A Quantitative Analysis of the Insecurity of Embedded Network Devices: Results of a Wide-Area Scan

Ang Cui and Salvatore J. Stolfo
Department of Computer Science, Columbia University
{ang,sal}@cs.columbia.edu

ABSTRACT

We present a quantitative lower bound on the number of vulnerable embedded device on a global scale. Over the past year, we have systematically scanned large portions of the internet to monitor the presence of trivially vulnerable embedded devices. At the time of writing, we have identified over **540,000** publicly accessible embedded devices configured with factory default root passwords. This constitutes over **13%** of all discovered embedded devices.These devices range from enterprise equipment such as firewalls and routers to consumer appliances such as VoIP adapters, cable and IPTV boxes to office equipment such as network printers and video conferencing units. Vulnerable devices were detected in **144 countries**, across 17,427 unique private enterprise, ISP, government, educational, satellite provider as well as residential network environments. Preliminary results from our longitudinal study tracking over 102,000 vulnerable devices revealed that over **96%** of such accessible devices remain vulnerable after a 4-month period. We believe the data presented in this paper provides a conservative lower bound on the actual population of vulnerable devices in the wild. By combining the observed vulnerability distributions and its potential root causes, we propose a set of mitigation strategies and hypothesize about its quantitative impact on reducing the global vulnerable embedded device population. Employing our strategy, we have partnered with Team Cymru to engage key organizations capable of significantly reducing the number of trivially vulnerable embedded devices currently on the internet. As an ongoing longitudinal study, we plan to gather data continuously over the next year in order to quantify the effectiveness of community's cumulative effort to mitigate this pervasive threat.

1. INTRODUCTION

Embedded network devices have become an ubiquitous fixture in the modern home, office as well as in the global communication infrastructure. Routers, NAS appliances, home entertainment appliances, wireless access points, web cams, VoIP appliances, print servers and video conferencing units reside on the same networks as our personal computers and enterprise servers and together form our world-wide communication infrastructure. Widely deployed and often misconfigured, embedded network devices constitute highly attractive targets for exploitation.

Although common wisdom enforces the suspicion that embedded devices tend to be less secure then general purpose computers and often trivial to exploit, evidence of such insecurities is mostly anecdotal. To fully appreciate the scope and scale of the embedded threat, we must move beyond analysis of individual embedded devices and their vulnerabilities. In order to formulate realistic and effective mitigation strategies against current and next generation embedded device exploitation, we first pose and answer several fundamental questions:

- How have embedded devices been exploited in the past? How feasible is large scale exploitation of embedded devices? (Section 2)
- How can we quantitatively measure the level of embedded device insecurity on a global scale? (Section 3)
- How can compromised embedded devices be used to benefit malicious attackers? (Section 4)
- How many vulnerable embedded devices are there in the world? What are they? Where are they? (Section 5)
- What are the most efficient methods of securing vulnerable embedded devices? (Section 6)

The purpose of our project is to quantify and trend the level of insecurity of embedded devices currently in the wild. To this end, we first establish an observed **lower bound** on the number of trivially vulnerable embedded devices on the internet. We do this by assuming the role of the least sophisticated malicious attacker (See Section 3.1), who only tries to log into publicly reachable embedded devices using well known **default root credentials**. Section 3 describes the default credential scanner we developed using standard tools such as **nmap**, which positively identified over **540,000** wide open embedded devices.

Vulnerable devices were detected in **144 countries**, in enterprise, ISP, government, educational, satellite provider as well as residential network environments[1]. We discov-

[1]Military networks are intentionally excluded from our scan, although a collaborative effort is currently underway to carry out the same scan on US military networks.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
ACSAC '10 Dec. 6-10, 2010, Austin, Texas USA
Copyright 2010 ACM 978-1-4503-0133-6/10/12 ...$10.00.

Total IPs Scanned	Devices Targeted	Vulnerable Devices	Vulnerability Rate
3,223,358,720	3,912,574	540,435	13.81%

Table 1: Scale and Result of the Latest Global Default Credential Scan.

ered vulnerable devices across a diverse spectrum of product types, including consumer appliances, home networking devices, office appliances, enterprise and carrier networking equipment, data-center power management devices, network security appliances, server lights-out-management controllers, IP camera surveillance systems, VoIP devices, video conferencing appliances as well as ISP issued modems and set-top boxes. Section 5 presents detailed analysis of the data collected by our default credential scanner.

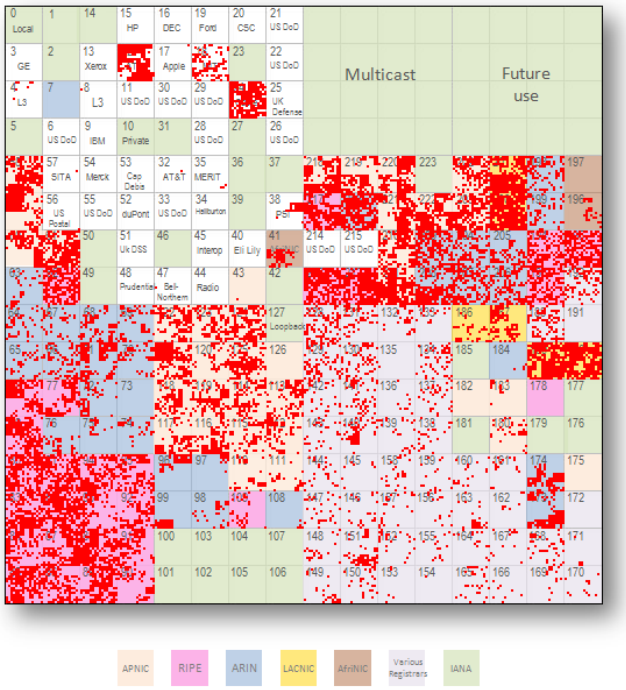

Figure 1: Distribution of Vulnerable Embedded Devices in IPv4 Space. Total Number of Vulnerable Devices Found: 540,435.

