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

Threat Intelligence Platform for Educational Systems

Atharva Auti[1] Jay Makwana[2] Vivek Mishra[3] Shrawani Pagar[4] Dr. Nilakshi Jain[5] Vishakha Shinde[6]
Engineering Student[1,2,3,4] Head of Department[5] Professor[6]
Department of Cyber Security[1,2,3,4,5.6]
Shah & Anchor Kutchhi Engineering College, Mumbai, India[1,2,3,4,5,6]

Abstract: A honeypot is a security mechanism that simulates a vulnerable system or service, attracting malicious actors and allowing security professionals to monitor their behavior. One such honeypot that has proven to be effective in the education sector is made using Snare, Tanner, and Dionaea. This honeypot collects logs of attack attempts, allowing security professionals to analyze and understand the methods and techniques used by attackers. This information can then be used to update security policies and educate students and staff on detecting and preventing such attacks. It can also improve the understanding of the evolving threat landscape and foster a culture of proactive security within educational institutions. The use of honeypots in education can enhance cybersecurity training by providing real-world examples of cyber-attacks and the ways to identify and respond to them.

***Index Terms* – Honeypot, Threat Intelligence, Network Security, Firewall**

I. INTRODUCTION

A honeypot is a security technique used to detect, deflect, or study malicious activities on a computer network. It is essentially a decoy system or service that is designed to attract attackers and record their activities. The purpose of a honeypot is to monitor and analyses the behavior of attackers and to gather intelligence on their methods and techniques. Honeypots can play a crucial role in securing the educational sector against cyber-attacks. With the increasing reliance on technology in education, the number of attacks on educational systems has also increased.

These attacks can cause significant damage to both educational institutions and students. By deploying honeypots, educational institutions can gain a better understanding of the types of attacks that they face and can use this information to improve their security posture. Additionally, studying honeypots in the educational sector can provide valuable insights into the overall security of the sector and help in developing best practices for securing educational systems.

II. LITERATURE SURVEY

In the Literature review of this project, we compared and reviewed 20 literature papers based on the concept of honeypot. Most of the papers were based on a comparison between two technologies and the data collected from previously set honeypots. Other papers were based on how the honeypots have evolved over the years and helped the industry. In the latest papers, the hardware had a great involvement in the honeypots, and new data was collected on the same. Due to this review of literature papers, we were able to gain more knowledge and insights about honeypots implemented in the industry. These literature papers helped us in coming up with the concept of HoneyTrack which has never been implemented before.

In all the papers we reviewed these are the following points which make our project: • Our project is not only a Honeypot or Tracking a hacker, but it is a combination of both, the concept of a honeypot while also backtracking the hacker. • The projects were very complex while our "HoneyTrack" is quick and easy to install and use. • We have a visual representation of all the sorted data we collect using HoneyTrack.

III. METHODOLOGY

Planning

To create our Educational Sector Honeypot, we plan to deploy a sensor using the Dionaea honeypot software, along with the Snare and Tanner tools for data collection and analysis. The sensor will be configured to emulate common services used in educational institutions, such as email, HTTP, FTP, and SSH.

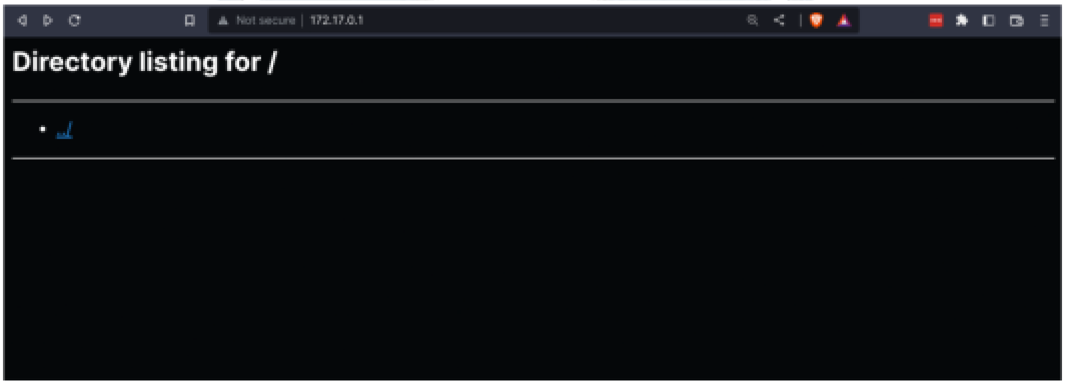
By exposing these services, the sensor will attract and capture any attempts to exploit them, providing valuable insights into the tactics and tools used by attackers.

