Exploitations of Wireless Interfaces via Network Scanning

Bryan Pearson
Computer Science
Stetson University
peabryan95@gmail.com

Nathalie Domingo
Electrical and Computer Engineering
Carnegie Mellon University
ndomingo@Andrew.cmu.edu

Kelvin Ly, Kaveh Shamsi, and Orlando Arias
Electrical and Computer Engineering (ECE)
University of Central Florida
(rangertime, kaveh, oarias)@knights.ucf.edu

Dr. Yier Jin and Dr. Shaojie Zhang
ECE and Computer Science
University of Central Florida
yier.jin@eecs.ucf.edu and shzhang@cs.ucf.edu

Abstract

In brainstorming for ways to exploit Internet of Things (IoT) security, we envisioned the following premise: an attack which compromises the security of multiple wireless interfaces. In exposing such data, we believe an attacker could plan a strategic attack towards a network that is much more active in nature. Goals of the attack:

- Expose network location, security, and activity.
- Visualize data.
- Affect as many networks as desired.
- Remain undetected.

Related Terms

- **Comma Separated Values (CSV)**
  - Text file in which each line is a data record
- **Keyhole Markup Language (KML)**
  - XML notation that can be read by Google Earth
- **Address Resolution Protocol (ARP)**
  - Protocol that maps an IP address to a local MAC address

Tools

- **Raspberry Pi 3 Model B v1.2**
  - Execute script; compile and upload data output
- **Ultimate GPS Breakout v3**
  - Track current location; accurate within 3 meters
- **Phantom 3 Advanced**
  - Combine with the previous tools for quick transportation

Figure 1: Raspberry Pi 3

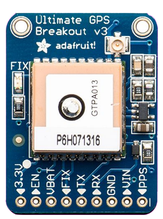

Figure 2: Ultimate GPS Breakout

Procedure

Attack outline:

- Execute script, output data to CSV
- Convert CSV to KML
- Overlay KML data with Google Earth for visualization

We start the attack by executing the data retrieval script, written in Python on the Raspberry Pi. The Pi is connected to our GPS module and can quickly obtain the coordinates of our current location.

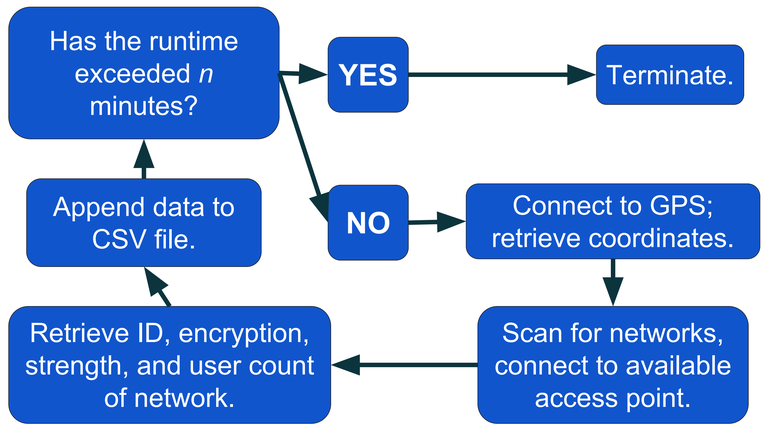

Figure 3: Network & GPS data collection

Once we output the CSV file, we execute another Python script that converts CSV to KML. This script also calculates the average user count and signal strength of each network.

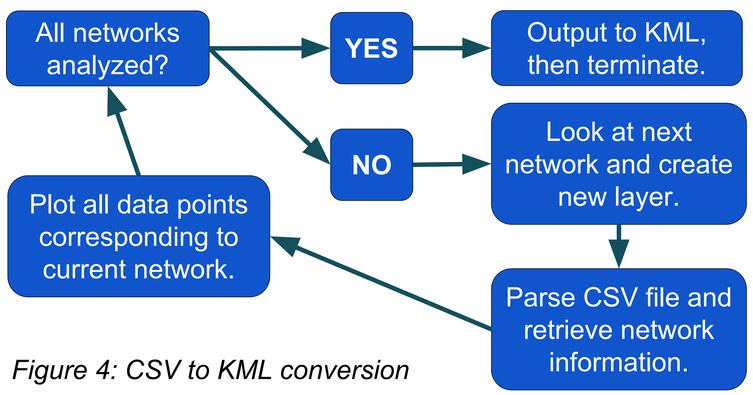

Figure 4: CSV to KML conversion

We then import the KML file into Google Earth in order to generate a map of our data.

Results

We successfully tested the attack at the University of Central Florida by generating a map which contains information on many of the networks found on campus. However, we were unable to thoroughly collect data on network activity due to deficiencies with ARP.

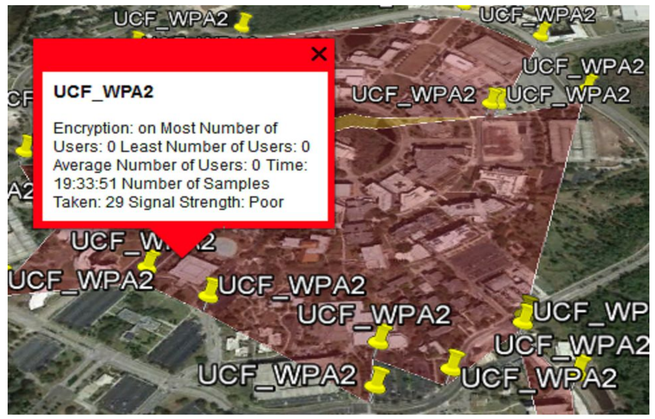

Figure 5: Network data visualization via KML

Future Work

The attack utilizes ARP scanning to analyze network activity. Unfortunately, the ARP protocol is slow, frequently inaccurate, and easily detectable by networks. To circumvent these issues, we propose a solution that forgoes ARP in favor of WireShark, a protocol that reliably analyzes network activity on the physical layer.

Acknowledgement

The support for this work was provided by the National Science Foundation REU program under Award No. 1560302. Any opinions, findings, and conclusions and recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.