While the observed quantity and distribution of embedded devices configured with default root passwords demonstrate a global, pervasive phenomenon, we believe the data presented in this paper represent a conservative lower bound on the actual population of vulnerable devices in the wild. Evidence suggests that this lower bound can be raised significantly by slightly escalating the level of sophistication of our assumed attacker [11].

1.1 Contributions

We present the first quantitative measurement of embedded device insecurity on a global scale, along with preliminary results from an ongoing longitudinal study of the same subject. By assuming the role of the least sophisticated attacker (see Section 3.1), we present an observed **lower bound** on the distribution of trivially exploitable network embedded devices over functional (Section 5.1), spatial (Section 5.2), organizational (Section 5.3) and temporal (Section 5.5) domains.

The embedded device default credential scanner created for this experiment is designed to identify efficiently and safely the vulnerable embedded devices on the network. It does this by testing whether one can remotely login into a device using well known default root credentials. The verification process is designed to use minimal resources on the target embedded device. The scanner currently supports 73 unique embedded device types including consumer appliances, home networking devices, office appliances, enterprise and carrier networking equipment, data-center power management devices, network security appliances, server lights-out-management controllers, IP camera surveillance systems, VoIP devices, video conferencing appliances as well as ISP issued modems and set-top boxes.

While the embedded security threat has been generally known for some time, the data presented in this paper provides a real-world quantitative assessment of the scale and scope of the embedded threat on a global level. Analysis of our results yields several interesting features within the observed vulnerability distributions. The features presented in Section 5 presents insights into the root causes of the existence of vulnerable embedded devices. By combining the observed vulnerability distributions and its potential root causes, we formulate a set of mitigation strategies and hypothesize about its quantitative impact on reducing the global vulnerable device population.

Many forces will undoubtably change the observable lower bound of embedded device insecurity as time goes on. For example, the out-of-the-box security of new embedded products may change. Network operators controlling large homogeneous sets of devices may improve their security, as may small and medium size organizations like private enterprises and educational organizations. The level of malicious exploitation will also indirectly contribute to the overall effort dedicated to improving embedded device security. Lastly, it is our hope that the data and mitigation strategies reported in this paper will generate more awareness of this pervasive threat. In order to quantify the scope of the embedded device insecurity threat over time and detect such forces at work, we plan to continue our scanning activities to conduct an ongoing longitudinal study over the next year. Section 5.5 discusses the preliminary results of our longitudinal study over the past four months.

1.2 Outline

The remainder of this paper is organized as follows: Section 2 surveys recent developments related to embedded device insecurity in white-hat and black-hat communities as well as popular literature. Section 3 describes our methodology with emphasis on the steps taken to ensure a safe and ethical experimental protocol. Section 4 describes a variety of novel malicious uses of the vulnerable devices discovered by our scanner. Section 5 presents the analysis of data gathered from our latest global scan as well as preliminary results from our ongoing longitudinal study. Section 6 presents a set of remediation strategies, along with an quantitative estimates of its potential effect with respect to the global vulnerable device population. We conclude in Section 7 with a summary of our contributions.

2. RELATED WORKS

Evidence of embedded device insecurity and exploitation has been presented in both white-hat and black-hat venues for quite some time. The creation and propagation characteristics of hypothetical malnets exploiting vulnerable wireless routers have been described by several researchers [10, 19]. For example, Traynor *et al.* showed that an adversary can potentially compromise over 24,000 routers in Manhattan in less than 2 hours [19]. The data from our scan indicates that trivially exploitable embedded devices exist in sufficient quantity and concentration for such hypothetical attacks to be feasible. Our data also corroborates that phishing attacks using compromised consumer electronics such as home routers [20] can be carried out on a large scale by technically unsophisticated attackers.

Existing evidence clearly reenforces the common wisdom that embedded devices are generally less secure then general purpose computers and are often trivial to exploit. However, the available literature tends to focus on specific vulnerabilities or vulnerable devices.

For example, a recent Wired.com article [9] announced a vulnerability found on the administrative interface of the SMC8014 series cable modem, potentially affecting 65,000 Time Warner customers. Numerous research projects [18, 11] targeting specific device types have demonstrated that large numbers of vulnerabilities within ubiquitous embedded device types. According to Bojinov *et al.*, an audit of common embedded administrative interfaces from 16 major manufacturers yielded significant vulnerabilities from all of the 21 devices considered [11].

The evolution of embedded device exploitation tools and techniques demonstrate an accelerating maturation of malicious attacks against embedded devices. While proof of concept Cisco IOS exploits and shellcode have been publicly available since 2003 [13, 16], recent evidence suggests that attackers are scanning for and exploiting consumer routers to build modest size bot-nets, mainly for DDOS purposes. The appearance of tutorials [5] and simple to use tools to find and control specific consumer routers indicate that embedded device exploitation techniques are beginning to diffuse out of research circles, and into the general black-hat community.

To the best of our knowledge, the first consumer router botnet, psyb0t, was reported by Dronebl.org in 2008 [6]. While no detailed analysis of the bot was published, we do know that it primarily targeted mipsel OpenWRT and DD-WRT devices using default passwords. It is suspected that the psyb0t botnet observed in 2008 was a proof of concept test of the technology [7], as the botnet was quickly shutdown by its operators following Dronebl.org's public announcement of its existence.

The current generation of embedded device malcode may be related to existing unix tools like Kaiten.c [1]. A survey of black hat literature circa 2008 shows at least one document describing the process of compromising similar consumer routers using password guessing and existing unix IRC bots [5]. This may help to explain why the majority of victim embedded devices exploited thus far have been unix-based consumer routers. For example, psyb0t targeted only home routers and heavily leveraged the unix-like operating environment found on its victim devices. Specifically, psyb0t used commands like wget and chmod to download its payload onto victim devices and used iptables to block all administrative interfaces to protect the device from other attackers.