Why Docker is used –

- Isolation: Running a honeypot in a Docker container provides a level of isolation and security, as the container runs in its own isolated environment.
- Scalability: With Docker, you can easily create multiple containers for each honeypot, making it easier to manage and scale the number of honeypots you are running.
- Portability: Docker, you can build a container image that can be easily moved and run on any machine that supports Docker.
- Configuration Management: Docker, you can manage the configuration of the honeypot through the Dockerfile and build process, making it easier to version control and manage changes to the configuration over time.

Implementation

. Implementation Dionaea can also be installed and run using Docker, which provides a convenient and easy way to run the honeypot in a containerized environment. After starting the Dionaea Docker container, you can access the Dionaea honeypot by connecting to the IP address of the Docker host on the specified ports (for example, http://:80).



Directory listing for /

Fig. 3.1 Browser view after opening http://<docker-host-ip>:80

By default, Dionaea does not serve any files over HTTP, as it is primarily designed to be a honeypot for attackers, not a web server.

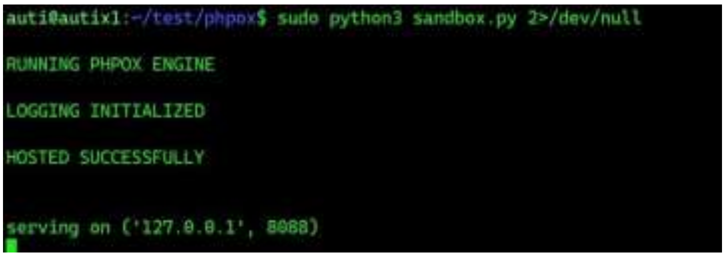
Here are some ports that may need to be exposed for these services:

- Websites: Port 80 (HTTP) and Port 443 (HTTPS)
- Databases: Port 1433 (Microsoft SQL Server), Port 3306 (MySQL), and others
- File sharing: Port 445 (SMB)
- SSH: Port 22

There are several other ports that may be used in the educational sector, depending on the specific services and protocols used. Here are a few examples:

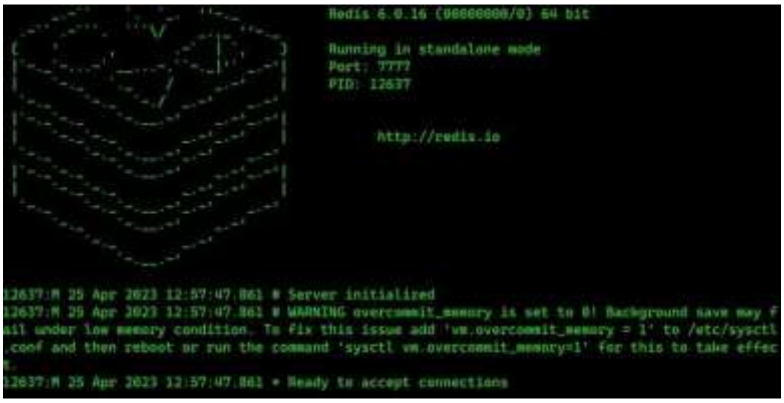
- Remote Desktop Protocol (RDP): Port 3389
- Telnet: Port 23
- Simple Mail Transfer Protocol (SMTP): Port 25
- Network Time Protocol (NTP): Port 123
- Dynamic Host Configuration Protocol (DHCP): Port 67 and 68
- Domain Name System (DNS): Port 53
- Internet Printing Protocol (IPP): Port 631

