A Survey on Current States of Honeypots and Deception Techniques for Attack Capture

Asst. Prof: Vaishali Shirsath
Vidyavardhini's College
of Engineering and technology
Vasai, Maharastra - 401202, India

Abstract **- Deception based mechanisms are accustomed for enhancing the security by inflicting misperception on assailants who take activities for barrier. In this paper we attempted to reviews problems related to honeypot and deception based defensive strategies inside the cyberworld, most importantly, need to characterize the honeypot wonder and abridge its preferences and drawbacks, this paper gives overview of honeypot techniques and various types of honeypots and the different deception techniques used for counter assaults**

Keywords – Cyber-Deception, Cyber-Baiting, Cyber-Security, Intrusion-Detection, Network-decoy, Network Security, honeypot, Network Tarpit.

I. INTRODUCTION

Security is the significant worry of today's cyber world in systems administration and Internet related research regions, where honeypot is one among the arrangements of secured network framework. honeypot machines are such sensible virtual machines which will be intended to find and catch the intruder; inside it can be planned on the possibility of virtual machines upheld service that one may see on an average machine.

In the space of Information security the term of honeypot alludes to a firmly observed figuring assets that one might want to be examined, assaulted and traded off [1] with honeypot, security system uses deception technique to provoke the assailants, deceptions are often creation of faux atmosphere to deceive intruders Therefore, Honeypot systems are meant for making a faux computing atmosphere so as to entrap the offender in a very faux system.

The motivation behind this paper is to examine and dissect the honeypot trickiness and give sensible foundation of honeypots and its sorts.

II. BACKGROUND

This paper investigates the historical backdrop of honeypots which show what sensible job they play in cybersecurity and what can or can't be accomplished with them, to outline the favorable circumstances and disservices of this tools and take a look at legitimate and moral issues that must be viewed as while beguiling the clients, in any event, for protective reason.

A. History of Honeypot

A honeypot is a well-known cybersecurity mechanism for detection and countering assailants' assaults. It's a decoy placed within the network, disguising itself as a sensitive asset or network vulnerability, once an assailant tries to access this faux confidential data, the honeypot notifies about the attempt.

A honeypot also collects and analyzes information on hacking attempts, fundamentally, a honeypot is a security framework made explicitly for pulling in various shorts of assaults. It works like as some other lure they make themselves a simple so that accordingly assailants can at first assault them. These frameworks license one to distinguish virus assaults and interruptions at a beginning time while likewise keeping these assaults from arriving at your genuine resources, consequently one can utilize honeypots as a safety effort. An average honeypot comprises of two segments: a vulnerability emulation framework and an observing tool. The essential is utilized to make the assailants feel that there's a weakness in your framework that they can settle. The observing framework, thus, advises you once assailants take the trap and attempt to exploit the vulnerability you've left uncovered.

B. Roles of Honeypot

While a novel tool, honeypots can't possibly Interchange different devices utilized in cybersecurity. Once endeavoring to gauge the ease of use of such an instrument, in cyber - assaults one can look at its intensity as far as counteraction, identification, and response [2].

Counteraction or Prevention is an act that makes it progressively troublesome and less rewarding for the assailants to breach the framework.

Identification or detection is a procedure of recognizing an assault and its effects on the framework inside the setting of secrecy, respectability, and handiness.

Response or reaction is the procedure of recovering from the assault. The best outcomes should strengthen the prevention and detection capacities just as limit the misfortunes.

