Cryptography Course Project

Part 1: Ransomware + Anti-Ransomware (75%)

This project simulates a real-world ransomware attack and its defense mechanisms.

ILOs:

1. Develop a ransomware program that encrypts files using a **suitable modern cipher**.
2. Deploy the ransomware on a victim machine via a **social engineering technique** (like phishing).
3. Decrypt the files using a **key recovery method**.
4. Detect and stop ransomware activity using **antimalware** of your own design**.**

Project Overview:

Imagine you're learning how cybercriminals attack systems using ransomware. In this project, you'll play both the attacker and the defender to understand the full cybersecurity cycle. You'll create a fake ransomware attack, see how files get locked, and then build a defense system to detect and stop it.

Background:

Ransomware is a type of malicious software that locks and encrypts a victim's computer or device data, then demands a ransom to restore access. The victim must pay the cybercriminal within a set amount of time or risk losing access forever. Ransomware uses encryption to block access to infected files, making them unusable and inaccessible to victims.
You can retrieve the secret key via sending money to the attacker, then he will send you back the secret key along with the decryption script.

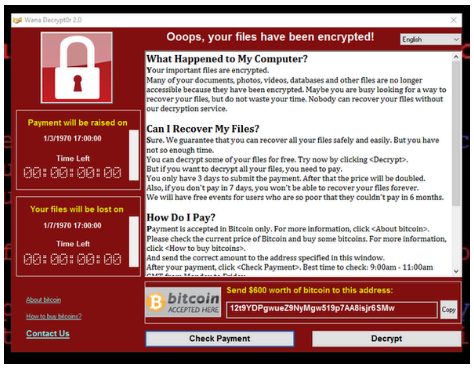
What You Will Do:

1. Create a stealthy ransomware script that encrypts a file system object.
2. Deliver the Ransomware via Phishing Email.
3. Encrypt files on a test file system object and try to recover them by obtaining the secret key.
4. Assuming you paid the attacker some money and obtained the key, you should decrypt the file system object again with zero errors in the process..
5. Write a clever script that accepts a file (or multiple files) and detects and blocks the ransomware, if flagged as a positive. OR a behavioral monitoring tool that constantly checks some OS metrics (rate of file renaming, rate of file deletion, unknown file extensions, etc.) and fires an alert if something goes outside the expected range.
6. Document all your work

Deliverables:

1. Ransomware script (for encrypting and decrypting file system objects)
2. Phishing email sender script (that sends phishing emails containing the malware)
3. Anti-Ransomware detector OR behavioral monitoring tool.
4. Project Document/Report/Slides/Readme/Video/any form of documentation

Module 1: Ransomware Script


Wana Decrypt0r 2.0

Ooops, your files have been encrypted!

What Happened to My Computer?
Your important files are encrypted.
Many of your documents, photos, videos, databases and other files are no longer accessible because they have been encrypted. Maybe you are busy looking for a way to recover your files, but do not waste your time. Nobody can recover your files without our decryption service.

Can I Recover My Files?
Sure. We guarantee that you can recover all your files safely and easily. But you have not so enough time.
You can decrypt some of your files for free. Try now by clicking <Decrypt>.
But if you want to decrypt all your files, you need to pay.
You only have 3 days to submit the payment. After that the price will be doubled.
Also, if you don't pay in 7 days, you won't be able to recover your files forever.
We will have free events for users who are so poor that they couldn't pay in 6 months.

How Do I Pay?
Payment is accepted in Bitcoin only. For more information, click <About bitcoin>.
Please check the current price of Bitcoin and buy some bitcoins. For more information, click <How to buy bitcoins>.
And send the correct amount to the address specified in this window.
After your payment, click <Check Payment>. Best time to check: 9:00am - 11:00am

Payment will be raised on
1/3/1970 17:00:00
Time Left
00:00:00:00

Your files will be lost on
1/7/1970 17:00:00
Time Left
00:00:00:00

Send $600 worth of bitcoin to this address:
12t9YDPgwueZ9NyMgw519p7AA8isjr6SMw

Inputs:

1. File system object (a folder or a file or a disk) **OR** an encrypted blob file
2. Secret key (either randomly generated or handpicked)

Outputs:

- Encrypted blob file **OR** Decrypted file system object

Functionality:

This script accepts a path to a target file system object, and a key. It scans if this target file is not encrypted, then it will just encrypt the file system object, deleting the previous object and exchanging it with another blob file.
The key is generated/picked during encryption and saved somewhere hidden.

