



ISSN: 2522-3429 (Print)
ISSN: 2616-6003 (Online)

International Journal for Electronic Crime Investigation (IJECI)



VOL: 6
ISSUE: 3 Year 2022

Email ID: ijeci@lgu.edu.pk

Digital Forensics Rcsarch and Service Center
Lahore Garrison University, Lahore, Pakistan.

LGU International Journal for Electronic Crime Investigation

Volume 6(3) Year (2022)

SCOPE OF THE JOURNAL

The IJEI is an innovative forum for researchers, scientists and engineers in all domains of computer science and technology to publish high quality, refereed papers. The journal offers articles, survey and review from experts in the field, enhancing insight and understanding of the current trends and state of the art modern technology. Coverage of the journal includes algorithm and computational complexity, distributed and grid computing, computer architecture and high performance, data communication and networks, pattern recognition and image processing, artificial intelligence, cloud computing, VHDL along with emerging domains like Quantum Computing, IoT, Data Sciences, Cognitive Sciences, Vehicular Automation. Subjective regime is not limited to aforementioned areas; Journal policy is to welcome emerging research trends in the general domain of computer science and technology.

SUBMISSION OF ARTICLES

We invite articles with high quality research for publication in all areas of engineering, science and technology. All the manuscripts submitted for publication are first peer reviewed to make sure they are original, relevant and readable. Manuscripts should be submitted via email only.

To submit manuscripts by email with attach file is strongly encouraged, provided that the text, tables, and figures are included in a single Microsoft Word/Pdf file. Submission guidelines along with official format is available on the following link; www.research.lgu.edu.pk

Contact: For all inquiries, regarding call for papers, submission of research articles and correspondence, kindly contact at this address:

IJEI, Sector C, DHA Phase-VI Lahore, Pakistan

Phone: +92- 042-37181823

Email: IJEI@lgu.edu.pk

LGU International Journal for Electronic Crime Investigation
Volume 6(3) Year (2022)

CONTENTS

Editorial

Dr. Syeda Mona Hassan

Nanotechnology: An Emerging Technology In Crime Investigation 01-02

Research Article

Prof Dr Aftab Ahmad Malik, Dr Waqar Azeem and Engineer Dr Mujtaba Asad
Child Kidnapping and Abuse by Gang-criminals and the Legitimate Custody
of Minor to Parents After Rescue and use of Geofencing to Arrest the
Absconding Criminals 03-10

Research Article

Talha Ashfaq and Muhammad Shairoze Malik

Forensics Artifacts on Remote Desktop Protocol and Service 11-18

Research Article

Asif Ibrahim and Syed Khurram Hassan

Forensic toxicology: importance in crime investigation 19-28

Research Article

Muhammad Saad and Muhammad Taseer

Study of the Anti-Debugging Techniques and their Mitigations 29-40

LGU International Journal for Electronic Crime Investigation
Volume 6(3) Year (2022)

Patron in Chief: Maj General (R) Shehzad Sikander, HI(M)
Vice Chancellor Lahore Garrison University

Advisory Board

Mr. Kaukab Jamal Zuberi, Director DFRC, Lahore Garrison University
Mr. Şukru Durmaz, CEO DIFOSE, Turkey
Dr. Fahad Pervaiz, Research Wing Microsoft, Seattle, USA.
Dr. Mansoor Pervaiz, Manager Software IBM, Boston, USA
Dr. Dil Muhammad, Dean LAW Department, University of South Asia.

Editorial Board

Dr. Aftab Ahmed Malik, University of Kent, England.
Dr. Ejaz Hussain, University of Pennsylvania, USA.
Prof. Dr. Shahid Tufail, PCSIR, Lahore
Prof. Dr. Saqib Shehzad, Institute of Forensic Science, Istanbul University, Turkey.
Prof. Dr. Shahana Ehsan, LCWU Lahore
Dr. Mujtaba Asad, Shanghai Jiao Tong University, China
Dr. Waqar Azeem, South Eastern Regional College, Downpatrick, Ireland UK
Prof. Dr. Ghazala Akram, University of Punjab, Lahore
Dr. Zahida Perveen, Lahore Garrison University.
Dr. Beenish Zehra, Lahore Garrison University
Dr. Ahmed Naeem, Lahore Garrison University
Dr. Tahir Alyas, ORIC Director, Lahore Garrison University

Chief Editor:

Dr. Syeda Mona Hassan, Lahore Garrison University

Assistant Editor:

Mr. Sajjad Sikandar, Lahore Garrison University
Mr. Qais Abaid, Lahore Garrison University

Reviewers Committee:

Prof. Dr. Peter John, GC University, Lahore
Prof. Dr. M. Pervaiz Khurshid, Govt College Science, Lahore
Dr. Maasoomah Sadaf, University of Punjab, Lahore
Dr. Tariq Qamar, FCCU, Lahore
Dr. Haroon Ur Rasheed, University of Lahore
Dr. Munawar Iqbal, University of Education, Lahore
Dr. Saima Naz, University of Education, Lahore
Dr. Shagufta Saeed, UVAS, Lahore
Dr. Shazia Saqib, University of Central Punjab
Dr. Mohsin Javed, UMT, Lahore
Dr. Sumia Akram, University of Education, Lahore

Nanotechnology: An Emerging Technology In Crime Investigation

Chief Editor

Dr. Syeda Mona Hassan

In United States, Canada, Australia and Europe nanotechnology is extensively used in the area of detection and investigation of criminal cases and forensics. These techniques are employing nanotechnology very effectively and they give positive results of investigations. In Pakistan, white collar crimes, banking frauds and misappropriation in the entrepreneurs are growing rapidly. The detection of the forensic evidences are investigated by nanotechnology particularly regarding fingerprints, DNA and handwriting.

At present, nanotechnology is most efficiently applied in forensic toxicology for the detection and quantification of various toxic substances from forensically essential biological evidence such as saliva, urine, blood, hair, sweat, vitreous humor, and also from bone remains.

The advent of nanotechnology is taking control of atoms and molecules individually, modifying and placing them to use with an extraordinary degree of accuracy. Nanotechnology is a combination of various fields of Physical, Chemical, And Biological Sciences that study the phenomenon on the nanometer scale.

Nanotechnology is one of the most promising emerging technologies among the exist-

ed technologies. Forensic Science is utilized in the process of using specific techniques and methods for inspecting, gathering, and analyzing trace evidences at the scene of crime. Nowadays, nanotechnologies in Forensic Science can be helpful in the processes of investigation by making them faster, more effective, more accurate, more sensitive, and easy to apply.

Nanotechnology can be utilized in the field of Forensic Science, such as Microchip Technology, Nano-Manipulators, Nano-Imaging tools for visualization. Moreover, Nano-Platform separately or in combination with other technologies is supposed to have major application in security, drug screening, explosive detection and DNA analysis, forensic investigators will be able to perform complex investigations or to discover the smallest traces of evidence at the crime scene. Nano sprayers can be utilized to analyze even the smallest trace evidence on the crime scene and the lab-on-a-chip device is able to perform multiple investigations.

Advanced art analytical devices would be used in replacement of existing chemical labs. All such Nano-tools are very important in considering their well-known advantages of selectivity and sensitivity. To improve the security and to reduce the crime, nanotechnology has demonstrated its efficacy. Security problems including prevention from fraud, safety of citizen, prevention measures

against organized crime and terrorism.

Analysis of forensic evidences is used to ensure that whether suspect is innocent or criminal. In most cases, analysis becomes difficult due to trace amount of forensic material and so, it is hard to confirm the connection between crimes with suspect. Finally, case remains in pending stage or unsolved. Nanotechnologies play an important role in addressing current security apprehensions to resolve this issue. For example, nanotechnology-based techniques improve analytical method of physical evidences present at the scene of crime which may not only transform the crime investigation process but also to the judiciary system. These innovative changes in forensic technology will continue to innovate forensic investigation and judiciary system along with the development of more tools to stop crime.

Moreover, use of art analytical techniques progresses the forensic science in locking up the suspect. Nano-analysis is one of the important nanotechnologies that is most frequently used in forensic science, including tools like the atomic force microscope (AFM), Raman micro spectroscopy, scanning electron microscope (SEM), and transmission Electron Microscope (TEM) (Micro-Raman).

In ancient times, criminologists have used fingerprints as a distinctive form of evidence. Recently, nanoparticles are considerably used in place of traditional materials like carbon black, aluminium

flake, and gentian violet which enhanced the sensitivity of the fingerprinting process. Even on a damaged and faded print, the nanoparticles can enhance the fingerprints by adhering to the ridges and grooves. The improvement and advancement of DNA analysis looks to be the most promising use of nanotechnology. Therefore, the current article is emphasized to explain the important role of nanotechnology in current forensic investigation issues.



Child Kidnapping and Abuse by Gang-criminals and the Legitimate Custody of Minor to Parents After Rescue and use of Geofencing to Arrest the Absconding Criminals

Prof Dr Aftab Ahmad Malik¹, Dr Waqar Azeem², and Dr Mujtaba Asad³

¹ Ph.D (University of Kent, England); M.Phil; MSc; LL.B.

Professor, Department of Criminology and Forensic Sciences,
Lahore Garrison University, Lahore

² South Eastern Regional College, Downpatrick, Ireland UK

³ Ph.D (Shanghai Jiao Tong University, China)

corresponding author: dr_aftab_malik@yahoo.com

Abstract:

This paper mainly focuses on the custody of a minor child, abducted and abused by the criminals from the place of parental residence. The cases regarding the kidnapping of minor young girls and teen aged women are increasing rapidly. The modus operandi of criminals vary from case to case. In some cases criminals torture the abducted minor, do rape and kill them; while in some other cases to escape from law enforcement agencies, they arrange “forced-illegal-fake” marriages under threat to life. In most of the offences an organized rich Gang of criminals is tangled and involved. The Gang members are given different responsibilities to perform and commit the crime by the leader of Gang. They have also the backing of most effective personalities, who safeguard them in existing system and such personalities remain effective and hidden. Therefore, the victim and the family do not get justice from any corner. The Gang and the “effective personality” invest huge amount of money “red light areas” or sale of limbs like kidney, heart or liver etc inland or abroad. However, the abductee undergoes a lot of torture and gang rape. Secondly, the authors focus and stress upon the need of strong legislation so that the criminals are punished, when their guilt is proved and also not bailed out in nonbailable offences. Thirdly and most importantly, the abductee must be handed over to the parents after recovery from the criminal Gang. This paper presents a case study of an abducted school-baby, who has not been handed over to the parents for the last six months. The faults of the legal system in Pakistan requires to be over viewed, revamped and overhauled. The antagonistic behavior of Police and investigating officers is also required to be modified, which adversely affects the merits of the case in courts of law.

Key words: Abduction, Kidnapping, custody of a minor abductee, rape, geofencing

1. Introduction

An exponential rise in the occurrence of crimes against teen aged girls and minor females have been observed during last few months. The criminals operate in the form of either an organized Gang or single handedly. While they operate as a Gang, they divide the offence into smaller categories of offences and nominate the members of the Gang different tasks to achieve their common objectives. The tasks are divided by chief of the Gang according to their competence and modus operandi to commit crime and archive their ulterior motive. For example, in the kidnapping and abduction cases, someone motivates the victim and after persuasion join hand with the other group to physically kidnap and transport to another destination. All or some members of the Gang commit rape and finally (perhaps) execute the killing or demand ransom from parents. At each stage all participants contributing in crime are fully safe by outside officials and unofficial protectors. Their links with some police officer and other effective personalities are very productive to safeguard them at all stages. Following terminologies are prevalent for the “criminal Gang “ such as syndicate, mafia or crime mob. They have an organized network like terrorists and behave like “sub-culture communal” sometimes called Gangland or underworld due to their hidden activities and operations. They possess most modern tools and equipment to achieve their objectives such as transport, wireless systems and “Information Technology Systems”. Regarding the topic of this research, according to [1], mostly they commit heinous and terrible offences such as kidnapping for ransom, sexual violence, Gang-rape, abuse, inflicting physical

torture to teenaged women and innocent girls.

