Survey on Alexa's Top 1 Million Websites – DROWN Vulnerability

Avinash Ravi (avinravi@indiana.edu)

Sravya Gudipudi (sgudipud@indiana.edu)

Introduction

DROWN (**D**ecrypting **R**SA with **O**bsolete and **W**eakened e**N**cryption) is a serious vulnerability that affects HTTPS and other services that rely on SSL and TLS. It is a new form of cross platform Bleichenbacher padding oracle attack that allows an attacker to decrypt the intercepted TLS sessions and gain access to passwords, credit cards and other sensitive information. Modern servers use TLS protocol for secure communication with the client. There are other versions of SSL like SSLv2, SSLv3 of which SSLv2 is vulnerable to many attacks. But the servers are configured to accept SSLv2 connections to support legacy clients. DROWN attack shows that just supporting SSLv2 in the servers is a threat to TLS connections. A server is vulnerable to DROWN

- If it allows SSLv2 connections, or
- If it's private key is shared with other servers that allows SSLv2 connections. This is common configuration in servers to use the same certificate for multiple servers like email, web servers.

Background on DROWN Attack:

SSLv2 Handshake – In SSLv2, a client initiates a connection by sending a ClientHello message to the server which includes a list of cipher suites supported by client and a client nonce. The server responds with a ServerHello which includes the list of cipher suites supported by the server, server nonce and public key of the server. The client then responds with a ClientMasterKey message that includes the cipher suite selected and a master key. For export ciphers, 11 bytes of this master key is sent as plain text and 5 bytes is encrypted. For non-export ciphers, the master key is completely encrypted. The server responds with a ServerVerify message and client responds with a Client-Finished message. Sever completes the handshake with a Server-Finished message. One of the vulnerabilities in SSLv2 is that the part of master key is sent as a plain text for export ciphers and can be decrypted by brute force. Here, the premaster secret is 48 bytes irrespective of the chosen cipher suite. The server waits for client finished message to authenticate the client after receiving the key exchange message. This plays a significant role in the attack.

PKCS#1 v1.5 Encryption Padding – SSL/TLS uses PKCS#1 v1.5 which pads the messages with random bytes of string before encryption. A random padding string PS is generated and appended to the message as 00||02||PS||00||message. The server that decrypts this message returns an error if the decrypted message is not PKCS#1 v 1.5 complaint. This leaks some information and Bleichenbacher attack makes used of this vulnerability to decrypt the key in SSLv2.

Bleichenbacher Oracle Padding Attack: The encrypted RSA cipher texts should decrypt to PKCS#1.5 format text. If the format is not valid, the server throws an error message or close the connections. This is a way information is leaked and Bleichenbacher has used this information leak to decrypt the cipher texts. The main idea of the attack is that knowing a cipher and the oracle that decrypts the cipher, the attacker keeps trying to decrypt the cipher. Since the oracle leaks information about the decrypted message, the attacker makes guess about the message to finally decrypt it. This attack has countermeasures in place for all the version of SSL/TLS except for SSLv2.

OpenSSL Cipher Selection Bug: OpenSSL has removed SSLv2 ciphers by default in 2010. But, the client can force an SSLv2 cipher to be used

ignoring the server advertised cipher suite. This makes the DROWN attack very feasible as it requires less computations to decrypt the secret key.

DROWN Attack:

The attack flow of DROWN attack to compromise a TLS session using SSLv2 is:

- The attacker collects many TLS handshake key exchange messages. If the attacker's main target is a server, then the server can be observed for TLS connections from many clients. If the attacker's victim is a client, the attacker has to patiently wait to collect as many TLS connections.
- The attacker then converts the collected TLS key exchange messages into SSLv2 conformant PKCS#1 version 1.5 messages using a technique described by Bardou et al which involves complex mathematical operations.
- After obtaining the SSLv2 conformant messages, Bleichenbacher attack can be performed on the SSLv2 conformant messages to decrypt the TLS handshake key messages. Bleichenbacher attack makes use of the fact that only a part of the premaster secret is encrypted in SSLv2 rather than the whole premaster secret for export ciphers. There is modified version of Bleichenbacher attack that exploits the TLS handshake mechanism where server sends a ServerVerify message before the client has authenticated itself. These two methods are used to guess the premaster secret using brute force.
- Once a premaster secret is obtained, the client write key and server write keys are computed which can be used to decrypt the whole TLS session.

