JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Elevating Network Security to a High-Level Protection Paradigm Using Decoy system

Shaik Sadaf Tabassum
B.Tech (IT) Student, Department of IT
G.Narayanamma Institute Of Technology And Science,
Hyderabad., India

Abstract: Internet security remains a critical concern in our daily lives, and despite decades of research and experience, creating completely secure computer systems remains a significant challenge. Achieving absolute security is an elusive goal due to various factors, and accurately measuring the security of computer systems adds another layer of complexity to the problem. Confidential information can exist in two primary states within a network infrastructure. It may be stored on physical media, such as hard drives or memory, or it may be in transit across the network in the form of data packets. Both of these states pose potential vulnerabilities, exposing opportunities for attacks from internal network users and external threats on the Internet. Our primary focus is on the latter—network security issues. While many acknowledge the importance of security, there is often reluctance to engage with it when it becomes intrusive. It is crucial to design systems and networks in a way that doesn't constantly disrupt the user experience with reminders of the security measures in place. Security is a collective responsibility, and achieving it requires the cooperation of everyone, guided by intelligent policies and consistent practices. A progressive solution gaining traction in the realm of network security and computer incident response is the implementation of "Decoy Systems." Also referred to as deception systems, honeypots, or tar pits, these are simulated components strategically deployed to lure unauthorized users. They present apparent system vulnerabilities while actively hindering unauthorized access to network information systems. In this paradigm, a security module has been integrated using authentic software to reshape the network's security landscape. The emphasis is on creating an environment where potential threats are enticed by decoy systems, diverting their attention from actual sensitive information. This proactive approach helps organizations stay ahead of potential breaches by not only fortifying their defenses but also actively engaging with and deterring potential attackers. It's a strategic blend of authentic security measures and deceptive elements, working cohesively to safeguard the integrity of the network.

Keywords **- Decoy system, network security, phony components**

I. INTRODUCTION

A contemporary approach gaining widespread acceptance involves the integration of decoy systems into production environments to enhance defensive network security measures. These compromised decoy systems provide a range of features that prove invaluable for intelligence data gathering and incident response. They contribute significantly to understanding the identity of attackers, the methods employed for unauthorized access, and the outcomes of the illicit activities. Such insights are crucial for a more informed incident response and may be utilized for potential legal actions against the perpetrators [1].

This research focuses on examining decoys featuring dynamic attributes, specifically exploring the influence of integrating blockchain technologies with decoys on an organization's information security. The study delves into the process of investigating cybercrime, a critical consideration given that the majority of cybercrimes go unnoticed until the attacker has successfully accessed sensitive data. The systematic analysis of pertinent literature, with a concentrated effort on evaluating the capabilities of decoy and blockchain technologies, serves as the foundation for identifying the primary benefits associated with decoys that leverage blockchain technology [2].

The escalating threat of cyber-attacks on a global scale underscores the growing necessity for advanced cyber defense techniques. Cyber adversaries frequently rely on the direct observation of cyber environments [3]. Consequently, this research seeks to contribute insights into how the integration of dynamic decoys, particularly those incorporating blockchain technology, can play a pivotal role in fortifying defenses against the evolving landscape of cyber threats.

Internet security is increasing in importance as more and more business is conducted there. Yet, despite decades of research and experience, we are still unable to make secure computer systems or even measure their security. As a result, exploitation of newly discovered vulnerabilities often catches us by surprise. Exploit automation and massive global scanning for vulnerabilities enable adversaries to compromise computer systems shortly after vulnerabilities become known. The project implements Java, and based on client and server technology the process is done. We have also deployed the cryptographic procedure for maintenance of security. In this concept, the alternate path selecting is main factor for eliminating the intruder in the network and also utilizes the network in better manner. we proposed and instantiated two decoy evasion attacks, namely, network traffic fingerprinting and system fingerprinting attacks, which allow sophisticated adversaries to circumvent existing decoy-based deception. In the due time a model developed a seamless real-time replay framework as a countermeasure to defeat identified evasion attacks [4].

Integrating decoy systems into an existing security framework introduces substantial value by augmenting the level of security within the network infrastructure. The data gleaned from a well-implemented decoy system often surpasses the utility of intrusion detection systems (IDS) data, primarily due to a notable reduction in both false positive and false negative alerts. Functioning as "set

and forget" IDS sensors, decoy systems are comprised of singular systems or networks of devices dedicated to capturing unauthorized activities. Every packet entering or leaving a decoy system is inherently treated as suspicious, streamlining the data capture and analysis process and furnishing valuable insights into the motives of potential attackers.

Many production networks and servers lag behind in implementing the latest Microsoft Windows security patches or suffer from configuration errors well-known to hackers. Exploiting these common vulnerabilities, attackers can freely utilize readily available tools to scan multiple networks in search of easily exploitable entry points. The strategic deployment of decoy systems leverages these prevalent issues, turning them into enticing opportunities for detection and response. Rather than merely serving as deterrents, decoy systems are designed to actively engage and impede hackers, thereby enhancing the overall resilience of the security infrastructure.

Here we propose software-based decoy system that aims to deceive insiders, to detect the exfiltration of proprietary source code. The proposed system generates believable Java source code that appear to an adversary to be entirely valuable proprietary software. Bogus software is generated iteratively using code obfuscation techniques to transform original software using various transformation methods [13][14] [15].

II. LITERATURE REVIEW

A. SECURITY SYSTEM:

Decoy system designed to look like a legitimate system an intruder will want to break into while, unbeknownst to the intruder, they are being covertly observed. Honey pots are effective precisely because attackers do not know if they are there and where they will be. However, honey pots are also a controversial technique; they essential bait and capture intruders skirting the fine line between keeping attackers out of a network versus inviting them in. A honey pot is a program that takes the appearance of an attractive service, set of services, an entire operating system, or even an entire network, but is in reality a tightly sealed compartment built to lure and contain an attacker (a sandbox where intruders cannot harm production systems or data), effectively shunting an intruder safely from production systems for covert analysis. Like a hidden surveillance camera, a honey pot monitors and logs every action an attacker makes including access attempts, keystrokes, and files accessed and modified, and processes executed.

