



# **NAVAL POSTGRADUATE SCHOOL**

**MONTEREY, CALIFORNIA**

## **THESIS**

**AN EXPANDED GRAPHICAL USER INTERFACE FOR  
WEB-BASED CYBER AUTOMATED RED TEAMING  
TOOL (CARTT)**

by

Ousmane M. Goumandakoye

September 2021

Thesis Advisor:

Co-Advisor:

Second Reader:

Alan B. Shaffer

Rudolph P. Darken

Gurminder Singh

**Approved for public release. Distribution is unlimited.**

THIS PAGE INTENTIONALLY LEFT BLANK

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                                                                 |                                                                |                                                         |                                                            |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------|----------------------------------------------------------------|---------------------------------------------------------|------------------------------------------------------------|
| <b>REPORT DOCUMENTATION PAGE</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                                                                 |                                                                | <i>Form Approved OMB<br/>No. 0704-0188</i>              |                                                            |
| Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                                                                 |                                                                |                                                         |                                                            |
| <b>1. AGENCY USE ONLY</b><br>(Leave blank)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                                                                 | <b>2. REPORT DATE</b><br>September 2021                        |                                                         | <b>3. REPORT TYPE AND DATES COVERED</b><br>Master's thesis |
| <b>4. TITLE AND SUBTITLE</b><br>AN EXPANDED GRAPHICAL USER INTERFACE FOR WEB-BASED CYBER AUTOMATED RED TEAMING TOOL (CARTT)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                                                                 |                                                                | <b>5. FUNDING NUMBERS</b>                               |                                                            |
| <b>6. AUTHOR(S)</b> Ousmane M. Goumandakoye                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                                                                 |                                                                |                                                         |                                                            |
| <b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</b><br>Naval Postgraduate School<br>Monterey, CA 93943-5000                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                                                                 |                                                                | <b>8. PERFORMING ORGANIZATION REPORT NUMBER</b>         |                                                            |
| <b>9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b><br>N/A                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                                                                 |                                                                | <b>10. SPONSORING / MONITORING AGENCY REPORT NUMBER</b> |                                                            |
| <b>11. SUPPLEMENTARY NOTES</b> The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                                                                 |                                                                |                                                         |                                                            |
| <b>12a. DISTRIBUTION / AVAILABILITY STATEMENT</b><br>Approved for public release. Distribution is unlimited.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                                                                 |                                                                | <b>12b. DISTRIBUTION CODE</b><br>A                      |                                                            |
| <b>13. ABSTRACT (maximum 200 words)</b><br><br>The Department of Defense's (DOD) Cyber Red Teams are often a group of ethical hackers and offensive-trained security professionals who must continuously stay ahead of cybercriminals' known tactics, techniques, and procedures. The Cyber Red Team certification and accreditation life cycle requires ongoing funding and time, while the United States' near-peer state actors and non-state actors are continuously trying to breach its cybersecurity defenses. The information technology infrastructure must be continuously assessed for vulnerabilities; failing to do so will impact United States national security. These assessments require time and qualified security personnel, both of which are in short supply. The Cyber Automated Red Team Tool (CARTT) was previously developed at NPS to address personnel, time, and cost-related shortages in Cyber Red Teams. This research extends CARTT's capabilities by implementing a web-based front-end graphical user interface (GUI) that enables non-expert users to identify vulnerabilities and launch exploits to help mitigate network vulnerabilities. This extended CARTT increases the DOD's network systems' cyber-readiness posture by providing necessary fixes, patches, and remediations. |                                                                 |                                                                |                                                         |                                                            |
| <b>14. SUBJECT TERMS</b><br>DOD cyber red teams, red team in a box, RTIB, Cyber Automated Red Team Tool, CARTT, user interface, user experience                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                                                                 |                                                                | <b>15. NUMBER OF PAGES</b><br>87                        |                                                            |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                                                                 |                                                                | <b>16. PRICE CODE</b>                                   |                                                            |
| <b>17. SECURITY CLASSIFICATION OF REPORT</b><br>Unclassified                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | <b>18. SECURITY CLASSIFICATION OF THIS PAGE</b><br>Unclassified | <b>19. SECURITY CLASSIFICATION OF ABSTRACT</b><br>Unclassified | <b>20. LIMITATION OF ABSTRACT</b><br>UU                 |                                                            |

THIS PAGE INTENTIONALLY LEFT BLANK

**Approved for public release. Distribution is unlimited.**

**AN EXPANDED GRAPHICAL USER INTERFACE FOR WEB-BASED CYBER  
AUTOMATED RED TEAMING TOOL (CARTT)**

Ousmane M. Goumandakoye  
Major, United States Marine Corps  
BS, University of Arizona, 2010  
MS, Embry-Riddle Aeronautical University, 2015

Submitted in partial fulfillment of the  
requirements for the degree of

**MASTER OF SCIENCE IN COMPUTER SCIENCE**

from the

**NAVAL POSTGRADUATE SCHOOL  
September 2021**

Approved by: Alan B. Shaffer  
Advisor

Rudolph P. Darken  
Co-Advisor

Gurminder Singh  
Second Reader

Gurminder Singh  
Chair, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

## **ABSTRACT**

The Department of Defense's (DOD) Cyber Red Teams are often a group of ethical hackers and offensive-trained security professionals who must continuously stay ahead of cybercriminals' known tactics, techniques, and procedures. The Cyber Red Team certification and accreditation life cycle requires ongoing funding and time, while the United States' near-peer state actors and non-state actors are continuously trying to breach its cybersecurity defenses. The information technology infrastructure must be continuously assessed for vulnerabilities; failing to do so will impact United States national security. These assessments require time and qualified security personnel, both of which are in short supply. The Cyber Automated Red Team Tool (CARTT) was previously developed at NPS to address personnel, time, and cost-related shortages in Cyber Red Teams. This research extends CARTT's capabilities by implementing a web-based front-end graphical user interface (GUI) that enables non-expert users to identify vulnerabilities and launch exploits to help mitigate network vulnerabilities. This extended CARTT increases the DOD's network systems' cyber-readiness posture by providing necessary fixes, patches, and remediations.

THIS PAGE INTENTIONALLY LEFT BLANK

## TABLE OF CONTENTS

|             |                                                               |           |
|-------------|---------------------------------------------------------------|-----------|
| <b>I.</b>   | <b>INTRODUCTION.....</b>                                      | <b>1</b>  |
| <b>A.</b>   | <b>PROBLEM STATEMENT .....</b>                                | <b>2</b>  |
| <b>B.</b>   | <b>SCOPE .....</b>                                            | <b>3</b>  |
| <b>C.</b>   | <b>BENEFITS OF STUDY.....</b>                                 | <b>3</b>  |
| <b>D.</b>   | <b>CHAPTER OUTLINE.....</b>                                   | <b>4</b>  |
| 1.          | Chapter II: Background.....                                   | 4         |
| 2.          | Chapter III: Design Methodology .....                         | 4         |
| 3.          | Chapter IV: Usability Testing and Evaluations.....            | 4         |
| 4.          | Chapter V: Conclusion and Future Work.....                    | 4         |
| <b>II.</b>  | <b>BACKGROUND .....</b>                                       | <b>5</b>  |
| <b>A.</b>   | <b>CYBERSECURITY TEAMS OPERATIONS AND TOOLS.....</b>          | <b>5</b>  |
| 1.          | Cyber Teams Roles, Responsibilities, and Expectations .....   | 5         |
| 2.          | Cybersecurity Red Team Penetration Testing Methods .....      | 7         |
| 3.          | DOD Cyber Red Teams' Operations and Rules of Engagement ..... | 10        |
| <b>B.</b>   | <b>CYBER AUTOMATED RED TEAM TOOL .....</b>                    | <b>10</b> |
| 1.          | Current CARTT Roles .....                                     | 11        |
| 2.          | CARTT Tools .....                                             | 12        |
| 3.          | CARTT Architecture Overview .....                             | 13        |
| 4.          | CARTT Server overview .....                                   | 13        |
| 5.          | Exploitation Process-Operator .....                           | 14        |
| 6.          | Exploitation Process Flow .....                               | 15        |
| 7.          | User Operations Overview .....                                | 16        |
| 8.          | User Experience Design Considerations .....                   | 16        |
| <b>C.</b>   | <b>USER INTERFACE ACCESS CONTROLS AND SECURITY.....</b>       | <b>18</b> |
| 1.          | Authentication and Authorization.....                         | 18        |
| 2.          | Security .....                                                | 20        |
| 3.          | Testing.....                                                  | 21        |
| <b>D.</b>   | <b>CHAPTER SUMMARY.....</b>                                   | <b>21</b> |
| <b>III.</b> | <b>DESIGN METHODOLOGY .....</b>                               | <b>23</b> |
| <b>A.</b>   | <b>VISUAL DESIGN.....</b>                                     | <b>23</b> |
| <b>B.</b>   | <b>PROTOTYPE .....</b>                                        | <b>23</b> |
| <b>C.</b>   | <b>CARTT UI OVERVIEW .....</b>                                | <b>23</b> |
| <b>D.</b>   | <b>NEEDS ANALYSIS/MISSION REQUIREMENTS .....</b>              | <b>23</b> |
| <b>E.</b>   | <b>USER ANALYSIS .....</b>                                    | <b>24</b> |

|     |                                                 |    |
|-----|-------------------------------------------------|----|
| F.  | PREVIOUS CARTT UI WALKTHROUGH.....              | 25 |
| G.  | USER INTERFACE DESIGN GUIDELINES .....          | 32 |
| H.  | CARTT’S NEW DESIGN .....                        | 36 |
| 1.  | Login Page .....                                | 37 |
| 2.  | Commander Page.....                             | 39 |
| 3.  | Message Page.....                               | 40 |
| 4.  | Administrator Page.....                         | 41 |
| 5.  | Operator Main Menu .....                        | 43 |
| 6.  | Create New Scan .....                           | 45 |
| 7.  | Check Scan Status.....                          | 46 |
| 8.  | Import Completed Scan .....                     | 47 |
| 9.  | Begin Target Exploitation .....                 | 48 |
| I.  | CHAPTER SUMMARY.....                            | 48 |
| IV. | USABILITY TESTING AND EVALUATIONS.....          | 49 |
| A.  | SCENARIO TESTING.....                           | 49 |
| 1.  | User Briefing.....                              | 50 |
| B.  | SCENARIO DESIGN .....                           | 50 |
| 1.  | High Level Task List.....                       | 50 |
| 2.  | User Scenarios .....                            | 51 |
| C.  | USABILITY SPECIFICATION.....                    | 52 |
| 1.  | Data Collection .....                           | 54 |
| D.  | RESULTS .....                                   | 54 |
| 1.  | Result Analysis and Discussion.....             | 59 |
| 2.  | Conclusions and Recommendations.....            | 60 |
| E.  | CHAPTER SUMMARY.....                            | 60 |
| V.  | CONCLUSIONS AND FUTURE WORK.....                | 61 |
| A.  | SUMMARY .....                                   | 61 |
| B.  | CONCLUSION .....                                | 61 |
| 1.  | Primary Research Question .....                 | 62 |
| 2.  | Second Research Question .....                  | 62 |
| C.  | FUTURE WORK.....                                | 62 |
| 1.  | Improve the Backend of the User Interface ..... | 62 |
| 2.  | CARTT User Expansion.....                       | 63 |
| 3.  | Improve the Scan Result .....                   | 63 |
|     | LIST OF REFERENCES .....                        | 65 |
|     | INITIAL DISTRIBUTION LIST .....                 | 69 |

## LIST OF FIGURES

|            |                                                                        |    |
|------------|------------------------------------------------------------------------|----|
| Figure 1.  | Comparing Red, Purple, and Blue Cybersecurity Teams. Source: [12]..... | 6  |
| Figure 2.  | Sample CARTT Architecture. Source: [4].....                            | 13 |
| Figure 3.  | CARTT Server Diagram. Source: [4]. ....                                | 14 |
| Figure 4.  | CARTT Previous Login Page. Source: [4]. ....                           | 19 |
| Figure 5.  | Previous CARTT Operator Main Page.....                                 | 26 |
| Figure 6.  | Previous CARTT Operator Scan Page.....                                 | 26 |
| Figure 7.  | Previous CARTT Scan Status Page .....                                  | 27 |
| Figure 8.  | Previous CARTT Scan Import Page.....                                   | 28 |
| Figure 9.  | Previous CARTT Exploitation System.....                                | 28 |
| Figure 10. | Previous Target Host IP Address Query .....                            | 29 |
| Figure 11. | Common Vulnerabilities and Exposures Query .....                       | 29 |
| Figure 12. | Modules Query.....                                                     | 30 |
| Figure 13. | CARTT Exploitation Test Page.....                                      | 30 |
| Figure 14. | Previous CARTT Administrator Menu.....                                 | 31 |
| Figure 15. | Corresponding Role Numbers.....                                        | 36 |
| Figure 16. | Human Readable Mapping .....                                           | 37 |
| Figure 17. | Password Input Control.....                                            | 37 |
| Figure 18. | New CARTT Login Page.....                                              | 38 |
| Figure 19. | New CARTT Registration Page.....                                       | 39 |
| Figure 20. | New CARTT Commander Page .....                                         | 40 |
| Figure 21. | New CARTT Message Page .....                                           | 41 |
| Figure 22. | New CARTT Administrator Page.....                                      | 42 |

|            |                                                  |    |
|------------|--------------------------------------------------|----|
| Figure 23. | New CARTT Administrator Delete Account Page..... | 43 |
| Figure 24. | New CARTT Operator Main Menu Page .....          | 44 |
| Figure 25. | Left Side Navigation Panel <Div>.....            | 44 |
| Figure 26. | CARTT Logout and Redirect to Login Page .....    | 45 |
| Figure 27. | New CARTT Operator Scan Network Menu Page.....   | 46 |
| Figure 28. | New CARTT Check Previous Scan Page .....         | 47 |
| Figure 29. | New CARTT Import Completed Page.....             | 48 |

