It's Not Where You Are, It's Where You Are Registered: IoT Location Impact on MUD

Anat Bremler-Barr
Tel-Aviv University
Israel

David Hay
The Hebrew University
Israel

Bar Meyuhas
Reichman University
Israel

Shoham Danino
Reichman University
Israel

ABSTRACT

We explore the impact of device location on the communication endpoints of IoT devices within the context of Manufacturer Usage Description (MUD), an IETF security framework for IoT devices. Two types of device location are considered: IP-based location, which corresponds to the physical location of the device based on its IP address; and user-defined location, which is chosen during device registration.

Our findings show that IP-based location barely affects the domain set with which IoT devices interact. Conversely, user-defined location drastically changes this set, mainly through region-specific domains that embody location identifiers selected by the user at registration.

We examine these findings' effects on creating MUD file tools and IoT device identification. As MUD files rely on allowlists of domain allowlists, we show that security appliances supporting MUD need to manage a significantly larger number of MUD rules than initially anticipated.

To address this challenge, we leverage EDNS Client Subnet (ECS) extension to differentiate user-defined locations without needing regional domains, consequently reducing the number of Access Control Entries (ACEs) required by security appliances.

ACM Reference Format:
Anat Bremler-Barr, David Hay, Bar Meyuhas, and Shoham Danino. 2023. It's Not Where You Are, It's Where You Are Registered: IoT Location Impact on MUD. In *Applied Networking Research Workshop (ANRW '23), July 22–28, 2023, San Francisco, CA, USA*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3606464.3606472

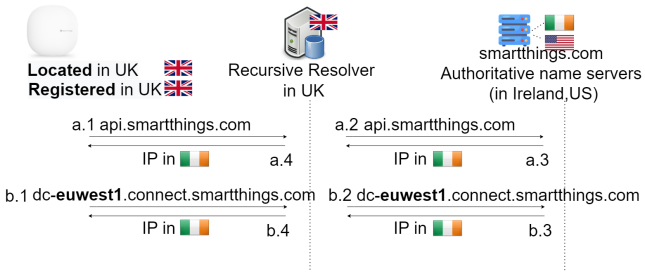
1 INTRODUCTION

The Internet of Things (IoT) technology, in which various physical devices are connected through a computer network, is becoming ubiquitous in almost every sector of our society and day-to-day life. IoT devices are powerful enough to host malicious code but, for economic and technological reasons, they do not have the means to protect themselves from being hacked. The Manufacturer Usage Description (MUD) is an allowlist-based cybersecurity framework recently proposed by the IETF to cope with the huge attack surface and steadily rising number of IoT devices connected to the Internet [1]. The MUD framework leverages the fact that, unlike general-purpose devices, such as computers and smartphones, IoT devices are characterized by the small number of endpoints with which they communicate. Thus, a MUD file specifies the domain names of services the IoT device is allowed to communicate with, along with the legitimate port numbers and protocols. We note that in rare cases when no domain name is given (e.g., for the DNS server itself), the MUD file may specify a fixed IP address instead.

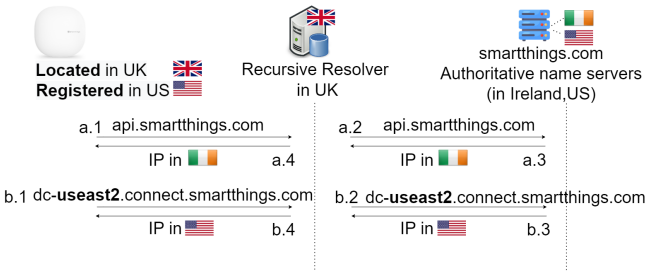
NIST has defined a list of environmental variables that can influence the network behavior of an IoT device [2] (i.e., internet connection, DNS blocking, human interaction), and hence, the domains with which it can communicate. These variations in networking behavior are especially important for MUD, since MUD files are often created based on network traffic trace analysis. However, neither NIST nor previous studies on IoT network behavior consider the *location* of IoT devices. As far as we know, this is the first study to consider the location of an IoT device as a factor that influences its network behavior and therefore its security. The only related work we are aware of deals with the influence of privacy regulations (e.g., GDPR [3], FTC's privacy regulation [4]) on the network behavior of IoT devices in the United Kingdom and the United States [5].