IV. RESULTS AND DISCUSSION



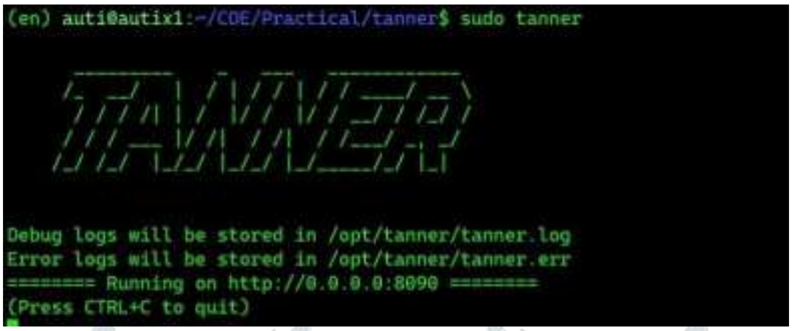
```
auti@autix1:~/test/phpox$ sudo python3 sandbox.py 2>/dev/null

RUNNING PHPOX ENGINE

LOGGING INITIALIZED

HOSTED SUCCESSFULLY


serving on ('127.0.0.1', 8088)
```

Fig. 4.1: Running Phpox Engine.



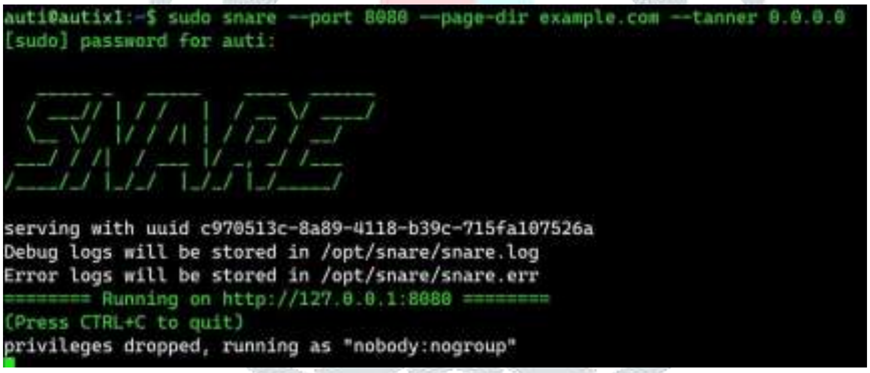
```
Redis 6.0.16 (00000000/0) 64 bit

Running in standalone mode
Port: 7777
PID: 12637

      http://redis.io

12637:M 25 Apr 2023 12:57:47.861 # Server initialized
12637:M 25 Apr 2023 12:57:47.861 # WARNING overcommit_memory is set to 0! Background save may f
ail under low memory condition. To fix this issue add 'vm.overcommit_memory = 1' to /etc/sysctl
.conf and then reboot or run the command 'sysctl vm.overcommit_memory=1' for this to take effec
t.
12637:M 25 Apr 2023 12:57:47.861 * Ready to accept connections
```

Fig. 4.2: Using Redis Server for storing data.



```
(en) auti@autix1:~/COE/Practical/tanner$ sudo tanner

Debug logs will be stored in /opt/tanner/tanner.log
Error logs will be stored in /opt/tanner/tanner.err
======== Running on http://0.0.0.0:8090 ========
(Press CTRL+C to quit)
```

Fig. 4.3: Running Tanner to start logging.

