Eyes on the Network

AUTHORS:
Paul Gurke

Technical Field

This project involves the technical fields of Network Security and Forensics.

Background Information

This project was brought about by the desire to reveal meaning to the unseen information moving across a private network. I have worked on multi-million dollar projects that hit delays due in part to regular network misconfigurations that were revealed by inspecting network traffic. The routine rouge device hunt became so frequent that I wanted to build a tool that could provide the information necessary to rectify network related issues to someone less technically adept.

Prior Art

Devices such as Bitdefender Box (*Bitdefender BOX - Home Network Security for All Connected Devices*, n.d.) and Firewalla (*Firewalla*, n.d.) exist as an internet connected solution and are often found with a subscription-based service model. The concept I have could operate independent from the world wide web. Free alternatives like Wireshark (*Wireshark · Go Deep.*, n.d.) require more knowledge and skill than even many professionals possess to effectively use the tool. The core principle is the ability to automate tasks that are dependent on observable network data.

Project Description

This project will introduce the first set of target information to capture and present. The target information will define devices on the network. This can be useful to home and SOHO applications but will be particularly useful in other private networks like industrial and closed areas. The device will capture broadcast traffic and compare the source address with a known list of addresses. If a address is not on the list, a notification will be generated and data captured on the new device.

The final goal of this project is to enable users to take predefined data acquisition tools and use them to build automated tasks. Versatility is the key to the project. With this, someone could automate an approval process for adding new devices to a white list, or simply generate a device inventory.

Innovation Claim

This project is innovative because it brings “unseen” network traffic into light using commercial off the shelf products and free to use tools.

Usage Scenario

The usage of this product can be applied to personal and commercial environments. The purpose of this project is to start down a path of simplification and unification of network dependent tasks. This will be a major benefit to security conscious professionals, since the network is a major means for performing an attack. Instead of using a mix of products, a single easy-to-use platform can be presented to save time and reduce complexity in security and operational tasks.

The matured product will be capable of sending emails, SMS and push notifications based on user configuration. A for example, a user could link a custom push notification that is triggered when a new device is detected on a restricted IP range. That would allow support activities to locate the device and resolve the conflict.

Evaluation Criteria

Does the device record devices on the network? [Complete]

Do the new devices have a timestamp record? [Future Update]

Is the device list viewable? [Complete]

Is there a distinct notification of new (not previously recorded) devices on the network? [Complete]

Is there a button to incorporate new devices into known devices? [Future Update]

Objectives and Tasks Associated with the Project

Data capture – Automated packet capture will be required to discover the networked devices.

Data processing – With the data generated by the packet capture, key information needs to be pulled and stored. This will be file based but to expand the scale, a DBMS will likely have to be implemented.

Display data – Initial state will be displaying data to a terminal window. Future state will have a front end that displays the devices (IP address, MAC address, timestamp of first sighting).

Notification – To identify new devices, they will be displayed at the top of the screen.

Environment

A Linux OS will be installed on Raspberry Pi hardware. Required packages like tcpdump will be installed. Custom script will be deployed to complete required tasks.

Development

Programming will begin with automating tcpdump to capture the information required to generate the target data.

A decision will have to be made on how to best parse and manipulate the data. Perl will be the first method to be evaluated. It is believed to be more efficient than Python in the required tasks.

Displaying data is going to be a special challenge. Just printing the data to a terminal will be the first step. I am concerned that this task will start an avalanche of complexity. Next level of complexity will require a webserver to be installed and configured to display the data.

With a webserver properly running, a webpage will need to be created to display the data. At this point, the device data will either be split into two files for new and existing devices, or there will be a data field for new and acknowledged devices. New devices will either be in a separate list or the top of a single list.

Finally, an acknowledge button will merge the new device with the existing devices. If issues or time constraints prevent this, the devices may be listed in order of timestamp.

Description of Design Prototype

The Prototype will be a Raspberry Pi that runs the created program as defined above. The Raspberry Pi will have a power bank attached to provide portability. It will start collecting data once a network connection is established.

Evaluation Plan

The evaluation will consist of taking pictures or screenshots that prove each item in the criteria list is complete.

Project Completion Assessment

In short: I have written some Linux scripts that listens and outputs IP addresses. I have the minimum required code to create a simple dynamically updating host IP address list.

This project, like many others, started with the image of a perfect and easy to use tool. The constraints and setbacks forced me to shrink the scope until the work required was more realistic.

Appendices

Appendix A: Process Flow – Flow.jpg