In this paper, we study the impact of the IoT device location on its network characteristics by differentiating between the IP-based location and the user-defined location. The IP-based location is the geo-location that corresponds to the device's current IP address. The user-defined location is the one chosen by the user while registering the device. By measuring each IoT device at up to 10 locations, we show that, surprisingly, the *user-defined location* is the primary factor determining the device's domain set, while the *IP-based location* itself has almost no effect. The unexpected finding that the same device, with the same firmware, behaves differently depending on the user-defined location has significant implications for device profiling, device identification [6–13], security appliances, and any tools that either use or generate MUD files (e.g., [14]). Additionally, the fact that the user-defined location is not visible to network administrators or service providers makes it challenging to monitor and protect IoT devices. MUD files, which are common to all devices using the same firmware, must allow all optional domains, for *every* user-defined location worldwide. This results in larger files that are not only more likely to contain

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. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
ANRW '23, July 22–28, 2023, San Francisco, CA, USA
© 2023 Association for Computing Machinery.
ACM ISBN 979-8-4007-0274-7/23/07…$15.00
https://doi.org/10.1145/3606464.3606472



(a) SmartThings Hub located (IP-based location) in the UK and registered (user-defined location) in the UK.



(b) SmartThings Hub located (IP-based location) in the UK and registered (user-defined location) in the US.

Figure 1: The impact of user-defined locations on domain selection by illustrating the device's regional and global domain interactions under two conditions: (1) registered and located in the UK and (2) registered in the US but located in the UK. Regional domains (*dc-useast2.connect.smartthings.com, dc-euwest1.connect.smartthings.com*) change with alterations in user-defined location, whereas global domains (*api.smarthings.com*) typically remain unchanged across different locations.

errors but also demand extensive maintenance when updates are due. In addition, the more Access Control Entries (ACEs) these files contain, the less explainable they become.

We delve into the underlying reason for the variation in the domain set resulting from the user-defined location. Our analysis groups the domains into two distinct categories. The first classification, termed *regional domains*, incorporates location identifiers within the domain name. The second group, referred to as *global domains*, consists of domain names that do not incorporate any location identifiers. We demonstrate that regional domains change with alterations in user-defined location (as depicted in Figure 1), while global domains typically remain unchanged across different locations. We observe that many manufacturers use several regional domains to distinguish between user-defined locations. We presume this occurs because DNS does not provide a means to convey application-level information, such as the user-defined location, without resorting to a separate domain for each user-defined location. To enable specific features in certain locations, many IoT devices support application-level decisions based on registration, for various reasons such as privacy regulation, legal compliance, or marketing strategies (e.g., MIUI is a Xiaomi Android fork for specific regions [15]).

Thus, in order to simplify MUD-based IoT device security, we propose using a single domain name rather than multiple domain names. Specifically, we suggest a unique use of the ECS field that is not currently implemented in DNS. Our proposal involves utilizing an *ECS per user-defined location* instead of the intended *ECS per IP-based location*. The core concept of our proposal is to transmit the user-defined location from the IoT device in the ECS field. This approach has the potential to streamline MUD-based IoT device security and significantly enhance its efficiency.

2 IOT TRAFFIC ANALYSIS

Our dataset is based on IoT network traffic captured from the router in our lab, as well as log files from Ren et al. [5], which is a representation of home IoT devices [5]. The dataset comprises 31 different IoT devices (e.g., plugs, cameras, bulbs, and so on) that are located in up to 14 countries and use all of their device functionalities. We made the entire dataset publicly available [16].

2.1 Impact of Device Location: Domains

As stated above, for each IoT device, we distinguish between two types of locations:

- The *IP-based location* of a device is the geo-location of the external IP address it uses.
- The *User-defined location* of a device is the location the user of the device chose when registering the device (e.g., through a profile in the IoT application).

In Figure 1a, the IP-based location of the IoT device is in the UK, where it connects through a British ISP. In Figure 1b, the US is the user-defined location of IoT devices whose users are registered in the US, even if they connect through a British ISP.