2.1 Next Generation Embedded Malcode

Existing embedded device malcode such as psyb0t depend heavily on its victim devices' similarity to traditional unix systems. While development of such malcode is relatively straightforward, it constrains the vulnerable population to low-end consumer appliances running unix-like operating systems. For example, enterprise networking devices like Cisco routers and switches run on proprietary operating systems like IOS, which do not resemble traditional unix architecture. However, recent advancements in exploitation and root-kitting techniques for proprietary operating systems like Cisco IOS [17, 14] could allow attackers to compromise high-end enterprise devices like backbone routers and firewalls. It is highly likely that the next generation of embedded device malcode will have greater ability to compromise heterogeneous device types, stealthier and more sophisticated command and control channels, as well as other malicious capabilities aside from DDOS.

Furthermore, as data presented in Section 5 suggest, the current population of trivially vulnerable embedded devices is quite high. Therefore, the next generation of malcode capable of compromising heterogeneous device types will easily be able to infect significantly more devices then psyb0t and kaiten.c in their current state.

3. EXPERIMENTAL METHODOLOGY

The default credential scanner is designed to quickly sweep large portions of the internet. Each scan takes approximately four weeks and involves two or three sweeps of the entire monitored IP space (Section 3.4 discusses how the monitored IP ranges are selected.)

Multiple sweeps across the same IP space is desirable for two reasons. First, embedded devices on residential networks have unpredictable availability. Therefore, multiple sweeps increase the scanner's probability of observing a vulnerable device when it is connected to the network. Second, multiple sweeps across the same address space over months and years allow us to conduct a **longitudinal** study on the vulnerability rates of embedded devices around the world.

In Section 5, we present the results of our latest scan, containing over 540,000 observed vulnerable devices, as well as analysis of preliminary data gathered by tracking approximately 102,000 vulnerable embedded devices over a span of four months in Section 5. This is an ongoing study, and we plan to publish the results of a detailed longitudinal study over the next year when the data becomes available.

3.1 Threat Model

For the sake of establishing a lower bound on the state of embedded device insecurity in the wild, we assume the role of the least sophisticated malicious attacker. The attacker has unrestricted access to the internet but is unable to exploit any vulnerabilities found on any devices. Instead, the attacker has access to the network scanner nmap and a list of well known factory default root credentials for popular network embedded devices.

For the remainder of the paper, we define a *vulnerable* device as any device that is reachable on the internet and allows the attacker to gain root privileges by using factory default credentials.

```
User Access Verification

Username:
```

Figure 2: Common Cisco Telnet Login Prompt.

```
root:
    username_prompt: ['sername:']
    username: cisco
    askuser: true
    passstr: ['assword:']
    incorrect: [sername, assword]
    success: ['\#']
    passwords: ['cisco']
    deviceType: cisco
    isActive: 'true'
```

```
root:
    authType: basicAuth
    passwds: ['admin']
    authRealm: WRT54G
    username: ''
    deviceType: linksys-wrt
    loginURL: '/'
    isActive: 'true'
```

3.2 Default Credential Scanner: A Three Phase Process

The default credential scan process is straightforward and can be broken down into three sequential phases: **recognizance**, **identification**, and **verification**.

Recognizance: First, nmap is used to scan large portions of the internet for open TCP ports 23 and 80. The results of scan is stored in a SQL database.

Identification: Next, the device identification process connects to all listening Telnet and HTTP servers to retrieve the initial output of these servers[2]. The server output is stored in a SQL database then matched against a list of signatures to identify the manufacturer and model of the device in question (See 3.3).
For example, Figure 2 shows a telnet login prompt common to Cisco routers and switches.

Verification: Once the manufacturer and model of the device are positively identified, the verification phase uses an automated script to attempt to log into devices found in the identification phase. This script uses only well known default root credentials for the specific device model and does not engage in any form of brute force password guessing. We create a unique *device verification profile* for each type of embedded device we monitor. This profile contains all information necessary for the verification script to automatically negotiate the authentication process, using either the device's Telnet or HTTP administrative interface. Figure 3.2 shows two typical device verification profiles, one for the administrative Telnet interface for Cisco switches and routers, the other for the HTTP administrative interface for Linksys WRT routers using HTTP Basic Authentication. Each device verification profile contains information like the username and password prompt signatures, default credentials as well as authentication success and failure conditions for the particular embedded device type. Once the success or failure of the default credential is verified, the TCP session is terminated and the results are written to an encrypted flash drive for off-line analysis. (See 3.5).

[2]In case of HTTP, we issue the 'get /' request

Total IPs Scanned	Number of Countries Scanned	Number of Organizations Scanned
3,223,358,720	193	17,427
Most Heavily Scanned Countries		
US	CN	JP
1,477,339,136	217,273,088	177,494,016
GB	DE	CN
111,457,280	107,387,648	77,328,896

Table 2: Key Statistics on the Scope and Geographical Distribution of the IP Ranges Currently Monitored by the Default Credential Scanner.

3.3 Device Selection

The full list of devices currently monitored by our default credential scanner can be found on our project webpage[3]. In order for an embedded device to be included in this list, its default root credentials must be well known and obtainable through either manufacturer documentation or simple search engine queries. The default credential scanner does not engage in any form of brute force password guessing.

The device selection process is manual and iterative. We begin by analyzing data gathered by the recognizance phase of our scanner, which collects the initial output from active Telnet and HTTP servers found by NMAP. We maintain three sets of signatures: non-embedded devices, non-candidate embedded devices and candidate embedded devices. Signatures of non-embedded devices include those of popular HTTP servers such as Apache and IIS as well as Telnet common authentication prompts of general purpose operating systems. Signatures of non-candidate embedded devices include those that do not ship with a well known default credential[4]. Signatures of candidate embedded devices include string patterns that positively identify the device as one that we are actively monitoring. After the recognizance data is tagged using these three signature sets, we manually inspect the remaining records, tagging, creating new signatures and device verification profiles.