Ken Wong et.al [5] We present an approach to tracking the behavior of an attacker on a decoy system, where the decoy communicates with the real system only through low energy blue tooth. The result is a low-cost solution that does not interrupt the live system, while limiting potential damage. The attacker has no way to detect that they are being monitored, while their actions are being logged for further investigation. The system has been physically implemented using Raspberry PI and Arduino boards to replicate practical performance.

Hadeal Abdulaziz et.al [6] To diagnose and evaluate a patient, the healthcare professionals need to access the electronic medical record (EMR) of the patient, which might contain huge multimedia big data including X-rays, ultrasounds, CT scans, and MRI reports. For efficient access and supporting mobility for both the healthcare professionals as well as the patients, the EMR needs to be kept in big data storage in the healthcare cloud. In spite of the popularity of the healthcare cloud, it faces different security issues; for instance, data theft attacks are considered to be one of the most serious security breaches of healthcare data in the cloud. In this paper, the main focus has been given to secure healthcare private data in the cloud using a fog computing facility.

To this end, a tri-party one-round authenticated key agreement protocol has been proposed based on the bilinear pairing cryptography that can generate a session key among the participants and communicate among them securely. Finally, the private healthcare data are accessed and stored securely by implementing a decoy technique.

Edwin K. Serem [7] Cybersecurity threats are a malicious act that seeks to damage, steal, or gain unauthorized access to information. In recent years there has been an attempt by cybersecurity specialists to come up with an effective system that proactively protects the systems from cyber-attacks. Cyber deception is one efficient method that makes use of decoys to entrap attacks and divert them from real systems. However, existing cyber decoys lack efficiency in hiding true identity due to impractical user activity and network simulation. In this paper, we propose a hybrid decoy system that combines the use of two-layered decoys in the front-end and back-end with an SSH tunnel in between. The front-end decoys will capture attacks and forward them to backend decoys for execution and feedback. General HOSTS framework was used to generate believable user and network activities that can effectively convince the attackers that they are attacking the real systems. All attacker activities are logged by Logstash and presented using Grafana with the Kibana user interface. The experimental results demonstrate that our system can effectively misdirect and misinform attackers by combining deceptive network setup and configurations as well as generating fake user and network activities. Key

Rupesh R Bhairat [8] Cloud computing promises to significantly change the way we use computers and access and store our personal and business information. With these new computing and communications paradigms arise new data security challenges. Existing data protection mechanisms such as encryption have failed in preventing data theft attacks, especially those perpetrated by an insider to the cloud provider. We propose a different approach for securing data in the cloud using offensive decoy technology. We monitor data access in the cloud and detect abnormal data access patterns. When unauthorized access is suspected and then verified using challenge questions, we launch a disinformation attack by returning large amounts of decoy information to the attacker. This protects against the misuse of the user's real data. Experiments conducted in a local file setting provide evidence that this approach may provide unprecedented levels of user data security in a Cloud environment

1S.Pothumani [9] One of the biggest threats of internet communication system is data theft. To handle this number of technologies like encryption algorithms, firewall, and intrusion detection system are used. But this paper discusses a new technique named "decoy". This may be a document, traffic, network or system. Whenever data security is needed, this decoy provides efficient services. This can detect the insider attacker and end number of focus information to the attacker and redirects the attacker from the

original data. Some systems support the decoy document is encrypted one. So, the attacker believes the received document is the original data. So, the attacker easily redirects from the original data. This paper surveys different types of decoy methods in various environments.

Sun et.al [10] Sophisticated adversaries usually initiate their attacks with a reconnaissance phase to discover exploitable vulnerabilities on the targeted networks and systems. To mitigate the effectiveness of persistent reconnaissance attacks, we develop a defensive mechanism that dynamically mutates network topology with a large number of decoys to invalidate the attacker's knowledge from network scanning. We combine the IP randomization technique with decoy techniques and solve two challenges, namely, service availability to legitimate users and service security against unauthorized users. First, our solution can minimize the probability of the real servers being identified and compromised by unauthorized users through deploying a large number of decoy nodes, which change their IP addresses along with the real servers to prolong the scanning time of the attackers. We implement a virtual machine-based system prototype and evaluate it using state-of-the-art scanning techniques. Both theoretical analysis and experimental results show that our solution can effectively mitigate network reconnaissance attacks without sacrificing service availability. To raise awareness of threats and vulnerabilities that exist on the Internet [11][12].

B. THREAT ASSESSMENT

Early detection of emerging security issues is crucial in mitigating their impact. Threat detection and assessment play a pivotal role in swiftly identifying previously unknown attacks, prioritizing their severity, and safeguarding vulnerable systems. Decoy system technology serves this purpose by deploying computer systems deliberately designed to be attractive targets for potential compromise. The decoy system generates virtual counterparts for general network monitoring, enabling the identification of new threats and an assessment of their potential impact on other computer systems. Furthermore, decoy systems act as a deterrent, confounding adversaries by concealing real computer systems amid virtual entities that hold no production value. While directed attacks may not be deterred, many attacks stem from Internet scanning, unable to distinguish between genuine and virtual systems. Designed to emulate systems that intruders would target; decoy systems restrict intruders from gaining access to the entire network if successfully breached. The effectiveness lies in deceiving and monitoring intruders without their awareness. Typically installed within firewalls for better control, decoy systems can also be deployed outside firewalls. The firewall in a decoy system operates inversely to a typical firewall; instead of limiting incoming traffic from the Internet, the decoy system's firewall permits all incoming traffic and restricts outbound communication. The operational model is illustrated in Figure 1.

