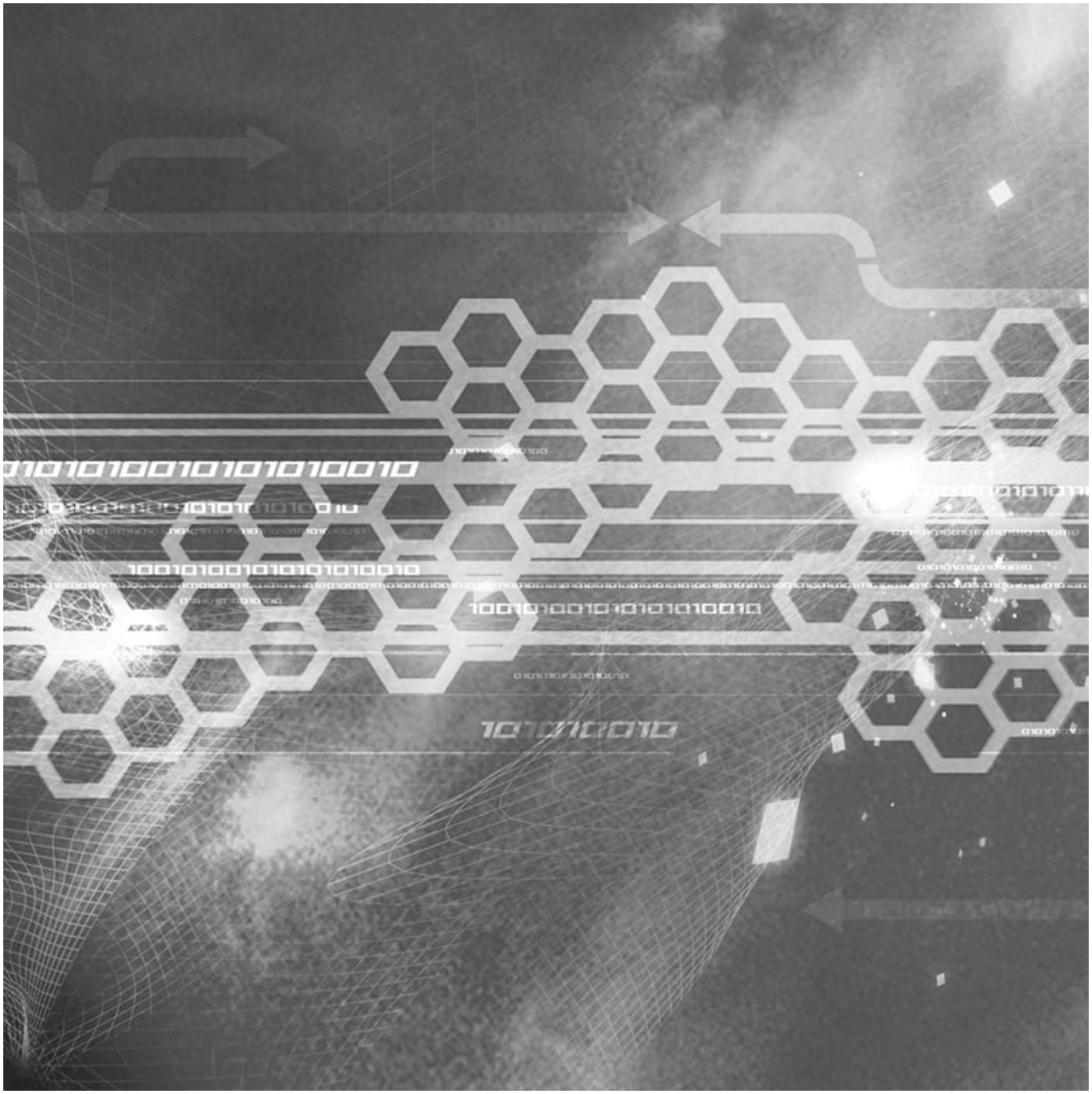
Pushdo - Analysis of a Modern Malware Distribution System