## LIST OF TABLES

|          |                                      |    |
|----------|--------------------------------------|----|
| Table 1. | Operator Usability Scale .....       | 52 |
| Table 2. | Operator Tasks .....                 | 55 |
| Table 3. | Administrator Tasks.....             | 57 |
| Table 4. | Commander Tasks .....                | 58 |
| Table 5. | Task Success Rate Criteria.....      | 58 |
| Table 6. | High-Level Task Success Results..... | 59 |

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF ACRONYMS AND ABBREVIATIONS

|       |                                                    |
|-------|----------------------------------------------------|
| APT   | advanced persistent threat                         |
| AD    | active directory                                   |
| CARTT | Cyber Automated Red Team Tool                      |
| CCDM  | combatant command                                  |
| CIA   | confidentiality, integrity, and availability       |
| CLI   | command-line interface                             |
| CSS   | cascading style sheets                             |
| CSSP  | Cyberspace Security Provider                       |
| C2    | command and control                                |
| CVE   | Common Vulnerabilities and Exposures               |
| DHCP  | Dynamic Host Configuration Protocol                |
| DISA  | Defense Information Systems Agency                 |
| Div   | division/section                                   |
| DNS   | Domain Name System                                 |
| DOD   | Department of Defense                              |
| DOT&E | Director, Operational Test and                     |
| DOS   | Denial of Service                                  |
| GOA   | Government Accountability Office                   |
| GUI   | graphical user interface                           |
| HML   | Hypertext Markup Language                          |
| HTTP  | Hypertext Transfer Protocol                        |
| IG    | inspector general                                  |
| IT    | information technology                             |
| IP    | internet protocol                                  |
| IPSec | Internet Protocol Security                         |
| MCCOG | Marine Corps Cyberspace Operations Group           |
| MITSC | Marine Corps Information Technology Support Center |
| MSF   | Metasploit Framework                               |
| MySQL | My Structured Query Language                       |
| NIST  | National Institute of Science and Technology       |

|            |                                                  |
|------------|--------------------------------------------------|
| Nmap       | network mapper                                   |
| NPS        | Naval Postgraduate School                        |
| NSA        | National Security Agency                         |
| OCO        | offensive cyber operations                       |
| OpenVAS    | Open Vulnerability Assessment System             |
| OSD        | Office of the Secretary of Defense               |
| OT&E       | operations, operational test and evaluation      |
| OU         | Organizational Unit                              |
| Pentesters | penetration testers                              |
| PTES       | Penetration Testing Execution Standard           |
| PHP        | hypertext preprocessor                           |
| RBAC       | Role-Based Access Control                        |
| RDBMS      | relational database management system            |
| RNOSC      | Regional Network Operations and Security Centers |
| ROE        | Rules of Engagement                              |
| RTIB       | Red Team in a Box                                |
| SQL        | structured query language                        |
| TTP        | tactics, techniques, and procedures              |
| UI         | user interface                                   |
| UIC        | unit identification code                         |
| USG        | United States Government                         |
| USSTRATCOM | U.S. Strategic Command                           |
| UX         | user experience                                  |
| VPN        | Virtual Private Networks                         |
| XML        | extensible markup language                       |
| XSS        | Cross-Site Scripting                             |

## **ACKNOWLEDGMENTS**

I want to thank the military for giving me the tools and opportunity to attend the Naval Postgraduate School. Special thanks to my advisors, Dr. Gurminder Singh, Dr. Alan Shaffer, and Dr. Rudolph Darken; my NPS Graduate Writing Center writing coach, Daniel J. Lehnherr; and the entire computer science faculty for their patience, guidance, and constant encouragement throughout this journey. Lastly, thanks to the United States Marine Corps for giving the opportunity to further my education.

Lastly and most importantly, thanks to God for giving me the opportunity to graduate; to my wife, Hassana; our sons, Farid, and Aydin; and to my parents for their constant support and guidance. I am truly grateful for the time, effort, and opportunity.

THIS PAGE INTENTIONALLY LEFT BLANK

## I. INTRODUCTION

The Cyber Automated Red Team Tool (CARTT) software was developed at the Naval Postgraduate School (NPS) to address the problems identified in the Department of Defense (DOD) Inspector General's March 2020 report [1]. The report outlines the constraints in maintaining and training the required Cyber Red Teams to continuously test DOD Information Technology (IT) infrastructures. The fast-growing cyber threats coupled with limited Red Team assets create a situation where the DOD needs to increase the number of trained advanced Cyber Red Teams [1].

Vulnerability assessments of DOD systems, networks, and facilities can take weeks or months, depending on the size of the network, complexity of the application architecture, and depth of the evaluation required. Cyber Red Teams are in acute shortage throughout the Department of the Navy (DON) due to the large amount of training and personnel required to maintain its networks. There are only a few available Red Teams regularly available to train system and network administrators.

Cyber Red Teams are groups of ethical hackers and offensive trained security professionals who must stay ahead of cybercriminals' known tactics, techniques, and procedures. A Red Team's role is to imitate real-world attacks and mimic all possible cyber attackers' techniques and methods while trying to break through multiple layers of an organization's IT's defensive perimeter. Therefore, every tool and technique available to cyber attackers must be analyzed and understood by Cyber Red Teams.

The DOD Cyber Red Team shortage can be addressed through the implementation of software capabilities such as Red Team in a Box (RTIB) and Cyber Automated Red Team Tool (CARTT). Such software automates the use of baseline tools required to be performed by a Cyber Red Team on a browser-based GUI. These tools are geared toward non-expert Navy and USMC system administrators to assess their networks and mitigate vulnerabilities.

Previous NPS research has been conducted in the development of CARTT as a proof-of-concept. This concept was intended to address the shortage of red team personnel

and training, and close the gap with the vulnerability assessment required in the DON systems, networks, and facilities not directly connected to the internet to maintain their confidentiality, integrity and available. The Office of the Secretary of Defense (OSD) has sponsored research to extend the development of CARTT to enable it to access potential unknown vulnerabilities in a system by applying open-source software tools used by a Cyber Red Team to assess network securities vulnerabilities.

CARTT is not intended to completely replicate Red Teaming capabilities. In fact, there is currently no tool available to fully automate red teaming actions. Joseph Plot's thesis [2] developed a prototype called RTIB that automated a series of standard command line interface (CLI) commands allowing the user to interact with Network Mapper (Nmap), the Metasploit Framework (MSF), and Open Vulnerability Assessment System (OpenVAS) to conduct vulnerability assessments on target systems and networks. Preston Edward's follow-on thesis [3] extended the capability of RTIB, rebranded as CARTT, by adding a GUI and capability to enable an operator to automate the process of vulnerability detection, analysis, and exploitation. Joseph Berrios's thesis [4] further extended CARTT from a standalone host-based implementation to a client/server architecture allowing remote access to system operators and administrators.

This current research has redesigned and improved the CARTT graphic user interface (GUI). This GUI is designed for use by for non-experts to assist them in identifying uncovered vulnerabilities in a vast number of IT infrastructures and generate a report. The development of a web-based GUI as a system administrator's interface to do a full network vulnerabilities assessment provides the operator an easy to use and fully functional software tool that follows the Red Team operations and assessment procedures, but without requiring the user to possess these expert skills.

## **A. PROBLEM STATEMENT**

The web-based GUI was designed to be efficient and effective such that it does not require the user to be a Red Team expert in utilizing the site. Its development continues the integration of modern open-source penetration testing software tools and techniques as a

web-based interface application enabling its users to better assess an organization's security postures.

The following research question was investigated in this work:

How can a web-based graphical user interface be developed for CARTT to allow non-expert users to build, integrate, and launch exploits using Cyber Red Teams tools and techniques?

## **B. SCOPE**

This thesis has created a usable and straightforward web-based GUI for a non-expert user with just what is required to do for the vulnerability assessments, host exploitations, and generate a report with all possible vulnerabilities and recommendations for remediation. The thesis scope includes creating a web-based GUI for CARTT that should allow a non-expert to be able to effectively use Cyber Red Teams open-source tools remotely in order to increase the cyber readiness throughout the DOD network.

This thesis did not examine any specific environmental assessments, training effectiveness assessments, or accurate equipment assessments. This thesis also did not evaluate a particular vendor's website or product.

## **C. BENEFITS OF STUDY**

This study has provided the DOD with the capability to assess the cybersecurity posture effectively and efficiently without solely relying on DOD Cyber Red Teams. DOD network administrators can use CARTT as a web-based dashboard application to conduct vulnerability assessments and generate meaningful network reports with mitigation recommendations. This prototype system will improve the overall DOD cybersecurity readiness by identifying vulnerabilities during combatant command (CCMD) operations, operational test and evaluation (OT&E) of systems, and agency-specified assessments [1].

## **D. CHAPTER OUTLINE**

### **1. Chapter II: Background**

This chapter includes an overview of prior work, the different tools and techniques used by Cyber Red Teams during their operations. It focusses on the differences on Red Teams and how each teams play their roles to maintain and evaluated their network defense in-depth. There is also a review on how CARTT is currently designed.

### **2. Chapter III: Design Methodology**

This chapter describes the web-based CARTT design process requirements as a conceptual model. The chapter also evaluates and considers the user experience analysis that focuses on the characteristics that affect the design decisions and impact the performance. A task and execution analysis focuses on the environment, hierarchical structures, encoding schemes, and on a conceptual design with objectives to predict and improve users' expectations in the GUI design.

### **3. Chapter IV: Usability Testing and Evaluations**

This chapter describes the test plan, the design, and the implementation of the web-based GUI. It discusses each test's expected outcome versus the observed performance of the previous GUI and the problems encountered during the web-based GUI testing. The fundamental evaluation methods discuss the outcome in terms of vulnerability assessments, generated report overviews, and vulnerability remediation implementation. It also assesses how well the user experience fits the fidelity prototype.

### **4. Chapter V: Conclusion and Future Work**

This chapter describes the results of the prototype testing of web-based CARTT and discusses lessons learned and recommendations for future work to improve the tool.

## **II. BACKGROUND**

In the endless cybersecurity battle between offensive and defensive operations, the attackers have virtually every advantage over the security analyst. Potential threats to the confidentiality, integrity, and availability (CIA) of an organization's data, applications, critical systems, and other assets are growing more rapidly than in previous years. The attackers have virtually every advantage over the security analyst. Cyber-attackers often operate with no ethical, moral, or legal boundaries. Therefore, they can circumvent any physical and legal rules put in place by authorities.

The DOD builds its IT network systems with a layered defense-in-depth approach to protect the CIA Triad of the network and its data [5] from unauthorized access while complying with frameworks from the National Institute of Standards and Technology (NIST) [5] and the Penetration Testing Execution Standard (PTES) [6]. With the daily increase of network attacks, the Defense Information Systems Agency's cyber-defense tools blocked 300 million malicious actions against DOD computer systems in 2019 [7]. The sheer number of daily cyberattacks on the DOD networks shows the importance of providing the Combatant Commander with the personnel and cutting-edge tools necessary to expeditiously identify the various network security vulnerabilities [8], [9], [10]. Red Teaming and penetration testing are the primary cyber operations tools used to secure IT systems.

### **A. CYBERSECURITY TEAMS OPERATIONS AND TOOLS**

#### **1. Cyber Teams Roles, Responsibilities, and Expectations**

Cyber Teams are critical as they provide the opportunity to test an organization's cybersecurity defenses in order to challenge and improve these defenses against rising threats. The need for Cyber Teams is critical. According to the Mimecast's research, there was a 51% rise in ransomware from 2019 to 2020 [11]. This Mimecast research also showed that in 2020, 85% of companies surveyed reported email-based spoofing and 60% reported an increase in impersonation fraud [11].

As shown in Figure 1, there are three types of cybersecurity teams: Blue Teams, Purple Teams, and Red Teams [12].

