

Nearly a quarter of malware now communicates using TLS

news.sophos.com/en-us/2020/02/18/nearly-a-quarter-of-malware-now-communicates-using-tls/

Luca Nagy

February 18, 2020



Encryption is one of the strongest weapons malware authors can leverage: They can use it to obfuscate their code, to prevent users (in the case of ransomware) from being able to access their files, and for securing their malicious network communication.

As websites and apps more widely adopt TLS (Transport Layer Security) and communicate over HTTPS connections, unencrypted traffic may draw even more attention, since it's easier for analysts and security tools to identify malicious communication patterns in those plain HTTP sessions. Malware authors know this, and they've made it a priority to adopt TLS and thereby obfuscate the contents of malicious communication.

One of the reasons why it's easier to find signs of malicious activity in unencrypted traffic is that recent releases of malware tends to phone home more frequently than it used to, and when it does, sends increasingly larger volumes of profiling information about the target machine and network back to its operators. After it identifies the victim, malware increasingly communicates with its operator(s) in order to perform network reconnaissance, and to send the collected information to its command and control server.

This kind of information theft can be a precursor to a subsequent targeted attack, or can be used for blackmail, or sold to other criminals who may abuse it in a variety of ways. Without the protective layer of TLS encryption obfuscating the contents of this communication, a

sharp-eyed analyst or data loss prevention tool might easily catch this type of theft in the act, before the malware may cause harm.

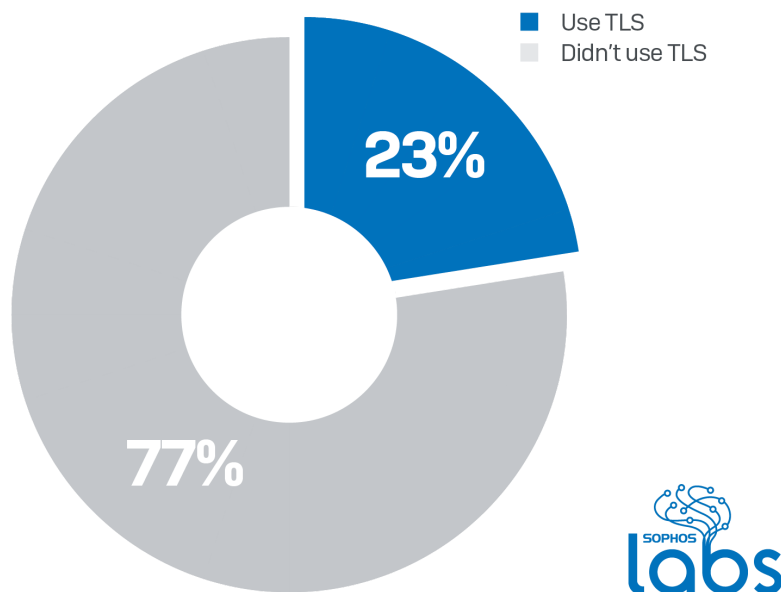
We've also observed that, increasingly, more malicious functions are being orchestrated from the command-and-control (C2) server, rather than implemented in the malware binary, and the C2s make decisions about what the malware should do next based on the exfiltrated data, which increases the volume of network traffic. Malware authors also want to empower their binaries with newer features and refresh them more often, which also increases the need for secure network communication, to prevent network-level protection tools from discovering an active infection inside the network every time it downloads an updated version of itself.

SSL/TLS usage by recent malware campaigns

Malware that does a poor job of hiding its C2 communication can be more easily detected. So, increasingly, cybercriminals have tended to adopt transport-layer encryption.

To see what the current state of the art is, we reviewed a representative sampling of malware analyses we've made over the past six months. The analyses included details about whether the malware connected to one or more machines on the internet; For simplicity's sake, we consider that sample to be a "TLS user" for the purposes of this research when the sample communicated over port 443/TCP (the standard port used for TLS-encrypted HTTPS communications) during the analysis.

