

Question 1 (Describe your malware lab):

My malware lab is a Ubuntu laptop. I did not require the use of special tools or Virtualization for this analysis.

Question 3 (Is the malware packed? If so, how did you determine it was?)

Yes, it was packed with a slightly modified version of UPX. UPX usually leaves the tell-tale markings of UPX0, UPX1, ..., UPX! in the packed executable. However, those strings were not to be found. When I performed a hexdump on the executable, I was quite quickly greeted with ABC0, ABC1 and ABC! A quick modification of the executable and the off the shelf version of UXP was more than happy to decrypt the file.

Question 2 (What information can you gather about the malware without executing it?)

Having recently unpacked the executable, I elected to now use the strings command to continue my analysis. Starting from the top of the strings, the first thing I found was that the unpacked version was most likely compiled with Microsoft Visual C++.

There is an early string of 'tftp.exe -i get' which might suggest some sort of early process download, but looking through the source code (I explain how I get it under Bonus Question 1) I find that there's a variable named 'tftp_Shellcode[]' which matches the hex pattern of the bytes around the string 'tftp.exe -i get'. Looking at the source, I'm unable to find where the tftp functions are being called, so I believe they aren't being executed.

Compare that to the uninstall procedure which are the some of the next interesting strings in the unpacked file. The uninstall function is called via command and control, either by a remove or an update command. Analysis can continue on a line by line basis, but rather than tread through all the strings, I'll go on to the high level functions...

Question 4 (Describe the malware's behavior. What files does it drop? What registry keys does it create and/or modify? What network connections does it create? How does it auto-start, etc?)

Okay, assuming I don't have the source to look at, and that I don't want to just execute it in a debugging environment, here's what I gather from the unpacked executable:

1. Upon execution, it looks as if this program drops a file named Winsec32.exe. As to where, I'm not certain from the strings, though it certainly could be in %WINDIR%, since the program does include a reference to the GetWindowsDirectoryA function. There is also a potential auto-update location of <http://www.Nivdav.net/Winsec32.exe>
2. The program is autoexecuted via the following Registry keys:
Software\Microsoft\Windows\CurrentVersion\Run, Software\Microsoft\Windows\CurrentVersion\Run
Software\Microsoft\OLE, SYSTEM\CurrentControlSet\Control\Lsa
3. Other features, which may or may not be used are:
 - a. A Key-logger as evidenced by the the presence of key-codes in the strings, as well as the strings "RealmBot (keylog.p.l.g)" and "Normal key logger active"
 - b. Password collection on particular sites (e-gold,PayPal, Gmail, etc...) as evidenced by the string "Pay sites key logger active."
 - c. Windows NET command (net use, net delete, etc...)
 - d. Password brute forcing as evidenced by password and username lists.
 - e. Shell access as evidenced by the string "[REALMBOT] << Remote shell ready. >>"
 - f. Updates and Downloads as evidenced by strings "[REALMBOT] << Downloading update from: %s >>" and "[REALMBOT] << Downloading URL: %s to: %s >>"

- g. Various information about the local machine such as User information, Shares, Services, etc...
 - h. There's also a built in ftp server and http server
 - i. Port scanners, TCP Redirection, Exploit vectors, UDP Flood, File uploads via FTP, infected machine reboots, etc...
10. Next is the IRC section. This is the command and control methods. The server the infected machine will connect to is testirc1.sh1xy2bg.net and the channel is #challenge. Some potential passwords (if required) are gemp123 and happy12. However, there's an interesting string near the start of the IRC section: '*@legalize.it'. this looks quite a bit like a host wildcard, and as such could be another method for command and control authentication.

11. The registry keys it changes are as follows:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\NetBT\Parameters]
```

```
"TransportBindName"=""
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SharedAccess]
```

```
"Start"=dword:00000004
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\wuauaserv]
```

```
"Start"=dword:00000004
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\wscsvc]
```

```
"Start"=dword:00000004
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Ole]
```

```
"EnableDCOM"="N"
```

```
"EnableRemoteConnect"="N"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa]
```

```
"restrictanonymous"=dword:00000001
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\PCT1
```

```
"Enabled"=hex:00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\lanmanserver\parameters]
```

```
"AutoShareWks"=dword:00000000
```

```
"AutoShareServer"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters]
```