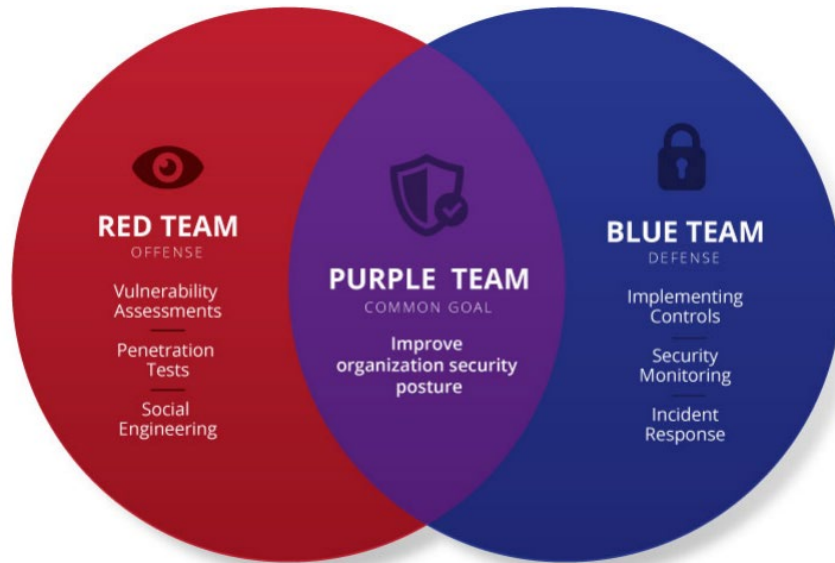


Figure 1. Comparing Red, Purple, and Blue Cybersecurity Teams.  
Source: [12]

The Department of Defense defines the Red Teams as “independent, multi-disciplinary groups ... certified, accredited, and authorized to identify vulnerabilities that impact the confidentiality, integrity, or availability” of IT systems [1]. The DOD Cyber Red Team is defined as a group of people (military, civilian, and contractors) with the mission to penetrate the defense perimeter using possible adversary TTPs against a predefined target using any means authorized by the DOD [13].

The Red Team’s main objective is to assess the DOD’s ability to execute its mission in the cyber domain by emulating its adversaries using the latest penetration tools, methodologies, and standards and then provide a report for their findings. And since zero-day vulnerabilities are unknown until they are discovered, our best “defense” is to conduct

a penetration testing of our systems continuously in order to identify and address every possible vulnerability beforehand and be in a position to mitigate discovered zero-day vulnerabilities.

A cybersecurity Red Team plays the role of a stealthy threat actor, also known as Advanced Persistent Threat (APT), but they are authorized to access a DOD organization's network to mimic an APT's tactics, techniques, and procedures (TTPs). They attempt to penetrate the network's defensive perimeter by any means necessary from the physical layer to the application layer [12]. Their main goal is to test the organization's detection and response capabilities [12].

In addition to the Cyber Red Team, there are also penetration testers (pentesters) who use a similar strategy to identify, prevent, and mitigate security vulnerabilities. The pentesters use tools that cyber defenders can detect, also known as loud techniques. Their goal is to identify gaps within the cyber ecosystem, and they don't try to hide their penetration activities [6]. In contrast, Cyber Red Teams are stealthy and avoid detection at all costs.

A Cyber Blue Team is a group of IT security staff who are tasked to defend the network and systems from any cyber-attacks launched by the Red Team or any adversary trying to penetrate the defensive perimeter [12]. A Cyber Blue Team is continuously trying to improve the organization's security posture.

A Purple Team is a group of Red and Blue Team members who collaborate to maximize organizational security defensive efforts by discovering and repairing vulnerabilities to prevent data exploits [12].

## **2. Cybersecurity Red Team Penetration Testing Methods**

A cybersecurity Red Team examines digital and physical assets within the technical and operational processes. The Red Teaming assessment provides a distinct overview of organization's IT system and network infrastructure from their perspective. Cyber Red Teams can assess the entire IT and network infrastructure or just a specific part of the network based on their assignment. Some of the Cyber Red Team testing methods are

somewhat reflective of industry standards which include physical and digital penetration testing, social engineering and phishing techniques, “threat modeling, vulnerability analysis, exploitation and post exploitation, and reporting” [6]. The Penetration Testing Execution Standard (PTES), which was created in 2009 by an ad hoc group of experts in the field of cybersecurity, provides guidelines for these techniques [6].

According to the PTES, physical penetration testing takes advantage of the loopholes within the physical layer [6]. For instance, adversary analyze shift rotation, employee habits (allowing piggybacking), surveillance cameras’ location and angles, and gate posture (which includes whether credentials are verified by gate guards or by computer systems). Digital penetration testing occurs when the Red Teams exploit the application layer through web-based applications to access a backend database [6]. Once the database’s vulnerabilities have been identified, the Red Team can launch SQL injection, cross-site scripting attacks, and impersonation attacks [6]. Information gathering is essential to any surgical attack process.

From the PTES guidelines, social engineering by email and telephony are the most common and successful schemes used by Cyber Red Teams to gain entry to an organization’s cyber network system. Phishing techniques are used to manipulate employees to reveal information to compromise their network security. The Cyber Red Team can enumerate all exploitable vulnerabilities by making active connections and directed queries to target IT systems [6]. One of the primary objectives here is to discover and hijack users’ accounts, misconfigured shared resources (e.g., unsecured files and shares), and older software versions that are known to have security vulnerabilities issues (e.g., web servers with remote buffer overflows). The primary intent is to bypass the first defensive perimeter [6].

The PTES describes how threat modeling is executed once the first entry point has been discovered; passive or active scanning assessment can further monitor network traffic and assets to exploit the system and network infrastructure [6]. The Cyber Red Team takes advantage of the network service weaknesses by exploiting primarily unpatched assets and misconfigured network assets. Network-based assessments are conducted from central locations on the network, often using agentless monitoring [6]. The hacking and threat

modeling approach gives the Cyber Red Team the ability to assess computers' vulnerability to attacks from within or outside the network and provides vulnerability information about network resources without being detected by monitoring agents, such as routers or firewalls [6].

During the vulnerability analysis, the Cyber Red Team uses agent-based monitoring tools on individual computers to conduct host-based assessments. In this approach, an agent is deployed on a host that requires a vulnerability risk assessment and it can provide a more accurate information while significantly reducing a false report (positive and negative) [6]. Another fundamental enumeration technique is banner grabbing, which involves connecting to a remote service and observing the response. Netcat and Telnet are manual tools used for grabbing banners and application information [6]. The above techniques can provide the Cyber Red Team the "make and model" of the online network services, which in many cases is enough to set vulnerability analysis in motion.

During launching and exploitation, noticeable denial effects (for instance, degradation, disruption, or destruction) or system manipulation leads to a denial of service or data exploitation [14]. The attack's main objective is to weaken the confidentiality, integrity, and availability of the system or data by making the targeted systems and networks unavailable or untrustworthy. The Joint Publication 3 12, Cyberspace Operations describes cyberspace exploitations as part of the offensive cyber operations (OCO) mission to gain intelligence, maintain systems and networks access, discover vulnerabilities, and perform other enabling actions required for current and future military operations collect information. Per national policies and international law, these actions are coordinated and deconflict with other United States Government (USG) departments and agencies [14]. During the exploitation, Cyber Red Teams want the system to execute the payloads while selecting the delivery method using the MSF or scripts created. To deliver the encoded payloads against a target, Cyber Red Teams use various built-in evasion and detection techniques.

Reporting is key to any cybersecurity assessment. The reports should be tailored to the appropriate audience. The higher leadership must understand the report output in order

for them to make a proper financial decision, to see that there is a cost-benefit on mitigation and remediation requirements to solve the identified issues [6].

### **3. DOD Cyber Red Teams' Operations and Rules of Engagement**

Red Teams are critical to the DOD network systems and structures evaluation. Only Red Teams certified by the National Security Agency (NSA) and accredited by U.S. Strategic Command (USSTRATCOM) are authorized to operate on DOD's live networks [13]. The DOD Cyber Red Teams are in a constant uphill battle against APTs, nation-state level threats, and state-sponsored groups.

The Rules of Engagement (ROE) provide explicit instructions and define what can and cannot be done by operators. The Joint Publication 3-12 states that "DOD conducts CO consistent with U.S. domestic law, applicable international law, and relevant USG and DOD policies. The laws that restrict military actions in U.S. territory also apply to cyberspace" [14]. The Cyber Red Teams must ensure that established and agreed ROE are well known and authorized by the targeted network owner before any vulnerability assessments on a specific target are conducted. The Cyber Red Teams must comply with laws and policies and take into consideration "the global nature of cyberspace and the geographic orientation of domestic and international law" [14].

### **B. CYBER AUTOMATED RED TEAM TOOL**

In their master's degree thesis research in 2019 and 2020, respectively, NPS computer science students Joseph Plot [2] and Preston Edwards [3] expanded on various types of available commercial and open-source tools for performing red teaming actions. In 2019, they developed CARTT, which provides the tools necessary to exploit undiscovered security vulnerabilities [2], [3] by automating OpenVAS and the MSF to scan, test, and discover vulnerabilities within a host. CARTT was developed to resolve the limitations associated with RTIB, developed to perform red team operations on target hosts [2], [3]. One major shortcoming of RTIB was that it had to be physically connected to the host. It was a standalone portable application run on a single device or laptop that needed to be directly connected to the target network. RTIB and CARTT were developed at the Naval Postgraduate School to address the shortage of Cyber Red Teams and augment their

operations by leveraging open-source software tools and frameworks. CARTT was the improved proof-of-concept tool that expended on RTIB.

Currently, CARTT has a command and control (C2) capability with multiple users in different roles who can directly interact with the CARTT Server [4]. The implementation of CARTT uses a client/server-based C2 model that automates vulnerability scanning and host exploitation of target networks. Each CARTT user with their specified role can perform red teaming functions not just on the network that the operator resides on but also on any remotely accessible system defined by the operator [4].

### **1. Current CARTT Roles**

The current roles within CARTT are the operator, administrator, and commander. The operator's role is to perform certain red teaming roles, such as vulnerability scanning and host exploitation [4]. All function outputs performed by the operator can be sent via a messaging system to the commander for assessment. Any message can be viewed by all assigned operators who fall under the same six-character alphanumeric code or Unit Identification Code (UIC) [4]. The commander located at the higher headquarters can task the operators to conduct network vulnerability assessments within the operator network and share information as required through the messaging system. The administrator's role is to handle account access requirements ranging from user permissions to role assignment [4].

In CARTT, the operator receives guidance from the commander before further actions are taken [4]. If the operator executes any offensive engagement without authorization, it can constitute a federal crime [15]. The ROE should provide the operators with the peace of mind that they have the appropriate approval to run any vulnerability assessments without prosecution. The CARTT C2 messaging system should be used as a means of providing authorization to the operator, and all message logs should be retained based on a predetermined timeframe.

## **2. CARTT Tools**

CARTT uses open-source software tools to be cost-effective and straightforward to deploy, manage, and maintain. The main cyber operations tools used in CARTT are MSF and OpenVAS, running on Ubuntu 20.04. Other tools and programs used to implement CARTT are the hypertext preprocessor (PHP) scripting language and My Structured Query Language (MySQL) [4].

MSF is intended to be used by an advanced security professional who has a solid understanding of penetration testing [2] implementation. MSF offers users an interactive console interface, a system command-line interface (CLI); a web interface is only available in the Professional version. MSF contains different modules that the user can interact with to employ exploits, payloads, auxiliary actions, and evasions [16], [17], [18]. Since early 2018, a new evasion module type was added to Metasploit version 5.0 so that penetration testers can attempt to bypass known antivirus products and build their own evasion modules based on Metasploit's research [17]. MSF uses a PostgreSQL database to store and access hosts, services, vulnerabilities, credentials, and user-agent strings captured by various scanning and exploit modules. MSF is packaged with real exploit modules that can result in real damage if not used properly [17].

Open Vulnerability-Assessment Scanner has many features that help Red Team operations, including simultaneous multiple host scanning, more than 50,000 network vulnerability tests, false-positive management, and scheduled scans [19]. Greenbone Networks has developed and maintained the scanner since 2009 [19].

Hypertext preprocessor is also an open-source server-side scripting language used for web development [20], based on Hypertext Markup Language (HML). It has one of the best built-in security protection layers. "PHP is used by 79% of all websites whose server-side programming languages are known [20]." Since all essential programs of CARTT are running on the CARTT server, PHP is used as the core of CARTT [4]. It is a straightforward language that can interact with MySQL database [21]. MySQL database is used in the CARTT account and messaging system to fully manage the user credentials. MySQL is the world's most popular open-source relational database management system

(RDBMS), distributed and supported by Oracle Corporation [22] and the most popular database for PHP applications [23]. PHP and MySQL are easy to learn, use, and deploy on nearly any machine due to their excellent open-source online documentation with a good framework of functions [21], [22].

### 3. CARTT Architecture Overview

CARTT users can authenticate and access their role remotely through the server with the appropriate credentials via the internet. Through the CARTT client, the operator can execute network vulnerability assessments and detect vulnerabilities within a target network, including the network in which the operator resides [4]. Figure 2 shows the different types of users and the client server relationship.

