Identifying Devices Across the IPv4 Address Space

Ryan Jicha, Dr. Mark W Patton, Dr. Hsinchun Chen
University of Arizona
Department of Management Information Systems
Tucson, AZ 85721, USA
rjicha@email.arizona.edu, mpatton@email.arizona.edu, hchen@eller.arizona.edu

Abstract—Many of today’s devices are internet-enabled with IPv4 internet addresses, exposing them to internet threats. To determine the true scale of vulnerabilities being introduced, particularly in the IPv4 internet address space, a new methodology of scanning the entire IPv4 internet space is required. To improve scanning speeds we created a framework combining fast connectionless port scanners with a thorough and accurate connection-oriented scanner to verify results. The results are stored to a database. This combined framework provides more robust results than current connectionless scanners, yet still scans the IPv4 internet fast enough to be practically usable for mass scanning.

Keywords—devices, Internet Protocol, IPv4, Supervisory Control and Data Acquisition systems, SCADA

I. INTRODUCTION

Many devices are internet-enabled in today’s world. In the home, televisions, cameras, washers and dryers, and even door locks are now connected to the internet to be controlled by smart phones or otherwise remotely monitored. Of particular importance are the large numbers of commercial Supervisory Control and Data Acquisition (SCADA) devices controlling critical infrastructure such as power, water, transportation systems, and gas lines. Ensuring the security of these devices is key to their safety and correct operations.

Most devices are connected using Internet Protocol version 4 (IPv4). IPv4 has nearly 4.3 billion unique IP addresses, and each of these addresses can communicate with other machines over 65,535 ports. While IPv6 is becoming more common, IPv4 is widely utilized and being supplemented but not replaced by IPv6. By scanning open ports on devices and determining the services and service versions running, one can often identify a given device, determine its operating system, and verify possible vulnerabilities. Network scanning is a fundamental way of determining potential vulnerabilities and is commonly used internally by organizations. However, current methods have allowed complete scans to be done across the entire IPv4 range. The main motivation for this research is to create a way to scan and identify devices in a reasonable amount of time across the entire IPv4 internet.

II. BACKGROUND

A. Connection-Oriented Scanners

In the case of connection-oriented scanners, a connection is made to a single device port. This connection is maintained until a response from that device is received. If no response is received, the scanner assumes the packet or response was lost and sends a new request to the same port. The scanner does not move on to the next port until either a certain amount of packets are dropped to the destination, or the device responds to the connection. Connection-oriented scanners are robust to network losses, but do so at the expense of time, scanning address spaces slowly. They often have additional functionality such as system identification through fingerprinting.

B. Connectionless Scanners

Connectionless scanners, often referred to as “asynchronous stateless TCP scanners,” do not track the connections the scanner is attempting to make. Instead, the scanner runs two separate processes. The first process sends out connection requests as fast as the network interface card will allow or at some user-defined rate. The second process then listens for responses. Generally, the second process continues running for a predetermined time after the final packet has been sent to allow responses to reach the scanner. The responses are captured and can be later compared to the connection attempts and analyzed further.

III. LITERATURE REVIEW

Nmap is an extremely robust connection-oriented scanner that includes the ability to identify applications and systems through fingerprints. In Ghanem’s study of a comparison of Nmap, Amap, and Ettercap to identify application fingerprint efficacy, Nmap had a 90% accuracy rate in determining running services compared to Amap’s 54% accuracy and Ettercap’s 50% accuracy. In regards to determining the correct service and version, Nmap was again ahead with a rate of 52% accuracy compared to Amap’s 4% accuracy and Ettercap’s 32% accuracy; when all of these tools were combined, the service and version accuracy rates increased to 94% and 87%, respectively [1].

Markowsky also used Nmap in research to determine the types of internet-enabled devices vulnerable to Heartbleed on a university network. Additionally, Markowsky used Nmap to scan a small network for open default printer ports [2].

In research by Patton et al., the Shodan search engine was used to manually search Shodan for devices with banners that would indicate they are SCADA devices [3]. According to the Shodan API, only 234 ports are scanned by Shodan [4]. This list of ports is not a comprehensive list of SCADA device ports; Shodan scans 8% of common SCADA ports. Despite this, the researchers were still able to find such devices as traffic control devices, printers, Niagara SCADA devices, and webcams.

