Digital Healthcare-Associated Infection: A Case Study on the Security of a Major Multi-Campus Hospital System

Luis Vargas, Logan Blue, Vanessa Frost, Christopher Patton, Nolen Scaife, Kevin R.B. Butler, and Patrick Traynor
Department of Computer & Information Science & Engineering
University of Florida
Email: {lfvargas14, bluel, vfrost, cjpatton, scaife, butler, traynor}@ufl.edu

***Abstract*—Modern hospital systems are complex environments that rely on high interconnectivity with the larger Internet. With this connectivity comes a vast attack surface. Security researchers have expended considerable effort to characterize the risks posed to medical devices (e.g., pacemakers and insulin pumps). However, there has been no systematic, ecosystem-wide analyses of a modern hospital system to date, perhaps due to the challenges of collecting and analyzing sensitive healthcare data. Hospital traffic requires special considerations because healthcare data may contain private information or may come from safety-critical devices in charge of patient care. We describe the process of obtaining the network data in a safe and ethical manner in order to help expand future research in this field. We present an analysis of network-enabled devices connected to the hospital used for its daily operations without posing any harm to the hospital's environment. We perform a Digital Healthcare-Associated Infection (D-HAI) analysis of the hospital ecosystem, assessing a major multi-campus healthcare system over a period of six months. As part of the D-HAI analysis, we characterize DNS requests and TLS/SSL communications to better understand the threats faced within the hospital environment without disturbing the operational network. Contrary to past assumptions, we find that medical devices have minimal exposure to the external Internet, but that *medical support devices* (e.g., servers, computer terminals) essential for daily hospital operations are much more exposed. While much of this communication appears to be benign, we discover evidence of insecure and broken cryptography and misconfigured devices, and potential botnet activity. Analyzing the network ecosystem in which they operate gives us an insight into the weaknesses and misconfigurations hospitals need to address to ensure the safety and privacy of patients.**

I. Introduction

Modern medical systems are digital, networked, and complex. From operations and finance to clinical departments, virtually every facet of a contemporary healthcare organization relies on interconnectivity with the wider Internet. Such connections bring with them significant benefits, from the ability to access patient records wherever they are needed to being able to incorporate revolutionary treatments as they become available. These systems hold the potential to improve quality of care while reducing patient costs and unnecessary therapies.

Much of this rise in connectivity can be attributed to the building of medical applications atop commercial off-the-shelf (COTS) systems and networks. As such, healthcare organizations have been able to rely upon much of the same expertise used in other industries to reap the advantages of "smart" devices and fast networks. This modernization has been rapid and has fundamentally transformed the way that healthcare is delivered in the developed world. However, connection to the larger Internet has come with notable risks. Like other systems relying on COTS components, hospital systems also inherit a large number of publicly disclosed and zero-day vulnerabilities [16], [22], [17]. Malware designed to send spam [33] and encrypt critical files for ransom [32] have already been found on machines within hospital networks, taking advantage of unpatched operating systems and weak security practices. Even devices long considered immune to compromise by nature of their isolation from other systems such as pacemakers and insulin pumps now allow remotely controlled malicious behavior [20], [28], [29]. All of these problems are amplified by unclear standards, resulting in the device manufacturers failure to fix vulnerable software for fear of requiring FDA safety reviews [24].

There is no doubt that medical systems have vulnerabilities. However, these individual examples and anecdotes fail to paint a broad picture of the current state of affairs because we lack a systematic, ecosystem-wide analysis of a modern healthcare system.

In medicine, Healthcare-Associated Infection (HAI) refers to the possible infections a patient may receive as part of the treatment they need while in a hospital. Our case study follows a similar goal but focuses on Digital Healthcare-Associated Infection (D-HAI). D-HAI can be described as either the characterization of health-related day-to-day traffic from network-enabled devices that are connected to the hospital network or the exposure to potential malware infection in their daily operational use. As the first such study, we focus on characterizing the hospital network traffic to assess its security as a whole rather than looking for specific signs of device infection. Critically, such an analysis must be done *without posing any potential harm to the hospital environment as it operates.* We make the first such characterization in this paper, and in so doing make the following contributions:

- **Assess a Digital Hospital Ecosystem:** We perform the first Digital Healthcare-Associated Infection (D-HAI) analysis on a major, multi-campus healthcare system. Our analysis captures traffic from across this system from January-July, 2018.

Network and Distributed Systems Security (NDSS) Symposium 2019
24-27 February 2019, San Diego, CA, USA
ISBN 1-891562-55-X
https://dx.doi.org/10.14722/ndss.2019.23444
www.ndss-symposium.org

- **Provide Guidelines for Ethical Research:** We explain our process and limitations of obtaining data from the hospital network in order to facilitate future academic research in studying this field in an ethical manner.
- **Categorize and Evaluate Outgoing Traffic Requests:** We collect and evaluate over 775 million DNS requests made from the hospital network. We show that traditional whitelisting and blacklisting efforts used to analyze the Internet do not cover a significant amount of traffic found in a hospital ecosystem. Additionally, while the majority of traffic appears to be benign, there are indications of malicious traffic within the network (e.g., potential botnet activity).
- **Characterize Security of Encrypted Communications:** We measure and evaluate the state of TLS/SSL communications based on our collection of 325 million handshakes across the hospital network. While we observe many positive trends (e.g., lower use of vulnerable versions of TLS/SSL compared to the larger Internet), we also record significant use of broken/deprecated cryptographic primitives and handshake modes, and some evidence of misconfigured devices.

While there are similar studies to this paper of various enterprises, no prior study has been done in a hospital network. The sensitive nature of this environment requires special care to ensure that our study is both ethical and safe to perform. As a case study, we discuss the *two year* process to bring together all of the necessary stakeholders to ensure that our study never threatened patient privacy or safety. While this process posed limitations and delayed our ability to conduct our study, we believe that it was an absolutely critical component of our efforts. We hope that other researchers will be able to follow a similar process in order to conduct an ethical and safe investigation in critically-sensitive environments.

The remainder of the paper is organized as follows: Section II gives a background on the network protocols studied and a topology of a hospital; Section III details the processes we undertook to ensure an ethical and safe study; Section IV explains our methodology and outlines the datasets; Section V shows our analysis of connection requests made by medical devices; Section VI focuses on the communication channel of medical devices; Section VII discusses our limitations and future work; Section VIII highlights related work; and Section IX offers concluding remarks.

II. Background

A. Hospital Networks

To provide timely patient care on a daily basis, hospitals rely on the availability and the infrastructure of their network. A hospital's network poses many unique challenges that other commercial networks may not. A major problem is that many devices throughout the network (mobile or stationary) need to have broad access to patient data at any given moment. Each device is thus a potential attack vector as even one infection could result in unauthorized access to thousands of personal records. Such leakage of data is not limited to just financial and personally identifiable information, but also health and diagnostic information not found elsewhere. While network-wide issues (e.g., DDoS attacks or outages) might create downtime and monetary losses to commercial-driven networks, *such issues can potentially be life-threatening* in a hospital setting. During such events a device used for patient care or the diagnostic information of a patient could become inaccessible.

Devices within the network include MRI machines, medical beds, surgical robots, and many other IP enabled medical devices. These critical devices must follow regulatory guidelines including FDA approval [24] and HIPAA compliance [30]. In addition to those unique devices only found in a hospital, other devices such as printers, accounting computers, and doctor's laptops make up a large portion of the network.

B. Network Protocols

Our study involves traffic from two networking protocols: DNS and SSL/TLS. We chose DNS because it allows us to see the domains being visited by devices within the network without revealing additional information that may compromise the privacy of those connections. SSL/TLS were chosen for this study because they are the most widely deployed security protocols.

The Domain Name System (DNS) maps human-readable domain names (e.g., `www.domain.com`) to machine-readable IP addresses (e.g., `1.2.3.4`) among a range of data types. DNS is separated into organizationally-controlled *zones* that are arranged in a hierarchical structure, with each zone having information about itself and links to the sub-domains beneath it [39]. Zones in DNS are named based on their position within the hierarchical structure: `www.domain.com` would have a top level domain (TLD) of `.com`, and second level domain (2LD) of `.domain`, etc. On occasion, the effective second level domain (e2LD) is used to signal the canonical name of the domain (e.g., `google.co.uk`) since registration is only allowed at the third level and below.

The response of a DNS request is sent via resource records (RR). Passive DNS records can indicate that a device inside a network has attempted to resolve the address of a known malicious domain. While the request alone does not imply that the device is necessarily compromised or malicious, it may warrant investigation of the device itself [11].

Once DNS provides a method for devices to find each other on the Internet, the Secure Socket Layer (SSL) protocol and its modern successor, the Transport Layer Security (TLS) protocol, provide a cryptographically secure communication channel between them. To establish a secure connection with a server using SSL/TLS, a client must first validate the identity of the server it is communicating with.[1] To do so, a server presents the client with an X.509 certificate containing the server's identity and signature. Certificates are issued by one of a number of Certificate Authorities (CAs) that assert the server's identity. It is up to the client to track which CAs it deems trustworthy; if the CA that issued the server's certificate is trusted by the client, then the client validates the X.509 certificate and begins secure communication with the server.

Many versions of both protocols have been deprecated and deemed insecure for various reasons, ranging from susceptibil-

[1] It is possible, but less common, for the server to also validate the client's identity.

ity to downgrade attacks to the use of insecure cryptography. At the time of writing, the current acceptable protocol standard in use is TLS 1.2; however, early implementations of TLS 1.3 are already being deployed [27].

III. Designing and Executing an Ethical Study

Designing this study required legal, institutional, regulatory, and self-imposed limitations to protect the safety and privacy of the hospital. There have been multiple papers published in the security community over the last decade that have caused significant discussions about ethics. While these papers are often cleared by the university's Institutional Review Boards (IRBs), the implications of the work are often not clear to these approval boards. The resulting papers clearly push the boundary of community norms and are published "asking for forgiveness instead of permission."

Such a cavalier approach is not possible in our setting. In addition to the potential to violate patient privacy and run afoul of the law (e.g., HIPAA in the United States), studies of medical ecosystems must also ensure that they do not interfere with patient care or safety, nor the anonymity of the hospital workers. Accordingly, we must make sure that our study *by design* minimizes any potential for such a negative impact.