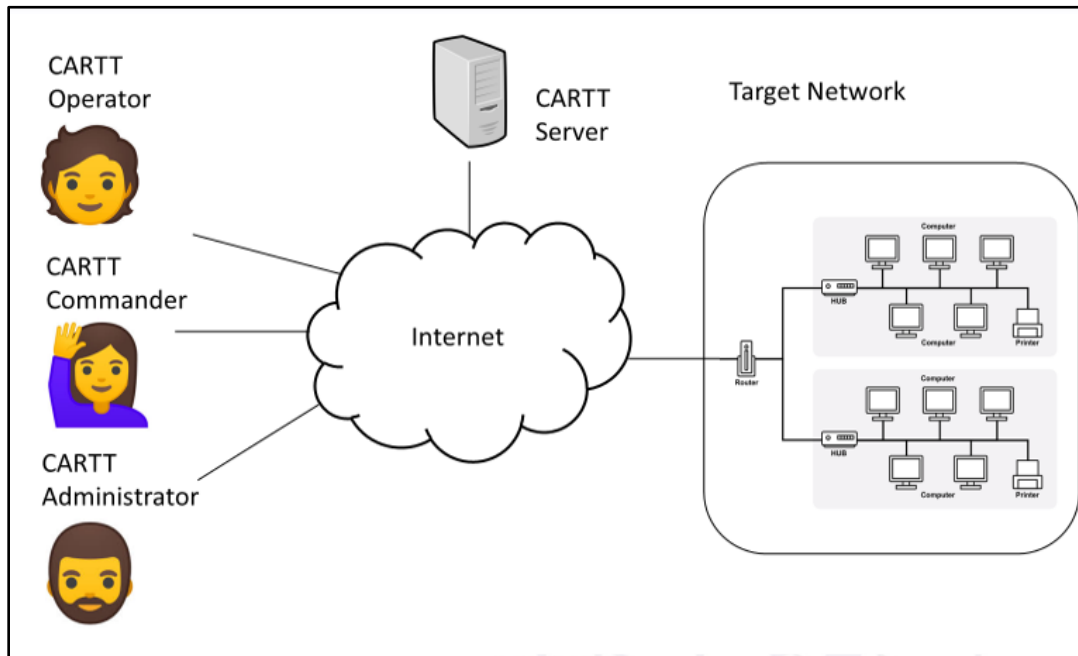


Figure 2. Sample CARTT Architecture. Source: [4].

### 4. CARTT Server overview

The clients connect to the PHP server within the CARTT Server, which runs different CARTT scripts that perform the actions of CARTT. The CARTT Scripts engage with the MSF and OpenVAS tools to do vulnerability assessment and host exploitation on

a target network. Figure 3 shows that at the operational level, the clients and CARTT Server are required to be within the target network in order to perform assessments and exploitations [4].

With OpenVAS, a report can be generated in various types of formats. The report format is selected based on the audience and the export available. OpenVAS can create pivot tables to collect statistics data or combine multiple scan reports into one [19]. A CARTT script imports the OpenVAS report into MSF in extensible markup language (XML) format, and then the vulnerabilities found will be available in the MSF PostgreSQL database for follow-on CARTT operations [17].

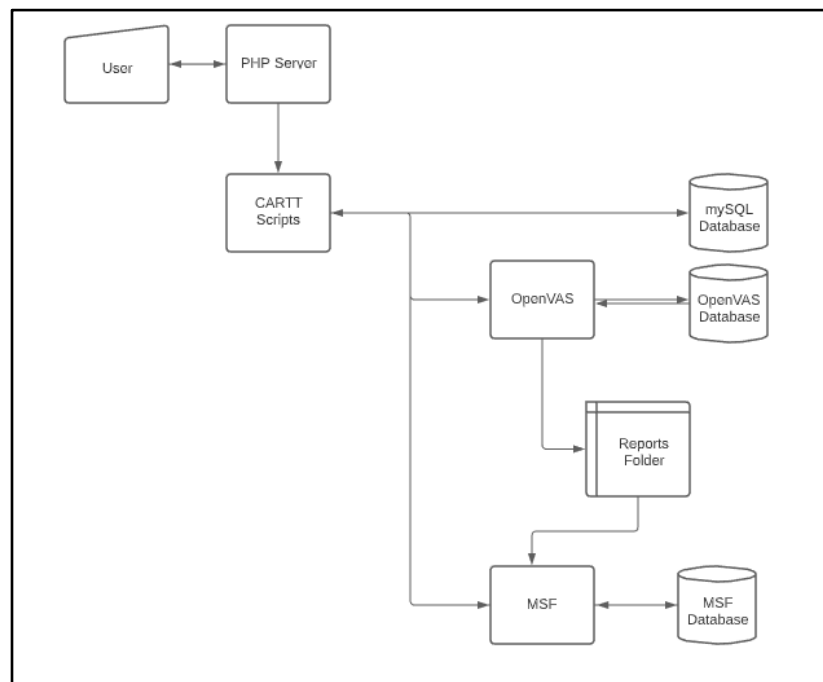


Figure 3. CARTT Server Diagram. Source: [4].

## 5. Exploitation Process-Operator

The CARTT operator's process flow is complex because most of the exploitation of a target host is done under the operator role. The operator uses the below steps to assess a target network's vulnerabilities while attempting to exploit the target host [4].

1. Configure and conduct scan through CARTT on the target network by filling in a webform to utilize OpenVAS to conduct a vulnerability scan.
2. Import the vulnerability scan into an MSF workspace.
3. Select MSF workspace with its PostgreSQL database. This step gives the operator the ability to use different workspaces with information on the targets scanned.
4. Select host, vulnerability, and MSF module for exploitation.
5. The operator will receive feedback from CARTT about the exploit status after all required selections are completed.
6. Now the operator can pass the result of the scan to the commander via the CARTT messaging system for review and await further guidance.

## **6. Exploitation Process Flow**

The operator connects to CARTT via the web browser with valid user credentials. The server validates the user's role, redirects the user to the role assigned to them (operator in this case), and displays the operator's main menu. For this exploitation process flow, the operator will get six options from the main menu. The server will point the selected options to OpenVAS. OpenVAS will then start the vulnerability scanning. The target will be getting the queries from OpenVAS, and then OpenVAS will display an analysis of the vulnerabilities. The user is provided with the possibility to view current or previous scans. The CARTT server will direct the user to an import option for the report. At this point, the operator can download the report where it can be put in a workspace. The operator can now enter the scan name to save it in a selected workspace. After the scan is completed, CARTT can now download the report from OpenVAS and redirect the operator to Metasploit. The list of hosts and vulnerabilities found in the generated workspace is now presented to the operator. Now the operator can select a host and a specific vulnerability. The operator will then get a list of the modules related to Metasploit's vulnerability that have been found. CARTT will package the module selected with the target host and target vulnerability,

attempt to exploit the target, receive feedback, and display the information back to the user [4].

In the current design of CARTT, the operator is still required to know the basic functions of red teaming. Training is required to teach them the basic CARTT functionality and how to navigate through the necessary functions on the main menu after the operator has authenticated through the server. The operator and commander are not required to know how the underlying CARTT tools, such as OpenVAS and MSF, interact. The administrator, however, needs to know how to assign the appropriate roles and permissions and have a basic understanding of the tools running under CARTT. The commander is required to understand the C2 messaging system functionality after login since this will allow the commander to receive updates and assign tasks to the operator [4].

## **7. User Operations Overview**

The user accesses CARTT by logging into CARTT via a client/browser interface that tracks the user's assigned role before any scan of the targeted network is possible. After login in, the user is required to have at a minimum the following information to conduct a scan: target IP range and create a name for the scan before performing a network vulnerability assessment. Once the results are provided to the user via the CARTT client from the scan, the user can select the host(s), the vulnerabilities, and payload. The user can now run the selected desired vulnerability and exploited modules on the host(s) from the client's information. The user has to balance the possible operation impact with the client scan information and the user's desired effect before making the appropriate selections. The user will receive a result from the client after the exploit is run. CARTT allows the user to run multiple vulnerabilities testing by selecting different options for the host, vulnerability, and module [4].

## **8. User Experience Design Considerations**

Usability is a critical part of the UX because it is how users will access the functionality within the application to meet their satisfaction [24]. The basic concept of UX design is to observe the user trying the interface and record where they run into problems so that the designer can go back and make the necessary correction to improve

the UX [24], [25]. The usability goal is to increase the effectiveness, efficiency, memorability, and satisfaction while decreasing the number of errors [25]. Because of CARTT's intended audience, effectiveness and error anticipation should be the top priorities in the interface design because the user is untrained in red teaming.

***a. Effectiveness***

In 2001, Whitney Quesenbery, the UX and usability expert defines effectiveness as whether the users can accomplish their task with a high degree of accuracy [24]. Therefore, the application's effectiveness can be measured by how easily a user can learn the system the first time without making mistakes. The quicker the user learns to navigate through the application, the more meaningful the tasks are. A few factors can improve effectiveness [24]. First, the designer should be familiar with the existing internet metaphors and use them in the design, it helps quickly familiarize the user with the website. Second, the consistency helps with the familiarity of the application, which in turn helps with the usability. The designer should make the application predictable and straightforward for the users. For instance, a search and help button should have a different function because they mean something different.

***b. Memorability***

Designers should create the application so that it is easy to recognize by making the task incredibly clear to become more comfortable the second time for the users. As human, we tend to forget the tasks or steps that are too complex [26]. A couple of examples to increase the user memorability are providing the user with something that they can visually relate to because we easily relate to visual icons. Consistency also plays a role in improving memorability.

***c. Satisfaction***

Users should not be leading the GUI application, but the other way around, so that the users get the result the designer expected, which provides user satisfaction. While navigating through the application, the user experience should be pleasurable, the designer should create an attractive design, with colors, fonts, sound, and visual cues. The designer

should create an application that allows the users to search and select the first rational option, a strategy known as satisficing, a combination of the word “satisfy” and “suffice” [27]. According to Simon, humans are not always looking for the best possible solutions to their problems. They operate in “bounded rationality” where time, environment, language, limitation in their acquisition of knowledge, awareness reasoning, and control over the outcome play a significant role in their decision making [24].

#### ***d. Errors***

Basic errors, like selecting the wrong control will occur because the CARTT audience are expected to be non-expert in Red Teaming. Designers should expect users to make mistakes. Therefore, focusing on handling the errors and helping the users to minimize errors and by solving the issue on their own is critical [24]. Any programming design is susceptible to error because of the ecosystems in which the program resides [28]. Error tolerance should be built in the design, which creates an environment where the users can correct their errors with the “redo” option. This option gives the users a sense of control and build their confidence, trust, and empathy.

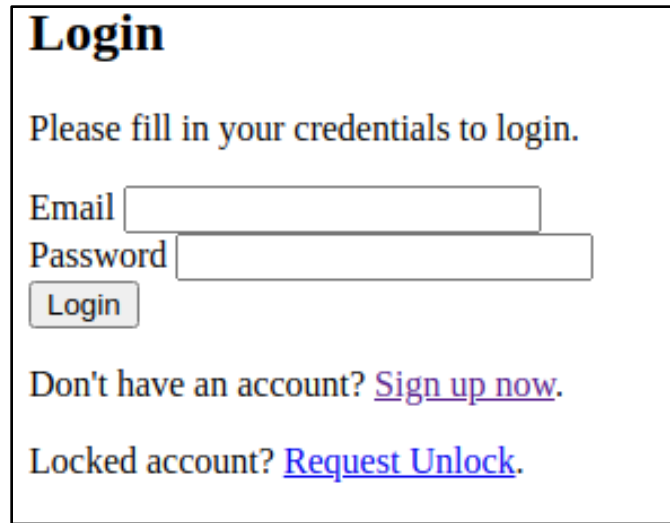
### **C. USER INTERFACE ACCESS CONTROLS AND SECURITY**

The end-user has always been the weakest link in the security of any system. The software developer and web developer must find a balance between usability and security. Web access control restricts access to application resources. A few basic security principles must be incorporated during the design process to preserve the data’s confidentiality, accessibility, and availability. User interface security is far more than just having an effective policy for passwords and encryption. It is critical to design security into the interface and it must be made visible and apparent to the user to meet security expectations and the minimum industry standards. Below are the most common secure/usability issues.

#### **1. Authentication and Authorization**

For a web application, authentication and authorization are essential for security control to ensure access to authorized users only. Authentication ensures that the users’ credentials and identities are validated as a prerequisite before authorizing them to access

information systems [29]. Username and password are a simple but highly effective form of single-factor authentication for CARTT. Short length of password is easy to hack and lengthy password are difficult to memorized. The login page will have to follow the current industry standards set by the DOD [30]. In the next chapter we will present the entire UI. Figure 4 shows the login page for the CARTT user.



**Login**

Please fill in your credentials to login.

Email

Password

Login

Don't have an account? [Sign up now.](#)

Locked account? [Request Unlock.](#)

Figure 4. CARTT Previous Login Page. Source: [4].

Access controls, also known as authorization, are required to reassure the users that their data is safe [29]. Authorization is essential to secure the access and control of the users' traffic and assure the user the data confidentiality in each application system [31]. According to the 2019 statistics report from Positive Technologies, every one out of three application had broken authentication, which led to unauthorized disclosure, modification, or data destruction [32]. In CARTT, the operator should not be granted the role of Commander or Administrator and allowed access to those roles' permissions.

In Role-Based Access Control (RBAC), the organization creates and enforces access authorization based on roles and responsibilities of each user, the organizational structure and the organization's security policy requirements [33]. It is the best access control method to implement in CARTT to restrict network access. This method ensures that only the information users required to access is presented to them.

Only authorized personnel should be granted access to CARTT applications without authorization. Placing users in their appropriate roles also determines what resources they are authorized to access. Authentication and authorization provide a security layer to applications and ensure that only verified and validated users explicitly access certain information and resources. There is currently no forgotten password option, as shown in Figure 4. This option should be given to the user to reset the password without going through the administrator. This option will provide the user the opportunity to reset their password or username by providing the email address that was used to register the account.

