International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

HYBRID CYBER DECEPTION FRAMEWORK FOR PROACTIVE AND INTELLIGENT NETWORK

Dr. Ann Zeki Ablahd Magdacy Jerjes[1], Dr.Anwar Abbas Hattab[2], Ihsan Hassan Hussein[3]

[1,3]Northern Technical University,Kirkuk, Iraq, [2]Mustansiriyah University ,Baghdad ,Iraq

[1]drann@ntu.edu.iq [2] anwarabbas76@uomustansiriyah.edu.iq [3] ihsan.bayoglu@ntu.edu.iq

Abstract

In this paper, a new system Hybrid Cyber Deception System (HCDS) is presented, designed and developed in python as a proactive tool against a complex network intrusion.

Honeypots and Honeytokens adopt the strengths strategy that is deployed in this system of deception techniques in a dynamically adaptive modality.

The realistic Honeypot service that the system (HCDS) simulates and successful redirection of the attackers in controlled and instrumented settings. The proposed framework is a system enabling the management of credentials smoothly connected with the alerting process using standardized logging libraries and SIEM-supported APIs and Honeytokens identifier track able decoy files in the form of an array.

An important innovation is network parameters and decoy distribution continuous, and autonomous modulation, which was done through Moving Target Defense (MTD) principles, which are reified through network administration utilities organized using Python. The key component of the system is a complex Deception-driven Alerts Correlation Engine, which was designed on the basis of Python programming language and Pandas Python library to handle data effectively and Scikit-learn framework to perform complex machine learning-based analysis of interactions within the deception environment. The high threat alerts are constructed by the intelligent analysis by the engine of the interaction patterns of the attacker in deployed honeypots and honeytoken and digest the data streams fed by Python-based monitoring agents. A proactive defense element is also included in the system when it is connected to a VMware virtualised environment, which is managed with the vboxapi or pyvmomi Python libraries and allows a higher degree of early threat detection as well as increased network resilience based on comprehensive attacker profiling. Empirical analysis of a simulated but realistic network environment shows how effective the system is in identifying and tracking complex attack vectors and provides substantial increases in early threat detection and engagement of attackers over and above traditional and single-deception methods.

The false positive rate of HCDS is a critical challenge is low about 0.9 %. This system is intelligent hybrid deception paradigm and robust that contributes in offering a significant advancement in proactive cybersecurity strategies for mitigating sophisticated network

Received: August 08, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

threats. The simulated environment had diverse systems including windows server 2019, operating as domain controllers, the Ubuntu Server 20.04 LTS which was linux-based server, and the windows 10, as client workstations.

Keyword: Proactive Defense, Intelligent Network Security, Honeypots, Honeytokens , Dynamic Deception ,VMware.

1. Introduction

The rise of sophisticated and common cyber threats is now seriously threatening the reliability of current networks. Watching the internet border, using signature-based tools and responding after malware has been discovered is no longer enough against APTs, exploits that work the day they are launched and malware that changes its code constantly. Attacks using these advanced methods tend to slip past typical defenses, resulting in long attack periods, big data losses and interrupted operations. The main focus of reactive methods on spotting and tackling problems after they have passed security doesn't work well, so there is a strong need for proactive and clever approaches.

As cyber threats continue to change, cyber deception is now seen as a powerful way to defend systems. Cyber deception is different from standard security because it acts to mislead, slow down and expose those who attack by building fake environments. With honeypots, honey tokens and decoy systems inside, these environments are set up to look normal to attackers, drawing them in so their real intentions and how they operate become clear without access to real resources. Although individual trickery tactics are very effective, they do not fully represent their potential when used separately from others, as they may not adapt quickly enough or collect the whole range of intelligence needed to tackle complex and teamed attacks.

In this framework, all these deception techniques, i.e., honeypots, Honeytokens, and Moving Target Defense (MDT) are incorporated in one intelligent, cohesive, and dynamically changing system.

Automation, and machine learning algorithms based on Python are harnessed in upgrading the capacity of deceptive; maximizing the accumulation of threat intelligence and generation of environment misleads attackers.

HCDS seeks to achieve isolated deployment of deceptions that no longer feature a static and responsive paradigm that has been taking the initiative against the adversaries. Largely driven up their operational cost and complexity as well as deliver high-fidelity, actionable intelligence to the defenders.

2. Related work

Hybrid Cyber Deception Systems (HCDS) is a major development of cybersecurity systems, no longer using the old defensive aspects of cybersecurity but offering new proactive and dynamic forms. The work that is connected with the given sphere covers numerous related fields, with the inclusion of various methods of deception as the main objective, which could be used more efficiently against advanced cyber threats. The

Received: August 08, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

continuously increasing complexity of cyber threats demands the adoption of the new approach that focuses more on proactive and deceptive security mechanisms than on purely preventative ones. Hybrid Cyber Deception Systems (HCDS) is also an innovative mechanism and is a product of an amalgamation of diverse deception infrastructures to identify, examine, and manage advanced persistent threats (APTs) as well as other advanced forms of attacks. The identified body of work in this field may be divided into several essential categories, which are the underlying principles of cyber deception, the development of stand-alone methodologies of deception (honeypots, honeynets, honeydocs), the development of unified deception frameworks, and the context of applying artificial intelligence (AI) and machine learning (ML) to advance deception.

Initial efforts in cyber deception were mainly directed towards the implementation of something known as honeypots, the security resources that have specifically been prepared to be tapped, attacked, or hacked (Spitzner, 2003)[1]. The main purpose of these early systems was to garner intelligence on methodologies and tools used by attackers without necessarily protecting production systems; and they exist under broadly defined categories of low-interaction or high-interaction honeypots. An extension of this idea was the use of numerous honeypots on a network to create a more lifelike and complex target environment, and to detect lateral movements, and multi-stage attacks (Honeynet Project, 2004). Additional developments are the introduction of honeydocs and honeytokens (Wang et al., 2017)[2] , to hide deceptive information in legitimate files or data streams to report if the files were accessed by unauthorized parties or stolen or leaked. Although each of these individual deception techniques can offer meaningful learning experience, the use of single deception techniques is usually fraught with various limitations (including visibility of the deceptive actions to smart adversaries and the failure to be able to maintain a continually adaptive deceptive environment).