3.4 Network Range Selection

We initially directed our scan towards the largest ISPs in North and South America, Europe and Asia. As we iteratively refined our scanning infrastructure, we gradually widened the scope of our scan to include select geographical locations within the United States. After testing our default credential scanner for over six months to ensure that it caused no harm to the scanned networks, we finally allowed the scanner to operate globally. Using a reverse lookup of the MaxMind GeoIP database [2], we included every /24 network in the IPv4 space which is associated to a geographical location. Table 2 shows some key metrics on the scope of the IP ranges which we currently monitor.

3.5 Ethical Considerations and Due Diligence

The technical methodology of our project is straightforward. However, the necessary means of gathering real-world data on the vulnerability rates of embedded device have raised an ethical debate.

[3]http://www.hacktory.cs.columbia.edu

[4]For example, the Polycom VSX 3000 video conferencing unit uses the device's serial number as the default password.

On one hand, the simple act of port scanning a remote network across the internet can be construed as a hostile and malicious attack. On the other hand, we can not move beyond vague and anecdotal suspicions of the embedded device security problem unless we gather large scale, quantitative evidence of the problem currently in the wild.

As advocated in a recent position paper on the ethics of security vulnerability research [15], this line of proactive vulnerability research serves an important social function and is **neither unethical nor illegal with respect to US law**.

The experimental results contain sensitive information on a large number of vulnerable devices in the world, some of which reside in sensitive environments. Therefore it is the responsibility of the research team to uphold a high standard for ethical behavior and due diligence when engaging in such sensitive research. The operating environment must be isolated and fortified against compromise and data exfiltration. Furthermore, each member of the research team must agree to adhere to a clear experimental protocol to ensure that **no harm is done**.

A trivial network scanner can be implemented with little work. However, using such a scanner openly on a global scale is irresponsible and ethically unacceptable. Therefore we have invested a large portion of of energy to create a secure research environment and a responsible experimental protocol in order to ensure that our activities cause no harm:

Doing no harm. Bound by the ethics principal of the duty not to harm, we have taken numerous steps to ensure that our research activities do not interfere with the normal operations of the networks we monitor. To this end, the default credential scanner is designed to use minimal external resources in order to accurately verify device vulnerability. We scan target networks in /24 blocks in non-sequential order in order to minimize the number of incoming TCP requests destined to any individual organization. Detailed activity logs are kept to ensure that no device or network is unnecessarily probed multiple times during a single scan. Overall, non-embedded devices and non-candidate embedded devices will receive at most 6 TCP packets over a period of several minutes. The scanner's outbound packet-rate is policed and monitored in order not to overwhelm any in-path networking devices. Lastly, each IP address used by our scanner runs a pubic webpage describing the intention and methodology of our project [3]. This page also provides instructions for permanently opting-out of the scan. (See Table 6). Such requests are monitored by both our research team as well as the Columbia University NOC, and are promptly honored without question.

Implementing a secure research environment. The scan system is contained in a DMZ network behind a Cisco ASA firewall. Scanning nodes are isolated from the university network. Inbound access to this protected network can only be established by using IPSec VPN. Outbound access by the scanning nodes are limited to the ports which they are scanning (Telnet, HTTP, etc).

Compartmentalization of access to sensitive information. VPN access to the scan system DMZ is granted only to active members of the research team. New students participating in research are first given access to a separate DMZ containing a development copy of the scan system with no sensitive data. Access to the production environment is given to students only after they have acknowledged and demonstrated understanding of the experimental protocol.

Proper handling of sensitive data at rest. Sensitive experimental data is purged from the production database regularly, then transferred to an IronKey [4] USB stick for encrypted offline storage. This is done to minimize the amount of data available for exfiltration in case of a compromise of the research environment.

Notifications of vulnerabilities through trusted channels. Significant vulnerabilities are reported to Team Cymru, who brokers communications between our research team and the appropriate contacts. Sensitive information detailing the vulnerable devices is either physically handed off to Team Cmyru members or transferred using encrypted channels.

4. MALICIOUS POTENTIAL OF EMBEDDED DEVICE EXPLOITATION

This section discusses several novel ways of exploiting vulnerable embedded devices due to their unique functions and hardware capabilities. After auditing the functional capabilities of many different embedded devices, we have concluded that the attacks described below are trivially possible among a majority of embedded devices within the appropriate functional categories. All attacks discussed below can be carried out through legitimate manipulation of the administrative interface. More importantly, as the data presented in Section 5 illustrate quantitatively, there exists a large population of embedded devices vulnerable to each of the attacks discussed below. Although DDOS attacks using embedded devices have certainly been carried out on a relatively large scale, most of the other attacks described in this section have not. However, considering the data presented in Section 5, we posit that it is only a matter of time before such attacks are carried out systematically on a large scale.

We have engaged several major organizations to mitigate some of the issues discussed below. Therefore, specific details regarding organization names and device model information are withheld when appropriate.

4.1 Massive DDOS Potential

The heterogeneous nature of embedded administrative interfaces makes orchestrating large DDOS attacks using embedded devices a logistic challenge. Vulnerable embedded devices clearly exist in large numbers in the wild. However, it is often believed that embedded operating systems are too diverse; and capturing the long tail of this diversity is required to carry out large scale exploitation. Data gathered by our default credential scanner reveal that many large vulnerable homogenous device groups exist in the wild. In fact, the top **3** most vulnerable device types represent over **55%** of all vulnerable devices discovered by our latest scan. In other words, there exists at least 300,000 vulnerable embedded devices which can be controlled via 3 similar Telnet-based administrative interfaces. The exact model of these three device groups have been anonymized. However, these three device groups are centrally managed by various service providers around the world, and thus can be systematically