2. Need for Strong Laws for Cases of Kidnapping

It is observed in all most all cases of abduction and kidnapping that the abductee undergoes tremendous amount of torture and sometimes gang rape. The authors focus and worry for the need of strong legislation so that the criminals are punished in accordance with nature of offence committed, when their guilt is proved and also not bailed out in nonbailable offences. Secondly, abduction is complete when the minor is transferred from the guardian's custody to that of a third party who is not entitled to it. Thirdly, according to the law, kidnapping is the crime of snatching, imprisoning, inveigling, or taking someone away by force or deceit, frequently for ransom money or in continuance of other crime. In [4] it is defined that child abduction is the illegal exclusion, retention, detention, or concealment of a kid or infant. A person is said to have been abducted if the person had been taken away using coercion, trickery, or overt force or violence. The loopholes, legal-gaps, ambiguities most of the time make case of prosecution weak in the court of law. Therefore, it is necessary to alter the most recent legal framework documents as described in [2]. Additionally, [3] strongly advises the calibration of forensic evidence, including its acquisition, conservation, and exhibition in court, adopting methodologies of FBI (Federal Bureau of Investigation). As a result, it is important to make the changes specified in [2] to the most recent legal instruments.

3. Criminology and Gang-led Organized crimes

The disciplines of Jurisprudence, Sociology of Law and Criminology are closely related with one another. The Jurisprudence deals with theory and development of Law on the basis of “what is right”, “what is wrong” and infringement of duties and rights, other areas discuss social norms, behaviors and justice system. The defects in one system directly affects the other adversely, for example corruption and bribe kill the essence of good governance and hence the ineffectiveness of these systems.

It is a well-known fact that the organized crimes occur and syndicates of criminals are operational in the countries performing various gang activities, which the naïve administration don’t effectively combat with them. Because organized crime advances due to the negligence of governments. Due to some incompetent employees/officers in the institutions and corruption, and the features of unsuccessful states, which are root cause of organized crime.

4. Case Study of a Kidnapped Minor Girl by Criminal-Gang

In this section, a case study is presented wherein several offences have been committed jointly by the 34-member criminal-gang having different modus operandi and role. These include offence under PPC sections 361, 362, 364, 364-A, 365, 365-B, 366-A 368 and 375 regarding kidnapping, abduction and rape. The members of Gang are very powerful, therefore for the last six months, they have upper hand and escaping from any punishment or detention. All criminals are on bail even for nonbailable offences. They are investing heavy amount of money on facing the trial by hiring several teams of lawyers and travel between province of Punjab and Sindh. In fact, a minor

girl, student of class 6 of an English medium School was abducted from her residence in Karachi by five members of the gang including one woman. She was smuggled after intoxication and drugs to Lahore, Punjab Province and later on, the gang kept on transporting the baby-girl to various cities including KPK, according to social media and court reporting. In this case the then Inspector of Police (IG) Sindh Province was ordered by the honorable court for recovery of minor-baby; due to noncompliance he was suspended or replaced from his position. The new Inspector General of Police had taken effective measure for the recovery of the minor-baby and successfully recovered her by Police. During the large time wasted in recovery of the child, the Gang managed to arrange a fake-paper-marriage of the minor under the age of marriage. However, the Police arrested the “Fake Maulvi member of gang” and a witness to the “Fake-paper-marriage”. Both the “fake-Maulvi” and witness denied their guilt in the court. They were bailed out along with two central criminals. The central criminals are real brothers and their mother “Noor” also accompanied them for smuggling the baby from Karachi to Punjab University Lahore, where the baby remained for three months under the “illegal custody” of Noor and the central criminals Zaheer and Shabbir. The minor baby was subjected to high degree of torture and intoxication as alleged in the court record. In spite of the arrest warrants, the woman “Noor” was never arrested apparently due to effectiveness of the gang and favorable system network. Police must use Geofencing technique to arrest Noor as her places in various cities are very well known to Police of Sindh and Lahore, where she often visits interchangeably.

5. Arrest of Criminals Using Geofencing Technology

It is important to introduce the Geofencing Technology and its applications, which is being widely used in Business. We propose its usage in crime world, for the localization of absconding criminals. This Technology has already been used in the detection of criminals who had murdered a famous and respected religious personality and member of parliament Maulana Sami-ul-Haqq. Also some other, criminals have been detected and arrested due to use of this Technology. According to [14], several applications in daily life, particularly in business can be effectively handled using Geofencing. The most useful aspect of Geofencing in the area of crime is that whosoever went near to marked location i.e.

crime-scene can be easily detected. According to [16], the Punjab Police is setting up and updating the present facilities to effectively utilize this technology in criminal detection and hence arrest. Police is also considering to strengthen Forensic systems regarding finger prints, detection of criminal using DNA, (CRD) Call Record Data storage and retrieval, geofencing, geo-mapping and the use of most relevant and applicable software. It is easy to understand the Geofencing Technology; as a software it creates a Virtual **Geo-boundary** everywhere round a real-location or object. The boundary can be expressed as map similar to google map. The working of the Geofencing depends upon area-based service. While working with geofencing, the following technologies shown in Table 1, which are used subject to requirements/applications:

Table 1: Technologies which can be interfaced with geofencing

Sr#	Tech-name	Uses
1.	<u>RFID</u>	Radio-frequency Identification works with electromagnetic field, which which can track the tags attached to it using a small radio transmitter and a receiver.
2.	<u>CRD</u>	Call Record Data is extremely useful for detection of criminals and can be use in conjunction with RFID for given GPS
3.	<u>GPS</u>	The Global Positioning System (GPS) is a U.S.-owned utility that provides users with positioning, navigation, and timing (PNT) services. This system consists of three segments: the space segment, the control segment, and the user segmen
4.	<u>WiFi</u>	cellular data to trigger a pre-programmed action when a mobile device or RFID tag enters or exits a virtual boundary set up around a geographical location, known as a geofence.

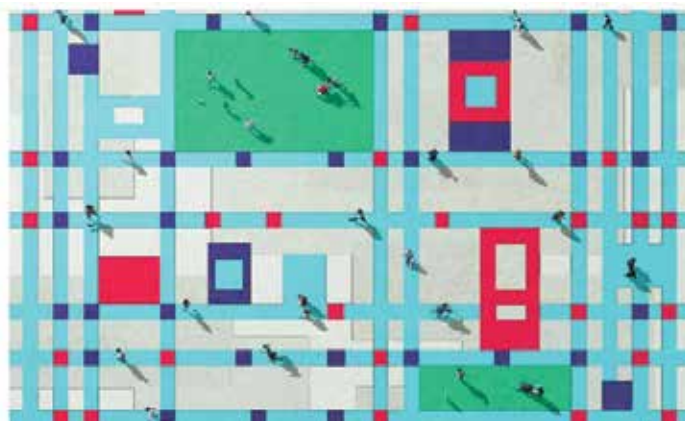


Figure 1: ‘Geofence’ Finds Anyone Who Went Near a Crime Scene. Curtesy Reference [15]

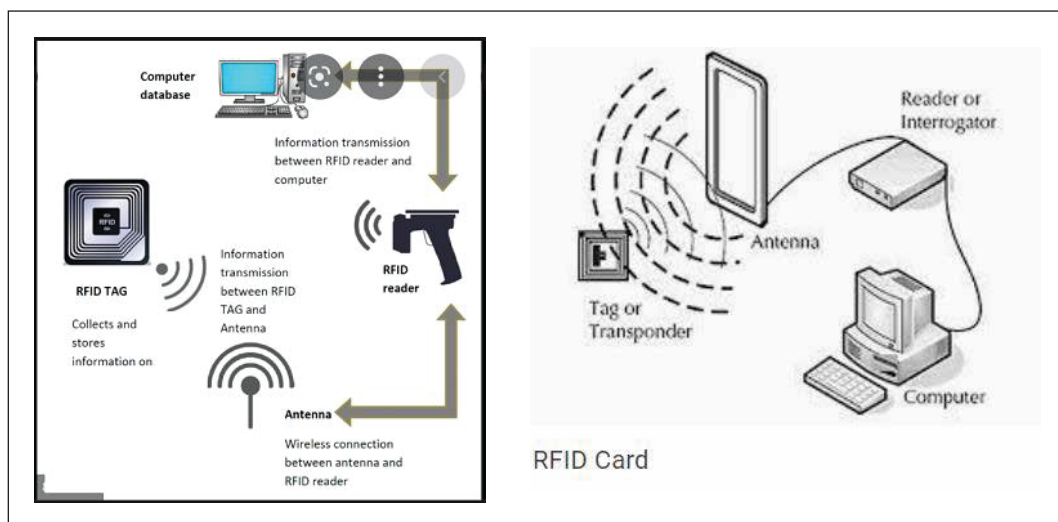


Figure 2: RFID and RIF Card . Reference Research Gate.net

6. Custody of a Minor Abductee and Legal Position

The practice in family courts is clear regarding the custody of a minor, when the dispute is between the parents as a result of separation or divorce. There are several decisions of the High Courts and Supreme Court in this regard, which are referred to by the lawyers. The courts then decide the matter regarding the custody, keeping in view the psychology and wellbeing of the minor.

In the case presented in section 4 of this paper, the most unfortunate situation has arisen particularly on the issue of the custody of the minor-baby. Since last six months, the custody of the minor-baby could not be handed over to the parents. She has been kept in a child protection home in Karachi. The gang being very effectively managed this aspect through the system to avert the custody to parents. Recently, the court of jurisdiction, decided to allow the parents to meet the minor-baby twice a month in the Child protection House. Two real

sisters aged 6 years and 2 years were also not allowed to meet the abductee, the minor-baby. Most importantly, the welfare of this child is being ignored totally, in particular the **studies and welfare**. She is not being provided the food she used to take at home. At the time of meeting with parents, there are two persons present, one lady doctor and other, a judge appointed by the court. The courts normally while deciding the custody, take into account the issues, such as age of the minor, the relationship of the child & parents and the welfare of the child. In case of the custody of the minor-baby discussed in section 4 of the case study, both mother and father are aggrieved. In such cases normally the court must hand over the custody to parents soon after the minor is rescued and recovered from the criminals. But for the last six months the issue is pending and minor baby is residing in the "Child Protection Home" at Karachi. The welfare of the minor and her schooling is at stake. She has lost one academic year of her education. The major reason for delays in this

case has been allegedly the adverse behavior of the four investigating officer which were changed during the investigation and allegedly protecting the criminal.

The Constitution of Pakistan guarantees the fundamental rights of all citizens, therefore, it is the duty of the courts to protect the fundamental rights. According to the well-known legal concept *sui juris*, the persons who have reached the age of majority can exercise right to the contract of marriage. In case of minor-baby discussed in our case study, she was declared a minor of age near 14 years, by a high powered Medical Board consisting of 12 senior members from medical profession. The was Board was constituted by competent court having jurisdiction. Therefore, her fake-paper-marriage is invalid and the central criminal Zaheer is legally debarred from taking the plea of marriage with this minor child. Hence the custody of the abductee needs to be handed over to her parents, the natural guardians by the honorable court. The family of gangsters kept the minor-baby under their illegal custody for more than three months and it is alleged that they were involved in rape. The gangster claimed the minor-baby to be his wife on the basis of “fake-paper-marriage”. The parents of the abductee Dua Zahra are requesting for acquiring the custody of their daughter.

7. Cancellation of Bail of Central Criminals

In order to meet the justice in Dua Zahra case, it is important to cancel the bails of central criminals and they be arrested and the case be decided on merits. The criminals be punished on the offences to be proven by prosecution for

offences under PPC sections 361, 362, 364, 364-A, 365, 365-B, 366-A 368 and 375 regarding kidnapping, abduction, forged marriage, illegal custody of minor, transporting and smuggling the minor for illicit reasons and rape. The minor baby be handed over to the parent without any further delay to save her academic year of school and ensure other matters related welfare of the child; to enable the parents to restore her in life as normal child in the society.