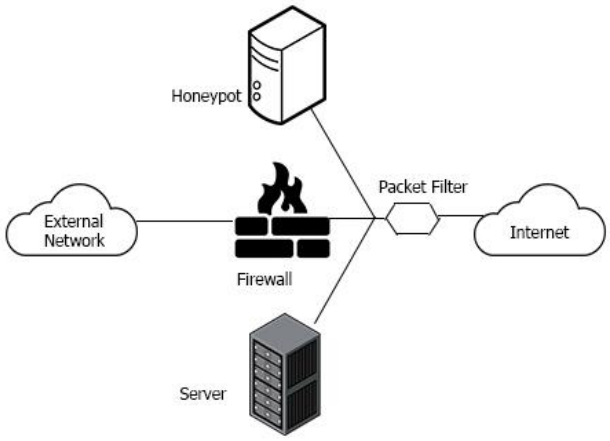

Fig. 1. Basic Framework

In any case, honeypot include extraordinary worth when worried about distinguishing the assaults. though as of now referenced Intrusion Prevention System (IPS) or Intrusion Detection Systems (IDS) are designed to distinguish assaults on production frameworks, which may deliver numerous faux positives since it is incredibly hard to perceive the assault among the surge of real traffic, faux negatives are conceivable too, from the indistinguishable explanation, or once the signature of assault is not known. Honeypots, then again, doesn't contain any genuine traffic whatsoever. This makes it helpful for distinguishing the assaults, while considering all the traffic malignant. The reaction or response is a zone where honeypots have the most significant potential. To effectively recuperate from an assault and strengthen our counteraction and identification capabilities, one ought to see the assault first. The honeypot offers absolutely that, utilizing this device, one may examine the means to breed the assault, built up the misused vulnerabilities and find the harm caused, in addition, the honeypot information isn't overflowed with genuine traffic.

C. Taxonomy of Honeypot

Depending on their purpose honeypots are dived in two group [3].

- Research honeypots
- Production honeypots

The motivation behind research honeypots is to examine the assault in detail to uncover intentions and real advances taken. This generally requires high-association honeypots, next to confronting a higher hazard. This sort of honeypots doesn't legitimately improve the security of an organization, in any case, the information picked up can encourage to improve the prevention, detection and reaction abilities. Research honeypots contribute enormously to the investigation of cybercrime, since it licenses to watch a criminal in real life or find a fresh malware. Therefore, universities, government, and military utilized it more frequently.

The production honeypots, then again, should be basically deployed, present insignificant or no risk to the organization and improve the security. That portrays low-communication honeypots, that fill the identification needs. The organization isn't interested about the thought processes of the assailant or the exact methodology, yet in reality the assault is going on and how to stop it right away. This kind of honeypot is regularly deployed in business organizations to help their cybersecurity.

The traditional way to deal with honey potting is keeping up a server, inactively anticipating assailants to return, be that as it may, there is another methodology utilizing the client as the proactive side [4]. Subsequently, we can isolate the honeypots into following classes.

- Server honeypot
- Client honeypot

The typical Client honeypot is represented by an internet browser client, creeping malignant sites. While less-intuitive honeypots can exclusively mimic the program, the high-intuitive client honeypots utilize the genuine one.

Another sort of honeypot characterization relies upon its structure.

- Physical honeypot
- Virtual honeypot

Virtual honeypots are recreated on a hosted machine unlike physical honeypot which itself is the genuine machine. The fundamental points of interest of virtual honeypots are higher partition and furthermore the hazard to run numerous honeypots on a solitary machine. The genesis of virtual honeypot system can be called to Honeyd [5].

TABLE I. TAXONOMY OF HONEYPOTS

Level of -Communication	Purpose	Form	Service	
High-Communication	Research	Physical	SSH	SMTP
Medium-Communication	Production	Virtual	FTP	TELNET
Low-Communication	Production	Virtual	HTTP	---

One can likewise recognize various kinds of honeypots relying upon the services it's giving or emulating SSH honeypots, SMTP honeypots, FTP honeypots etc.as suggested by M¨uter, Michael, et al. Along these lines, we have Totally various service gives distinctive data to gather anyway likewise causes various issues, with usage and data extraction.
In table I, the different kind of honeypot characterization is outlined.

D. Classification of Honeypot

Honeypots are divided into two classifications:

- Low-communication honeypots
- High-communication honeypots

We should take a closer look at every one of these classifications.