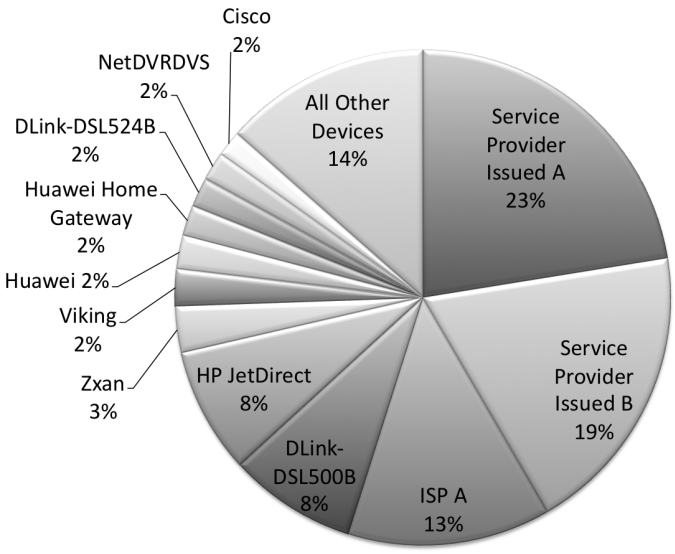

Figure 3: Distribution of Vulnerable Devices Across Unique Device Types. The Top 3 Device Types Constitute 55% of the Entire Vulnerable Device Population.

secured in a feasible manner. Figure 3 shows the distribution of the top 12 most frequently encountered vulnerable embedded device types.

4.2 VoIP Appliance Exploitation

VoIP adapters like the Linksys PAP2, Linksys SPA and Sipura SPA are consumer appliances, which provide a gateway between standard analog telephones and VoIP service providers. In many cases, the publicly accessible HTTP interface of such devices will display diagnostic information without requiring any user authentication. This information usually includes the name of the customer, their phone number(s), a log of incoming and outgoing calls, and relevant information regarding the SIP gateway to which the device is configured to connect. Once authenticated as the administrative user, an attacker can usually retrieve the customer's SIP credentials, either by exploiting trivial HTTP vulnerabilities[5] or redirecting the victim to a malicious SIP server.

4.3 Data Leakage via Office Appliance Exploitation

Enterprise printers servers and digital document stations are ubiquitous in most work environments. According to our data, network printers also constitute one of the most vulnerable types of embedded devices. For example, our default credential scanner identified over **44,000** vulnerable HP JetDirect Print Servers in **2,505** unique organizations worldwide. Since high-end print servers and document stations often have the capability of digitally caching the documents it processes, we posit that an attacker can use such devices not only to monitor the flow of internal documents, but also to exfiltrate them as well.

[5]Credentials are sometimes displayed in clear-text within HTML password fields. While this appears to hide the passwords in the web browser, it does not hide it in the HTML source.

4.4 Enterprise Credential Leakage via Accidental Misconfiguration

It is common practice for organizations that operate large homogenous collections of networking equipment to apply the same set of administrative credentials to all managed devices. While this significantly reduces the complexity and cost of managing a large network, it also puts the network at risk of total compromise. Using a single master root password for all networking devices is safe so long as every device is correctly configured at all times, and the master password is not leaked. If an enterprise networking device is brought online with both factory default credentials, as well as the master credentials of the organization, an attacker can easily obtain the master root password for the entire network. While this event is unlikely, the probability of such a misconfiguration quickly increases with the size and complexity of the organization, specially when human error is taken into account. We have not verified that such an attack is feasible; however, our data indicate that enterprise networking devices residing within large homogenous environments have been misconfigured with default root credentials.

5. ANALYSIS OF RESULTS

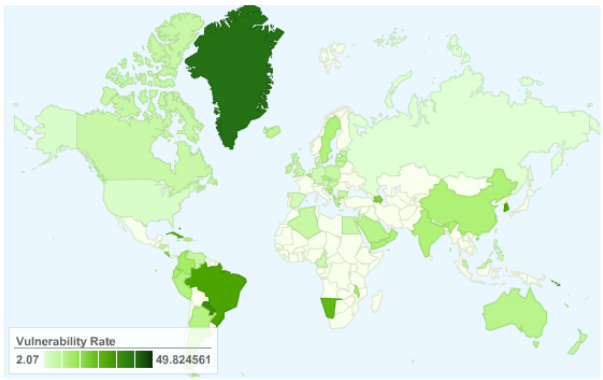

Figure 4: Embedded Device Vulnerability Rates of Monitored Countries (Threshold = 2%).

In this section we present latest data gathered by our default credential scanner as well as preliminary results from our ongoing longitudinal study, tracking approximately 102,000 vulnerable devices over a span of four months. We also present statistics on the level of human and organizational responses received by Columbia University regarding our scanning activities. Figure 4 shows a heat map of embedded device vulnerability rates across monitored countries.

Section 5.1 shows the breakdown of vulnerable embedded devices across **9 functional categories**; Enterprise Devices, VoIP Devices, Home Networking Devices, Camera/Surveillance, Office Appliances, Power Management Controllers, Service Provider Issued Equipment, Video Conferencing Units, and Home Brew Devices. Section 5.2 shows the breakdown of vulnerable embedded devices across **6 continents**. Section 5.3 shows the breakdown of vulnerable devices across **5 types of organizations**; Educational, ISP, Private Enterprise, Government, and Unidentified.

