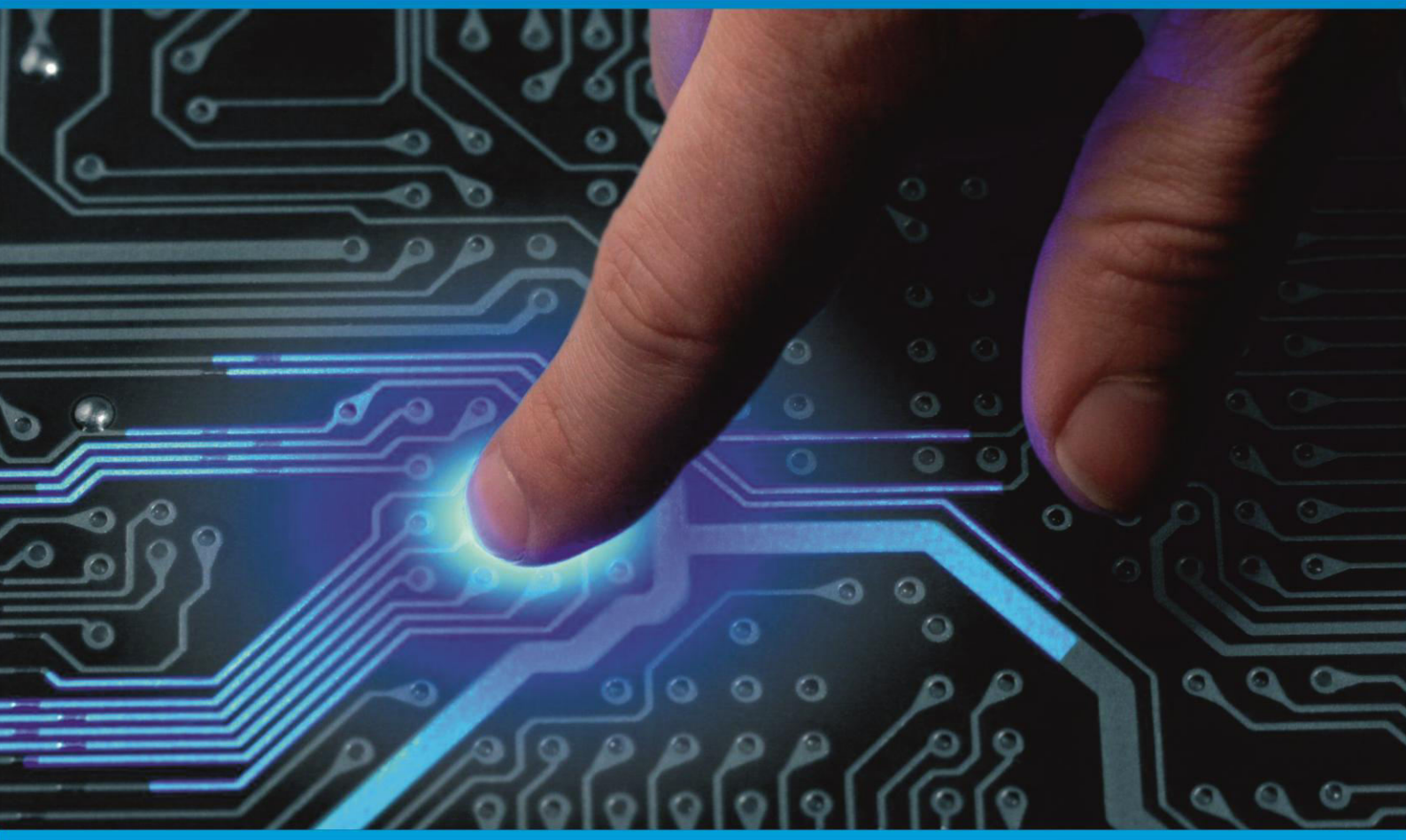
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 9, Issue 5, May 2021

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.488

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.488 |

|| Volume 9, Issue 5, May 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0905144|

Detecting Spammers on Social Networks Applied for Machine Learning Algorithm

Dr.K.Ramar[1] ., M.Renuka[2]., R.Sangeetha[3]., A.Gowsalya[4].,

Dean R&D, Department of Computer Science and Engineering, Muthayammal Engineering College (Autonomous), Tamilnadu, India[1]

UG Student, Department of Computer Science and Engineering, Muthayammal Engineering College (Autonomous), Tamilnadu, India[2]

UG Student, Department of Computer Science and Engineering, Muthayammal Engineering College (Autonomous), Tamilnadu, India[3]

UG Student, Department of Computer Science and Engineering, Muthayammal Engineering College (Autonomous), Tamilnadu, India[4]

ABSTRACT: On-line Social Networks (OSNs) are increasingly influencing the way people communicate with each other and share personal, professional and political information. Increasing reports of the security and privacy threats in the OSNs is attracting security researchers trying to detect and mitigate threats to individual users. With many OSNs having tens or hundreds of million users collectively generating billions of personal data content that can be exploited, detecting and preventing attacks on individual user privacy is a major challenge. Most of the current research has focused on protecting the privacy of an existing online profile in a given OSN.