Achieving these ends has taken *over two years* of planning and effort. Prior to presenting a detailed study to our IRB, we met with legal counsel for both our university and the hospital system. We then worked in conjunction with IT staff from both organizations to determine the feasibility of any requested analysis and its potential impact. We also provided such information to our funding agency.

A. Design Process and Limitations

Agreeing to the details of our study required multiple rounds of discussion with stakeholders. In particular, the legal team and hospital IT staff requested more specificity from our original proposal regarding the following issues:

Limiting traffic source collection: We were required to select only data sources that pose low risks to the hospital. More importantly, we needed to ensure that private information of the patients or hospital workers was not present in this case study. Data coming from traffic payloads, packet captures, or protocols that may contain unencrypted information, such as HTTP (containing usernames, passwords, or paths to files) or P2P protocols, were forbidden as they had a high risk of containing private information. Additionally, the hospital deemed the collections of DHCP to carry a moderate risk as this data can be used to track or deanonymize hospital workers. Similar studies in the future must carefully identify the potential risk of each traffic type before collecting them and must coordinate with the potentially impacted parties.

Ensuring undisrupted daily operations of the hospital: We also needed to perform our case study in such a way that the hospital's daily operations would not be interrupted. For example, active analysis or active probing of network-enabled machines could have revealed a more thorough characterization of hospital devices (e.g., determining the services running on a device). However, adding probing traffic to devices with low resources could accidentally bring them down and thereby hinder the hospital's daily operations. As such, the legal team deemed the use of network scanning tools such as nmap or Nessus to have a high risk of interfering with daily operations and thus were not used as data sources in our case study. Future research in this area needs to be mindful of which devices can be scanned without overloading them with extra traffic.

Eventually, all parties agreed to a limited and purely passive analysis of the network. The analysis focusing solely on DNS and TLS/SSL traffic could be conducted without risking patient privacy or safety and would allow meaningful characteristics to be extracted. For DNS, we only focus on the IP address information returned and no other information such as email routing or additional domain names that allow for reverse IP lookups. This leaves an unknown amount of traffic unseen by our study, which could affect the results. However, we consciously accept this limitation to preserve the privacy and daily operation of the hospital.

These requirements give us access to significant amounts of data in a protected fashion. For instance, patient data such as electronic medical records are unlikely to be captured in this configuration, nor are we capable of impacting the availability of any device due to unexpected probing. However, these requirements also create important limitations. Whereas previous studies of networked environments are able to conduct in-depth analyses of specific machines or users [49], we were not able to do so. While appropriate in the case of an enterprise environment, our prioritization of privacy and safety forbid such analyses. We attempted to compensate for these limitations and mitigate threats to external validity by using public sources of data [10], [19], [21], [35] to provide ground truth for our observations.

After all requirements were met and agreed upon, data collection started and was conducted by both the hospital staff (i.e., hospital traffic) and our team (i.e., OSINT information to complement collected data). The hospital traffic collection process was handled entirely by the hospital. While letting us set up the collection mechanism would allow us more freedom in getting extra information, it is critical for the hospital to take care of this step. This way any information shared with us was pre-filtered to meet their privacy requirements. However, after seeing the exact fields that were being collected from the monitoring tool, the hospital did not require extra pre-filtering steps. Additionally, since the data gathered would be purely passive, we did not add extra traffic load (e.g., active probing) to hospital devices.

B. Disclosure of our Findings

In our community, it is standard practice for researchers to notify the impacted/responsible parties of any issues found during research. Besides the limitation posed by the legal team and IRB, we also agreed with the hospital to report any concerning findings to them in a timely fashion. Our agreement with administrators entailed one-way communication in the reporting of issues. Our responsibility was to report any possible vulnerabilities, but it was up to the hospital staff to take actions (without needing to report back to us). While this limiting agreement was specific to our study, it was necessary to establish a relationship built on trust so that we can perform future research beyond this case study. Additionally, we agreed

to keep the identity of the hospital anonymous for confidentiality. As such, to disseminate our analysis, we first allowed the hospital staff to look at our work and suggest fixes where needed. The report we present purposefully abstracts results in a way that reveals the minimum amount of information about the hospital while maintaining meaningful results. A similar agreement could be made by other researchers that wish to examine hospital security.

We believe that the careful design of our experiments, the inclusion of legal and IT professionals from all parties, and the strict requirement for limited passive techniques were necessary for conducting a safe and ethical analysis of the hospital. While our case study gives us an insight into a large healthcare ecosystem, generalizing the results to other hospitals may constitute a threat to population validity as many unknown variables may have affects on the analysis (e.g., size of the hospital, funding available). As we discuss further in Section VII-B, arranging such study a requires the collaboration of multiple administrations. However, we hope that this work and the guidelines suggested can aid future researchers in designing similar ethical experiments.

IV. Methodology

Our research focuses on performing a D-HAI analysis. Before explaining our methodology, we first need to distinguish *medical devices* from *medical supporting devices*. In the context of our work, a medical device has direct contact with the patient while they are inside the hospital (e.g., MRI machines, hospital beds). Conversely, medical supporting devices aid in patient care but do not necessarily come in direct contact with the patient (e.g., laptops, computer terminals, databases). Both sets of devices are essential in the daily operations of a hospital.

Prior research examined the security of specific medical devices for possible vulnerabilities [25], [28], [29]. However, while medical devices are connected to the network and may be vulnerable, we found in our analysis that network administrators use multiple mitigation techniques to isolate these devices. For example, the vast majority of medical devices are locally connected to *aggregation points*. Access to such aggregation points is extremely limited even from within the network (e.g., limited VPN access to the aggregation points). Additionally, much of the prior work has shown attacks on medical devices happening on the first-hop communication (i.e., wireless channel) rather than communication in the network. In other words, from the network's perspective these medical devices are mostly invisible[2] to other devices inside the network, let alone the wider Internet. As such, much of the hospital's exposure to malicious activities comes from medical supporting devices, rather than medical devices themselves. *This observation is extremely important to note because it tells us that while previous research in the medical device security field is important, it does not necessarily address the attack surface that a hospital may present.* This case study is meant to complement previous literature in medical device security.

[2]There may be a few exceptions, such as MRI machines, that may require an Internet connection for updates from the manufacturing company.

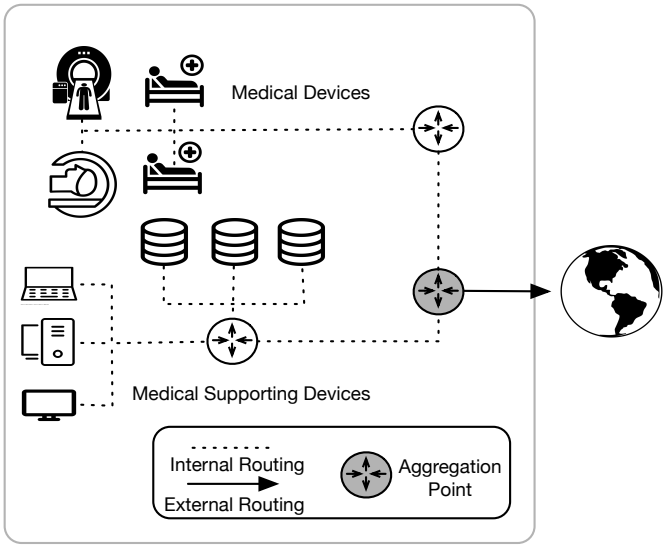

Fig. 1: Illustration of the typical topology of network-enabled *medical devices* (e.g., MRI machines, hospital beds) and *medical supporting devices* (e.g., laptops, computer terminals, mobile devices).

A. Hospital Traffic

The modern hospital system we studied, which is representative of many modern hospital environments, is comprised of a partnership between a hospital and an academic research center (e.g., a university). Collaborating with academic research centers allows for various research opportunities that benefit both sides. While access to electronic health records (EHR) is limited in the academic research center and administration/security standards might differ between networks, we decided to include traffic from both networks in our study since they have access to private health records. For the rest of the paper (and for simplicity), we define hospital. to include both the hospital itself and the academic research center that has access to private data.

It is common for hospitals to offer public WiFi, but in several cases (including ours), this is a different physical network than the ones used for medical and medical supporting devices. As such, traffic coming from the public network was not analyzed as it would not be representative of the hospital's ecosystem

To analyze traffic from all devices in the hospital network, we partnered with an anonymous multi-campus, state-wide hospital. Hospital IT staff deployed and configured the Bro Network Security Monitor (version 2.5.2) [41] on our behalf. Our instance aggregated data at a single point in the network trunk that connects the hospital to their Internet service provider (ISP) (shown in Figure 1). From this vantage point within the network, we are able to see all ingress and egress traffic to the hospital network regardless of which campus it originates from. We collected DNS requests, TLS/SSL sessions and resumed handshakes, and X.509 certificates for a period of six months, from January 1st to July 1st, 2018. Traffic was passively collected to avoid disturbing or probing any devices in the network. Our monitor was placed in a

location allowing us to see traffic from the hospital's external and internal IP addresses before being remapped by network address translation (NAT) for outside use.

Throughout our collection period, we were in constant communication with hospital administrators and reported any anomalies or malicious activities we found. Unless explicitly stated, all DNS analyses in the paper are evaluated using `A` and `AAAA` requests.[3] From this data, we are able to see both medical and medical supporting devices.

B. Ground Truth

We also collected data from public sources in order to compare the hospital's traffic to the rest of the Internet. Due to the lack of prior data with which to compare our hospital data, we establish a ground truth for benign DNS traffic by gathering the daily lists of the top one million domains from Alexa [10] and OpenDNS [19]. Traffic rankings found in these lists are reflective of popularity by how many users throughout the Internet have requested information about them. Our assumption is that we can use the most popular domains as a whitelisting source because user-generated traffic would tend to query suspicious websites less often than those found in the top domain rankings. Additionally, we use both of these lists to better understand how a hospital's traffic might differ from Internet traffic as a whole. We used the Alexa top sites list because it is widely used for whitelisting [12], [36] and provides a popularity ranking more sensitive to effective second level domains (e2LDs) (e.g., `example.com` and `google.co.uk`). By being sensitive to e2LDs, the Alexa top list represents the largest entities (e.g., service providers) on the Internet accurately, making it ideal for rating certificates. The OpenDNS top one million list, in turn, provides a more sensitive popularity ranking of e2LD's subdomains (e.g., `analytics.domain.com`). This makes the OpenDNS dataset a more accurate representation of the largest services used on the Internet and is thus ideal for ranking our DNS data. Our goal is to be conservative with our whitelisting definition. As such, we used the top 100,000 domains (rather than all sites) of OpenDNS as a way to whitelist traffic found in the DNS data.

