

An Intelligence Defense System with SNORT Rules

Yi-Cheng Lai*, Chiao-Lin Yu*, Man-Ling Liao*, Yu-Shan Lin**, Yao-Chung Chang*** and Jiann-Liang Chen*

* Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan

** Department of Information Science and Management Systems, National Taitung University, Taitung, Taiwan

***Department of Computer Science and Information Engineering, National Taitung University, Taitung, Taiwan

m11007505@mail.ntust.edu.tw, m10907505@mail.ntust.edu.tw, m11007502@mail.ntust.edu.tw, ysl@nttu.edu.tw, ycc@nttu.edu.tw, lchen1215@gmail.com

Abstract—Misconfiguration of firewall rules has always been considered a serious issue. The handwritten rule is messy and buggy under the increasingly complex firewall architecture. To avoid being attacked behind an insecure firewall. This study defines an intelligence defense system. Combined with data analysis, feature extraction, optimization, and firewall technology. Its main purpose is to replace handwritten firewall rules and provide immediate and reliable protection against diversified attacks. In the verification, 68,936,206 packets collected by Cowrie honeypot were used as the test data. The accuracy rate of classifying different attack behaviors reached 99.5%, and the packet coverage of Snort rules also achieved 98%. This thesis proposes a system that can effectively identify and defend from diverse attacks.

Keywords—MITRE ATT&CK, K-means, PCRE Regular Expression, Snort Rule

I. INTRODUCTION

With the rapid development of technology, the rise of the Internet world has brought about an epoch-making revolution. Since the development of the late 20th century, the Internet has long been inseparable from people's lives. With the combination of software and hardware and the development of Internet technology, the Internet world is boundless among individuals, institutions, enterprises, and even countries. It is full of information, including private information, corporate secrets, and state secrets. While providing information to all parties to facilitate exchanges, people with intentions are also eyeing it, and cybercrimes emerge. Protecting essential data in real-time and effectively has become an issue that no one can ignore.

For a long time, the inefficiency and loopholes caused by the artificial configuration of firewall rules have been a topic of concern [1,2]. In recent years, nearly 60% of the security breaches on the Internet have been caused by system and network administrators [3]. Such as the Cloudflare outage incident in 2019 [4], Due to misconfiguration rules deployed in the Cloudflare Web Application Firewall, 100% of CPU spikes caused customers to see 502 errors, and the worst traffic loss reached about 82%. Obviously, with the growth of network services, increasingly complex firewalls are challenging for firewall administrators and network administrators to manage

effectively. Meanwhile, the handwritten rule is messy and buggy under the increasingly complex firewall architecture.

To solve the above problem, this study uses the k-means algorithm to classify attacks collected through Cowrie, mark them according to the Enterprise Tactics of MITRE ATT&CK, and then use the genetic algorithm to automatically generate PCRE Regular that matches all packet payloads in the cluster Expression, finally synchronously configured in the snort firewall, and then conduct a detection test through a python-based scapy script. After a complete system process, firewall rule sets can be built automatically to defend against malicious attacks from all parties. The techniques mentioned above will be described in detail in the next section.

II. RELATED WORK

Due to the increasing number of cyber-attacks, honeypot systems have become an active research area. Among many protocols, SSH has been the subject of much research. Bryson et al. [5] showed that the Cowrie honeypot is efficient for collecting and analyzing the activities of malicious actors, especially for SSH and Telnet.

This thesis uses Cowrie to collect high-value data for further research. There has been much research using Cowrie to monitor attacker activity. Başer et al. [6] discuss capturing brute force attacks against default usernames and passwords but lack the analysis of the command entered, which could actually cause serious problems in the target system. Fraunholz et al. [7] discuss analysis based on timing behavior, focus on network activity, and mention the feasibility of machine learning to classify attack behavior. The major problem is defining a suitable basis for classification and the systematically represent of classified attacks.