A number of fake account detection approaches rely on the analysis of individual social network profiles, with the aim of identifying the characteristics or a combination thereof that help in distinguishing the legitimate and the fake accounts. Machine learning classification algorithms were used to decide the target accounts identity real or fake, those algorithms were support vector machine (SVM), neural Network (NN), and our newly developed algorithm, SVM-NN.

KEYWORDS Social networks, Privacy, Web crawling, Data breaches, Graph theory

I. INTRODUCTION

Social networks provide a way for users to remain in contact with their friends. The increasing popularity of social networks allows social site users to gather large amounts of individual information about their friends. Among numerous sites, Twitter is the fastest growing website. Its popularity has also attracted many spammers to use large amounts of spam to penetrate legitimate users' accounts.

The proposed spam detection system can obtained higher accuracy precision, recall and F-measure compared to the existing classifiers such as naïve Bayes and Support vector machine (SVM). All these usual methods depend on the experience of the user to determine the spam manually. But, nowadays, we need some automatic tools for the detection and close up spammers account. In addition, we need more accurate but effective spam detection methods to avoid dissatisfaction with legitimate users. It is a supervised learning approach, which is used to resolve the regression as well as classification problem. In this approach, the data is plotted in the n-dimensional space and each data comprises of feature value along with their coordinate. The hyperplane has been determined to differentiate normal and spam text. In this research, SVM is used to separate the category of normal text from the spam text.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.488 |

|| Volume 9, Issue 5, May 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0905144|

II. METHOLOGY

2.1 Neural network (NN)
2.2 Support vector machine (SVM)
2.3 Friend-list Recovery Attack
2.4 Profile Recovery Attack
2.5 Spam Bot Analysis

2.1 Neural network (NN)

The datasets that we have used cannot be classified using linear classifier. So, non- linear classifier with Gaussian kernel is used. They applied Neural Networks with resilient back propagation (Rprop) and SVM with C-support vector classification and polynomial kernel (polydot) as kernel functions. The findings of this paper showed that using PCA as a dimension reduction produce better accuracy results than using all the features without any selection.

2.2 Support vector machine (SVM)

An SVM classifies data by finding the best hyperplane that separates all data points of one class from those of the other class. The best hyperplane for an SVM means the one with the largest margin between the two classes. An SVM classifies data by finding the best hyperplane that separates all data points of one class from those of the other class. The support vectors are the data points that are closest to the separating hyperplane.

2.3 Friend-list Recovery Attack

This attack enables an adversary to recover a target's friend-list that the target explicitly configures to protect. An adversary initially discovers some of the target's friends, and recursively gathers their friend-lists to successively recover the target's friend-list.

2.4 Profile Recovery Attack

This attack discloses a user's preferences in his profile page. The adversary first collects a victim's friend-list, which we assume to be accessible, for example by launching the friend-list recovery attack. The adversary then utilizes web scraping for collecting the profiles of the victim's friends. The group member and like can potentially leak sensitive information so that the adversary can identify members of the groups and users who are interested in the objects.

2.5 Spam Bot Analysis

The spam bots that we identified showed different levels of activity and different strategies to deliver spam

2.5.1. Displayer: Bots that do not post spam messages, but only display some spam content on their own profile pages. In order to view spam content, a victim has to manually visit the profile page of the bot. This kind of bots is likely to be the least effective in terms of people reached. All the detected MySpace bots belonged to this category, as well as two Face book bots.

2.5.2. Bragger: Bots that post messages to their own feed. These messages vary according to the networks: on Face book, these messages are usually status updates, while on Twitter these are the tweets. The result of this action is that the spam message is distributed and shown on all the victims' feeds.

