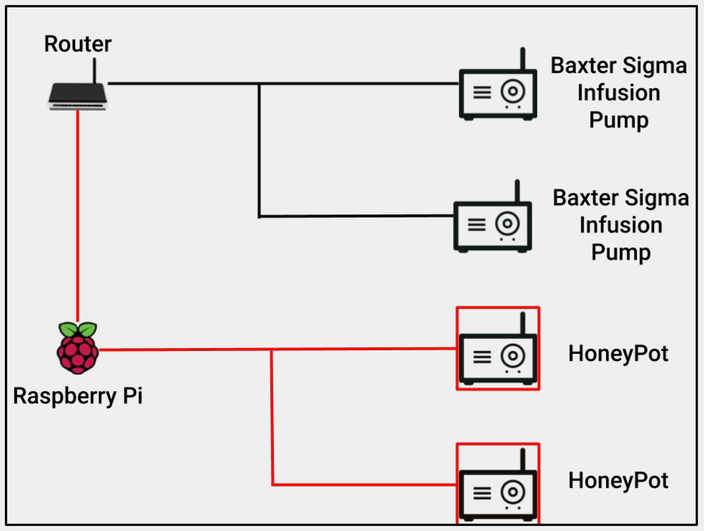

Figure 2: Virtual Honeypot Deployment

METHODOLOGY

- Create a virtual honeypot server that can automatically inject virtual Baxter SIGMA Spectrum Infusion Pump honeypots into the network.
- HoneyD is the software used to create the honeypot server,
  - the software will be installed in a device that will act as the head of the server.
  - The device that holds the software will be able to configure and deploy the virtual honeypots, as well as host any information gained from the honeypots.
- Set up a raspberry pi as a host device:
  - The Raspberry Pi will hold the Honeyd software and control all the virtual honeypots that they release into the network. To release the virtual honeypots the Raspberry Pi must be connected to the router/switch of the network.
  - A useful aspect of Honeyd is that it has log feature, if an actor accesses one of the virtual honeypots it will log the attacker's actions and send the log back to the host which will be the Raspberry Pi.
- Deployment of the virtual honeypot:
  - The Raspberry Pi must be connected on the existing network,
  - The design will relies on a Raspberry Pi running the Honeyd software that will generate network traffic between the virtual honeypot and the infusion pump to emulate connectivity
  - The virtual honeypot is not physically connected to the network but the traffic is being routed by the software running on the Raspberry Pi.
  - That allows the administrator to shut down the virtual honeypot with ease and have a central device that holds all the logs that the honeypot generates which can then be analyzed by the security team.

Figure 3: Honeypot HTTP Service

```
--MARK--,"Fri 10 Apr 2020 03:51:04 PM HDT","apache/HTTP","","",,,
"GET /favicon.ico HTTP/1.1
Host: 192.168.1.75
Connection: keep-alive
User-Agent: Mozilla/5.0 (X11; Linux i686) AppleWebKit/537.36 (KHTML, like Gecko
) Chrome/80.0.3987.162 Safari/537.36
Accept: image/webp,image/apng,image/*,*/*;q=0.8
Referer: http://192.168.1.75/
```

Figure 4: Honeypot HTTP Service Log

RESULTS

- In this project we implemented the methodology using a Raspberry Pi with HoneyD on a home network. On the home network we tested the information logging of the virtual honeypots, by breaching into the honeypot and matching the logs with the commands inputted. With the success of the proof of concept, it displays that this method works, a positive aspect of HoneyD is that it is incredibly scalable.
- Our Honeypot represent a Linux 2.2.13 (SuSE 6.3) and four open ports that each represent and log a specific service, reference Figure 3,4, and 5.

```
PORT   STATE SERVICE
21/tcp open  ftp
22/tcp open  ssh
23/tcp open  telnet
80/tcp open  http
MAC Address: 00:00:24:1B:A4:86 (Connect AS)
Device type: general purpose
Running: Linux 2.2.X
OS CPE: cpe:/o:linux:linux_kernel:2.2.13
OS details: Linux 2.2.13 (SuSE 6.3)
```

Figure 5: Honeypot Ports and OS

CONCLUSION

- This project aimed to develop a virtual honeypot solution for the Baxter SIGMA Spectrum Infusion Pump
- Virtual honeypots were deployed using Honeyd
  - Allows for easy configuration and management of the honeypots
- Although testing was done on a Raspberry Pi, the software can be run on heavier hardware
  - Solution is very scalable
- Currently the solution is just for the Baxter SIGMA Spectrum Infusion Pump, but it can be easily configurable to replicate other devices

ACKNOWLEDGEMENTS

We would like to thank INOVA for sponsoring this project and especially Matthew Wilkes who has supported our team throughout this project. Furthermore, we would also like to thank Dr. Manzo for mentoring our team.

REFERENCES

[1]https://github.com/DataSoft/Honeyd
[2]https://www.baxter.com/sites/g/files/ebysai746/files/styles/portrait_image/public/2018-06/spectrum-iq-infusion-system.png?itok=KkmIHF1A
[3]Mahajan, S. (2020). Intrusion Detection System Using Raspberry PI Honeypot in Network Security. [online] Pdfs.semanticscholar.org. Available at: https://pdfs.semanticscholar.org/9c2e/b0a9817be134c28c73c24a73600dd1cd15b8.pdf [Accessed 3 Mar. 2020].
[4]http://www.hackinsight.org/news,90.html

April 27, 2020