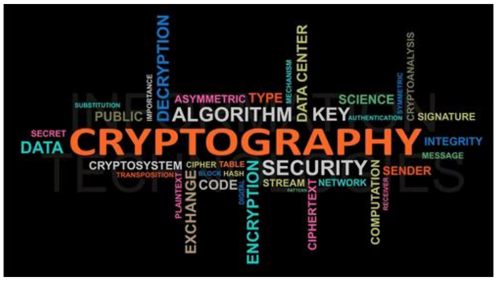
CSE 40567/60567: Group Project Announcement

❖ **Group projects:** Project topics will be related to cybersecurity knowledge and practices. You will be asked to select a topic and use related techniques to solve the proposed problems.

- 3-4 students per group
- Select a seed idea for your group project
- Fully motivate the problem (5%)
- Survey related work (10%)
- Develop your solutions (including GUI tools) and conduct thorough empirical evaluations (40%)
- A fully developed project report (25%): **You should NOT copy anything from anywhere!!**
  8-10 pages in ACM SIG Tighter Alternate style
  https://www.acm.org/publications/proceedings-template
- Project presentation (20%): 15 mins presentation + 5 mins Q/A for each group

T1: Cryptography: Techniques and Practices

Motivation and Background: This project focuses on the techniques and practices of cryptography algorithms. You are asked to build an interactive tool to explore how the following algorithms work.



- Cipher algorithms:

- **Vigenère cipher:** is a method of encrypting alphabetic text by using a series of interwoven Caesar ciphers, based on the letters of a keyword. It employs a form of polyalphabetic substitution [link].
- **RSA:** RSA is a public-key cryptosystem that is widely used for secure data transmission. It is also one of the oldest [link].
- **Triple DES:** 3DES replaces the original Data Encryption Standard (DES) algorithm, which hackers eventually defeat with relative ease. At one time, Triple DES was the recommended standard and the most widely used symmetric algorithm in the industry. 3DES is an extension of the DES algorithm covered in class and a quick introduction can be found at [link].
- **The Advanced Encryption Standard (AES):** AES is the algorithm trusted as the standard by the U.S. Government and numerous organizations. A detailed description about how AES works can be found in this link: [link].

- Requirements:

- Build a software tool to implement all the above encryption/decryption algorithms. (***Note that <u>NOT</u> existing libraries can be used***).
- Develop an interactive graph user interface (GUI) to show how to encrypt a message, and how to decrypt it. More specifically, the developed GUI tool should:
  - Allow users to select a cipher.
  - Allow users to encipher any types of messages (e.g., texts, images, and binaries). (Note that for Vigenère cipher you may only encrypt/decrypt texts.
  - Allow users to enter or import the key.
  - Decrypt messages with given keys.
  - Export encrypted or decrypted messages by saving messages or displaying them to textbox.
- Analyze the efficiency of encryption and decryption of these cipher algorithms. You should conduct experiments on all these ciphers and analyze the relations between execution time, size of target message.

- Submissions:

1. Code in any programming language your team prefers and developed software including GUI tool.
2. Report containing a) current status of each algorithm, and b) user manuals to your program and analysis.
3. Demo in class.

- T1 Primary TA: Mingxuan Ju, email: mju2@nd.edu

T2: Mini Game Cracking

Motivation and Background: This designed project will enable you to understand how software could be cracked by attackers and facilitate you to get familiar with various techniques of software reverse engineering. Follow the following tutorial and practice how to crack the Windows game Minesweeper. Then, you are asked to crack the Windows game Freecell or Solitaire.

- Tutorial and Pre-exercise:

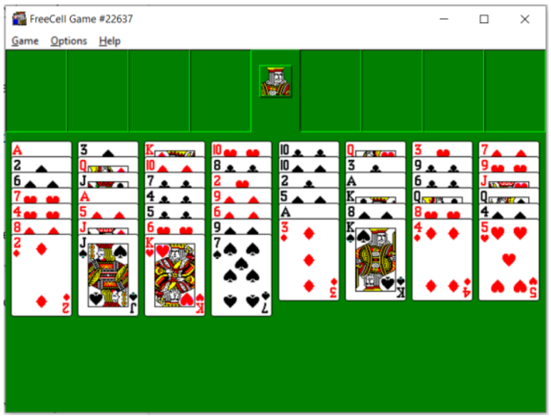
- Follow the tutorial of Reverse Engineering for Beginners: https://www.begin.re/
- Complete the challenge to crack the minesweeper with the tutorial.

- Mini Game Cracking:

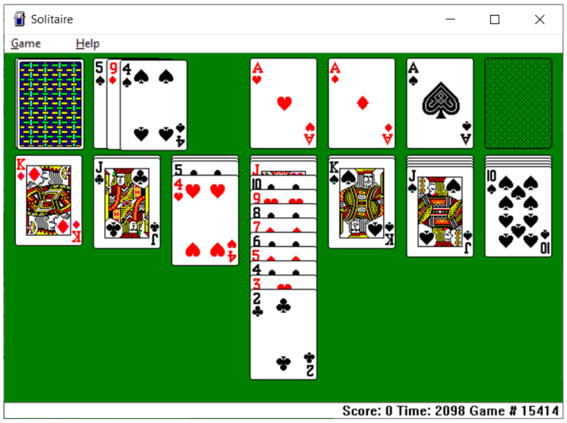
- You can choose a Windows mini game (either Freecell or Solitaire) you like and crack it!