To study the effects of both location types on the domain name sets, we let $\mathcal{D}(d, i\ell, u\ell)$ be the domain name set of device d, whose IP-based location is $i\ell$ and user-defined location is $u\ell$. As part of our studies, we extracted the DNS domain set from the traffic captures. In our dataset, the set of the domain names for most devices did not change after the first day of observation. We assumed this was due to the fact that certain controlled operations on the IoT devices in laboratory settings were performed, such as periodically powering on/off and other actions. Therefore, even shorter traces contain relatively rare events. Some IoT devices utilize a pool of domain names, maintaining the same domain suffix for a particular service from the pool. We therefore treated these domain names as a single entity, represented by the corresponding regular expression (e.g., `czfe[10-120].front01.iad01.production.nest.com`). For some devices, the set of domain names changed throughout the observation period. This includes, for example, smart TVs that are capable of connecting to any content provider or even browse the Internet. Next, we define the *user-defined similarity* and the *IP-based similarity*:

Definition 1. *For a device d, IP-based location $i\ell$, and two user-defined locations $u\ell$ and $u\ell'$, the* user-defined similarity *denoted $uds(d, i\ell, u\ell, u\ell')$ is*

$$uds(d, i\ell, u\ell, u\ell') = \frac{|\mathcal{D}(d, i\ell, u\ell) \cap \mathcal{D}(d, i\ell, u\ell')|}{|\mathcal{D}(d, i\ell, u\ell) \cup \mathcal{D}(d, \ell, u\ell')|};$$

namely, the *uds* is the Jaccard similarity coefficient of the domain

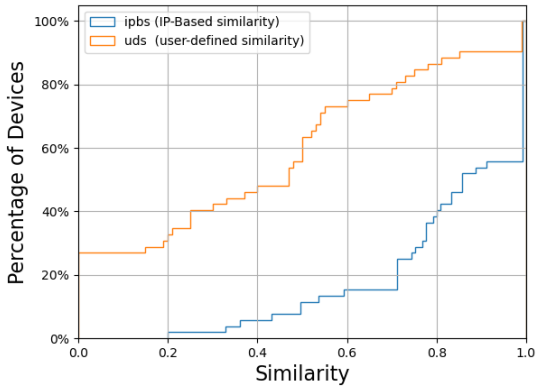


Figure 2: The Cumulative Distribution Function (CDF) of IP-based and user-defined location similarities for 26 devices from the US and UK. User-defined locations impact 90% of devices, markedly more than the 44% influenced by IP-based locations.

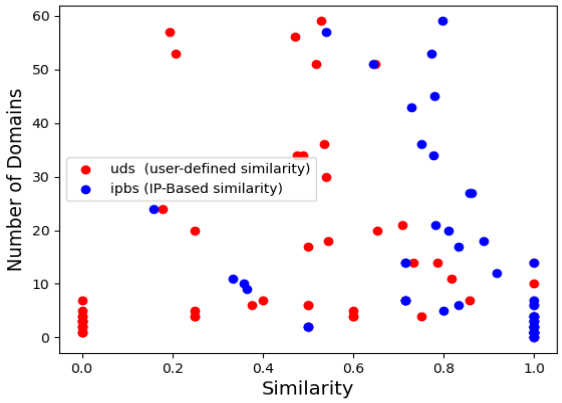


Figure 3: Scatter graph of the user-defined and IP-based similarities of devices in our dataset, by the maximum number of domain names they use, across the two-locations.

name set when we switch the user-defined location from location $u\ell$ to $u\ell'$, while the IP-based location remains in $i\ell$. In the example shown in Figure 1, the value of user-defined similarity $uds(d, UK, UK, US)$ is 0.33. Essentially, when the IoT device is located in the UK, and the registration occurs once in the UK and once in the US, the sets' intersection consists of 1 domain name and the sets' union consists of 3 domain names.

Definition 2. *For a device d, user-defined location $u\ell$, and two IP-based locations $i\ell$ and $i\ell'$, the* IP-based similarity, *denoted $ipbs(d, u\ell, i\ell, i\ell')$, is*

$$ipbs(d, u\ell, i\ell, i\ell') = \frac{|\mathcal{D}(d, i\ell, u\ell) \cap \mathcal{D}(d, i\ell', u\ell)|}{|\mathcal{D}(d, i\ell, u\ell) \cup \mathcal{D}(d, i\ell', u\ell)|};$$

namely, the *ipbs* is the Jaccard similarity coefficient of the domain name set when we switch the IP-based locations from location $i\ell$ to $i\ell'$, while the user-defined location remains in $u\ell$.