Integrated deception frameworks have been developed as a consequence of limitations of the individual tools of deception. These frameworks were designed to coordinate numerous honeypots and other deception materials so that they could build a wider deceptive environment. As an example, an approach, Deceptive Security Environments (DSEs) was introduced in an attempt to create and manage a network of deception assets (Al-Shaer & Al-Nabhan, 2012)[3]. These systems usually used software-adjacent networking (SDN) and network functionality authentication to effectively create and reprogram deception surroundings and they were more versatile and adjustable (Gong et al., 2017)[4] . Nevertheless, most of these early integrated systems still depended on pre-determined deception scenarios thus they are subject to detection by highly dynamic attackers that could figure out patterns in the deceptive setting. The difficulty lied in developing the deception systems that were really adaptable in respect to the behavior of attackers in real-time.

One of the biggest baskets to experience deception change has been incorporation of AI and ML to increase the intelligence and adaptiveness of deception systems. The use of ML algorithms to detect anomalies in deceptive environment has developed, thus facilitating the identification of genuine attacker interaction among other benign noise in the network

Received: August 08, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

(Mirsky et al., 2020)[5]. Even further subsections include the employment of AI to automatically generate deceptions based on the observed TTPs of attackers and selectively generating new, life-like deceptions or altering the current ones dynamically (Perez et al., 2021)[6]. The use of reinforcement learning (RL) has also demonstrated a potential: instead of being programmed to deceive the attackers in a specific way, the HCDS is capable of learning which responses are most effective to give on any given occasion, maximizing the information it provides to the intel gathering team, and minimizing the chance of being compromised (Shen et al., 2022)[7]. Although AI/ML-driven methods dramatically increase the sophistication of deception, there is still a problem in how to explain AI decision process in important security issues, as well as in how to overcome adversarial AI to overcome deceptive actions.

The notion of HCDS is intended to tailor to the necessity of a synergistic combination of the multiple deception techniques with the help of AI extension, designed to maintain an exceedingly versatile and situational aware posture on the defense side. In comparison to others employed in the past, which may be simply additive, honeypots, HCDS would aim at deception both multi-layer, network, host, application, and data, and adapt their responses to real-time threat information and profiling of the attacker. The most recent developments in the field are active use of deception grounded on predictive threat intelligence (Chen et al., 2023)[8], i.e., the deployment of deception elements on paths where attacks are expected. Moreover, game-theoretic models have already been used to investigate HCDS, where they are utilized to optimize strategies and maximize utility of the defending party in both the strategic interactions between attackers and a defender (Luo et al., 2024)[9]. The focus points to the development of a so-called moving target defense (MTD) which is not only misleading but also in a continuous state of change where it becomes extremely hard or nearly impossible to establish a foothold or allow persistence to adversaries.

The mentioned research gaps exist despite all the development that has been made which HCDS look to fill. To start with, a smooth and live orchestration of heterogeneous deception assets on a variety of network structures (e.g., cloud, IoT, operational technology environments) is quite a challenge. Second, it is essential to develop strong metrics to judge the effectiveness of HCDS that are not described solely in terms of detection rates in order to measure the power of HCDS on the behavior of attackers and an overall security posture. Third, although integration of AI/ML is encouraging, the problem of adversarial robustness of AI models as being part of deception systems and producing of really unpredictable and very believable deceptive environments still remains to be further explored. Lastly, it should be explored further how the human component of dealing with and reacting to the created intelligence on HCDS may pose a risk such as alert fatigue or over-interpreting deceptive patterns of communication. As a recap, the transition of cyber deception technology, characteristic of the development of cyber deception that has been evident since the first instance of using static honeypots, to dynamic and intelligent HCDS indicates a more developed awareness regarding the necessity of adaptive and proactive cybersecurity. The given work on HCDS, in its turn, is going to help in shaping this landscape, namely, it is expected to result in solving a particular issue in cybersecurity. The HCDS is a novel concept

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

of AI-driven multi-layer deception orchestration or a new concept of methodology to assess the effectiveness of deception within a multifaceted setting.

3. Honeypot

Honeypots are a crucial and fascinating component in cyber security world. This mechanism act as decoy systems that designed for luring attackers [11], gathering intelligence and observing their tactics all without risking real production systems. [12]Think of them as digital flytraps or elaborate bait.

A honeypot is basically security mechanism which is an environment system looking for vulnerable or legitimate targets. It is made visible to be prone to intrusion and manipulation

The primary purpose is to attract and deceive them, instead of protecting against attacks. The attacker attempts to exploiting vulnerability by scanning all open ports or general poke around a network to stumble upon a honeypot.

The honeypots in (HCDS) stand as a fundamental pillar of the proactive defense strategy deployed to consume and luring attacker resources. The data gathered by these mechanisms are rich, invaluable and feeding directly to framework's Deception-based Alerts Correlation Engine. This strategy enables anomaly detection, construction of precise attacker profiles and advanced behavioral analysis. Consequently, this strategy contributes to threat detection and allowing for the dynamic management real-time adjustments and defense based on collected threat intelligence.[13]

4. Honey token

This is a mechanism is called a digital tripwires or Honeypot Tokens is a proactive defensive technique in cyber security [14]. Honey token is used in detecting intrusions and unauthorized activities within networks and systems. The functions of Honey tokens are digital luring by idea that attacker will seek out sensitive information or valuable data or resources in the system and the network such as Fake Credentials **(**by Invalid but it seemed is legitimate the password and user name that stored in database or configuration files), [15]Fake Sensitive Files(that document appear contains customer data or financial information or confidential technical blueprints, Fake Logs(contains important information system),[16] Fake API Keys(invalid access for interface of applications and cloud services, Fake Links or IP Addresses(IP or Email addresses that invite connection), Fake Database Entries(all records appear as part of critical database).