If the script senses that the given file is an already encrypted blob file, then it asks the user to pay some money in order to reveal the key. If the key is given, then it decrypts the blob file, outputting the original file system object with zero errors and zero modified bits.

Decide how you will feed the target file and the key without the victim noticing. Document your decision. Remember that the victim will voluntarily run the script, so he must be tricked and never realize what this script does under the hood.

Properties:

1. Should be as undetectable as possible.
2. The encryption algorithm must be modern and very strong.
3. Decryption does not lose any data.
4. Should work for very large file system objects.
5. The chosen key should be as long as the ones used nowadays using the same encryption algorithm (long key).
6. Paying money should be simulated (example: click on a button to pay some money, the key is then revealed to you, you input the key to the ransomware script to decrypt) decide how you will simulate this step.

Module 2: Phishing email sender script

Inputs:

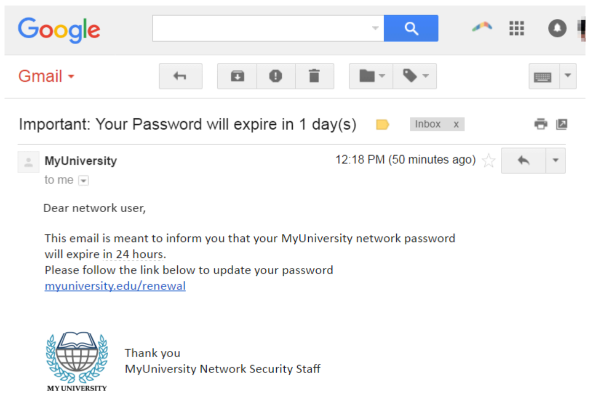
- A list (or text file) containing various victim emails

Outputs:

- Successful sending of phishing emails to all victims with a social engineering twist

Functionality:

This script accepts a list of emails, then constructs a well-structured email that seems authentic to the naked eye. The Ransomware script should be attached somewhere in the email (think and decide how you will make the user download your script, either by attaching the file directly, or linking to an external repo, etc…), document your decision and the final form of the email.



Important: Your Password will expire in 1 day(s)

MyUniversity
to me

12:18 PM (50 minutes ago)

Dear network user,

This email is meant to inform you that your MyUniversity network password will expire in 24 hours.
Please follow the link below to update your password
myuniversity.edu/renewal

Thank you
MyUniversity Network Security Staff

Properties:

1. Should be as authentic as possible.
2. Mail servers should not flag the email as malicious or spam.

Module 3 (OPTION1): Anti-Ransomware (Antimalware)

Inputs:

- A file system object (file or directory)

Outputs:

- Positive or Negative

Functionality:

This script processes the input folder/file for certain patterns that correlate with being a ransomware, there are several patterns and indicators to look for inside the subject folder/file.
Document all your scanned patterns and indicators.

Properties:

1. Should be generic and detect generic patterns.
2. Should be as clever and accurate as possible (high F1-score, search about it).
3. Should be easy to use (end-to-end).

Module 3 (OPTION2): Behavioral Monitoring tool

Inputs:

1. File system behavior indicators.
2. Process behavior indicators.
3. Network indicators.
4. System configuration indicators.
5. Etc.. (you decide which, with documenting)

Outputs:

- ACTION: stopping the malicious process and alerting the user in a certain way.