Further, the parents after attaining the custody of the baby must take adequately strict security measures to save her from the Gangster attacks, during her stay at home and school. The ordinary security in this case may be insufficient. The central criminal of the gang has already threatened the parents, it is alleged in court proceedings, to kidnap and abduct two little sisters of Dua Zahra aged 6 and 2 years.

8. Recommendations

- a. According to [12], the marriage of a minor without the approval of Wali is not be recognized by the courts.
- b. Enhanced punishments for abduction of minor females be prescribed.
- c. Most recent and modern technologies such as geo-fencing, according to [14], [15], and [16] must be used to localize and arrest the criminals by law enforcement agencies.
- d. During the pendency of such case as that of Dua Zahra, the honorable courts may restrict the gangsters to judicial custody till final verdict of the case is announced.
- e. The abducted baby has not been handed

over to the parents for the last six months. The faults of the legal system in Pakistan requires to be over viewed, revamped and overhauled.

9. Acknowledgement

The authors acknowledge the suggestion by Mr Kaukab Jamal Zuberi, Head Department of Criminology and Forensic Sciences.

10. References

- [1] Dr Aftab Ahmad Malik, Mujtaba Asad and Waqar Azeem (2020);” Importance of Prosecution Witnesses in Terrible Crimes of Sexual Violence, Abduction, Abuse, Torture, Rape and Killing against Innocent Women And Children “, Volume 4 , issue 4 PP: 03-14 , International Journal for Electronic Crimes Investigation”(IJECEI) ISSN 2522-3429.
- [2] Dr Aftab Ahmad Malik, Mujtaba Asad and Waqar Azeem (2020);”Promulgate Strong legal framework for child protection against offences of torturing, abusing or killing”; International Journal for Electronic Crimes Investigation”(IJECEI) Published in Volume 4 issue 2, April-June 2020 ; PP 1-10
- [3] Dr Aftab Ahmad Malik, “Standardization of forensic evidence its procurement preservation and presentation in court of using FBI techniques by FIA”; International Journal for Electronic Crimes Investigation (IJECEI); Published Volume 4 issue 1 , Jan - March 2020 PP :1-6.
- [4] Child Abduction. Collins English Dictionary. Copyright © HarperCollins Publishers.
- [5] Legal information Institute, Cornell Law School, “Fornication”, <https://www.law.cornell.edu/wex/fornication>
- [6] Dua Zehra timeline: A complex case of alleged kidnapping vs legal marriage <https://www.dawn.com/news/1696486>
- [7] The Prevention Of Trafficking In In Persons Act, 2018
- [8] Sindh Child Marriage Restraint Rule 2016
- [9] Child Marriage Restraint (Amendment) Bill, 2018
- [10] Dua Zehra case: (July 16, 2022).” Investigating Officer confirms Zaheer’s presence in Karachi on incident day; Pakistan Today, National Issue
- [11] U.S. Department of State — Bureau of Consular Affairs
- [12] Marriage Without the Approval of a Wali, <https://fiqh.islamonline.net/en/marriage-without-the-approval-of-a-wali>
- [13] “Dua Zahra brought back to Karachi” July 29, 2022, The News International
- [14] Amber Kemmis (2020): What Is Geofencing? Everything You Need to Know About Location-Based Marketing, Published in Smart Bug <https://www->

w . s m a r t b u g m e d i a . c o m / -
blog/what-is-geofencing SEP 4, 2020
7:00 AM

- [15] Sidney Fussel (2020), “Creepy ‘Geofence’ Finds Anyone Who Went Near a Crime Scene”, Published in Wired; <https://www.wired.com/story/creepy-geofence-finds-anyone-near-crime-scene/>
- [16] Geofencing the modern investigation system” 2019, Punjab Police, Government of Punjab <https://punjabpolice.gov.pk/node/6589>



Forensics Artifacts on Remote Desktop Protocol and Service.

Talha Ashfaq and Muhammad Shairoze Malik

School of Electrical engineering and computer Sciences, National University of Science and Technology, Islamabad.

Corresponding author: 13beemmalik@seeecs.edu.pk

Abstract:

Remote Desktop Protocol provides user a graphical user interface to access system remotely and its implementation is called “remote desktop services”. This is widely used by network administrators and remote workers. Due to vulnerabilities and weak configurations, the protocol is hugely abused by threat actors and hackers to perform malicious acts as data infiltration, deploying backdoors, malwares and lateral movements. In this article, there will be discussion on importance of RDP in digital forensics, understanding RDP based artifacts and there use in forensics investigation where RDP was suspected to be involved.

Key words: RDP, artifacts, Log analysis, bitmaps, Registry artifacts

1. Introduction

Remote desktop protocol (RDP) is propriety Protocol developed by Microsoft which enables users to remotely access computers over the internet. [1][2] Operations sent to the remote server are executed as it was performed by the user (local) itself. The main focus of the protocol is the complete representation of the screen content of the remote-controlled computer [4]. This is widely used tool in areas where remote work, assistance and administration are required. The user use RDP client for remote connection to

the RDP server. RDP Client exists on most of operating system such as Microsoft Windows, UNIX, macOS, IOS and android, while RDP servers are built on Microsoft Windows. [3] In order to establish RDP connection local and remote machines need to authenticate via username and password. Microsoft terms implementation of Remote Desktop Protocol (RDP) as “Remote Desktop Service”. [1] [3]

Importance of RDP in Forensics Investigation

As explained, RDP is very widely used proto-

col for employees and Administrators to provide graphical access to remotely connected devices. Despite providing useful features, RDP is abused very often by threat actors. Threat actors commonly target RDP as a primary method to gain access in an organization's network. Once initial foothold is achieved, threat actors or hacker(s) can deploy malwares, ransomwares, can perform data exfiltration or lateral movement without being detected. The Federal Bureau of Investigation (FBI) even released a warning in 2018 addressing dark markets selling RDP access. Threat actors can easily Weak password policies and misconfigured endpoint security play a big role in this. RDP is also prone to vulnerabilities and been exploited for reconnaissance, command and control and lateral movements. [11][10]. RDP was initial attack vector for 50% of the ransomware attack reported by unit 42. [12]. Reported by ESET telemetry, "In the first quarter of 2020, we saw 1.97 billion connection attempts. By the fourth quarter of 2021, that had jumped to 166.37 billion connection attempts, an increase of over 8,400%!" [18]

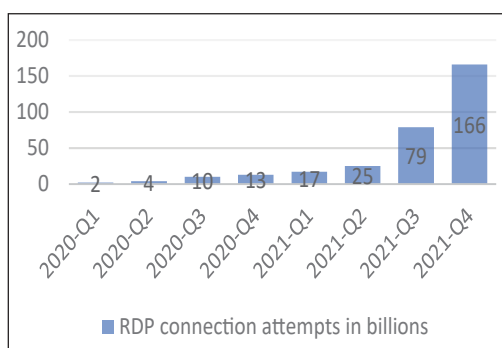


Figure 1 Malicious RDP connection attempts detected worldwide (source: ESET telemetry). Absolute numbers are rounded

Many sophisticated ransomwares such as Matrix, SamSam uses RDP vulnerabilities to gain initial foothold reported by Sophos [13]. In darknet, there are various marketplace that offers complete RDP data for low prices. [4] Thus it is important for a forensics analyst for analyse RDP connection and related artifacts in digital investigations.

RDP artifacts

RDP artifacts can have probative value when investigating a cyber-attack or incident, they include which user made connection to remote server on what time, what activity a user performed from the remotely established connection. These artifacts can be categorized in artifacts related to connectivity of remote server and artifacts for activity performed by the user using remote server.

2. RDP Log analysis

To establish a remote desktop connection, RDP uses user's credential to authenticate login in the remote server. Windows event logs is an audit feature by Microsoft to record user events and activities on a system, also are potential source of evidence for forensics investigations [20]. Event logs resides usually at location C:\Windows\System32\winevt\Logs. RDP connection usually follows a uniform flow chart and they also leave traces in term of events in Windows Event logs. Here, some forensically relevant field of Windows event logs pointing RDP connection, Login, Session disconnection and Logoff, these events are chronologically ordered as [4]:

Event ID	Network Connection
1149	This Event log does not actually indicates remote controlled system but recorded only when network connection between the client and server is successfully connected. This would initiates login prompts for authentication. Event ID 1149 will only initiate when Network level Authentication is enabled.
	Authentication
4624	This event log occurs when a user is successfully authenticated on the server. If Logon Type 10, 7 is observed means the user is reconnected to the server. Type 3 would be observed when NLA (network level authentication) is enabled.
4625	Type 3 if NLA is enabled and/or type 10
	Login
21	If source network address is "LOCAL", this is not an indication of an RDP login. "LOCAL" is also generated at PC start/reboot.
22	This event follows immediately after event 21. This event indicates a successful RDP login and starting a shell. Unless the source network is "LOCAL", this login shows RDP login.
	Session Disconnect/reconnect
24	This event shows the user disconnected from an RDP session if source network address is not "LOCAL". This event usually appears with event ID 40.
25	This event shows the user has reconnected to an RDP session if source network is not "LOCAL". This also appears with event ID 40.
39	Connection formally ended, not just closed the window
40	Connection was terminated for reason code X.

4778	This event occurs when user connects to ongoing RDP session. Session name, client address and login ID can be used to identify the source will. This event often connects with event id 25.
4779	Occurs when a user disconnects from an RDP session. Often in connection with event ID 24 and with 39 and 40.
	Logoff
23	user initiated a system logoff, usually with connection with event ID 4634
4634	This event occurs when a user terminates an RDP connection or log out from the remote system.
9009	"Desktop Windows Manager" has ended occurs when an RDP connection is lost, which terminates the RDP desktop interface.

2. RDP bitmaps

According to Microsoft "Bitmap caches are used by the client and server to store graphic bitmaps. Each bitmap cache holds bitmaps of a specified size in pixels (known as the "tile size"). If a bitmap does not fit into a single cache entry, the server uses a tiling algorithm to divide the bitmap into tiles that will fit into the cache entries so that they can be stored separately into the cache" [14].

When a user connects to other system using RDP, small bitmaps images are created in the Client's (user's) system. If same image is used in the session it can be quickly retrieve and preventing same images to load once again. This caching can provide efficiency in performance even over low-bandwidth connections [7]. While analysing RDP, this can help investigators what the user was seeing in their RDP

For manually stitching the RDP bitmaps “RDPCacheStitcher” is a helpful tool to ease the tedious process. This supports output from ANSSI-FR/bmc-tools as input, provides a GUI and several placement heuristics for stitching the tiles together [17] Also some open source scripts exists where you can attempt to automatically parse extracted RDP bitmaps, such a “RDPPieces”. [16].

3. Registry artifacts

Remote desktop protocol does leave artifacts in system registry under HKCU\Software\Microsoft\Terminal Server Client\Default. Under this key, history of clients private IP addresses are listed. RDP also leaves username used for the connection and also with the client PC names under the registry key HKCU\Software\Microsoft\Terminal Server Client\Servers. [6]

Client.Default

Name	Type	Data
(Default)	REG_SZ	(value not set)
MRU0	REG_SZ	10.10.10.10
MRU1	REG_SZ	10.10.10.10
MRU2	REG_SZ	10.10.10.10
MRU3	REG_SZ	10.10.10.10
MRU4	REG_SZ	10.10.10.10
MRU5	REG_SZ	10.10.10.10

Figure 5 list of IP servers connected with client in registry

Client.Servers

Name	Type	Data
(Default)	REG_SZ	(value not set)
UsernameHint	REG_SZ	MicrosoftAccount\JEUSER

Figure 6 Username hint with IP address Server's in Windows Registry

4. Correlating RDP artifacts

RDP artifacts can help the forensics examiner to reconstruct events happened on the system. As explained, remote connection times, RDP client's IP, logon patterns and what a user was visualised on the remote desktop client can be retrieve from windows event logs, windows registry and remote desktop bitmap cache resident on user's profile. To prove a suspected logon initiated from system being analysed, Client IP, and username can be seen from Windows Registry. For a successful RDP logon, Event ID 1149 from remoteConnectionManager.evtx is recorded, followed by ID 4624 (Security.evtx), 21 (LocalSessionManager.evtx) and event id 22 (LocalSessionManager.evtx).