1) Low-communication honeypots

Low-communication honeypots emulate exclusively the basic parts of a weakness. For instance, when one has to find an endeavor to access the computer with a web server on it,

there's no impulse to introduce an entire web server. one can execute a small script which will emulate an open port on the framework and a couple of fundamental replies, at that point use it to deceive the vulnerability scanners employed by assailants.

The best thing concerning low-communication malware identification frameworks is that they're easy to convey and maintain and function admirably as an essential line of the intrusion detection framework. Nonetheless, it can catch exclusively a confined measure of data and aren't suitable for recognizing significant level of assaults performed through an authentic affiliation.

2) High-communication honeypots

In differentiation to low-communication honeypots, their high-communication analogs empower assailants to proceed openly with the assault up to some extent. Interestingly, high-communication honeypots make it about outlandish for assailants to tell if it's a genuine situation or a virtual domain that was explicitly created to divert the assault.

Below figure shows the essential structure of a high-communication honeypot.

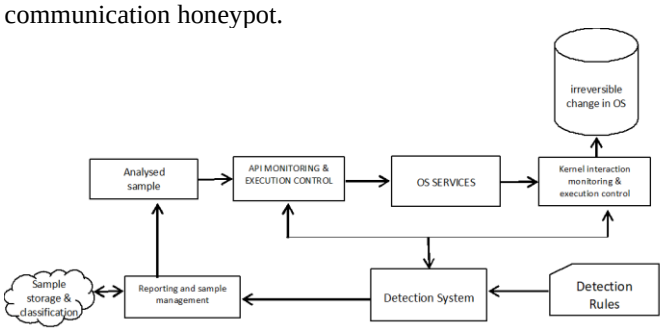

Fig. 2. High -Communication Honeypot

TABLE II. LOW-COMMUNICATION VS HIGH COMMUNICATION

	Low -Communication	High -Communication
Installation	Easy	More difficult
Maintenance	Easy	Time consuming
Risk	Low	High
Need control	No	Yes
Data gathering	Limited	Extensive
-Communication	Emulated Services	Full control

E. Advantage and Disadvantage of Honeypot

Now, let's take an in-depth look at advantages and disadvantages of honeypots. [3] [6] [7].

The benefits of honeypots are as follow.

Significant Data – honeypots give us distinctive information about the assailants, utilizing research honeypots, one can find the assailant's thought processes, personality and assets, this kinds of data can't be acquired with some other cybersecurity instrument, or it is exceptionally increasingly hard to do thusly.

Diminished Data Set – utilizing honeypots as a different figuring asset, the data acquired are a far distance of all real traffic. The examination is in this way so plentiful and simpler, without the chance of faux positives.

Finding New Assaults – one can ready to set up new malignant substance that is being spread, investigate it, and are accessible with the fitting barrier in time, utilizing honeypots, assailants give us their new weapons store intentionally, adjacent to an exhibit on how they use them.

Adaptability – The idea of deluding the assailants is pivotal however honeypots show up in different structures. The engineer can execute the usefulness to precisely match his/her needs relying upon what sort of data they need to collect, what hazard and cost of maintenance they need to support.

The drawbacks of honeypots are as follow.

Limited View – Though of incredible worth once assaulted, honeypots become futile when the assailants don't target them, this can be the basis, honeypots can't replace other cybersecurity tools in detection functionality.

Environment Risk – Due to many interactive honeypots deployed in the environment risk is far greater. on the off chance that one allows controlling the computer by assailants so as to contemplate their conduct, they can in the end utilize the honeypot to dispatch various assaults or to taint different system inside the network, without our insight.

Disclosure of Honeypot – Once assailants find the nearness of a honeypot, they adequately get around it, debilitating its advantages, what's more terrible, they can control the data the honeypot assembles to delude the safeguards.

