Examining DES-based Cipher Suite Support within the TLS Ecosystem

Vanessa Frost
University of Florida
vfrost@ufl.edu

Dave (Jing) Tian
University of Florida
daveti@ufl.edu

Christie Ruales
University of Florida
cruales@ufl.edu

Vijay Prakash
University of Florida
vijay.prakash@ufl.edu

Patrick Traynor
University of Florida
traynor@ufl.edu

Kevin R. B. Butler
University of Florida
butler@ufl.edu

ABSTRACT

In July 2018, over a decade after the DES encryption algorithm was retired, 3DES was also officially deprecated. While previous work suggests a successful deprecation of DES, with fewer than 1% of observed SSL/TLS handshakes using some form of DES up until 2018, such work tends to be limited in scope and does not necessarily capture the true persistence of DES across the entire TLS ecosystem. In this paper, we actively investigate online support for DES and DES-derivative ciphers by querying IP addresses responsive to port 443 connection attempts. To achieve this, we design and implement our own Internet scanning tool built upon ZMap and attempt to negotiate handshakes exclusively using DES ciphers. In total, we have scanned over 31 million unique IP addresses and found that nearly half of them can still successfully establish an HTTPS connection using at least one DES cipher. Moreover, we also find that many servers still support DES40 (which can be broken in seconds) and anon ciphers (which offer no certificate verification and are vulnerable to man-in-the-middle attacks). Our investigation demonstrates the biases and misunderstandings in previous weak cipher studies within the TLS ecosystem, and discloses the severity of this problem by targeting DES-based cipher suites.

CCS CONCEPTS

• **Security and privacy → Web protocol security**; *Network security*.

KEYWORDS

DES; TLS; measurement

ACM Reference Format:
Vanessa Frost, Dave (Jing) Tian, Christie Ruales, Vijay Prakash, Patrick Traynor, and Kevin R. B. Butler. 2019. Examining DES-based Cipher Suite Support within the TLS Ecosystem. In *ACM Asia Conference on Computer and Communications Security (AsiaCCS '19), July 9–12, 2019, Auckland, New Zealand*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3321705.3329858

AsiaCCS '19, July 9–12, 2019, Auckland, New Zealand

ACM ISBN 978-1-4503-6752-3/19/07…$15.00
https://doi.org/10.1145/3321705.3329858

1 INTRODUCTION

Securely communicating on the Internet has become an area of intense focus. The past few years have seen broad new initiatives to ensure safer computing, including the increasing use of TLS across websites - spurred in part by the Electronic Foundation Frontier's "HTTPS Everywhere" initiative [20] and the decisions of browsers such as Google Chrome to highlight when connections are being made without TLS [34].

The use of cryptography to secure communication and messages is not new. The Data Encryption Standard (DES) was introduced by the US National Bureau of Standards (now NIST) in 1977. DES protected information for decades, but was publicly broken for the first time in 1997 [13] and deprecated by NIST in 2004 [30]. To replace DES, which has seen widespread deployment as well as dedicated hardware for using it, NIST recommended the use of the Triple-DES (3DES) cipher [30]; however, 3DES has in turn faced attacks of increasing strength, and was itself officially deprecated by NIST in 2018 [31].

Recent work surveying the state of TLS usage on the Internet [27] has seemingly pointed to DES and its variants (including 3DES) being used in a minuscule number of connections - less than 0.3% in 2018. However, what analyses such as these do not take into account is that when they make connections to servers, they connect over the *preferred* cipher offered by the client. The server will attempt to support the most secure ciphersuite offered by the client. However, these studies do not take into account the universe of ciphersuites that can be *supported* by these servers. The implications are that a server that may otherwise support modern ciphersuites may also allow connection over insecure ciphersuites that incorporate DES.

In this work, we go beyond past work by performing an in-depth examination of how many servers across the Internet still support the use of DES and its variants. Our results are surprising: across the 31,619,709 IP addresses we examined that represent servers responding to TLS handshakes on port 443, we find that 40.5% of them support one or more of the 36 ciphersuite families involving a variant of DES or 3DES, with evidence of even the entirely insecure DES40 cipher being supported as a legitimate means of encrypted communication.

We thus make the following contributions:

- **Active IPv4 Measurement.** We query each IPv4 address with explicitly defined DES ciphers in order to determine existing support for these deprecated ciphers, supplementing studies that contribute analysis on passive data to determine *use* of such ciphers.

- **Analysis of Hosts.** We examine the host names belonging to IP addresses which accept DES ciphers in order to reconcile multiple unique IPs to a single shared organization when possible (e.g., google.com and google.co.uk).
- **Geographic Prevalence.** Lastly, we give a global representation of DES support by city and country.

The remainder of this paper is organized as follows: Section 2 introduces DES-based ciphers, TLS, and TLS scanning; Section 3 provides the design and implementations our TLS scanning tool; Section 4 covers data analysis based on DES-based cipher usage, host names, and geographic information; Section 5 speculates our findings with national policies and discusses the limitation of our methodology; Section 6 summarizes TLS attacks and scans; and Section 7 concludes.