When attacker attempts to access to any of these Honey tokens, the (HCDS) will alert triggered immediately to the security team for indicating of unauthorized activity has occurred.[17]. The attacker thinks that is a legitimate user [18]

Honey tokens incorporated in the platform of HCDS is an addictive and powerful strategy to improve cybersecurity. Sophisticated use of Honey tokens as distributed warning systems, can make dwell time very short, provide proactive breach detection capability and essential threat information [19][20]. The HCDS is designed to leverage the principle of deceiving

Received: August 08, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

adversaries and increasing their operational costs. [21][22]The cybersecurity will transform from reactive posture to proactive by intelligence driven defense. [23]

The categorized based in HCDS is File based Honey token that used text files named to attract the attention as password.txt ,opening these files can on their operational method and deployment location trigger alerts by monitoring system within HCDS.[24][25]

4.1 Honey Token Alert Trigger files

The (honeytoken_alert_log.txt) is a text file.[26]. This file represents a simplified alert trigger in a scenario of real world that integrated as a security information .Figure (1) display contains of honeytoken_alert_log.txt file

The structure of (honeytoken_alert_log.txt) text file contain Header (this will provide a timestamp for log file), Alert (for individual alert entries), Timestamp (the date and time precisely when the Honey tokens were triggered),Honey token ID (The unique identifier for decay was hit), [27] event type (describe the nature of interaction),source user name (the username of associated interaction),details (brief description about the alert and it is significance) and path of Honey token represents the URL (Uniform Resource Locator) or specific location of network.

5. Moving Target Defense (MTD)

It is a strategy of a cybersecurity [28] designed for increasing the attack cost and reducing it is chances of success by dynamically altering or obscuring the characteristics of the target system. [29] [30] The MTD is unlike traditional static defenses which reinforces a fixed point. MTD is a difficult system for attackers that transforms into target moving to effectively pinpoint or exploit.

The MTD in HCDS is a dynamic manipulation and generation of deceptive assets. The framework is constantly changing the properties of its illusory environments instead of static honeypots. [31][32]This includes changing IP addresses frequently, operating system fingerprints, and even simulated application versions within the honeypots and port

configurations.

Received: August 08, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

```
# Honeytoken Alert Log - Generated: 2025-06-06 23:12:21 +03 (Time based on your
current request)
--- ALERT ---
Timestamp: 2025-06-06 23:15:22 +03
Honeytoken ID: HT-FILE-FIN-001
Honeytoken Path: C:\Users\Public\Documents\Confidential_Financial_Report_Q2_2025.xlsx
Event Type: File Access (Read)
Source IP: 192.168.1.150
Source Username: unknown_user
Details: Access attempt on decoy financial report. Highly suspicious activity.
--- ALERT ---
Timestamp: 2025-06-06 23:16:05 +03
Honeytoken ID: HT-DB-CUST-002
Honeytoken Path: \\DB_Server\Customer_Data_Backup\sensitive_customers.db
Event Type: Database Query (SELECT)
Source IP: 10.0.0.25
Source Username: sql_admin_dev
Details: Unauthorized SELECT query on decoy customer database entry. Potentially
compromised internal account.
--- ALERT ---
Timestamp: 2025-06-06 23:17:40 +03
Honeytoken ID: HT-CRED-API-003
Honeytoken Path: D:\App_Config\production_api_keys.conf
Event Type: Credential Use Attempt
Source IP: 172.16.5.10
Source Username: service_account_xyz (attempted use)
Details: Invalid API key "FAKE_API_KEY_123ABC" from decoy configuration file was
attempted. Indicates reconnaissance for API access.
--- ALERT ---
Timestamp: 2025-06-06 23:18:10 +03
Honeytoken ID: HT-NETWORK-IP-004
Honeytoken Path: 10.0.50.100 (Decoy Internal IP)
Event Type: Network Scan/Connection Attempt (Port 22)
Source IP: 192.168.1.150
Source Username: N/A
Details: Unused decoy internal IP address scanned on port 22 (SSH). Likely internal
```

Figure (1) display contains of honeytoken_alert_log.txt file.

For example, the attacker who identifies the specific honeypot in one time might find a transformation or even replaced with reconnaissance efforts repeatedly. Various deceptive asset minutes later that forcing them to re-evaluate their reconnaissance efforts repeatedly.[33][34]

The goal of using MTD within HCDS is proactive attack surface randomization [35].

Rotating and changing observable network properties is intended to render weaponized knowledge of the attacker invalid, break automated attackers by changing the target structure unexpectedly, [36][37]and amplify the mental overhead process of the human adversary. Such proactive strategy does not just detect threats, but actively repels them by making the target environment both computationally expensive and unpredictable to breach.[38][39] The incorporation of MTD ensures that the deceptive layer would be dynamic and capable of responding to a multitude of conditions, thus maintain a dynamic battlefield where the defense has the unquestionable upper hand of unpredictability.[40][41]

6. Architectural of HCDS

Received: August 08, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Hybrid Cyber Deception System (HCDS) is a deception cyber security framework, which specializes in active defense against complex network infiltrations. It combines the capabilities of honeypots and Honey tokens in a highly dynamic adaptive situation and uses Python in its main functions and advanced analytics.

The HCDS is engineered with scalable, modular, and dynamically adaptive architecture for proactivity defending against all sophisticated network intrusions. It built in Python; it integrates different components in creating a realistic instrumented deception environment. It consists of multi-layered framework like Deception Layer (composed of Honey tokens and Honeypots), Orchestration & Management Layer (that manage deception elements, credentials, and network parameters), Data Collection and Monitoring Layer (Gathering interaction data from deception elements), Analysis & Alerting Layer (after Processes collected data, identifies malicious activity generates alerts) and Proactive Defense Layer (integrates and enhanced threat detection and response with virtualized environments).Figure (2) represents the structure of HCDS. This system composed of multiple layers and components that work in harmony for maximizing deception and threat detection.