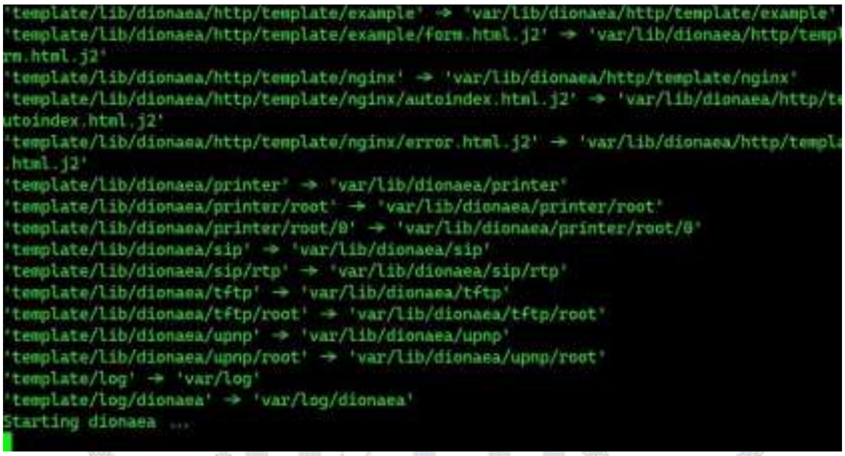


```
auti@autix1:~$ sudo snare --port 8080 --page-dir example.com --tanner 0.0.0.0
[sudo] password for auti:

serving with uuid c970513c-8a89-4118-b39c-715fa107526a
Debug logs will be stored in /opt/snare/snare.log
Error logs will be stored in /opt/snare/snare.err
======== Running on http://127.0.0.1:8080 ========
(Press CTRL+C to quit)
privileges dropped, running as "nobody:nogroup"
```

Fig. 4.4: Running Snare and Hosting example.com domain.

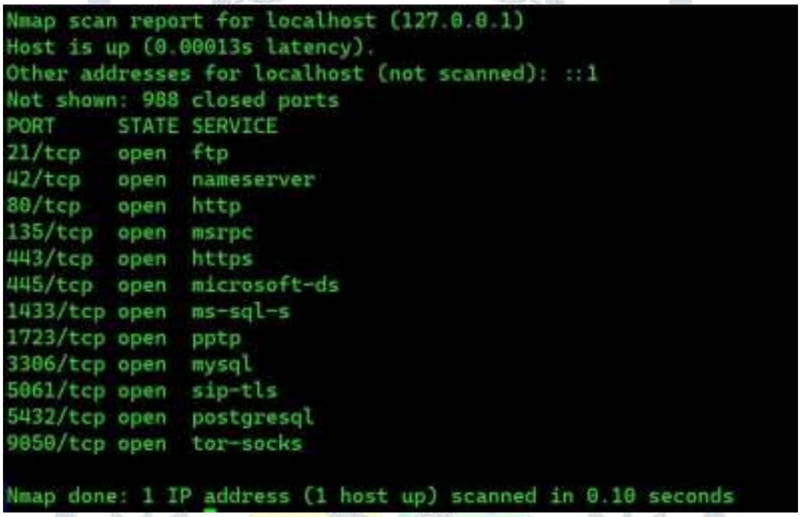
Figure 4.1 represents the running of the Phpox Engine, a software component used for processing PHP code and generating dynamic web content. Figure 4.2 represents the utilization of Redis Server for storing data, showcasing its role in efficiently storing and retrieving data for various applications.

Figure 4.3 represents the activation of Tanner to start logging, highlighting its function as a logging tool for recording and tracking system events, actions, and errors. Figure 4.4 represents the running of Snare and hosting the example.com domain, illustrating the use of Snare as a network monitoring and intrusion detection system while hosting a specific domain.