By luring a hacker into a system, a decoy system serves several purposes:

The administrator can watch the hacker exploit the vulnerabilities of the system, thereby learning where the system has weaknesses that need to be redesigned.

Stopped while trying to obtain root access to the system.

Invulnerable to future hackers.

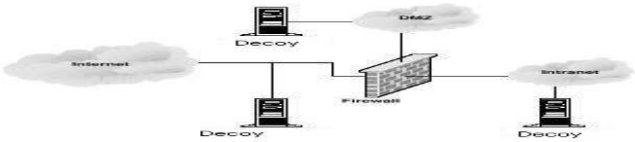

Fig 1 Implementation of Decoy system

Decoy system tokens can manifest in various forms, ranging from inactive, fabricated accounts to database entries intentionally selected only by malicious queries. This versatility makes the concept highly effective for safeguarding data integrity, as any utilization of these tokens raises suspicion, if not indicating outright malicious intent. While decoy system tokens may not outright prevent tampering with data, they provide administrators with an additional layer of confidence in maintaining data integrity.

In the realm of computer terminology, a decoy system serves as a strategic trap designed to identify, divert, or counter attempts at unauthorized use of information systems. Typically, it involves a computer, dataset, or network site that mimics being part of a network but is, in fact, isolated and fortified. The deceptive system appears to contain valuable information or resources that would attract potential attackers.

A decoy system is valuable as a surveillance and early-warning tool. While often a computer, a decoy system can take on other forms, such as files or data records, or even unused IP address space. One very practical implication of this is that decoy systems designed to thwart spam by masquerading as systems of the types abused by spammers to send spam can categorize the material, they trap 100% accurately, it is all illicit. A decoy system needs no spam-recognition capability, no filter to separate ordinary e-mail from spam. Decoy systems can carry risks to a network, and must be handled with care. If they are not properly walled off, an attacker can use them to actually break into a system.

C. DECOY SYSTEM DETECTION

In the ongoing battle against spammers, just as decoy systems serve as a weapon, detection systems designed to identify decoy systems act as a countermeasure employed by spammers. Detection systems are likely to leverage the distinctive characteristics of specific decoy systems for identification. The proliferation of numerous decoy systems in operation contributes to a larger and more formidable set of unique characteristics, complicating the task of those attempting to detect and identify them. This presents a unique scenario in software, where the presence of a multitude of versions of the same software, each slightly different from the others (referred to as "versionitis"), proves to be beneficial. Additionally, there is an advantage in deploying some decoy systems that are intentionally easy to detect, further complicating the efforts of those seeking to identify and counteract them.

D. HONEYPOTS

Honeypots function as closely monitored network decoys, serving multiple purposes within the realm of cybersecurity. These deceptive systems are designed to divert adversaries away from more valuable machines on a network, offer early warnings about

emerging attack trends, and enable thorough examination of adversaries' activities during and after exploitation of the honeypot. Remarkably flexible, honeypots serve various security applications, contributing to prevention, detection, and information gathering rather than addressing a singular issue.

The key characteristic shared by all honeypots is their designation as security resources devoid of any production or authorized activity. Essentially, the deployment of honeypots in a network should not impact critical network services and applications. A honeypot is strategically positioned to be probed, attacked, or compromised, deriving its value from these interactions. Honeypots can be employed for diverse purposes, including the creation of honeypot farms with distributed instances that funnel network traffic to a central honeypot architecture for data collection and analysis. Additionally, they can function as straightforward and efficient Virtual Private Networks (VPNs) for various purposes. One notable advantage of Honeynets is their rapid deployment capability, allowing organizations to establish them within a short timeframe to enhance their cybersecurity posture.

III. IMPLEMENTATION OF DECOY SYSTEM

The primary objective of the "Decoy System" is to prevent the unauthorized entry of intruders into the system. This project has been developed to showcase the access of both authorized and unauthorized users, utilizing Java programming for implementation. The process begins with user registration on the server, where new users need to complete the registration procedure. Existing users can directly access the server, with each user's authentication verified at each login. Upon initiation, the software programs the server, and the firewall is activated. Assuming there are 50 files stored on the server, the firewall mirrors these files, maintaining the same names but with false content. The server retains the original content of the files. Multiple clients can request files from the server. The "Honeypot" concept is integrated to trap potential intruders.

To retrieve a file from the server, users must register with their ID and password. After registration, users can request any file from the server. Personal queries are embedded in the firewall to identify potential intruders. Requests are sent through the firewall to the server, triggering the user to answer personalized queries. If the user correctly answers all the queries, they are deemed genuine, and the request is forwarded from the firewall to the server.

Authorized users receive the requested file directly from the server, ensuring access to the original content. In contrast, unauthorized users, identified through the honeypot mechanism, are denied access. This approach enhances security by implementing a multi-layered authentication process, effectively distinguishing between genuine and potential intruders based on their ability to navigate the personalized queries successfully.

If any one of the queries is answered wrongly by the user then he/she is called as intruder. Once the decoy system depicts the intruder the requests of the file from the intruder is not forwarded to the server. Instead the firewall itself sends the false message of the requested filename to the intruder in alternate path.

The content of the file when transferred from server or from firewall to both genuine user and intruder is in encrypted form and retrieved by decrypting the message using the "RANDOM NUMBER ACCESS" algorithm.