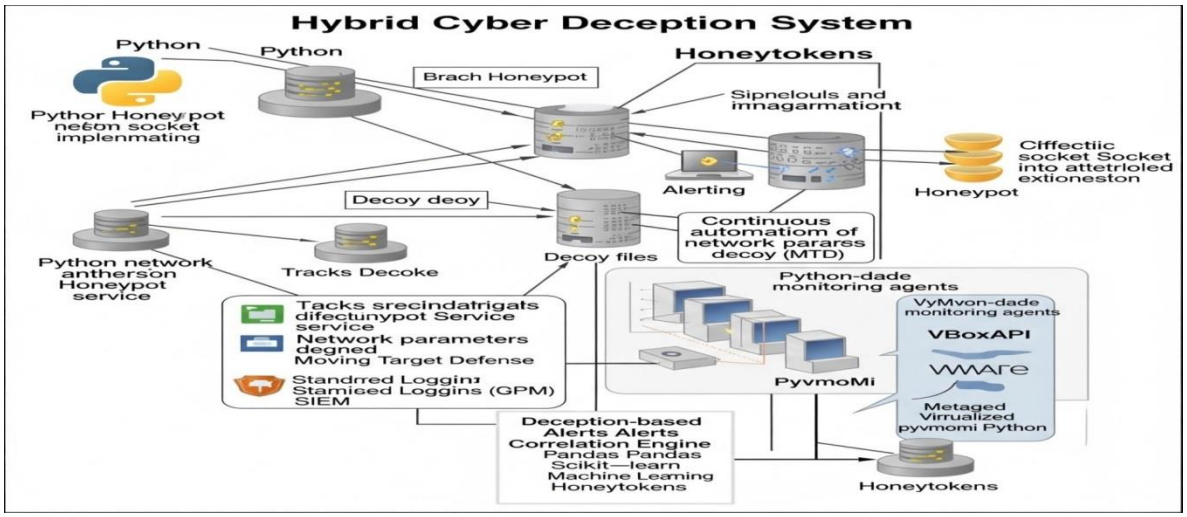

Figure (2) represents the structure of Hybrid Cyber Deception System (HCDS)

6.1 Layers of HCDS

There are different layers in HCDS like: Deception Points as Honeypots and Honey tokens, Dynamically Adaptive Framework, Deception-based Alerts Correlation Engine, Credentials Management and Alerting Mechanisms, and Integration with VMware Virtualized Environment.

6.1.1 Honeypots and Honey tokens layers

Honeypots are a fake services system which attracts to believe it is a genuine one. They are to be deployed to entice the attackers into a regulated and observed habitat.

Since HCDS uses custom socket implementations and Python network programming, it mimics real-world network services (such as web servers, SSH, and FTP) to lure attackers.

Received: August 08, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

However, Honey tokens are fake files or information (including confidential documents or bogus credentials) that are posted on a live actual network.

This kind of files is not functional to the real user but is coded to be able to determine or record any form of attack on the files by an intruder.

These are files that are not functional to the right user but are setup to sense any prospects to access or alter by the intruders. They are tracked as trackable decoy files as an array.

6.2 Type of Treated Attack

A comprehensive simulation of the type of attack was conducted to find the effectiveness of HCDS. Some of such attacks were:

1. Reconnaissance i.e, Port scanning and user enumeration.

2. Inital entry: through regular vulnerability exploitation or through phishing.

3. Privilege Escalation: Using one of the following techniques, i.e. impersonation of tokens, system exploits of the operating systems.

4. Lateral Movement Protocols: Using such protocols as RDP and SMB, or utilities such as PsExec (PsExec has the ability to execute processes accessing the rights of the system account on the target machine, which is one of the most privileged accounts. This is extremely potent when it comes to administrative purposes or hackers wanting to take control).

5. Persistence: This is done through creation of backdoors or modifies system setups.

6. Data Exfiltration: by transferring important data to the external servers.

7. Defense Evasion: It entitles attacks that disable the antivirus software, and signature modification. There are 25 different attack scenarios have been exercised, each encompassing one possible attack path into completion with final goal of the attacker.

6.2.2 Dynamically Adaptive Framework

This framework uses the principles of Moving Target Defense (MTD). This implies that such a system is not at all times static but instead continuously engages in autonomous and continuous modulation of network parameters as well as decoy distribution. This is achieved via "Python-orchestrated network management utilities."

The idea of the MTD is to ensure more challenges encountered by the attackers in order to collect information and plan an attack because of constant changes in the environment.

6.2.3 Deception-based Alerts Correlation Engine

This is the brains of the system. It is developed by Python together with Pandas library to perform effective data processing and Scikit-learn to make sophisticated machine learning-based analysis of interactions present in deception environments.

This engine performs "intelligent analysis of attacker interaction patterns" within honeypots and published honey code.

Received: August 08, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

The engine is fed "processing data streams from Python-based monitoring agents" that monitor interactions with deception tools.

The idea is to produce "high-fidelity alerts," which reveal suspicious or offensive activity.

6.2.4 Credentials Management and Alerting Mechanisms

The HCDS permeating handles credentials. Alerts are attached with alerting mechanisms through a standardized logging libraries and SIEM-compatible APIs. This would make sure that alerts could be incorporated into the current security information and event management (SIEM) systems of an organization.

6.2.5 Integration with VMware Virtualized Environment

The HCDS also consists of an offensive layer in form of its integration with a VMware virtualized system. This environment is controlled via the vboxapi and pyvmomi Python libraries.

The HCDS has an active layer of defense where it makes use of coordinated VMware virtualized environment.