2 BACKGROUND

Due to the focus on three different DES ciphers, it is necessary to distinguish between the family of DES ciphers and the original 56-bit DES cipher. Thus, for the remainder of this paper, we use "DES56" to refer to the 56-bit DES cipher, and refer to the family of related encryption algorithms as simply "DES".

2.1 The Data Encryption Standard

In 1977, the National Institute of Standards and Technology (NIST) adopted the Data Encryption Standard (DES56) as the standard algorithm with which to encrypt sensitive government information. Developed by IBM, and later refined by the NSA, DES56 is a symmetric-key block cipher with a key length of 56 bits. Due to the short key length, DES56 is vulnerable to brute-force attacks and was publicly cracked by a network of computers during the DES Challenge (DESCHALL) in 1997 [13]. In May of 2005, NIST officially withdrew DES56 from standards and encouraged the use of Triple DES instead [30].

As a variation on the DES cipher, Triple DES (3DES) applies the DES algorithm three times to encrypt data, giving it a maximum key length of 168 bits. Strong theoretical attacks against 3DES prompted its deprecation, and it too was officially withdrawn from standards by NIST in 2018 [31]. A second variation, DES40, has a 40-bit key, making it far more susceptible to brute-force attacks. While a DES56 key can be found in less than 27 hours using dedicated hardware, DES40 can be broken in seconds [12, 21].

Other attacks can be leveraged against DES that are faster than brute force using differential cryptanalysis [10], linear cryptanalysis using known-plaintext [28] and chosen-plaintext attacks [26], and "meet-in-the-middle" attacks against reduced-round versions of DES [15]. More recently, the SWEET32 birthday attacks take advantage of DES's 64-bit block sizes [8] by exploiting the "birthday-bound" of block ciphers. A cipher with a block size of n would have a corresponding birthday-bound of $2^{n/2}$, meaning that as the number of attempted keys grows to $2^{n/2}$, the likelihood of finding a collision between two encrypted blocks is over 50%. Cryptographers advise frequent key changes to combat this vulnerability, but few implementations enforce early key renewal.

Despite the deprecation of DES56 and its related encryption ciphers, millions of servers around the world continue to support its use in network communication to clients.

2.2 TLS

TLS has become the de facto protocol for network communication since its introduction in 1999. Originally written as an upgrade from SSL 3.0, TLS 1.0 quickly became the preferred standard due to its upgraded security principles and compatibility with older protocols [11]. While TLS typically supports the use of newer encryption protocols (such as AES), a server using TLS could still connect and communicate with a client using older encryption ciphers (such as 3DES). TLSv1.0 was since displaced by versions 1.1 and 1.2 for stronger security principles, and those in turn have officially been obsoleted by the recent release of TLS 1.3 [33], but these older versions of TLS are still widespread. Specifically for DES, this means that legacy implementations of TLS can support vulnerable ciphers: DES40 is supported in TLS up to version 1.0, while DES56 is supported up to TLS 1.1 and 3DES is supported up to TLS 1.2. More TLS attacks are summarized by Kotzias et al. [27].

Information about ciphersuite use in TLS can be gleaned by observing the TLS handshake. This consists of two messages: the Client Hello message, whereby the client indicates the set of ciphersuites it is capable of negotiating, and the Server Hello, whereby the server selects its preferred ciphersuite from the client's list. Neither of these messages are encrypted, allowing for their observation.

2.3 Scanning for TLS Servers

There are two primary means by which scanning for TLS-enabled servers can be achieved. In *passive* approaches, information about TLS handshakes is collected and stored for later query. An example of this approach is the ICSI SSL Notary [5], which collects TLS connection metadata from a number of universities and other research networks. An important note to this approach is that connection data is collected, i.e., the actual parameters that a TLS client and server negotiated. In *active* approaches, the observation tool itself initiates TLS handshakes with servers and records the handshake results. An advantage to this approach is that the client can be selective about which ciphersuites are communicated to the server, such that if a stronger cipher (e.g., an AES-based cipher) is not presented by the client, the server may instead negotiate a weaker cipher such as DES56 in its Server Hello message. Such ciphersuite negotiation represents a perfectly valid TLS connection, but as discussed above, leaves the connection vulnerable to attack.

Approaches to scanning the entire Internet such as ZMap [19] can be highly efficient in establishing a server's responsiveness to external scans. ZMap sends a single TCP SYN packet to a target IP address, and will immediately close the connection by sending an RST packet if it receives a response. This means that an individual scan can occur very quickly; however, such an approach will not provide information about protocols supported by the contacted host, such as whether it is a TLS server and what ciphersuites it may support. Censys [17] provides not only scans of the Internet through ZMap but also a list of TLS-enabled servers through the use of ZGrab [1]. While this approach will allow for discovery of TLS-enabled servers by initiating a Client Hello and recording the Server Hello response, it defaults to negotiating with the strongest cipher available to both server and client (typically AES). As a result, Censys data on its own will not reveal support for DES variants

amongst TLS-enabled servers. As we discuss in the next section, a new approach is required to capture this diversity of DES use.