Figures 2 and 3 present our comparison between the IP-based and user-defined similarities. The study was done on 26 devices with IP-based and user-defined locations in the US and UK. For each device d we had two values for user-defined similarities: uds(d, US, US, UK) and uds(d, UK, US, UK), as well as two values for IP-based similarities: ipbs(d, US, US, UK) and ipbs(d, UK, US, UK). A change in the IP-based location was done by setting a VPN tunnel from one country to another, and sending the traffic through that tunnel. Surprisingly, our findings showed that the *user-defined location* of the device has a more significant impact than the IP-based location. In our results, 44% of the devices do not experience any difference while changing their IP-based location (with perfect similarity score of one), while 90% of the devices express a difference when changing their user-defined location, their similarity score is less than one. As can be seen in Figure 2, even when there are changes in the domain name set, they are relatively minor, when considering IP-based similarity. Figure 3 shows that when the size of the domain name set is small, the sets tend to be disjoint for user-defined locations or equal for IP-based locations.

We have further used our dataset to understand the differences across user-defined locations. We have found that 80% of the devices used sub-domains for different user-defined locations. An example of this is *dc-**useast2**.connect.smartthings.com*, as presented in Figure 1. Nonetheless, 9% of the devices in the dataset exhibited a difference in the top-level domain (TLD), for example *api.xiaoyi.com.**tw*** of Yi camera.

In Figure 4, we present a heat map of an IoT device located in Israel, for which we took measurements in up to 10 user-defined locations. It can easily be observed that each of the presented devices supports several user-defined *regions*, each with different domain names. In many cases, the domain name sets are disjoint across regions. Moreover, the heat map reveals that there are instances where multiple registration options correspond to the same region, which isn't necessarily determined by geographical distance (e.g., Australia and France map to the same region).

2.2 Regional Domains vs. Global Domains

We classify the domains in our study into two categories: regional domains with location identifiers and global domains without. In our datasets, the majority of the domains (77%) are global domains, whereas 23% of the domains are regional domains, as indicated by the presence of location identifiers in their Fully Qualified Domain Names (FQDNs). To evaluate the effect of location on the IPs of the domains in our dataset, we mapped IP addresses to countries using IPinfo [17].[1] We then analyzed each DNS response and collected the first IP address in the response, which is typically the address a device uses [18].

We have checked if there were anycast IP addresses, which reside in multiple locations. While it is very common to use anycast addresses for open resolvers[2], the domains in our dataset do not resolve to anycast IP. We assume this is the case because anycast is

[1]We removed from the analysis smart devices that allow installation of third-party application and skills such as smart TVs.
[2]Google's Public DNS is 8.8.8.8, which is an anycast address that corresponds to 338 different servers, spread over the globe.

used for stateless services and most IoT services required state-full functionality. Our analysis also showed that for 87% of the regional domains, the geo-location of their IP addresses is consistent with the region specified in the corresponding location identifier; thus, it indicates that the IP-based location of the device had no effect on them. For example, as illustrated in Figure 1, the Smart-things Hub device connects to regional domains that depend on its user-defined location *(dc-**euwest**/dc-**useast**.smartthings.com)*. Moreover, the IP addresses of these regional domains are located in user-defined locations. Of the remaining domains, 13% correspond to a single vendor, TP-Link, which has several domains that are regional: *us.tplink.com* and *uk.tplink.com*. Notably, the IP addresses retrieved for these domains are located in both the US and UK, presumably for load-balancing purposes.

In our global domain analysis, we discovered that 87% resolve to IP addresses either in their corresponding IP-based location (49%) or a single worldwide location (39%). However, for 13% of these domains, despite having multiple global instances[3], the IP addresses we analyzed are not located solely in the IP-based location of the device. This is due to typical geographical load-balancing practices [19], where in some cases, there may be no instance of the domain in the device's location.

Figure 4: Heatmap of user-defined similarities of Yi camera. Measurements were done with IP-based locations in Israel. For example, the value of the top-right cell is uds(Yi Camera, IL, US, Germany).