K-means clustering algorithm is a commonly used unsupervised learning algorithm in data mining, and it is often used in various fields, such as earthquake analysis [8] and COVID-19 risk determination [9]. Due to the feature of different attack types being significant and the similarity of a single attack mode being high, K-means can efficiently classify different attack behaviors.

MITER ATT&CK is a database that records information security attacks and information sharing [10]. Its globally-

accessible knowledge base covers almost all attacks on the Internet today. It will be updated regularly to keep track of new types of attacks, which allows it to represent classified attacks more systematically.

Snort is a powerful, lightweight Network Intrusion Detection (NIDS) program proposed by Martin Roesch in 1989. It has been widely used for real-time network traffic analysis and packet logging. E Satria et al. proposed a way to automate the generation of snort rules using honeypot logs in 2021 [11], but it only focuses on brute force attacks. In this study, we classify data before signature extraction, which can easily observe high-threatening attacks and formulate them into appropriate snort rules. The efficiency is much higher than extracting signatures from massive log information.

III. PROPOSED METHOD

The defense system proposed in this study is mainly divided into three parts. A. Uses the K-means algorithm to classify the suspicious behavior collected by Honeypot and label it with MITRE ATT&CK, B. Combine PCRE Regular Expression with Snort to Generate high-coverage firewall rule sets, C. A testing script to Verify firewall rule coverage. The proposed system flow is shown in Figure 1.

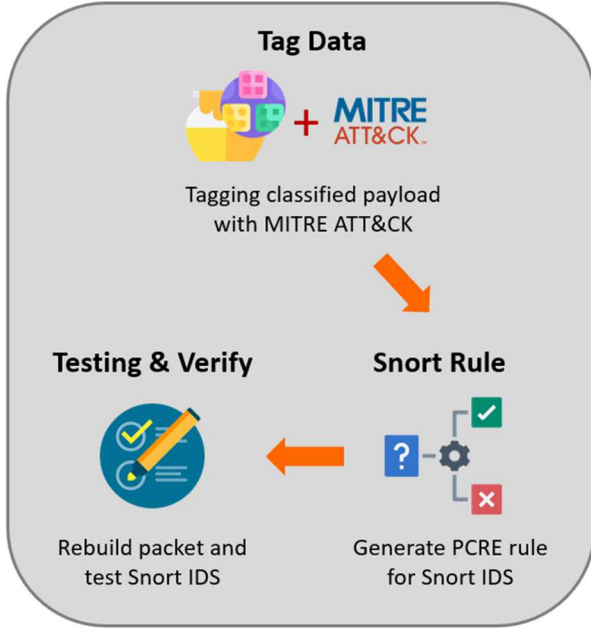


Figure 1. Proposed System Flow

After processing the malicious attack behavior collected from Cowire into a specific format, discard some redundant data, and all payloads are converted into a vector through an encoder. Sprinkle a few center points throughout the vector space and divide them into groups according to the distance close to the center point so that similar payloads can be clustered and marked for groups instead of individual payloads. The k-means formula and processing flow are as follows:

$$\min \sum_{i=1}^k \sum_{x_j \in S_i} (x_j - \mu_i)^2 \quad (1)$$

K-means algorithm is an unsupervised learning algorithm, as shown in Figure 2. The processing flow includes the following:

1. The parameter K is first set, used as the number of cluster centers in the vector.
2. The cluster center is scattered in the vector space for each data. Calculate the shortest Euclidean distance between each data to all cluster centers.
3. Take the cluster with the shortest distance for each data and classify it as that cluster.
4. Finally, repeat steps 2 and 3. Finish until all data points don't change too much from the cluster center.