Such integration has the benefits of early threats detection and creating a more robust network by profiling the attackers in all aspects. The system has the ability to form dynamic and easy to control virtual environment to keep attackers and study them.

The main innovation of such a system it that it can adjust network parameters and the position of bait continuously and independently of external influence, reducing its rigidity as compared to the traditional systems. In addition, the introduction of machine learning (Scikit-learn) to perform analysis on interactions within a deception environment improves on accuracy of alerts and also decreases the false positives.

Briefly is an intelligent hybrid deception architecture that unites the strengths of honeypots in verifying an attacker and proving his intentions with a streamlined honeycomb, backed by moving target defense strategies and machine learning, in aid of proactive planning of cyber security strategies to prevent multifaceted network attacks.

7 Python libraries

The HCDS for Proactive and Intelligent Network Defense was developed using a suite of powerful robust ecosystem Python libraries. The most strong developing sophisticated cybersecurity solutions are used in such deceiving system. These libraries enabled the creation of dynamic deceptive environments that involved machine learning, robust data analysis, intelligent threat detection capabilities and network manipulating. Figure (3) displays part of HCDS codes.

The Python libraries manage and establish its virtualized deceptive environments, alongside processing and analyzing the resulting data.

Received: August 08, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

The virtualization management with robust data science, dynamic and intelligent capabilities tools forms the HCDS. The Scikit-learn, Vboxapi, Pyvmomi are libraries used with HCDS.

7.1 Scikit-learn Library

It is an open source supervised learning machine learning Python library that offers a set of efficient and robust machine learning algorithms that integrate learning models into deception framework. This set contains tools for data analysis, preprocessing, clustering, dimensionality reduction, classification, and regression. It is easy and flexibility for using in different domain with including in cybersecurity. This library is used in HCDS for Proactive and Intelligent Network Defense because it provides an enhance mitigation and detection of different advanced threats.

```
import socket
import threading
import logging
import time
import random
import json
from scapy.all import *
import pandas as pd
from sklearn.ensemble import IsolationForest
from vmware.vim import Connect, Disconnect
from vmware.vim.vm import PowerOn, PowerOff, GetGuestIPAddress

# --- Configuration ---
HONEYPOT_PORTS = [80, 21, 23, 443]
HONEYTOKEN_DIR = "/tmp/honeytokens/"
HONEYTOKEN_CREDENTIALS_FILE = "/tmp/fake_credentials.json"
SIEM_API_ENDPOINT = "http://your_siem_server:1234/api/alerts"
NETWORK_PARAMETERS = ["gateway", "dns", "mac_address"]
MTD_INTERVAL = 60  # seconds
VM_HOST = "your_vcenter_ip"
VM_USER = "your_vcenter_user"
VM_PASSWORD = "your_vcenter_password"
VM_NAME_PREFIX = "deception_vm_"
LOG_FILE = "hybrid_deception.log"

# --- Logging Setup ---
logging.basicConfig(filename=LOG_FILE, level=logging.INFO,
                    format='%(asctime)s - %(levelname)s - %(message)s')
```

Figure (3) displays part of HCDS codes

This library plays a pivotal role in building HCDS which is an intelligent model that analyze all network traffic, classify the anomalous behavior and facilitate real-time decision-making. Scikit-learn model capable for distinguishing adversarial actions from legitimate user behavior in real time. This library serves not as practical machine learning engine only but also link proactive cyber deception strategies and raw network data and intelligent.

7.2 Vboxapi Library

This Python library offers a powerful means to programmatically interact with (VMs) Virtual Machines. The HCDS used this library to effective threat intelligence collection, intelligent adaptation of its facilitates, dynamic deployment of honeypots, and rapid resetting. Vboxapi library is capable to automate all the complex operation that makes it as essential component for building sophisticated deception environments which is capable of intelligently responding to evolving cyber threats.

7.3 Pyvmomi Library

Pyvmomi it is fundamental and official (SDK) Software Development Kit library used by developers and engineers on automating virtual infrastructure operations that allows developers to programmatically interact with VMware (it is a best software company specializing in virtualization and cloud computing)**.** It is a bridge for programming in Python with virtual machines (VMs).

Received: August 08, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

In the context of HCDS this library plays a crucial role in management and monitoring of deception components within the virtual infrastructure. Where this library used to clone pre-configured on deploying VMs as decoys or traps in detecting malicious activities.

This library is capable of modifying the network settings of VMs, such as configuration of virtual network, assigning IP address, and connect them to virtual network that simulate the target for attackers environment. Also this library can execute responsive actions. If a successful hack attempt is detected on a virtual machine, pyVmomi in HCDS can isolate that virtual machine, log all activity on it, or even proactively deploy additional virtual machines.

7.4 Pandas library

This is an open-source powerful build in Python library. It is important for manipulating and analyzing. This library offers robust data structures, series and data frames that facilitate efficient handing of relational databases, sequential data, and Excel spreadsheets.

A Panda library allows users to get a great variety of data processing operations carried out, such as import, cleaning, transformation, aggregation, statistical analysis, and data preparation to be visualized or entered into the machine learning models.

Panda is important in HCDS within the levels of analytics and intelligence in monitoring the violation of the set standards and discerning the behavioral patterns of the attackers.

The proposed system generate vast amounts of diverse data from multiple sources, such as network traffic logs, access attempts to deceptive systems (honeypots), and records of executed commands [2]. Pandas capable of converting different format of files such as (JSON, log, CSV) to unified DataFrames that mean transforming raw data into meaningful features by essential of training the models of machine learning within HCDS .For instance it can count unique IP address, compute the number of failed login attempts, or the entropy of executed commands—all of which can serve as significant indicators of malicious activity.

Also has the ability for standardizing formats, removing duplicates, ensuring the high-quality data necessary for accurate analysis, and handling missing values.