3 LOCATION IMPACT ON MUD

MUD is an IETF standard [1] that aims to reduce the attack surface for IoT devices by allowing device manufacturers and network administrators to collaborate by describing their appropriate traffic patterns. MUD files consist of Access Control Lists (ACLs), each with several Access Control Entries (ACEs). Each ACE is defined as a 5-tuple consisting of legitimate endpoints, protocol, source port, destination port, and direction. The legitimate endpoints are commonly defined by domain names, although the standard also allows other options such as IP addresses, or MAC addresses for monitoring intra-LAN communications. The MUD ACLs are allow-lists, in which their ACEs define the traffic that is permitted, while all other traffic (i.e., the default ACE) is dropped. Figure 5 shows the MUD files for the device presented in Figure 1, which differ due to user-defined location effects.

The MUD framework itself consists of several components. A *MUD manager*, also known as the *MUD controller*, is responsible for obtaining and processing the MUD information. For each IoT device, the MUD manager first obtains the MUD file from its manufacturer's *MUD server*. The MUD server's address for the IoT device is stored as a *MUD URI* in the device's firmware.

With the MUD file at hand, the MUD manager parses the file and installs the corresponding ACL rules on a network security device, such as a firewall or AAA server. This helps reduce the attack surface on the device, the chance of misconfiguration, and the firewall complexity. Because the number of ACEs is often a bottleneck in many network security scenarios [20, 21], the number of MUD ACEs should be minimized, especially in enterprise settings that manage hundreds of IoT devices. In current MUD architectures, the MUD manager fetches MUD files from the MUD file server. On one hand, in this scenario, the MUD file must be adapted to the user-defined location. But, on the other hand, the user-defined location is determined by the user and can be changed as per the user's decision in the application level. One can suggest a 'trivial' solution, in which the manufacturer maintains a single MUD file. This proposal would need high maintenance due to the many different domains required across locations.

4 ECS-BASED USER-DEFINED LOCATION

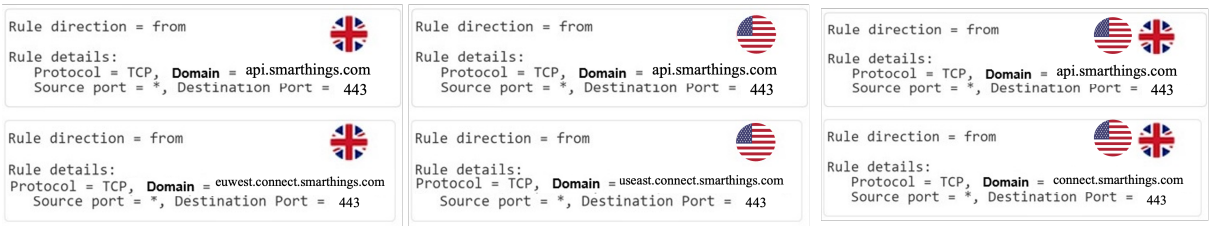
DNS enables the mapping of domain names to server IP addresses based on the geolocation of the DNS request's origin[4]. However, it does not aim to convey application-level information, such as the user-defined location, without using a separate domain per user-defined location.

Hence, we introduce a new distinctive method that leverages the ECS field to achieve a user-defined location mapping of domains to servers. This technique allows us to reduce the number of domains used by IoT devices, as they are not required to use a different regional domain per user-defined location.

Originally, ECS was intended to deal with situations where the client initiating the DNS request is not close to the resolver. Traditionally, when a domain name is mapped to multiple IP addresses, the authoritative server returns the IP address closest to the recursive resolver that issues the DNS query. If the resolver is located within an ISP, its location is a good indicator of the end-user's location. However, nowadays there is an increase in open public DNS services [23]. Moreover, resolvers are not always close to the user [24]. This led to the introduction of the EDNS-Client-Subnet (ECS) solution in RFC 7871 [25]. With ECS, recursive resolvers can convey to authoritative name servers a prefix of the IP address for the client requesting a resolution service domain. This conventional ECS usage is referred to as ECS-based geoIP. We suggest using the ECS to support *user-defined locations*. In our ECS-based user-defined

[3]Several IoT devices, like smart cameras, demand high network performance. To achieve this, many IoT services globally distribute server instances, mapping devices to nearby servers based on their IP-based location

[4]This is usually configured in the authoritative resolver by using different zone files per geo-IP or using different Name Servers [22].