## **2. Security**

Securing a web application from possible threats will prevent things like leaking confidential information and breaking the application system database [34]. The National Institute of Standards and Technology (NIST) recommends that web applications move to multifactor authentication [5]. A report released by Positive Technologies indicates that the most common web application vulnerability is security misconfiguration [32]. The report details that security misconfigurations were found in 84 percent of web applications.

Cross-Site Scripting (XSS) is used to execute a script to capture the user's session identifier (Session Hijacking), pushing malicious software, bypassing restrictions, impersonate the application's user to steal authentication cookies, and a Denial of Service (DOS) attack in order to make IT systems and devices inaccessible [31]. According to NIST, the safest way to prevent XSS is to validate and filter all data properly before they are sent to the database.

Broken authentication and authorization were found in 45 percent of web applications [32]. The sessions control authentication and authorization. With XSS, hackers can steal the users' credentials or try to access the application. An attacker can exploit XSS to brute force the credentials, accesses the web application, and escalates the application access privilege. Using CARTT login should prevent session attacks or manipulations. A session timeout or a logout after a period should be implemented. The HTTP (Hypertext Transfer Protocol) should contain no stored information to prevent

searching into the cache information and session identity numbers should not be stored where the cache is required within the web browser [31].

SQL injection were found in 29 percent of web applications [32]. It is an injection by an attacker of potential harmful SQL queries and executing a command to the server to access the backend database in order to manipulate the data by deleting, inserting, and updating the data. To prevent SQL injection and the risk of manipulating SQL statements, the web application developer must ensure that all the client side's input is filtered before entering the database. The validation of users' input will prevent SQL injection on the database [31].

### **3. Testing**

Testing is vital for any web design. User interface (UI) testing and functionality testing are especially vital for a web application [35]. UI testing is a problem-solving approach that improves navigation, layout, browser compatibility, and content. User interface testing focuses on the overall appearance and navigation flow from the user perspective across multiple experimental studies. It is also an opportunity to learn more about the application's users and improve the application system. Functionality testing helps to test the logic and flow behind the application. It is also part of the quality assurance. The list of testing required to ensure that the web application is working correctly to the required specifics will be discussed in detail in the Testing Chapter. CARTT will have to be thoroughly tested to ensure it only does what it claims to do, and nothing more, before it is used on any DOD system or network.

## **D. CHAPTER SUMMARY**

This chapter covered the importance of DOD Cyber Red Teams. It discussed notable red team tools, roles and responsibilities, rules of engagement, and products, including an overview of current NPS work in developing CARTT. It also analyzed how the UI design and the security implementation process can help create a usable and functional CARTT application capable of closing the gap with the vulnerability assessment requirements of DOD systems, networks, and facilities. In the next chapter, we discuss the CARTT GUI prototype implementation.

THIS PAGE INTENTIONALLY LEFT BLANK

### **III. DESIGN METHODOLOGY**

This chapter focuses on the web design methodology used to develop CARTT's UI. The objective was to provide a pleasant aesthetic user experience, communicate the intended functions, and facilitate ease of access to the content presented. Below is the design methodology used to develop the CARTT web-based GUI design by analyzing the needs and wants of each end user.

#### **A. VISUAL DESIGN**

Visual design is used to make the data and the user interaction with the content of the application clear and presentable. Visual design helps to draw the user's eye and focus on the correct message and functionality. Nielsen's ten usability heuristics guidelines elements of quality attributes of design have been used to increase customer experience and clearly communicate the messages behind the object.

#### **B. PROTOTYPE**

The exploration of the problem space has been determined in Chapter 2 and the work done by previous students to initiate envision of the system. The Design methodology explored the solutions available and analyze the UI for the system design and the user's needs.

#### **C. CARTT UI OVERVIEW**

CARTT provides a client/browser interface that allows local network defenders to perform Red Team vulnerability assessment on their own networks. The UI is automated to interact with MSF, OpenVAS, and other penetration testing tools. CARTT is designed to run on a single server that runs the authentication credentials and processes the applications needed to perform basic red team assessment. As stated previously, CARTT users can log in with three profiles: operator, commander, or administrator.

#### **D. NEEDS ANALYSIS/MISSION REQUIREMENTS**

The CARTT needs analysis outlines the requirements of the web design, as stated in Chapter II. In summary, the main objective is to provide the local DOD network defenders

with an automated tool to conduct a red team assessment within their network. CARTT performs the basic red team functions to help increase the cybersecurity readiness of the DOD's networks without the need for the user to go through expensive and time-consuming training required by red teams.

A CARTT operator can scan a target host or network for vulnerabilities. At the conclusion of the scan, the operator can view the list of target vulnerabilities from the OpenVAS vulnerability scan via MSF, and then view a list of applicable MSF exploit modules by selecting a host and vulnerability. The user should feel that they are in control and that the process flow is spontaneous. The feedback received from the selected exploit module by MSF provides the relevant information to the operator. It is crucial that this feedback be in plain language that is easy for non-expert red team users to understand.

If the user is not satisfied with the tool's results, he or she might stop using the application. Therefore, the user should have visibility of the network topology with the number of internet protocols used within a subnet. The exploitation possibilities associated with the modules should be critical to the user because it provides them with relevant data on the vulnerabilities present within the network. The success of the mapping of the network, view of the Common Vulnerabilities and Exposures (CVE), exploitation, and messaging systems are used as acceptance criteria in the previous CARTT web design [4].

## **E. USER ANALYSIS**

The user analysis focuses on enumerating the attributes of CARTT users to identify their respective needs that must be addressed by the UI. The user analysis identifies what impact design decisions have on performance so that we have a baseline for measuring performance against actual user types.

CARTT is designed for three primary user roles:

Commander: The network operation manager has the role of assigning tasks to the subordinate unit. The Commander is not required to perform any red team assessments. The Commander is a user who is at higher headquarters (e.g., Tier I DISA, Tier II MCCOG(CSSP), and Tier III MITSC/RNOSC), and is responsible for the operations and the

maintenance functions of an enterprise system, equipment, or infrastructure throughout their respective area of responsibility. The Commander maintains authority for Active Directory (AD) Organizational Unit (OU) structure and permission delegation, and to administer objects and their containers at the enterprise level. The AD OU comprises critical elements that provide effective management and security of systems. The Commander has the authority to authorize or delegate a vulnerability assessment at the lowest level to a Network Operator.

Network Operator: In CARTT, the operator is expected to perform the basic red team tasks, ranging from network configuring to conducting a vulnerability scan of the network through a web-UI. The permissions are delegated by the appropriate regional network operation manager at a tier level to support the mission requirements. The network operator operates and maintains the servers performing enterprise level service. The UI was designed to support the network operator as the primary user of CARTT.

Network Operation Administrator: The administrator role in CARTT is to manage and maintain user accounts and access privileges to the network and its programs, processes, and applications. They are accountable and responsible for functions related to the specific enterprise directory, messaging, file sharing, domain controller, Dynamic Host Configuration Protocol (DHCP), and Domain Name System (DNS) functions. Network administrators receive the training to install, operate, secure, and maintain DOD computer networks and database. The network operation administrator's skills require the administrator to be familiar with routing protocols, Virtual Private Networks (VPN), Internet Protocol Security (IPSEC), subletting, traffic monitoring and cybersecurity. We expect network operation administrator users to have the basic knowledge outlined in the DOD Directive 8140, which establishes the required training, certification, and management of all DOD employees who conduct cybersecurity functions.

## **F. PREVIOUS CARTT UI WALKTHROUGH**

The following is a description of the CARTT UI as it appeared prior to the work done by Joseph Berrios in his thesis [4]. Each of the user roles executes different tasks and their access privileges are based on their login. Figure 5 represents the operator main page with six action options:

The screenshot shows a web browser window with the address bar displaying '127.0.0.1:4567/ops\_frontpage.php'. The page title is 'Welcome to CARTT'. Below the title is the section 'Operator Main Menu'. A message states: 'Please choose an option below to be taken to the corresponding interface.' There are six radio button options: 'Configure and Conduct a Scan', 'Check the Status of a Previously Started Scan', 'Start a Previously Configured Scan', 'Import a Completed Scan', 'Begin Exploit Process', and 'View/Send Messages'. A 'Submit' button is located below the options.

Figure 5. Previous CARTT Operator Main Page

The “Configuration and Conduct a Scan” selection provides the operator the opportunity to conduct a scan within a specific IP address range, and to provide a name for the scan as shown in Figure 6. After the required fields are completed with the appropriate inputs, the operator can click “Submit” to conduct a scan.

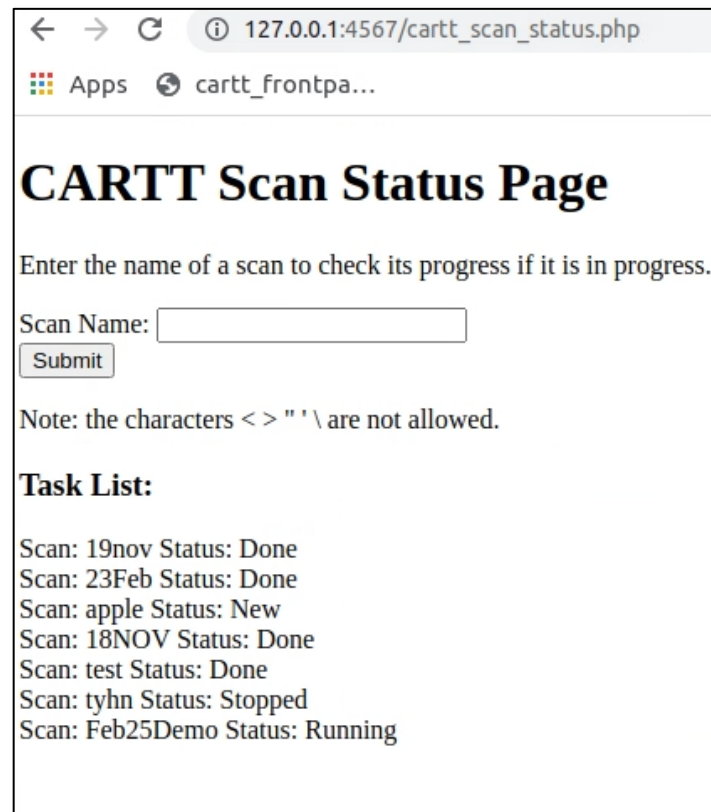
The screenshot shows a web form titled 'CARTT Scan Page'. A message states: 'Enter and IP Address and Subnet for the target network as well as a unique name for the scan.' There are three input fields: 'IP Address:', 'Subnet:', and 'Scan Name:'. A 'Submit' button is located below the fields. A note at the bottom states: 'Note: the characters < > " ' \ are not allowed.'

Figure 6. Previous CARTT Operator Scan Page

Upon submission of the input inquiry, CARTT validates the IP address within the provided subnet. The submission starts MSF and loads OpenVAS into MSF. CARTT sends OpenVAS the query parameters and starts the network scan. A visual status is generated to provide the user with a percentage of completion. When the requested scan completes,

CARTT will redirect the operator to the main welcome page, where they can import the scan results using “Import a Completed Scan” (see Figure 5). This program sends the scan to MSF to begin target exploitation. Closing the browser UI does not stop the scanning process, which provides the user the flexibility to attend to other tasks.

Figure 7 shows the status of previously started and completed scans. The operator has the option to view the status of previous scans.



← → ↻ ⓘ 127.0.0.1:4567/cartt\_scan\_status.php

Apps cartt\_frontpa...

## CARTT Scan Status Page

Enter the name of a scan to check its progress if it is in progress.

Scan Name:

Note: the characters < > " ' \ are not allowed.

**Task List:**

- Scan: 19nov Status: Done
- Scan: 23Feb Status: Done
- Scan: apple Status: New
- Scan: 18NOV Status: Done
- Scan: test Status: Done
- Scan: tyhn Status: Stopped
- Scan: Feb25Demo Status: Running

Figure 7. Previous CARTT Scan Status Page

Figure 8 shows the option for the user to import a scan into a workspace by providing the operator with a list of previously completed OpenVAS vulnerability reports.

**Report List:**

Report: 17nov  
Report: tyhn  
Report: 13oct0003

## CARTT Scan Import Page

Enter the OpenVAS Report and desired MSF workspace in order to import the scan.

Report:

Workspace:

Note: the characters < > " ' \ are not allowed.

Figure 8. Previous CARTT Scan Import Page

After submission of the import request, CARTT sends a request to MSF for OpenVAS to download the vulnerability report submitted as an XML file. In turn, MSF ingests that XML file into a workspace named by the operator.

Upon completion of the scan import, the operator will be redirected to the CARTT Exploitation System page as shown in Figure 9, where the provided name for the MSF workspace will auto-populate in the desired workspace field.

## CARTT Exploitation System

### Workspace Selection

Enter the desired MSF workspace. The workspace selected must have previously had a vulnerability report imported.

Desired Workspace:

Note: the characters < > " ' \ are not allowed.

