**CMSC 691 Malware Analysis HW 8**

Name:

Assigned: 5/7/2025

Due: 5/13/2025 11:59pm

**Assignment Overview**

For this assignment, you are being given the MOTIF dataset and another dataset of benign files. Your task for part 1 is to evaluate existing, open-source YARA rules for a malware family in the MOTIF dataset. For part 2, you will write a new YARA rule for a family in the MOTIF dataset. In part 3, you will try to write a heuristic YARA rule which is able to detect as many malware samples as possible, while not matching benign files.

More information about the MOTIF dataset can be found at:

* <https://github.com/boozallen/MOTIF>
* <https://arxiv.org/pdf/2111.15031.pdf>

**HW Setup Instructions**

Download MOTIF\_dataset.7z and hw8\_benign.7z on your VM and extract them. The password to both is “infected”. Since the malware samples in MOTIF have been disarmed so they cannot run, you can leave your VM in NAT mode.

MOTIF\_dataset.7z:

* <https://drive.google.com/file/d/1UPAIRcYK3TznKz66ldmSbt-0W7OEUzkY/view?usp=share_link>

hw8\_benign.7z:

* <https://drive.google.com/file/d/1xelLNJJq8e_SRtV1RGEQAas3VDslcbB6/view?usp=sharing>

Make sure the latest version of YARA is installed on your VM by opening the command prompt **as Administrator** and running the command chocolatey install yara

**YARA Reference**

YARA documentation:

* https://yara.readthedocs.io/en/v4.5.0/gettingstarted.html
* <https://yara.readthedocs.io/en/v4.5.0/writingrules.html>
* <https://yara.readthedocs.io/en/v4.5.0/modules/pe.html>

Below is a helpful command for listing shared strings. It can be run in Powershell on your Windows 7 VM. Use the cd command to change your directory to MOTIF\_dataset/ and replace “ramnit” with the name of the malware family you’re writing your rule for.

$(foreach ($filename in ls .\ramnit\_\*){strings64.exe $filename | sort | uniq}) | sort | uniq -c | sort -Descending | less

Example YARA rule for the Ramnit family:

**import "pe"**

**rule Ramnit {**

**strings:**

**$s1 = "KyUffThOkYwRRtgPP" fullword ascii**

**$s2 = "Srv.exe" fullword ascii**

**$b1 = { 60 E8 00 00 00 00 5D 8B C5 81 ED ?? ?? ?? ?? 2B }**

**condition:**

**uint16(0) == 0x5a4d and**

**all of ($s\*) and**

**$b1 at pe.entry\_point and**

**pe.sections[pe.number\_of\_sections - 1].name == ".rmnet"**

**}**

**Part 1: Treasurehunt Family (100 pts)**

In Part 1, you will write a rule for the Treasurehunt family. The MOTIF dataset includes 10 files belonging to Treasurehunt. There are two different versions of Treasurehunt in the dataset (one of which is much more common than the other). You can list the Treasurehunt files by cd’ing into the MOTIF\_dataset/ folder and running the command ls treasurehunt\_\*

1) A string in each Treasurehunt file identifies which version of Treasurehunt it belongs to. What two versions of Treasurehunt are in the dataset? (10 pts)

1. First version:
2. Second version:

2) Another string in each Treasurehunt file is a Program Database (“PDB”) string. Each of the two versions of Treasurehunt have a different PDB string. What are these two strings? (10 pts)

1. First PDB string:
2. Second PDB string:

3) List three other strings you believe are unique to the Treasurehunt family. These strings should not appear in any other files in the MOTIF dataset or in the benign dataset. For each string, justify why you believe it is a good choice for including in a YARA rule. (15 pts)

1. String 1:   
   Justification:
2. String 2:   
   Justification:
3. String 3:   
   Justification:

Open two Treasurehunt files in Ghidra (one from each version). While importing the files, you will see a blue message that says “please select a language”. Select x86 32-bit little-endian for Visual Studio. For both files, navigate to the entry point in the Ghidra listing view. Then, answer the following questions:

In each Treasurehunt file, there is a call to the GetModuleFileHandleA, followed by a LEA instruction. The sequence of bytes beginning with the LEA instruction is unique to Treasurehunt. Beginning with the bytes for the LEA instruction, provide a byte sequence that you can include in a YARA rule for the Treasurehunt family. You will need to use wildcarding so that the byte sequence matches all Treasurehunt files. Make sure that the byte sequence is at least 16 bytes long, not counting wildcards.