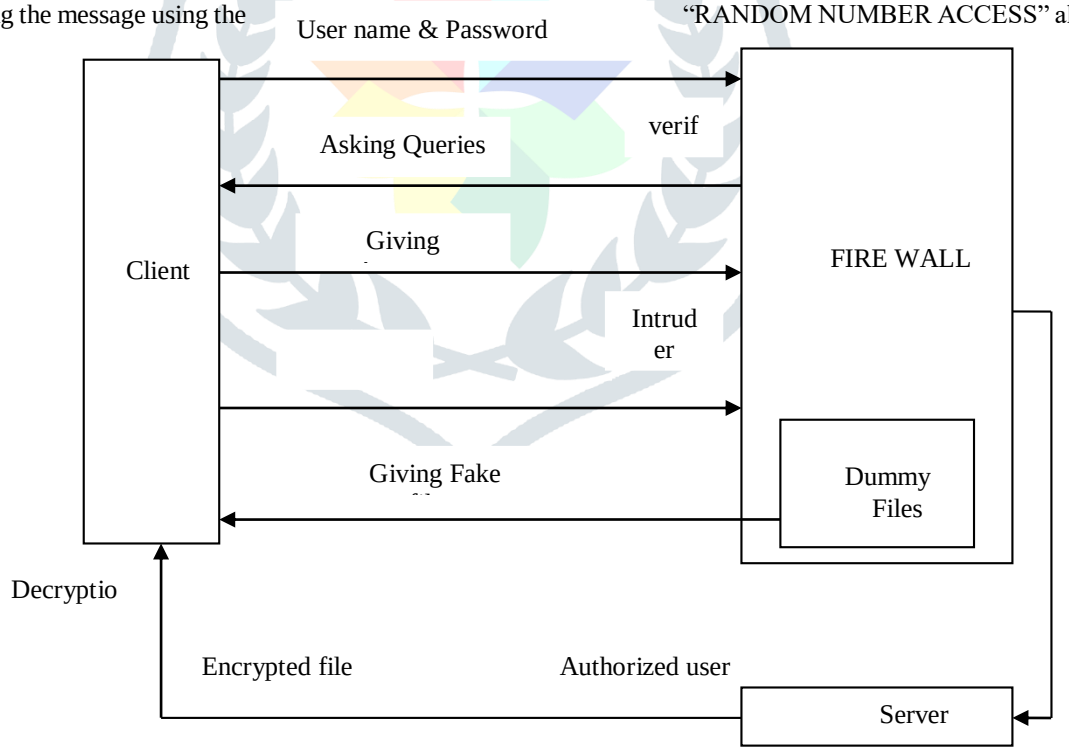

Fig 2 RNA Access algorithm

Therefore, the decoy system is constructed to sting the hackers, not just keep them out. Though both the genuine user and the intruder can access, the original content of the file is read only by the authorized user thereby cheating the intruder by giving the false content of the file. Thus, by implementing decoy system for security purpose serves successfully in better way.

IV. SYMMETRIC KEY ENCRYPTION AND DECRYPTION

A. ENCRYPTION

Encryption is the process of converting data into an unintelligible form, known as ciphertext, using a specific key. Decryption, on the other hand, is the use of a key to revert the ciphertext back to its original, readable form, or plaintext.

In practical terms, the strength of encryption is determined by its ability to safeguard data for the duration during which it remains valuable to a potential malicious actor. For instance, if the goal is to keep a bid on a contract confidential only until the contract is awarded, an encryption method that can be deciphered in a few weeks may suffice. Conversely, when safeguarding sensitive information like a credit card number, a more robust encryption method that withstands decryption attempts for many years is essential.
There are two primary types of encryptions employed in computer security:

1. Symmetric Key Encryption: In this approach, the same key is used for both encryption and decryption. Both the sender and the recipient must possess and share the same secret key for secure communication.

2. Asymmetric Key Encryption: Also known as public-key cryptography, this method involves the use of a pair of keys—a public key for encryption and a private key for decryption. The public key is shared openly, while the private key is kept confidential. This ensures secure communication between parties without the need to exchange a common key.

These encryption techniques play a crucial role in securing information, with the choice between symmetric and asymmetric encryption depending on the specific security requirements and use cases.

B. SYMMETRIC KEY ENCRYPTION AND DECRYPTION

Symmetric-key algorithms, also known as single-key or private-key encryption, are cryptographic algorithms that employ the same cryptographic key for both the encryption and decryption of data. In this approach, you and your counterpart agree on a specific algorithm and a shared key, which is then used for both encrypting and decrypting files.

Symmetric key encryption is known for its speed and simplicity in implementation. The process involves the agreement between parties on the encryption algorithm and the shared key. Once established, either party can encrypt or decrypt a file using this common key. Behind the scenes, symmetric key encryption algorithms are typically implemented as a network of black boxes, which may include hardware components, software, or a combination of both. Each box applies a reversible transformation to the plaintext and passes it to the next box, with each transformation altering the data. The security of a symmetric key algorithm hinges on the challenge of determining which boxes were used and how many times the data passed through this set of boxes, adding a layer of complexity to potential attackers. Refer to Figure 3 for a visual representation of the Symmetric Key Algorithm.

The protocol for symmetric key encryption involves the following steps:
1. The sender and receiver agree on how the key will be used to encrypt and decrypt data.
2. The sender and receiver jointly decide on a specific symmetric key for the encryption and decryption process.

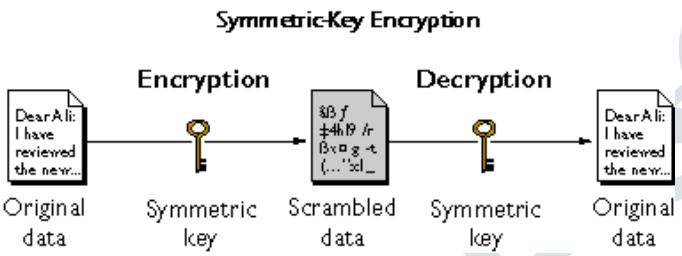

Fig 3 Symmetric Key Algorithm

.

- The algorithms are highly efficient. They are not time consuming to execute.
- The information is authentic because the information with one symmetric key can only be decrypted with the same key.
- The sender and receiver know that they are talking to each other if the symmetric key has not been exposed. Fig 4 shows the Encryption and Decryption with a Symmetric Key.