Defense against a DROWN Attack:

Nothing can be done from the client side to prevent DROWN attack. The servers need to be updated to use a different certificate for servers that support SSLv2. Also, OpenSSL and Microsoft have released patches for countermeasures against DROWN attack and the servers need to be updated to include these patches. Network Security Services (NSS) 3.13 version has disabled SSLv2 by default. The server administrators should update NSS to that version to protect against the DROWN attack.

Points to be noted:

- DROWN attack does not reveal the private key of the server, but instead compromises the communication messages of a single session.
- The DROWN attack requires a lot of computing power and will take 2^{40} symmetric encryption operations for a 40-bit export cipher.
- Bleichenbacher oracle padding attack cannot be performed on TLS since there are countermeasures in place for this attack implemented in SSLv3 and higher versions, but there are no counter measures for this attack in SSLv2.

Proposal:

We surveyed on top 1 million websites for identifying the websites or its included servers that are vulnerable to DROWN Attack. Survey focuses on evaluating how many websites are still supporting SSLv2 along with TLS and also how many of them support weak encryptions (export ciphers) for SSL handshake. We also identify the highest SSL/TLS versions currently supported by the web servers. Our goal is to finally categorize the results based on top level domains to identify what kind of top level domain (for example .edu or .com) websites are mostly supporting SSLv2 and combined with weak encryptions. We also surveyed the results based on the rank of the websites to observe the distribution of vulnerable websites based on the rank.

Our project survey can be used to learn how many Web servers are still not been patched against DROWN attack. Before DROWN, even if SSLv2 is enabled along with TLS, as most client are updated to use TLS, it is not found to cause any security hole. But with DROWN attack we

can break TLS session by exploiting vulnerabilities in SSLv2, so it is very important that the vulnerable servers are patched as soon as possible. The Patch for DROWN is released by OpenSSL very recently, so there is high chance that there will be many servers that are vulnerable.

Important point to note here, although we evaluated a huge number of websites, there we will be servers, which are hiding under the firewall that are vulnerable to DROWN. Such web servers can only be tested from server side and hence, cannot be included in our study.

Executed Plan:

March 22 - March 31: Spent time to understand the Drown paper and research on what would be the best approach to scan the million websites.

April 1 – April 8: Spent time to learn bash scripting and other Linux commands like awk, host and openssl implementation.

April 9 – April 15: Spent time implementing the scanner and testing it on multiple websites to make sure the scanner is issue free.

April 16 – April 18: The scanner was used to test the million websites on Amazon EC2 instances.

April 18 – April 22: Spent time to consolidate the results and insert them into a database using a Java program.

April 23 – April 26: Spent time to query the database to get the required data and prepared survey report.

Design and Implementation

To survey Alexa's 1 million websites, we had built an openssl scanner that opens an SSL/TLS conncetion with the website. This scanner is written is Linux Bash and is executed on 20 Amazon EC2 micro instances. The scanner consists of two parts – first is a single host scanner that scans a single website (scanHost.sh) and the second part handles reading the input file and calling the scanHost.sh multiple times for each website (scanAll.sh). It took around 2 days to scan the websites on 20 instances.

The first part is a single website scanner that takes the website name and rank as input and prints the rank, top-level domain, domain name, if SSL is supported, version of TLS/SSL supported, if SSLv2 is supported, if SSLv2 supports weak ciphers, if same key is shared by multiple servers and if the given website is vulnerable to DROWN as a comma separated value. The single website scanner is run as:

Eg: scanHost.sh –server google.com –rank 1

For a given website, the scanning is performed as follows:

- Given a website, the script performs a DNS lookup using the 'host' command to get all the unique ip addresses associated with the website. Host command is used for given website with and without appending 'www' and both the results are merged to get the unique IPv4 addresses.
  Host <website> gives all the ip's associated.
- For each ip address obtained, the scanner checks if port 443 is open by using a tool called 'tnsping'. If the port is closed, it signifies that the website does not support any TLS/SSL versions or the website is protected by a firewall and does not allow the port to be scanned.
- For all the ip's with port 443 open, an openssl connection is created to that ip using openssl s_client. The highest version of TLS is determined and the certificate returned by the server is stored.
  Eg: Openssl s_client –connect 8.8.8.8:443 –servername google.com –tls1_2
- If the handshake fails, a handshake failed message is returned by the server. This message is generally directed to the stderr. But, to use a single function to

parse the output from the server, the error is directed to stdout using 2>&1 command.