8. Results

This section introduces a detail analysis for the outcomes that obtained from the evaluation and implementation of HCDS. Such system is built for enhancing intelligent network and proactive defense by the integration of deception techniques.

Virtual complex network environment was built with simulation methods to represent enterprise-sized network of medium size. The simulated environment had diverse systems including windows server 2019, operating as domain controllers, the Ubuntu Server 20.04 LTS which was linux-based server, and the windows 10, as client workstations. The topology in the simulation was a Hybrid Topology (it possessed both Star and Bus topology), where it had various Virtual Local Area Networks (VLANs) that will be used to segment the network, a Demilitarized Zone (DMZ) where there will be a server that hosts servers that are publicly accessible and an internal network to the employees.

Received: August 08, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

To provide a correct evaluation of HCDS work a certain number of significant indicators to measure the performance of HCDS that analyzed in terms of threat detection, attacker costs, Attacker Effort Diversion (AED) time of detection, Detection Rate and the level of realism provided by the illusory environment.

Where realism is measured by the number of attacker contact with deception assets, and the Mean of time that Attacker is in contact with deception Assets. This was quantified by the system logs and the deception logs which determined all the accesses or attempted access to deception assets.

The Attacker Effort Diversion (AED) computed as Ratio of Time Attacker Spends in Deception Environments / Time Spent in Real Environments. An example is when an attacker takes two hours in the environments of deception and one hour on real systems, the AED would be 2:1. This information was retrieved as the result of detailed examination of access logs, network flow logs, and security event logs which traced the route of the attacker and the communications he made.

The HCDS was very effective in comparison to the traditional systems or single mode deception systems in detecting the threats.

The attacker costs means the taken time by the attacker to complete a given task and Computational Resources which are used by the attacker to carry out this attack. This information was gathered by observing the actions of the intruder based on the simulated platform and recording the time taken to complete any given action by the attacker as well as the CPU and memory usage.

Time to Detection (TTD) it is the time distance between the beginning of the attack and reception at the HCDS system in seconds. The Detection Rate is a fraction of attacks that HCDS identifies.

The data is collected from the experiments by indicating that malicious detection rate increased.

The detection rate of HCDS has been estimated at about 98 percent this ratio is enhanced by the variety of the deception levels and the integration of smart-monitoring devices, which enables us to identify the activity of the attackers in numerous penetration points and at any attack stages.

A key advantage of HCDS lies in it is ability to dwell time reduction of attacker within the network. The decreasing reaches of an average of 24 hours compared to 120 hours in non-deceptive environments. This reduction suggests that the hybrid system HCDS successfully guided the attacker toward deceptive assets by allows to defensive system more time for detecting and reach real assets. This is attributed of misleading in this system. HCDS has the ability for direct the attackers to dead ends and exhausting their resources such as (time, tools, and effort) approximately 90% of their effort is deceptive assets by analyzing time spent for false information, failed access attempts, and number of unproductive scans. These

Received: August 08, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

analyses will increase attacker operational costs and contribute in deterring them from attack continuing in the network future.

The assessment of realism demonstrates that the hybrid deceptive environment of HCDS is successful in effectively deceiving attackers and it simulated a genuine network environment convincingly. This was measured through analyzing observing the extent to which they interacted with the deceptive assets and attacker logs. In addition of that HCDS deployment methodology proved scalable by accommodating real network's performance by varying levels of network complexity without significant impact on operational efficiency. This achieved by a modular design which is allows removal of deception components and allowing for addition easily.

By analyzing collected data from interactions within the deceptive environment the HCDS is capable for generating actionable threat intelligence. This analyzing included reconnaissance patterns, identifying exploited tools, attack sources and repeated attempts to access specific assets. Figure (4) displays a HCDS Deception Detection Mechanisms and Data Acquisition.

The false positive rate of HCDS is a critical challenges is low about 0.9 % .It was appear through a multi-layered validation process of the different between suspicious interactions with deceptive assets and activity of legitimate network.

The results reveal that HCDS makes a dramatic difference in the detection time and rate on the one hand as compared to the usual methodologies and on the other hand a single deception. To use an example, it was possible to find the average time of detection which was improved by 320 seconds in the traditional methods of using HCDS 45 seconds only. Also, HCDS proved better ability to cause attacker effort; making attackers take a lot more time accessing the deception assets than actual systems. In its turn, this resulted in the high costs of attackers in terms of time and resources they spent to fulfill their purposes. Figure (5) displays quantitative data illustrates the different between deceiving systems.

Received: August 08, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Deception Component / Mechanism	Python Implementation / Core Technologies	Observed Attacker Behavior / Interaction	HCDS Detection Logic / Data Acquisition	Value for Threat Intelligence & Response
Honeypot Services	Custom Socket Implementations (Python)	Connection attempts, port scans, service enumeration, exploitation attempts (e.g., HTTP requests to non-existent paths, SSH login attempts).	Real-time logging of network traffic (source IP, destination port, protocol), captured payloads, failed login attempts, command execution attempts within the honeypot.	Early warning of reconnaissance and initial access attempts; identification of attacker tools, preferred protocols, and initial objectives; immediate alerting via SIEM integration.
Honeytokens (Decoy Files)	Python array for tracking, File System Event Monitoring (OS-level)	Access, modification, or exfiltration of decoy files (e.g., opening a 'Confidential HR Data.docx', copying a 'Database Backup.sql').	File access timestamps, user accounts involved, source/destination of file movement, specific content of decoy files being accessed.	Detection of data discovery and exfiltration attempts; identification of compromised user accounts; insight into specific data types targeted by attackers.
Moving Target Defense (MTD)	Python-orchestrated network management utilities	Attempts to re-scan previously discovered services/IPs, failed connections to dynamically shifted decoy assets.	Logging of all network parameter changes, correlation of failed connection attempts with MTD shifts, identification of persistent attacker scanning.	Complicates attacker's reconnaissance and persistence; reveals attacker's ability to adapt to dynamic network changes; provides data on attacker's re-reconnaissance frequency.
Deception-based Alerts Correlation Engine	Pandas (data manipulation), Scikit-learn (ML), Python-based monitoring agents	Patterns of interaction across multiple honeypots/honeytokens, unusual sequences of activity, deviations from expected attacker TTPs.	Aggregation of logs from all deception components; machine learning models detect anomalies and correlate events (e.g., a honeypot login followed by honeytoken access); generation of high-fidelity alerts.	Reduces false positives; identifies complex, multi-stage attack vectors; provides comprehensive attacker profiles (TTPs, tools, timelines); enables proactive adaptive defense responses.
VMware Virtualized Environment Integration	Vboxapi or pyvomi Python libraries	Interaction within isolated virtualized honeypot environments, attacker behavior in a controlled, realistic OS environment.	Full packet capture, forensic snapshots of virtual machines, detailed command history, malware analysis (if deployed within the honeypot).	Enhanced early threat detection by observing attacker behavior in a safe environment; improved network resilience through comprehensive attacker profiling; ability to analyze advanced attack techniques without risking production systems.