Conversely, to create a ground truth list of malicious traffic, we crawled six different publicly available blacklists.[4] While no blacklist is complete, we believe that the combination of our sources gives us a reasonable basis for classifying ill-intended Internet traffic. These data sources include domains that have been tied to malware, phishing, or other traffic thought to be malicious. Combining these blacklists, however, engenders some challenges. First, some blacklists contain false positives, which can happen when reputable domains are reported for various reasons (e.g., advertising) that are not necessary ill-intended. Usually, if such requests get the reputable domain added to the blacklist, it will only remain for a short time. However, those domains would still appear in our dataset since we keep track of all added domains. To fix this issue for our analysis, we manually removed any domain that appeared in our whitelist described above from the malicious traffic found in the blacklists. Additionally, DNS domains that belong to content distribution networks (CDNs) are problematic for these blacklists. CDNs will redirect traffic to multiple sites and will be added to the blacklists if a domain it serves is used maliciously. Since we do not know if the domains used by the CDN were malicious at the time of query, we removed them from our analysis to be conservative with our results.

While we set a ground truth for DNS traffic, we also need to establish a baseline for the hospital's TLS/SSL communications. As no such baseline exists, specially for hospital networks, we collected weekly data from Censys [21], a platform that scan the entire publicly accessible IPv4 address range and collects information from various protocols. Over 97% of TLS/SSL traffic from the hospital goes to TCP port 443. Accordingly, to get a control population for communications and cipher usage, we specifically looked at traffic scans to TCP port 443 of Internet devices found on Censys.

Finally, we collected certificates from Certificate Transparency (CT) logs [35]. These are public append-only logs that keep track of all certificates presented to them. The logs are constantly updated with new certificates, making them tamper-evident: it is difficult for certificates to be issued without alerting either the domain owner or clients. Additionally, the logs are both monitored and publicly auditable, allowing any user to check their integrity at any time. We use the certificates collected from the CT logs to check for mis-issuance of X.509 certificates.

V. DNS Analysis

During our collection period, we collected over 775 million DNS queries to over 17.2 million unique Fully Qualified Domain Names (FQDN). The responses of these queries returned over 2.5 million distinct destination IP addresses. The collected raw data comprised over 179 GBs. In this section, we analyze the DNS data and compare it to the OpenDNS dataset and blacklisted traffic to characterize potentially malicious behavior.

A. General DNS Behavior

We conducted an analysis on the top 100 e2LDs present in both our dataset and the OpenDNS data. We chose to analyze only the top 100 domains because this analysis was conducted manually in order to determine affiliation and role of every domain. To accurately assess the role of every domain, we resolved it and then categorized it as either entertainment, service related, or hospital related. The top 100 domains account for over 60.98% of the traffic in our dataset.

Shared Top Domains: Overall, the top 100 domains from our dataset and the OpenDNS dataset had 36 domains in common. Domains that fell into this category included large entertainment domains (e.g., `netflix.com`), search engines, and large services such as `microsoft.com` that are likely used by the hospital. These shared domains accounted for approximately 31% of the total traffic in our dataset. We had initially expected to see a larger overlap between the top sites from the hospital and OpenDNS. However, upon observing our

[3] 27.4% of the data had Null as its record type. We manually inspected the data and concluded that these queries should have been labeled `A` or `AAAA`. These records have been kept in the dataset used in this paper. Additionally, while data can be encoded inside a DNS request, we observed no such activity in regards to PII exposure.

[4] We collected data from Phishtank [6], Zeus Tracker [7], MalwareDomains [4], Dshield [2], and OpenPhish [5].

own top e2LDs it became clear why there is a large deviation between the two datasets.

Infrastructure Domains: 36 of our dataset's top e2LDs were directly related to the hospital. These domains were either controlled by the hospital itself, large-scale service providers employed by the hospital (for services including security, anti-virus, data analytics, etc.), or directly intended for use by medical providers (e.g., the National Institutes of Health's domain `nih.gov`). Overall, infrastructure related-domains generated 18.6% of the total traffic in the network.

Other Unshared Top Domains: The healthcare-specific domains were a part of a larger set of 64 domains that were present in the hospital's top domains but not in the OpenDNS dataset. The non-infrastructure domains accounted for 11.3% of the total traffic in our dataset. The other domains that were not affiliated to healthcare were predominately content distribution networks, software provider domains (e.g., `mozilla.com`), smaller entertainment domains, or various regional domains. We believe that these sites simply represent the unique subculture and demographic makeup of the hospital system which we were studying. We do not go into any further detail about these domains to prevent deanonymizing the hospital which we observed.

By comparing the hospital network's top domains to the top domains of the Internet at large, we observe the hospital network significantly deviating in several ways. From this D-HAI analysis, we saw that the top domains for medical supporting devices contained a large quantity of services that are directly related to the hospital's functions. Additionally, we saw a large number of domains that were associated with anti-virus and network security providers. More interesting are the services that were not present in our dataset. Overall it appears that medical supporting devices interact with entertainment domains, cloud providers, and non-work related domains less than the Internet at large. While expected, we believe this is still worth mentioning for the following reason: given the sensitive nature of medical records, having a limited domain footprint decreases the likelihood of a computer being compromised through a web browser via a drive-by download or malicious JavaScript.

B. Traffic Categorization

As part of the D-HAI analysis, we categorized our data into three sets in order to contextualize it. Our data is divided into whitelisted, blacklisted, and unknown categories. In total, we categorized $502,051,633$ requests or 64.78% of the DNS traffic as whitelisted. The traffic that was whitelisted came from 119,117 (0.69%) unique FQDN that were associated to 29,085 (1.68%) distinct e2LDs. From these values, we can see that on average each e2LD has 4.09 unique FQDNs (e.g., `mail.domain.com` is an FQDN of `domain.com`) associated with it. We categorized 84,669 requests or 0.01% of the traffic queries as connections to blacklisted domains. Overall, the blacklisted traffic was intended for $2,483$ unique domains across $2,281$ unique e2LDs. Unlike the whitelisted domains, it appears that blacklisted domains have a lower association rate between unique FQDNs and distinct e2LDs. On average, an e2LD in our blacklisted dataset has approximately 1.08 FQDNs associated with it.

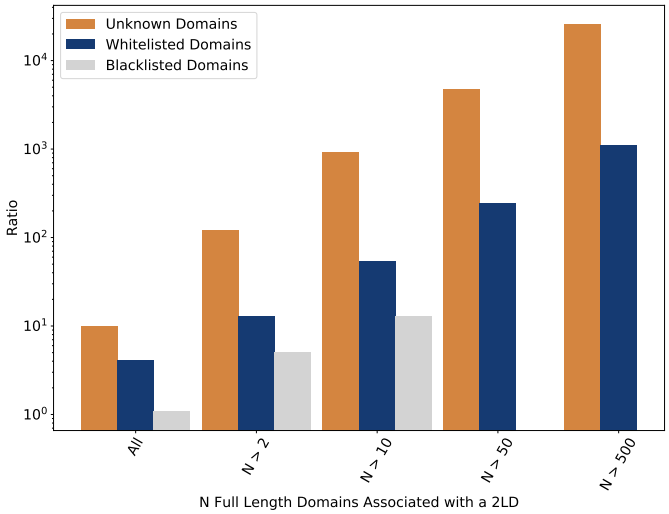

Fig. 2: Each category of domain is divided into five divisions based off the number of FQDNs that associated to each e2LD. We observe that domains found in our unknown traffic regularly have a higher amount of self-association than blacklisted domains. This gives us reason to believe that the majority of domains with the unknown category are more similar to those in the whitelisted category and therefore benign.

We believe this occurs because reputable domains are more likely to want to reuse their e2LD for multiple sub-domains. By doing so, reputable domains can easily pass on their reputations to their sub-domains. Unlike reputable domains, malicious domains do not wish to pass their reputation to newer sites and are thus more likely to change their e2LDs frequently. This is illustrated in Figure 2, which shows the average number of sub-domains for each e2LD within a certain subset of that category. The categories are divided into subsets by the number of FQDNs that are associated to each e2LD. For example, the $N > 2$ subset contains only those e2LDs that had at least 3 unique FQDNs associated with it. If we assume our whitelisted and blacklisted categories are accurate representations of benign and suspicious traffic, then we can gain some perspective on the unknown category through the same analysis.

We can use this same technique to characterize malicious behavior in the unknown category of domains. In total, the unknown category contains 271,363,949 unique requests or 35.02% of the total traffic. These connections went to 17,186,699 (99.72%) distinct FQDNs across 1,721,311 (99.76%) e2LDs. On average, e2LDs in the unknown category are associated with 9.98 unique FQDNs. This rate of self-association makes the unknown category appear to be less malicious. However, this category had a long tail of FQDNs (72.4%) that were only queried once. 93.9% of the domains had under 10 total queries targeting them. When we look at the traffic distribution as it relates to e2LDs, queries that had more than 50 requests accounted for only 4.64% of our total e2LDs but over 85.03% of the traffic in the unknown category. In fact, some of the most visited domains in our dataset ended up in the unknown category.

While the above metrics tells us that the distribution of

queries is highly concentrated to a few e2LDs, it does not give us insight into how we should categorize the queries. To see if unknown traffic is indeed behaving similarly to whitelisted traffic, we collected all e2LDs present in our whitelisted traffic and checked if the unknown category also had those e2LDs present. In the case of an e2LD collision, we mark the query as benign due to self association. By examining these collisions, we mark almost half (45.25%) of the unknown traffic as benign. This occurs because many services tend to use one time DNS requests to encode information (e.g., anti-virus services). While the service's e2LD may be part of the top 100,000 domains, the one-time DNS request will not be.

Finally, after removing the e2LD collisions of the unknown queries, only 19.17% of the hospital's total traffic remained in the unknown category. After manually checking the top e2LDs, the remaining traffic appears to be predominately comprised of domains useful to the hospital's operations. These include domains internal to the hospital, outside services (e.g., customer relations management), and software that was purchased by the hospital for daily healthcare operations (e.g., payroll and administrative). This is consistent with the previous subsection's analysis of the hospital's top 100 domains. Although we cannot claim that all traffic categorized as unknown is universally benign, it appears that the majority of it is.