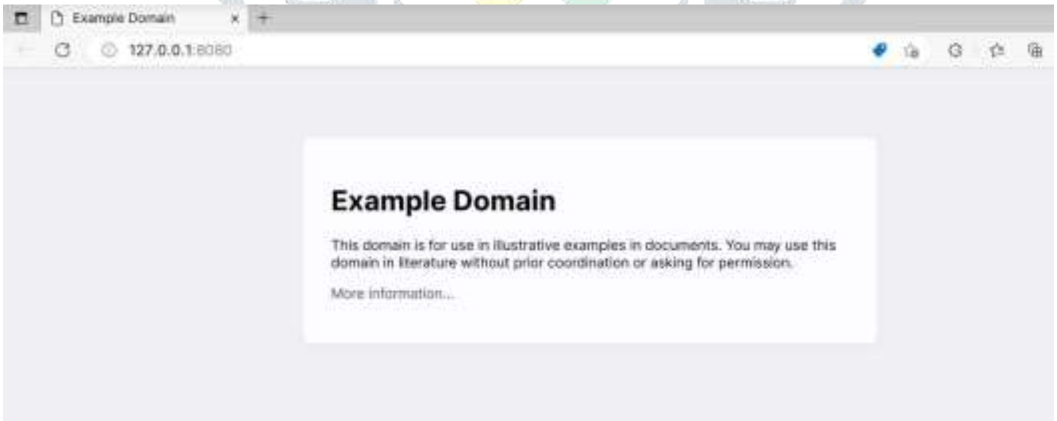
```
'template/lib/dionaea/http/template/example' → 'var/lib/dionaea/http/template/example'
'template/lib/dionaea/http/template/example/form.html.j2' → 'var/lib/dionaea/http/templ
rm.html.j2'
'template/lib/dionaea/http/template/nginx' → 'var/lib/dionaea/http/template/nginx'
'template/lib/dionaea/http/template/nginx/autoindex.html.j2' → 'var/lib/dionaea/http/te
utoindex.html.j2'
'template/lib/dionaea/http/template/nginx/error.html.j2' → 'var/lib/dionaea/http/templa
.html.j2'
'template/lib/dionaea/printer' → 'var/lib/dionaea/printer'
'template/lib/dionaea/printer/root' → 'var/lib/dionaea/printer/root'
'template/lib/dionaea/printer/root/0' → 'var/lib/dionaea/printer/root/0'
'template/lib/dionaea/sip' → 'var/lib/dionaea/sip'
'template/lib/dionaea/sip/rtp' → 'var/lib/dionaea/sip/rtp'
'template/lib/dionaea/tftp' → 'var/lib/dionaea/tftp'
'template/lib/dionaea/tftp/root' → 'var/lib/dionaea/tftp/root'
'template/lib/dionaea/upnp' → 'var/lib/dionaea/upnp'
'template/lib/dionaea/upnp/root' → 'var/lib/dionaea/upnp/root'
'template/log' → 'var/log'
'template/log/dionaea' → 'var/log/dionaea'
Starting dionaea ...
```

Fig. 4.5: Running Dionaea in Docker.



```
Nmap scan report for localhost (127.0.0.1)
Host is up (0.00013s latency).
Other addresses for localhost (not scanned): ::1
Not shown: 988 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
42/tcp   open  nameserver
80/tcp   open  http
135/tcp  open  msrpc
443/tcp  open  https
445/tcp  open  microsoft-ds
1433/tcp open  ms-sql-s
1723/tcp open  pptp
3306/tcp open  mysql
5061/tcp open  sip-tls
5432/tcp open  postgresql
9050/tcp open  tor-socks

Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
```

Fig. 4.6: Services running on Localhost.