2.5.3. Poster: Bots that send a direct message to each victim. This can be achieved in different ways, depending on the social network. On Face book, for example, the message might be a post on a victim's wall. The spam is shown on the victims feed, but, unlike the case of a "bragger", can be viewed also by victim's friends visiting her profile page. This is the most effective way of spamming, because it reaches a greater number of users compared to the previous two. Eight bots from this category have been detected, all of them on the Face book network.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.488 |

|| Volume 9, Issue 5, May 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0905144|

III. ARCHITECTURE OF THE SYSTEM

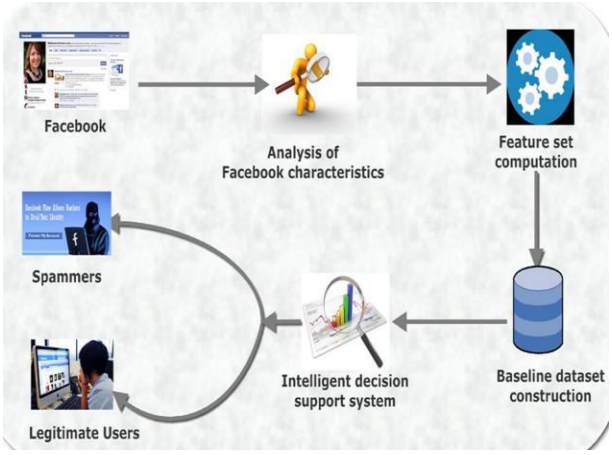
In short text such as in Twitter and SMS domain, spam detection is more challenging problem due to noisy and small size of these messages. We proposed a novel deep neural architecture combing CNN and LSTM. In the proposed approach, we include semantic information in sentence representation. We have proposed an Adaptive Privacy Policy Prediction (A3P) system that helps users automate the privacy policy settings for their uploaded images. The A3P system provides a comprehensive framework to infer privacy preferences based on the information available for a given user. We also effectively tackled the issue of cold-start, leveraging social context information. Our experimental study proves that our A3P is a practical tool that offers significant improvements over current approaches to privacy. They also showed how their techniques help to detect spam profiles even when they do not contact a honey-profile. They believe that these techniques can help social networks to improve their security and detect malicious users. In fact, they develop a tool to detect spammers on Twitter.

In particular, if an administrator's privilege is acquired by an attacker through a privilege escalation attack, the attacker can operate the entire system and the system can suffer serious damage. In this paper, an additional kernel observer (AKO) method is proposed. It prevents privilege escalation attacks that exploit operating system vulnerabilities. We focus on the fact that a process privilege can be changed only by specific system calls. Privilege escalation attack (PEA) is one of the serious threats. Privilege escalation attacks constitute a type of attack that exploits OS vulnerabilities.

In a privilege escalation attack, an attacker can promote the privilege of a process to a higher level by exploiting OS vulnerability. The attacks targeted by AKO are those that exploit Linux kernel vulnerabilities and tamper with the privilege data of processes stored in the kernel space during system call processing. Because the process privileges are stored in the kernel space and cannot be referenced by user applications in the user space, they cannot be directly tampered with by applications executed at the user level.

The proposed method can prevent privilege escalation attacks that tamper with privilege information during system call processing. However, there are privilege escalation attacks that do not tamper with privilege information during system call processing, and these attacks cannot be prevented by our method. AKO holds the data of privilege information during the system call processing. Thus, to mitigate AKO, an attacker must tamper with both the privilege information of the running process and the stored privilege information. In addition, the attacker must tamper with it during one system call processing. Either altering the privilege information of the process table or altering the stored privilege information would result in the detection of AKO.

Figure 3.1: Architecture of The System



| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.488 |

|| Volume 9, Issue 5, May 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0905144|

3.2 Privilege Escalation Attacks

A privilege escalation attack enables an attacker to gain illegally elevated access to resources by exploiting a bug, design flaw, etc. In real attacks, many attackers acquire the administrator's authority. By acquiring administrative privileges, the attacker can operate the entire system, which can result in information leakages and denial of service.

A privilege escalation attack (PEA) is all about acquiring unauthorized system rights .It is illicit intrusion on a system, application or network in which the design flaws and program errors are exploited to attain elevated access to the programs and resources the system holds.

A privilege is security characteristics attributed to a user access level, needed to perform particular operations. Privilege escalation attacks occur not only from infiltration from outside the network but within the network as well. Authorized users of the system may try to attain increased privileges to gain access to areas or levels in the systems network for which they are not authorized.
In either case, these attacks are dangerous and can be extremely harmful to any system or program.

IV. OPERATION AFTER PRIVILEGE ESCALATION ATTACK DETECTION

4.1 Invalidation of privilege escalation attacks: It is important to prevent an attacker from acquiring administrative privileges. AKO overwrites the privilege information using the stored privilege information when it detects a privilege escalation attack.

4.1 Termination of a running process: An attacker's process often executes a shell or other malicious program with the root privilege; then, the executed program achieves the attacker's purpose, such as information theft.