Utilizing a lure framework to delude the consideration of assailants has the two points of interest and inconveniences. Here are the primary advantages of exploitation honeypots for malware discovery:

- They help to keep assailants from offending your genuine resources.
- They have a lower rate of faux positives.
- One can utilize logs gathered by honeypots concerning the capacities and expectations of assailants.

Concerning the disadvantages of recognizing malware with the help of honeypots, there are numerous difficulties relying upon the kind of malware one would need to distinguish.

III. CYBER SECURITY WITH DECEPTION

This paper gives the outline of various terms, standards, and methods associated with honeypots and misdirection for the most part utilized in cybersecurity.

1) Honeytokens

Honeytokens are regularly viewed as honeypots also. In general honeypot appearance is consider as a computer, physical or virtual. On the off chance that one can look at the honeypot definition, it has a reasonable watchword "resource", rather than a computer. This incorporates extra advanced assets like documents, credit card numbers, email addresses and so on the point, notwithstanding, stays indistinguishable their worth lies in unapproved use, one can for the most part place some honeytokens in the environment which he/she need to watch and afterward be cautious for their use or getting to, due to honeytokens, one can find the fraud as well as who did it, how they attempted to utilize that asset and what sort of asset was captivating for the assailants, being referenced that the service can be imitated in low-

communication honeypots, close by with the working framework or the entire machine. However, it demonstrated very fruitful to copy referred to vulnerabilities inside the service also. It's an incredible bait for the assailants and doesn't require resource stacks [9].

2) *Tarpitting*

Tarpitting is a method used to overload assailants. The underlying thought of tarpits was to forestall spreading worms and other web misuses, for example, spams and broad scanning. It was initially upheld as a Tarpitting honeypot LaBrea [10]. The point of this honeypot is to squander assets of assailants. Instances of tarpits in service are SMTP, where the single email causation technique may take a few hours and is dismissed in the long run. Before the assailants acknowledges there is something incorrectly, he burns through stacks of time. This is especially helpful for contents, which probably won't notice anything by any means. LaBrea can hinder the essential phase of cyber-assault, observation, since it tunes in to pathetic ARP solicitations and reactions to them resulting assailants can't choose which IP is associated with a genuine computer.

3) *Honeynets*

Honeynets, as sketched out by Spitzner [11], are the intense of research honeypots. It incorporates a conveyed honeypot system, normally genuine ones, where nothing is imitated. It leverages foe the total intensity of the genuine condition. We can become familiar with an amazing arrangement of data from such honeynet [12].

IV. DISCUSSION

The basic principle of honeypots deployment is that the deception service tools are straightforward to setup and maintain, it's given the nice result as scans.

Below fig shows honeypot framework, where deception play the pivot role of honeypot success, deceptive honeypots can be used effectively if they are employed in integrated manner with different security tools such as IDS and firewall honeypots analysis can be used to modify the network security configuration consistent to the security policy.

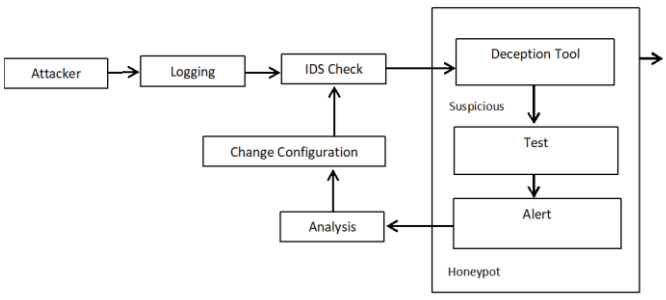

Fig. 3. Honeypot Framework

V. EXAMPLES OF HONEYPOT

A few frameworks engineers will in general group honeypots dependent on the focused-on programming they are endeavoring to shield or uncover. while a rundown of honeypots may be broad, beneath are recorded probably the most famous ones here:

- Spam honeypot: conjointly called as spam trap, this honeypot is explicitly made to get spammers before they hit authentic email boxes. These commonly have open transfers in order to actuate assaulted, and work intimately with RBL records to square vindictive traffic.