While the effectiveness of blacklists/whitelists for the open Internet are still unknown, *this shows that blacklists and whitelists used to categorize Internet traffic may miss large amounts of traffic seen in healthcare networks.* We believe that hospital medical supporting devices and their networks could benefit from more customized whitelists and blacklists for domains specific to the hospital. Given our limited visibility into the internal network, we are unable to perform this analysis here.

C. Potentially Malicious Behavior

While investigating the DNS data, we found several signs of potentially malicious activity. Specifically, we looked for known botnet command and control (C&C) channels, as well as spam networks and other known malicious actors. We used a curated list of known entities from `emergingthreats.net` collected on April 3 and July 30, 2018. This list contained several categories of threats including IP addresses for the Feodo and Zeus botnets, spam nets identified by SpamHaus, and the top attackers listed by DShield. We compared this list to our full dataset. We found $5,552$ connections to IP addresses that were members of the `emergingthreats` dataset. When we looked for when these queries were made, there appeared to be no discernible pattern throughout our collection period. This is concerning given the highly sensitive data that medical supporting devices access. However, further analysis is needed to determine whether or not our concern is warranted.

Of the threat categories contained in the `emergingthreats` list, only IP addresses related to Zeus and Feodo were found in the hospital network traffic. In addition, only 0.0007% of the total traffic in our dataset was related to potential bot activity. While the low volume of potential activity appears reassuring, the fact that any exists is still concerning as DNS data does not reveal additional communication occurring over other protocols. The low rate of bot activity could represent false positives generated by misclicks, temporal artifacts of the `emergingthreats` list (e.g., a site visited months before or after it was deemed malicious), or collisions caused by CDNs. We were not able to confirm the intent of these queries from our network vantage point, but we notified system administrators of our findings.

The vast majority of bot activity seen was related to Zeus, accounting for over 94.78% of potential bot queries. The connections were made to $1,722$ unique FQDNs associated with $1,514$ distinct e2LDs. Interestingly, all the domains in our botnet traffic resolved to just 37 unique IP address. The top 3 IP address had 540, 341, and 308 different e2LDs associated with them, accounting for over 77% of unique e2LDs observed. On further investigation, we found that two of the three IP addresses were controlled by domain hosting sites. The last IP address was for a traffic redirection site for Internet advertisements. While these services may have once hosted malicious activity, we have no indication that they are still actively malicious.

D. Summary

The DNS analysis highlights several aspects of the observed hospital network. First, the top domains that traverse this network are substantially different from those of the Internet at large. The traffic indicates that the majority of domains visited on the network are related to the hospital's healthcare role, thus shrinking the network's attack surface. While the hospital may benefit from whitelists and blacklists intended for general use on the Internet, this type of categorization misses a large section of domains specifically related to the hospital (e.g., their AV service) and a more domain specific categorization method would greatly benefit the community. Additionally, there were detectable, albeit small, signs of malicious actors in the network. While our analysis was not conclusive with regards to their benign or malicious activity, administrators should be concerned that these actors represent potential threats to the hospital network and could lower the network's overall ability to provide patient care if left unchecked.

VI. TLS/SSL Communications

Because hospital networks contain EHRs and other personal identifiable information (PII), secure communication is important for ensuring the integrity and confidentiality of such data. DNS requests tell us from whom the medical supporting devices are requesting information, but provide only a limited view of the security of their communications. To broaden the scope of the D-HAI analysis, we also collected quality metrics for TLS/SSL sessions we observed. In particular, this section focuses on the protocols and cipher suites negotiated in established sessions, as well as certificates presented by the server in fresh (non-resumed) sessions. The vast majority of sessions (97.66%) involved no client certificate authentication, as is expected. The conclusions in this section uses the Censys' data as a baseline, which reflects the server's preference of TLS/SSL establishment parameters.

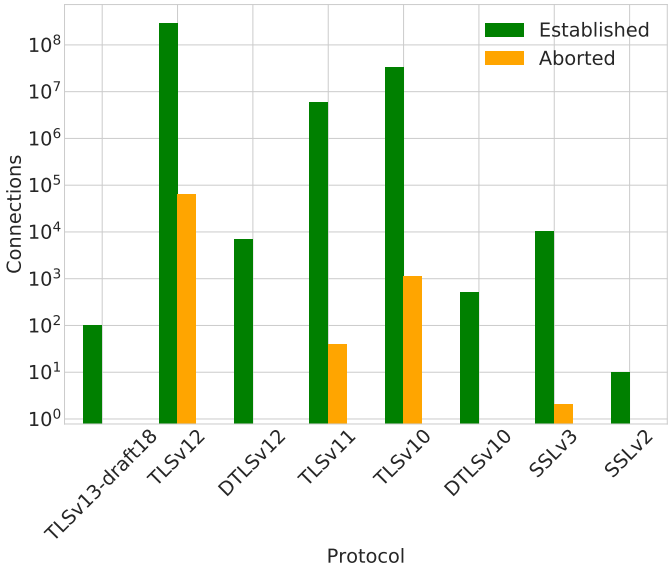

Fig. 3: Number of handshakes for each TLS/SSL protocol. TLS 1.2 usage is at least an order of magnitude greater than any other protocol version.

A. TLS/SSL Usage

We observed approximately 325 million TLS/SSL established handshakes during the six month data-collection period. Figure 3 provides a summary of handshakes classified by protocol: TLS 1.2 made up the vast majority of the traffic, comprising 87.88% of all observed handshake attempts; TLS 1.0 was the second most used at 10.33%, and TLS 1.1 was the third most used at 1.78%. The remaining 0.01% consisted of TLS 1.3,[5] DTLS 1.0 and 1.2, and the deprecated SSL 2/3 protocols. The connection rates were highest for (D)TLS 1.2 and TLS 1.0, in which the majority of handshakes completed. The majority of handshakes did not complete for all other protocols.

Our data covers all established TLS/SSL connections that passed through the hospital network. We found that over 97% of connections were made using TCP port 443; thus, we created a baseline by using the IPv4 scans of TCP port 443 found on Censys (Section IV-B). The scans indicate that 76.61% of servers on the Internet prefer TLS 1.2; 20.90% prefer TLS 1.0, 1.59% prefer 1.1, and only 0.89% prefer SSL 3.[6] Interestingly, of all inbound and outbound connections, the rate at which TLS 1.2 (87.88%) is actually used by devices within the hospital network is significantly higher than the proportion of servers on the Internet that prefer this protocol. Unfortunately, the above metric is not a one-to-one comparison because it is not possible for us to get all traffic from all the Internet servers. However, looking at this metric gives us a good idea of how medical supporting devices are behaving in regards to preferred communication methods. This indicates that the rate of TLS 1.2 usage is likely higher among sessions in our network than the Internet at large. (We cannot say for sure, since the Censys data is limited to server scans.) Another positive observation is that the rate at which TLS 1.0, which is vulnerable to POODLE downgrade attacks [40], is negotiated (10.44%) is significantly lower than the baseline. From these results, we can see that medical supporting devices in the network we observed make use of secure protocols more frequently than the rest of the Internet.

[5]The TLS 1.3 protocol was an Internet draft during the data-collection period. Compliant implementations of this protocol indicate which draft is being used; we observed only draft 18.

[6]The Censys baseline also contained a negligible number ($< 0.001\%$) of connections referring an "unknown" protocol.

B. Cipher Suite Quality

On top of the TLS/SSL protocol analysis, we also looked for the cipher suite negotiated in each connection involving medical supporting devices. To measure the overall cipher quality used in these connections we assigned each negotiated cipher to one of four categories:

- *Secure.* The session uses strong primitives with no known attacks: AES-GCM or ChaCha20+Poly1305 for encryption, ephemeral (EC)DH for key agreement, ECDSA or RSA for authentication, and SHA2 or higher for hashing.
- *Weak.* The session uses strong primitives, but there is a known attack against it: sessions using CBC-mode for encryption are vulnerable to the "Lucky 13" attack [8], a sophisticated variant of the "padding-oracle" attack [46] that recent versions of TLS are designed to mitigate; and sessions using RSA encryption for key transport are vulnerable to ROBOT [17], a modern variant of Bleichenbacher's attack against PKCS#1 v1.5 [16] that allows an adversary to break the confidentiality of a TLS session.
- *Insecure.* The session uses one or more insecure primitives: we observed the RC4 stream cipher [38], the 3DES block cipher [14], and the SHA1 hash function [44] in wide use.
- *Broken.* The session has effectively no practical security: the DES or export-grade primitives used for encryption, the "null" cipher, resulting in no encryption at all, anonymous Diffie-Hellman (DH), which permits a trivial man-in-the-middle attack, and the broken MD5 hash function.

In Figure 4, we show the overall cipher quality for each day of our collection period. The cycles seen in this figure are due to weekday and weekend traffic patterns for the hospital workers. Throughout our collection period, the cipher quality appears to be relatively stable. However, by our measure of quality only 53.21% of the sessions are deemed secure; 19.61% are weak, 27.08% are insecure, and 0.11% are broken. While the cipher quality for connections appears to be stable throughout our collection period, all secure cipher suites were exclusively negotiated using TLS 1.2 or higher. This suggests that as deprecated protocols disappear, the frequency of secure cipher negotiations will be higher.