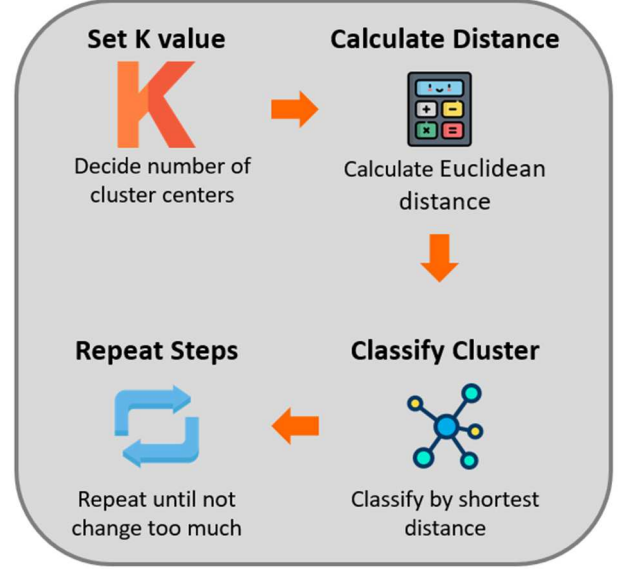


Figure 2. Flowchart of K-means

A. MITRE ATT&CK

MITRE ATT&CK was then added here to label classified data with Tactic, Technique, Sub-technique, or TTP for short. TTP can easily tag the specific attack behavior, which is helpful for further analysis. Some of the TTPs used in this study are shown in Table 1 as examples. In the case of Command and Scripting Interpreter under TA002, Sub-techniques Unix Shell (T1059.004) indicates that the attacker may abuse Unix shell commands and scripts for execution. If the payload matches Unix Shell this Sub-technique, the label will be recorded as [T1059.004], and If there are no specific Sub-techniques that match the attack behavior, it will be recorded as [T1059.000].

TABLE 1. THE CURRENT MITRE ATT&CK TACTIC

Tactic	Description
Initial Access TA0001	Attackers use various network access technology on victim devices.
Execution TA0002	Attackers use the execution command on the victim's device.
Persistence TA0003	Attackers maintain access to victim devices, remotely reboot, change credentials, and more.

Privilege Escalation TA0004	Attackers try to gain elevated privileges on the victim's device.
Defense Evasion TA0005	Attackers try to hide their tracks on a victim's device.
Credential Access TA0006	Attackers steal credentials such as account passwords on victim devices.
Resource Development TA0042	Attackers achieve their goals by creating, buying, destroying, stealing, and more.
Reconnaissance TA0043	Attackers actively or passively collect victim device information to achieve their goals.

TABLE 2. SOME TECHNIQUES IN TACTIC EXECUTION

Tactic: Execution (TA0002)	
Technique	Description
Command and Scripting Interpreter (T1059)	Attackers may use automatic or manual methods to execute commands.
User Execution (T1204)	An adversary may rely upon specific actions by a user in order to gain execution.
Inter-Process Communication (T1559)	Attackers may exploit inter-process communication (IPC) to execute commands.

TABLE 3. THE DETAIL IN TECHNIQUE: COMMAND AND SCRIPTING INTERPRETER (T1059)

Technique: Command and Scripting Interpreter (T1059)		
Subtechnique	Name	Description
T1059.001	PowerShell	Attackers may use PowerShell commands to execute.
T1059.002	AppleScript	Attackers may use AppleScript to execute.
T1059.003	Windows Command Shell	Attackers may use the Windows command shell to execute.
T1059.004	Unix Shell	Attackers may use Unix shell commands and scripts to execute.

T1059.005	Visual Basic	Attackers may use Visual Basic (VB) to execute.
T1059.006	Python	Attackers may use Python commands and scripts to execute.
T1059.007	JavaScript	Attackers may use various implementations of JavaScript to execute.
T1059.008	Network Device CLI	Attackers may use scripting or built-in command line interpreters (CLI) on network devices to execute malicious commands and payloads.