3 DESIGN & IMPLEMENTATION

Collecting data through active methods requires substantially more connections and computational overhead compared to passive approaches, which typically record existing connection data rather than making new connections. There are 36 variants of DES-based ciphersuites that can be negotiated within TLS, and to determine whether they are supported or not requires attempting to negotiate on all 36 of them.

We first downloaded lists of millions of IPv4 addresses responsive on port 443 from Censys via Google's BigQuery, and further split the partitioned lists. We built a multi-threaded Java program to accept the lists as inputs and create child threads to handle a relatively small list of IP addresses to query. Each thread performs a ZGrab2 query for each of 36 DES ciphers on each IP address. After receiving results of the handshake, the program stores both the IP address queried and the results into a JSON file. These files are then loaded into an Apache Spark server for data analysis. Figure 1 provides a high-level overview of our implementation pipeline.

Where ZMap is able to query the entire IPv4 address space in less than 5 minutes, our program was able to attempt SSL/TLS handshakes with about 800,000 IP addresses in roughly 24 hours. This is largely due to the difference in implementation between ZMap and ZGrab2 (we discuss the operation of ZMap in Section 2.3). ZGrab2 will initiate a TLS handshake with a given IP address and attempt to fully establish a session, thus taking significantly longer to generate key pairs and return pertinent server information. Additionally, we configured the ZGrab2 automator to attempt handshakes with each DES cipher individually, meaning that a thread would attempt 36 handshakes for every IP address. Exceptions included various timeout errors (e.g., connection-timeout) where the automator would abort further handshake attempts with the unresponsive IP address after the first unsuccessful scan.

Once we obtained results from our ZGrab2 scans, we selected unique IP addresses that accepted at least one DES cipher in order to perform reverse-DNS queries. Similar to the ZGrab2 automator, we used a threaded Python program to read in lists of IP addresses and query each one once for a hostname. These hostnames, and additional geolocation data gathered from Censys, were then joined on their corresponding IP addresses and added to the total set of raw data for analysis.

4 ANALYSIS

Our collection period spanned a little over five months from 16 November 2018 to 1 May 2019. During that time, we made over 939 million handshake attempts to over 31 million IPv4 addresses, resulting in over 274GB of raw data. In this section, we analyze this data to determine the prevalence of DES cipher support among unique IP addresses and popular domains, investigate general location data of supporting servers, and characterize hosts which most commonly provide DES support.

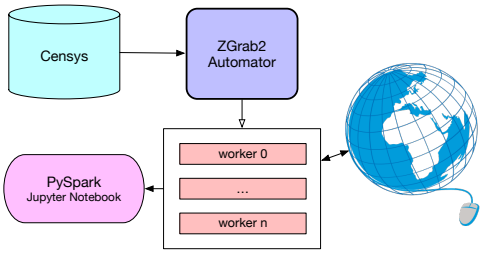


Figure 1: The ZGrab2 automator takes lists of IP addresses from Censys and creates worker threads that query each IP with a DES cipher. It stores the results of the attempted handshake on an internal server, where we analyze it using PySpark.

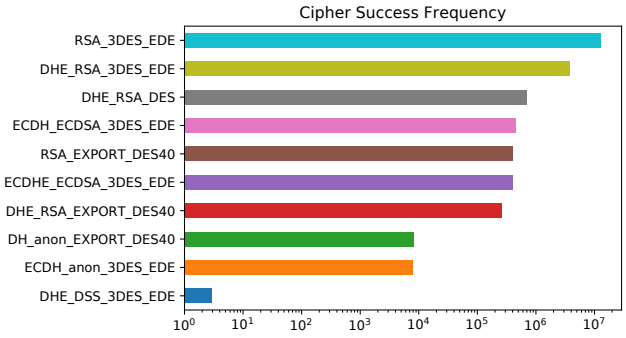


Figure 2: A broad overview of the number of times a cipher was negotiated in a successful TLS handshake.

4.1 DES Cipher Use

We query over 31 million unique IPv4 addresses out of a total of 41 million addresses reported by Censys. These were queried over port 443 using ZGrab2 and we find that 40.5% of them accept at least 1 of the 36 DES ciphers negotiated in an SSL/TLS handshake. In total we had 12,829,045 servers accept a TLS handshake negotiated with a DES cipher out of 31,619,709 queried servers. Of the 36 ciphers presented as handshake encryption algorithms, only 10 were successfully negotiated. An itemized breakdown is given in Figure 2.

Many successful handshakes (673,302) were made with export ciphers. Since encryption was treated as a munitions by the US government, creating weaker forms of encryption was often necessary to allow the export of such algorithms to other countries [14]. Export ciphers are prohibited from using RSA with moduli greater than 512 bits, reducing public key sizes to at most 512 bits and reducing the integrity of the ciphersuite.