Example Domain

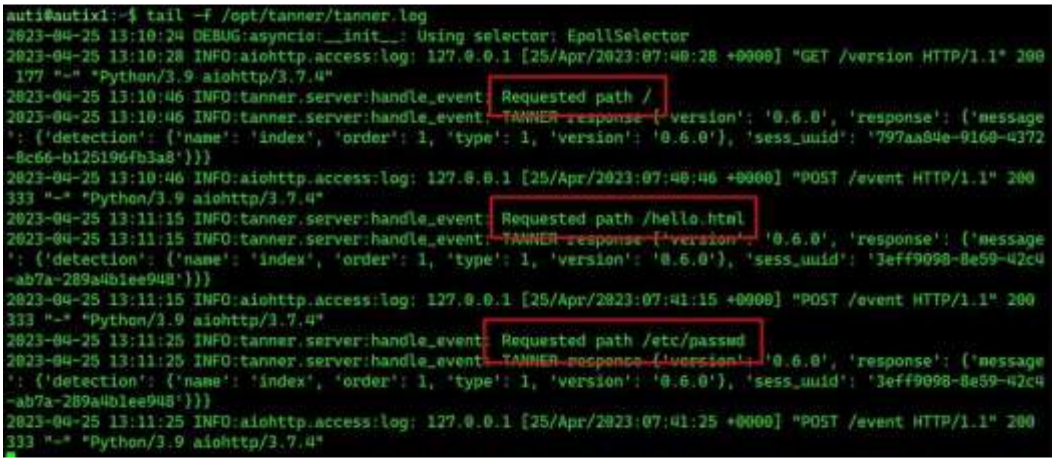
This domain is for use in illustrative examples in documents. You may use this domain in literature without prior coordination or asking for permission.

More information...

Fig. 4.7: HTTP Page on Localhost.

Figure 4.5 represents the running of Dionaea in Docker, showcasing the deployment of Dionaea as a honeypot software within a Docker container to attract and analyze malicious activities. Figure 4.6 represents the services running on Localhost, providing an overview of various active applications or services operating on the localhost network interface.

Figure 4.7 represents an HTTP page on Localhost, displaying a web page hosted on the localhost accessible through a local web server using the Hypertext Transfer Protocol (HTTP). Figure 4.8 represents the Tanner Logs, presenting the logs generated by the Tanner logging tool, which capture and document system events, errors, and other relevant information for analysis and troubleshooting purposes.



```
auti@autix1:~$ tail -f /opt/tanner/tanner.log
2023-04-25 13:10:24 DEBUG:asyncio:__init__: Using selector: EpollSelector
2023-04-25 13:10:28 INFO:aiohttp.access:log: 127.0.0.1 [25/Apr/2023:07:40:28 +0000] "GET /version HTTP/1.1" 200
 177 "-" "Python/3.9 aiohttp/3.7.4"
2023-04-25 13:10:46 INFO:tanner.server:handle_event: Requested path /
2023-04-25 13:10:46 INFO:tanner.server:handle_event: TANNER response {'version': '0.6.0', 'response': {'message
': {'detection': {'name': 'index', 'order': 1, 'type': 1, 'version': '0.6.0'}, 'sess_uuid': '797aa84e-9160-4372
-8c66-b125196fb3a8'}}}
2023-04-25 13:10:46 INFO:aiohttp.access:log: 127.0.0.1 [25/Apr/2023:07:40:46 +0000] "POST /event HTTP/1.1" 200
333 "-" "Python/3.9 aiohttp/3.7.4"
2023-04-25 13:11:15 INFO:tanner.server:handle_event: Requested path /hello.html
2023-04-25 13:11:15 INFO:tanner.server:handle_event: TANNER response {'version': '0.6.0', 'response': {'message
': {'detection': {'name': 'index', 'order': 1, 'type': 1, 'version': '0.6.0'}, 'sess_uuid': '3eff9098-8e59-42c4
-ab7a-289a4b1ee948'}}}
2023-04-25 13:11:15 INFO:aiohttp.access:log: 127.0.0.1 [25/Apr/2023:07:41:15 +0000] "POST /event HTTP/1.1" 200
333 "-" "Python/3.9 aiohttp/3.7.4"
2023-04-25 13:11:25 INFO:tanner.server:handle_event: Requested path /etc/passwd
2023-04-25 13:11:25 INFO:tanner.server:handle_event: TANNER response {'version': '0.6.0', 'response': {'message
': {'detection': {'name': 'index', 'order': 1, 'type': 1, 'version': '0.6.0'}, 'sess_uuid': '3eff9098-8e59-42c4
-ab7a-289a4b1ee948'}}}
2023-04-25 13:11:25 INFO:aiohttp.access:log: 127.0.0.1 [25/Apr/2023:07:41:25 +0000] "POST /event HTTP/1.1" 200
333 "-" "Python/3.9 aiohttp/3.7.4"
```

Fig. 4.8: Tanner Logs.