TABLE 4. CLUSTER LABELED BY MITRE ATT&CK FRAMEWORK

Cluster ID	Tactic	Subtechnique	Malicious Score
1	TA0002	T1059.004	0.8
2	TA0002	T1059.004	0.8
3	TA0002	T1059.004	0.8
4	TA0002	T1059.004	0.8
5	TA0007	T1120.000	0.6
6	TA0002	T1059.004	0.8
7	TA0002	T1059.004	0.8
8	TA0002	T1059.004	0.8
9	TA0001 TA0002	T1190.000 T1059.004	0.8
10	TA0002	T1059.004	1
11	TA0007	T1087.001	0.6
12	TA0002	T1059.004	0.8
13	TA0001 TA0002 TA0007 TA0040	T1190.000 T1059.004 T1087.001 T1485.000	0.8
14	TA0001 TA0002 TA0040	T1190.000 T1059.004 T1485.000	0.6
15	TA0007 TA0002	T1087.001 T1059.004	1
16	TA0002	T1059.004	0.8
17	TA0002	T1059.004	0.8
18	TA0002	T1059.004	1
19	TA0002	T1059.004	0.8
20	TA0002	T1059.004	0.8
21	TA0002	T1059.004	0.8
22	TA0002	T1059.004	0.8
23	TA0002	T1059.004	0.8
24	TA0002	T1059.004	0.8
25	TA0002	T1059.004	0.8
26	TA0002	T1059.004	0.8

27	TA0002	T1059.004	0.8
28	TA0040 TA0002	T1485.000 T1059.004	0.8
29	TA0001	T1190	0.2
30	TA0006	T1110.000	0.6
31	TA0002	T1190.000 T1059.004	0.6

TABLE 5. THE STANDARD OF MALICIOUS SCORE

The standard of Malicious score	
0	The command is a blank, line feed, and other meaningless commands.
0.2 – 0.4	The command tries to connect.
0.6	Brute force crack account password.
0.8 - 1	Enter commands, Exploit, and other actions on the system.

B. Open-source Intrusion Prevention System (IPS) Snort

The open-source Intrusion Prevention System (IPS) Snort used in this study supports the PCRE Regular Expression rule configuration, which helps organize clustered data into short alignment rules.

The method to generate snort rules by classified payloads in this system is based on a Regular Expression generator based on a genetic algorithm [12], which is an efficient Snort Regular Expression generator for massive strings. The complex payloads under the cluster can generate appropriate snort rules after several times selections, crossovers, and mutations. The single snort rule will eventually generate the form in Figure 3.

```

alert tcp any any -> <$HOME_NET> any(
sid:<$SSID_CNT>;
msg:<$CLUSTER_ID>;
pcre:"/<$PCRE_REGEX_RULE>/s";
rev:<$REV_NUM>
)

```

Figure 3. Snort rule with PCRE

TABLE 6. PARAMETER OF GENETIC ALGORITHMS

Snort Regular Expression Rule (SSH)	
Initialization	Random
Parent selection	Roulette wheel Selection
Generation	30
Population	100

To verify whether the rules generated by the above generator can effectively intercept attack packets, the corresponding environment and scripts are required for detection and testing.

Therefore, the test script is written by python scapy. It will reproduce attack packets and patch the src_ip and src_port to the target Snort server to simulate the attack activity.

The detailed environment structure is shown in Figure 4. The snort firewall is independently set up on an intranet. The rule set generated by the Snort Regular Expression generation is added to the snort configuration file. The network card and IP range are specified to start attack detection. After the config is completed, run the scapy script on the test computer under the same intranet to send out test packets continuously, and check whether the corresponding packet information is recorded in the snort firewall log file.