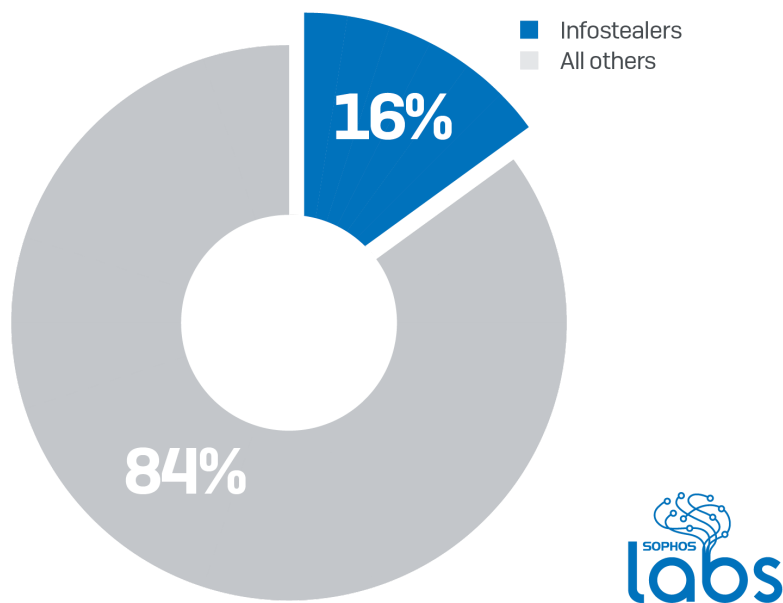
Nearly a quarter of malware that made an internet connection used TLS to communicate



Out of all the malware that made some kind of network connection during their infection process, about 23% communicated over HTTPS, either to send or receive data from the C2, or during installation when they may use HTTPS to conceal the fact that they are retrieving malicious payloads or components.

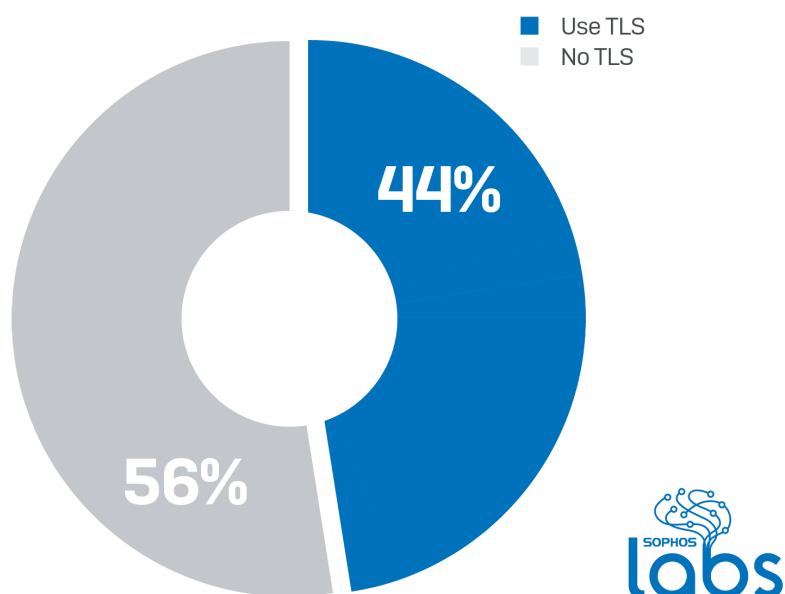
Network traffic encryption is more important for Trojans, especially information stealers. An information stealer's main goal is to collect as much data about the victim as possible, including sensitive financial information, and remain undetected while doing so. Among our sample set, information stealers made up 16% of the total number of samples tested during the time period.

Information stealers comprise 16% of the malware samples that communicate



Information stealers relied more heavily on HTTPS to communicate than any other type of malware. Even though they make up only a little more than an eighth of the total samples that made any kind of internet connection during their infection process in our analyses, about 44% of the information stealers communicate using TLS over the standard HTTPS ports.

Nearly half of malware that steals information uses TLS to exfiltrate



Using SSL/TLS gives malware the ability to conceal commands sent to the client, hide data exfiltration, or prevent the detection of downloads of additional modules or payloads. In this analysis, we consider any of those activities a *use* of TLS.

Stacking on more encryption

But malware creators do not rely on TLS alone to conceal the true nature of their activities. Many add another layer of security, such as an additional, symmetric cipher to encrypt their data, for concealment within TLS.

Malware authors usually employ higher level APIs for network communication, like WinInet.dll or URLmon.dll. With lower level APIs, the developers can craft sockets manually, so they can use custom protocols, which may be more difficult to identify.