5.1 Breakdown of Vulnerable Devices by Functional Categories

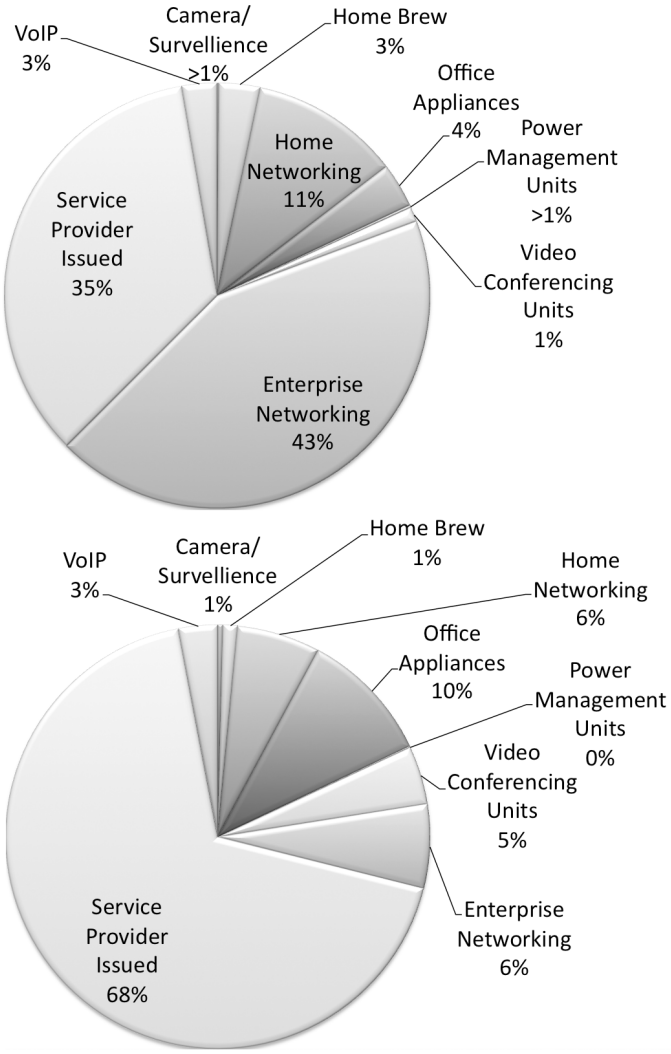

Figure 5: Discovered Candidate Devices (Top) and Vulnerable Devices (Bottom) By By Organization Type.

We organized 73 unique embedded device types monitored by our scan into 9 functional categories. Detailed categorization of monitored devices can be found on our project webpage[6]. Figure 5 shows the distribution of all discovered candidate embedded devices (top) and the distribution of vulnerable embedded devices (bottom) across the different functional categories. Table 3 shows the total number candidate embedded devices discovered within each functional category as well as their corresponding vulnerability rate.

- While **Service Provider Issued Equipment** accounts for only 35% of all discovered candidate embedded devices, it represents 68% of all vulnerable embedded devices.
- While **Enterprise Networking Equipment** accounts for 43% of all discovered candidate embedded devices, it only represents 6% of all vulnerable embedded devices.

[6]http://www.hacktory.cs.columbia.edu

5.2 Breakdown of Vulnerable Devices by Geographical Location

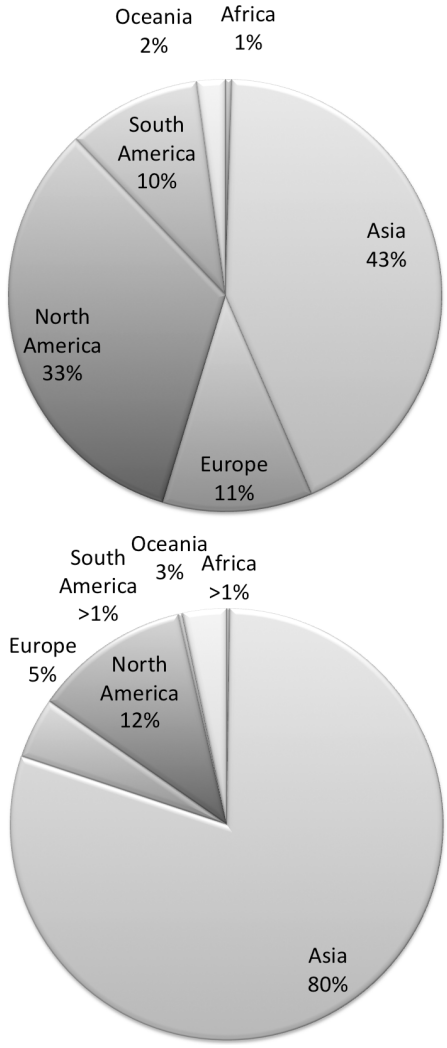

Figure 6: Discovered Candidate Devices (Top) and Vulnerable Devices (Bottom) By Geographical Distribution.

Using the MaxMind GeoIP database[2], we categorized all discovered candidate and vulnerable embedded devices according to the continent in which they are located. Figure 6 shows the distribution of all discovered embedded devices (top) and the distribution of vulnerable embedded devices (bottom) across 6 continents. Table 4 shows the total number of candidate embedded devices as well as the corresponding vulnerability rate within each continent.

- **Asia** represents the continent with the most number of candidate embedded devices and accounts for approximately 80% of all discovered vulnerable embedded devices.
- **South Korea** contains the largest number vulnerable embedded devices out of all monitored nations.
- While 33% of all discovered candidate embedded devices reside within **North America**, only 12% of all vulnerable embedded devices are found there.

5.3 Breakdown of Vulnerable Devices by Organizational Categories

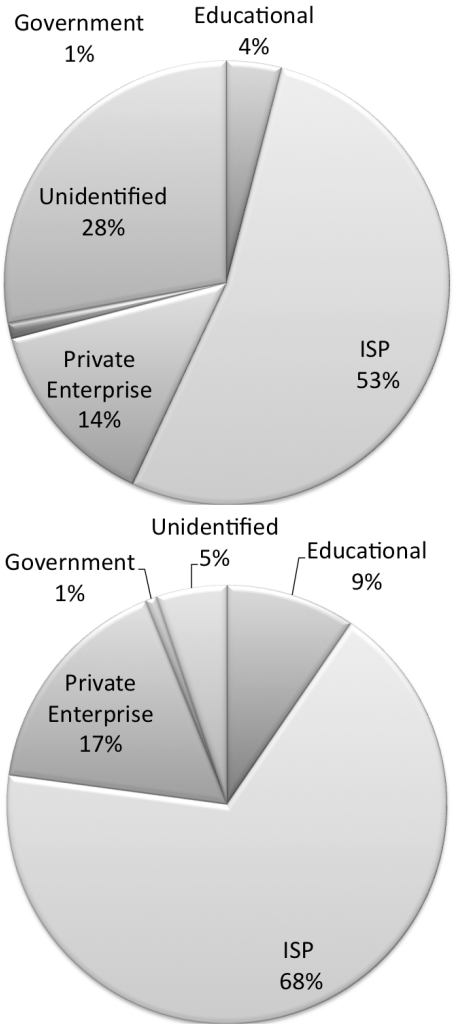

Figure 7: Discovered Candidate Devices (Top) and Vulnerable Devices (Bottom) By By Organization Type.

