Writeup

Kevin Turkington

I. Malware defense

During last weeks lectures we learned the appropriate tools to conduct both live and file analysis of malware on samples provided by Mr. Beek. This week with Mr. Schmugar we went over how to detect malware at the start before it infects your machine. This is done through yara signatures, these signatures work by scanning any binary or file of your choice for a set of strings or hexicode bytes followed by various rules such as concatenation, kleene star, and others, to match to specific types of files under the determined rule. Before getting into how to defened against malware an anti malware analyst needs to identify the 4 stages of malware itself:

1) First Contact: the delivery of the malicious content itself.
2) Local Execution: Convining the target audience to execute the malware.
3) Establish a presence: blending into the background or implanting its services.
4) Malicous Activity: collecting or leveraging data against the target audience.

A. Yara Signitures

To prevent malware at the beginning during the first contact stages it is important for the analyst to create an accurate yet consise signiture that applies to mainly the target binary. This can be done through the use of the FileInsight program searching all strings contained in the binary and deconstructing what is and isnt important to the malware. This method can be helpful in thwarting attacks from polymorphic malware. When creating a rule for a target binary it is important to look for commonalities between serveral samples as well as using only a mix of ascii and wide characters since those are the only ones supported by yara signatures.

```
# yara signiture example
# sourced from github/Yara-Rules        APT_APT1.yar
rule LIGHTDART_APT1
{

    meta:
        author = "AlienVault Labs"
        info = "CommentCrew-threat-apt1"

    strings:
        $s1 = "ret.log" wide ascii
        $s2 = ”Microsoft␣Internet␣Explorer␣6.0” wide ascii
        $s3 = ”szURL␣Fail” wide ascii
        $s4 = ”szURL␣Successfully” wide ascii
        $s5 = ”%s&sdate=%04ld−%02ld−%02ld” wide ascii

    condition:
        all of them
}
```

B. Cuckoo

Another program used in malware defense is cuckoo, it acts as a all in one similar to the previously used malware analyzation tools like fakenet, Flypaper, AntiSpy, and ProcMon. Except the virtual enviroment is managed by the software itself (creating its own collection of sacrifical goats). Cuckoo generates csv lists containing all the results of the malware specifically:

- traces of win32 API calls
- files being created/deleted/edited
- memory dumps
- network traces
- screenshots of the desktop