Figure 9. Previous CARTT Exploitation System

After submission, CARTT displays the list of all the hosts detected and all vulnerabilities from Metasploit within the named workspace of the scan requested. The operator now has the option to pick a specific host from the displayed list for vulnerability exploitation and evaluation. As shown in Figure 10, the operator enters the selected host IP address in the Target Host IP Address field then clicks the “Submit Query” button.

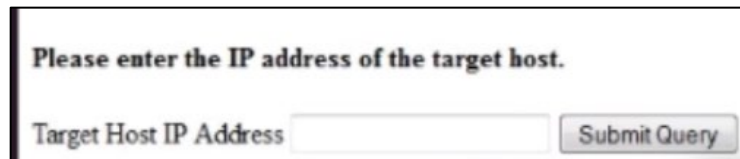
A screenshot of a web form titled "Please enter the IP address of the target host." Below the title is a text input field labeled "Target Host IP Address" and a "Submit Query" button.

Figure 10. Previous Target Host IP Address Query

After submission of the query, the operator is presented with all the vulnerabilities associated with the target host. The operator now can choose the desired Common Vulnerabilities and Exposures (CVE) number from the list presented for exploitation, and then copy it in the provided field as shown in Figure 11.

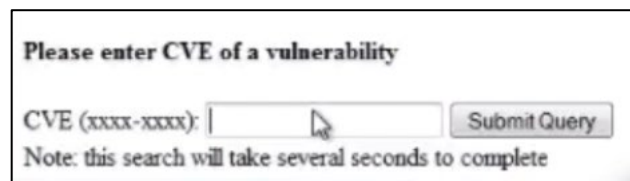
A screenshot of a web form titled "Please enter CVE of a vulnerability". Below the title is a text input field labeled "CVE (xxxx-xxxx):" with a mouse cursor pointing at it, and a "Submit Query" button. Below the input field is a note: "Note: this search will take several seconds to complete".

Figure 11. Common Vulnerabilities and Exposures Query

After submission of the query, MSF opens and searches for any available exploit module for the submitted CVE. As shown in Figure 12 the operator must copy and paste the desired MSF exploit module to construct an exploit.

Copy and paste a module into the text field to construct an exploit

Module:

Figure 12. Modules Query

After submission of the query, CARTT creates a script to execute the exploit. CARTT configures the exploit, set the target host, and attempt to launch the exploit. As shown in Figure 13, the operator receives feedback from MSF upon completion of the submitted query. The feedback provides the operator with the existence of a capability to gain a foothold in the target host. As the user is not fully trained on red teaming, the expectation for the operator is to have a clear indication of the result that the exploit worked as shown in Figure 13, which indicates that an MSF exploit session was opened.

```

127.0.0.1:4567/cartt_exploit x +
127.0.0.1:4567/cartt_exploit_test.php
Apps cartt_frontpa...

CARTT Exploitation Test Page

Exploit configured
Attempting to use exploit/windows/smb/ms17_010_psexec on host 125.125.125.130

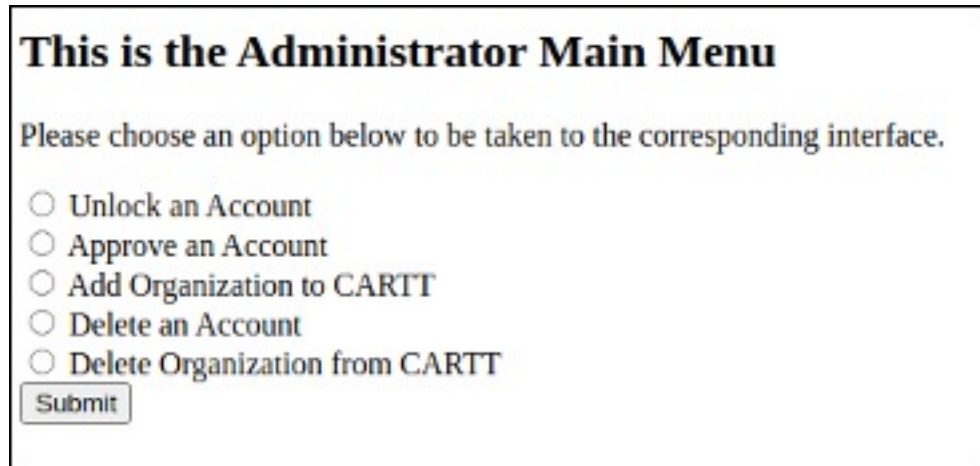
Attempting Exploit.....

Results:
[*] Spooling to file user_data/results_qwert.txt...
resource (user_data/exploit_qwert.rc)> exploit -z
[*] Started reverse TCP handler on 125.125.125.128:4444
[*] 125.125.125.130:445 - Target OS: Windows 5.1
[*] 125.125.125.130:445 - Filling barrel with fish... done
[*] 125.125.125.130:445 - <-----| Entering Danger Zone | ----->
[*] 125.125.125.130:445 - [*] Preparing dynamite...
[*] 125.125.125.130:445 - [*] Trying stick 1 (x86)...Boom!
[*] 125.125.125.130:445 - [+] Successfully Leaked Transaction!
[*] 125.125.125.130:445 - [+] Successfully caught Fish-in-a-barrel
[*] 125.125.125.130:445 - <-----| Leaving Danger Zone | ----->
[*] 125.125.125.130:445 - Reading from CONNECTION struct at: 0x81756ce0
[*] 125.125.125.130:445 - Built a write-what-where primitive...
[+] 125.125.125.130:445 - Overwrite complete... SYSTEM session obtained!
[*] 125.125.125.130:445 - Selecting native target
[*] 125.125.125.130:445 - Uploading payload... GDCEBKDKs.exe
[*] 125.125.125.130:445 - Created \GDCEBKDKs.exe...
[+] 125.125.125.130:445 - Service started successfully...
[*] 125.125.125.130:445 - Deleting \GDCEBKDKs.exe...
[*] Sending stage (175174 bytes) to 125.125.125.130
[*] Meterpreter session 1 opened (125.125.125.128:4444 -> 125.125.125.130:1148) at 2021-02-25 17:02:07 -0800
[*] Session 1 created in the background.
resource (user_data/exploit_qwert.rc)> exit -y

```

Figure 13. CARTT Exploitation Test Page

Figure 14 shows the main page for the administrator with their roles to manage CARTT user accounts and organizations. The executable tasks are displayed in the labeled radio buttons with text fields showing the options available for the administrator.



The screenshot shows a web interface titled "This is the Administrator Main Menu". Below the title is a prompt: "Please choose an option below to be taken to the corresponding interface." There are five radio button options listed vertically: "Unlock an Account", "Approve an Account", "Add Organization to CARTT", "Delete an Account", and "Delete Organization from CARTT". At the bottom left of the form is a "Submit" button.

Figure 14. Previous CARTT Administrator Menu

In the original CARTT implementation, there is no visualization used to make the data and the content of the application or site clear and presentable to its users. Cascading style sheets (CSS) were not implemented in the structure, nor in graphical user interface design. Therefore, the content was not made to be easily understandable by UI designer. There were no colors used in the original CARTT, as the only colors were black and white, whereas human minds react more positively to colors. There was a lack of structured pages and content boxes with a symmetric design suitable for traditional users. Also, there was no password security reinforcement during the user account setup. On the previous design, password security did not comply with NIST Special Publication 800–63B rev. 3 [36]. The main objective of the original CARTT was the functionality rather than the UI, and this new design tried to put the user’s needs and requirements while maintaining and improving functionality.

## **G. USER INTERFACE DESIGN GUIDELINES**

In user interface design, we are concerned about how the interface behaves under specific user scenarios, not how the software is designed or constructed. Design guidelines must be followed to make sure that the user is not frustrated and to improve the UX satisfaction.

To address the UX we should consider and apply Jakob Nielsen and Rolf Molich's ten general principles for interaction design heuristics guidelines [3], [37], [38], [39], [40]. These guidelines were used to design CARTT's graphical user interface:

1. "Visibility of system status": Nielsen recommends that users should always be aware of what is going on and have visibility of the system's status through appropriate feedback and within a reasonable time [37], [39]. The UI should be clear and presentable to the user by making the content of the data readable and easy to understand. The visualization tells the story by guiding the user to a series of tasks and indicating what possible options are available to them. Data visualizations help communicate the relationships between different statistical data, relationships in a data set, users, and designers [37], [38]. In CARTT, the focus was on computer programs for desktop devices, and no input was collected on website efficiency and usability from the users' perspective.
2. Matching the system to the real world: According to Nielsen, designers "should speak the users' language. Use words, phrases, and concepts familiar to the user, rather than internal jargon. Follow real-world conventions, making information appear in a natural and logical order" [39]. The design should use useful recognizable metaphors and language used in the DOD wherever possible. The information should be presented to the user in a logical manner. The end-state is to reduce the cognitive burden and make the systems more comfortable to use [3].
3. Providing "control and freedom" to the user: Nielsen advises designers to give users "a clearly marked 'emergency exit' to leave the unwanted

action without having to go through an extended process” [39]. The users should feel they are in control by giving them the initiative so that they have the option to undo and redo previous tasks [37].

4. Ensuring “consistency and standards”: Nielsen suggests that the “users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform and industry conventions” [39]. Inconsistency brings friction and makes tasks complex. The reverse is that consistency brings clarity. The designer should always use the same patterns in language, imagery, color scheme, borders, font, size, and symbols in the same way to facilitate efficiency [30], [37]. Consistency helps with user memorability [41].
5. Preventing errors: Nielsen warns that “Good error messages are important, but the best designs carefully prevent problems from occurring in the first place. Either eliminate error-prone conditions, or check for them and present users with a confirmation option before they commit to the action” [39]. Error minimization while designing the application is essential to minimize the user’s frustrations. When errors occur, the user should be able to identify them quickly, and the adverse impact must be small [37]. Saving user information for a short period and an autofill option are useful to minimize user errors [41], [37]. For the user, run-time errors are the most significant, compile-time errors are less significant. The UI designer should plan on testing and anticipate the possible types of errors that a user might encounter [37].
6. Focusing on “recognition rather than recall”: Nielsen advocates that the designer “minimize the user’s memory load by making elements, actions, and options visible. The user should not have to remember information from one part of the interface to another. Information required to use the design (e.g., field labels or menu items) should be visible or easily retrievable when needed” [39]. The designers should not force the user to remember commands and actions; they should help them choose. The

application should help minimize the cognitive load, effort exerted, or required during their reasoning and thinking [37]. Our attention span has gradually declined over the years, from 20 seconds to 8 seconds in 2020 [42]. The psychologist George Miller, put forward the idea that people can store “between five and nine items in short-term memory” [26] and which he called “the magic number 7.” Our memory, in general, can hold plus or minus seven numbers. Therefore, using familiar terms from other sites and keeping digits memorization to a minimum will help the user recognize the tasks, minimize the learning process, and memorize the number, which will ultimately help the user to learn the web CARTT process faster [41].

7. Applying “flexibility and efficiency of use”: Nielson recommends the use of “Shortcuts — hidden from novice users — may speed up the interaction for the expert user such that the design can cater to both inexperienced and experienced users. Allow users to tailor frequent actions” [39]. The designer should balance learnability with efficiency. The system should be usable by learners with the intent to make them efficient experts [37]. Efficiency measures how quickly a task is completed after the learning phase [43].
8. Applying “aesthetics and minimalist design”: Nielson mentions that the “interfaces should not contain information which is irrelevant or rarely needed. Every extra unit of information in an interface competes with the relevant units of information and diminishes their relative visibility” [39]. The designer should keep clutter to a minimum. The human attention span and the short memory must be considered because all information competes for the user’s attention [26]. You should only present what is needed or required to do the task, minimizing complexity by keeping the cognitive load low. The designer should remove all unnecessary functions and information by using progressive disclosure and breaking down complex tasks into comprehensible steps. They should prioritize the views

of the task to only what is required. These processes help the users to focus on the relevant information [37].

9. Helping users recognize, diagnose, and recover from errors: Nielson advocate to the designer that “error messages should be expressed in plain language (no error codes), precisely indicate the problem, and constructively suggest a solution” [39]. Web CARTT should be designed to use simple language to bring a user back to a known state, something that is familiar so that the user can recovered from slips or mistakes. Web CARTT should provide a messaging feedback system that suggests a solution to help the users identify, diagnose, and recover from errors independently [37].
10. Helping and documentation: Nielson supports the idea that “it’s best if the system doesn’t need any additional explanation. However, it may be necessary to provide documentation to help users understand how to complete their tasks” [39]. Help menus and documentation provide an architectural overview of the web application [41]. Each help menu should be tailored to the specific task, and they should be easily understood by a non-technical user. They should provide the content categories and features of the applications and websites. The designer should create “Help menus” with the intent that the user would only read this as a last result in order to interact with the system and the process must be usable to all types of users [37].

Once we identified who the users are in each assigned role as described in Section B.1, we can apply these guidelines to increase the user experience (UX). The Interaction Design Foundation defines the UI design as the “process where the designers use to build interfaces in software” applications and interact with the users guidelines above [37]. Currently, CARTT was not designed primarily with the UX in mind, focusing on looks or style. The main focus was on software functionality.