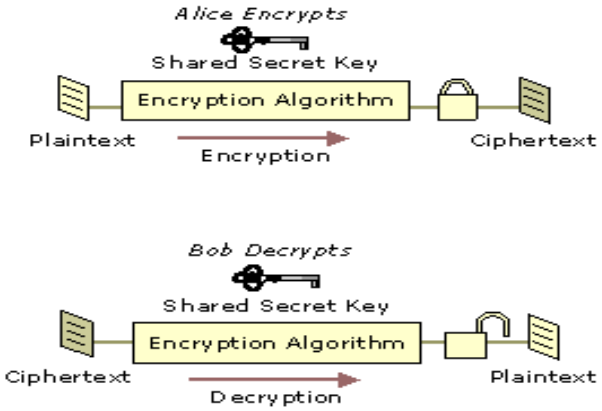

Fig.4 Encryption and Decryption with a Symmetric Key

Symmetric key encryption outpaces public key encryption significantly, often exhibiting speeds 100 to 1,000 times faster. Due to the considerably lighter computational load placed on computer processors, symmetric key technology is preferred for the bulk encryption and decryption of information. This efficiency makes symmetric keys a prevalent choice for ensuring the confidentiality of large-scale data transactions. Security protocols frequently utilize symmetric keys as session keys for securing online communications, ensuring the confidentiality of the exchanged information. Additionally, technologies that require bulk encryption for persistent data, such as email messages and document files, commonly rely on symmetric keys. Notable examples include the use of symmetric keys in Secure/Multipurpose Internet Mail Extensions (S/MIME) to encrypt messages for confidential email communication and in the Encrypting File System (EFS) for encrypting files to maintain confidentiality. The streamlined nature of symmetric key encryption makes it a robust solution for securing a wide range of data types and communication channels.

C. CRYPTOGRAPHICALLY SECURE PSEUDORANDOM NUMBER GENERATORS

A pseudo-random number generator (PRNG) suitable for cryptographic applications is referred to as a cryptographically secure PRNG (CSPRNG). The key distinction between a PRNG and a CSPRNG is straightforward: a CSPRNG should exhibit randomness indistinguishable from true randomness to any algorithm, whereas a PRNG typically only needs to appear random according to standard statistical tests. Cryptographically secure PRNGs, designed explicitly for cryptographic applications, prioritize features like ISAAC, known for its speed and security recommendations, including a notably large expected cycle time. In the realm of randomized algorithms or probabilistic algorithms, these algorithms are permitted to leverage a truly random coin flip. In practice, this often involves access to a pseudo-random number generator on the implementing machine. Such algorithms utilize the random bits as auxiliary input to guide their behavior, aiming for optimal performance in the average case. Formally, the algorithm's performance becomes a random variable influenced by the random bits, with an expected value representing its anticipated runtime. This expected value is known as the expected runtime, emphasizing the algorithm's reliance on pseudo-random number generation for achieving robust and secure cryptographic outcomes.

V. FIREWALL ARCHITECTURE

A. FIREWALL ARCHITECTURE

Firewall solutions can be structured in various architectures, ranging from simple configurations using a single system to more intricate setups involving multiple systems. The effectiveness of a firewall solution is often optimized when designed to ensure that all network traffic passes through it. Different commonly recognized firewall architectures showcase this implementation characteristic.

B. PACKET FILTERING ROUTER

A packet filtering router is a router configured to screen packets between two networks. It routes traffic between the two networks and uses packet filtering rules to permit or deny traffic. Implementing security with a router is usually not that easy. Most routers were designed to route traffic, not to provide firewall functionality, so the command interface used for configuring rules and filters is neither simple nor intuitive.

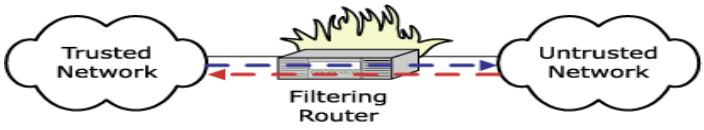

Figure 5 Packet-filtering Router

C. SCREENED HOST (BASTION HOST)

The screened host, commonly known as the bastion host, is usually positioned within the trusted network and shielded from the untrusted network by a packet filtering router. The packet filtering router directs all incoming traffic to the screened host, while outbound traffic may or may not be routed through it. Typically, this firewall type is software-based, operating on a general-purpose computer with a secure version of the operating system. Security measures are predominantly implemented at the application level..

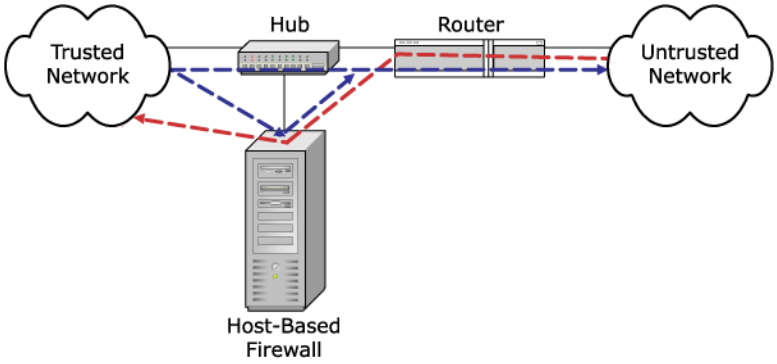

Fig 6 Screened-host or Bastion-host Firewall

D.DUAL-HOMED GATEWAY

A dual-homed gateway typically sits behind the gateway (usually a router) to the untrusted network and most often is a host system with two network interfaces. Traffic forwarding on this system is disabled, thereby forcing all traffic between the two networks to pass through some kind of application gateway or proxy. Only gateways or proxies for the services that are considered essential are installed on the system. This particular architecture will usually require user authentication before access to the gateway/proxy is allowed. Each proxy is independent of all other proxies on the host system.