- The output from this command is parsed and the cipher, certificate values are read and stored in an array. This output is parsed by using Linux's 'awk' tool which matches the output using regular expressions.
- The support for SSLv2 is also checked by forcing the server using openssl '-ssl2' option. SSLv2 does not support SNI here. So we exclude the server name option.
  Eg: Openssl s_client –connect 8.8.8.8:443 –ssl2
- If the server accepts SSLv2 connections, its support for weak ciphers is checked by forcing the server using openssl '-ciphers' option. Though EXP-RC4-MD5 and EXP-RC2-MD5 are the export ciphers that are supported by SSLv2, all the export ciphers supported by openssl are tested by the scanner.
- EXP-EDH-DSS-DES-CBC-SHA,EXP-EDH-RSA-DES-CBC-SHA, EXP-ADH-DES-CBC-SHA, EXP-DES-CBC-SHA, EXP-RC2-CBC-MD5, EXP-KRB5-RC2-CBC-SHA, EXP-KRB5-DES-CBC-SHA, EXP-KRB5-RC2-CBC-MD5, EXP-KRB5-DES-CBC-MD5, EXP-ADH-RC4-MD5, EXP-RC4-MD5 , EXP-KRB5-RC4-SHA, EXP-KRB5-RC4-MD5, EXP-RC2-MD5, EXP1024-DES-CBC-SHA, EXP1024-RC4-SHA, EXP1024-DHE-DSS-DES-CBC-SHA, EXP1024-DHE-DSS-RC4-SHA are the export ciphers that are forced when connecting to SSLv2.
- If the cipher is not supported, then a no cipher matched error is returned. If the handshake fails, a handshake failed error is returned.
  Eg: Openssl s_client –connect 8.8.8.8:443 –ciphers 'EXP-RC4-MD5: EXP-RC2-MD5' –ssl2
- The certificates of all the ip's associated with a given website are compared to see if the private key is shared among servers.
- Some ip addresses have port 443 open, but do not support SSL/TLS. The output from the openssl command just displays a connected message, but no communication happens. These type of servers are counted as the ones that do not accept SSL/TLS connections.
- Once processing for all the ip's is done, the website is marked as directly vulnerable to DROWN if it accepts SSLv2 connections on the same server or there is another server with shared certificate that accepts SSLv2. The result from the single host scanner are printed onto the console.
- Also, the number of websites that support SSLv2 export ciphers are noted.

The second part of the scanner reads the csv file and calls the single website scanner. This process scans 6 websites in parallel and each of these scans are run as background processes. Also, the process ensures that the background processes does not exceed the limit of processes set for a given Linux machine. All the results are printed on to a CSV file.

Handling of CSV files with large data sets is hard when required for surveying. Also, the results file contains the output from tnsping. Thus, unnecessary output is removed and the result from Scan Host is processed by a java program and inserted to a single Host table from PostgreSQL database. This database is queried to get all the required results. The database is indexed on the rank of the website to improve the performance of the queries.

Highest Version Supported	Count of Websites
TLSv1.2	587095
TLSv1.1	1225
TLSv1.0	99786
SSLv3	863
SSLv2	364
No TLS Support	310667

Table 1- Highest Version of SSL/TLS supported

Evaluation

The database is queried to get the required results. Table 1 summarizes the count of website with the highest version of TLS/SSL being used in top 1 million websites. 31% of the top 1 million websites do not support any version of SSL/TLS. 58% of top 1 million websites support TLSv1.2 and lower versions. The rest 11% supports TLSv1.1 or lower versions.

It is observed that 1227 websites still use the insecure versions of SSL – SSLv3 and SSLv2 making them vulnerable to many attacks. Of the scanned million websites, it is observed that 35728 (5%) websites are vulnerable to DROWN attack. Figure 1 gives the comparison of the number of websites that supports TLS vs the number of websites that are vulnerable to DROWN attack. Of these 35728 websites, 7439 websites directly support SSLv2 and higher versions. The rest 28289 websites share their private key with other servers that support SSLv2. Also, 22695 of these websites support the export ciphers with SSLv2 making them more vulnerable by reducing the computational power required for the attacker to decrypt the session.