IV. CONCLUSION

In conclusion, this project aimed to research and implement honeypots in the educational sector. Through the research, we gained insights into the various types of honeypots used in the educational sector and the characteristics of the technology landscape. We implemented a honeypot system using Dionaea on top of Snare and Tanner using Docker on a Raspberry Pi. The project has provided valuable insights into the use of honeypots in the educational sector and demonstrated their effectiveness in detecting and capturing attacks.

Future-Scope

The project has opened up several avenues for future research and development. Firstly, the implemented honeypot system can be further improved by integrating additional features such as machine learning algorithms to better detect and respond to attacks. Secondly, the honeypot system can be deployed in different educational institutions to evaluate its effectiveness in different environments. Lastly, the findings from this project can be used to develop cybersecurity awareness and training programs for educational institutions to help them understand and defend against cyber-attacks.

Limitations

- The Software Docker isn’t optimized for Windows.
- If the attacker is using a proxy server, then the actual location of the intruder is difficult to track.
- In case of minimal requirements, if DoS attacks increase the server may fail.
- Data fragmentation while bypassing the firewall and honeypot can lead the attacker to the server

V. ACKNOWLEDGEMENT

We have great pleasure in presenting the project on “Threat Intelligence Platform For Educational Systems”. We take this opportunity to express our sincere thanks to our Guide, Ms. Vishakha Shinde, and our Co-Guide, Dr. Nilakshi Jain and the faculty in the Department of Cyber Security in Shah and Anchor Kutchhi Engineering College for guiding us and suggesting regarding the line of work.

We would like to express our gratitude towards their constant encouragement, support, and guidance throughout the progress. Also, we would like to thank our principal – Dr. Bhavesh Patel and Dr. Nilakshi Jain, Head of Cyber Security Department, for their help, support & guidance for this project. We are also thankful to all Faculty members of our department for their help and guidance during completion of our project.

REFERENCES

[1] R. Vishwakarma and A. K. Jain, "A Honeypot with Machine Learning based Detection Framework for defending IoT based Botnet DDoS Attacks," 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), 2019, pp. 1019-1024, doi: 10.1109/ICOEI.2019.8862720.

[2] M. Du and K. Wang, "An SDN-Enabled Pseudo-Honeypot Strategy for Distributed Denial of Service Attacks in Industrial Internet of Things," in IEEE Transactions on Industrial Informatics, vol. 16, no. 1,pp. 648-657, Jan. 2020, doi: 10.1109/TII.2019.2917912.

[3] L. Shi, Y. Li, T. Liu, J. Liu, B. Shan and H. Chen, "Dynamic Distributed Honeypot Based on Blockchain," in IEEE Access, vol. 7, pp. 72234-72246, 2019, doi: 10.1109/ACCESS.2019.2920239.

[4] Cheng Huang, Jiaxuan Han, Xing Zhang, Jiayong Liu, "Automatic Identification of Honeypot ServerUsing Machine Learning Techniques", Security and Communication Networks, vol. 2019, Article ID 2627608, 8 pages, 2019. https://doi.org/10.1155/2019/2627608

[5] Lee S, Abdullah A, Jhanjhi N, Kok S. 2021. Classification of botnet attacks in IoT smart factory using honeypot combined with machine learning. PeerJ Computer Science 7:e350 https://doi.org/10.7717/peerj-cs.350

[6] Naik, N., Jenkins, P., Savage, N. et al. A computational intelligence enabled honeypot for chasing ghosts in the wires.

