Sardar Patel Institute of Technology

Department of Computer Engineering
Academic Year 2016-17

Advanced Honey Encryption: An Escape-less Trap for Intruders

Name - Ajay Singh, MTech.(Comp.)
Project Guide : Dr. Anant Nimkar

Abstract

Most digital networks rely on cryptosystems that provides the confidentiality and integrity of traffic carried across the network even in presence of third party. This research paper concentrates on honey encryption, the proposed technique that converts its defensive action into both detection (tracing the hacker) and deflection action with the help of generation of special type of fake key. It will also show how the Victims will be notified about the attack along with other users who also have the probability of being attacked.

Introduction

Most security breaches involve accessing unauthorized data during network communication between two ends illegally. For the recent years, the intruders have demonstrated increased technical knowledge, developed new ways to exploit network vulnerabilities and created advanced software tools to automate attacks. To overcome these problems, one team of researchers, have finally found a way called honey encryption, which turns their defence into attack on the hackers themselves—by spewing fake keys at them and sending them drowning. This concept is used to deflect attackers when they try to decrypt the captured data that has been encrypted. It not only prevents the hacker from decrypting and retrieving the original data but also let that hacker sink in the pool of false keys.

Aim

To understand honey encryption technique that converts its defensive action into both detection (tracing the hacker) and deflection action with the help of generation of special type of fake key.

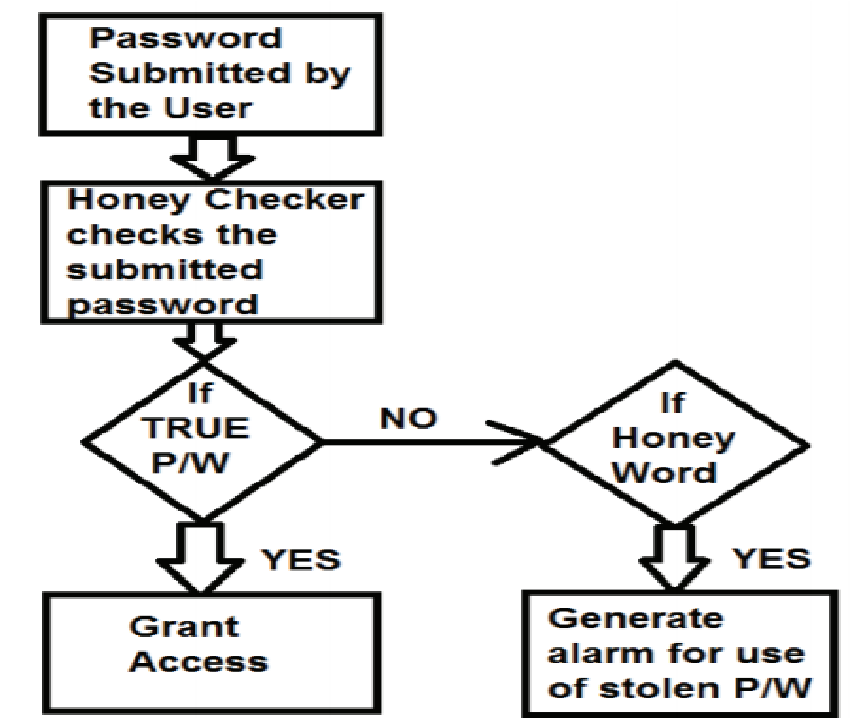
Objective

The proposed technique will keep the user passwords strong along with the ability to trace the IP address of intruders, notify the victim as well as other users whose chances of becoming next victim is high.