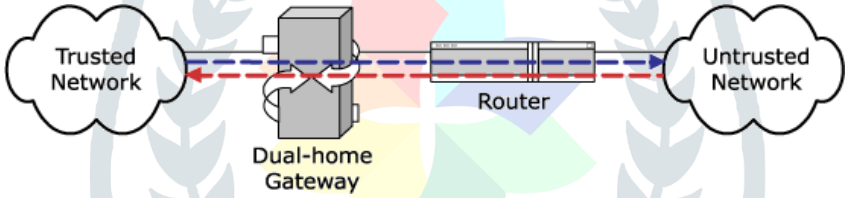

Fig 7 Dual-homed Gateway

E. SCREENED SUBNET OR DEMILITARIZED ZONE (DMZ)

A screened subnet or DMZ is typically created between two packet filtering routers. When using this architecture, the firewall solution is housed on this screened subnet segment along with any other services available to the untrusted network. Conceptually, this architecture is similar to that of a screened host, except that an entire network rather than a single host is reachable from the outside.

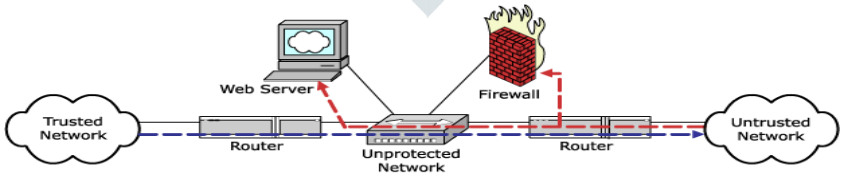

Fig 8 Screened Subnet or Demilitarized Zone (DMZ)

F. FIREWALL APPLIANCE

A firewall appliance typically sits behind the gateway (usually a router) to the untrusted network. This architecture resembles the packet filtering router and dual-homed Gateway architectures in that all traffic must pass through the appliance. In most instances these appliances come pre-configured on their own box. They may also have other services built in, such as Web servers and e-mail servers. Because they usually don't need the extensive configuration that other firewalls often require, they are touted as being much simpler and faster to use. Some manufacturers market them as "plug-and-play" firewall solutions.

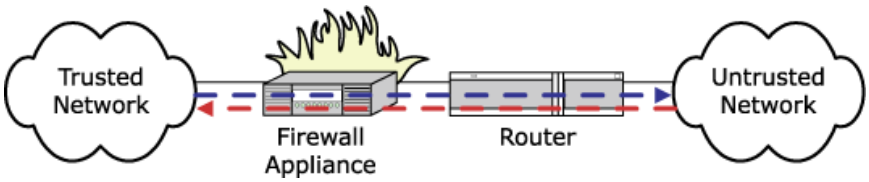

Fig 9 Firewall Appliance

For some networks, implementing more than one firewall solution may be a more effective option. For example, implement a packet filtering router at the entrance to the network for perimeter security and then configure an application gateway for a specific department or building. This type of solution would not only protect the trusted network from the outside, but would also protect a specific department or building from unauthorized users on the trusted network.

VI. RESULTS AND DISCUSSIONS

A.Simulation procedure

Java is an object-oriented programming language is used to develop the model with 40 GB hard disk, 128 MB RAM, 32 Bit NIC card, with Windows 2000 Operating System

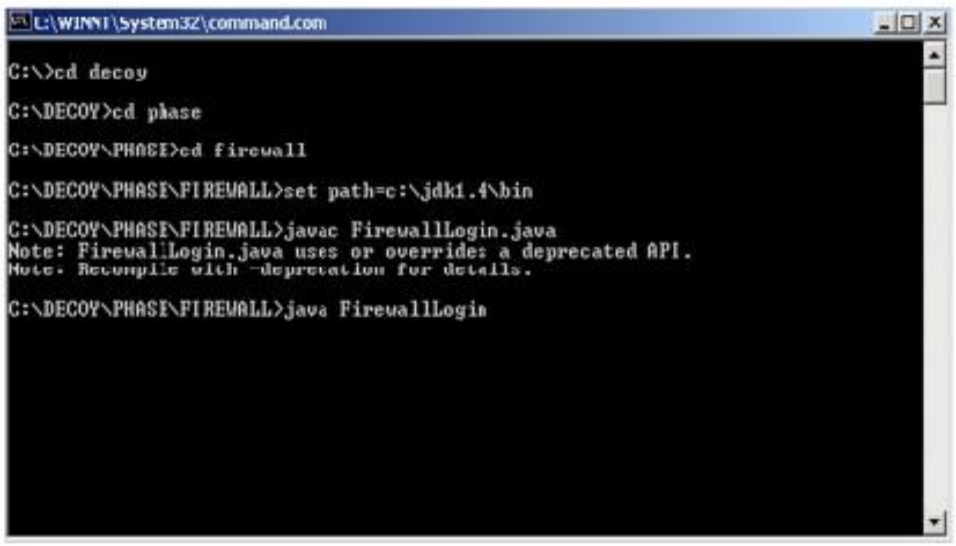

Fig 9 Firewall login

The Initiation of java firewall login is shown in Fig 9 where the Execution process can be monitored.

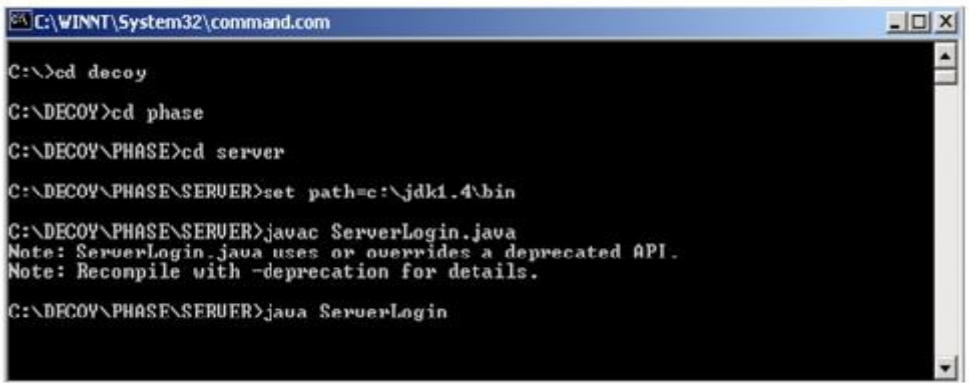

Fig 10 server login

The Initiation of java server login is shown in Fig 10 where the Execution process can be monitored.