Figure 7 Event logs on Successful RDP Logon

Similarly, for RDP session logoff, find event log 23, 4634, 4647 and 9009 respectively.

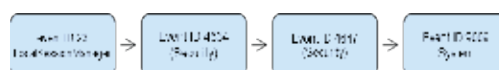


Figure 8 Event logs on RDP Session Logoff

RDP Session disconnect will generate following event ID as Event ID 24, 39, 40, 4779 and event ID 4634.



Figure 9 Event Log on RDP Session Disconnect

Logon times are important to be view here as they can be used to find specific RDP bitmap

caches from client's system. RDP cache files (bcache.bmc and cache###.bin) created for each session at the time of RDP connection logon. After finding related bitmap cache then you can parse out bitmaps from the cache file to view activity done using RDP client. Bitmaps extracted from cache are usually in large numbers and analysing them is tedious process, as explained you can limit the search by correlating logon time from event logs with RDP cache creation times. Remote Desktop service also does leaves artifacts in windows registry where you can confirm IP and computer name of RDP server the client connecting to.

5. Summary

Remote Desktop Protocol is propriety of Microsoft Windows to provide GUI access to remote device. RDP follows client server model, where multiple clients exists on different operating systems, while the device to be remotely connected (remote desktop server) is usually on Windows environment. Threat actors uses RDP as their favourite initial attack vector in there malicious activities, various vulnerabilities also exists for remote desktop protocol. This makes RDP an important aspect to investigate in forensics investigations. Artifacts related to RDP can be retrieve from Windows Event logs, Windows Registry and RDP bitmap cache files to prove or disprove malicious activity attempted with the remote protocol. Differing usual Windows events from suspicious RDP activity can be tedious and difficult to understand. In this article system locations, Event Log Fields, tools and methods are discussed to understand RDP related activities.

6. References

1. Understanding Remote Desktop Protocol (RDP) - Windows Server. (2021a, September 24). Microsoft Learn. <https://learn.microsoft.com/en-us/troubleshoot/windows-server/remote/understanding-remote-desktop-protocol>
2. Paubox. (2022, September 14). What is a remote desktop protocol attack? <https://www.paubox.com/resources/what-is-remote-desktop-protocol-attack/>
3. Remote Desktop Services (Remote Desktop Services) - Win32 apps. (2020, December 10). Microsoft Learn. <https://learn.microsoft.com/en-us/windows/win32/termserv/terminal-services-portal?source=recommendations>
4. Lauckner, K.N., 2020. Forensischer Nachweis eines Lateral Movement nach unberechtigtem RDP Zugriff (Doctoral dissertation).
5. Duranec, A. & Gruicic, Savina & Zagar, Marinko. (2020). Forensic analysis of Windows 10 Sandbox. 1224-1229. 10.23919/MIPRO48935.2020.924522
6. Kerai, P., 2010. Remote access forensics for vnc and rdp on windows platform.
7. ANSSI-FR/bmc-tools: RDP Bitmap Cache parser (github.com)
8. Chan, J. (2020, March 13). Do You Even Bitmap Cache, Bro? All Things DFIR. <https://www.allthingsdfir.com/->

- do-you-even-bitmap-cache-bro/
9. Swoboda, Andrew, Lane Thames, and Tyler Reguly. "RDP Fuzzing."
 10. Ingalls, S. (2022, March 26). Addressing Remote Desktop Attacks and Security. eSecurityPlanet. <https://www.esecurity-planet.com/threats/rdp-attacks/>
 11. Remote Desktop Protocol Use in Ransomware Attacks. (2022, May 6). RH-ISAC. <https://rhisac.org/ransomware/remote-desktop-protocol-use-in-ransomware-attacks/>
 12. 2020 Unit 42 Incident Response and Data Breach Report. (n.d.). Palo Alto Networks. <https://www.paloaltonetworks.com/resources/research/2020-unit42-incident-response-and-data-breach-report>
 13. Mark Loman, SOPHOS, How Ransomware attacks
 14. [MS-RDPEGDI]: Bitmap Caches. (2021, June 24). Microsoft Learn. https://learn.microsoft.com/en-us/open-specs/windows_protocols/ms-rdpegdi/2bf92588-42bd-4527-8b3e-b90c56e292d2
 15. ANSSI-FR/bmc-tools: RDP Bitmap Cache parser (github.com)
 16. brimorlabs/rdpieces: The home of the BriMor Labs rdpieces Perl script that tries to rebuild parsed RDP Bitmap Cache images (github.com)
 17. BSI-Bund/RdpCacheStitcher: RdpCacheStitcher is a tool that supports forensic analysts in reconstructing useful images out of RDP cache bitmaps. (github.com)
 18. Goretsky, A., & Goretsky, A. (2022b, September 13). RDP on the radar: An up-close view of evolving remote access threats. WeLiveSecurity. <https://www.welivesecurity.com/2022/09/07/rdp-radar-up-close-view-evolving-remote-access-threats/>
 19. Schatz, Bradley. (2007). Digital evidence: representation and assurance.
 20. Do, Quang, et al. "Windows event forensic process." IFIP International Conference on Digital Forensics. Springer, Berlin, Heidelberg, 2014.



Forensic toxicology: importance in crime investigation

Asif Ibrahim¹ and Syed Khurram Hassan²

¹ Department of Mathematics, FC College University, Lahore.

² Institute of Quality and Technology Management, University of the Punjab, Lahore, Pakistan.

Corresponding author: khuramshah6515@gmail.com

Abstract

By using chemical and analytical techniques, forensic toxicology contributes to the establishment of facts in forensic investigations by studying the medical and legal aspects of the detrimental effects of a drug on organisms. Forensic practises is crucial in cases of deadly poisoning and those that could be connected to criminal activity. In most murder, suicide, or accident cases, poisons are discovered. They are a crucial component of the silent weapon that silently and covertly ends life. The current study is focused on the field of forensic chemistry and toxicology, which is entirely centered on the introduction, categorization, effects, and influencing factors of poisons, as well as their detection and testing. This article's goal is to examine how they behave and perform after they enter the human body. Poisons have serious effects and might even be fatal if not treated appropriately.

Key words: Forensic, Toxicology, Poisoning, Neurotoxins, Irritants

1. Introduction

The word "forensic" comes from the Latin word "forensis," which means "public," "to the prospect and discussion"[1]. Legal concerns can be resolved in a variety of ways thanks to rhetorical science. Rhetorical science, which includes a variety of disciplines such as rhetorical chemistry, rhetorical social science, rhetorical biology, rhetorical medicine, rhetorical engineering, rhetorical material sciences, machine rhetorical, and others, is generally used to settle legal disputes, fairly enforce criminal and civil laws, and protect the

public. A broad phrase that covers most of the duties performed by the law laboratory is forensic chemistry[2]. Trace analysis and medicine are techniques employed in the field of rhetorical chemistry. Analytical chemistry that is applied is forensic chemistry. Rhetorical chemistry adds comparison study to the assignment whereas analytical chemistry covers both quantitative and qualitative chemical analysis [3]. Similarly, spectrometry can quickly determine whether a sample is made of nylon or polythene. Therefore, analytical chemistry offers the quantitative and qualitative data needed to respond to rhetorical questions [4].

Forensic Analysis

The use of scientific knowledge that is based on legal issues is known as forensic science [5]. In order to determine what happened, when it happened, and who was responsible, forensic science primarily examines biological and physical evidence. To attain accuracy and precision, forensic scientific proficiency is crucial [6].

2. Forensic Toxicology

The study of poisons' and drugs' detrimental effects on living things is known as toxicology [7]. It include the investigation of the signs, causes, effects, and methods of handling certain toxins and medications. If the use of medications and poisons results in death in dubious circumstances, it becomes rhetorical pharmacological medicine [8].

The field of medicine may include pharmacological medicine as a sub-field. The study of medicine includes all interactions between drugs and other chemicals on living things [9]. Drug administration, body absorption, activities and interactions, metabolism, and excretion are all aspects of medicine [10].

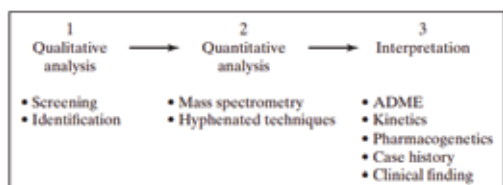


Fig: Stages of forensic toxicology

3. History of Forensic Chemistry

Rhetorical chemistry advancements started to become apparent by the middle of the 19th century. Blood tests were developed at this

time, the Marsh test for arsenic was created in 1832, and experiments on bullet "fingerprinting" were conducted in the 1980s. Christian Friedrich Schönbein (1799–1868), a German–Swiss scientist, developed the first accurate method for differentiating human blood in 1863 [11]. For more than a century and a half, arsenic has been widely used as a toxin. Its excellence goes back to the seventh century. During the 16th century, the Arab philosopher Abu Musa Jabir Ibn Hayyan (about 721–ca. 815), also known as Geber, found the process for transforming the grey, metallic-looking elemental arsenic, an elemental chemical compound (As_2O_3 ; a material) into (White, flavourless, and odourless powder). Arsenic may just be an extra chemical in person's food or beverage without raising suspicion [12]. Over the following sixty years, attempts to employ bullet "fingerprinting" in criminal investigations were few [13]. But by the 1890s, a number of events spurred renewed interest in the method as a means of identifying possible offenders. Of those aspects, the development of the replacement method known as "grooving" for making gun barrels was perhaps the most crucial. The process of grooving involves carving spiral grooves into the inner surface of tubing. When a bullet travels through the barrel, the grooves allow it to spin; this motion prevents the bullet from tipping over after it exits the tube. Different grooving techniques have been employed by numerous gun makers [14].

Poisoning

It is referred to as the harmful consequence of a poison or harmful chemical agent. It causes the emergence of negative reactions to dangerous substances or chemicals [15]. Basically, it

may be divided into three groups: homicidal, suicidal, and accidental [16].

Homicidal Poisoning

The victim is frequently subjected to attempts to "nurse" them back to health by poisoners [17]. Serial poisoners typically enjoy the rush of having control over the victim's life and suffering, and poisoners frequently take delight in seeing their victims suffer. Homicide by poisoning perpetrators frequently work in the healthcare or medical industries. The substances that are most appealing to offenders are those that are deadly in little doses. The ideal poison for a homicide has no taste, is undetectable, has no odour, and exhibits symptoms that are comparable to those of illnesses that are found in nature [18]. Since current scientific techniques and advancements have made poison detection simpler, it has becoming more and more challenging to find a poison having all of these characteristics [19].

Suicidal Poisoning:

Self-poisoning, the non-violent way of suicide, most frequently involves the use of medicine, either over-the-counter (paracetamol) or prescribed (such as antidepressants and prescription analgesics), chemicals (pesticides), or illegal narcotics [20].

Accidental poisoning

Accidental poisoning, which includes accidental drug overdose, occurs when a person inadvertently poisons themselves. Alcohol, opioids (such as heroin or methadone), sedatives, psychiatric pharmaceuticals (such as antidepressants), antiepileptic, and anti-inflammatory medications are some of the substances from which poisoning may result [21].

Classification of poisons

Poisons are divided into two categories based on how they affect the body and depending on their chemical and physical characteristics.

Classification based upon the effect of poison on the body:

A) Corrosive: When poisons come into contact with tissues or organs, they become corrosive, for example: a. Strong acids like H_2SO_4 , HCl , HNO_3 , etc. and strong alkalis include NH_4 , Na/K hydroxides, etc [22].

B) Neurotoxins: Toxins known as neurotoxins cause damage to nerve tissue. Exogenous chemicals known as neurotoxins are a broad category of neurological insults that can negatively impact the function of both growing and mature brain tissue. Lead, glutamate, ethanol (drinking alcohol), botulinum toxin (e.g., Botox), tetanus toxin, nitric oxide and tetrodotoxin are typical examples of neurotoxins [23].