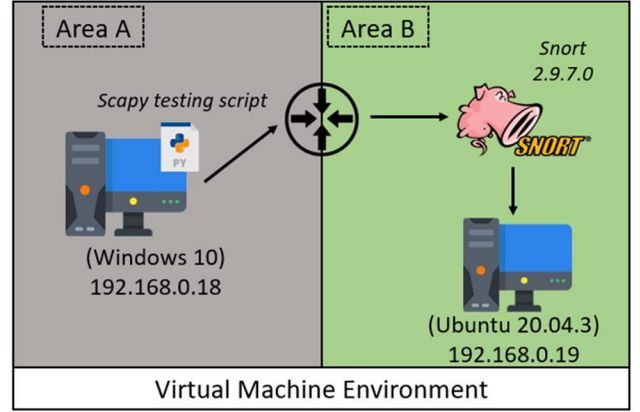


Figure 4. Environment structure

IV. PERFORMANCE ANALYSIS

In this study, attack packets are collected by Honeybot from 9 ISPs in Taiwan. Table 7 shows the percentage of SSH interactions collected by Cowrie Honeybot for one week and ranked by K-means clustering and feature labeling regarding MITRE ATT&CK. The Cluster ID: 31 and Cluster ID: 30 represent "brute force password cracking" and "simple SSH connection." These two clusters occupy most of the attack behaviors. Malicious score and attack behavior also show an inversely proportional trend. That is, if a complete study on high-threat attacks is needed in the future, the groups with a small percentage should be paying more attention.

TABLE 7. WEEK-LONG SSH INTERACTION DATA

Cluster ID	Packet number	Percentage	Malicious Score
31	51648542	74.90%	0.6
30	17095046	24.80%	0.6
5	122275	0.18%	0.6
11	38150	0.06%	0.6
18	8313	0.01%	1
13	6917	0.01%	0.8
24	5202	0.01%	0.8
22	3675	0.01%	0.8
3	2892	0.004%	0.8
14	2253	0.003%	0.6
12	652	0.001%	0.8
1	549	0.001%	0.8

15	343	0.0005%	1
16	307	0.0004%	0.8
17	305	0.0004%	0.8
4	187	0.0003%	0.8
9	148	0.0002%	0.8
2	119	0.0002%	0.8
7	117	0.0002%	0.8
10	57	0.0000083%	1
20	43	0.0000062%	0.8
6	42	0.0000061%	0.8
19	28	0.0000041%	0.8
8	13	0.0000019%	0.8
28	11	0.0000016%	0.8
23	10	0.0000015%	0.8
21	7	0.000001%	0.8
26	2	0.0000003%	0.8
25	1	0.00000015%	0.8

This study attempts to use the Within-cluster sum of squares (WCSS) [13] to find the appropriate K in the K-means algorithm. The mathematical formula for WCSS is as follows:

$$WCSS = \sum_{k=1}^K \sum_{i \in A_k} ||y_i - c_k||^2 \quad (2)$$

Among them, A is the K cluster division represented by the vector y_i in the feature space of multiple dimensions, which is composed of non-null and non-overlapping clusters A_k , and each cluster has a center point c_k ($k = 1, 2 \dots K$).

Figure 5. shows the K value after about 20. The slope obtained by WCSS gradually slows down. After our analysis and observation of the 450,000 data, the first few large groups account for a high proportion of the total 450,000 data, and the similarity is very high. Suppose the value of K is too large. In that case, the remaining groups with small proportions will only be divided into more pieces, which is not conducive to analysing and researching aggressive attack behavior. Therefore, The K value around 30 is appropriate for the K-means algorithm clustering in this study.

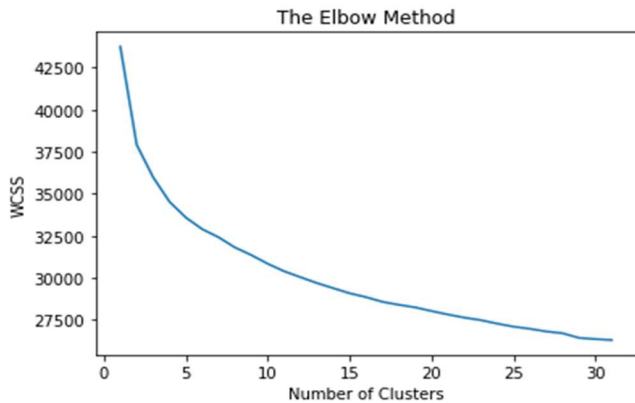


Figure 5. The Elbow Method

Table 8 is the confusion matrix of the data through the K-means algorithm. TP is the number of packets that are grouped correctly after the algorithm, FP is the number of packets that

are incorrectly grouped, FN is the number of packets that cannot be grouped but can be classified into clusters, and TN is the number of packets that cannot be grouped after the algorithm judges.