Fig 11 decoy system login

The decoy security system window is designed and displayed with security features shown in Fig 11.

ENTER USERNAME AS fire1
AND PASSWORD AS fire1

Fig 12 login and password window

The decoy security system window with username and password is displayed for authentic logins shown in Fig 12.

ENTER USERNAME AS user1
AND PASSWORD AS user1

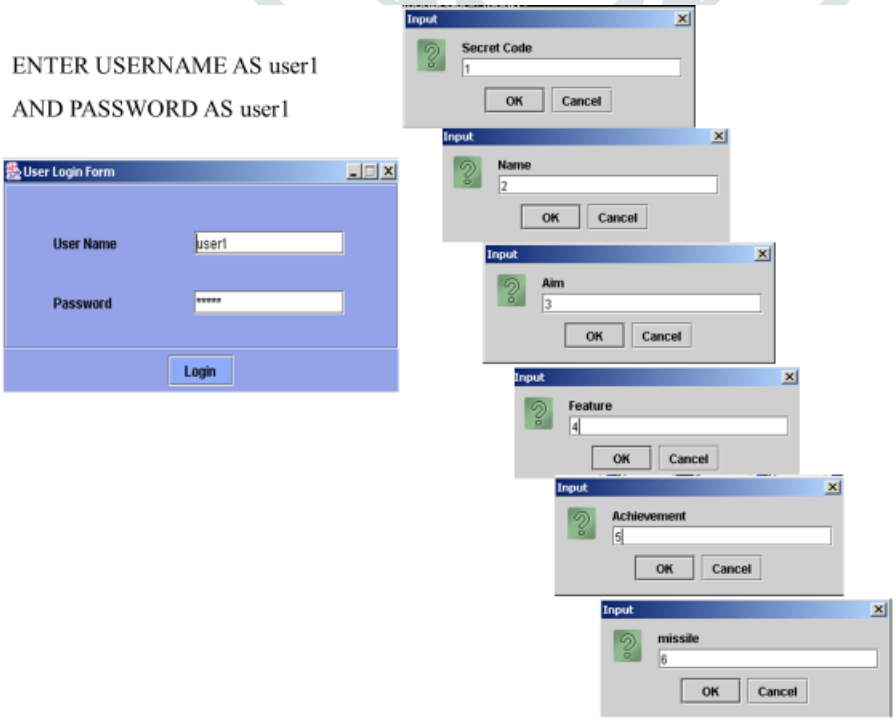

Fig 13 Assigning users

Security features which involves secret code, name, aim, feature, achievement, missile with its security code shown in Fig 13

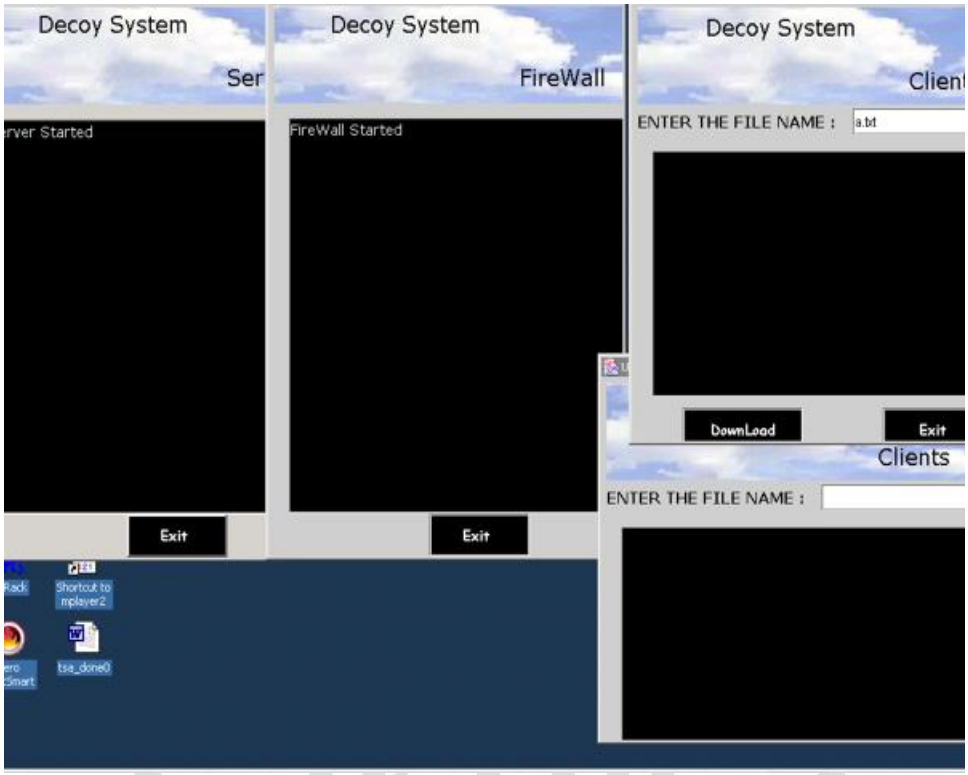

Fig 14 Decoy system Firewall client model
window with server client and firewall integration system extreme security system shown in Fig 14