C) Irritants poisons: They mostly cause inflammation at the point of contact, particularly in the skin, gastrointestinal system, and respiratory tract [24]. A poison is categorized as one that affects a system the most when it causes death as a result of a systemic impact, such as a heart poison, brain poison, or spinal poison. The inorganic toxin arsenic is a hefty metallic irritation. Due to its insoluble nature in water and inability to be absorbed by the digestive system, metallic arsenic is not harmful. Arsenic trioxide, often known as sankhyal or somalkar, is toxic. Arsenobetaine and arsenocholine are two organic arsenic non-toxic forms that are typically present in food that humans frequently ingest [25]. Cod, haddock, and shellfish contain them.

Poisons are also Categorized According to their Characteristics:

A) Inorganic Poisons

i) Metallic Poisoning

After exposure, microscopic metal molecules build up in your body and cause heavy metal

poisoning. Without treatment, heavy metals can produce symptoms that are potentially fatal because they adhere to your cells and stop them from functioning. Your body can become poisonous to a variety of metals [26]. The most prevalent poisonous metals are:

Types of Metals	Where they can be found
Cadmium	metal plating, Cigarette smoke ,batteries.
Arensic	Topical creams, polluted water, shellfish, algae, pesticides, fungicides, insecticides, and fungicide-based paints, enamels, and glass.
Lead	lead pipes, batteries, paint, gasoline, and building materials have all tainted the water.
Mercury	Batteries, shellfish, topical antiseptics, dental amalgam ("silver") fillings, thermometers, and lightbulbs
Thallium	Rodenticides, fireworks , pesticides,.

ii) Non Metallic Poisons

A) Phosphine and phosphides

A variety of industrial operations employ phosphonate, a highly poisonous colourless gas with a pungent garlic or fishy odour. It is also produced when phosphides are exposed to moisture.[27] In underdeveloped nations, aluminium phosphide is widely utilized as a cheap and efficient grain fumigant and rodenticide [28].

B) Bromide

The colourless gas methyl bromide has historically been employed as a refrigerant and in fire extinguishers, but it is most frequently utilized as an insecticidal fumigant for grain storage

and soil. Methyl bromide is a metabolite that produces the bromide ion and is linked to unintentional poisoning, especially in work environments [29]. By passive diffusion, the bromide ion is quickly absorbed from the stomach and proximal small intestines. Bromide ions are mostly found in extracellular fluid, where they have similar properties to chloride ions. The kidney is the most significant organ for elimination. Their half-life of elimination is relatively lengthy, lasting around 10 days after an acute dose or many weeks after stopping a long-term consumption, especially in situations of bromide intoxication.

C) Cyanides

The majority of cases of severe or deadly cyanide poisoning involve the suicide intake of cyanide salts. A very poisonous volatile liquid, hydrogen cyanide. When cyanide salts react with acids or are formed in the stomach after oral consumption, hydrogen cyanide fumes are released [30]. Although HCN has a distinctive almond-like odour, up to 50% of people cannot detect it. Even while this might have been connected to air flow ventilation systems in post-mortem rooms, it was surprising that it was not a distinguishing feature during autopsy of a significant number of cyanide suicide fatalities. The industrial applications of potassium and sodium cyanide as soluble salts of cyanide include electroplating, metal processing, and laboratory reagents [31].

B. Organic poisons

a) Ethanol: A much of ethanol is toxic, so avoid using it.

b. Other alcohols: Poisonous alcohols include methyl and isopropyl.

Methanol, a substance used in the chemical and polish industries as well as clandestine alcoholic beverages, may be fatal when consumed.

c. Phenol: Carboxylic acid or phenol may be toxic. The main purpose of it is as a disinfectant .

d. Other substances: Poisonous industrial chemicals include benzene, chloral hydrate, chlorinated hydrocarbons, and others. Chloral hydrate might be present in illegal alcoholic

beverages in a number of poisoning instances [32].

Route & Site of Exposure

When administered intravenously, toxic substances often have the most impact and the fastest reaction. For alternative routes, inhalation, intraperitoneal, subcutaneous, intramuscular, intradermal, oral, and topical would roughly be listed in decreasing order of efficacy [33]. Additionally, the method of delivery might affect an agent's toxicity. For instance, it would be reasonable to anticipate that a substance that is detoxified in the liver would be less hazardous when administered orally through the portal circulation than systemically (inhalation) [34].

Duration & Frequency of Exposure

The toxic effects brought on by a single exposure to multiple substances differ significantly from those brought on by repeated exposure. For instance, benzene's major acute toxic symptom is central nervous system depression, but prolonged exposure can cause leukaemia. Acute exposure to quickly absorbed substances is likely to result in immediate toxicity, but it is also possible for acute exposure to result in delayed toxicity that may or may not be comparable to the toxic consequences of chronic exposure. In contrast, repeated administration of a hazardous agent may result in certain short-term (acute) side effects in addition to the agent's long-term, low-level, or chronic effects [35].

Local versus Systemic Toxicity

The central nervous system is the target organ of toxicity that is most commonly engaged in systemic toxicity. Damage to the central

nervous system, particularly the brain, can be proved by the use of proper and sensitive procedures, even if many substances have obvious effects elsewhere. The circulatory system, the blood and hematopoietic system, visceral organs such the liver, kidney, and lung, and the skin follow in order of frequency of involvement in systemic toxicity. The least often targeted tissues for systemic effects include muscle and bone. The frequency of tissue reactions when drugs have a localized effect primarily depends on the portal of entrance (skin, gastrointestinal tract, respiratory tract) [36].

Reversible versus Irreversible Toxic Effects

Chemical toxicity can have both reversible and irreversible consequences. The tissue's capacity to regenerate will play a significant role in determining whether a chemical insult to a tissue results in reversible or irreversible damage. Therefore, whereas most injuries to the central nervous system are mostly irreversible due to the differentiated cells of the central nervous system being unable to divide and be replaced, most injuries to a tissue like the liver, which has a high capacity for regeneration, are reversible. Chemicals can cause cancer and permanent harmful consequences [37].

4. Conclusion

By analyzing the medical and legal elements of a drug's harmful effects on organisms, forensic toxicology uses chemical and analytical tools to help establish the facts in forensic investigations. In situations of fatal poisoning and those that may be related to criminal conduct, forensic practise is essential. Poisons are typically found in murder, suicide, or accident

situations. They are an essential part of the silent weapon that kills people stealthily and invisibly. The subject of the current research is forensic chemistry and toxicology, which is fully concerned with the introduction, classification, effects, and influencing factors of poisons, as well as their detection and testing. The purpose of this article is to investigate how they act and function after they enter the human body.

5. References

1. Ibrahim, M. S., & Abdullah, M. (2010). Forensic Accounting in Malaysia: Some Insights from Practitioners. *Asian Journal of Accounting Perspectives*, 3(1), 14-21.
2. Kloosterman, A., Mapes, A., Geradts, Z., van Eijk, E., Koper, C., van den Berg, J., ... & van Asten, A. (2015). The interface between forensic science and technology: how technology could cause a paradigm shift in the role of forensic institutes in the criminal justice system. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 370(1674), 20140264.
3. Yaman, F. (2021). Examining students' quality and perceptions of argumentative and summary writing within a knowledge generation approach to learning in an analytical chemistry course. *Chemistry Education Research and Practice*, 22(4), 985-1002.
4. Hibbert, D. B. (2007). *Quality assurance in the analytical chemistry laboratory*. Oxford University Press.

5. Robertson, B., Vignaux, G. A., & Berger, C. E. (2016). Interpreting evidence: evaluating forensic science in the courtroom. John Wiley & Sons.
6. Koehler, J. J. (2017). Forensics or fauxrensic? Ascertaining accuracy in the forensic sciences. *Ariz. St. LJ*, 49, 1369.
7. Klaassen, C. D., Hardman, J. G., Limbird, L. E., Molinoff, P. B., Ruddon, R. W., & Gilman, A. G. (2006). Principles of toxicology and treatment of poisoning. Goodman and Gilman's The Pharmacological Basis of Therapeutics, Eleventh Edition., McGraw Hill, Columbus, OH, USA, 1739-1752.
8. Shorter, E. (2008). Before Prozac: The troubled history of mood disorders in psychiatry. Oxford University Press.
9. Yuan, H., Ma, Q., Ye, L., & Piao, G. (2016). The traditional medicine and modern medicine from natural products. *Molecules*, 21(5), 559.
10. Li, L., Liu, T., Fu, C., Tan, L., Meng, X., & Liu, H. (2015). Biodistribution, excretion, and toxicity of mesoporous silica nanoparticles after oral administration depend on their shape. *Nanomedicine: Nanotechnology, Biology and Medicine*, 11(8), 1915-1924.
11. Newton, D. E. (2007). Forensic chemistry. Infobase Publishing.
12. Sharma, A. K., Tjell, J. C., Sloth, J. J., & Holm, P. E. (2014). Review of arsenic contamination, exposure through water and food and low cost mitigation options for rural areas. *Applied Geochemistry*, 41, 11-33.
13. Spitz, W. U., & Diaz, F. J. (2020). Spitz and Fisher's medicolegal investigation of death: guidelines for the application of pathology to crime investigation. Charles C Thomas Publisher.
14. Pétillon, J. M., & Ducasse, S. (2012). From flakes to grooves: A technical shift in antlerworking during the last glacial maximum in southwest France. *Journal of Human Evolution*, 62(4), 435-465.
15. Sharma, M., Jabin, S., & Sharma, M. PHARMACEUTICAL POLLUTION: A GRAVE CONCERN!!.
16. Sikary, A. K. (2019). Homicidal poisoning in India: A short review. *Journal of forensic and legal medicine*, 61, 13-16.
17. Van Landeghem, A. A., De Letter, E. A., Lambert, W. E., Van Peteghem, C. H., & Piette, M. H. (2007). Aconitine involvement in an unusual homicide case. *International journal of legal medicine*, 121(3), 214-219.
18. Gunn, A. (2019). Essential forensic biology. John Wiley & Sons.
19. Trestrail, J. H. (2007). Types of Poisons. Criminal Poisoning: Investigational Guide for Law Enforcement, Toxicologists, Forensic Scientists, and Attorneys, 29-46.
20. Bonvoisin, T., Utyasheva, L., Knipe, D., Gunnell, D., & Eddleston, M. (2020).