Of some concern is the support for DES40 with 673,302 accepted handshakes. As discussed previously, DES40 is trivially easy to break. While it sees relatively little use compared to 3DES, these servers remain vulnerable to eavesdropping and traffic tampering. Despite the slightly longer key, DES56 (with 711,202 accepted handshakes) is subject to similar concerns.

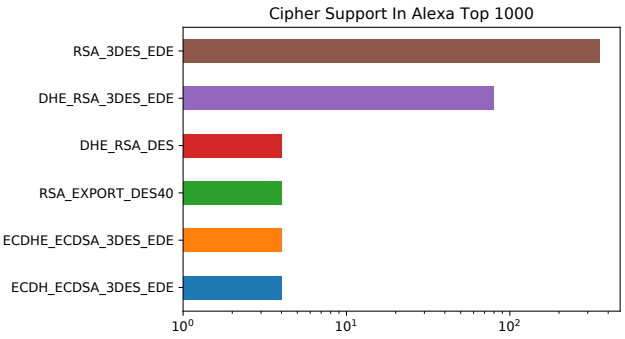


Figure 3: DES support within the Alexa Top 1000.

The DH-anon and ECDH-anon ciphers (accepted in 8,414 and 8,066 handshakes, respectively) are ciphers which do not authenticate a server's certificate, and are thus vulnerable to man-in-the-middle attacks. The support for DH-anon is particularly egregious, as it combines a disregard for server certificate authentication with a thoroughly broken encryption cipher [24].

Fortunately, it is evident that the majority of accepted handshakes are negotiated with 3DES ciphersuites (17,487,797 handshakes). While officially deprecated, 3DES is stronger than DES56 and was encouraged by NIST as a temporary alternative algorithm immediately following the deprecation of DES56 until 2017. It should be expected that 3DES use will decrease over time as more organizations and websites upgrade their infrastructure to support AES instead of DES56 and 3DES, but as the continued use of DES ciphers prove, this is not a guarantee and unlikely to happen for several more years.

Alexa Top 1000. Despite being the most-accessed websites in the online world, arguably communicating and processing sensitive network traffic every day, domains belonging in the Alexa Top 1000[1] continue to support 3DES. However, as seen in Figure 3, the *overall* support for DES ciphers (34%) is lower here than in the IPv4 space in general (40.5%), with only three servers accepting DES40 (youdao.com, 4399.com, and book18.com), and none allowing anonymous ciphers to be used.

4.2 Geolocation Data

In addition to unique IP addresses and domains that accepted DES ciphers, we wished to measure the prevalence of DES support around the world. To that end, we rely on geolocation data provided by Maxmind through Censys.[2] Figures 4, 5, and 6 display the resulting plotted location data. Bubble size is proportional to the number of accepting servers in a location, while color depth represents total number of supporting servers for each cipher in each country. Smaller values are also represented as bubbles to show more granular server locations, in addition to their respective countries' heatmap. (No such filters were needed to render the later choropleth plots.) It is worth noting that the maps representing both DES40 and DES56 are plotted on a smaller scale than 3DES, as 3DES sees far more support on servers worldwide.

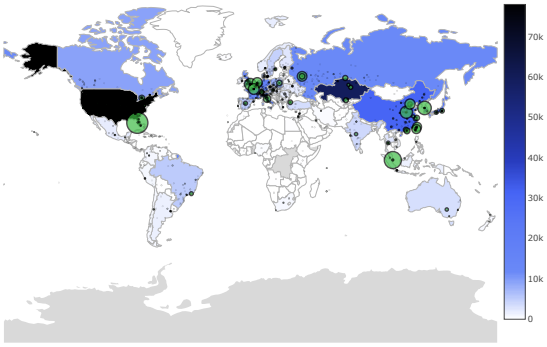


Figure 4: Support of DES40 by servers around the world.

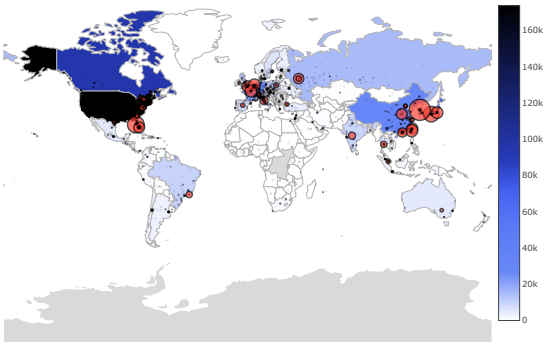


Figure 5: Support of DES56 by servers around the world.

