**TDL4 Carrier to Glupteba.**

TDSS/TDL series are kernel Rootkits while the latest TDL4 can infect windows 7 X86/X86_64 platforms.

The Rootkit has been in existence since July 2010 and still manages to successfully infect thousands of machines each day. Before TDL4, its previous version (TDL3) managed to bypass only the Windows 7 x86 32bit platform which does not employ the patchguard protection mechanism. Patchguard will deny any loaddrv attempt to insert driver which is not signed into the kernel.

Most of the analysis on TDL4 carried out using the CAMAL (COSEINC Advanced Malware Lab) device which simply saves me great deal of time performing the analysis. The main advantage of using the CAMAL is that it employs sophisticated emulation technology which incorporates full emulation and it is fast enough not to be detected by malwares.

My manual analysis was focused more on the delivery of the Glupteba using the TDL4 as its carrier.

**The infection Process**

**Delivery:**
The malware is distributed via SEO (search engine optimization) using clickjacking to affect the search results. All infected web sites were hosted in a Russian ISP named ***masterhost.ru.***

We encountered 12 infected machines which are all hostsed in masterhost.ru. The malware which is being carried by the TDL is the Glupteba (Win32/Glupteba.D)
the URL consist of the following format: <command_id><encryption_key><URL>
login=b0bah&amp;amp;key=2b15ea4e5eb2bbd734081c051a14fa41&amp;amp;affSid=0.
After we have examined the binary we were sure it was “stock” TDL4 which simply got command from the C&C to download the Glupteba and execute it. When the dropper detect it is being executed on 64bit machine it uses its ldr16 component to hook into the BIOS int13h which is the I/O interrupt.

The malware uses the IOCTL_SCSI_PASS_THROUGH_DIRECT to write directly into the physical drive and modify the MBR (master boot record) which consist of 512 bytes located inside the first sector of the hard drive. The malware copies the original MBR image to its encrypted section so it will manage to boot properly.

Why does the malware hook into the BIOS int13h? By doing so, every time the system needs I/O operation it will call the malware code as well which then waits for specific calls into memory.

The next step is to disable the signing mechanism by modifying the BCD (boot configuration data) to load without signing mechanism. The reason the TDL infect the MBR is that it simply loads first before the operation system and of course, any other security softwares. The malware boot the system by using the

*ZwRaiseHardError+OptionShutdownSystem parameter* as soon as it can so that Patchguard will be disabled.

To perform the writing process into the physical drive the root kit uses MS10-092 vulnerability in the task scheduler which allows elevation of un-privileged user. The functions which are being used to perform the elevation are AddPrintProvidor and ZwConnectPort.

**Evasive Methods:**

Every HIPS engine will perform integrity check of the MBR (or by accessing the malware infected sector), the Malware intercept the request and return false data (it possess the original MBR code). Besides intercepting the call the malware perform schedule verification to its own infected MBR and if it finds its own MBR being wiped, it simply restore it by copying it back.

To verify it is not being executed on virtual machine, the first checks by the malware are being verified inside the packer before initial infection.

**Conclusion:**

One of the biggest threats of the TDL4 is its ability in disable the Patchguard hence loading kernel drivers into 64bit operating system and disables HIPS products (meaning no protection).

There is no doubt that we will see new versions of the TDSS family and I'm sure they will create new threats and more challenging tasks for the security researchers.

Udi Shamir
Senior Malware Researcher
Advanced Malware Lab
COSEINC