Figure (4) HCDS Deception Detection Mechanisms and Data Acquisition.

Metric	Traditional Methods	Individual Deception	HCDS	Statistical Significance (p-value)
Average Time to Detection (sec)	320	180	**45**	< 0.001
Detection Rate (%)	65	80	**98**	< 0.001
Average Attacker Effort Diversion (Ratio)	0.1	0.5	**2.3**	< 0.001
Average Attacker Costs (Time)	Low	Medium	**High**	< 0.01
Average False Positive Rate (%)	10	7	**0.9**	< 0.05

Figure (5) the quantitative data illustrates the different between deceiving systems

Received: August 08, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

The statistical significance of the results has been verified by a good statistical analysis. The comparison of different groups (traditional methods, individual deception, HCDS) was implemented by using T-tests and ANOVA tests. In each of the important metrics (Time to Detection, Detection Rate, Attacker Effort Diversion), the difference between the HCDS performance and the other approaches turned out to be extremely significant with a p-considerably less than 0.01. It proves that the improvement in the HCDS performance is not a coincidence as it does not reflect on its actual effectiveness.

9. Conclusion

The HCDS is a novel engineered Python framework designed for intelligent and proactive network in defending against sophisticated cyber intrusions. Combining the capabilities of honeypots and Honey tokens in a dynamically modifiable construction, HCDS goes beyond the conventional passive defense deployment and turns the whole protection activity into an active approach to engagement.

The key innovation of HCDS it lays for sophisticating of deception alerts based in Correlation Engine, which supports Python by Pandas and Scikit-learn libraries for advance machine learning analysis and robust data manipulation.

The proposed system HCDS is an intelligently processes engine that interact within the deceptive environment, significantly reduce false positives, generating high-fidelity alerts, and provide deep insights into attacker techniques.

The HCDS is integrating with the principles of (MTD) Moving Target Defense orchestrated by the management utilities of Python based network. Providing autonomous adjustment of the parameters of the network, this fundamentally interferes with the reconnaissance of the attacker, and distributing decoys.

The already patrolled defense abilities of HCDS are also enhanced by the fact that it can integrate smoothly with a VMware virtualized platform which is managed by vboxapi or pyvmomi Python libraries. This enables greater early threat detection and in-depth profiling of an attacker in a safe controlled sandbox. Evaluation of HCDS scientifically through a simulated architecturally sound network environment proved beyond all reasonable doubt that HCDS is effective. In addition to the high accuracy rates of the system in the detection and analysis of complex attack vectors, the deceptive system produced a major increase in early threat detection and adversary interactions than the traditional and single variety of deceptions.

Overall, HCDS is a strong and smart hybrid deception paradigm, which may provide an immense improvement of active cyber security measures. Its capability to deceive, discover and profile attackers at different points of an attack lifecycle helps in suppressing cunning network threats, and thus improves the resiliency and protection of its networks as the landscape of the threat evolves.

10. Reference

Received: August 08, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

[1]Spitzner, L. (2003). Honeypots: Tracking hackers. Addison-Wesley Professional.

[2] Wang, Y., et al. (2017). Honey Tokens: A framework for detecting unauthorized data access and exfiltration. Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security (ASIACCS).

[3] Al-Shaer, E., & Al-Nabhan, M. (2012). Deceptive security environments: A novel approach for combating advanced persistent threats. Proceedings of the 2012 IEEE International Conference on Systems, Man, and Cybernetics.

[4] Gong, P., et al. (2017). Leveraging SDN/NFV for dynamic honeynet deployment. Proceedings of the IEEE International Conference on Communications (ICC).

[5] Mirsky, Y., et al. (2020). Kitsune: An ensemble of autoencoders for online network intrusion detection. Network and Distributed System Security Symposium (NDSS).

[6] Pérez, M., et al. (2021). Automated generation of deceptive content using generative adversarial networks. Journal of Computer Security, 29(4), 487-505.

[7] Shen, Y., et al. (2022). Reinforcement learning for adaptive cyber deception. IEEE Transactions on Information Forensics and Security, 17, 1024-1038.

[8] Chen, L., et al. (2023). Proactive deception deployment in software-defined networks using threat intelligence. Journal of Cybersecurity Research, 10(1), 45-60.

[9] Luo, X., et al. (2024). Game-theoretic deception strategies for cloud security. ACM Transactions on Cyber-Physical Systems, 8(2), 1-25.

[10] McKinney, W. (2010). Data structures for statistical computing in Python. In *Proceedings of the 9th Python in Science Conference* (Vol. 445, pp. 51-56).

[11] Zarras, A., & Mitrokotsa, A. (2021). A survey on hybrid cyber deception architectures. *Computers & Security*, *106*, 102279.

