SEVERAL WAYS TO SKIN A RAT

Let’s start with the tail

Jamie “Gleeda” Levy

Purpose

- DFIR investigations spanning multiple machines
- Provides a mechanism for cutting up the data into smaller digestible chunks

- Make use of mechanisms from the disk forensics realm:
  - Baselining/Whitelisting/Blacklisting
  - Indicators of Compromise (IOCs)
    - CybOX
  - Profiling

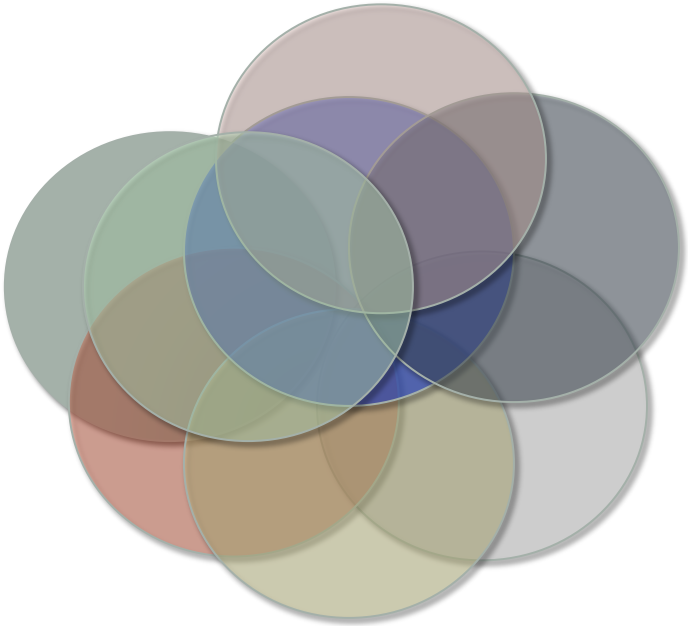
- Being proactive:
  - Hunting using prior knowledge

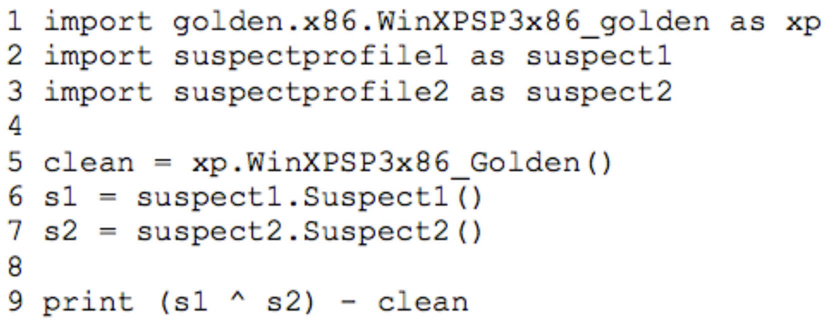
Profile Library

- Container for artifacts:
  - Processes
    - DLLs
    - Imports
    - Injected code
    - Handles
    - Heritage
    - SIDs
    - Privileges
  - Services
  - Mutexes
  - Modules
  - Drivers
  - Callbacks
  - Connections
  - Hooks
  - Registry Keys / Values
  - …

Multiple Profiles

- We can use this (set) logic against several machines at once
- Each machine (or each software/malware sample) has its own profile
- We can combine them or use differences/ intersections to see their relationships
- Profiles have different output options:
    - Text, JSON, CybOX and Profile (Python code)





```
import golden.x86.WinXPSP3x86_golden as xp
import suspectprofile1 as suspect1
import suspectprofile2 as suspect2

clean = xp.WinXPSP3x86_Golden()
s1 = suspect1.Suspect1()
s2 = suspect2.Suspect2()

print (s1 ^ s2) - clean
```

Methodology

- Build a baseline of “known good” items from a clean machine
  - Volatility plugin: “profiler”
- “Stalk” this machine over time in different states to expand the baseline
  - Volatility plugin: “stalker”
- Fill in gaps with data from disk
  - Files
  - Registry
- Examine other machines over the Enterprise to see if things pop out when compared to the baseline

Profiler Plugin

- Automates collecting all of these supported artifacts (useful for baselines by default)
- Has the following outputs:
  - Text
  - JSON
  - CyBOX
  - STIX
  - Profile
- Often inherited in order to find out specific things about the machine

Demo

- Baseline

Stalker Plugin

- Incrementally adds new artifacts from the clean machine(s) to the existing baseline

- Queries for new processes (most often)
  - If new ones exist, add all their artifacts

- Randomly queries for other new artifacts and adds those to the baseline

Stalker Plugin

- Also has an option to alert if new items are found
  - When items are found drops a profile
  - Able to generate CyBOX / STIX rules from the profile (or other profiles)
  - Able to search for these items on other machines
    - Volatility plugin: “hunter”

- Has the following outputs:
  - Text
  - JSON
  - CyBOX
  - STIX
  - Profile

Hunter Plugin

- Takes in a profile / CyBOX rule / STIX rule
- Allows you to “hunt” for artifacts in the given file against other machines or memory samples

- Has the following outputs:
  - Text
  - JSON
  - CyBOX
  - STIX
  - Profile

RegComp Plugin

- Takes in a registry file from a clean machine
- Compares keys from the registry file against those found in registry hives in memory

- Has the following outputs:
  - Text
  - JSON (compatible with profiles)
  - CyBOX
  - STIX

Demo

- Stalker
- Hunter
- Regcomp

Demos

- Dark Comet
- TeamViewer
- Mask
- Poison IVY

Questions?

Email: jamie@memoryanalysis.net
Twitter: @gleeda

Upcoming trainings:

- Austin, TX: Dec 8th-12th 2014
- San Francisco, CA: Jan 12th-16th 2015
- São Paulo, Brazil: Feb 2nd-6th 2015
- NYC, NY: May 11th-15th 2015
- Reston, VA: April 13th-17th 2015
- Amsterdam, NL: Aug 31st-Sept 4th 2015

References

- CybOX http://cybox.mitre.org/
- Leveraging CybOX with Volatility http://volatility-labs.blogspot.com/2013/09/leveraging-cybox-with-volatility.html
- Python sets http://docs.python.org/2/library/sets.html
- Baseliner EnScript https://github.com/gleeda/misc-scripts/blob/master/EnScripts/Baseliner.EnScript
- Every Step You Take: Profiling the System http://downloads.volatilityfoundation.org/omfw/2013/OMFW2013_Levy.pdf