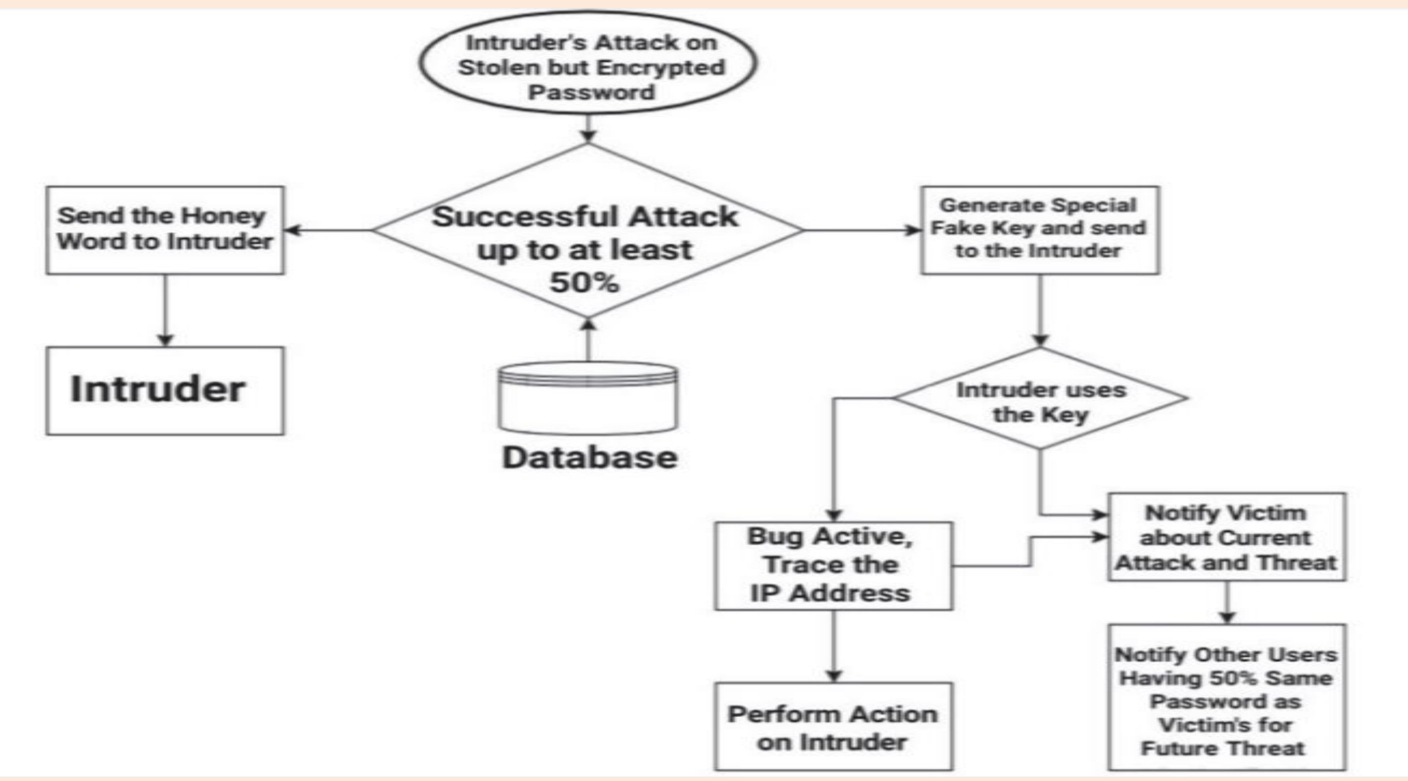
Methodology

As the intruder receives the encrypted data or the encrypted vault password, his first action will be to decrypt it and receive the original data by brute force attack. Honey encryption can be modified in such a manner that if attempts are made to crack the password the researched action comes into play. After the attack, the attempted key by the intruder will be matched by the original key. If less than 50% of the attempted key is found to be similar to the original key then the account will be considered in the safe zone. After which the honey words will be supplied to the intruder. When 50% or more of the attempted key is found to be similar to the original key the original key is moved since it could lead to other victim with similar key and action is taken on the intruder.

