# ResearchCoders' For Programmers Series Honeyfiles: Deceptive Files for Intrusion Detection V.1

Mohammed Q. Hussain
mqh@reseachcoders.dev

January 2019

This document is one of "For Programmers" series, a part of ResearchCoders project. It explains the ideas of [1] for programmers to help them implement them. Please visit our website for more information: http://www.reseachcoders.dev

## 1 Introduction

The work in [1] presents the concept of *honeyfile* which can be used by the users of a given system as a simple intrusion detection system (IDS). Honeyfiles technique is a *deception defense mechanism*. When honeyfiles are introduced to a system and the users start to employ this functionality, it is going to help them to detect the attacks on the system. As its name indicates, honeyfiles uses files to deceive the attackers in order to detect them. In the next few section, we are going to introduce the concept of *deception defense* and then the technical details of honeyfiles technique will be given for people who are interested in implementing such a system.

## 2 Deception Defense

In computer security, *deception defense* is a way of protecting systems from attacker but instead of traditional way which aims to prevent the attacks, in deception defence the attacker will be *deceived*. We all know that perfect security of any given system is unattainable. Therefore, multiple ways other than completely-preventing-the-attacks has been presented in the scientific literature. Deception defense one of those ways, the other one is Moving Target Defense.

The work in [1] is one of deception defense techniques. Another example of deception attacks is *honeypot*, which is a normal machine (or virtual machine) in a network which appears to provide some services, for example serving a website, but in reality, the honeypot tries to appeal the attackers to intrude it. Once the attacker take the bait and attack it, the honeypot is going to log detailed information about the attacker in order to be analyzed later by the system administrator. Of course, the honeypot has no valuable data for the attacker, so by attacking it, the attacker just wasted her time and has been exposed for the system administrator.

## 3 Honeyfiles

Honeyfiles are normal files in the system, with appealing names for the attacker (e.g. passwords.txt), the legitimate user knows that these files are honeyfiles and have no meanings, therefore he is not going to interact with them, but the attacker doesn't know that. For example, let the user defined a new honyfile called "credit-cards.txt" on her own directory in the network. Then she put some random numbers in this honeyfile to appear as a real confidential file. When an attacker gains an access to the system somehow, this file is going to be appealing for him [1], so he is going to open, download or copy it to obtain its content. Any interaction to a honeyfile is going to cause an alert, this alert will be sent to the legitimate user who is in this way will know that there is somebody it trying to access a honeyfile, which means there is probably and attacker compromised the system.

The original paper [1] proposed this concept to be used in file server (e.g. FTP), but the sky is the limit. The concept of honeyfiles can be implemented in many ways, for example, FUSE can be used to implement a filesystem that present honeyfiles as a future, or even a kernel's level filesystem can be modified to present such a concept, use your imagination :-).

## References

[1] Jim Yuill, Mike Zappe, Dorothy Denning, and Fred Feer. Honeyfiles: deceptive files for intrusion detection. In *Information Assurance Workshop, 2004. Proceedings from the Fifth Annual IEEE SMC*, pages 116–122. IEEE, 2004.

[1] We can see the deception part here.