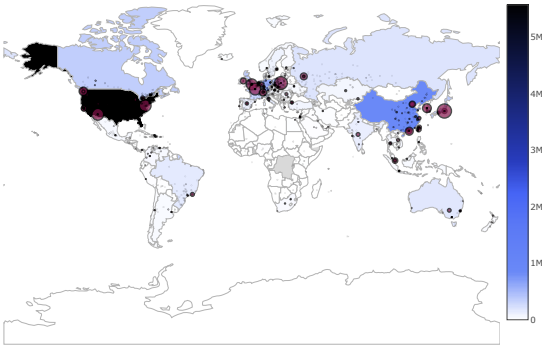
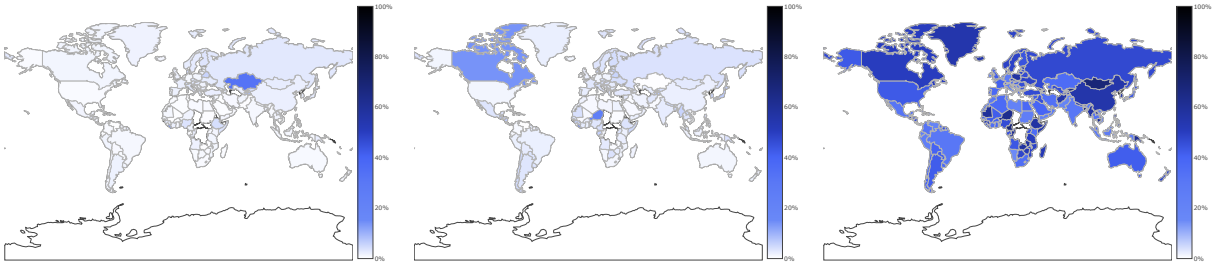


Figure 6: Support of 3DES by servers around the world.

For Figures 4, 5, and 6, many patterns appear to repeat themselves. The majority of IP addresses are concentrated in densely populated areas of the US, Europe, and East Asia. As expected, 3DES sees far more global use than either DES40 or DES56. Due to the omission of areas with fewer than 50 DES-accepting IPs, large areas in South America, Africa, the Middle East, and Australia are bare. Countries

[1]While we focused on the Alexa Top 1000, 78 servers did not respond to an attempted TLS connection and timed out, leaving us with 922 IP addresses to analyze.
[2]Determining the geolocation of IP addresses is often an imprecise art. As such, these locations likely cannot give a fine-grained perspective for DES support, but for the purposes of this paper, it is more than sufficient.



(a) DES40 use. (b) DES56 use. (c) 3DES use.

Figure 7: DES variant usage as percentage of available servers.

typically seem to support DES40 on fewer servers than they support DES56, and support for 3DES is orders of magnitude larger almost without exception.

Few exceptions do exist, however. Contrary to other countries' proportional support of DES, Singapore and Haiti maintain some support for DES56 but seemingly *more* support for DES40. Comparing population bubbles between Figure 4 and 5, we can see a slight decrease going from DES40 to DES56. In Kazakhstan, virtually *no* DES56 ciphers are supported despite being the largest global supporter of DES40. Where other countries have much more support for 3DES in proportion to their DES40 and DES56 values, Kazakhstan has very little 3DES presence in Figure 6.

Despite these abnormalities, it is clear that DES40 and DES56 find limited support relative to 3DES cipher support.

Perhaps more telling is the percent that each country supports specific DES ciphers relative to the number of unique servers residing within their borders. Figure 7a gives a breakdown of DES40 acceptance relative to other countries. In particular, 32.7% of servers we were able to query within Kazakhstan support DES40, reinforcing previous findings in Figure 4. Other countries that accept DES40 at higher rates include Liberia (17.8%), Saint Martin (9.7%), and Lebanon (9.1%). While North Korea reports a DES40 acceptance rate of 100%, we were only able to query a single North Korean server, thus it is excluded in this analysis. For similar reasons, countries reporting fewer than 100 IP addresses were also excluded. All other countries accept DES40 with fewer than 9% of domestic servers. Figure 7b tells a similar story, with the only major DES56 acceptance found in Niger (24.1%), Liberia (19.1%), Canada (13.7%) and Hong Kong (12.4%). Other countries maintain DES56 support with fewer than 10% of their servers.

Lastly, Figure 7c provides an overview of 3DES support per country. Where the weaker two DES ciphers were typically supported by a small fraction of servers in each country, 3DES is highly supported by over 40% of servers in a large majority of nations. Puerto Rico sits just shy of a majority of 3DES supporting servers with 49.7% 3DES support rates, while 68.6% of Mongolian servers and just 19.3% of Pakistani servers support 3DES. 3DES also finds significant support in Europe with 41.9% of servers in the UK and 26.3% of all German servers accepting 3DES handshakes. In North America, Mexico and the US see 3DES support with 31.4% and 42.6% of domestic servers, respectively.

While few countries house servers that offer significant support for DES40 and DES56, a majority of nations support 3DES ciphers with over 40% of domestic servers, leading to a global percentage of 40.5% of IPs accepting TLS handshakes negotiated with a DES encryption algorithm. As seen by our reverse DNS aggregation in Table 1 (listed in the Appendix), many, if not most, of these servers appear to be owned by companies and organizations offering various Internet services. Lastly, popular websites seem to support fewer DES ciphers than the Internet in general, with 34% of IPs accepting a DES cipher.