- Malware honeypot: This kind of honeypot is made to recreate powerless applications, APIs and frameworks to acquire malware assaults. The data that is then gathered will later be utilized for malware design observation, to help in making successful malware indicators.

- Database honeypot: Databases are a standard objective of web assailants, and by setting up a database honeypot one can watch and learn diverse assault procedures like SQL infusion, benefit misuse, SQL service abuse and undeniably more.

- Spider honeypot: This kind of honeypot works by making bogus sites and connections that are exclusively open by web-crawlers, not by people. When the crawler gets to the honeypot, it's distinguished related to its headers for later examination, commonly to help with blocking noxious bots and ad-network crawlers.

Numerous honeypot instruments were made exclusively as a proof of idea, furthermore, are never again bolstered by their maker. Just few are future term ventures. Notwithstanding, these are commonly described by straightforward deploying and use. A few of the consequent honeypots were created as a major aspect of The Honeynet Project, referenced prior. Aside from honeypot programming, they are answerable for a few tools for malware analyses like Cuckoo4 or intrusion detection signature generator Nebula5.

a) *Low-communication client and server honeypot*

There are a few modules and subsequent follow-ups that pre-owned this product, similar to Honeycomb [13] for making intrusion detection Signature or HOACD [14] that packaged Honeyd as a prepared to-utilize programming. All the referenced projects are, in any case, never again bolstered.

Another honeypot with an extraordinary effect was LaBrea [10] which we previously depicted in area III. It's Tarpitting strategies were utilized conjointly in sweet snare [15] honeypot as an assurance against worms.

client honeypots speak to the dynamic viewpoints. They at some point slither the Internet, looking for vindictive servers and record their conduct. Low-communication honeypots typically just reenact the perusing client, utilizing contents.

Once more, several client honeypots were offered as a proof of idea honeypots with no extra help. These models incorporate PhoneyC [16], HoneyC [17], Monkey-Spider [18], Spy Bye [19], or ADSandbox [20]. Task Thug [21], successor of PhoneyC, is partner effectively developed client honeypot in Python. It imitates entirely unexpected internet browsers and their vulnerabilities and examines the got malware utilizing Google V8 JavaScript motor.

b) *High-communication client and server honeypot*

Then again, high-communication client honeypots don't appear to be confined by looking at the reaction. Rather, the entire framework is reachable to the assailants and the choice about the site is made relying upon the progressions to the framework once the website is visited, from the ended activities, we can list the underlying MITRE's Honey Client [22] venture, the Capture BAT [23] and Capture-HPC [24] by The Honeynet Project, WebExploitFinder [25], Shelia [26], and parcels more. A fascinating expansion of Capture-HPC honeypot was upheld by HoneyIM [27]. They produce an imitation texting (IM) clients, which might be considered honeytokens.

When the recently got malware needs the high-communication honeypots started to show up a lot later than low-communication ones, and their range is lower further. Their improvement is harder further due to the upkeep costs. Argos [28], discharged in 2006, last refreshed in 2014, is a notable emulator for giving a high-communication condition to catching zero-day assaults and prior inconspicuous adventures. It utilizes memory spoiling system to accomplish that. Honey wall [29] is high-communication honeypot from The Honeynet Project. It uses Sebek [30] on the grounds that the honeypot center, more seasoned high-communication honeypot programming, and offers a GUI for organization and recognition. It hasn't got any help for quite a while too. High -Communication Honeypot Analysis Tool (HI-HAT) [31] is also called high-communication honeypot which permits changing any PHP application to an intelligent honeypot.