- Suicide by pesticide poisoning in India: a review of pesticide regulations and their impact on suicide trends. *BMC public health*, 20(1), 1-16.
21. Manzar, N., Saad, S. M. A., Manzar, B., & Fatima, S. S. (2010). The study of etiological and demographic characteristics of acute household accidental poisoning in children-a consecutive case series study from Pakistan. *BMC pediatrics*, 10(1), 1-6.
 22. Chibishev, A., Pereska, Z., Chibisheva, V., & Simonovska, N. (2012). Corrosive poisonings in adults. *Materia socio-medica*.
 23. Hu, Y., Chen, J., Fan, H., Xie, P., & He, J. (2016). A review of neurotoxicity of microcystins. *Environmental science and pollution research*, 23(8), 7211-7219.
 24. Wasserman, S., & Watson, W. (2011). Food allergy. *Allergy, Asthma & Clinical Immunology*, 7(1), 1-7.
 25. Borak, J., & Hosgood, H. D. (2007). Seafood arsenic: implications for human risk assessment. *Regulatory Toxicology and Pharmacology*, 47(2), 204-212.
 26. Pandey, G., & Madhuri, S. (2014). Heavy metals causing toxicity in animals and fishes. *Research Journal of Animal, Veterinary and Fishery Sciences*, 2(2), 17-23.
 27. Stejskal, V., Vendl, T., Aulicky, R., & Athanassiou, C. (2021). Synthetic and natural insecticides: Gas, liquid, gel and solid formulations for stored-product and food-industry pest control. *Insects*, 12(7), 590.
 28. Francois, M. R., & Stephen, F. (2015). Phosphorus Compounds. In *Hamilton & Hardy's Industrial Toxicology* (pp. 383-390). Hoboken, New Jersey: John Wiley & Sons, Inc..
 29. Kaushik, R. D. (2021). Methyl bromide: Risk assessment, environmental, and health hazard. In *Hazardous Gases* (pp. 239-250). Academic Press.
 30. Thompson, J. P., & Marrs, T. C. (2012). Hydroxocobalamin in cyanide poisoning. *Clinical Toxicology*, 50(10), 875-885.
 31. Dash, R. R., Gaur, A., & Balomajumder, C. (2009). Cyanide in industrial wastewaters and its removal: a review on biotreatment. *Journal of hazardous materials*, 163(1), 1-11.
 32. Mishra, A. (2020). Forensic Chemistry and Toxicology. In *Medical Toxicology*. IntechOpen.
 33. Donovan, M. D. (2009). Effect of route of administration and distribution on drug action. In *Modern Pharmaceutics Volume 1* (pp. 173-198). CRC Press.
 34. Boey, A., & Ho, H. K. (2020). All roads lead to the liver: metal nanoparticles and their implications for liver health. *Small*, 16(21), 2000153.
 35. Jett, D. A., Sibrizzi, C. A., Blain, R. B.,

- Hartman, P. A., Lein, P. J., Taylor, K. W., & Rooney, A. A. (2020). A national toxicology program systematic review of the evidence for long-term effects after acute exposure to sarin nerve agent. *Critical reviews in toxicology*, 50(6), 474-490.
36. El-Boghdadly, K., Pawa, A., & Chin, K. J. (2018). Local anesthetic systemic toxicity: current perspectives. *Local and regional anesthesia*, 11, 35.
37. Grandin, E. W., Ky, B., Cornell, R. F., Carver, J., & Lenihan, D. J. (2015). Patterns of cardiac toxicity associated with irreversible proteasome inhibition in the treatment of multiple myeloma. *Journal of cardiac failure*, 21(2), 138-144.



Study of the Anti-Debugging Techniques and their Mitigations

Muhammad Saad and Muhammad Taseer

School of Electrical Engineering and Computer Sciences, NUST, Islamabad, Pakistan

Corresponding author: 12msccsmsuleman@seecs.edu.pk

Abstract:

The major goal of this study is to provide anti-debugging and anti-reversing strategies/techniques employed by executables, DLLs, and packers/protectors, as well as to examine strategies that can be utilized to bypass or disable these protections. Anti-debugging techniques are designed to make sure that a program is not being executed inside a debugger. In most cases, the anti-debugging process slows down the reverse engineering [1] process but doesn't stop it. This information will allow malware analysts and researchers to identify the techniques used by the malware. This information may also be used by security researchers, reverse engineers those want to slow down the process of reverse engineering in order to add security [2] to their software. It causes some difficulties for a reverse engineer, but, of course, nothing stops a skilled, knowledgeable, and committed reverse engineer.

Keywords: malware analysis, anti-debugging, anti-reversing, protectors, packers

1. Introduction

Prior to then, malware Development served as a showcase for malware coders. Malware analysts have used debuggers to run a malware program's instructions one by one, introducing modifications to memory spaces, settings as well as variable values. Debuggers are the most commonly used reverse engineering tools, such as Interactive Disassembler (IDA), x64dbg, and OllyDBG. If debugging is successful, it helps to understand malware behavior and its capabilities. This is something

malware developers would like to avoid. That is why they must implement anti-debugging techniques. Anti-debugging techniques[3] can be used to merely detect the presence of a debugger, deactivate it, lose control of it, or even take advantage of a flaw in the debugger. Disabling or avoiding debugger checks can be done generally and specifically. However, you can exploit this vulnerability against specific debuggers. Furthermore, The Supervisory Control and Data Acquisition (SCADA)[4] system has a vulnerability, according to the Trend Micro report "Unseen Threats, Imminent losses," which is the part of industrial

control systems (ICS)[5]. In addition, In many situations, knowing how to apply anti-debugging techniques to malicious code to prevent it from being tracked down and evaluated is also helpful. One of the main tools used by malware analysts and reverse engineers is the debugger. What is a debugger? A debugger is software that is used to evaluate and control the flow of execution of other executables or software. By using a debugger, we can execute each instruction step by step and can note down the changes that can be displayed on the stack, memory dumps, registers, etc. Most packers use these techniques to determine whether the system is running a debugger or if a process is being debugged. These debugger detection methods[6] include checks that are relatively basic all the way up to ones that are applicable to native Application Programming Interfaces (APIs) and kernel objects[7]. This section discusses how anti-debugging techniques work. Each process's user space contains a data structure called a Process Environment Block (PEB), which holds information about the related process. Each process's user space contains a data structure called a Process Environment Block (PEB), which holds information about the related process. It is intended to access Windows API (WinAPI) but access is not restricted by this. Process Environment Block (PEB) can be accessed directly from memory. Checking the value of the Process Environment Block (PEB) structure that has been debugged is a relatively straightforward implementation and technique. As we know that there are so many Applications Programmable Interfaces (APIs) which are documented and undocumented. For example, IsDebuggerPresent, which we will discuss later in this paper. To enhance, we can also check the APIs

manually. The fs segment register can access the Process Environment Block (PEB) at fs: [30]. On an x86 [8] computer, this register corresponds to a Thread Information Block (TIB). There is also a flag below the Process Environment Block (PEB) that indicates whether the first memory space of the process was created in debug mode. Provide an offset of 0x18 in the Process Environment Block (PEB). So, here I break down the anti-debugging techniques into two categories: static anti-debugging and dynamic anti-debugging [9], as seen in the Table 1 below.

Table 1. Static Vs Dynamic Techniques Difference

	Static	Dynamic
Difficulty Level	Easy, Medium	Hard
Main Idea	Use System Info	Reverse and exploit Debugger
Target	Detect Debugger	Hide its own code and data
Time Point	When debugging start	While debugger is running
Defend Method(s)	API Hook, debugger plugin	API hook, Debugger Plugin
Example(s)	PEB, TEB, TLS	Breakpoints (INT3), TimingCheck

In our research we will discuss we will discuss some of the main anti-debugging techniques and how a reverse engineer can be able to identify them easily for example in this paper we will discuss about the IsDebuggerPresent, TimeChecks, NtQueryInformationProcess, NtSetInformationThread, SwitchDesktops, SeDebugPrivilege, ParentProcess, DebuggerWindow, DeviceDrivers etc.

Anti-Debugging Techniques Mechanism:

Anti-debugging[10] is the implementation of

one or more techniques in computer code that make it difficult to reverse engineer or debug the target process. These techniques are ways for a program to detect whether it is running under the control of a debugger[11]. If a debugger is detected, the malware will execute arbitrary code, usually code to terminate. The anti-debugging process slows down the reverse engineering process but doesn't stop it.

2. Is Debugger Present:

The easiest debugger detection technique is to check the BeingDebugged flag in the Process Environment Block (PEB). The kernel32IsDebuggerPresent() function was introduced in Windows 95, and the Application Programmable Interface (API) checks the value of this flag to identify the process whether it is in the user-mode debugger. This code (same 32-bit or 64-bit Windows environment) can be used for verification to check the 32-bit or 64-bit Windows environment. As we can see the assembly code of the IsDebuggerPresent() in Figure 1.

```

i call kernel32!IsDebuggerPresent()
call IsDebuggerPresent
test al, al
jnz .debugger_found

i check PEB.BeingDebugged directly
mov eax, dword [eax+30] ; EAX = PEB.ProcessEnvironmentBlock
movzx eax, byte [eax+0x02] ; AL = PEB.BeingDebugged
test eax, eax
jnz .debugger_found
  
```

Figure 1. Assembly code of IsDebuggerPresent()

C/C++ Code:

As we can see in the example if IsDebuggerPresent() in Figure 2.

```

if (IsDebuggerPresent())
    ExitProcess(-1);
  
```

Figure 2. C/C++ code of IsDebuggerPresent()

Solution:

This technique can be easily bypassed by manually patching the Process Environment Block (PEB). BeingDebugged flag with the value 0x00 in the bytes.

3. Nt Query Information Process () / Check Remote Debugger Present ()

CheckRemoteDebuggerPresent() is another a debugger should be attached to a process? Use this Check Remote DebuggerPresent() to decide. The API calls ntdll!ProcessDebugPort inside the kernel A value that is not zero in the DebugPort field tells that the process is being debugged in user mode by the debugger. If so, ProcessInformation will be set to 0xFFFFFFFF, otherwise the value of ProcessInformation will be 0x0. The CheckRemoteDebuggerPresent()[12] function in Kernel32 is functional. On either the 32-bit or 64-bit version of Windows, the check can be made by using this 32-bit code to look at the 32-bit window environment. The Function The function CheckRemoteDebuggerPresent() takes 2 parameters; the first parameter is the (PID), and the A pointer to a Boolean variable serves as the second parameter. That will hold TRUE if the process is being debugged. As we can see from the C/C++ code in Figure 3.

```

BOOL CheckRemoteDebuggerPresent
{
    HANDLE hProcess,
    PBOOL pbDebuggerPresent
}
  
```

Figure 3 C/C++ Code for CheckRemoteDebugger

Ntdll! NtQueryInformationProcess() has 5 parameters. To detect the debugger, the

ProcessInformation class is set to as ProcessDebugPort as we can see C/C++ code in the Figure 4.

```
NTSTATUS NTAPI NtQueryInformationProcess()
{
    HANDLE          ProcessHandle,
    PROCESSINFOCLASS ProcessInformationClass,
    PVOID           ProcessInformation,
    ULONG           ProcessInformationLength,
    PULONG          ReturnLength
}
```

Figure 4. C/C++ Code for NtQueryInformationProcess()

This example shows how the call to the CheckRemoteDebuggerPresent() and To see whether the current process is being debugged, utilize the NtQueryInformationProcess function. as we can see in Figure 5 and Figure 6.

```
; using kernel32!CheckRemoteDebuggerPresent()
lea     eax, [.hDebuggerPresent]
push    eax
push    0xffffffff
call    [CheckRemoteDebuggerPresent]
cmp     dword [.hDebuggerPresent], 0
jne     .debugger_found
```

Figure 5 Assembly code of CheckRemoteDebuggerPresent()

```
; using ntdll!NtQueryInformationProcess(ProcessDebugPort)
lea     eax, [.dwReturnLen]
push    eax
push    4
lea     eax, [.dwDebugPort]
push    eax
push    ProcessDebugPort
push    ProcessInformationClass
push    0xffffffff
call    [NtQueryInformationProcess]
dword [.dwDebugPort], 0
cpe     .debugger_found
```

Figure 6. Assembly code of NtQueryInformationProcess()

Solution:

One solution is to set NtQueryInformationProcess(return)'s value is a breakpoint. ProcessInformation is patched to a DWORD value of 0 when the breakpoint is reached of 0.

4. Nt SetInformation Thread:

NtSetInformationThread()[13] is usually used to set the priority of a thread. It can also be used

to hide threads from the debugger. It can also be done with the help of a non-documented value, which is not documented but can be used. THREAD_INFORMATION_CLASS::ThreadHideFromDebugger (0x11). When a thread is hidden in the debugger, it will not be informed of anything pertaining to that thread not be informed of anything pertaining to that thread. The thread is also capable of anti-debugging methods, such as examining debug flags, code checksums, etc. If there are hidden breakpoints in the thread, If we try to keep the main thread hidden from the debugger, either the process will crash or the debugger will gets stuck. An example of calling the NtSetInformationThread would be like this, as we can see in Figure 7.

```
push    0
push    NULL
push    ThreadHideFromDebugger
push    0xffffffff
call    [NtSetInformationThread]
```

Figure 7. Assembly code of NtSetInformationThread()

C/C++ Code:

As we can see, C/C++ code in Figure 8.

```
bool AntiDebug()
{
    NTSTATUS status = ntdll::NtSetInformationThread(
        NtCurrentThread(),
        ntdll::THREAD_INFORMATION_CLASS::ThreadHideFromDebugger,
        NULL,
        0);
    return status == 0;
}
```

Figure 8. C/C++ code of NtSetInformationThread()

Solution:

The breakpoint is set to ntdll!NtSetInformationThread(), and when the breakpoint is hit, reverse engineers can modify the EIP, to prevent the API calls from reaching the kernel and being called from other functions.