1) Hash Functions: The high number of insecure sessions is due to the continued use of the SHA1 hash function during the handshake. We expect to see SHA1 in wide use since it was deprecated just last year [44]. However, we also expect its usage to decrease over time as servers do routine updates. As shown in Figure 5, no such trend is observed during the six months we collected data. As evidence of this observation, we performed an Augmented Dickey-Fuller (ADF) test. This is a null hypothesis test that checks if a time series is stationary

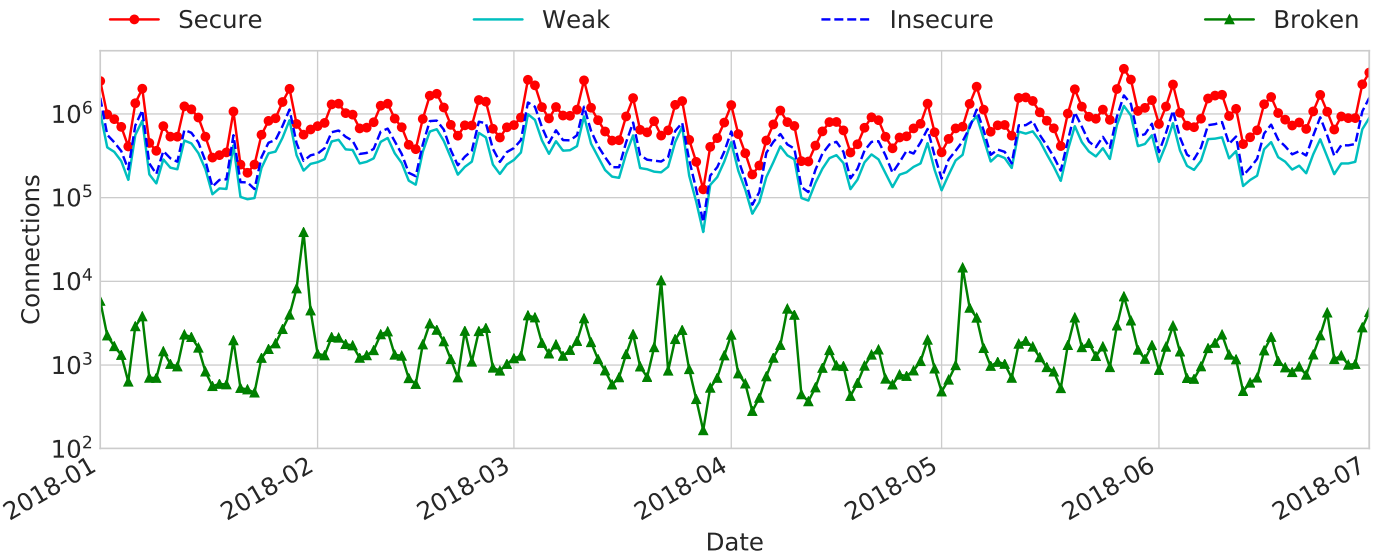

Fig. 4: Quality of cipher suites used in TLS/SSL sessions by medical supporting devices. The volume of sessions established with non-secure cipher suites appears to be stable throughout our collection period.

or non-stationary. We set $\alpha = 0.05$ and determined that the SHA1 usage for medical supporting devices is stationary ($p < 0.0001$). This indicates that SHA1 is likely to remain in use for the foreseeable future perhaps as long as TLS 1.2 remains the most used protocol of servers. We expect its use to decrease as TLS 1.3 enters wide adoption.

We also looked for the use of other cryptographic hash functions both by medical supporting devices and the Internet at large (Figure 5).[7] First, we note that the broken MD5 hash function is almost completely phased out from TLS/SSL communications as connections made with this hashing algorithm make up less than 1.5% of daily sessions in both datasets. Next, we looked at the percentage of SHA2 variants: SHA256 and SHA384. With respect to hashing, the biggest discrepancy between medical supporting devices and the Internet is due to SHA384 as it appears to be nonexistent in the Internet (less than 0.01%) while making up 25.93% of the daily established sessions for medical supporting devices. To determine how many sessions use a secure hashing algorithm, we must add all daily sessions made using SHA256 and SHA384 for both datasets (though secure, SHA512 was not found in either dataset). From this addition, we determined that medical supporting devices use secure hashing more often than other Internet devices (a difference of 10.82%). This higher rate may be attributed to a combination of factors ranging from the browsers used by medical supporting devices being updated more frequently to hospital networks having stricter policies. Finally, we note that no connections were established using the SHA3 hash function in both datasets. SHA3 represents the most modern development in hash function design [13].

2) Forward Secrecy: A session is said to be forward secret if a key compromised in the current session does not permit an adversary to decrypt prior sessions between the two parties. Sessions that use ephemeral DH have this property; resumed sessions, or sessions that use static DH or RSA for key transport do not. In the context of medical supporting devices, forward secrecy is a desired property because it limits the possible exposure of data traversing the hospital to a single session as opposed to multiple (e.g., EHRs transferred over the network in multiple connections). Overall, 81.14% of sessions

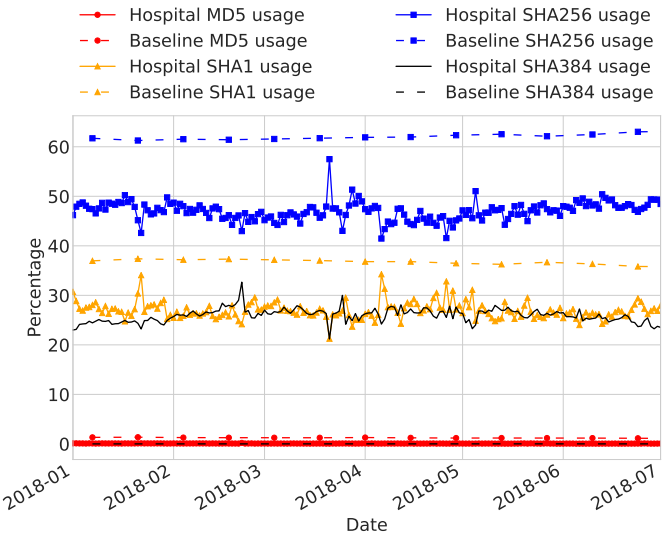

Fig. 5: Percentage of daily connections using various hashing algorithms. In total, medical supporting devices show an average of 72.84% secure sessions (SHA256 and SHA384 combined) while the baseline only accounts for 62.02% secure sessions. Interestingly, while SHA384 is used in about 25% percent of daily communications in a hospital, it only makes up 0.01% of the baseline.

[7]We note that for this study, we resampled the Censys baseline data to summarize two weeks, rather than one, due to a small sample size in mid-February.

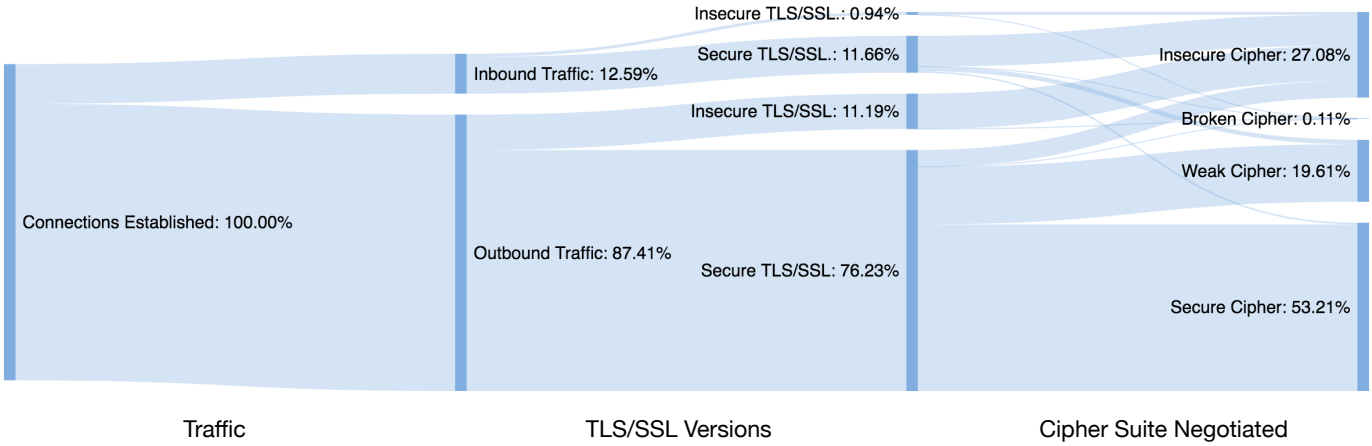

Fig. 6: Breakdown of established connections. We say the version is secure if (D)TLS 1.2 or higher was negotiated, since secure cipher suites were only negotiated in these protocols. The quality of the ciphersuite is categorized according to our criteria in Section VI-B.

for medical supporting devices were found to be forward secret. On the other hand, only 72.3% of servers scanned by Censys prefer a cipher suite that is forward secret; depending on the rate of resumption of sessions on the Internet at large, this may indicate a higher rate of forward secrecy among sessions in our network.

3) Broken Ciphers: Only 0.11% of the sessions used a broken cipher suite. While the percentage is low, these account for 351,105 sessions that have no security against an on-path adversary. Below, we highlight the more significant findings.

First, we found that 0.08% of sessions used the `TLS_RSA_WITH_RC4_MD5` cipher suite in TLS 1.0. These use RSA for key transport and authentication, RC4 for encryption, and MD5 for hashing. We regard this as broken because MD5's weaknesses permit the adversary to easily forge ciphertexts and transmit them to the client or server. The use of RC4 is also concerning, since it is known to leak part of the plaintext to the adversary. These connections seem to only have two end points: a mail server likely serving legacy clients outside the network and another server owned by a health care company that deals with medical IT. The latter is concerning as it allows data packets to travel outside the network with little integrity and weak confidentiality.

Second, 0.03% of sessions that have been established negotiated `TLS_RSA_WITH_NULL_SHA256` in TLS 1.2 with an external server. This suite uses RSA for authentication and SHA2 for hashing, but data in these sessions traversed the network entirely in the clear (with no encryption). Thus, anyone on the communications path can access the data.

C. Directional Traffic

So far we have addressed the security of established TLS sessions overall, but in order to better understand the security of medical supporting devices in the network, it is necessary to investigate the behavior of inbound and outbound connections separately. (Inbound connections are made to a server in the network from a client outside the network, and outbound connections made to a serer outside the network from a client inside the network.) This gives us crucial insights into how these endpoints might be configured without actively scanning the end devices and disrupting the hospital's daily activity.

Figure 6 breaks down the established connections into the following categories: inbound and outbound; among these categories, whether (D)TLS 1.2 or above was negotiated; and among these categories, whether the negotiated cipher suite was secure, weak, insecure, or broken according to our criteria outlined in Section VI-B. Outbound connections comprised 87.41% of the traffic, while inbound connections comprised only 12.59%. The large difference is expected as devices inside the network will make more connections to external servers than external clients will connect to devices inside the hospital. We observe that most (over 85%) of both inbound and outbound connections used modern protocols (TLS 1.2 or higher); however, this did not correlate with the overall quality of cipher suites. In particular, we found that the nearly all inbound connections used insecure cipher suites, while the majority of outbound connections were secure by this meausre. Digging deeper, we noticed that SHA1 is used much more frequently for inbound connections than for outbound. Since the trend towards deprecating SHA1 is relatively recent, this discrepancy may indicate lag in patching medical supporting devices. However, we cannot say for certain without actively scanning them to determine their configuration and cipher suite preferences. (It is conceivable that the connecting clients do not support SHA2.)