TABLE 8. CONFUSION MATRIX OF THE DATA THROUGH THE K-MEANS ALGORITHM

	True Yes	True No
Predict Yes	TP 68,575,748	FP 245,758
Predict No	FN 98,942	TN 15,776

Table 9 shows the accuracy rate obtained by K-means and the coverage rate of the attack pattern by the snort ruleset. Experiments have confirmed that the coverage rate can reach 100%; it can perfectly filter all kinds of attacks collected by Cowrie. Establish a suitable defense mechanism.

TABLE 9. WEEK-LONG SSH TESTING INFO

Snort Regular Expression Rule (SSH)	
Cluster num	31
Packet number	68936206
Abnormal Clustering	245683
Accuracy of Clustering	0.995
Rule number	198
Generate Time (100 core)	9.61 sec
AP Coverage	98%

V. CONCLUSIONS

This study proposes an intelligence defense system that can automatically generate snort firewall rules. It eliminates the omissions and inefficiencies caused by the misconfiguration of handwritten firewall rules. Applying the K-means algorithm and MITRE ATT&CK can quickly identify high-threat attacks, which are valuable for intensive study. Besides, classifying massive attacks in small groups dramatically improves the efficiency of generating snort rules. The verification shows that the attack coverage rate of snort rules generated by this system can reach 98%, which can block attacks effectively. In summary, with the increasing number of network attacks and the emergence of new attacks, finding a way to protect devices efficiently is necessary. The system proposed in this thesis can continuously discover new kinds of attacks and finally generate corresponding firewall rules to provide real-time protection.

ACKNOWLEDGMENT

Thank for the research support from the National Science and Technology Council, Taiwan, with the Grant Numbers: 111-2221-E-011 -114 -MY2(National Taiwan University of Science and Technology) and 111-2221-E-143 -006 -MY2(National Taitung University).

REFERENCES

- [1] Wool, "A Quantitative Study of Firewall Configuration Error," in *Computer*, vol. 37, no. 6, pp. 62-67, June 2004
- [2] Wool, "Trends in Firewall Configuration Errors - Measuring the Holes in Swiss Cheese," in *IEEE Internet Computing*, vol. 14, no. 4, pp. 58-65, July-Aug. 2010
- [3] (2019) Verizon. Data Breach Investigations Report, [Online]. Available: <https://www.verizon.com/business/resources/reports/dbir/2019/summary-of-findings/>
- [4] (2019) The Cloudflare Blog [Online]. Available: <https://blog.cloudflare.com/cloudflare-outage/>
- [5] S. Udhani, A. Withers and M. Bashir, "Human vs Bots: Detecting Human Attacks in a Honeypot Environment," *Proceedings of the 7th International Symposium on Digital Forensics and Security*, pp. 1-6, 2019.
- [6] M. Başer, E.Y. Güven, M.A. Aydın, "SSH and Telnet Protocols Attack Analysis Using Honeypot Technique: Analysis of SSH AND TELNET Honeypot," *6th International Conference on Computer Science and Engineering (UBMK)*, pp. 806-811, 2021.
- [7] Fraunholz, D. Krohmer, S. D. Anton and H. Dieter Schotten, "Investigation of cyber crime conducted by abusing weak or default passwords with a medium interaction honeypot," *2017 International Conference on Cyber Security And Protection Of Digital Services (Cyber Security)*, 2017, pp. 1-7
- [8] J.E. Ricardo, J.J.D. Menéndez , I.F.B. Arias , J.M.M. Bermúdez and N.M. Lemus, "Neutrosophic K-means for the analysis of earthquake data in Ecuador," *Neutrosophic Sets and Systems*, Vol 44, Article 29, 2019.
- [9] J. Hutagalung, N.L.W.S. Ginantra, G.W. Bhawika , W.G.S. Parwita, A. Wanto and P.D. Panjaitan , "COVID-19 Cases and Deaths in Southeast Asia Clustering using K-Means Algorithm," *Phys.: Conf. Ser.* 1783 012027, 2021.
- [10] B.E. Strom, A. Applebaum, D.P. Miller, K.C. Nickels, A.G. Pennington, and C.B. Thomas, *Mitre ATT&CK: Design and Philosophy*, MITRE PRODUCT, MP180360R1, 2018.
- [11] E. Satria, T. P. S. Huda, M. Iqbal, F. W. Sarjana, "The investigation on cowrie honeypot logs in establishing rule signature snort," *Conf. Ser.: Earth Environ. Sci.* 644 012031, 2021.
- [12] P.J. Chuang, P.C. Chao, H.C. Lu, "Regular Expression Generator by using Genetic Algorithm," [Online]. Available: <https://github.com/maojui/Regex-Generator/blob/master/Regular%20Expression%20Generator%20by%20using%20Genetic%20Algorithm.pdf>, 2020.
- [13] T. M. Kodinariya and P. R. Makwana, "Review on Determining of Cluster in K-means Clustering," *International Journal of Advance Research in Computer Science and Management Studies*, November 2013.