5. SwitchDesktop()

Platforms based on Windows NT allow for multiple desktop sessions. The windows of the previous active desktop can be hidden by choosing a different active desktop, but there is no visible way to return to the previous desktop. the mouse and keyboard events won't be sent to the debugger from the debugger's desktop.[13] , they no longer divulge their source, either. Debugging could become impossible as a result. Both the 32-bit and 64-bit versions of Windows can be used to make this call. Here is an example of a 32-bit version of Windows as we can see in Figure 9.



```


xop    eax, eax
push   edx
;DESKTOP_CREATEWINDOW
;+ DESKTOP_WRITEOBJECTS
;+ DESKTOP_SWITCHDESKTOP
push   182h
push   eax
push   eax
push   offset 11
call   CreateDesktop
push   eax
call   SwitchDesktop
db     "MyDesktop", 0

```

Figure 9. Assembly code of SwitchDesktop()

C/C++ Code:

As we can see the C/C++ code in the Figure 10.



```

BOOL Switch()
{
    HDESK hNewDesktop = CreateDesktop(
        _T("MyDesktop"),
        NULL,
        NULL,
        0,
        DESKTOP_CREATEWINDOW | DESKTOP_WRITEOBJECTS | DESKTOP_SWITCHDESKTOP,
        NULL);
    if (!hNewDesktop)
        return FALSE;

    return SwitchDesktop("MyDesktop");
}

```

Figure 10 C/C++ code of SwitchDesktop()

6. Execution Time / Timing Checks

When a reverse engineer tries to debug a

process and uses a single step in code, there is a significant delay between the execution of the individual's instructions[13]. The process is running under a debugger if the amount of time required is excessive compared to a typical execution. Here is a list of some instructions that can be used to increase the execution time of the instruction.

- RDTSC (Read Time-Stamp Counter)
- RDPMS (Read Performance-Monitoring Counters)
- GetLocalTime
- GetSystemTime
- GetTickCount

Now we will take an example of a timing check.

As we can see in Figure 11.



```

rdtsc
mov     eax, edx
;... more instructions
nop     eax
push    eax
pop     eax
;... more instructions
; compute delta between RDTSC instructions
rdtsc
; check high order bits
cmp     edx, eax
ja      .debugger_found
; check low order bits
sub     eax, edx
cmp     eax, 0x200
ja      .debugger_found

```

Figure 11 Assembly Code of GetTickCount()

We check the synchronization using the kernel32 GetTickCount() API or manually verify that the SharedUserData structure's TickCountLow and TickCountMultiplier entries are always set to 0xc. Identifying these timing techniques can be challenging, especially when RDTSC is used as spam, when other obscure techniques are used to mask them.

Solution:

One of the solutions is to identify where the time checks are and try to avoid stepping into them. and the code between these time checks. Reverse Engineers can place a breakpoint before that delta and execute instead of steps until a breakpoint is reached or a breakpoint is reached. We can also set a breakpoint in GetTickCount() to specify where to call it or to change its return value. Mitigations During Debugging: just fill time checks with NOPs and set the result of these checks to the appropriate value. For anti-debugging solution development: there is no great need to do anything with it, as time checks are not very reliable, but you can still hook timing functions and accelerate the time between calls.

Mitigations:

- During Debugging, just fill time checks with NOPs and set the result of these checks to the appropriate value.
- For anti-debugging solution development: there is no great need to do anything with it, as time checks are not very reliable, but you can still hook timing functions and accelerate the time between calls.

7. SeDebugPrivilege:

By default, the SeDebugPrivilege permission is disabled for the process access token. When a debugger like x32dbg, OllyDBG, etc. loads a process, SeDebugPrivilege permission is enabled. This is because these debuggers keep trying . SeDebugPrivilege permissions are inherited.

If the process can open the CSRSS.EXE process, then SeDebugPrivilege is active when the process is accessed.

Token pointing to the process being debugged. The test is valid for the following reasons: The Process Security Descriptor CSRSS.EXE allows the system access to the process.

However, if the process has SeDebugPrivilege privilege, other processes have independent access to the Security Descriptor. This permission is only granted to administrative groups by default, as we can see in Figure 12.

```

; query for the PID of CSRSS.EXE
call [CsrGetProcessId]

; try to open the CSRSS.EXE process
push     eax
push     FALSE
push     PROCESS_QUERY_INFORMATION
call     [OpenProcess]

; if OpenProcess() was successful
; process is probably being debugged
test     eax, eax
jnz      .debugger_found

```

Figure 12 Assembly Code of SeDebugPrivilege()

This control uses ntdll! The CSRSS.exe GetProcessId() API gets the Process ID (PID) from CSRSS.EXE. You can get it manually by looking at the Process ID CSRSS.EXE processes. If OpenProcess() succeeds, SeDebugPrivilege is activated, indicating that the process is currently running and debugging, too.

Solution:

The ntdll breakpoint can be hit by setting a breakpoint as a solution. Returns from NtOpenProcess(). If PID passed by CSRSS.exe is CSRSS.exe, set the EX-value to 0xC0000022 (STATUS_ACCESS_DENIED).

Parent Process:

Users launch apps by clicking on the executable's icon that the shell process displays (Explorer.exe). By clicking on the executable's icon that the shell process displays, users can launch apps (Explorer.exe). Due to this, Explorer.exe becomes the parent process of the active process. This will show that the program was created by someone else and suggest that you can debug it.

1. Using Process32First/Next(), it will list every process and note explorer.exe. PROCESS32.szExeFile and the PROCESSENTRY32.th32parentProcessID are the two fields that provide the process ID and the parent process ID of the current process, respectively.
2. The target is being debugged if the Process ID (PID) of the parent process differs from the Process ID (PID) of the explorer.exe.

Solution:

We need to patch the element of Kernel32!Process32NextW() that contains the code that performs a return after setting the value of EAX to 0.

8. Debugger Window:

The presence of the debug window is a flag that the debugger is running system[13]. Because the debugger creates windows with special class names (OllyDBG for OLLYDBG and WinDbgFrameClass for WinDbg), user32 can easily identify these debug windows! FindWindow() or User32! FindWindowEx().

Solution:

One solution is to set breakpoints in FindWindow() and FindWindowEx() When the breakpoint is hit, modify the value of the lpClassName string parameter to prevent the API from functioning. Setting the return value to NULL is another option.

9. Debugger Process:

List all the processes on the system and see whether the process name matches the name of the debugger to find out if it is currently running (for example, OLLYDBG.EXE, windbg.exe, etc.). Simple to implement; just use Process32First / Next() after confirming that the image name corresponds to the name of the debugger.

Sometimes these methods also use Kernel32 ReadProcessMemory() to read process memory and then look for debugger-related strings such as "x64dbg", "IDA", "OllyDBG", etc. to reverse engineer the debugger. To implement. After getting the debugger. The malware will stop his execution and silently exit or terminates the process.

Solution:

Another solution is to check the main process, including patching the kernel 32 patch! Process32NextW() always fails and prevents the developer from enumerating the process.

10. Device Drivers

An old technique is to verify that the debugger is running in a Kernel Mode in the system and try to, access device drivers. This technique is very simple and consists of simply making a call to the against well-known device names

used by kernel-mode debuggers, such as SoftICE, using `Kernel32!CreateFile()`. Some versions of Soft-ICE also add numbers to the device name, making it to check. The reversing forum's suggested technique is to brute force the corresponding digits until the right device name is discovered[14]. The new packer also uses device driver detection techniques to detect system monitors such as "Process Monitor" etc.

Solution:

Establishing a breakpoint in `kernel32` is the simple fix. When the breakpoint is reached, `CreateFileFileW()` should either handle the `FileName` parameter or alter its return value to `INVALID_HANDLE_VALUE (0xFFFFFFFF)`.

Process Memory:

A process can check or interact with its own memory for the presence of a debugger. This section includes anti-hitch methods[15] such as process memory and thread context checking, breakpoint DETECTION, PATCHING function and debugging functions.

11. Breakpoint and Patching Detection:

To verify if our code has any software breakpoints, we may still inspect the process memory, and we can also check the CPU debug registers to see if any hardware breakpoints have been set.

12. Software Breakpoints Detection:

Software breakpoints are defined as breakpoints that are created by altering the code at the target location and replacing it with the

byte value `0xCC` (`INT3 / Breakpoint Interrupt`)[17]. Finding the byte `0xCC` in the API code and protector code will help you locate software breakpoints as seen by the example of assembly code in Figure 13.

```

cld
mov     edi, Protected_Code_Start
mov     ecx, Protected_Code - Protected_Code_Start
mov     al, 0xCC

repne   scasd
jnz     .breakpoint_found
if      (byte XOR 0x55 == 0x99) then breakpoint found
where   0x99 == 0xCC XOR 0x55
  
```

Figure 13 Assembly Code of Software Breakpoint Detection

C/C++ Code:

As we can C/C++ code in the Figure 14.

```

bool CheckForSpecificByte(BYTE cByte, PVOID pMemory, SIZE_T nMemorySize)
{
    PVOID pBytes = (PBYTE)pMemory;
    for (SIZE_T i = 0; i < nMemorySize; i++)
    {
        // Break on RET (0xC3) if we don't know the function's size
        if (((nMemorySize > 0) && (i < nMemorySize)) &&
            ((nMemorySize == 0) && (pBytes[i] == 0xC3)))
            break;

        if (pBytes[i] == cByte)
            return true;
    }
    return false;
}

bool IsDebugged()
{
    PVOID functionsToCheck[] = {
        0Function1,
        0Function2,
        0Function3,
    };

    for (auto funcAddr : functionsToCheck)
    {
        if (CheckForSpecificByte(0xCC, funcAddr))
            return true;
    }
    return false;
}
  
```

Figure 14. C/C++ code of Software Breakpoint Detection

Solution:

Hardware breakpoints can be reverse engineered if software breakpoints are identified. If you need to set a breakpoint in the API code, and when the packer tries to find a breakpoint in the API code, reverse engineering the UNICODE API version allows for the setting of breakpoints. That eventually calls the ANSI version, such as `LoadLibraryExW` `LoadLibrar-`

yA or the native API corresponding to Load-DLL to replace.

13. Hardware Breakpoints:

DR0, DR1, DR2, and DR3 are debug registers that can be obtained from the thread context. Debug registers 0-3 are used to store virtual address of the so-called hardware breakpoints. C/C++ Code:

As we can see C/C++ code in the figure 15.

```
bool IsDebugged()
{
    CONTEXT ctx;
    ZeroMemory(&ctx, sizeof(CONTEXT));
    ctx.ContextFlags = CONTEXT_DEBUG_REGISTERS;

    if (!GetThreadContext(GetCurrentThread(), &ctx))
        return false;

    return ctx.Dr0 || ctx.Dr1 || ctx.Dr2 || ctx.Dr3;
}
```

Figure 15 C/C++ code of Hardware Breakpoints

14. Memory Checks:

This section includes methods for directly inspecting or modifying a process's virtual memory in order to spot and stop debugging[18].

15. Nt Query Virtual Memory ():

The memory page of the process in which the code is located is shared by all processes prior to the page being written. Then the OS creates a replica of this page and allocates it to the process's virtual memory[19], so the page is no longer "shared". Now we can see how to declare NTDLL, as we can see in figure 16.