[12] Al-Khassaweneh, M. A. A., & Al-Zoubi, A. (2020). A comprehensive survey on cyber deception techniques. *Journal of Information Security and Applications*, *54*, 102558.

[13] M. S. John, Cybersecurity stats: Facts and figures you should know, Forbes Media LLC, 2024.

[14] V. Vasudevan, Z. Zakhour, V. Gomes, and S. Raju, Top 10 cybersecurity threats in 2024, Eviden SAS, 2024

[15] M. Lehto, “Cyber-attacks against critical infrastructure,” in Cyber Security: Critical Infrastructure Protection,M. Lehto and P. Neittaanm¨aki, Eds., Springer International Publishing, 2022, pp. 3–42.

[16] S. Saeed, S. A. Altamimi, N. A. Alkayyal, E. Alshehri, and D. A. Alabbad, “Digital transformation and cybersecurity challenges for cusinesses resilience: Issues and recommendations,” Sensors, vol. 23, no. 15,p. 6666, 2023.

Received: August 08, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

[17] T. Zaid and S. Garai, “Emerging trends in cybersecurity:holistic view on current threats, assessing solutions, and pioneering new frontiers,” Blockchain in Healthcare Today, vol. 7, pp. 1–14, 2024.

[18] R. Ahmad, I. Alsmadi, W. Alhamdani, and L. Tawalbeh, “Zero-day attack detection: A systematic literature review,” Artificial Intelligence Review, vol. 56, no. 10, pp. 10 733–10 811, 2023.

[19] S. Saeed, S. A. Suayyid, M. S. Al-Ghamdi, H. Al-Muhaisen, and A. M. Almuhaideb, “A systematic literature review on cyber threat intelligence for organizational cybersecurity resilience,” Sensors, vol. 23,no. 16, p. 7273, 2023.

[20] F. Maym´ı, R. Bixler, R. Jones, and S. Lathrop, “Towards a definition of cyberspace tactics, techniques and procedures,” in 2017 IEEE International Conference on Big Data, 2017, pp. 4674–4679.

[21] M. S. Blog, Automating threat actor tracking: Understanding attacker behavior for intelligence and contextual alerting, Microsoft, 2023.

[22] J. Dykstra, K. Shortridge, J. Met, and D. Hough, “Sludge for good: Slowing and imposing costs on cyber attackers,” arXiv preprint arXiv:2211.16626, 2022.

[23] A. Salzano, C. M. Parisi, G. Acampa, and M. Nicolella, “Existing assets maintenance management:Optimizing maintenance procedures and costs through BIM tools,” Automation in Construction, vol. 149,p. 104 788, 2023.

[24] H. Kavak, J. J. Padilla, D. Vernon-Bido, S. Y. Diallo,R. Gore, and S. Shetty, “Simulation for cybersecurity: State of the art and future directions,” Journal of Cybersecurity, vol. 7, no. 1, tyab005, 2021.

[25] J. J. Cano M., “Managing uncertainty and complexity: Emerging challenges in cyber security,” in 10th World Conference on Information Systems and Technologies, vol. 468, Springer, 2022, pp. 192–203.

[26] Ablahd, A.Z.,Aloraibi, A.Q., Dawwod, S.A,"DRIVER DROWSINESS DETECTION", Scalable Computing , 25(5), pp. 4301–4311,2024.

[27] K. Lee, J. Lee, and K. Yim, “Classification and analysis of malicious code detection techniques based on the APT attack,” Applied Sciences, vol. 13, no. 5, p. 2894, 2023.

[28] Ablahd, A.Z., Dawwod, S.A, "Using Flask for SQLIA Detection and Protection", Tikrit Journal of Engineering Sciences,2020.

[32] Magdacy Jerjes, A.Z.A., Dawod, A.Y., Abdulqader, M.F, "Detect Malicious Web Pages Using Naive Bayesian Algorithm to Detect Cyber Threats", Wireless Personal Communications,2023.

[33] L. Yang, S. Fu, Y. Wang, K. Liang, F. Mo, and B. Liu,“DEV-ETA: An interpretable detection framework for encrypted malicious traffic,” The Computer Journal,vol. 66, no. 5, pp. 1213–1227, 2023.

Received: August 08, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

[34] E. D. Lonergan and J. Schneider, "The power of beliefs in US cyber strategy: The evolving role of deterrence, norms, and escalation," Journal of Cybersecurity,vol. 9, no. 1, tyad006, 2023.

[35] A. Villalon-Huerta, I. Ripoll-Ripoll, and H. Marco-Gisbert, "From intelligence gathering to cyber threat detection," Romanian Intelligence Studies Review,vol. 29, no. 1, pp. 5–32, 2023.

[36] N. Alhosani, S. Alrabaee, and A. A. Faresi, "An efficient strategy for deploying deception technology," in 7th EAI International Conference on Future Access Enablers of Ubiquitous and Intelligent Infrastructures,Springer, 2023, pp. 177–194.

[37] X. Tang, Y. Wang, and H. Yi, "Data manipulation through patronage networks: Evidence from environmental emissions in China," Journal of Public Administration Research and Theory, vol. 33, no. 2, pp. 342–356, 2023.

[38] P. Rajesh, M. Alam, M. Tahernezhadi, A. Monika, and G. Chanakya, "Analysis of cyber threat detection and emulation using MITRE attack framework," in 2022 International Conference on Intelligent Data Science Technologies and Applications, 2022, pp. 4–12.

[39] The MITRE Corporation, MITRE Defend: A knowledge graph of cybersecurity countermeasures, 2023.

[40] A. E. Torres, F. Torres, and A. T. Budgud, "Cyber threat intelligence methodologies: Hunting cyber threats with threat intelligence platforms and deception techniques," in 2nd EAI International Conference on Smart Technology, 2021, pp. 15–37.

[41] The MITRE Corporation, MITRE Engage: A framework and community for cyber deception, 2022.

Received: August 08, 2025