4.2 Avoiding Privilege Escalation Vulnerabilities

Privilege escalation vulnerabilities may arise for different reasons:

- **Programming errors**: This includes vulnerabilities that lead to web attacks but also other vulnerabilities such as buffer overflow.
- **Mis configurations:** Especially risky when the principle of least privilege is not followed and normal users have too many privileges.
- **Lack of security hygiene:** For example, delayed patches and updates for the operating system and other software.
- **Social engineering**: Attackers may gain access to accounts by exploiting gullible users.

4.3 Injection through cookies

Cookies are structures that maintain persistence for web applications by storing state information on the client's machine. To restore a session, a web application often retrieves the cookie from the client. Since the client has complete control over cookies, an attacker could craft cookies such that when the application server builds an SQL query based on the cookies content, the structure and function of the query is modified.

4.4 Cookie:

A cookie, also known as an HTTP cookie, web cookie, or browser cookie, is used for an origin website to send state information to a user's browser and for the browser to return the state information to the origin site. The state information can be used for authentication, identification of a user session, user's preferences, shopping cart contents, or anything else that can be accomplished through storing text data. Cookies are not software. They cannot be programmed, cannot carry viruses, and cannot install malware on the host computer. However, they can be used by spyware to track user's browsing activities – a major privacy concern that prompted European and US law makers to take action. Cookies could also be stolen by hackers to gain access to a victim's web account.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.488 |

|| Volume 9, Issue 5, May 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0905144|

4.5 Limitations:

Web application vulnerability scanners are not always capable of detecting all of the vulnerabilities and attack vectors that exist. In consequence, they may assert numerous false-negatives and false-positives.

4.6 SQL Injection

A SQL injection attack consists of insertion or "injection" of a SQL query via the input data from the client to the application. A successful SQL injection exploit can read sensitive data from the database, modify database data (Insert/Update/Delete), execute administration operations on the database (such as shutdown the DBMS), recover the content of a given file present on the DBMS file system and in some cases issue commands to the operating system.

4.7 Second-Order Injection

Second-order injection occurs when incomplete prevention mechanisms against SQL injection attacks are in place. In second-order injection, a malicious user could rely on data already present in the system or database to trigger an SQL injection attack, so when the attack occurs, the input that modifies the query to construe an attack does not come from the user, but from within the system itself.

4.8 Session Hijacking

The session hijacking is also refers to cookies hijacking. And it exploits the valid computer session. By the session key attacker gains the unauthorized access to the information and services to the computer system. This deals with monitoring the activities of the users until they sign in to the account or transaction and create their important information. The infected software will perform unauthorized actions in the user's account, such as transferring funds from user account to the attackers account, without the user's knowledge.

TCP session hijacking is a security attack on a user session over a protected network. The most common method of session hijacking is called IP spoofing, when an attacker uses source-routed IP packets to insert commands into an active communication between two nodes on a network and disguising itself as one of the authenticated users. This type of attack is possible because authentication typically is only done at the start of a TCP session. Another type of session hijacking is known as a man-in-the-middle attack, where the attacker, using a sniffer, can observe the communication between devices and collect the data that is transmitted.

Detecting Click-Spam on Face book Ads So far, we have discussed the performance of our anomaly detector in detecting diverse attack strategies. Next, we demonstrate another real world application of our technique: detecting click-spam on Face book Click-Spam in Face book To gain a preliminary understanding of Face book click spam, we signed up as an advertiser on Face book. We set up an ad campaign targeting users in the USA aged between 15 and 30. The campaign advertised a simple user survey page about Face book's privacy settings. When clicked, the ad leads to our heavily instrumented landing page to capture any user activity such as mouse clicks, mouse movement, or keyboard strokes. To distinguish between unintentional clicks and intentional clicks followed by lack of interest in our page, we ran Bluff that are ads with identical targeting parameters as the original ad, but nonsensical content. Our bluff ad content was empty. Shows that our bluff ad performed identically to the original ad, both qualitatively and quantitatively; of 301 clicks in roughly the same time-frame as the original ad, almost 30% did not complete first HTTP, etc. Anomalous Clicks in Face book Ads In order to analyze anomalous user behavior; our approach requires information from the user's profile. Due to a change in how Face book redirects users on ad clicks [we were unable to identify the users that clicked on our ad in the experiment above. Fortunately, Face book offers a different type of ad campaign optimization scheme—maximizing likes—where the destination must be a Face book page as opposed to an arbitrary website. With such ads, it is possible to identify the users that clicked on such an ad, but not possible to instrument the landing page to get rich telemetry as above. We chose this campaign optimization option for maximizing likes to the advertised page.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.488 |

|| Volume 9, Issue 5, May 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0905144|

4.9 Using Packet Sniffers

It can be seen that attack captures the victim's session ID to gain access to the server by using some packet sniffers.