Survey Based on Rank:

The results are also evaluated based on the rank of the websites. Figure 2 gives the comparison of vulnerable websites based on the rank of website. The ranks are divided into ranges of 1 to 100, 100 to 1000, 1000 to 10000, 10000 to 1 lakh, 1 lakh to 5 lakhs, 5 lakhs to 1 million. The bar chart shows a minimal increase in the percentage of websites vulnerable to DROWN, it is observed that the vulnerability of all the websites is distributed across the websites. The top 10% of websites have as many vulnerable sites as the bottom 10% websites of Alexa's top 1 million. Similar is the case when the top 30% and bottom 30% of the websites are compared. There are 2 vulnerable websites ranked below 100, 12 vulnerable websites ranked below 1000, 200 vulnerable websites ranked below 10000, 3045 vulnerable websites ranked below 1 lakh, 14075 vulnerable websites ranked below 5 lakhs, 18394 vulnerable websites ranked below 1 million. Figure 3 gives the list of websites with DROWN vulnerability whose rank is less than 1000.

Survey Based on Top-Level Domain:

The results are surveyed by grouping the websites based on the top-level domains com, org, net, int, edu, gov and mil. Figure 4 summarizes the results from survey based on top-level domains. There are a total 543629 websites with com top-level domain of which 384662 websites support any version of SSL/TLS. 18837(4.9%) of these websites are vulnerable to DROWN. 34145 websites of 50356 websites with 'org' top-level domain support SSL/TL of which 2127 (6%) websites are vulnerable. 34145 websites of 52607 websites with 'net' top-level domain support SSL/TLS of which 1783 (5.2%) are vulnerable. 41 websites of 90 websites with 'int top-level domain support SSL/TLS of which 3 (7.3%) are vulnerable. 4644 websites of 7645 websites with 'edu' top-level domain support SSL/TLS of which 477 (10.3%) are vulnerable. 3954 websites of 8079 websites with 'gov' top-

level domain support SSL/TLS of which 535 (13.5%) are vulnerable. 97 websites of 140 websites with 'mil' top-level domain support SSL/TLS of which 5 (5.2%) are vulnerable. For the other domains, there are 226219 websites that support TLS/SSL of which 11961 (5.3%) of websites are vulnerable to DROWN.

Below are some interesting stats that are noted from the survey:

- ✓ The gov and edu websites are the most vulnerable to DROWN compared to other domains.
- ✓ 13.5% of gov websites and 10.3% of edu websites are vulnerable.
- ✓ 51% of gov websites and 40% of edu websites does not support any version of TLS/SSL which signifies that security aspect is overlooked in government and educational institutions.

Other Stats from the Survey:

It is observed that there is a widespread reuse of certificates. 153175 websites share their certificates other servers which is the main reason why 90% websites are vulnerable to DROWN. Also, some websites only support SSLv2 which is the most insecure versions of SSL/TLS.

Comparison of Results from DROWN Paper:

The researchers of the DROWN project have done a survey on Alexa's top 1 million websites and it is observed that approximately 25% (82k) of servers are vulnerable to DROWN (dated March 1st 2016). Many famous websites like Yahoo, Samsung, Daily Motion, Buzz feed etc were vulnerable. By March 26th 2016(after our proposal), the vulnerable servers count has come down to 15%. It is observed from our survey that only 5% (35k) servers are vulnerable to DROWN which signifies that the system administrators are responding well to the recently discovered vulnerabilities and patching the servers. The DROWN survey concentrated mostly on port wise statistics like SMPT, HTTPS, POP3 and other ports. But, our survey is highly concentrated on most popular websites. Our survey on Alexa's top 1 Million websites is useful to understand how these websites are concentrating and keeping their websites up to date of recent vulnerabilities in SSL. Using the results we can also identify that the distribution of vulnerability is spread across all the 1 million websites and not just lower ranked websites. We are even able to identify a couple of websites in top 100, after 2 months of attack release, which is interesting. From top 1000 websites there are 12 of which CNBC is one of the well-known stock market website. Also, we identified that the vulnerability is high in percentage among gov and edu websites.

