**Lab #2 - Static Malware Reverse Engineering**

The following is based on true events from 2014 which occurred in a small start-up company in Monroe, CT.

A new small to medium sized eCommerce start-up based in Monroe, CT has recently begun to notice anomalies in their financial records. They have also recently received a number of customer complaints saying that invoices received from the company often bill more services than the customers asked for, and they are often directed to send check payments to a PO Box that does not look legitimate.

Upon reviewing their most recent set of invoices (saved as plain text files), the accounting department notices that the information in the invoices is incorrect, as reported by the customer complaints. The CEO orders an immediate investigation of the three accountants' computers, which had only recently been purchased last month from a bankrupt internet cafe popular for LAN parties.

As a growing company, they only employ a small team of six part-time IT support professionals. hey undertake an initial check of the systems, and they find a suspicious process running on one of the computers which seems to be connected to a McAfee service with a large amount of data being sent outside the company firewall. At this point, they do not feel that they have the expertise to carry out a full-scale malware/forensic investigation, and becoming nervous of a possible attack, the IT team scribbles down the name of the suspicious process ("greencat-2") and quickly shuts down all three accounting department computers and unplugs their power cords.

As there is increased competition in the start-up eCommerce domain, the company is anxious to ensure that their systems have not been compromised. The CEO quickly employs a cyber forensic investigator (YOU) to determine what malicious activity has taken place, where it originated from, and if the company is to blame (i.e., could be sued) for any damages.

Unfortunately, because the accounting department computers contain sensitive customer financial information, the CEO will only allow you access to the suspicious greencat-2 binary. By investigating only this binary, can you determine:

1. Could greencat-2 or disgruntled accounting employees be responsible for manipulating the company's financial records and customer invoices?
2. Was greencat-2 targeting the startup eCommerce company all along, or could it have come from somewhere else?
3. What could explain the "large amount of data being sent outside the company firewall" that the company's IT observed? What functionality in greencat-2 could cause a large amount of data to be sent out over the network?

**Instructions:**

1. Download webc2-greencat-2.7z from Canvas: Files > assignments > #1 - Static Malware Reverse Engineering.
2. Move this zip file up to an GHIDRA, unzip it using the following 7z command. The 7z file is password protected. The password is "infected" (no quotes).

7z e webc2-greencat-2.7z

1. Load the executable into GHIDRA. This malware is a standard 32-bit PE executable.
2. Starting from the WinMain(...) function located at 0x0040297D, follow the control flow in the disassembly. Imagine that you are a recursive-descent disassembler --- starting from the entry of each function and going until the return of that function.
3. Comments should be placed on every block of logically-related instructions. "Logically-related" is defined as 5 or fewer sequential instructions that together accomplish a single (easily commented) action. Control flow instructions (e.g., call, jmp) cannot be in the middle of a commented block --- they can only be the first or last instruction of the block!
4. Since this binary is dynamically linked, you will have to Google any library functions that you do not know. This will be important for understanding the return values and side effects (e.g., the memory copied by strcpy) of the library functions.
5. It will be helpful if you keep track of the values in memory! For example: some library calls will save data into a buffer in memory (e.g., GetUserNameExA) and then greencat-2 will use that data later on. GHIDRA can help with this --- it shows empty space where static variables should be. You can double-click any static variable's name to jump to its location and add comments or dummy values. Unfortunately, GHIDRA cannot help with dynamically allocated variables. You **do not** have to turn in anything related to memory tracking, but it will be helpful to you.
6. Optional: Provide answers to the 3 investigation questions from above in a text box in canvas. MAX of 5 sentences per answer. 5 bonus points per correct answer.

**Submission:**

Your commented ASCII Listing file exported from GHIDRA (called greencat-2.txt)

**Note:** You can download the webc2-greencat-2.7z file directly into your AWS workspace. After you are done with commenting, you can submit your ASCII Listing file directly from the AWS workspace (Highly recommended). Doing this will help you avoid transferring the file from the AWS workspace to your personal computer.

**Exporting Comments from GHIDRA:**

**-5 points for not following these steps properly.**

1. **Default export settings in GHIDRA will truncate comments!**
2. Click on File and then click on Export program as shown in figure below.

A screenshot of a cell phone

Description automatically generated

1. This will open a new pop-up in that choose ASCII format as shown in figure below.

A screenshot of a cell phone

Description automatically generatedText Box

1. Click on options to increase the end of line width (max comment width).

A screenshot of a cell phone

Description automatically generatedText Box

**Transferring Files:**

To transfer files from your personal device to the AWS workspace:

1. Create a zip folder of all the files that you would like to transfer to the AWS workspace.
2. Every GT student has [Box](https://gatech.account.box.com/) and [OneDrive](https://gtvault-my.sharepoint.com/) accounts given free by the institution. Login to either of those two and upload the desired files.
3. Now go back to the AWS workspace and login to either of those two services where you uploaded you zip folder. Download folder to the AWS workspace and use the appropriate 7z command to unzip your folder.

**Grade: 100 points**

**Grading Criteria:**

The grade will be based on how well you understood the functionality and capabilities of greencat-2.

-10 points for each key malware functionality that your team fails to comment or entirely misunderstands (e.g., your comments say the code is reading a file, but it is really writing to a socket). All key malware functionality in greencat-2 are reachable via careful recursive-descent (there are no anti-disassembly tricks). I will not deduct points for misunderstanding single instructions.

**Teams:**

This assignment can be done individually or in a team of 2. **Please join a group in Canvas prior to submission.** Groups can be joined from the People tab --- please choose any available Lab #2 group for your team. After all members of your team have joined the group, **only one team member** needs to submit on behalf of everyone in their group. Zoom provides ability to collaborate and video conference with your teammate.

If you would like to set it up, your team can collaborate using a GHIDRA server. The instructions for that can be found at the following blog post. Make sure to configure your server in a way that is not publicly available. This is outside the scope of this class, so we cannot provide IT support for GHIDRA servers, but please post any questions or tips on Ed!

<https://byte.how/posts/collaborative-reverse-engineering/>

For people new to GHIDRA and would like to get some insight, here is a blog post for that as well.

<https://byte.how/posts/what-are-you-telling-me-ghidra/>

**Bonus:**

5 bonus points will be awarded for each correct answer to the 3 investigation questions above.

**Submission Instructions:**

Upload your commented ASCII Listing file exported from GHIDRA (called greencat-2.txt) to this Assignment.

**What to do when you encounter technical difficulties?**

If you are experiencing technical difficulty such as being unable to access the AWS workspace, please submit a ticket to the "Digital Learning Tools and Platforms" team at <https://gatech.service-now.com/continuity>. And on the ticket, please put "Route to the DLT Team" at the top of the ticket because it will help the Service Desk know where to send it.

**Is ChatGPT (or other AI assistance) allowed for this lab?**

The short answer is yes, ChatGPT or other AI assistance software is allowed for this assignment, since after all real practitioners will use all of the tools at their disposal to fight threat actors. Please keep in mind the following, however:

* Do not directly copy anything from the AI assistant chat window. It is crucial for you as the student to be able to articulate your understanding in your own words.
* ChatGPT among other AI assistants are notoriously bad at interpreting assembly, especially assembly that a malware author is intentionally trying to obfuscate for a human analyst! Do not blindly assume that the AI assistant’s interpretation is correct.
* Often AI assistant’s interpretation of assembly does not capture the true intention of the code, and instead regurgitates the actions of the CPU for each instruction. We are expecting your comments to encapsulate the high-level reason for the code’s existence, and it is not sufficient to explain which registers are storing which values without understanding the underlying reasoning.