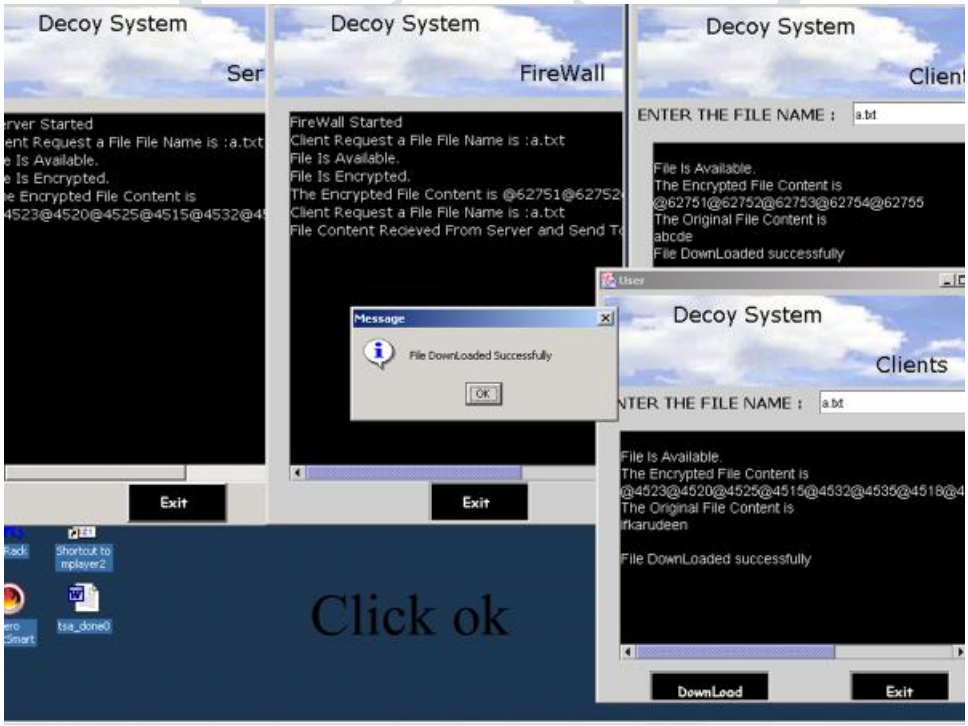

Fig 15 Decoy system /Firewall transferred files with security
Window with server/ client and firewall integration system extreme security system accessing with successful file transfer shown in Fig 15

VII. CONCLUSION

We proposed a defensive framework designed to establish a dynamically mutable network featuring multiple decoys, safeguarding real servers against scanning attacks. Our solution offers uninterrupted connection migration accompanied by IP address randomization. By deploying a decoy bed, we ensure the dual objectives of service availability and service security for the authentic servers. A prototype, implemented using virtual machines (VMs), demonstrates the scalability and flexibility of our system. The results indicate that our solution exhibits robust performance, allowing for frequent server migrations and simultaneous migration of multiple connections with acceptable network and system performance overhead.

Decoy systems continue to evolve, offering enhanced features such as improved logging, cost-effectiveness, and a diverse array of system variations. Ongoing advancements may incorporate intelligent systems leveraging artificial intelligence techniques and survivable system methods. Looking forward, the future trajectory of decoy systems aligns with the evolution witnessed in intrusion detection systems.

REFERENCES

[1] Decoy Systems: A New Player in Network Security and Computer Incident Response Kellep A. Charles, CISSP International Journal of Digital Evidence Winter 2004, Volume 2, Issue 3.

[2] A Model Of Decoy System Based On Dynamic Attributes For Cybercrime Investigation Sviatoslav Vasylyshy et al.

[3] Examining the Efficacy of Decoy-based and Psychological CyberDeception Kimberly J. Ferguson-Walter et al.

[4] Towards a Believable Decoy System: Replaying Network Activities from Real System, 2020 IEEE Conference on Communications and Network Security (CNS), Jianhua Sun et.al 978-1-7281-4760-4/20/$31.00 ©2020 IEEE

[5] Bluetooth for Decoy Systems: A Practical Study, 2017 IEEE Conference on Communications and Network Security (CNS): IEEE CNS 2017 – Ken Wong et.al

[6] A Security Model for Preserving the Privacy of Medical Big Data in a Healthcare Cloud Using a Fog Computing Facility With Pairing-Based Cryptography August 20, 2017, accepted September 23, 2017, date of publication September 28, 2017, date of current version November 7, 2017.Digital Object Identifier 10.1109/ACCESS.2017.2757844

[7] International Journal of Electrical Engineering and Technology (IJEET) Volume 12, Issue 6, June 2021, pp. 281-292, Article ID: IJEET_12_06_027 Available online at https://iaeme.com/Home/issue/IJEET?Volume=12&Issue=6 ISSN Print: 0976-6545 and ISSN Online: 0976-6553 DOI: 10.34218/IJEET.12.6.2021.027 Deceptive Decoys: Combining Believable User And Network Activities And Deceptive Network Setup In Enhancing Effectiveness.

[8] International Engineering Research Journal (IERJ), Volume 2 Issue 6 Page 2044-2046, 2016 ISSN 2395-1621 Fog Computing :Using Decoy Technique, Rupesh R Bhairat et.al

[9] Decoy Method On Various Environments – A Survey International Journal of Pure and Applied Mathematics Volume 116 No. 21 2017, 35-38 ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (on-line version) url: http://www.ijpam.eu S.Pothumani

[10] Decoy-Enhanced Seamless IP Randomization IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications Jianhua Sun, Kun Sunetal.

[11] William W. Martin, CISSP; Honey Pots and Honey Nets - Security through Deception; SANS, May 25, 2001

[12] DecoyRouting: TowardUnblockableInternetCommunication Josh Karlin et.al

[13] Krol, E. (1992). The Whole Internet Guide and Catalog. O'Reilly and Associates.

[14] Design and Analysis of Decoy Systems for Computer Security, ee discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/266044707, All content following this page was uploaded by Brian Bowen on 28 June 2019.

[15] Software-based Decoy System for Insider Threats Younghee Park etal, ASIACCS '12, May 2–4, 2012, Seoul, Korea. Copyright 2012 ACM 978-1-4503-1303-2/12/05