In Middleton’s conference paper about analyzing...
vulnerabilities between seven countries, a combination of Nmap and Nessus was used, Nessus being a well-known vulnerability assessment tool. Due to the differences in size of the address spaces being compared, an extremely small sample of devices was scanned. Only 0.001% of the total IP addresses from each country identified as actively being used were scanned at random, and only 10% of those devices were tested, again at random, for vulnerabilities. This resulted in fewer than 400 devices being scanned in total [5]. Nmap was used to identify active IP addresses, and Nessus scanned for vulnerabilities. The small sample size was used to ensure an “unbiased sample” and to create a “manageable sample” for scanning, most likely due to the seven selected countries making up more than 10% of the IPv4 address space. Because Nmap maintains a connection to the devices it is scanning, it has a limited scanning speed. Scanning hundreds of millions of IP addresses with Nmap and Nessus would therefore take a significant amount of time; using a connectionless port scanner would be much quicker and more reasonable [5].

Durumeric et al., in their 2013 paper, did a comparison of Zmap and Nmap using a one GbE connection. The results show that Zmap is able to scan a single port in the entire IPv4 range in less than one hour, sending two probes per machine for improved accuracy [6]. By setting Nmap to a much faster rate of scanning, it was estimated that it would finish a scan of the entire IPv4 range in just over 116 days using a maximum of 2 probes per IP address [6].

In the 2014 paper by Durumeric et al., Zmap was modified to send Heartbeat requests to port 443 of 1% of the devices on the IPv4 range to identify Heartbleed vulnerabilities [7]. Zmap can also be modified to send other payloads to ports to identify services or vulnerabilities related to services that may be running on those ports [7].

IV. RESEARCH GAPS

From the literature review, there are two main points that can be gathered. First, existing internet scans such as Shodan lack in SCADA port coverage. To truly determine what SCADA ports are vulnerable in the IPv4 space, more than 8% of the known SCADA ports need to be scanned. There are many different SCADA devices and broad sampling techniques across the breadth of the internet can easily miss low frequency but critical vulnerabilities. Next, most “mass scanning” research is done on a small subset of the IPv4 range. In the case of Durumeric, 1% of the IP addresses were scanned to show Heartbleed trends in the IPv4 address space [7]. For Middleton, the vulnerability patterns among countries were based on .0001% of all devices in the address range [5]. This sampling is capable of showing trends, but comprehensive scanning is needed to fully map both widespread and the infrequent but critical vulnerabilities across the IPv4 internet address space.

V. METHODOLOGY

A. Research Question

The question to be answered by the research presented in the paper is “Can we quickly and effectively identify devices in the IPv4 address space based on their open ports?”

To address this question, the entire project was split into two sections. The first section was creating a framework that would combine a connection-oriented scanner and a connectionless scanner. This section was then split into two tasks: researching tools, and then engineering the framework and writing the code. The second section was about testing the newly created framework. This section was also split into two tasks: benchmarking, and analyzing the benchmark results.

B. Research Design

The task of researching tools took on three parts. First, tools needed to be discovered. Through the aforementioned literature, Amap and Nmap were identified as viable connection-oriented scanners; though after analysis of Ghanem’s research, it was learned that Nmap outperforms Amap as a standalone version detection tool [1]. For connectionless scanners, Zmap, Masscan, and UnicornScan were compared. Unicornscan was eliminated due to a lack of development, and www.unicornscan.org is no longer active. This left Zmap and Masscan, both of which would be tested in the tool. The final stage of research was looking at past tool combinations to determine which worked together effectively. The only literature referencing combining tools was by Ghanem, which involved combining Amap, Nmap, and Ettercap. No other literature utilized multiple tools to improve the speed of scanning a large network.

C. System Engineering

First, the steps taken to identify devices needed to be planned. The framework would start by selecting a “profile” for which to scan. If searching for web servers, devices running ports 80, 443, 8080, and 8443 would be a good place to start. In order to store these profiles and pull them, a MySQL database is ideal.