Using the MaxMind GeoIP Organization database[2], we categorized all monitored network ranges into 17,427 individual organizations. This was then divided into 4 general organization types; Educational, Internet Service Provider (ISP), Private Enterprise, and Government. 9118 organizations could not be accurately classified, and were left in Unidentified category. Figure 7 shows the distribution of all discovered embedded devices (top) and the distribution of vulnerable embedded devices (bottom) across the 5 organization types. Table 5 shows the total number of candidate embedded devices as well as the corresponding vulnerability rate within each organization type.

- **ISP** networks contain the most number of candidate embedded devices and house over 68% of all discovered vulnerable embedded devices.
- While **Educational** networks contain only a modest number of candidate embedded devices, it has the highest per category vulnerability rate of 32.83%

	Vul. Rate	Total Devices
Enterprise Devices	2.03%	1,689,245
VoIP Devices	15.34%	104,827
Home Networking	7.70%	445,147
Camera/Surveillance	39.72%	5,080
Office Appliances	41.19%	132,991
Power Management	7.23%	7,429
Service Provider Issued	27.02%	1,362,347
Video Conferencing	55.44%	43,349
Home Brew	4.93%	122,159

Table 3: Vulnerability Rate by Device Category.

	Vul. Rate	Total Devices
Africa	5.36%	19,363
Asia	21.69%	1,731,089
Europe	4.76%	450,019
North America	4.12%	1,335,575
South America	0.37%	402,163
Oceania	17.98%	85,941

Table 4: Total Discovered Candidate Embedded Devices and Corresponding Vulnerability Rates By Geographical Location (Continental).

	Unique Orgs	Vul. Rate	Total Devices
Educational	1,371	32.83%	156,992
ISP	2,374	17.43%	2,095,292
Priv. Enterprise	4,070	16.40%	554,101
Government	494	10.38%	44,460
Unidentified	9,118	2.54%	1,103,775

Table 5: Vulnerability Rate By Organization Type.

5.4 Community Response to Default Credential Scanner Activity

The default credential scanner is designed to direct interested parties to a public webpage which describes the intent and methodology of our project[3]. Each IP address used by the scanner also hosts a public HTTP server which redirects visitors to the public project webpage. We tracked access to this webpage using Google Analytics as a way to gauge the global community's awareness of our scanning activities. Figure 8 shows the number and geographical distribution of visitors over the past six months. The initial spike of visitors in October 2009 coincided with the publication of an article regarding preliminary results of our project [8]. Since then, our continuous scanning activity attracted **87 visitors** over the last 5 months.

Total Conversations	Opt-Out Requests	Request for Information, but Not Opt-Out
36	14	22
Tone of Counter-Party		
Supportive	Neutral	Hostile
14	15	7

Table 6: Email Correspondences Received from Network Operators Regarding Scanning Activity.

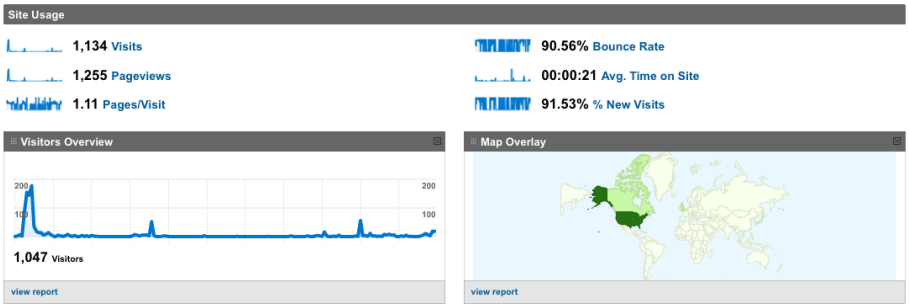

Figure 8: Daily Page Access Analytics For Scan Project Information Page [www.hacktory.cs.columbia.edu]. Oct 19, 2009 - April 12, 2010.

Vulnerable Devices Tracked	102,896
Tracked Devices Currently Online	54,429
Tracked Devices Currently Vulnerable	52,661

Table 7: Preliminary Longitudinal Study Tracking 102,896 Vulnerable Devices Over 4 Months.

Table 6 shows a breakdown of all communications between the operators of the networks monitored by our scanner and our research team. The conversations were all initiated by the counter-party via email, usually requesting further information or to be excluded from the scan. We answered 36 conversations in total, 14 of which requested certain IP ranges to be permanently excluded. 1,798 /24 networks were excluded as a result of these requests. **61%** of all interested parties which detected our scanning activity and contacted us decided to allow the scan to continue. The geographical location of the counter-parties correlates closely to the heat map in Figure 8. We did not receive any correspondence from ISP organizations or organizations from Asia, even though the majority of vulnerable devices were discovered within such IP ranges.

5.5 Preliminary Longitudinal Results

Table 7 shows the preliminary results of our longitudinal study. We retested 102,896 vulnerable embedded devices discovered at the end of December, 2009. As of April 20, 2010, 54,429 of the retested devices are still publicly accessible, out of which 52,661 devices remain vulnerable.

In other words, approximately **96.75%** of accessible vulnerable devices are still vulnerable after a 4 month period, and factory default credentials have been removed from only **3.25%** of the same set of devices.