4) What byte sequence (with wildcards) did you find? (20 pts)

5) Write a YARA rule for the Treasurehunt family. The rule should include a meta section describing your rule, a strings section that includes the 5 strings and the byte sequence from your previous answers, and a condition section that allows the rule to match both versions of the Treasurehunt family. The condition should also check that the file is a PE file. Provide a screenshot of your rule below. (25 pts)

6) In a few sentences, describe your design choices for the YARA rule. Make sure to mention how you chose the condition section for the rule (10 pts)

7) Run your YARA rule on the MOTIF dataset and on the benign files. Provide a screenshot(s) showing that only Treasurehunt files are detected. (10 pts)

**Writing a YARA Rule for Heuristic Malware Detection (25 EXTRA CREDIT pts)**

As described in the paper “Attributes of Malicious Files” (<https://www.sans.org/reading-room/whitepapers/malicious/attributes-malicious-files-33979>), unusual PE metadata is a very useful indicator that a file may be malicious. For this extra credit assignment, you will write a YARA rule for detecting malware based on the indicators listed in this paper. You will evaluate the precision, recall, and F1 measure of your YARA rule using the malware in MOTIF\_dataset/ and hw8\_benign/. Students with the highest ten F1 measures will earn an additional **10 extra credit points**. To prevent you from gaming the system (and to make sure your YARA rule can generalize), the F1 measure for the additional 10 extra credit points will be computed using a different dataset than the one provided. The test dataset has an equal number of malicious and benign files.

If you complete the extra credit, **please upload the YARA rule to blackboard,** named as [your last name]\_hw8.yar.

Write a YARA rule that:

* Matches as many files in MOTIF\_dataset/ as possible
* Matches as few files in hw8\_benign/ as possible
* Checks that the file is a PE file
* Has at least five conditions based on metadata listed in the “Attributes of Malicious Files” paper
* You may include anything in your YARA rule that you wish, except for conditions related to file timestamp or file size (due to limitations on the benign files I included in hw8\_benign)

1) Upload your YARA rule to blackboard along with your completed HW document (5 extra credit pts)

2) In a few sentences, describe your design choices for the YARA rule. (5 extra credit pts)

Run the following commands on your Windows VM:

* yara64.exe -r [YARA rule] [path to MOTIF\_dataset] | wc -l
* yara64.exe -r [YARA rule] [path to hw8\_benign] | wc -l
* ls [path to MOTIF\_dataset] | wc -l
* ls [path to hw8\_benign] | wc -l

These commands list the number of malicious and benign files detected by your YARA rule, and the total number of malicious and benign files.   
  
  
3) Provide a single screenshot showing the output of all four commands (5 extra credit pts)

True positives (TP): The number of malicious files detected by your YARA rule (good!)

True negatives (TN): The number of benign files that were NOT detected by your rule (good!)

False positives (FP): The number of benign files accidentally detected by your YARA rule (bad)

False Negatives (FN): The number of malicious files not detected by your YARA rule (bad)

4) Answer the following: (4 extra credit pts)

* 1. How many true positives (TP) did your YARA rule have?
  2. How many true negatives (TN) did your YARA rule have?
  3. How many false positives (FP) did your YARA rule have?
  4. How many false negatives (FN) did your YARA rule have?

5) Use the formulas above to answer the following: (6 extra credit pts)

1. What is the precision of your YARA rule?
2. What is the recall of your YARA rule?
3. What is the F1 score of your YARA rule?