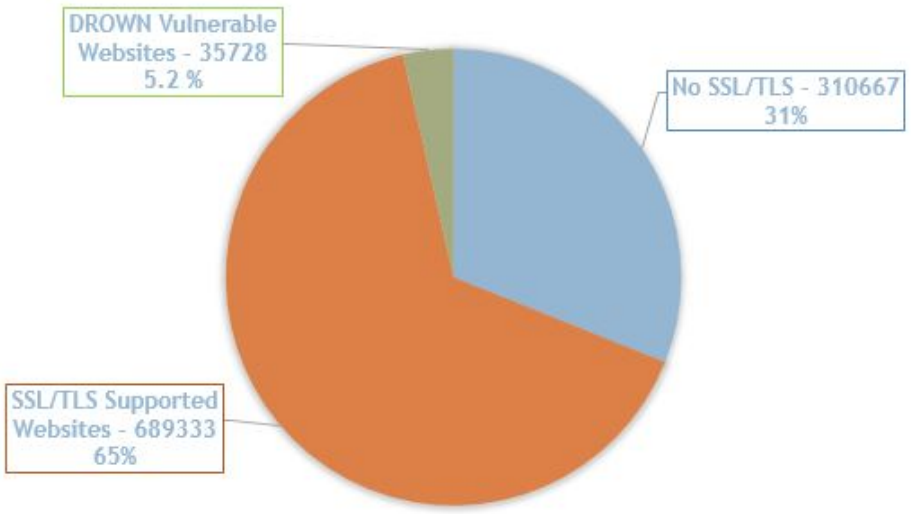

Figure 1 – Summary of all the websites vulnerable to DROWN

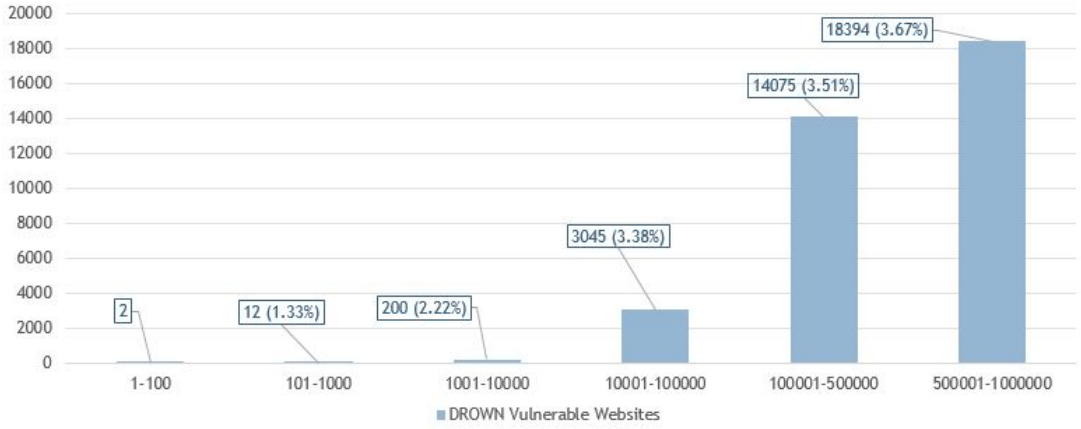

Figure 2 – Rank based survey

RANK	DOMAIN	DROWNVULNERABLE
19	weibo.com	t
97	sogou.com	t
117	youth.cn	t
150	globo.com	t
173	goo.ne.jp	t
240	iqiyi.com	t
371	digikala.com	t
421	spiegel.de	t
466	thefreedictionary.com	t
524	acfun.tv	t
545	58.com	t
778	cnbc.com	t
823	51sole.com	t
935	xmediaserve.com	t

(14 rows)