## H. CARTT'S NEW DESIGN

As previously stated, CARTT has been designed with three user categories in mind: operator, commander, and administrator. Each category is given the option to login, and their login credential is associated with their role on the CARTT server. This new login process provides simplicity for the user, with options to sign up for a new account, log out, and request to unlock an account. The new design used human-readable string values using a hash table versus forcing the users to remember numbers that correspond to specific roles as shown in Figure 15.

```
$role_map["no_role"] = 0;
$role_map["ops"] = 1;
$role_map["cmdr"] = 2;
$role_map["ops + cmdr"] = 3;
$role_map["admin"] = 4;

$selected_role = $role["no_role"];

$email = $_POST["email"];
$role = $role_map[$_POST["role"]];

$sql_update = "UPDATE users SET role=$role WHERE email = '$email'";
```

Figure 15. Corresponding Role Numbers

Figure 16 shows how the user roles are mapped to their corresponding values defined in the database.

```

<div class="form-group <?php echo (!empty($role_err)) ? 'has-error' : ''; ?>">
<p class="select_role_text">Select role from dropdown list</p>
  <select name="role" class="select_role">
    <option value="admin">Administrator</option>
    <option value="ops">Operator</option>
    <option value="cmdr">Commander</option>
  </select>
  <span class="help-block"><?php echo $role_err; ?></span>
</div>

```

Figure 16. Human Readable Mapping

Figure 17 shows the security implementation code to reinforce p criteria during the account setup.

```

<div class="form-group <?php echo (!empty($password_err)) ? 'has-error' : ''; ?>">
  <input type="password" name="password"
  class="form-control" placeholder="Password"
  pattern="(?!.*\d)(?!.*[a-z])(?!.*[A-Z]).{8,}"
  title="Must contain at least one number, one uppercase, lowercase letter, 8 or more characters"
  value="<?php echo $password; ?>">
  <span class="help-block"><?php echo $password_err; ?>
</span>
</div>

```

Figure 17. Password Input Control

## 1. Login Page

The new login screen, shown in Figure 18, is easy to understand and does not require the user to think too much. The login page itself is quite simple, with the form taking the center of the screen. Users can login with an approved email address and password created during previous registration. The page offers a bit of usability-friendly guidance to help reduce the possibility of invalid input. Another type of validation will appear during a failed login attempt and specifies what went wrong. All validations are connected to warning colors and other visual cues.

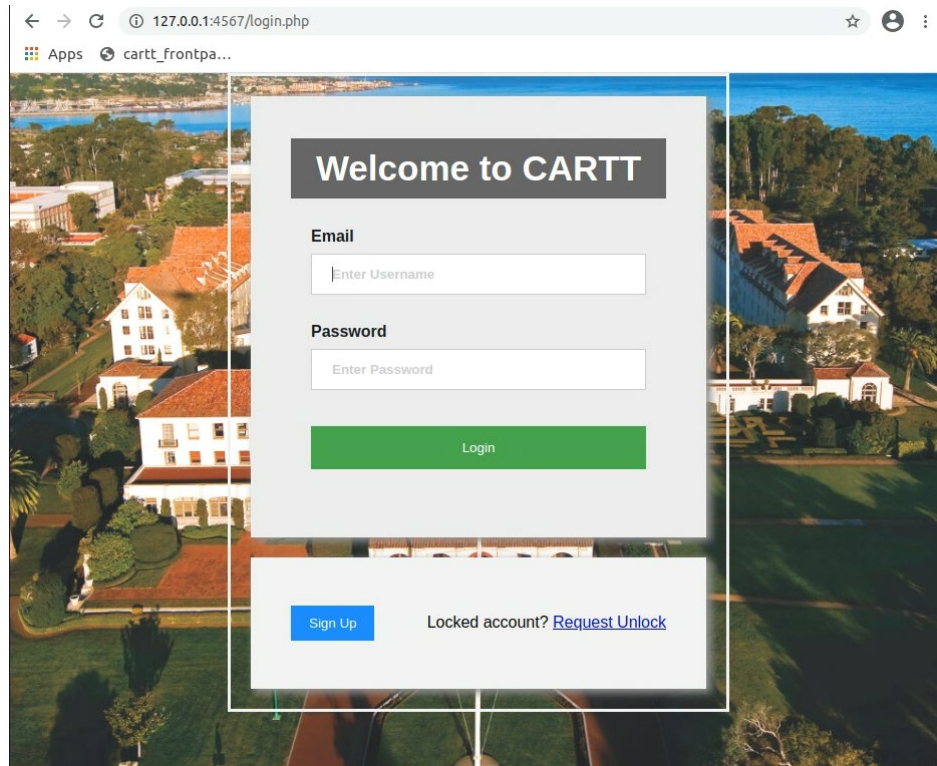
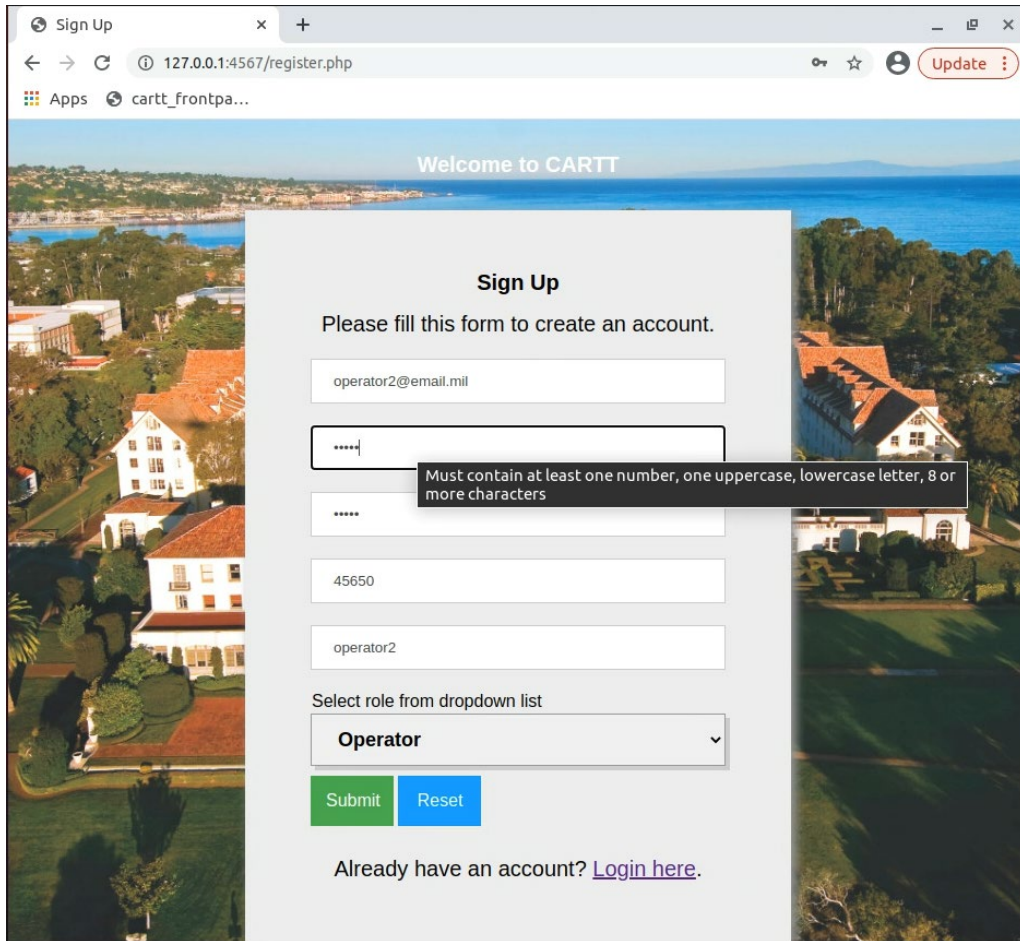


Figure 18. New CARTT Login Page

The registration page uses the same framework as the login page so that the user maintains familiarity with the page shown in Figure 18. This helped to increase memorability and simplified the creation of a PHP webpage. The registration email requires a unique email address that is also used as a unique digital identifier since the DOD email is unique to each employee. The password required during the account creation must be a minimum of 8 characters, and must contain at least one number, one uppercase letter, and one lowercase letter. The pop-up dialog box shown in Figure 19 provides feedback to the user to make sure their input conforms to the password requirements set for each entry field. The Unit Identification Code (UIC) provides a layer of security during the account request sign up because each user is assigned to a command's UIC. During the sign-up process, the user has the option to use the dropdown list which allows role selection via a collapsible list (Operator, Administrator, and Commander). This is an efficient form intended to reduce user error during the input entry.



The screenshot shows a web browser window with the title "Sign Up" and the URL "127.0.0.1:4567/register.php". The page features a background image of a coastal town and a central registration form. The form is titled "Sign Up" and includes the instruction "Please fill this form to create an account." The form fields are: email (operator2@email.mil), password (masked with asterisks), password confirmation (masked with asterisks), phone number (45650), username (operator2), and a role dropdown menu (Operator). A tooltip message states: "Must contain at least one number, one uppercase, lowercase letter, 8 or more characters". Below the form are "Submit" and "Reset" buttons. At the bottom, there is a link: "Already have an account? [Login here.](#)".

Figure 19. New CARTT Registration Page

## 2. Commander Page

The commander page shown in Figure 20 is designed to reflect the role of a CARTT user at the higher headquarters. The expectation is for a commander to send messages with instructions to the operator or the administrator to perform specific tasks. The commander is not expected to conduct red team assessments as their primary function.

**CARTT**

**Commander Message Creation**

Receiving Command's UIC:

Message: (max 255 chars)

Submit

Figure 20. New CARTT Commander Page

### 3. Message Page

The “Message” page is used by all CARTT roles to send text messages between CARTT users. The messages submitted in CARTT are viewable by all users. The UIC provided determines which unit the message is directed to. CARTT users can only send emails to a UIC that is registered in CARTT; the user would get an error if they entered an unregistered UIC. The maximum characters allowed within the message body is set to 255, as shown in Figure 21.

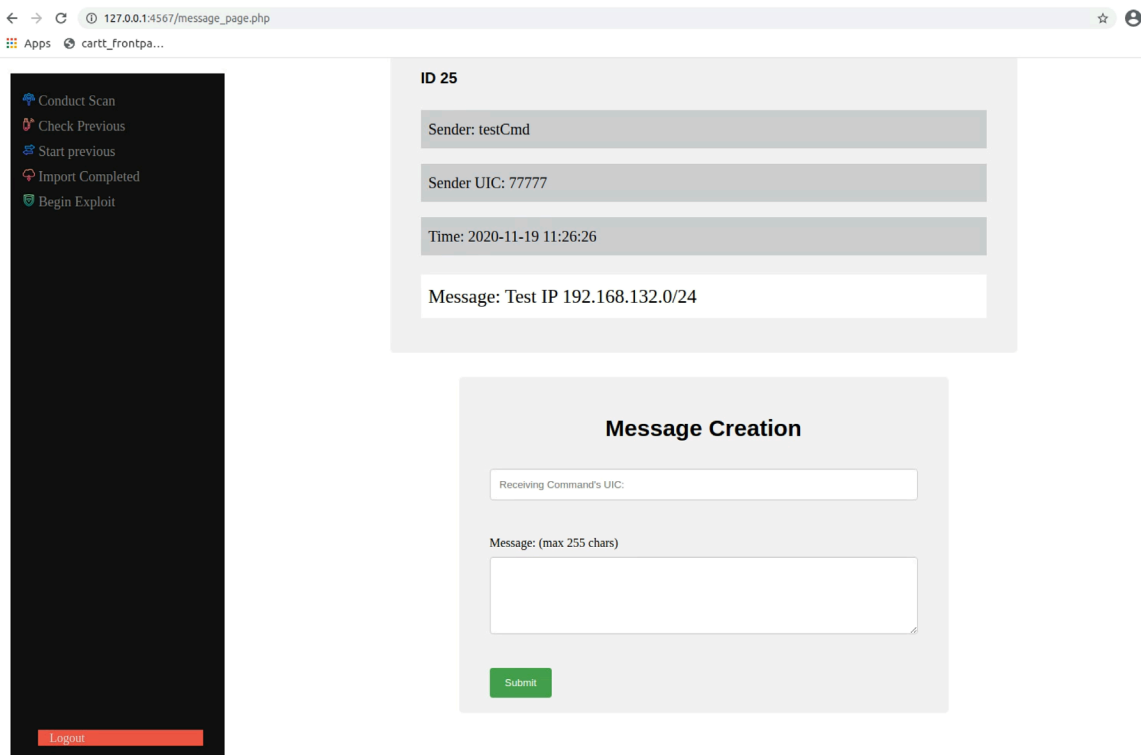


Figure 21. New CARTT Message Page

#### 4. Administrator Page

Figure 22 shows the main page for the administrator with direct access to the menus required to do their assigned tasks. Clear label buttons are provided with text fields to make sure the user understands what action each button provides. As an example, the “Unlock Account” button allows the administrator to unlock accounts that have been locked due to multiple login failures. The buttons serve as a middleman between the user and the action required. They also serve as a translation action between the user and CARTT.