V. ON GOING CHALLENGES

While tending to honeypot problems, one can partition it into two fundamental regions [8]. The essential territory is the advancement of the honeypot, its productive organization, and economical upkeep. The subsequent territory is the examination of gathered data, its representation, information extraction and higher subjective procedure upheld the information, while entirely unexpected service and honeypot types face various issues in these two territories, during this paper, abridge the general issue that emerge in the vast majority of the honeypot examples, with accentuation on the momentum condition of the honeypot look into.

a) *Challenges to Develop Honeypot*

Developing the vulnerability emulation framework is probably the greatest test in honeypot improvement, with the degree of working framework associations being the primary driver of issues, the more malware activities one needs to identify, the higher the number of working framework collaborations. Besides, one has to comprehend not just a lot of conceivable pernicious activities performed by an infection or malware yet in addition where precisely one can recognize such activities. The unpredictability of tasks required for getting malware with a honeypot shift from composing a characterization framework channel driver. and infusing it into the framework to estimating framework execution swing memory assignment.

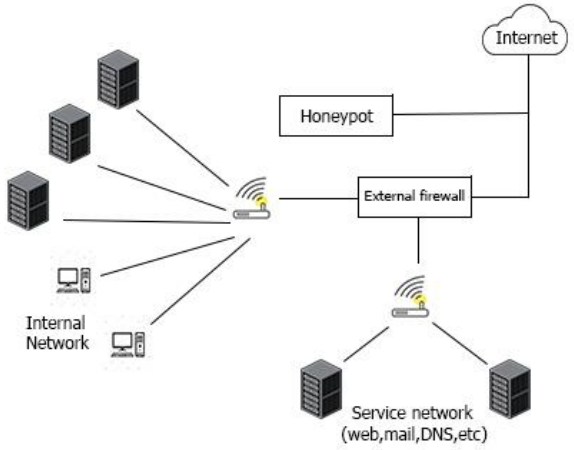

Fig. 4. Honeypot deployment

Another extraordinary test of building up a honeypot for identifying infection assaults and malware covers up inside the issue of imitating the conduct of a genuine framework. Executing a sandbox matched with an observing device is in fact entangled.

Since one can't foresee what type of activities malware will perform once actuated or what sort of information it'll require, one can't deliver a totally secure sandbox. in this manner, even once set in a honeypot, malware ought to be given some opportunity inside the genuine framework; else, it might leave the Honeynet Project.

b) *Challenges to Scale Honeypot*

In spite of the development complexity, the primary issue of utilizing high-communication honeypots for recognizing infection assaults and malware is quantifiability putting away individual conditions requires a heaps of disk space, and to have some accessible space, one should urge a great deal of time tidying up these situations, simultaneously, it won't have impact on the further malware execution since there will be a few curios staying inside the framework.

VI. CONCLUSIONS

To conclude no tool can be great and perfect, security is scarcely accomplished with the blend of all, despite the fact that deception can provide us with significant data about the assault and how to forestall it later on, it can't stop the assault itself, honeypots can be one of the developing computer security innovations. the primary thought behind the honeypot is utilize the duplicity to assemble the information regarding the assailant’s exercises and strategy. while the honeypot innovation is not a silver slug in the cybersecurity, it has a few unmistakable highlights. one can exceptionally see the assailants inside the cyberspace and respond quickly

to the current threats. Getting simple to convey and keep up is an unquestionable requirement so that it can be an incredible device for the cybersecurity network also within the succeeding years. The ability to stay undetected is additionally legitimately connected with their ease of use.

VII. REFERENCES

[1] M. Z. Xie, "HoneyIM: Fast detection and suppression of instant messaging malware in enterpriselike networks.," in *Twenty-Third Annual Computer Security Applications Conference*, 2007.

[2] K. Vasilomanolakis, "Hostage: a mobile honeypot for collaborative defense," *in Proceedings of the 7th International Conference on Security of Information and Netwrok . ACM*, p. 330, 2014.

[3] R. M. Spitzner Lance, "The value of honeypots, Definitions and values of honeypots," *https://www.symantec.com/connect/articles/value-honeypots-partone-definitions-and-values-honeypots, last accessed April 2019*, vol. I, 2011.