To further analyze each medical supporting device and to gain a network understanding into how they communicate with other external devices, we investigated the rate at which they establish secure connections throughout our collection period. We separated devices by IP address and computed the ratio of secure connections over total connections established. While this gives us the individual performance for each device, we wanted to see how they compare to each other and how they affect the hospital network as a whole. We compute the average rate of secure connections as

$$R_{\text{avg}} := \frac{\sum_{i \in IPs} S_i / T_i}{|IPs|} \tag{1}$$

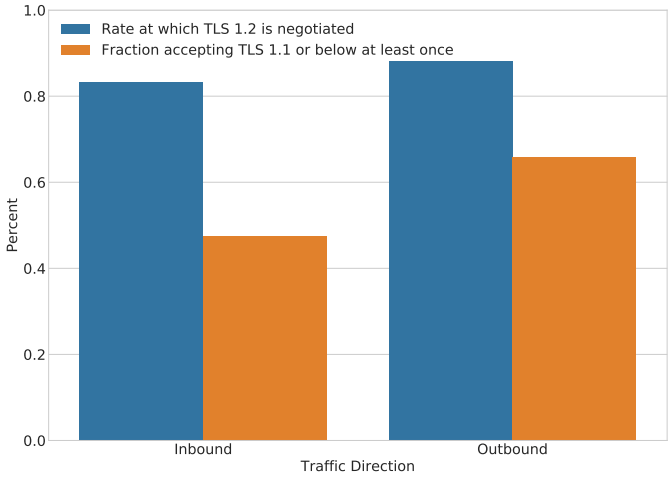

Fig. 7: Average secure connection ratio per endpoint (blue) versus the fraction of devices (orange) for which we observed negotiation of an obsolete protocol (SSL 3 or TLS 1.1 or below).

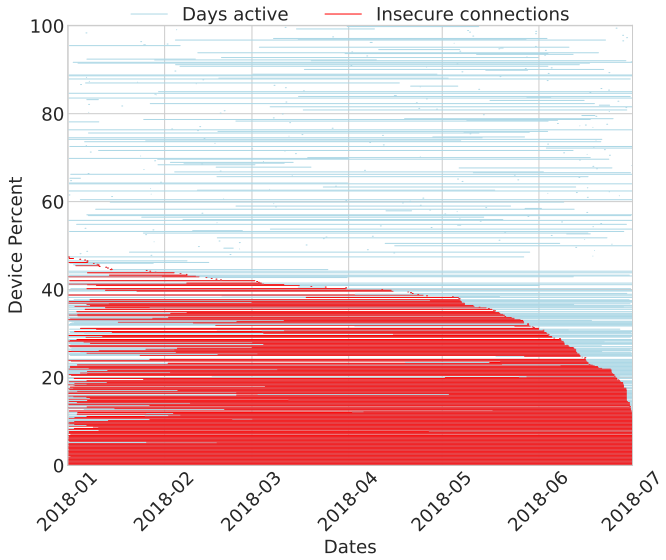

Fig. 8: While the majority of the medical supporting device in the inbound traffic only negotiated a modern TLS version (1.2 or above), 47.56% of those devices also negotiated an obsolete version (1.1 or below). In some cases, this trend did not change during the entire collection period.

where *IPs* denotes the set of observed IPs and for each $i \in IPs$, T_i is the total number of established connections involving i and S_i is the number of that were secure. We computed this metric for both inbound and outbound connections. In Figure 7, we show that inbound and outbound traffic do not significantly differ in terms of average secure connections made by each device (83.38% for outbound and 88.20% for inbound traffic). We can additionally infer a partial configuration of medical supporting devices by analyzing their connections individually.

Since establishing a connection requires both the client and the server to agree on a protocol and cipher suite, established connections can tell us what the medical supporting device is able to accept. As such, we can infer that a device supports an obsolete TLS/SSL version if we find connections in which such a protocol was negotiated; if no such connection was made, then it is likely that one or both of the endpoints only supports modern protocols. (Again, we cannot say for sure without an active scan.) Figure 7 shows the fraction of endpoints to which we observed at least one connection established using an obsolete protocol (TLS 1.1 or below). In the case of outbound traffic, the existence of such connection tells us that the medical supporting device is communicating with a server that has not been updated to deny insecure protocols and that the device itself is offering the deprecated version.

We found that 65.87% of medical supporting devices established an *outbound* connection with an obsolete protocol at least once. On the other hand, we found that 47.56% of these devices made an *inbound* connection with an obsolete protocol at least once. Since the devices in the inbound traffic are operated by the hospital and we have access to all the traffic of established connections, we can further analyze each server individually to infer any changes or updates made during our collection period. In Figure 8, we show the first and last time an inbound connection was made to each medical supporting device[8] as shown by the blue lines. While the majority of medical supporting devices appear to have dropped support of obsolete TLS/SSL protocol versions, many other devices have not. In some cases, connections were exclusively established using TLS 1.0.

Since our dataset only contains established connections, and not all TLS/SSL handshake attempts, the apparent decline in the use of obsolete versions can have different explanations. First, the fact that we do not see devices making any obsolete connections after a certain time period might suggest that the medical supporting device was patched to drop support for older versions. Second, the lack of established connections could also mean that clients have no longer requested handshakes with outdated protocols. In either case, insecure connections were no longer established.

D. Certificate usage

Since many of the medical supporting devices may have access to patient information or EHRs, properly authenticating the server that they are connecting to is crucial. We collected information about the certificate presented by the server in each fresh session. Of these sessions, 9.03% were established while having reported to our monitor with issues relating to unknown issuer, self-signed, or expired certificates. To understand how these errors came about, and what they mean for the security of the sessions, it was necessary to study these certificates in detail. The following analysis accounts for 98.3% of the traffic in this category, covering $4,672$ distinct certificates.

a) Unable to get local issuer: The majority (56.37%) of the non-resumed sessions with an issue was due to the network monitor not knowing the issuer of the certificate. Looking closer, 76.33% of these connections appeared to have certificates that were issued by reputable organizations (e.g., Apple,

[8]We intentionally removed the actual number of devices and replaced it with a percentage to avoid disclosing any information about the hospital.

Microsoft, Samsung, IBM). We note that our network monitor contained the certificate root store provided by Mozilla, the same root of trust for Firefox; other clients, such as Safari or Edge, may have a different set of root certificates; this likely accounts for this large number of certificates with unknown issuers. However, we found seven certificates, which account for 11.26% of the traffic for this category, that have no issuer at all. Coincidently, these certificates have subjects belonging to cloud-based medical companies. Because we do not know who issued these certificates, we cannot definitively say that these certificates are being properly validated. One possibility is clients connecting to these servers are implementing a custom validation logic that validates the certificate chain, beginning with a public key associated with the issuer. This would be bad cryptographic "hygiene", but is theoretically secure.

b) Self-signed Certificates: A self-signed certificate has the same subject and issuer. From the established connections that reported issues, 43.1% were caused by either a self-signed certificate presented to the client or a self-signed certificate found in the validation chain. The acceptance of a self-signed certificate essentially bypasses the security goal of a public key infrastructure because the client would now place the trust anchor on any server that presented the self-signed certificate, rather than the CA that properly asserts the server's identity. In reality, many enterprises make use of self-signed certificates for their internal networks as they are free to make and they can place the trust anchor on themselves. While self-signed certificates can be considered an attack vector for medical supporting devices, further investigation needs to go into who the issuers of these types of certificates are (e.g., verifying that the issuer of the certificate is part of the trusted root stores in the medical supporting device).

c) Expired Certificates: Finally, 0.51% of non-resumed connections used an expired certificate. These connections have a total of 721 distinct certificates that overwhelmingly (96.26% of the certificates) appear to be for non-medical services (e.g., advertising services). While this shows that the connections established with these certificates are minimal, medical supporting devices should have properly closed a session when presented with a certificate that is no longer valid. This issue does not necessarily imply that the session is vulnerable to an attack; however, it does mean the security of the connection is dependent on the client and server implementation. For example, accepting an expired certificate could be dangerous, but a certificate being expired does not mean the corresponding secret key has necessarily been compromised.

E. Certificate Transparency

We were able to collect a total of 350,580 distinct certificates passing through the hospital network.[9] To see how many of those are widely seen by the Internet, we compared our set of certificates to those found in the Certificate Transparency (CT) logs.[10] CT is intended to prevent the numerous pitfalls discussed in the previous section. By comparing the certificates in the hospital network to those present in CT logs we can get a sense of the overall quality of certificates seen within the network. We found that 84.41% of the certificates seen were available in at least one CT log (no collision found with any self-signed certificates found in our dataset). We expect this number to be high because medical supporting devices connect with many major services across the Internet.

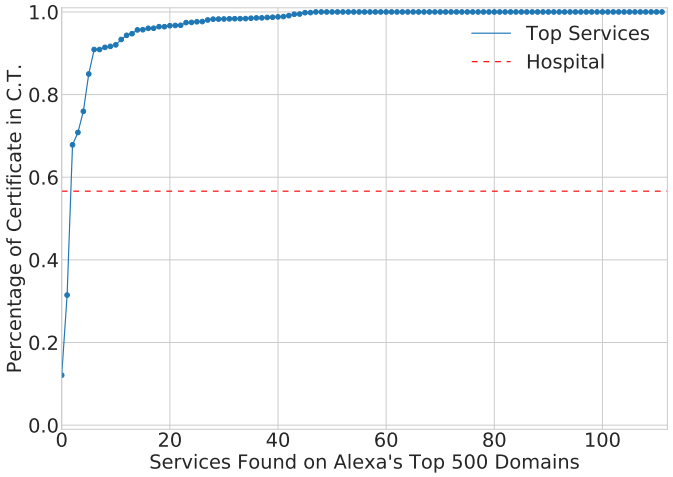

Fig. 9: Certificates belonging to a hospital are documented less frequently in Certificate Transparency logs than those from services found on Alexa's top 500.