(a) SmartThings Hub UK and US **(b) SmartThings Hub: MUD using ECS**

Figure 5: A contrast of the Access Control Entries (ACEs) required for two user-defined locations (US and UK) of the SmartThings Hub (a) using traditional MUD and (b) a MUD using ECS. This demonstrates a more efficient method of managing user-defined location IoT network behavior.

location, the *IoT device*, and not the resolver, will add the ECS. In this way, the ECS will be based on the user-defined location of the IoT instead of its geo-IP. Using this approach, multiple regional domains can be replaced by a single global domain. In Figure 6, we illustrate how the two regional domains shown in Figure 1 are replaced by a single global domain with an ECS-based user-defined location. Our distinctive way of using ECS raises the question of whether the current standard of ECS, RFC 7871 [25] supports our ECS-based registration without requiring modification. First, our solution requires the IoT device, and not the recursive solver,to add the ECS field. Therefore, the recursive resolver will receive a DNS request with the ECS and forward it without modifying it. We note that the RFC 7871 [25] supports the above since it allows a stub-resolver to add the ECS, and in our case, the IoT device acts as a stub-resolver. In addition, to implement the solution, it is necessary to map the user-defined location to ECS, which is a network address. In the original ECS-based geo-IP scheme, this mapping was straightforward as it was based on the network where the client's IP address was located. However, in our case, the mapping needs to be based on the IoT user-defined location, which is currently classified by country or region. The requirement is that the IoT device and the authoritative DNS server map the same region to the same ECS, i.e., the same network. This is feasible since both the authoritative DNS information and the IoT device are managed by the IoT vendor.

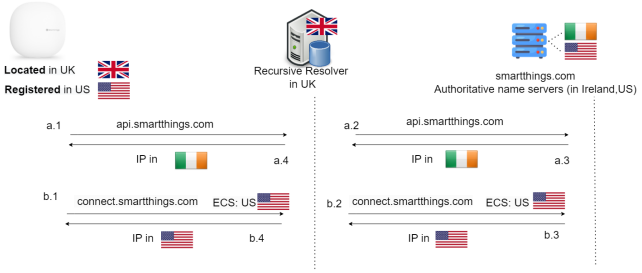


Figure 6: Using the ECS-based user-defined location for the example in Figure 1(b) eliminates the need for regional domains.

Therefore, the task is relatively simple and we recommend two approaches. The first is to naively use a list of networks to map IP addresses to countries. The second is to expand the DNS standard by adding names to the ECS option configuration that indicate either a country or a region. This change will only be implemented in the DNS software and not the DNS protocol. As such, the resolver can translate the country or region name in the ECS to the appropriate IP prefix and send it to the authoritative DNS server.

4.1 ECS Support in the Wild

While the IETF standard supports our ECS-based defined location, we used globally-distributed RIPE Atlas agents [26] to check whether the standard has been adopted by resolvers in the wild. Specifically, we first acquired a domain name and set up our authoritative name server on a virtual machine in Azure. Then, we sent requests from RIPE DNS clients to our domain with ECS and checked to see that we received the requests without modification in the authoritative server. We identified that 56.7% of the 7,737 RIPE Atlas probes, from 159 different countries, used resolvers that forward the ECS. Table 1 summarizes the findings for the top resolvers, including information on the type of resolver (open or ISP) and its market share, if available [27]. Cloudflare is a large open resolver that does not support ECS, due to privacy concerns [28] Based on our results, most DNS providers forward ECS. Moreover, IoT devices can be configured to work with one of the many open resolvers that support ECS.

It is worth noting that the widespread support of ECS is more pertinent to our proposed solution rather than its current usage, which Al-Dalky et al. suggest is limited [29]. We believe that the use of ECS for user-defined location serves is a compelling reason to embrace its adoption.

4.2 MUD Using ECS

When using an ECS-based user-defined location, the manufacturer maintains a single shorter MUD file, as depicted in Figure 5 (b). In our dataset, for the devices that were recorded across 10 distinct countries, we observed that on average the MUD file was reduced to 30% of its initial size.

There are several MUD architectures proposed by NIST [30] in which the MUD manager installs the corresponding ACEs by intercepting DNS requests or by issuing DNS requests [31]. For our ECS-based solution to work, even when the MUD manager issues DNS requests, it must send them with no ECS configured. Then, the authoritative DNS server replies with a list of all the corresponding IPs of all the available zones.