[4] Spitzner, "Honeytokens: The other honeypot," 2003.

[5] C. Seifert, "Capture–A behavioral analysis tool for applications and documents," in *Digital Investigation - 4*, 2007, pp. 23-30.

[6] R. Rocaspana, "Shelia: a client-side honeypot for attack detection," [Online]. Available: https://www.cs.vu.nl/~herbertb/misc/shelia/. [Accessed May 2019].

[7] N. a. T. H. Provos, "Virtual honeypots: from botnet tracking to intrusion detection," 2007, Pearson Education.

[8] N. Provos, "A Virtual Honeypot Framework," *USENIX Security Symposium*, vol. 173, 2004.

[9] T. Project, "Sebek," June 2019. [Online]. Available: https://www.honeynet.org/project/sebek.

[10] B. D. H. Project, "Announce: Hoacd 1.0 (bootable openbsd + honeyd cd)," https://lwn.net/Articles/ 90941/.

[11] G. H. Portokalidis, "Sweetbait: Zero-hour worm detection and containment using low-and high-interaction honeypots," *Computer Networks*, vol. 51, no. 5, pp. 1256 - 1274, 2007.

[12] G. S. H. Portokalidis, "Argos: an emulator for fingerprinting zero-day attacks for advertised honeypots with automatic signature generation," *IEEE*, vol. 40, no. 4, 2006.

[13] B. Portokalidis, "Sweetbait: Zero-hour worm detection and containment using low-and high-interaction honeypots".

[14] J. Nazario, "PhoneyC: A Virtual Client Honeypot," *LEET*, vol. 9, pp. 911 - 919, 2009.

[15] W. Nawrocki, "A survey on honeypot software and data analysis," *ArXive-prints*, 2016.

[16] M. Nawrocki, "A survey on honeypot software and data analysis," *arXiv preprint arXiv:1608.06249*, 2016.

[17] I. M. A. Mokube, "Honeypots: concepts, approaches, and challenges," in *Proceedings of the 45th annual southeast regional conference*, ACM, 2007.

[18] MITRE, "Honeyclientproject," [Online]. Available: https://github.com/dkindlund/honeyclient. [Accessed May 2019].

[19] T. B. M. M¨uller, "WEF – Web Exploit Finder, Detecting Drive-By-Downloads using VMware and Rootkittechnologies," [Online]. Available: http://2014.kes.info/archiv/material/bsikongress2007/poster-mack.pdf. [Accessed May 2019].

[20] D. R. Lukas, "Current State of Honeypots and Deception," in *11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, 2019.

[21] T. Liston, "Tom Liston talks about LaBrea," in *http://labrea.sourceforge.net/Intro-History.html*, 2019.

[22] C. J. C. Kreibich, "Honeycomb: creating intrusion detection signatures using honeypots.," in *ACM SIGCOMM computer communication*, 2004.

[23] T. V. D. Kaur, "Comparison of network security tools-firewall intrusion detection system and Honeypot," *Int. J. Enhanced Res. Sci. Technol*, no. 200 - 204, 2014.

[24] R. Juels, "Honeywords: Making password-cracking detectable," in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, 2013.

[25] R. Hes, "The Capture-HPC client architecture," Victoria University of Wellington, 2009.

[26] W. Han, "HoneyMix: toward SDN-based intelligent honeynet," in *Proceedings of the 2016 ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization*, 2016.

[27] A. Dewald, "ADSandbox: Sandboxing JavaScript to fight malicious websites," in *Proceedings of the 2010 ACM Symposium on Applied Computing*, 2010.

[28] A. Dell'Aera, "Thug: a new low-interaction honeyclient," *Recuperadoel*, vol. 11, 2012.

[29] S. &. Bruce, "Secrets and lies: digital security in a networked world," *John Wiley & Sons*, 2011.

[30] C. Bailey, "A hybrid honeypot architecture for scalable network monitoring," *Technical Report CSE-TR-499-04*, 2004.