Yi-Cheng Lai was born in Taipei, Taiwan, in 1998. He received his B.S. degree in 2021. He is pursuing an M.S. degree in electrical engineering from the National Taiwan University of Science and Technology, Taipei. His main research interests include cyber security, vulnerability research, and defense.



Chiao-Lin Yu was born in Taipei, Taiwan, in 1997. He received his M.S. degree in electrical engineering from the National Taiwan University of Science and Technology, Taipei, Taiwan, in 2022. His main research interests include cyber security, vulnerability research, and defense.



Man-Ling Liao was born in Tainan, Taiwan, in 1998. She received her B.S. degree in 2021. She is pursuing an M.S. degree in electrical engineering from the National Taiwan University of Science and Technology, Taipei. Her main research interests include data analysis, machine learning, and cyber threat intelligence.



Yu-Shan Lin is a professor in the Department of Information Science and Management Systems, National Taitung University, Taitung, Taiwan. She received her Ph.D. degree from National Sun Yat-sen University, Kaohsiung, Taiwan. Her research interesting areas include e-Learning, Information Technology Education, and Internet Marketing. She has published about 20 journal papers, and some are highly-cited papers. Moreover, she received the Best Paper Award from IEEE ICASI 2018, IC3 2018, 2019 and TANET 2020, 2021.



Yao-Chung Chang (Member, IEEE) is a professor in the Department of Computer Science and Information Engineering, National Taitung University, Taitung, Taiwan. He received a Ph.D. degree in Computer Science and Information Engineering from National Dong Hwa University (NDHU), Hualien, Taiwan. Also, he received the Outstanding Teaching Award from Nation Taitung University. His main research interests include Information & Network Security, Cloud & Mobile Computing, AIoT & Machine Learning.



Jiann-Liang Chen (Senior Member, IEEE) was born in Taiwan in December 1963. He received his Ph.D. in electrical engineering from the National Taiwan University, Taipei, Taiwan, in 1989. Since August 1997, he has been with the Department of Computer Science and Information Engineering, National Dong Hwa University, where he is a Professor and the Vice Dean of the Science and Engineering College. He joins the Department of Electrical Engineering, National Taiwan University of Science and Technology, as a Distinguished Professor. Also, Prof. Chen is the Dean, College of Electrical Engineering and Computer Science, National Taiwan University of Science and Technology. His current research interests include cellular mobility management, cyber security, personal communication systems, and the Internet of Things (IoT).