```
"NameServer"=""
```

```
"ForwardBroadcasts"=dword:00000000
```

```
"IPEnableRouter"=dword:00000000
```

```
"Domain"=""
```

```
"SearchList"=""
```

```
"UseDomainNameDevolution"=dword:00000001
```

```
"EnableICMPRedirect"=dword:00000000
```

```
"DeadGWDetectDefault"=dword:00000001
```

```
"DontAddDefaultGatewayDefault"=dword:00000000
```

```
"EnableSecurityFilters"=dword:00000001
```

```
"AllowUnqualifiedQuery"=dword:00000000
```

```
"PrioritizeRecordData"=dword:00000001
```

```
"TCP1320Opts"=dword:00000003
```

```
"KeepAliveTime"=dword:00023280
```

```
"BcastQueryTimeout"=dword:000002ee
```

```
"BcastNameQueryCount"=dword:00000001
```

```
"CacheTimeout"=dword:0000ea60
```

```
"Size/Small/Medium/Large"=dword:00000003
```

```
"LargeBufferSize"=dword:00001000
```

```
"SynAckProtect"=dword:00000002
```

```
"PerformRouterDiscovery"=dword:00000000
```

"EnablePMTUBHDetect"=dword:00000000
"FastSendDatagramThreshold " =dword:00000400
"StandardAddressLength " =dword:00000018
"DefaultReceiveWindow " =dword:00004000
"DefaultSendWindow"=dword:00004000
"BufferMultiplier"=dword:00000200
"PriorityBoost"=dword:00000002
"IrpStackSize"=dword:00000004
"IgnorePushBitOnReceives"=dword:00000000
"DisableAddressSharing"=dword:00000000
"AllowUserRawAccess"=dword:00000000
"DisableRawSecurity"=dword:00000000
"DynamicBacklogGrowthDelta"=dword:00000032
"FastCopyReceiveThreshold"=dword:00000400
"LargeBufferListDepth"=dword:0000000a
"MaxActiveTransmitFileCount"=dword:00000002
"MaxFastTransmit"=dword:00000040
"OverheadChargeGranularity"=dword:00000001
"SmallBufferListDepth"=dword:00000020
"SmallerBufferSize"=dword:00000080
"TransmitWorker"=dword:00000020
"DNSQueryTimeouts"
=hex(7):31,00,00,00,32,00,00,00,32,00,00,00,34,00,00,00,38,00,00,00,30,00,00,00,00,00
"DefaultRegistrationTTL"=dword:00000014
"DisableReplaceAddressesInConflicts"=dword:00000000
"DisableReverseAddressRegistrations"=dword:00000001
"UpdateSecurityLevel " =dword:00000000
"DisjointNameSpace"=dword:00000001
"QueryIpMatching"=dword:00000000
"NoNameReleaseOnDemand"=dword:00000001
"EnableDeadGWDetect"=dword:00000000
"EnableFastRouteLookup"=dword:00000001
"MaxFreeTcbs"=dword:000007d0
"MaxHashTableSize"=dword:00000800
"SackOpts"=dword:00000001
"Tcp1323Opts"=dword:00000003
"TcpMaxDupAcks"=dword:00000001
"TcpRecvSegmentSize"=dword:00000585
"TcpSendSegmentSize"=dword:00000585
"TcpWindowSize"=dword:0007d200
"DefaultTTL"=dword:00000030
"TcpMaxHalfOpen"=dword:0000004b
"TcpMaxHalfOpenRetried"=dword:00000050
"TcpTimedWaitDelay"=dword:00000000
"MaxNormLookupMemory"=dword:00030d40
"FFPControlFlags"=dword:00000001
"FFPFastForwardingCacheSize"=dword:00030d40
"MaxForwardBufferMemory"=dword:00019df7
"MaxFreeTWTcbs"=dword:000007d0
"GlobalMaxTcpWindowSize"=dword:0007d200
"EnablePMTUDiscovery"=dword:00000001
"ForwardBufferMemory"=dword:00019df7
[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings]
"MaxConnectionsPer1_0Server"=dword:00000050

"MaxConnectionsPerServer"=dword:00000050

Bonus Question 1 (Is it possible to find the malware's source code? If so, how did you do it?)

Yes, it's possible. Since the string 'RealmBoT' was all over the unpacked executable, I began my Googling there. I quickly found that there's a World of Warcraft 'Realm Bot' which is not what I was looking for. Crafting some Google negations ("realmcraft -warcraft -wow") I found a few links to Crx-realmbot. Changing my approach to focus on that string, Google quickly found a few rapidshare.com links to download. A quick pass through the files found a few indicators that this is at least a version of the same software.

Bonus Question 2 (How would you write a custom detection and removal tool to determine if the malware is present on the system and remove it?)

The simplest would be to test for the presence of the %WINDIR%\Winsec32.exe file. However, that would be short-sighted. Having the source code is quite handy in figuring out what changes are standard and which are per-botnet specific. It looks like the default executable name is updater.exe, as well as a lol.dll. We could certainly put those on the botlist as well.

One interesting method would be to use the specific registry keys as a fingerprint. Since they are set that way in the Base.cpp, they are less likely to change.

Another method would be to scan all .exe files in %WINDIR% for some of the strings that make it through the packing procedures that we would expect not to change: "Yahoo!", "UDPFLOOD", "DDoS", "WTF!?", etc...

As for removal, I would not be comfortable with less than a wipe and reinstall of the computer due to the complete control the program has of the infected computer. If that wasn't possible, the executable removal would be fairly trivial. Searching the registry for the executable name and removing it from the 4 locations given would be fairly easy. Setting the rest of the registry entries back to what they were before would be the more difficult part. I would first check the automatic registry backups to see if they have the original settings. Failing that I would restore them to factory settings.