Additionally, our traffic aggregation point is in a position to collect certificates that have been seen both internally to the hospital and externally to the rest of the Internet. This unique perspective allows us to perform a measurement of the transparency rate of certificates owned by the hospital versus those owned by popular services. To do the comparison, we collected all the unique services (e.g., `google.com` and `google.fr` would be considered as one service) found in Alexa's top 500 domains and looked for matching strings of those services inside the subject field of the certificate. Since this process was done using regular expressions, we removed any service that had a name of less than five characters to prevent false positives. We looked for services rather than domains to prevent inaccurate results. Certificates may contain abbreviations or other similar discontinuities that may cause a misclassification when looking for domains only. For example, if we search for `netflix.com` we would not account for `*.1.nflxso.net`, which is a certificate for a content distribution domain used by the Netflix service. For each service, we checked how many of the certificates seen by the hospital were available in the CT logs. This process gives us the transparency rate of our network and the top Internet services. Since only a subset of the certificates of each service pass through our hospital dataset, to avoid bias from small samples in this study, we also removed any service that had fewer than 10 distinct certificates as a small sample size can have drastic changes in the transparency metric.

Figure 9 shows the transparency rate of each service along with the transparency rate of the hospital. We note that the CT logs only account for 56.6% of all certificates belonging to a domain (or sub-domain) of the hospital. In comparison, the services found in Alexa's top 500 domains show over 90%

[9]Note that not all of these certificates were used to establish a connection to medical supporting devices.

[10]The logs in our data set included those that were compliant with Chrome's policy [1] (e.g., Digicert, Comodo, Cloudfare, Google, Venafi).

transparency for all but 5 services. Many factors may play into this observation, such as network administrators not using CT or purposefully neglecting CT to avoid revealing network details. Adversaries could potentially use CT logs to help map a hospital's network for an attack. However, The exact reason for the low percentage is not known to us.

F. TLS/SSL Summary

While communication channel setup for medical supporting devices is not perfect, they appear to use secure standards more frequently than the rest of the Internet. As part of our D-HAI analysis, we saw that medical supporting devices use secure protocols (i.e., TLS 1.2) at a higher rate than our Internet baseline with an 11% difference. Regarding cipher suite quality, the baseline for the Internet had a similar breakdown to the data observed from medical supporting devices. The major difference lay in the hashing algorithms that were used, and was an area in which the hospital operated with more security. The hospital network is observed to use the weak SHA1 hashing algorithm roughly 10% less frequently than our baseline data. Correspondingly, the use of the secure hashing algorithms is higher than the baseline, with the broken MD5 algorithm seeing low use in either case. While ideally this percentage would be less than observed, the hospital is still performing significantly better than online connections in general. Furthermore, when examining the inbound traffic we discovered that 47% of the servers administered by the hospital were negotiated an obsolete protocol at least once. While some appeared to have been updated early on to not support these protocols, other servers have continued accepting old SSL/TLS versions throughout the whole collection period.

Finally, 84.41% of all certificates seen throughout our collection period were found in CT. Of the certificates used to establish non-resumed TLS/SSL connections with medical supporting devices, only a few (3,464) appeared to have an issue regarding unknown CA, self-signed, or expired certificates. Though the problem is minimal, certificate usage by medical supporting device needs improvement.

VII. Discussion

A. Hospital Ecosystem

While our analysis sees mostly benign traffic, there is a small portion of traffic that appears malicious or uses poor security protocols and encryption. Given the critical role of hospitals and the high value of data they keep, we believe that this small portion of bad behavior is still concerning. Accordingly, all findings have been turned over to the hospital for further investigation.

While our research covers major Internet protocols used in the hospital network, the security of a hospital's ecosystem is multidimensional. As shown in Figure 10, the majority of hospital related data breaches reported to HIPAA[11] in the United States during our collection period were caused by the unauthorized access of data. In the context of our work, these include stolen credentials, misplaced items, improper disposal, thefts, or employees accessing data which is not directly related to their work. From the viewpoint of this study, detecting unauthorized accesses similar to these is highly unlikely. These kinds of unauthorized accesses rarely leave a network trace, making them opaque to our study. Addressing such issues requires research into proper access control techniques that range from the classification and compartmentation of data to requiring multi-factor authentications. While hospitals already make use of some of these techniques [34], further research into access control security in the context of a hospital ecosystem is required.

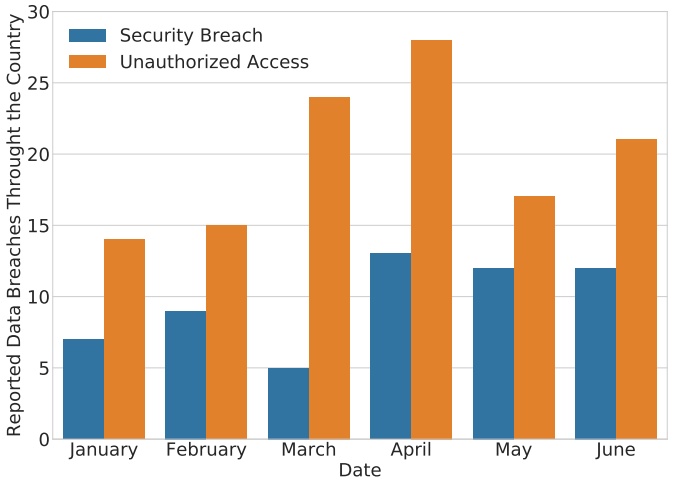

Fig. 10: Data breaches caused by unauthorized access (rather than security breach) are more commonly reported to HIPAA during our collection period.

B. Future Work

a) Medical Devices: In Section IV, we mentioned that our work does not focus on medical devices as they are protected from other devices inside the network and the Internet by using a combination of firewalls and VPNs. While such protection leaves them mostly invisible from a network perspective (and our analysis), they still make up part of a hospital ecosystem and can be vulnerable from attacks near their physical location [42]. Further analysis into the local area network connectivity of these devices would greatly benefit the community's knowledge of a hospital's ecosystem. Such large scale work can be used to create fingerprinting profiles of various medical devices based on the network calls made. These profiles can then in turn be used to mitigate lateral expansion of attacks from within the hospital. Similar to this study, safe design and ethical decision making need to be a top priority as such research would be done with devices that can have a direct effect on the patient care.

b) Generalizing Hospital Ecosystems: Our case study focused on the traffic from one hospital for six months. While our analysis and techniques can be used to study other hospital networks, generalizing these results to all hospitals may constitute a threat to population validity. To generalize trends to other hospitals, a large scale multi hospital analysis needs to be done. Doing such work would require a large collaborative

[11]This data was collected through monthly reports found in HIPAA Journals [3]. While the reports separate data breaches into various categories, we combined the data in such a way that represents what we could see from a network perspective.

effort between researchers and the different administrations from each hospital in order to ensure the safety of the patients and the data. Our goal of the case study and Section III was to ease the process of obtaining data from various hospitals and make a larger scale future study less challenging to perform.

c) Understanding Non-technical Issues: As mentioned in the previous subsection, a hospital's security is multi-dimensional. Understanding the root causes of data breaches from a non-technical aspect is important in order to develop defensive protocols that are suited for healthcare systems. As such, future work on this end would require surveying the hospital employees to see what security protocols they follow. Such survey would consider employees with different roles (e.g., doctors/nurses, HR, researchers, medical device technicians) that have different levels of 1) access to medical data, 2) dependence of network connectivity for work-related material, and 3) IT knowledge to understand the root cause of EHR leakage or cyber-attacks.

VIII. Related Work

Society relies on the constant availability of hospitals for emergency needs. In recent years, the stance of the research community has been more directed towards embedded devices [25], [28], [29], [20] and body area networks (BAN) [37], [42]. Rubin et al. give a comprehensive discussion on the current state of BAN and embeddable medical devices. In their, work Rubin et al. address the emerging threats and challenges that exist with these types of devices, such as remote attackers. Additionally, companies have also spent time in reverse engineering currently infected medical devices but much of the approach focuses on specific cases and is not a scalable solution [45]. While this research is extremely useful in security, medical devices are only part of a hospital's whole ecosystem. A full hospital network analysis has yet to be done.

Prior researchers have collected passive DNS data [47], [36], [15], [11] similar to what we have done here. In these prior works, the researchers have used their data to reverse engineer and predict network communications made by malware-infected devices. Passive DNS analysis has shown to be useful for detecting various C&C of botnets [48] as well as prediction of malicious domain names created by domain generated algorithms (DGAs) [12]. By having a global view of the network, passive DNS analysis is able to identify individual infected devices and prevent malware from propagating to other devices in the network. While active DNS probing has been useful in detecting bad-natured traffic [31], we focus on passive DNS data so that we do not disturb traffic from critical devices in the hospital network.

Another analysis method used by researchers involves the collection of SSL certificates [23], [18], [9] (i.e., X.509). While passive DNS analysis tells us about infected devices based on network calls, SSL certificates can give insight into the configuration of the devices. More specifically, previous work has shown that many devices do not adequately validate a certificate [26], accept deprecated (and insecure) cryptographic standards, or suffer from MitM vulnerabilities [43]. As a way to battle mis-issuance of such certificates, publicly verifiable logs [35] have been created in order to make certificate issuance more transparent.

In our paper, we combine many of the network analysis methods mentioned above while focusing on medical supporting devices and the network context in which they reside.

IX. Conclusion

Understanding the threat surface of the modern healthcare system requires a characterization of not only the individual devices within the environment, but a holistic analysis of the entire ecosystem, and must be done in a manner that does not endanger operational networks. We performed the first Digital Healthcare-Associated Infection (D-HAI) analysis of a major, multi-campus healthcare system. Our longitudinal case study examined all Internet-facing traffic over a six-month period from January to June 2018. We find that while the majority of medical devices have minimal exposure to the Internet, *medical supporting devices* that support the hospital environment make millions of connections. We examined over 775 million DNS queries across 17 million domains and discovered low amounts of malicious traffic (0.01%) and small but significant traces of botnet activity. We also observed 325 million SSL and TLS handshakes over the analysis period, discovering that almost 46% of these connections are made with weak or insecure ciphers, largely because of a reliance on SHA1 but also because of outdated cipher suites. More importantly, we found that 47% of the servers negotiated an obsolete version of TLS (1.1 or below) at least once. Our D-HAI analysis of the hospital ecosystem is repeatable, non-invasive, and is designed in accordance to ethical considerations. We find that while much has been done to secure medical devices, more must be done to ensure the protection of this critical environment. Aside from the analytical results presented, our hope is for this paper to aid future researchers in doing ethical research by outlining the possible limitations they might have when working in sensitive environments.