Once this profile is pulled from the database, the connectionless scanner can scan the designated range of IP addresses for those ports, regardless of whether the target is a small network, a large network, or the entire IPv4 address space. These scan results are then saved in the MySQL database for processing to determine devices that fit the desired profile. To assist with identifying devices that partially fit the profile, weights are added to the table storing the port profiles, and a weight threshold is set by the user. With the results from the connectionless scanner, weights are added among the ports open by IP addresses; if the cumulative weight is greater than the threshold, that device is considered a potential device and added to a list to then be scanned by the connection-oriented scanner.

The final step of the framework’s basic functionality is to use a connection-oriented scanner to verify that the devices are running expected services on open ports. Again, these results are saved to MySQL, including service detection and operating system details from the scan. These results can then be queried to determine which devices truly fit the profile being scanned for.

D. Benchmarking

The completed framework was tested in a controlled
environment and compared to using Nmap as a standalone tool. First, the connectionless scanner was tested for speed and accuracy. Next, Nmap was tested for speed and accuracy using a scan that should return only open ports with no service or operating system detection. Then, Nmap’s service identification and operating system detection scan was tested for speed and accuracy. Finally, the framework was tested as a whole.

E. Test Analysis

With the four tests described in the Benchmarking section, there are two result pairs to be compared. The results from the connectionless scanner are compared to Nmap to get the open ports. The second pair is the Nmap service identification for the entire test environment compared to the framework as a whole. The prior will rely on Nmap to determine if ports are open before testing for service and operating system, while the latter will rely on the connectionless scanner to inform Nmap of the exact ports that are open for service and operating system detection.

VI. TESTS

Research Environment and Scanning Results

For testing purposes, and to avoid overloading the university and lab networks, Masscan was limited to 50,000 packets per second. Two targets were used for testing the framework. The first target was an ESXi machine running on the local university network. The second device was a honeypot set up by another researcher at the University of Arizona for research purposes. The devices were first scanned with Nmap to determine how many ports were open on the machines. Various sets of scans were run on each machine for the benchmarking process. All scans assumed we wanted to know the states and services of all 65,535 ports. In the case of “Nmap Targeted Probing” in the Framework section, this scan only checked ports found open on each device from the Masscan port scan.

The Nmap scan times in Table I show that a comprehensive port scan of all 65,535 ports can easily take hours for a single IPv4 internet address, while Table II shows a combined scanning speed of approximately three minutes, resulting in a speed increase often exceeding 200 times the original rate.

<table>
<thead>
<tr>
<th>Framework Benchmark</th>
<th>Masscan Time (sec)</th>
<th>Nmap Targeted Probing (sec)</th>
<th>Total Framework Speed (sec)</th>
<th>Ports Found by Framework</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ESXi</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>166.38</td>
<td>178.38</td>
<td>11</td>
<td>100.00%</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>166.30</td>
<td>179.30</td>
<td>11</td>
<td>100.00%</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>167.75</td>
<td>179.75</td>
<td>11</td>
<td>100.00%</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>166.32</td>
<td>179.32</td>
<td>11</td>
<td>100.00%</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>167.72</td>
<td>181.72</td>
<td>11</td>
<td>100.00%</td>
<td></td>
</tr>
<tr>
<td><strong>Honeypot</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>165.65</td>
<td>178.65</td>
<td>8</td>
<td>100.00%</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>164.33</td>
<td>177.33</td>
<td>7</td>
<td>87.50%</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>164.50</td>
<td>178.50</td>
<td>6</td>
<td>75.00%</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>165.70</td>
<td>178.70</td>
<td>5</td>
<td>62.50%</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>164.01</td>
<td>176.01</td>
<td>5</td>
<td>62.50%</td>
<td></td>
</tr>
<tr>
<td><strong>Combined</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>168.07</td>
<td>180.07</td>
<td>19</td>
<td>100.00%</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>167.78</td>
<td>181.78</td>
<td>19</td>
<td>100.00%</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>167.94</td>
<td>180.94</td>
<td>18</td>
<td>94.74%</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>168.02</td>
<td>182.02</td>
<td>19</td>
<td>100.00%</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>168.88</td>
<td>180.88</td>
<td>19</td>
<td>100.00%</td>
<td></td>
</tr>
</tbody>
</table>

REFERENCES