Note: You can get the following binaries through the shared google drive:
https://drive.google.com/drive/folders/1Fg5L6TWLSUKVz6RjshNKXb9OzlRjSNlM?usp=sharing
Freecell.exe (SHA1:71e8dc557697ff81896c456fc224bbfdcf167674)
Solitaire.exe (SHA1:1c4741cb333d5e835b34cc64558cb70bfb07c6b9)
Cards.dll (SHA1:52a43a1e43a778eb43a792697c3219a0fdfa046f)
Please use the password (cse40567/60567) to unzip the file.

FreeCell:



Solitaire:



If you haven't heard of Freecell and Solitaire - now is the right time. Get acquainted with the rules and play it a couple of times :) [Freecell Wiki] [Solitaire Wiki]

- Requirements:

- You are asked to write a program/software that reads data (i.e., information that you think is necessary for winning the game, like current status of the game board) from the game and figure out how to win (e.g., a decision making algorithm that utilizes the data read from the game and helps players).

- [Bonus] Complete the challenge by automatically moving the cards according to the output of your program/software.

- **Note:** If you skip the playing process and teleport directly to the victory screen, it does not count as completing the challenge.

- Tools you may use:

- **OllyDbg**: [Download Link] is often used for reverse engineering of programs. It is often used by crackers to crack software made by other developers. For cracking and reverse engineering, it is often the primary tool because of its ease of use and availability; any 32-bit executable can be used by the debugger and edited in assembly in real time. It is also useful for programmers to ensure that their program is running as intended, and for malware analysis purposes.
- **IDA**: [Download Link] is a disassembler for computer software which generates assembly language source code from machine-executable code. It supports a variety of executable formats for different processors and operating systems. It also can be used as a debugger for Windows PE, Mac OS X Mach-O, and Linux ELF executables.
- **Exeinfope**: [Download Link] is a program that lets you verify .exe files and check out all their properties. You can also change the file name, directly open the .exe, or simply delete it. Another piece of info provided is the exact size and the point of entry. In short, you can access dozens of different options to edit any Windows executable file.
- **WinHex**: [Download Link] is in its core a universal hexadecimal editor, particularly helpful in the realm of computer forensics, data recovery, low-level data processing, and IT security. An advanced tool for everyday and emergency use: inspect and edit all kinds of files, recover deleted files or lost data from hard drives with corrupt file systems or from digital camera cards.

- Submissions:

- Code in any programming language your team prefers and developed software including GUI tool. (Your program/software should prompt players while the game is in progress.)
- Report containing a) detailed processes of how you crack the mini game, and b) user manuals to your program and analysis.
- Demo in class.

- T2 Primary TA: Mingxuan Ju, email: mju2@nd.edu

T3: Intelligent Anti-malware Software

Motivation and Background: This designed project will enable you to understand how anti-malware software could use machine learning (ML) techniques to detect malware based on the extracted feature set (e.g., Win API Calls). Two sub-datasets of PE files are prepared, each of which contains an equal amount of benign files and malware. The first dataset (4k training samples) is used for training your ML algorithms and the second dataset (1k testing samples) is used for testing (validation should be conducted within the training set). You may use your own datasets to improve the performance of your developed anti-malware detection model, and you are welcomed to use any learning algorithms based on the extracted features of the given PE files.

Raw dataset link:
https://practicalsecurityanalytics.com/pe-malware-machine-learning-dataset/
Dataset sampled from the above link for this project has been shared in the following Google drive (including 4k training samples + 1k testing samples):
https://drive.google.com/drive/folders/1g1I291oxOboXrE4JsDHCPfIFnyatm3z-?usp=sharing
(Password to unzip the file will be shared separately if your team selects this topic)

WARNING! The link will download an encrypted zip file that contains live malware. Handle the contents with care. The file extensions have been removed from all of the samples in order to prevent accidental execution; however, it's highly recommended working on this project in a sandboxed environment. As an additional precaution, you should also change the permissions of the folder to deny "Execute" permissions to all files in the folder. Please carefully follow the instructions and you will be fully responsible for the consequences resulted from your exercises. The sample set is merely used for the research purpose. You should **NOT** disseminate any malware samples to anyone anywhere; otherwise, you will be fully responsible for the consequences.