5 DISCUSSION

It is evident that DES ciphers are still supported by many IPs, ranging from data hosting services to telecommunications providers to news organizations and more. 3DES algorithms are easily the most frequently supported deprecated ciphersuites, likely due to the recent deprecation of 3DES, whereas DES56 has been deprecated for over a decade and sees relatively less support.

Outdated infrastructure has been cited as a potential reason for continued support of deprecated systems, but we do not find strong correlations between the frequency in support (or quality of) DES ciphers and the stage of development of a country.

5.1 Limitations

Though we were able to quickly perform handshakes with servers in a span of six months, the nature of our study could not capture longitudinal data, and Censys data on responsive IPs quickly became outdated. Due to the Internet's rapid and dynamic characterization, many hosts that were online during a Censys scan were not reachable by our queries. Several servers that we attempted to query resulted in connection and I/O timeouts that might affect the validity of our findings. As such, though we observed 40.5% of IP addresses accepting some form of DES cipher, it is possible that percentage has fluctuated some as TLS versions are upgraded or old hosts are reintroduced to the network.

Lastly, while we were able to successfully query over 31 million unique IP addresses on a global scale, our initial ambitions were to explore the entire TLS space of about 41 million servers. As discussed in Section 2, active scanning requires significantly more computational overhead and bandwidth. We needed to make 36 TLS handshakes per IP address to comprehensively analyze use of DES

ciphersuites. Combined with a need to not overload target servers and networks with requests, we necessarily obtain relatively lower throughput than other approaches.

5.2 Future Work

Future work will expand the dataset of sampled IP addresses in order to more accurately gauge global support for DES, especially in countries where there were few reported IP addresses. Future scans will provide a longitudinal perspective on deprecated cipher support, supplementing existing knowledge of DES use over time.

In addition, we will examine the TLS fingerprint of servers that support DES in order to determine the types of machines, configurations, and packages that typically support DES ciphers.

6 RELATED WORK

Holz et al. [23] scanned a large number of popular HTTPS servers for X.509 certificates analysis, revealing a lack of stringent certification. The ICSI SSL notary [5] provides passive scanning on SSL/TLS connections from universities and research institutes in North America. ZMap [19] enables fast Internet scanning by leveraging optimized probing, eliminating per-connection state, and avoiding retransmissions. Durumeric et al. [18] conducted a large-scale study of the Heartbleed vulnerability's impact involving 150K hosts, demonstrating the dynamics of workflow patching. Censys [17] performs active, periodic Internet-wide TLS scans by leveraging ZMap, but does not provide a list of offered ciphersuites by server. Holz et al. [22] scanned TLS use, focusing on application protocols such as IMAP, IRC, XMPP, etc. SSL pulse [32] provides coarse statistics of SSL/TLS quality for Alexa's most popular websites, but do not give details on weak ciphers (except RC4), and only cover 150K machines/websites. Recently, Kotzias et al. [27] showed the evolution of TLS ciphersuite use over the last six years, reporting fewer than 1% of connections use DES-related ciphersuites, but without considering available server ciphersuites. To our knowledge, our work is the first Internet-wide active scan for TLS with a focus on DES-based ciphersuite use, covering over 31 million unique IPv4 addresses. We discuss attacks against TLS in the Appendix.

7 CONCLUSION

We scanned over 31 million IPv4 addresses half a year after 3DES was officially deprecated by building our own active scanning tool and focusing on DES-based ciphersuite support within the TLS ecosystem. We found that nearly half of them can still successfully establish an HTTPS connection using at least one DES cipher. We also note the use of DES40 and anonymous ciphers, which can be broken easily or enable man-in-the-middle attacks. Our further analysis on hostnames and geographic information shows that the use of DES-based ciphersuites are still popular among many ISP-like organizations and the global TLS ecosystem in general.

Acknowledgments

We thank the Censys team for access to their dataset, and Adam Bates for providing a server from which to issue ZGrab2 queries. This work was supported in part by the US National Science Foundation under grant number CNS-1562485. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily represent the views of the National Science Foundation.