Complex Intell. Syst. 7, 477–494 (2021). https://doi.org/10.1007/s40747-020-00209- 5

[7] Armin Ziaie Tabari and Xinming Ou. 2020. A Multi-phased Multi-faceted IoT Honeypot Ecosystem. Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security. Association for Computing Machinery, New York, NY, USA, 2121– 2123. DOI: https://doi.org/10.1145/3372297.3420023

[8] Suratkar, S., Shah, K., Sood, A. et al. An adaptive honeypot using Q-Learning with severity analyzer. J Ambient Intell Human Comput (2021). https://doi.org/10.1007/s12652-021-03229-2

[9] Mondal, Avijit & Goswami, Radha. (2020). Enhanced Honeypot cryptographic scheme and privacy preservation for an effective prediction in cloud security. Microprocessors and Microsystems. 81. 103719. 10.1016/j.micpro.2020.103719.

[10] M. Dodson, A. R. Beresford and M. Vingaard, "Using Global Honeypot Networks to Detect Targeted ICS Attacks," 2020 12th International Conference on Cyber Conflict (CyCon), 2020, pp. 275-291, doi: 10.23919/CyCon49761.2020.9131734.

[11] P. S. Negi, A. Garg and R. Lal, "Intrusion Detection and Prevention using Honeypot Network for Cloud Security," 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), 2020, pp. 129-132, doi: 10.1109/Confluence47617.2020.9057961.

[12] O. Surnin et al., "Probabilistic Estimation of Honeypot Detection in Internet of Things Environment," 2019 International Conference on Computing, Networking and Communications (ICNC),2019, pp. 191-196, doi: 10.1109/ICCNC.2019.8685566

[13] I. M. M. Matin and B. Rahardjo, "Malware Detection Using Honeypot and Machine Learning," 2019 7th International Conference on Cyber and IT Service Management (CITSM), 2019, pp. 1-4, doi: 10.1109/CITSM47753.2019.8965419.

[14] W. Tian, M. Du, X. Ji, G. Liu, Y. Dai and Z. Han, "Honeypot Detection Strategy Against Advanced Persistent Threats in Industrial Internet of Things: A Prospect Theoretic Game," in IEEE Internet of Things Journal, vol. 8, no. 24, pp. 17372-17381, 15 Dec.15, 2021, doi: 10.1109/JIOT.2021.3080527

[15] A. H. Anwar, C. Kamhoua and N. Leslie, "Honeypot Allocation over Attack Graphs in Cyber Deception Games," 2020 International Conference on Computing, Networking and Communications (ICNC), 2020, pp. 502-506, doi: 10.1109/ICNC47757.2020.9049764.

[16] D. Pliatsios, P. Sarigiannidis, T. Liatifis, K. Rompolos and I. Siniosoglou, "A Novel and InteractiveIndustrial Control System Honeypot for Critical Smart Grid Infrastructure," 2019 IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links andNetworks (CAMAD), 2019, pp. 1-6, doi: 10.1109/CAMAD.2019.8858431.

[17] Nadiya El Kamel, Mohamed Eddabbah, Youssef Lmoumen, Raja Touahni, "A Smart Agent Design for Cyber Security Based on Honeypot and Machine Learning", Security and Communication Networks, vol. 2020, Article ID 8865474, 9 pages, 2020. https://doi.org/10.1155/2020/8865474

[18] S. Sibi Chakkaravarthy, D. Sangeetha, M. V. Cruz, V. Vaidehi and B. Raman, "Design of Intrusion Detection Honeypot Using Social Leopard Algorithm to Detect IoT Ransomware Attacks," in IEEE Access, vol. 8, pp. 169944-169956, 2020, doi: 10.1109/ACCESS.2020.3023764.

[19] V. Sethia and A. Jeyasekar, "Malware Capturing and Analysis using Dionaea Honeypot," 2019 International Carnahan Conference on Security Technology (ICCST), 2019, pp. 1-4, doi: 10.1109/CCST.2019.8888409.

[20] A. Auti, S. Pagar, V. Mishra, J. Makwana and S. Borade, "HoneyTrack: An improved honeypot," 2023 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), Bhopal, India, 2023, pp. 1-6, doi: 10.1109/SCEECS57921.2023.10063105.