Perhaps unsurprisingly, the adoption rate of TLS by ransomware, specifically, is much lower than the average across all malware families, because ransomware is less reliant on stealthy communication, or on stealth in general, once it has done the damage.

Legitimate services for malicious content

From the malware author's point of view, using a legitimate service to store and host malicious content has several benefits. One is that the legit services, in nearly all cases, use TLS by default, and as a side benefit, the malicious content can stay hidden and the malware distributor doesn't need to obtain their own TLS certificate for their website.

The malicious use of legitimate services like Pastebin, Dropbox, or other cloud storage services has also tended to grow. In the last six months, 0.8% of the samples we surveyed communicated directly with Pastebin, including Trojans, RATs, and infostealers.

Some of the uses of these Pastebin mal-pastes include 2nd stage downloaders, VBA or PowerShell scripts, malicious PE files, or lists of URLs where other content may be obtained. In most cases we observed, the content of these mal-pastes were also encoded with simple XOR, RC4, or Base64 to further obfuscate their true nature.

Recent widespread families using SSL/TLS

TrickBot

TrickBot's main goal is to steal information from the victim machine. It collects information about the system, user, their browsers, the network on which the computer is running, the email accounts that belong to the victim, and particularly, bank or financial account passwords or other credentials.

3886	869.910082	192.168.0.53	5.34.177.50	TCP	66	49205	→	443	[SYN]	Seq=0	Win=8192	Len=0	MSS=1460	WS=256	SACK_PERM=1	
3887	870.226375	5.34.177.50	192.168.0.53	TCP	66	443	→	49205	[SYN, ACK]	Seq=0	Ack=1	Win=29200	Len=0	MSS=1452	SACK_PERM=1	WS=128
3888	870.226852	192.168.0.53	5.34.177.50	TCP	54	49205	→	443	[ACK]	Seq=1	Ack=1	Win=66560	Len=0			
3889	870.229105	192.168.0.53	5.34.177.50	TLSv1	149											Client Hello
3890	870.545129	5.34.177.50	192.168.0.53	TCP	54	443	→	49205	[ACK]	Seq=1	Ack=96	Win=29312	Len=0			
3891	870.548545	5.34.177.50	192.168.0.53	TLSv1	1473											Server Hello, Certificate, Server Key Exchange, Server Hello Done
3892	870.563214	192.168.0.53	5.34.177.50	TLSv1	188											Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
3894	870.880355	5.34.177.50	192.168.0.53	TLSv1	113											Change Cipher Spec, Encrypted Handshake Message
3895	870.884134	192.168.0.53	5.34.177.50	TLSv1	443											Application Data
3896	870.884604	192.168.0.53	5.34.177.50	TLSv1	235											Application Data
3897	871.200904	5.34.177.50	192.168.0.53	TCP	54	443	→	49205	[ACK]	Seq=1479	Ack=800	Win=32512	Len=0			
3938	883.799606	5.34.177.50	192.168.0.53	TLSv1	235											Application Data
3939	884.011581	192.168.0.53	5.34.177.50	TCP	54	49205	→	443	[ACK]	Seq=800	Ack=1660	Win=66560	Len=0			

SSL/TLS handshake of TrickBot

This family distributes itself with its own malspam payload, and may also be delivered by other malware, such as Emotet. Emotet's effective spam campaigns have proven to make a successful combination with TrickBot.

TrickBot applies several techniques to evade detections, including process hollowing, or disabling some security tools. It has a modular structure, having several modules for stealing, moving laterally for propagation, or to provide remote access for the attackers. TrickBot usually downloads its modules using https, then injects the modules into an instance of the legitimate Windows component svchost.exe. After that, TrickBot exfiltrates any information the malware can collect with an https POST method. Besides using the standard TLS port 443, in some cases it uses unusual and distinctively-nonstandard ports such as 449/TCP, to communicate over TLS.

The TrickBot binary uses the *WinHttpRequest*, *WinHttpSendRequest* in WINHTTP.dll, with both the GET and POST methods, to download modules or send sensitive information to the server. The data sent to the C2 includes the group ID and client ID of the specific malware distribution, and one or more commands. It uses CryptoAPI to further encrypt the exfiltrated data.

IcedID

IcedID is a banking Trojan that performs web injection attacks against browsers in order to steal information. It injects itself into svchost.exe and can spread laterally through the network to infect other machines.