DNS Resolver	Type	ECS Forward	Market Share
Google	Open	Yes	35.94%
Cloudflare	Open	No	13.80%
Liberty Global	ISP	Yes	4.16%
Comcast	ISP	Yes	2.28%
Nevalink	ISP	Yes	1.30%
Claro S.A.	ISP	Yes	1.29%
Korea Telecom	ISP	Yes	1.0%
Telekom Austria	ISP	Yes	0.88%
Quad9	Open	Yes	0.78%
OpenDNS	Open	No	0.03%
Yandex DNS	Open	No	0.09%
Comodo	Open	Yes	Unknown
Verisign	Open	Yes	Unknown
Alternate	Open	Yes	Unknown
AdGuard	Open	Yes	Unknown
UncensoredDNS	Open	Yes	Unknown
UltraRecursive	Open	Yes	Unknown
DNS.Watch	Open	Yes	Unknown
Neustar	Open	Yes	Unknown

Table 1: ECS status of the top resolvers (open or ISP) by market share and 13 major public DNS resolvers

5 CONCLUSION

We demonstrate that some services used by IoT devices depend on the user-defined location. These services are accessed via corresponding regional domains that have servers located in the region specified by the domain name. We suggest this may be due to the limitations of DNS in supporting user-defined decisions. However, using an innovative approach to the ECS field of the DNS, we are able to expand the DNS to support user-defined decisions. Our study also highlights the implications of user-defined location for the MUD framework, as well as for other areas such as IoT identification and anomaly detection algorithms. Finally, our study underscores the importance of capturing the activity of an IoT device in multiple user-defined locations for accurate profiling.

ACKNOWLEDGMENT

This work is partly supported by the Israeli Innovation Authority. The authors would like to thank also Eliot Lear and Michael Richardson for the useful discussions, specifically on the implications on MUD framework.

REFERENCES

[1] E. Lear, R. Droms, and D. Romascanu, "Manufacturer Usage Description Specification," RFC 8520, Mar. 2019. [Online]. Available: https://rfc-editor.org/rfc/rfc8520.txt

[2] NIST National Institute of Standards and Technology, "Methodology for characterizing network behavior of internet of things devices," Apr 2020. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.04012020-draft.pdf

[3] European Parliament and Council of the European Union, "Regulation (EU) 2016/679 of the European Parliament and of the Council." [Online]. Available: https://data.europa.eu/eli/reg/2016/679/oj

[4] F. T. Commission, "Privacy and security," Jun 2023. [Online]. Available: https://www.ftc.gov/business-guidance/privacy-security

[5] J. Ren, D. J. Dubois, D. Choffnes, A. M. Mandalari, R. Kolcun, and H. Haddadi, "Information exposure from consumer iot devices: A multidimensional, network-informed measurement approach," in *IMC Conference*. New York, NY, USA: ACM, 2019, pp. 267–279.

[6] H. Guo and J. Heidemann, "Detecting iot devices in the internet," *IEEE/ACM Transactions on Networking*, vol. 28, no. 5, pp. 2323–2336, 2020.

[7] M. H. Mazhar and Z. Shafiq, "Characterizing smart home iot traffic in the wild," in *2020 IEEE/ACM Fifth International Conference on Internet-of-Things Design and Implementation (IoTDI)*. IEEE, 2020, pp. 203–215.

[8] G. Hu and K. Fukuda, "Toward detecting iot device traffic in transit networks," in *2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*. IEEE, 2020, pp. 525–530.

[9] S. A. Hamad, W. E. Zhang, Q. Z. Sheng, and S. Nepal, "Iot device identification via network-flow based fingerprinting and learning," in *2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*. IEEE, 2019, pp. 103–111.

[10] R. Perdisci, T. Papastergiou, O. Alrawi, and M. Antonakakis, "Iotfinder: Efficient large-scale identification of iot devices via passive dns traffic analysis," in *2020 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2020, pp. 474–489.

[11] S. J. Saidi, A. M. Mandalari, R. Kolcun, H. Haddadi, D. J. Dubois, D. Choffnes, G. Smaragdakis, and A. Feldmann, "A haystack full of needles: Scalable detection of IoT devices in the wild," in *IMC '20, 20th ACM Internet Measurement Conference*. Virtual Event, USA: ACM, 2020, pp. 87–100.