Dataset description:

Field	Description	Example
id	The identifier for the sample that corresponds to the name of the file in the samples directory.	5

md5	The MD5 hash of the file.	ad27f1a72dda61d1659810c406f37ab8
sha1	The SHA1 hash of the file.	f8fd630c880257c7e74c1f87929993477453d989
sha256	The SHA256 of the file.	984d732c9f32197232918f2fce0aa9cedc1011d93e32acb4ad01e13f2f76d599
total	The total number of antivirus engines that scan this file at the time of the query.	67
positives	The number of antivirus engines that flag this file is malicious at the time of the query.	0
list	Either blacklist or whitelist indicating whether or not the file is malicious or legitimate respectively.	Whitelist
filetype	This field will always be exe for this data set.	exe
submitted	The date that the sample was entered into my database.	6/24/2018 4:18:38 PM
user_id	Redacted.	1
length	The length of the file in bytes.	211,456
entropy	The Shannon entropy of the file. The values will range from 0 to 8.	2.231824

- Requirements:

- You are expected to extract useful and meaningful features (e.g., Win API calls, n-gram binaries, op instructions, etc.) from the given PE files. You may also extract possible features and exploit cutting-edge techniques to conduct feature selection (if so, you may explain how you perform the feature selection in your report).
- After feature extraction, you may choose/develop a machine learning algorithm such as Support Vector Machine (SVM), Random Forest, or Deep Neural Network (DNN) to train the model based on your extracted feature set(s). A vallina example for code of the learning process can be found in this link (sklearn in python). Then you need to test your model on the second/test sample set.
- Finally, you need to implement a GUI tool for users to scan uploaded PE files. Your program should conduct the predictions in the backend and display the detection results (benign vs. malicious).

- Submissions:

- Code in any programming language your team prefers and developed anti-malware software including GUI tool. (Your anti-malware software should shows the detection of any given executables in a format of PE.)
- Report containing feature extraction method, training/detection algorithm and experimental results.
- Demo in class that shows the detection of any given executables in a format of PE.

- T3 Primary TA: Yiyue Qian, email: yqian5@nd.edu

T4: Gaining Deep Insights into the Online Underground Ecosystem

Motivation and Background: Nowadays, many sophisticated underground markets (e.g., underground forums, social media groups) have emerged over the Internet, where cybercriminals exchange information with fellow criminals on abusive tactics and engage in the sale of illicit goods and services. The function for these markets is not only for social contact within users, but also to support criminal activities, such as buying or selling crimeware such as malware and crimeware-as-a-service (CaaS) such as hacking services. In order to allow law enforcement communities to devise effective disruptive strategies, there's an urgent need for novel techniques and tools to gain valuable insights into these underground markets.

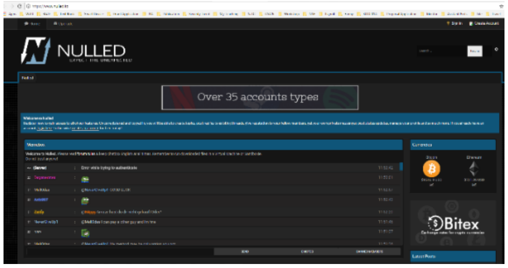


Figure 1. Nulled Forum

To gain deep insights into underground markets and better understand the cybercrime ecosystem, in this project, you are asked to first find at least five active underground markets, such as underground forums (e.g., ***Nulled***: https://www.nulled.to/#!Marketplace, ***Hack Forums:*** https://hackforums.net/forumdisplay.php?fid=107). You need to describe and summarize how you find these underground markets and explain why these markets are significant for gaining insights into the online underground ecosystem. Then conduct the following research tasks:

1. Select one underground market you explore, based on which each group needs to focus on at least one particular kind of crimeware (e.g., exploits, botnets) and one kind of CaaS (e.g., malware attack, hacking service) traded in the markets. For each type of crimeware (denoted as P1) or CaaS (denoted as S1), you need to first develop your own solutions/tools to collect a number of threads (>50 threads for P1 and >50 threads for S1) and their related comments (>5comments/thread) for further analysis (see 2-3). Describe and summarize how you collect the data.
2. Base on the collected data, you are asked to develop your own solutions/tools to analyze: (1) crimeware/CaaS trading **threads**: i.e., extract username and profile of vendor, product/service name of each thread, price, payment method, # of comments (i.e., replies), # of reviews; (2) **comments**: i.e., classify each comment (i.e., username and profile of commenter, trading [Yes/No/Uncertain], contracted customer [Yes/No], review [Positive/Negative/Neutral], Q&A [Yes/No], other). You need to submit the analysis results using the required template shared in Canvas. Your analysis/annotation will be validated by cross-validation during grading.
3. Based on the above steps, please develop your own solutions/tools for in-depth analysis: (1) find out the **key players** (i.e., most active vendors and buyers) for the kind of crimeware (P1) and CaaS (S1) you explore; (2) further analyze the top key players (one vendor and one buyer) for P1 and S1 to find out: i) whether he/she is an individual or organization; ii) what other products he/she sell or buy; iii) how he/she influent others in the market; iv) if he/she is active in other markets and how he/she will have the impacts in the cyberspace, etc. Describe the storyline and provide the case studies to elaborate your findings.
4. Develop GUI tool to demo your project.
5. Based on the above findings and analysis, devise your solutions to help inform effective countermeasures.
6. A fully developed project report with required format should be submitted.
7. Finally, present your project in the class.

- T4 Primary TA: Yiyue Qian, email: yqian5@nd.edu