REFERENCES

[1] 2018. zgrab: A banner grabber, in go. https://github.com/zmap/zgrab.
[2] David Adrian, Karthikeyan Bhargavan, Zakir Durumeric, et al. 2015. Imperfect forward secrecy: How Diffie-Hellman fails in practice. In *ACM CCS*.
[3] Nadhem J AlFardan, Daniel J Bernstein, Kenneth G Paterson, Bertram Poettering, and Jacob CN Schuldt. 2013. On the Security of RC4 in TLS.. In *USENIX Security Symposium*.
[4] Nadhem J AlFardan and Kenneth G Paterson. 2013. Lucky thirteen: Breaking the TLS and DTLS record protocols. In *IEEE S&P*. IEEE, 526–540.
[5] Bernhard Amann, Matthias Vallentin, Seth Hall, and Robin Sommer. 2012. Extracting certificates from live traffic: A near real-time SSL notary service. *Technical Report TR-12-014* (2012).
[6] Nimrod Aviram, Sebastian Schinzel, Juraj Somorovsky, et al. 2016. DROWN: Breaking TLS with SSLv2. In *USENIX Security Symposium*.
[7] Benjamin Beurdouche, Karthikeyan Bhargavan, Antoine Delignat-Lavaud, et al. 2015. A messy state of the union: Taming the composite state machines of TLS. In *IEEE S&P*.
[8] Karthikeyan Bhargavan and Gaëtan Leurent. 2016. On the practical (in-) security of 64-bit block ciphers: Collision attacks on HTTP over TLS and OpenVPN. In *ACM CCS*.
[9] Karthikeyan Bhargavan and Gaëtan Leurent. 2016. Transcript collision attacks: Breaking authentication in TLS, IKE, and SSH. In *ISOC NDSS*.
[10] Eli Biham and Adi Shamir. 1991. Differential cryptanalysis of DES-like cryptosystems. *Journal of CRYPTOLOGY* 4, 1 (1991), 3–72.
[11] PCI Security Standards Council. 2016. Migrating from SSL and Early TLS. PCI Security Standards.
[12] crack.sh. 2016. The World's Fastest DES Cracker. ToorCon Information Security Conference. https://crack.sh
[13] Matt Curtin and Justin Dolske. 1998. A Brute Force Search of DES Keyspace. USENIX ;login.
[14] T. Dierks and C. Allen. 1999. The TLS Protocol Version 1.0. RFC 2264.
[15] Orr Dunkelman, Gautham Sekar, and Bart Preneel. 2007. Improved meet-in-the-middle attacks on reduced-round DES. In *INDOCRYPT*.
[16] Thai Duong and Juliano Rizzo. 2011. Here come the XOR ninjas. *White paper, Netifera* (2011).
[17] Zakir Durumeric, David Adrian, Ariana Mirian, Michael Bailey, and J Alex Halderman. 2015. A search engine backed by Internet-wide scanning. In *ACM CCS*.
[18] Zakir Durumeric, Frank Li, James Kasten, et al. 2014. The matter of heartbleed. In *ACM IMC*.
[19] Zakir Durumeric, Eric Wustrow, and J. Alex Halderman. 2013. ZMap: Fast Internet-wide Scanning and Its Security Applications. In *USENIX Security Symp.*
[20] EFF. 2014. HTTPS EVERYWHERE. https://www.eff.org/https-everywhere.
[21] EFF. 2016. EFF DES CRACKER MACHINE BRINGS HONESTY TO CRYPTO DEBATE. https://www.eff.org/press/releases/eff-des-cracker-machine-brings-honesty-crypto-debate.
[22] Ralph Holz, Johanna Amann, Olivier Mehani, Matthias Wachs, and Mohamed Ali Kafaar. 2016. TLS in the wild—An Internet-wide analysis of TLS-based protocols for electronic communication. In *NDSS*.
[23] Ralph Holz, Lothar Braun, Nils Kammenhuber, and Georg Carle. 2011. The SSL Landscape: A Thorough Analysis of the x.509 PKI Using Active and Passive Measurements. In *ACM IMC*.
[24] S. Josefsson and Y. Nir. 2018. Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS) Versions 1.2 and Earlier. RFC 8422.
[25] John Kelsey. 2002. Compression and information leakage of plaintext. In *International Workshop on Fast Software Encryption*. Springer, 263–276.
[26] Lars R Knudsen and John Erik Mathiassen. 2000. A chosen-plaintext linear attack on DES. In *International Workshop on Fast Software Encryption*. Springer, 262–272.
[27] Platon Kotzias, Abbas Razaghpanah, Johanna Amann, et al. 2018. Coming of Age: A Longitudinal Study of TLS Deployment. In *ACM IMC*.
[28] Mitsuru Matsui. 1993. Linear cryptanalysis method for DES cipher. In *Workshop on the Theory and Application of of Cryptographic Techniques*. Springer, 386–397.
[29] Bodo Möller, Thai Duong, and Krzysztof Kotowicz. 2014. This POODLE bites: exploiting the SSL 3.0 fallback. *Security Advisory* (2014).
[30] NIST. 1999. Data Encryption Standard (DES). FIPS Publication 46-3.
[31] NIST. 2018. Transitioning the Use of Cryptographic Algorithms and Key Lengths. Draft NIST Special Publication 800-131A.
[32] Qualys. 2019. SSL Pulse. https://www.ssllabs.com/ssl-pulse/.
[33] Eric Rescorla. 2018. The Transport Layer Security (TLS) Protocol Version 1.3. RFC 8446. https://doi.org/10.17487/RFC8446
[34] Emily Schechter. 2018. A secure web is here to stay. Google Security Blog. https://security.googleblog.com/2018/02/a-secure-web-is-here-to-stay.html
[35] Synopsis. 2014. The Heartbleed Bug. http://heartbleed.com/.