[12] S. J. Saidi, A. M. Mandalari, H. Haddadi, D. J. Dubois, D. R. Choffnes, G. Smaragdakis, and A. Feldmann, "Detecting consumer IoT devices through the lens of an ISP," in *ANRW '21, Applied Networking Research Workshop*. Virtual Event, USA: ACM, 2021, pp. 36–38.

[13] S. J. Saidi, S. Matic, O. Gasser, G. Smaragdakis, and A. Feldmann, "Deep dive into the iot backend ecosystem," in *Proceedings of the 22nd ACM Internet Measurement Conference*, ser. IMC '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 488–503. [Online]. Available: https://doi.org/10.1145/3517745.3561431

[14] A. Hamza, D. Ranathunga, H. Gharakheili, M. Roughan, and V. Sivaraman, "Clear as mud: Generating, validating and applying iot behavioral profiles," in *Workshop on IoT Security and Privacy*. USA: Association for Computing, 2018, pp. 8–14.

[15] Wikipedia contributors, "Miui — Wikipedia, the free encyclopedia," 2022, [Online; accessed 21-May-2022]. [Online]. Available: https://en.wikipedia.org/w/index.php?title=MIUI&oldid=1086751908

[16] B. Meyuhas, A. Bremler-Barr, and R. Shister, "Mud files dataset in different locations." 2021. [Online]. Available: https://github.com/barmey/IoT_mud_files_locations

[17] IPinfo, "Ip geolocation api: Free version with up to 100k lookups and paid geoip extended version." [Online]. Available: https://ipinfo.io/products/ip-geolocation-api

[18] "Domain names - implementation and specification," RFC 1035, Nov. 1987. [Online]. Available: https://www.rfc-editor.org/info/rfc1035

[19] T. P. Brisco, "DNS Support for Load Balancing," RFC 1794, Apr. 1995. [Online]. Available: https://www.rfc-editor.org/info/rfc1794

[20] A. Wool, "A quantitative study of firewall configuration errors," *Computer*, vol. 37, no. 6, pp. 62–67, 2004.

[21] M. Yoon, S. Chen, and Z. Zhang, "Reducing the size of rule set in a firewall," in *2007 IEEE International Conference on Communications*. IEEE, 2007, pp. 1274–1279.

[22] BIND, "Configurations and zone files," January 2023. [Online]. Available: https://bind9.readthedocs.io/en/v9_18_6/chapter3.html

[23] Google, "google open dns ip addresses," 2022, [Online; accessed 20-April-2022]. [Online]. Available: https://developers.google.com/speed/public-dns/faq#locations_of_ip_address_ranges_google_public_dns_uses_to_send_queries

[24] W. B. De Vries, R. van Rijswijk-Deij, P.-T. De Boer, and A. Pras, "Passive observations of a large dns service: 2.5 years in the life of google," *IEEE transactions on network and service management*, vol. 17, no. 1, pp. 190–200, 2019.

[25] C. Contavalli, W. van der Gaast, D. C. Lawrence, and W. A. Kumari, "Client Subnet in DNS Queries," RFC 7871, May 2016. [Online]. Available: https://www.rfc-editor.org/info/rfc7871

[26] RIPE NCC., "Ripe atlas," 2022, [Online; accessed 20-April-2022]. [Online]. Available: https://atlas.ripe.net

[27] R. Radu and M. Hausding, "Consolidation in the dns resolver market–how much, how fast, how dangerous?" *Journal of Cyber Policy*, vol. 5, no. 1, pp. 46–64, 2020.

[28] "Cloudflare support ecs - cloudflare community forum." [Online]. Available: https://community.cloudflare.com/t/1-1-1-1-supports-ecs/17424/15

[29] R. Al-Dalky, M. Rabinovich, and K. Schomp, "A look at the ecs behavior of dns resolvers," in *Proceedings of the Internet Measurement Conference*. New York, NY, USA: Association for Computing Machinery, 2019, pp. 116–129.

[30] NIST, "Securing small-business and home internet of things (iot) devices," May 2021. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.1800-15.pdf

[31] M. Richardson and W. Pan, "Operational Considerations for use of DNS in IoT devices," Internet Engineering Task Force, Internet-Draft draft-ietf-opsawg-mud-iot-dns-considerations-05, Oct. 2022, work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/draft-ietf-opsawg-mud-iot-dns-considerations/05/