Acknowledgments

The authors would like to thank our shepherds, David Evans and Michelle Mazurek, and our anonymous reviewers for their helpful comments and guidance. We would also like to thank all of the hospital staff that gave us access to the data and made themselves available to answer any questions we had throughout the whole process. The icons found in Figure 1 were created by Joni, Arafat Uddin, Vectors Market, Gan Khoon Lay, Creative Stall, Stonehub, Diego from the Noun Project under the Creative Commons 3.0 License. Finally, this work was supported by the National Science Foundation grant numbers CNS-1562485 and CNS-1540217. Any findings, comments, conclusion found in this work are from the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] "Certificate transparency known logs," 2018, http://www.certificate-transparency.org/known-logs.

[2] "Dshield," https://www.dshield.org/suspicious_domains.html, 2018.

[3] "Hipaa journal," https://www.hipaajournal.com/, 2018.

[4] "Malwaredomains," http://www.malwaredomains.com/, 2018.

[5] "Openphish," https://openphish.com/, 2018.

[6] "Phishtank," https://www.phishtank.com/, 2018.

[7] "Zeus tracker," https://zeustracker.abuse.ch/blocklist.php, 2018.

[8] N. J. AlFardan and K. G. Paterson, "Lucky thirteen: Breaking the TLS and DTLS record protocols," in *2013 IEEE Symposium on Security and Privacy*. IEEE, 2013, pp. 526–540.

[9] J. Amann, O. Gasser, Q. Scheitle, L. Brent, G. Carle, and R. Holz, "Mission accomplished?: Https security after diginotar," in *Proceedings of the 2017 Internet Measurement Conference*. ACM, 2017, pp. 325–340.

[10] Amazon Company, "Alexa Company," 2018. [Online]. Available: https://www.alexa.com/

[11] M. Antonakakis, R. Perdisci, W. Lee, N. Vasiloglou, and D. Dagon, "Detecting malware domains at the upper dns hierarchy." in *USENIX security symposium*, vol. 11, 2011, pp. 1–16.

[12] M. Antonakakis, R. Perdisci, Y. Nadji, N. Vasiloglou, S. Abu-Nimeh, W. Lee, and D. Dagon, "From throw-away traffic to bots: Detecting the rise of dga-based malware." in *USENIX security symposium*, vol. 12, 2012.

[13] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche, "Keccak sponge function family main document," *Submission to NIST (Round 2)*, vol. 3, no. 30, 2009.

[14] K. Bhargavan and G. Leurent, "On the Practical (In-)Security of 64-bit Block Ciphers: Collision Attacks on HTTP over TLS and OpenVPN," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '16. New York, NY, USA: ACM, 2016, pp. 456–467. [Online]. Available: http://doi.acm.org/10.1145/2976749.2978423

[15] L. Bilge, E. Kirda, C. Kruegel, and M. Balduzzi, "EXPOSURE: Finding Malicious Domains Using Passive DNS Analysis."

[16] D. Bleichenbacher, "Chosen ciphertext attacks against protocols based on the rsa encryption standard pkcs #1," in *Proceedings of the 18th Annual International Cryptology Conference on Advances in Cryptology*, ser. CRYPTO '98. London, UK, UK: Springer-Verlag, 1998, pp. 1–12. [Online]. Available: http://dl.acm.org/citation.cfm?id=646763.706320

[17] H. Böck, J. Somorovsky, and C. Young, "Return of bleichenbacher's oracle threat (robot)," Cryptology ePrint Archive, Report 2017/1189, 2017, https://eprint.iacr.org/2017/1189.

[18] C. Brubaker, S. Jana, B. Ray, S. Khurshid, and V. Shmatikov, "Using frankencerts for automated adversarial testing of certificate validation in ssl/tls implementations," in *Security and Privacy (SP), 2014 IEEE Symposium on*. IEEE, 2014, pp. 114–129.

[19] Dan Hubbard, "Cisco Umbrella 1 Million," 2016. [Online]. Available: https://umbrella.cisco.com/blog/2016/12/14/cisco-umbrella-1-million/

[20] T. Denning, A. Borning, B. Friedman, B. T. Gill, T. Kohno, and W. H. Maisel, "Patients, pacemakers, and implantable defibrillators: Human values and security for wireless implantable medical devices," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2010, pp. 917–926.

[21] Z. Durumeric, D. Adrian, A. Mirian, M. Bailey, and J. A. Halderman, "A search engine backed by Internet-wide scanning," in *22nd ACM Conference on Computer and Communications Security*, Oct. 2015.

[22] Z. Durumeric, F. Li, J. Kasten, J. Amann, J. Beekman, M. Payer, N. Weaver, D. Adrian, V. Paxson, M. Bailey, and J. A. Halderman, "The matter of heartbleed," in *Proceedings of the 2014 Conference on Internet Measurement Conference*, ser. IMC '14. New York, NY, USA: ACM, 2014, pp. 475–488. [Online]. Available: http://doi.acm.org/10.1145/2663716.2663755

[23] S. Fahl, M. Harbach, T. Muders, L. Baumgärtner, B. Freisleben, and M. Smith, "Why eve and mallory love android: An analysis of android ssl (in) security," in *Proceedings of the 2012 ACM conference on Computer and communications security*. ACM, 2012, pp. 50–61.

[24] Food and Drug Administration, "Quality and Compliance (Medical Devices)," 2018, https://www.fda.gov/MedicalDevices/DeviceRegulationandGuidance/MedicalDeviceQualityandCompliance/default.htm.

[25] K. Fu and J. Blum, "Controlling for cybersecurity risks of medical device software," *Communications of the ACM*, vol. 56, no. 10, pp. 35–37, 2013.

[26] M. Georgiev, S. Iyengar, S. Jana, R. Anubhai, D. Boneh, and V. Shmatikov, "The most dangerous code in the world: validating ssl certificates in non-browser software," in *Proceedings of the 2012 ACM conference on Computer and communications security*. ACM, 2012, pp. 38–49.

[27] Google Chrome, "Chrome Platform Status TLS 1.3," 2018. [Online]. Available: https://www.chromestatus.com/feature/5712755738804224

[28] D. Halperin, T. S. Heydt-Benjamin, K. Fu, T. Kohno, and W. H. Maisel, "Security and privacy for implantable medical devices," *IEEE pervasive computing*, vol. 7, no. 1, 2008.

[29] D. Halperin, T. S. Heydt-Benjamin, B. Ransford, S. S. Clark, B. Defend, W. Morgan, K. Fu, T. Kohno, and W. H. Maisel, "Pacemakers and implantable cardiac defibrillators: Software radio attacks and zero-power defenses," in *Security and Privacy, 2008. SP 2008. IEEE Symposium on*. IEEE, 2008, pp. 129–142.

[30] Health Insurance Portability and Accountability Act, "Summary of the HIPAA Security Rule)," 2018. [Online]. Available: https://www.hhs.gov/hipaa/for-professionals/security/laws-regulations/index.html

[31] T. Holz, C. Gorecki, K. Rieck, and F. C. Freiling, "Measuring and detecting fast-flux service networks." in *NDSS*, 2008.

[32] A. Kharraz, W. Robertson, D. Balzarotti, L. Bilge, and E. Kirda, "Cutting the gordian knot: A look under the hood of ransomware attacks," in *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*. Springer, 2015, pp. 3–24.

[33] C. Kreibich, C. Kanich, K. Levchenko, B. Enright, G. M. Voelker, V. Paxson, and S. Savage, "On the spam campaign trail." *LEET*, vol. 8, no. 2008, pp. 1–9, 2008.

[34] J. Kwon and M. E. Johnson, "Security practices and regulatory compliance in the healthcare industry," *Journal of the American Medical Informatics Association*, vol. 20, no. 1, pp. 44–51, 2012.

[35] B. Laurie, A. Langley, and E. Kasper, "Certificate transparency," Tech. Rep., 2013.

[36] C. Lever, M. Antonakakis, B. Reaves, P. Traynor, and W. Lee, "The Core of the Matter: Analyzing Malicious Traffic in Cellular Carriers." in *NDSS*, 2013.

[37] M. Li, W. Lou, and K. Ren, "Data security and privacy in wireless body area networks," *IEEE Wireless communications*, vol. 17, no. 1, 2010.

[38] I. Mantin and A. Shamir, "A practical attack on broadcast rc4," in *Fast Software Encryption*, M. Matsui, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 152–164.

[39] P. Mockapetris and K. J. Dunlap, *Development of the domain name system*. ACM, 1988, vol. 18, no. 4.

[40] B. Möller, T. Duong, and K. Kotowicz, "This poodle bites: exploiting the ssl 3.0 fallback," *Security Advisory*, 2014.

[41] V. Paxson, "Bro: a system for detecting network intruders in real-time," *Computer networks*, vol. 31, no. 23-24, pp. 2435–2463, 1999.

[42] M. Rushanan, A. D. Rubin, D. F. Kune, and C. M. Swanson, "Sok: Security and privacy in implantable medical devices and body area networks," in *Security and Privacy (SP), 2014 IEEE Symposium on*. IEEE, 2014, pp. 524–539.

[43] D. Sounthiraraj, J. Sahs, G. Greenwood, Z. Lin, and L. Khan, "Smvhunter: Large scale, automated detection of ssl/tls man-in-the-middle vulnerabilities in android apps," in *In Proceedings of the 21st Annual Network and Distributed System Security Symposium (NDSS'14*. Citeseer, 2014.

[44] M. Stevens, E. Bursztein, P. Karpmann, A. Albertini, and Y. Markov, "The first SHA-1 collision," Technical Report, 2017, https://shattered.io/static/shattered.pdf.

[45] TrapX Research Labs, "MEDJACK.2 Hospitals Under Siege," 2018. [Online]. Available: https://trapx.com/wp-content/uploads/2017/08/AOA_Report_TrapX_MEDJACK.2.pdf

[46] S. Vaudenay, "Security flaws induced by CBC padding — Applications to SSL, IPSEC, WTLS..." in *Advances in Cryptology — EUROCRYPT 2002*. Springer Berlin Heidelberg, 2002, pp. 534–545.

[47] F. Weimer, "Passive dns replication," in *FIRST conference on computer security incident*, 2005, p. 98.

[48] S. Yadav, A. K. K. Reddy, A. Reddy, and S. Ranjan, "Detecting algorithmically generated malicious domain names," in *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*. ACM, 2010, pp. 48–61.

[49] T.-F. Yen, V. Heorhiadi, A. Oprea, M. K. Reiter, and A. Juels, "An epidemiological study of malware encounters in a large enterprise," in *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2014.