NTDLL declarations:

```
namespace ntddll
{
    // ...
    #define STATUS_INFO_LENGTH_BEHATCH 0xC0000004
    // ...

    typedef enum _MEMORY_INFORMATION_CLASS {
        MemoryBasicInformation,
        MemoryWorkingSetList,
    } MEMORY_INFORMATION_CLASS;

    // ...

    typedef union _PSAPI_WORKING_SET_BLOCK {
        ULONG_PTR;
        struct {
            ULONG Protection : 5;
            ULONG ShareCount : 3;
            ULONG Shared : 1;
            ULONG Reserved : 3;
            ULONG VirtualPage : 20;
        };
    } PSAPI_WORKING_SET_BLOCK, *PPSAPI_WORKING_SET_BLOCK;

    typedef struct _MEMORY_WORKING_SET_LIST {
        ULONG NumberOfPages;
        PSAPI_WORKING_SET_BLOCK WorkingSetList[1];
    } MEMORY_WORKING_SET_LIST, *PMEMORY_WORKING_SET_LIST;

    // ...
}
```

Figure 16 NTDLL Declaration of NtQueryVirtualMemory()

C/C++ Code:

As we can see the C/C++ code in the Figure 17.

```
bool IsDebugged()
{
    CONTEXT ctx;
    ZeroMemory(&ctx, sizeof(CONTEXT));
    ctx.ContextFlags = CONTEXT_DEBUG_REGISTERS;

    if (!GetThreadContext(GetCurrentThread(), &ctx))
        return false;

    if (ctx.Dr0 || ctx.Dr1 || ctx.Dr2 || ctx.Dr3)
        return true;

    if (!NtQueryVirtualMemory(GetCurrentProcess(),
        (PVOID)0,
        MemoryWorkingSetList,
        &ctx.Dr0,
        &ctx.Dr1,
        &ctx.Dr2,
        &ctx.Dr3))
        return true;

    return false;
}
```

Figure 17 C/C++ Code for Hardware Breakpoints

16. Detecting A function Patch:

Calling kernel32 is a common approach to find a debugger. IsDebuggerPresent(). By altering the outcome in the EAX register or hacking the kernel32, you may easily get around this check! IsDebuggerPresent(). Instead of

looking for breakpoints in the process memory, we can check to see if `kernel32IsDebuggerPresent()` has been altered[20]. The first few bytes of this function can be read and compared to the same function's bytes from other processes. Windows libraries are loaded at the same base address throughout the process, even if the Address Space Layout Randomization (ASLR) feature is enabled. The base address only changes across reboots but remains the same for the duration of the session.

Mitigations:

- During Debugging: Enter the function that conducts the Step-Over check and run it till the end(Ctrl + F9).
- Finding the specific check and either path it with NOPs or setting the return to a value that permits the application to keep running are the best ways to mitigate all "memory" techniques, including anti-step over.

Conclusion:

To defend itself against reverse engineering analysis, the malware employs anti-debugging techniques. Debug analysis can be avoided by anti-debugging techniques. Reverse engineers need advanced debuggers and knowledge to analyze malware using anti-debugging techniques. By applying common sense and slowly debugging the process, it is possible to identify the majority of anti-debugging techniques. For example, if you see that the code is terminating too rapidly in a conditional jump, which could mean preventing debugging technical. The most widely used anti-debugging methods involve fs access: [30h] by using a Windows API or performing a time check.

Of course, as with all malware analysis, the best way to learn how to stop it by using debugging techniques by continuously testing malware. Malware developers are constantly coming up with new techniques to evade debuggers and keep security researchers like you on their toes.

17. References:

- [1] V. Bhardwaj, V. Kukreja, C. Sharma, I. Kansal, and R. Popali, "Reverse Engineering-A Method for Analyzing Malicious Code Behavior," in *2021 International Conference on Advances in Computing, Communication, and Control (ICAC3)*, Dec. 2021, pp. 1–5. doi: 10.1109/ICAC353642.2021.9697150.
- [2] M. N. Gagnon, S. Taylor, and A. K. Ghosh, "Software Protection through Anti-Debugging," *IEEE Security & Privacy Magazine*, vol. 5, no. 3, pp. 82–84, May 2007, doi: 10.1109/M-SP.2007.71.
- [3] J.-W. Kim, J. Bang, Y.-S. Moon, and M.-J. Choi, "Disabling Anti-Debugging Techniques for Unpacking System in User-level Debugger," in *2019 International Conference on Information and Communication Technology Convergence (ICTC)*, Oct. 2019, pp. 954–959. doi: 10.1109/ICTC46691.2019.8939719.
- [4] T. Akhtar, B. B. Gupta, and S. Yamaguchi, "Malware propagation effects on SCADA system and smart power grid," in *2018 IEEE International Conference on Consumer Electronics (ICCE)*, Jan.

- 2018, pp. 1–6. doi: 10.1109/ICCE.2018.8326281.
- [5] G. Wang, L. Zhuang, T. Liu, S. Li, S. Yang, and J. Lan, “Formal analysis and verification of industrial control system security via timed automata,” in *2020 International Conference on Internet of Things and Intelligent Applications (ITIA)*, Nov. 2020, pp. 1–5. doi: 10.1109/ITIA50152.2020.9312289.
- [6] A. J. Smith, R. F. Mills, A. R. Bryant, G. L. Peterson, and M. R. Grimaila, “REDIR: Automated static detection of obfuscated anti-debugging techniques,” in *2014 International Conference on Collaboration Technologies and Systems (CTS)*, May 2014, pp. 173–180. doi: 10.1109/CTS.2014.6867561.
- [7] J. Raber, “Stealthy Profiling and Debugging of Malware Trampolining from User to Kernel Space,” in *2011 18th Working Conference on Reverse Engineering*, Oct. 2011, pp. 431–432. doi: 10.1109/WCRE.2011.62.
- [8] J. G. Alcalde, G. Chua, I. M. Demabildo, M. A. Ong, and R. L. Uy, “CALVIS32: Customizable assembly language visualizer and simulator for intel x86-32 architecture,” in *2016 IEEE Region 10 Conference (TENCON)*, Nov. 2016, pp. 214–217. doi: 10.1109/TENCON.2016.7847992.
- [9] Chan Lee Yee, Lee Ling Chuan, M. Ismail, and N. Zainal, “A static and dynamic visual debugger for malware analysis,” in *2012 18th Asia-Pacific Conference on Communications (APCC)*, Oct. 2012, pp. 765–769. doi: 10.1109/APCC.2012.6388211.
- [10] Xu Chen, J. Andersen, Z. M. Mao, M. Bailey, and J. Nazario, “Towards an understanding of anti-virtualization and anti-debugging behavior in modern malware,” in *2008 IEEE International Conference on Dependable Systems and Networks With FTCS and DCC (DSN)*, 2008, pp. 177–186. doi: 10.1109/DSN.2008.4630086.
- [11] P. Chen, C. Huygens, L. Desmet, and W. Joosen, “Advanced or Not? A Comparative Study of the Use of Anti-debugging and Anti-VM Techniques in Generic and Targeted Malware,” 2016, pp. 323–336. doi: 10.1007/978-3-319-33630-5_22.
- [12] P. Xie, X. Lu, Y. Wang, J. Su, and M. Li, “An Automatic Approach to Detect Anti-debugging in Malware Analysis,” 2013, pp. 436–442. doi: 10.1007/978-3-642-35795-4_55.
- [13] A. Mylonas and D. Gritzalis, “Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software,” *Comput Secur*, vol. 31, no. 6, pp. 802–803, Sep. 2012, doi: 10.1016/j.cose.2012.05.004.
- [14] P. Chen, C. Huygens, L. Desmet, and W. Joosen, “Advanced or Not? A Comparative Study of the Use of Anti-debugging and Anti-VM Techniques in Generic and Targeted Malware,” 2016, pp. 323–336. doi: 10.1007/978-3-319-33630-5_22.
- [15] J.-W. Kim, J. Namgung, Y.-S. Moon, and M.-J. Choi, “Experimental Compar-

- ison of Machine Learning Models in Malware Packing Detection,” in *2020 21st Asia-Pacific Network Operations and Management Symposium (APNOMS)*, Sep. 2020, pp. 377–380. doi: 10.23919/APNOMS50412.2020.9237007.
- [16] R. R. Branco and G. N. Barbosa, “Distributed malware analysis scheduling,” in *2011 6th International Conference on Malicious and Unwanted Software*, Oct. 2011, pp. 34–41. doi: 10.1109/MALWARE.2011.6112324.
- [17] K. Coogan, S. Debray, T. Kaochar, and G. Townsend, “Automatic Static Unpacking of Malware Binaries,” in *2009 16th Working Conference on Reverse Engineering*, 2009, pp. 167–176. doi: 10.1109/WCRE.2009.24.
- [18] G. Jeong, E. Choo, J. Lee, M. Bat-Erdene, and H. Lee, “Generic unpacking using entropy analysis,” in *2010 5th International Conference on Malicious and Unwanted Software*, Oct. 2010, pp. 98–105. doi: 10.1109/MALWARE.2010.5665789.
- [19] C. R. Hill, “A real-time microprocessor debugging technique,” *ACM SIGPLAN Notices*, vol. 18, no. 8, pp. 145–148, Aug. 1983, doi: 10.1145/1006142.1006179.
- [20] R. Sihwail, K. Omar, K. Zainol Ariffin, and S. al Afghani, “Malware Detection Approach Based on Artifacts in Memory Image and Dynamic Analysis,” *Applied Sciences*, vol. 9, no. 18, p. 3680, Sep. 2019, doi: 10.3390/app9183680.

Editorial Policy and Guidelines for Authors

IJECE is an open access, peer reviewed quarterly Journal published by LGU Society of Computer Sciences. The Journal publishes original research articles and high quality review papers covering all aspects of Computer Science and Technology.

The following note set out some general editorial principles. A more detailed style document can be download at www.research.lgu.edu.pk is available. All queries regarding publications should be addressed to editor at email IJECE@lgu.edu.pk. The document must be in word format, other format like pdf or any other shall not be accepted.

The format of paper should be as follows:

- Title of the study (center aligned, font size 14)
- Full name of author(s) (center aligned, font size 10)
- Name of Department
- Name of Institution
- Corresponding author email address.
- Abstract
- Keywords
- Introduction
- Literature Review
- Theoretical Model/Framework and Methodology
- Data analysis/Implementation/Simulation
- Results/ Discussion and Conclusion
- References.

Heading and sub-heading should be differentiated by numbering sequences like, 1. HEADING (Bold, Capitals) 1.1 Subheading (Italic, bold) etc. The article must be typed in Times New Roman with 12 font size 1.5 space, and should have margin 1 inches on the left and right. Length of paper should not be longer than 15 pages, including figures, tables, exhibits and bibliography. Table must have standard caption at the top while figures below with. Figure and table should be in continues numbering. Citation must be in according to the IEEE 2006 style

LAHORE GARRISON UNIVERSITY

*L*ahore Garrison University has been established to achieve the goal of excellence and quality education in minimum possible time. Lahore Garrison University in the Punjab metropolis city of Lahore is an important milestone in the history of higher education in Pakistan. In order to meet the global challenges, it is necessary to touch the highest literacy rates while producing skillful and productive graduates in all fields of knowledge.

VISION

*O*ur vision is to prepare a generation that can take the lead and put this nation on the path to progress and prosperity through applying their knowledge, skills and dedication. We are committed to help individuals and organizations in discovering their God-gifted potentials to achieve ultimate success actualizing the highest standards of efficiency, effectiveness, excellence, equity, trusteeship and sustainable development of global human society.

MISSION

*A*t present, LGU is running Undergraduate, Graduate, Masters, M.Phil. and Ph.D. programs in various disciplines. Our mission is to serve the society by equipping the upcoming generations with valuable knowledge and latest professional skills through education and research. We also aim to evolve new realities and foresight by unfolding new possibilities. We intend to promote the ethical, cultural and human values in our participants to make them educated and civilized members of society.

Contact: For all inquiries, regarding call for papers, submission of research articles and correspondence, kindly contact at this address:

Sector C, DHA Phase-VI Lahore, Pakistan

Phone: +92- 042-37181823

Email: ijeci@lgu.edu.pk