Functionality:

This script is always running in the background, typically run before running the ransomware.
It constantly checks for some indicators (like rapid encryption of files, mass file renaming/deletion, unusual outbound network traffic, etc.).
Any normal user has a set of "normal" interactions with the system, this data is collected during runtime. If an anomaly is detected, then something is wrong and there must be adversary work in-process.
If this monitor notices anything wrong, it must get the source PID of the malicious process then kill it, then notify the user one way or another. (you can only implement the notification part without the part that handles extracting the malicious PID).

Properties:

1. Should be generic and detect generic patterns.
2. Should be as clever and accurate as possible (high F1-score, search about it).
3. Should be easy to use (end-to-end).
4. Should be run one time only, then left running.
5. It is ok to detect the malware after it has encrypted some files. (latency is not penalized but greatly despised)

Criteria:

Module 1 (30%):

1. Enc/Dec functionality (20%)
2. Key recovery simulation (ex: paying the attacker) (5%)
3. Stability during encrypting/decrypting large folders (5%)

Module 2 (5%):

1. Being authentic (2.5%)
2. Avoiding mail server flaggings (2.5%)

Module 3 (30%):

1. Functionality (15%)
2. List of scanned patterns/indicators (10%)
3. Ability to be generalized to different inputs while working as intended (5%)

Testing (10%):

1. Ransomware will be tested against some predefined detectors, then the grade is put based on the F1-score. [also this will be manually assessed for how you achieved stealthiness]
2. Detector will be testing against some predefined test files, then the grade is put based on the F1-score.
3. 5% For testing against trivial test cases, the other 5% based on extensive testing based on the f1-score after running against more test cases.

Documentation (0%):

Documentation has no significance in the grade, BUT, if any team lacks documentation, they will be assumed to not understand the functionality of a certain module thus losing some grades.

Documentation is very easy, don't make it too formal, just tell me what you did, consider it as a log file where you dump all your thoughts and steps. But a slightly structured log file 🙂

Part 2: CTFs (Capture The Flags) (25%)

Scrooge McDuck has passed out, you discover that you are related to him in a certain way and legally, you have a percentage in his loot. Scrooge McDuck left a secret message having your name on it, stating that there is a hidden box under his bed containing a $5k check.
But Scrooge McDuck didn't want anyone to loot him, he made it clear that no ordinary human can break this chest.

You examine the chest and discover that this chest contains other chests that you will need to open.
They are like stacking dolls!!
And each doll/chest requires a certain key to open.

There are 4 nested chests.
Chests become incrementally harder to break
No ordinary human can crack those chests, it requires some super human, a Computer Engineer for example.

You are free to use any tools, you are also encouraged to explore the available tools and try to use them (free work experience yay) 🙂.

The key structure is in the form of:
1- KEY{SOME_TEXT}
2- CMPN{SOME_TEXT}

Deliverables:

1. Folders containing .md or .txt files or any other form of documentation on the detailed steps on how you solved each CTF.
2. If you use any code, you must incorporate it.
3. Be ready to be asked about everything regarding your solution.

Project phases:

Phase (1) Week 9: Online overview discussion

Teams will have to schedule a meeting with the TAs to review some points regarding part 1,
For example, the following:

1. The used encryption algorithm
2. Pseudocode or block diagram or plan on how to implement the ransomware
3. The final form of the phishing mail (code is not necessary)
4. Will you settle on a behavioral monitor or an anti-malware?
5. List of indicators / patterns you will scan for.

No grades are given in this phase, however it is very important to keep us updated and be guided through this project, we offer theoretical help for everyone.

You are not obliged to have written some code by week 9, but it is very advisable to begin implementing modules as soon as you have solid intuition about them.

Phase (2) Week 13: Final Submission

Teams will submit and discuss Part 1 and Part 2.
Grades are put in this phase.

Further Restrictions:

1- Teams of 4, EXACTLY 4 (We will handle the remaining students our way)
2- The entire team is expected to attend the discussion in their assigned slot, so cross-tutorial teaming is ok as long as students can attend at the designated time for the team.
3- Plagiarism is penalized
4- Latency in discussion (or DDoSing the discussion is not tolerated), we will give each team very enough time for discussion.