secureworks.com/research/pushdo

Joe Stewart

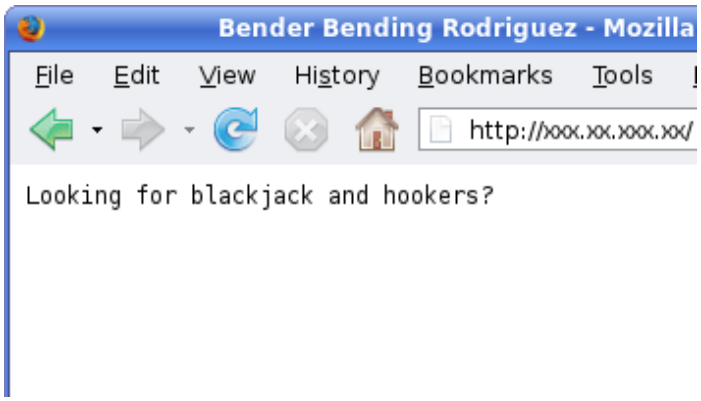


- **Author:** Joe Stewart
- **Date:** December 16, 2007

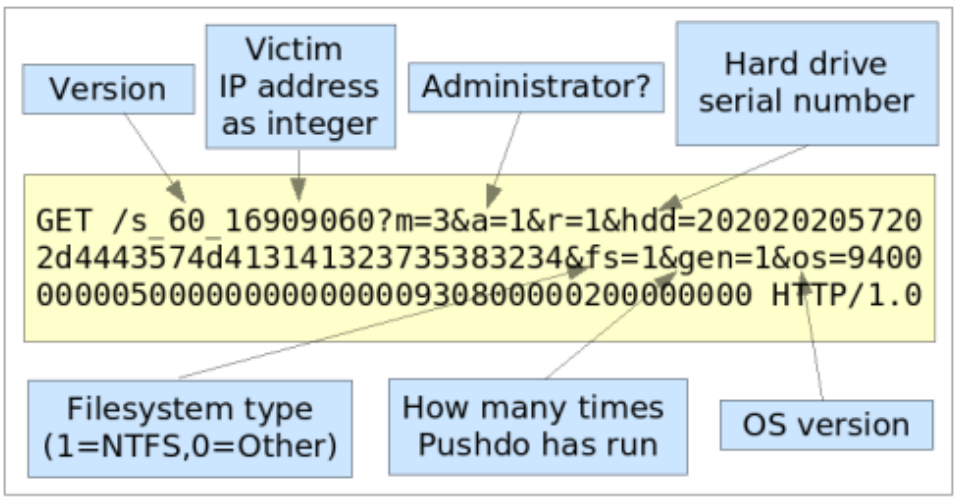
Recently, Sophos published a blog entry detailing the trouble they are having with the **Pushdo trojan**, a fairly new and prolific threat being circulated in **fake "E-card"** emails. From their description, it is clear that the author(s) of **Pushdo** are making a concerted effort to spread their malware far and wide. But what exactly is Pushdo, and how does it work? We decided to take a closer look at this malware family.

Pushdo is usually classified as a **"downloader" trojan** - meaning its true purpose is to download and install additional malicious software. There are dozens of downloader trojan families out there, but Pushdo is actually more sophisticated than most, but that sophistication lies in the Pushdo control server rather than the trojan.

When executed, Pushdo reports back to one of several control server IP addresses embedded in it code. The server listens on TCP port 80, and pretends to be an Apache webserver. Any request that doesn't have the correct URL format will be answered with the following content.



The Bender Bending Rodriguez text is simply misdirection to mask the true nature of the server - if the HTTP request contains the following parameters, one or more executables will be delivered via HTTP.



Pushdo HTTP Request Variables

The malware to be downloaded by Pushdo depends on the value following the "s-underscore" part of the URL. The Pushdo controller is preloaded with multiple executable files - the one we looked at contained 421 different malware samples ready to be delivered. The Pushdo controller also uses the GeoIP geolocation database in conjunction with whitelists and blacklists of country codes. This enables the Pushdo author to limit distribution of any one of the malware loads from infecting users located in a particular country, or provides the ability to target a specfic country or countries with a specific payload.

Pushdo keeps track of the IP address of the victim, whether or not that person is an administator on the computer, their primary hard drive serial number (obtained by SMART_RCV_DRIVE_DATA IO control code), whether the filesystem is NTFS, how many times the victim system has executed a Pushdo variant, and the Windows OS version as returned by the GetVersionEx API call.

