Practical Assignment 1

FakeAV Malware Analysis

You are given a piece of malware zip file on your Windows Desktop named `FakeAV.zip` with zip password *infected*. You have been told that this software installs a fake antivirus program that then introduces nagware on your system reminding you to pay for this antivirus software. The folder contains the following executable:

```
File Name: setup.exe

File Hashes:
     MD5:  981931159e45242cc1c3dcbdb47846d7  setup.exe
```

You have until Wednesday 12 February at 11:59 PM to complete your assignment and turn in your report via Sakai.

I suggest you reserve two hour blocks for analysis of this malware and release your reservation (pressing the *I'm Done* button as soon as you are finished). Take screen shots, make good notes, record everything necessary for either your report or continued analysis in a later netlab session.

I strongly suggest you make a task list identifying all the steps you plan to take in your analysis in the order in which you take them, e.g.:

1. Win: Start RegShot and take 1st shot.
2. REMnux: Start fakedns
3. REMnux: Start INetSim
4. Win: Start Process Explorer
5. …

This malware is more complicated than the examples you've seen in the text. The malware requires installation followed by a reboot then takes action immediately. To be able to use ProcMon, you will need to select the *Enable Boot Logging* Option from the *Options* menu. When the process starts at boot time, it may not have the same name. The malware may be packed. You should verify this using the methods you've learned about it class. If, indeed the malware is packed, RL!dePacker (installed on the desktop) may be able to unpack it. If you are unable to unpack it, you should provide static analysis on both the packed and unpacked versions.

Your report should take the following form:

1. Report Title Block

The report title block should include

a) Report title

b) Date of preeparation

c) Your name

d) Your email address

e) Assignment and Class

2. Executive Summary

Briefly describe the purpose and behavior of this software including a summary of any obfuscation methods, notable imports, registry activity, file system activity, and network activity. Also briefly note any host or IPS signatures that might be used.

3. Static Analysis

Check at least the following:

a) Is the program packed? Use multiple indicators and identify your analysis of each.

b) Identify the compilation date of the program.

c) Identify whether the program is a Windows GUI or Command-line program.

d) Identify any functions imported by the program.

e) Identify any relevant strings (ip addresses, urls, etc.).

f) Identify the program sections and their possible contents.

g) Use ResourceHacker to investigate resources (if any).

4. Dynamic Analysis

Check at least the following:

a) Note any odd behaviors that occur after installation and reboot. (If some executables don't seem to function, ou may be able to move and rename them in order to make them work.)

b) Machines the malware attempts to contact by domain/host name.

c) Machines the malware attempts to contact by IP address.

d) Registry Keys created/modified by the malware.

e) Files created/modified by the malware.

f) Processes started by the malware.

5. Indicators

Identify both host-based indicators of infection and network behavior indicators that could be used by an IPS. Be specific.

6. Disinfection

Suggest any actions you identify that one could take to disable/remove the infection.