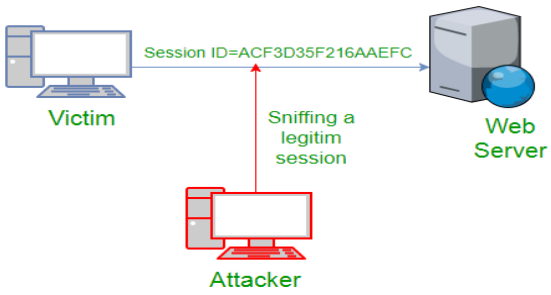


Figure:4.9 Using Packet Sniffers

4.10 Cross Site Scripting (XSS Attack)

Attacker can also capture victim's Session ID using XSS attack by using JavaScript. If an attacker sends a crafted link to the victim with the malicious JavaScript, when the victim clicks on the link, the JavaScript will run and complete the instructions made by the attacker.

V. AKO PROCESS

AKO to prevent privilege escalation attacks by focusing on privilege changes during system call processing. AKO detects privilege escalation attacks by checking the change of the privilege after system call processing is completed, whereby the invoked system call does not change the privilege of processes. At the time of detection, AKO issues a warning to the administrator. Subsequently, the process can be terminated or the change of the privilege data can be invalidated by overwriting the original privilege data.

- AKO is based on the following features of privilege data management and system calls.
- The privilege of the process is stored in the kernel space.
- To modify data in the kernel space, it is necessary to proceed through a system call.
- The function of each system call is subdivided. By applying AKO

We can prevent a privilege escalation attack that changes privilege information that should not be changed in the original system call processing, regardless of the content of the vulnerabilities.

VI. CONCLUSIONS

We proposed AKO, a method that prevents privilege escalation attacks from exploiting Linux kernel vulnerabilities. We described its implementation and presented evaluation results. AKO is based on the following features of privilege data management and system calls. First, the privilege of the process is stored in the kernel space. Second, to modify data in the kernel space, it is necessary to proceed through a system call. Third, the function of each system call is subdivided. By applying AKO, we can prevent a privilege escalation attack that changes privilege information that should not be changed in the original system call processing, regardless of the content of the vulnerabilities. We implemented AKO and performed experiments involving privilege escalation attacks using exploit codes available on the web. The evaluation results showed that AKO could detect multiple types of privilege escalation attacks. Thus, AKO can adopt countermeasures against privilege escalation attacks before damage to the system occurs.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.488 |

|| Volume 9, Issue 5, May 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0905144|

REFERENCES

1. Akira Yamada, Tiffany Hyun-Jin Kim and Adrian Perrig,(2012), "Exploiting Privacy Policy Conflicts in Online Social Networks", IEEE.
2. Bimal Viswanath, Mark Crovella, Mark Crovella, Ahmad Bashir .M (2017), "Indoor Towards Detecting Anomalous User Behavior In Online Social Networks" , IEEE.
3. Carlini . N and Wagner.D,(2014), ''ROP is still dangerous: Breaking moderndefenses,'' inProc. 23rd USENIX Secur. Symp. (USENIX Secur.),,pp. 385–399.
4. Castro .M, Costa .M, and Harris .T, (2006),''Securing software by enforcing data-flow integrity,'' inProc. 7th Symp. Oper. Syst. Design Implement.,pp. 147–160.
5. Chen .S, Xu .J, Sezer .E.C, Gauriar .P, and Iyer .R.K,(2005),''Non-control- data attacks are realistic threats,'' inProc. 14th USENIX Secur. Symp. (USENIXSecur.), pp. 177–192.
6. Criswell.J, Dautenhahn.N and Adve.V(May 2014),''KCoFI: Complete control-flowintegrity for commodity operating system kernels,'' inProc. IEEE Symp.Secur. Privacy, pp. 292–307.
7. Ge .X, Talele .N, Payer .M, and Jaeger .T, (March 2016),''Fine-grained control-flowintegrity for kernel software,'' inProc. IEEE Eur. Symp. Secur. Privacy, pp. 179–194.
8. Giuffrida .M, Kuijsten .A, and Tanenbaum .A.S,(2012),''Enhanced operating system security through efficient and fine-grained address space random-ization,'' inProc. 21st USENIX Secur. Symp. (USENIX Secur.), 475–490.

INNO SPACE
SJIF Scientific Journal Impact Factor

Impact Factor:
7.488

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर
NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details