Figure 3- List of Vulnerable Websites in Top 1000

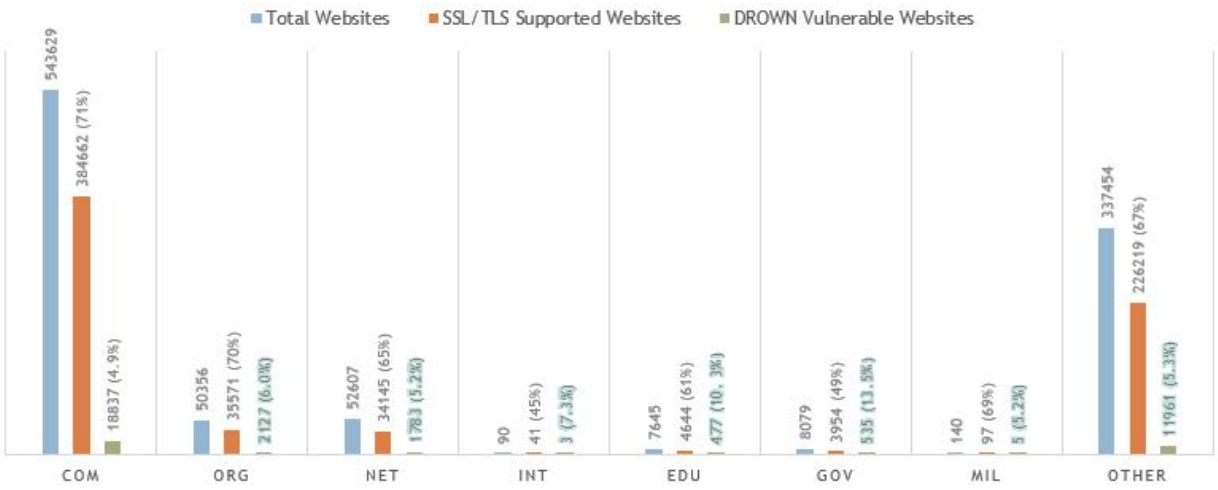

Figure 4 – Survey based on rank of the website

Related Work

The DROWN researchers have performed a more generic survey on all the IPv4 address space, but have not performed a detailed analysis of the reports based on the rank or top-level domain of the websites. There was also a survey done by Mozilla on ciphers to remove RC4 from Firefox. There have also been many other surveys done by other researchers on Alexa's top 1 million websites, but they are more specific to the field of research and most of these surveys just concentrate on top million websites, but do not survey the websites based on top-level domains or the ranks of the websites.

Conclusion

DROWN is a serious vulnerability that affects HTTPS and other services that rely on SSL and TLS. Though the server administrators have responded well on discovery of the vulnerability and the percentage of vulnerable sites came down to 5% from 25%, 35728 websites are still vulnerable and the administrators of these websites should take immediate action to disable SSLv2. Also, the discovered vulnerabilities were distributed equally across the ranks of websites. The number of vulnerable websites in top 10% of websites is equal to the number of vulnerable websites in the bottom 10%. There are still 1227 websites that only support SSLv2 or SSLv3. The use of these version should be discouraged and one easy way to handle this is browsers making changes to alerting users about the insecure version of SSL being used .The use of SSLv2 and SSLv3 is also distributed equally across the ranks. The government and educational institutions should realize the importance of security and train the server administrators to act immediately in case of serious vulnerability being discovered. Though same private key is shared by multiple servers for different purposes, the server administrators should be educated about the importance of server's certificate and should be encouraged to use a different certificate for each server.

References

1. Nimrod Aviram, Sebastian Schinzel, Juraj Somorovsky, Nadia Heninger, Maik Dankel, Jens Steube, Luke Valenta, David Adrian, J. Alex Halderman, Viktor Dukhovni, Emilia Käsper, Shaanan Cohney, Susanne Engels, Christof Paar and Yuval Shavitt: *DROWN: Breaking TLS using SSLv2.*
2. http://drownattack.com
3. http://aws.amazon.com
4. http://postgresql.org
5. http://drownattack.com
6. http://www.linux-magazine.com/Online/Features/OpenSSL-with-Bash
7. https://linuxconfig.org/bash-scripting-tutorial
8. http://www.freeos.com/guides/lsst/index.html
9. https://jve.linuxwall.info/blog/index.php?post/TLS_Survey
10. http://www.mkssoftware.com/docs/man1/openssl_ciphers.1.asp
11. http://linuxco.de/tcping/tcping.html
12. http://www.thegeekstuff.com/2010/01/awk-introduction-tutorial-7-awk-print-examples/