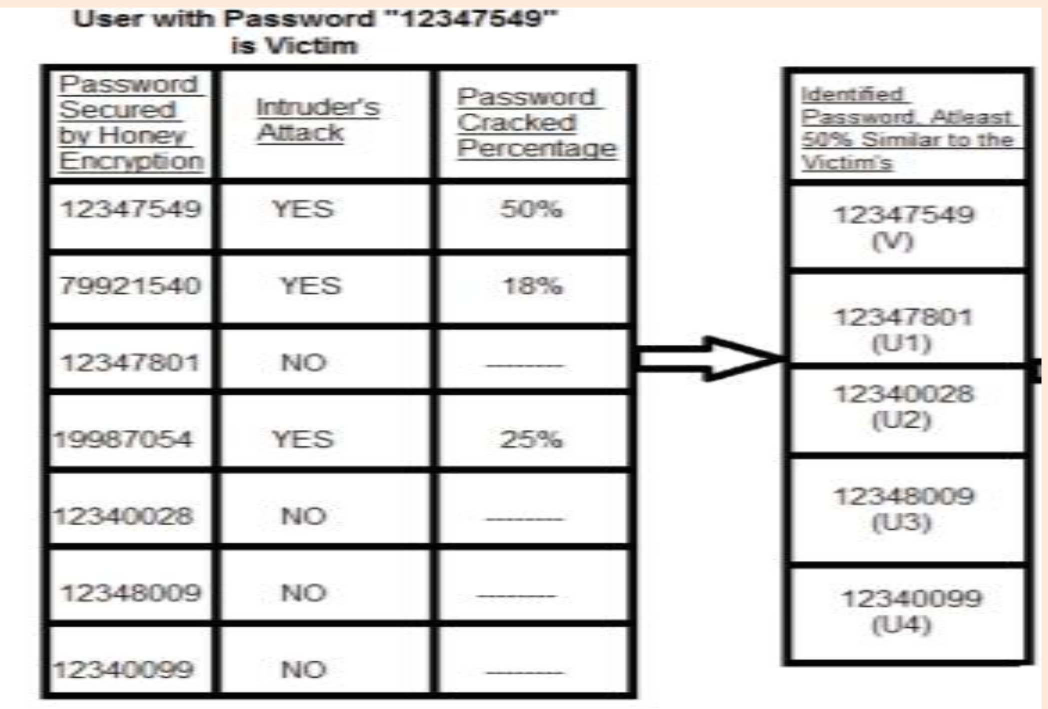
$$\text{Sweet Words} = \left[\begin{array}{l} P1 \\ P2 \\ . \\ . \\ Pi \text{ (Original Password)} \\ . \\ . \\ Pn \end{array}\right.$$


Password Submitted by the User
Honey Checker checks the submitted password
If TRUE P/W
NO
If Honey Word
YES
YES
Grant Access
Generate alarm for use of stolen P/W

Design


Intruder's Attack on Stolen but Encrypted Password
Successful Attack up to at least 50%
Send the Honey Word to Intruder
Intruder
Database
Generate Special Fake Key and send to the Intruder
Intruder uses the Key
Bug Active, Trace the IP Address
Notify Victim about Current Attack and Threat
Perform Action on Intruder
Notify Other Users Having 50% Same Password as Victim's for Future Threat

Result



User with Password "12347549" is Victim

Password Secured by Honey Encryption	Intruder's Attack	Password Cracked Percentage
12347549	YES	50%
79921540	YES	18%
12347801	NO	-------
19987054	YES	25%
12340028	NO	-------
12348009	NO	-------
12340099	NO	-------

Identified Password, Atleast 50% Similar to the Victim's
12347549 (V)
12347801 (U1)
12340028 (U2)
12348009 (U3)
12340099 (U4)

Conclusion

The proposed method of honey encryption lets the user as well as software developers know about the mind level of intruders so that that can be prepared for future threats as well.

References

[1] A. Juels, T. Ristenpart, "Honey Encryption: Security beyond the BruteForce Bound" in Advances in Cryptology—Euro crypt 2014, IEEE Journal.