1130	500.991045	10.9.4.103	93.189.41.44	TLSv1	172 Client Hello
1155	503.100986	10.9.4.103	93.189.41.44	TLSv1	172 Client Hello
1161	503.102106	10.9.4.103	93.189.41.44	TLSv1	172 Client Hello
1163	503.102297	10.9.4.103	93.189.41.44	TLSv1	172 Client Hello
1167	503.103006	10.9.4.103	93.189.41.44	TLSv1	172 Client Hello
1173	503.104744	10.9.4.103	93.189.41.44	TLSv1	172 Client Hello
1175	503.104799	10.9.4.103	93.189.41.44	TLSv1	172 Client Hello
1707	505.820062	10.9.4.103	93.189.41.44	HTTP	164 GET /data2.php?AD94A3F72C8EBEE9 HTTP/1.1
1757	803.676646	10.9.4.103	93.189.41.44	TLSv1	175 Client Hello
1770	805.170875	10.9.4.103	93.189.41.44	TLSv1	175 Client Hello

IcedID traffic filtered on its SSL/TLS Client Hello messages

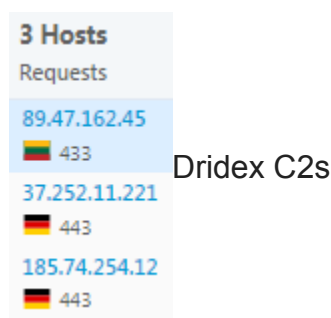
It uses WINHTTP.DLL to perform its network communication, just as TrickBot does. SSL/TLS is used in communication to and from the C2 server. It also downloads its configuration files over TLS, and the body of the responses are also encrypted using the RC4 cipher. Besides TLS, it can use unencrypted HTTP GET requests to transmit stolen information (as can be seen on the following screenshot of a packet capture). The requests include the Bot ID, and the malware's internal version number.

Dridex

Dridex is a banking Trojan, delivered by phishing campaigns and as a payload of the Emotet botnet. It was first spotted in 2011, and it is still under constant development. The recent samples use several types of code injection techniques, like Atom Bombing or Process Hollowing, against legitimate Windows executables.

Dridex has a modular framework; the loader module downloads the main module, which can perform several core functions beside downloading additional modules. It has the capability to steal credentials, cookies, certificates, keystrokes, and even take screenshots. Dridex frequently uses https on port 443 to download payload modules or send the collected data. The exfiltrated data can additionally be encrypted using RC4, if the attacker desires.

A Dridex sample communicated with the following ports and IP addresses with POST method using WinInet.dll with the usual *HttpOpenRequestW*, *HttpSendRequestW* functions



Hosts	Requests
89.47.162.45	433
37.252.11.221	443
185.74.254.12	443

TLS is here to stay

As we have seen, TLS doesn't provide malware with 100% security, as there are network inspection tools that are able to peek inside the encrypted tunnel and identify and block the malicious traffic. The proportion of malware implementing TLS to protect its communication has been and will likely continue to increase. This raises strong concerns about the ability to detect and defend against the adoption of transport layer security by malicious actors. The above-mentioned malware families include the most widespread and dangerous threats in recent years, and they are not likely to go away in the foreseeable future.

In order to protect yourself, it's important to inspect network traffic and check the TLS certificate details of https communications. You should pay significant attention to unusual or unexpected volumes of https traffic to unknown domains or using invalid or forged TLS certificates, in particular during financial transactions and when entering personal or sensitive information into browsers. Invest in a network security product that can perform these kinds of TLS communication inspections, and, ideally, can communicate and coordinate with your anti-virus product, VPN, firewalls, and/or your IDS/IPS to halt suspicious or known malicious network communications.

IoCs

The following samples were used to generate network traffic used as examples in this post.

Trickbot:

- 2baf66b83d6cd0b52e3dae66c42a0a3a3c279319c68b77e02141a2c355698409
- e17149663a7d2f9ec19d28102d8379b764c5dd83c1ec8c7278300c58893e7600

IcedID:

- da7d9687c5776eccc90f7d11dbe32623f9aa1f44fc2eff5088e0540014adcb0e
- f09fe918108769ef3200e7978d983722b407b26cc2d3fe07d3804e8e331a0986

Dridex:

- 585e245b0ba761e2ec27bf33e02dec44b888469b868d65d2511bfd5155917
- 8be38d81426e338c5daa60e35776000f5d199f562c74a57a55356861ba21703d