Analysis of Hidden Tear: An Open Source Ransomware-like Crypter Kit

Daniel Kim

December 15, 2015

What is Hidden Tear?

Hidden Tear is a ransomware-like software written by Turkish security expert Utku Sen for educational purposes. The software works both on networks and entirely offline. Hidden Tear uses the AES algorithm for encryption. The AES algorithm uses weaker symmetric encryption, whereas the worst variety of ransomware that one can get would use asymmetric 1024 bit encryption. The software is very small, only taking up 12KB. It was also undetectable by antivirus programs as of August 15, 2015. The pubic repo can be found here:
`https://github.com/utkusen/hidden-tear`

Analysis of Source Code

Encrypter Class

Hidden Tear is mainly written in C#. The main encryption work is done in the file "Form1.cs". The main object is a "Form" class with the encryption implementation of the AES algorithm as well as methods for encrypting files and directories. There is a target URL that one edits. There are also methods for creating and sending a random password for encryption. The "EncryptFile" method takes in a filename and password as strings and converts them to their raw bytes. The password raw bytes are then hashed with SHA256. Afterward, the file bytes are passed to "AES_Encrypt", the class's AES algorithm implementation method. The files are also modified to have a ".locked" file extension. The "encryptDirectory" method has an array of strings of file extensions that are to be encrypted within the directory; one can edit this array to include other file extensions. As with "EncryptFile", the directory and password are taken in as strings. "encryptDirectory" encrypts not only all the files in the current directory (using "EncryptFile") but also recursively goes down into all child directories and encrypts all files in those until all files starting from the passed-in directory have been encrypted. There is also a method to create and save a message to the victim; the default message is "Files have been encrypted with hidden tear. Send me some bitcoins or kebab. And I also hate night clubs, desserts, being drunk." The method "startAction" proceeds to encrypt everything starting from the specified user path. There is a timer that can be modified to specify when to begin encryption.

Decrypter Class

The decrypter class is simply the opposite of what the encrypter class does. There is a method named "AES_Decrypt" which implements the decryption part of the AES algorithm. Whereas the encrypter class had methods “EncryptFile” and “encryptDirectory”, this class has “DecryptFile” and “DecryptDirectory”. As before, “DecryptFile” takes in a filename and password as strings and converts them to bytes. The file bytes are then sent to “AES_Decrypt” for decryption. “DecryptDirectory” works the same way as “encryptDirectory”, recursively going down the child directories of the passed in directory until all files have been decrypted. The software overall seems to be meant for use on Windows systems as shown by the libraries used and the fact that the user directory assumed to be present is “C:\Users \”. Making Hidden Tear work on different systems would entail changing a lot of code to use different paths and different library functions.

Offline Version

The difference between the main encrypter class and the offline version is that there is no target URL to attack. The encryption key is written to a file that is meant to be on the USB stick being used to execute Hidden Tear. Instead of sending the encryption key to a target URL, the encryption key is stored as a file on the target machine. The actual work of encrypting files and directories is still the same as the online version. Like the online version, one can modify when the encryption of the target machine begins. There is only an offline version of the encrypter class since the decrypter class does not need to make any distinction between online and offline.

The way to use the offline version is to have a USB stick with the .exe file of the Hidden Tear offline program with a .pdf icon as well as a normal-looking pdf file according to whatever “social engineering scenario” the attacker is in. There should also be a text file made beforehand; this is the file where the encryption key is stored. The initialization of the class then runs the pdf which leads to running the encryption. The .exe is copied to the computer and is run, beginning encryption after the specified wait time. The encryption in fact takes mere seconds to finish, a perfectly good span of time that could easily be explained away as a quick transfer of files.

Warnings

The creator very explicitly warns to not use Hidden Tear as actual ransomware. Doing so will definitely involve legal consequences; one can potentially face obstruction of justice charges for even running Hidden Tear at all.