The use of the physical hard drive serial number as a identifier is interesting - it not only provides a unique ID for the infected system, but can also reveal information such as whether the code is running in a virtual machine or not. For instance, a VMware system might return a serial number of "00000000000000000001" or simply "00", which is very easily spotted in a list of serial numbers of major hard drive vendors. This could be a way for the malware author to spy on anti-virus companies using automated tools to monitor the malware download points.

As another anti-anti-malware function, Pushdo will look at the names of all running processes and compare them to the following list of anti-virus and personal firewall process names:

- avp.exe
- Armor2net.exe
- kpf4ss.exe
- blackd.exe
- PXAgent.exe
- ipfsrv.exe
- safensec.exe
- mcagent.exe
- mpsevh.exe
- mcuimgr.exe
- mcpromgr.exe
- mcusrmgr.exe
- mcupdmgr.exe
- mclogsrv.exe
- mctskshd.exe
- NPFSVICE.exe
- outpost.exe
- symlcsvc.exe
- sspfwtry2.exe
- vsmon.exe

- xcommsvr.exe
- vsserv.exe
- livesrv.exe
- drweb32w.exe
- nod32krn.exe
- PAVFNSVR.exe
- PAVSRV51.exe

Instead of killing off these processes, as many other trojans/viruses attempt to do, Pushdo merely reports back to the controller which ones are running, by appending "proc=" and a list of the matching process names to the HTTP request parameters. This type of reconaissance is useful when determining which anti-virus engines or firewalls are preventing the malware from running or phoning home, by their absence from the statistics. This way the Pushdo author doesn't have to maintain a test environment for each AV/firewall product.

Most of the 421 malware samples from the Pushdo controller we examined were either the Wigon rootkit or the Cutwail spam trojan, however the following other trojans were being served by the controller:

- PRG/Wsnpoem
- PSW.LdPinch.NEL
- TrojanDownloader.Agent.NPQ
- Agent.AIA
- BHO.NAT
- Rustock.NBK
- TrojanDownloader.Small.NYK

The large proportion of Cutwail/Wigon leads us to believe the same group is behind all three malware families. The Wigon rootkit is dropped onto the system when Pushdo is first executed, and is used to hide the Pushdo process and any subsequent malware that Pushdo might download.

It is able to determine which processes to hide by looking for a specific byte at a predetermined offset of the PE header. Cutwail also seems to share some similar code/programming techniques (as well as the use of the Wigon rootkit) with Pushdo. For instance, the use of environment variables to determine system paths, rather than more canonical Win32 API calls. This programming approach may also indicate the author of Pushdo (and Cutwail and Wigon) is more at home with Unix-like operating systems than Win32 platforms, although clearly he/she has proficiency on both.

The fact that other malware families are being distributed using the Pushdo system suggests that the author is also willing to take payments from other malware authors in return for use of his distribution channel. Such arrangements are becoming more and more common, as participants in the malware economy seek out niches in which to provide services in the underground marketplace.

The Pushdo controller is remotely administered using a custom protocol over the HTTP channel. An administrator connects using the same URL that an infected system might, except the version parameter is set to a predetermined key. At that point the following commands can be issued over the TCP channel:

- **STAT** (gets the server status)
- **TRCK** (dump the statistics log)
- **CARG** (upload a new malware payload database)
- **FLTR** (upload a new whitelist/blacklist filter database)
- **FINA** (end session)

As we were writing this analysis, we received an e-card email containing a newer variant of Pushdo. Apparently taking notice that the Bleeding Snort project had published a signature (sid 2006377) to detect the Pushdo request variables in transit, the author has now changed the request to be less fingerprintable. An example of the new request format is:

GET
/40e800142020202057202d4443574d414c393635393438366c0000003c66000000007600000002
HTTP/1.0

The length of the request will likely change between different service pack levels of Windows. IDS/IPS signatures can still be written around such a request, taking advantage of the fact that no other HTTP headers are sent as one characteristic to key in on. However, even with this approach, false positives may still occur.

Clearly the author of Pushdo is intent on evading detection for as long as possible, in order to have the maximum amount of time to seed Cutwail spambots into the wild. Although it is unclear just how large the Cutwail botnet has become, the ambition of the project rivals that of other more well-known spam botnets, such as Storm. Only time will tell if it will rival Storm in size as well.