Domain	IPs	anon	EXPORT	DES40	3DES
gradwell.com	1011	0	2	2	4
ztomy.com	1510	2	3	3	5
cologlobal.com	1806	0	2	2	4
leaseweb.com	1978	0	2	2	4
xenosite.net	2655	0	2	2	4
hn.kd.ny.adsl	8143	2	3	3	5
static.kpn.net	11035	2	3	3	5
google	150350	2	3	3	5
amazonaws	430482	2	3	3	5

Table 1: A breakdown of ciphers accepted by the investigated domains, discussed in more detail in Section A.2. It should be noted that an accepted cipher may be counted twice, such as EXPORT-DES40 which will be counted as both an EXPORT cipher and a DES40 cipher. In addition to the listed ciphers, each domain also supported one DES56 cipher.

A APPENDIX

A.1 Procedure for Scanning Across Hosts

With the knowledge that running multiple ZGrab2 queries on a server is expensive, we sought to decrease any undue burden our automator might cause. Much of our efforts were inspired by the recommended practices laid out by the authors of ZMap [19].

(1) **Coordinate with local network admins:** While we could not gain permission from our university's IT department to perform a large-scale Internet scan from within the university network, another university was willing to loan a remote server from which we could launch our ZGrab2 queries.
(2) **Signal our scan's benign nature:** In order to inform vigilant network administrators why an unrecognized IP initiated a handshake with a server in their network, we hosted a simple webpage stating our general research intentions. Additionally, we clarify that we are probing IP addresses semi-randomly to prevent confusion and assure readers that their network is not being specifically targeted.
(3) **Provide an easy opt out:** We include an email address on the hosted webpage where network admins could opt out of future scans or ask questions about our research.
(4) **Distribute scans over time and IP space:** The automator read in the lists from Censys in no particular order, sparing IP blocks from being queried all at once or small subnets from being overloaded. Our scan's highest throughput was capped at 800,000 scans a day due to the varying amounts of time it could take a worker to finish iterating through its list of IP addresses. Subsequent days often produced fewer IP scans as a result of workers finishing and not reading new IP lists that were reserved by other workers. While this limited the amount of traffic we could observe in five months, it prevented both exorbitant egress from our hosting server and the possibility of flooding local networks.

A.2 Reverse DNS Lookups

Our primary focus is on the number of unique IP addresses accepting DES ciphers, but it is usually the case that IP address blocks are allocated for specific organizations or regions. Aggregating support for DES ciphers across these organizations can thus potentially provide further insights.

We focus only on IP addresses that accepted a DES cipher to minimize the number of queries made (and thus, reduce the burden placed on the network). These hostnames[3] were saved with their corresponding IP address and joined to our handshake result data for analysis. In addition to aggregating numbers, we also take a closer look at domains that own the largest numbers of IP addresses accepting DES. This involves manually investigating hostnames and visiting domain webpages, and is thus time-intensive. As such, we detail some interesting findings in Table 1, including hostname, the number of IP addresses accepting DES ciphers, and a general overview of the ciphers each domain accepts. It is unclear whether *hn.kd.ny.adsl* is a real domain given that the .adsl TLD cannot be publicly resolved, and in other cases, DES support could be a function of customer configuration as opposed to hosting services. Nonetheless, it is interesting that many of these services include colocation and communication providers (e.g., *gradwell.com*, *xenosite.net*, *cologlobal.com*). Given that webservers within these domains are supporting very weak DES ciphersuites, it would behoove these organizations to ensure that they or their customers discontinue their support.

A.3 Attacks Against TLS

TLS Attacks. BEAST [16] allows data decryption from MitM attackers due to the use of CBC mode and predictable IVs in earlier SSL/TLS versions. CRIME allows HTTPS session hijacking via exploiting vulnerabilities in secret cookies that use data compression [25]. Lucky 13 [4] is a cryptographic timing attack against TLS/DTLS connections that use CBC mode to recover plain texts. Unfortunately, RC4, as a temporary workaround for Lucky 13, is also vulnerable to statistical analysis due to its internal bias [3]. Heartbleed [35] leaks sensitive information from process memory due to an OpenSSL implementation bug. POODLE [29] exploits TLS clients' vulnerability to downgrade to SSL3 and attacks against the CBC mode. FREAK [7] and Logjam [2] downgrade TLS connections to use export-grade cipher suites, which provide weak bit security guarantees. SLOTH [9] demonstrates that the authentication in TLS 1.2 could be broken due to the use of RSA-MD5 signatures. DROWN [6] is a cross-protocol attack against TLS using the obsolete SSLv2 support, breaking the confidentiality of TLS connections. Sweet32 [8] might be the only attack targeting DES/3DES cipher suites with 64-bit block size by launching a birthday-bound attack on CBC mode. While different TLS scans have covered studies of different TLS attacks specifically, our work is DES focused and meaningful, especially considering 3DES's deprecation last year.

[3] Over 3.9 million servers used some form of reverse DNS protection ("no-reverse-dns-set"), returned results that give little to no information about the hostname ("no-data"), or responded with an error. We omit these results from the analysis in this section.