6. REMEDIATION STRATEGY

The least sophisticated attacker modeled in this experiment can be defeated by simply discontinuing the use of well-known default credentials on embedded devices. However, the overall cost of implementing this naive mitigation strategy will likely be quite high in reality. In the unlikely event that all embedded device manufacturers universally agree to discontinue the use of well-known default passwords henceforth, we are still faced with the challenge of retroactively fixing the vulnerable legacy embedded devices in use throughout the world today. Therefore, it is reasonable to assume that the embedded security threat will likely persist and grow endemically for the near future. In order to effectively reduce the total population of vulnerable embedded devices in the wild, we must carefully consider the best methods for securing existing legacy devices. Since existing devices are by definition under the administrative control of some individual or organization, successful mitigation strategies must actively engage these network operators in order to fix the problem.

According to the data presented in Section 5, a few groups of network operators contribute disproportionally large numbers of vulnerable embedded devices to the global population. For example, we discovered over 300,000 vulnerable embedded devices operating in homogenous environments within two ISP networks in Asia. Overall, embedded devices operated by residential ISPs constitute over 68% of the entire vulnerable population. Since ISPs centrally manage large numbers of vulnerable embedded devices, they are the ideal candidates to engage to mitigate the embedded security threat.

While immediately effective, engaging individual organizations and manufacturers to fix pockets of vulnerable devices can only impede the growth of the embedded security threat but not solve it. In order to improve categorically the security posture of both new and legacy embedded devices, we must develop methods of delivering effective host-based protection onto large numbers of proprietary embedded devices running heterogeneous operating systems. We believe that a novel, injectable code structure called Parasitic Embedded Machines (PEM) [12] currently under development by the Columbia Intrusion Detection Systems Lab provides a viable solution to this challenging problem.

7. CONCLUSION AND FUTURE WORKS

We presented the first quantitative measurement of embedded device insecurity on a global scale as well as a preliminary longitudinal study tracking vulnerable embedded devices over a 4 month period. We developed an embedded device default credential scanner capable of efficiently and safely identifying vulnerable embedded devices on the network. The scanner does this by testing whether one can remotely login into a device using its well-known manufacturer supplied default credentials. Using this scanner, which currently monitors 73 common embedded device types, we identify over **540,000** publicly accessible vulnerable devices in 144 countries. Vulnerable embedded devices were discovered in 17,427 unique organizations on 6 continents including government, ISP, private enterprise, educational and satel-

lite provider networks. Preliminary results from our longitudinal study tracked 102,896 vulnerable devices discovered in December 2009. Out of the 54,429 devices currently online from the original population, **96.75%** such devices still remain vulnerable today. By breaking down the observed vulnerable embedded device population across functional, geographical and organizational categories, we were able to identify key groups which contribute a disproportionately large number of vulnerable devices to the global population. Lastly, using observations derived from the presented data, we proposed a set of realistic mitigation strategies to effectively reduce the total population of vulnerable embedded devices. This study demonstrates that there is a very large population of trivially vulnerable embedded devices available for exploitation by the least sophisticated adversary. We posit that the size of this vulnerable population can be significantly increased by escalating the level of sophistication of the assumed attacker. Since no widely available host-based defenses exist, vulnerable embedded devices constitute a serious and pervasive security problem.

8. REFERENCES

[1] kaiten.c IRC DDOS Bot. http://packetstormsecurity.nl/irc/kaiten.c.

[2] MaxMind GeoIP. http://www.maxmind.com/app/ip-location.

[3] Embedded Device Vulnerability Assessment Initiative. http://www.hacktory.cs.columbia.edu.

[4] IronKey Personal D200. http://www.ironkey.com/personal-solutions.

[5] The End of Your Internet: Malware for Home Routers, 2008. http://data.nicenamecrew.com/papers/malwareforrouters/paper.txt.

[6] Network Bluepill. Dronebl.org, 2008. http://www.dronebl.org/blog/8.

[7] Psyb0t' worm infects linksys, netgear home routers, modems. ZDNET, 2009. http://blogs.zdnet.com/BTL/?p=15197.

[8] Scan of internet uncovers thousands of vulnerable embedded devices. http://www.wired.com/threatlevel/2009/10/vulnerable-devices/, 2009.

[9] Time warner cable exposes 65,000 customer routers to remote hacks. http://www.wired.com/threatlevel/2009/10/time-warner-cable/, 2009.

[10] P. Akritidis, W. Y. Chin, V. T. Lam, S. Sidiroglou, and K. G. Anagnostakis. Proximity breeds danger: Emerging threats in metro-area wireless networks. In *In Proceedings of the 16 th USENIX Security Symposium*, pages 323–338, 2007.

[11] Hristo Bojinov, Elie Bursztein, Eric Lovett, and Dan Boneh. Embedded management interfaces: Emerging massive insecurity. Black Hat USA, 2009, 2009.

[12] Ang Cui and Salvatore J. Stolfo. Generic rootkit detection for embedded devices using parasitic embedded machines. Columbia University, New York. cucs-009-10., 2010.

[13] Felix "FX" Linder. Cisco Vulnerabilities. In *In BlackHat USA*, 2003.

[14] Felix "FX" Linder. Cisco IOS Router Exploitation. In *In BlackHat USA*, 2009.

[15] Andrea M. Matwyshyn, Angelos D. Keromytis Ang Cui, and Salvatore J. Stolfo. Ethics in security vulnerability research. *IEEE Security and Privacy (Vol. 8, No. 2)*, 2010.

[16] Michael Lynn. Cisco IOS Shellcode, 2005. In BlackHat USA.

[17] Sebastian Muniz. Killing the myth of Cisco IOS rootkits: DIK, 2008. In EUSecWest.

[18] Petko D. Petkov. Router Hacking Challenge, 2008. http://www.gnucitizen.org/blog/router-hacking-challenge/.

[19] Patrick Traynor, Kevin R. B. Butler, William Enck, Patrick McDaniel, and Kevin Borders. malnets: large-scale malicious networks *ia* compromised wireless access points. *Security and Communication Networks*, 3(2-3):102–113, 2010.

[20] Alex Tsow. Phishing with consumer electronics - malicious home routers. In Tim Finin, Lalana Kagal, and Daniel Olmedilla, editors, *MTW*, volume 190 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2006.