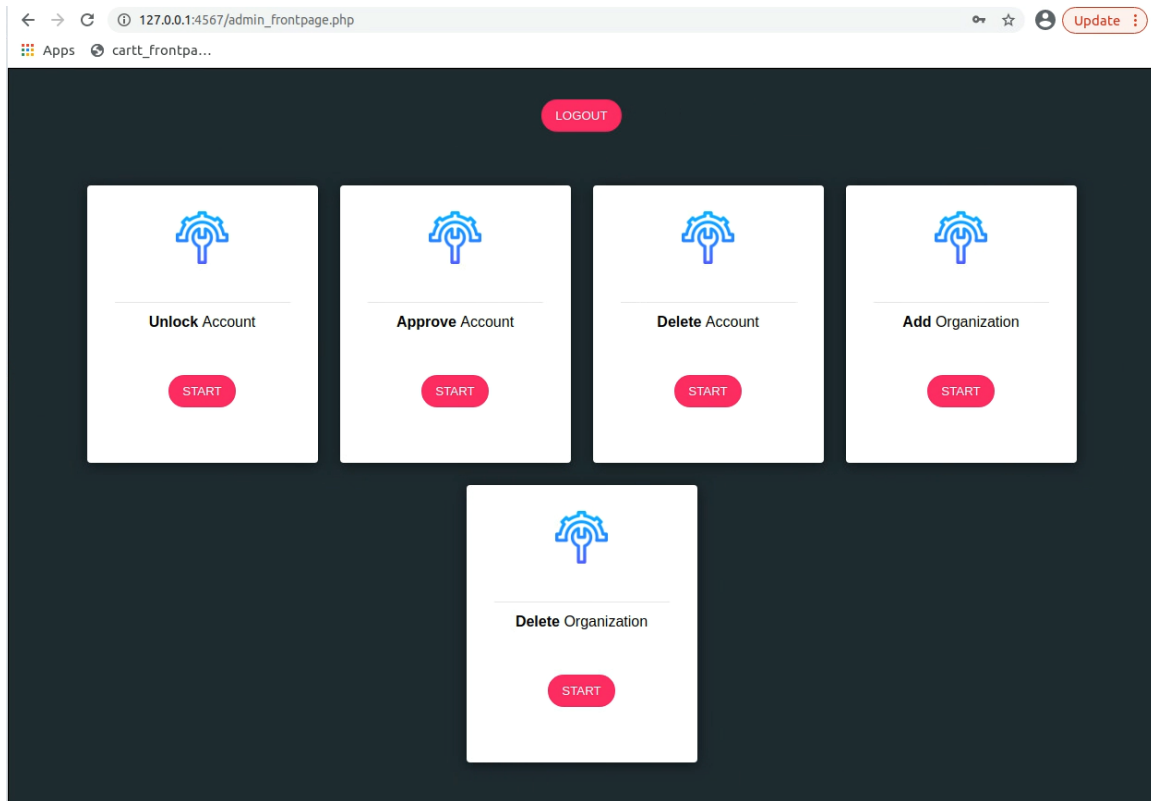


Figure 22. New CARTT Administrator Page

Throughout the new design, warnings have been explicitly provided for irreversible actions so that the user understands their implications. Figure 23 shows an example of this, with warnings displayed so that the administrator has a clear understanding that when an account is deleted, it can no longer be recovered.

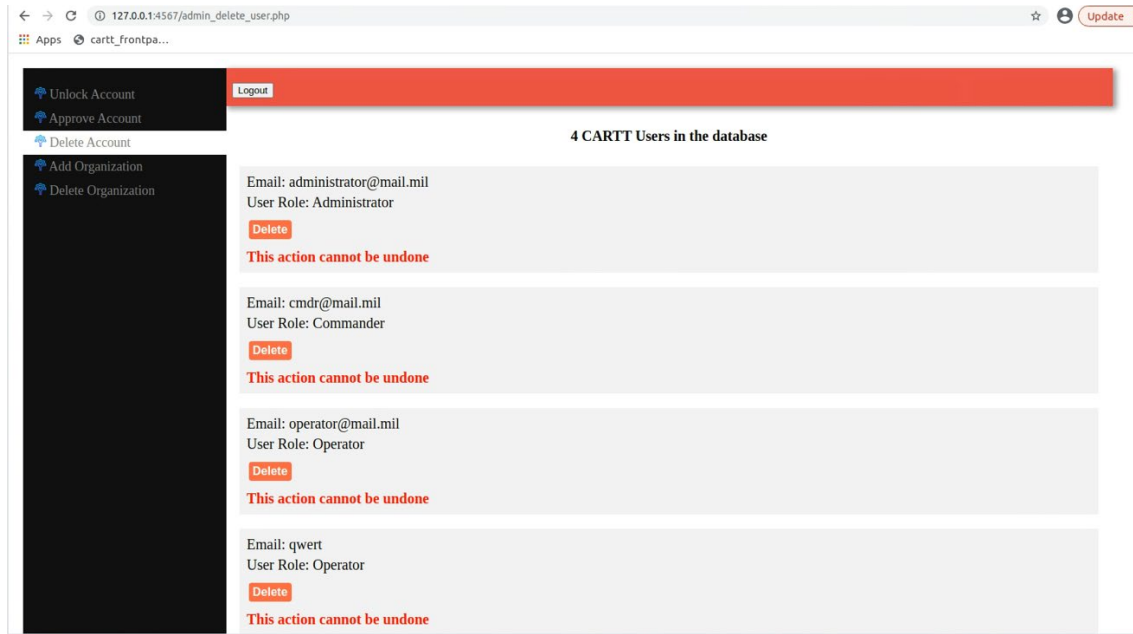


Figure 23. New CARTT Administrator Delete Account Page

## 5. Operator Main Menu

The CARTT operator is the most demanding because most of the red teaming activities are done by this role. The operator menu as shown in Figure 24 has a similar look and feel to the administration menu (see Figure 22) for simplicity and efficiency of design. The action buttons are designed to ensure the operator can easily select the desired task and are displayed in the typical sequence the operator should use to conduct red teaming tasks assigned by the commander. The sidebar navigation menu was designed to the left side of the page (shown in Figure 23) for ease of user action access and readability. The left menu bar will remain available even after the operator selects a task. The name of each icon is used to visually communicate what each button does. Figure 25 shows the implementation code of the left side navigation panel <div> (division/section). Here the block-level container for the HTML file is used to add structure/group components containing shortcuts on the main menu. This Skeuomorphism is used because users are familiar with this concept, and it mimics real-world email applications.

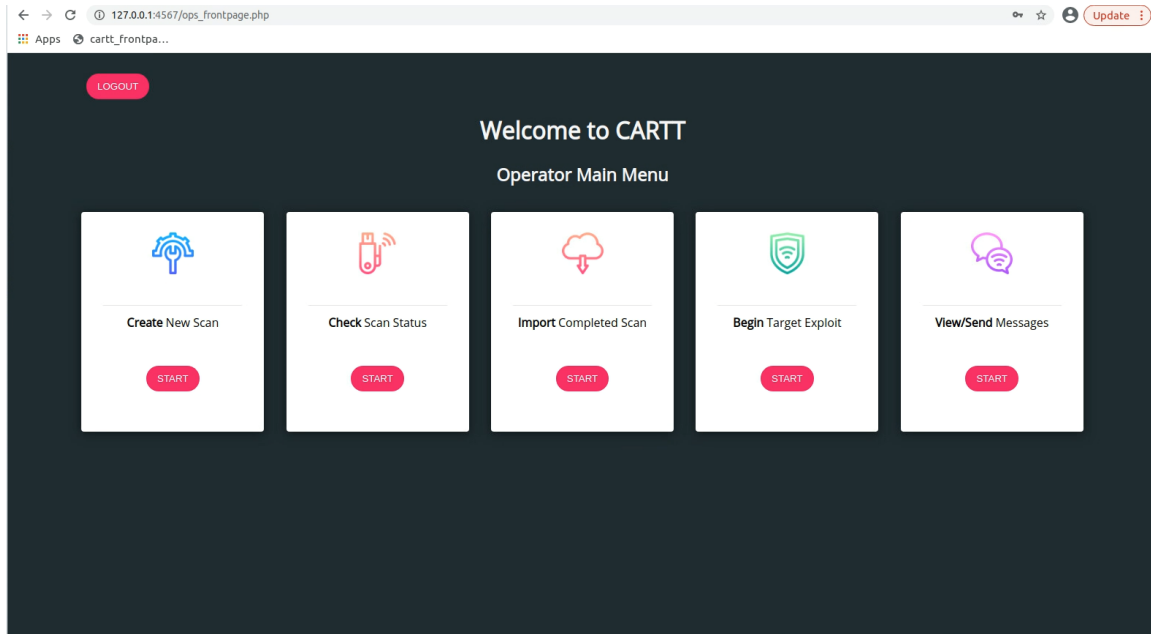


Figure 24. New CARTT Operator Main Menu Page

```
<div class="sidenav">
  <a href="cartt_scan.php">
     Conduct Scan
  </a>

  <a href="cartt_scan_status.php">
     Check Previous
  </a>
  <!--
  <a href="cartt_scan_start.php">
     Start previous
  </a>
  -->
  <a href="cartt_import_scan.php">
     Import Completed
  </a>

  <a href="cartt_workspace.php" class="selected_link">
     Begin Exploit
  </a>

  <a href="message_page.php">
     View/Send Message
  </a>
  <a href="logout.php">
    <div class="bottom_position">
      <label class="logout_button" name="full">Logout</label>
    </div>
  </a>
</div>
```

Figure 25. Left Side Navigation Panel <Div>

A logout option has been added in all pages. This gives the security and the flexibility to the use to completely end a session and return to the login page. Figure 26 shows the code implementation for the logout and redirection.

```
...
<?php
session_start();
session_unset();
session_destroy();
header('Location: login.php');
exit();
?>
```

Figure 26. CARTT Logout and Redirect to Login Page

The following sub-sections describe each of the operator action buttons in the main menu.

## 6. Create New Scan

This action button allows the operator to create and run a vulnerability scan on a target. The operator is required to enter the inputs for a specific target IP address and subnet and provide a scan name as shown in Figure 27. This input format is required because it is the minimum information needed by OpenVAS to perform a scan. This simple UI design is a carryover of the guidelines described in the login page to increase user memorability, learnability, satisfaction, and minimize errors [40]. In order to reduce the possibility of input error, invalid-character checks have been integrated, and the user will receive a pop-up error message to correct an erroneous input. The left menu is used here as well because eye-tracking studies have found that most users look from the top to the left area of a computer monitor, mimicking the most natural reading pattern (left to right and top to bottom) [39]. Once again, the left menu provides the operator with quick access to all the tools available to them.

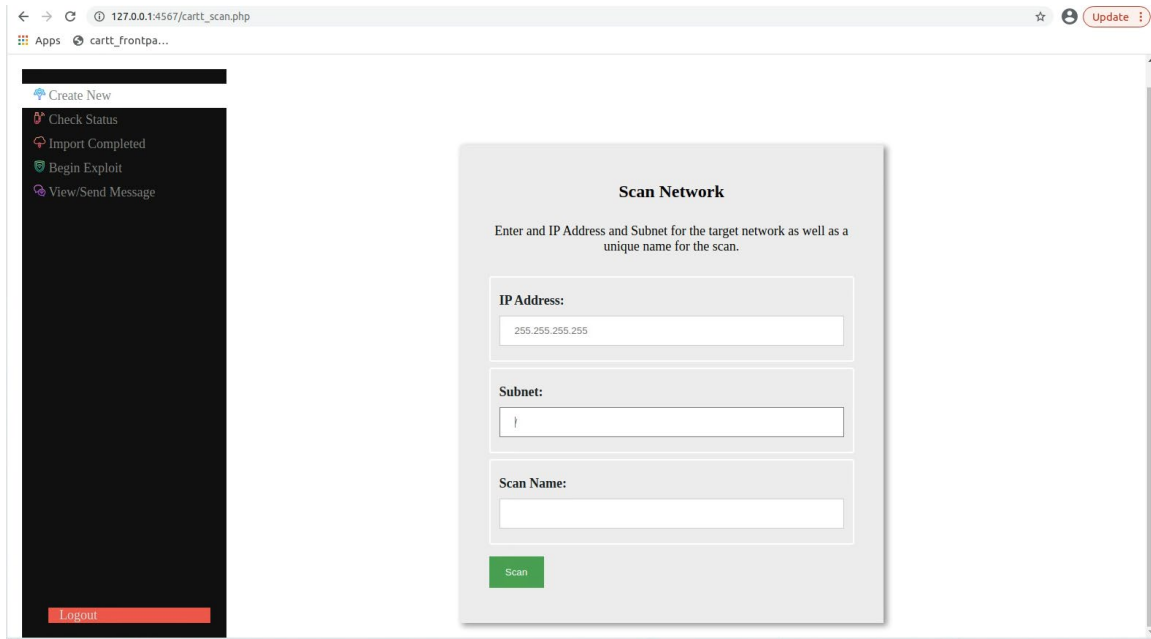


Figure 27. New CARTT Operator Scan Network Menu Page

## 7. Check Scan Status

This action button allows the operator to check the status of a previously scanned network. If the CARTT Scan Status menu shows the user that a scan has completed, this indicates to the operator that the scan is ready for import. The operator can then select the “Import Completed” button on the left to begin importing previously completed scan. Whenever a scan is not completed, the user is notified of the scan status as “In progress.” The operator can import the scan within the same page without selecting a different web page as shown in Figure 28.

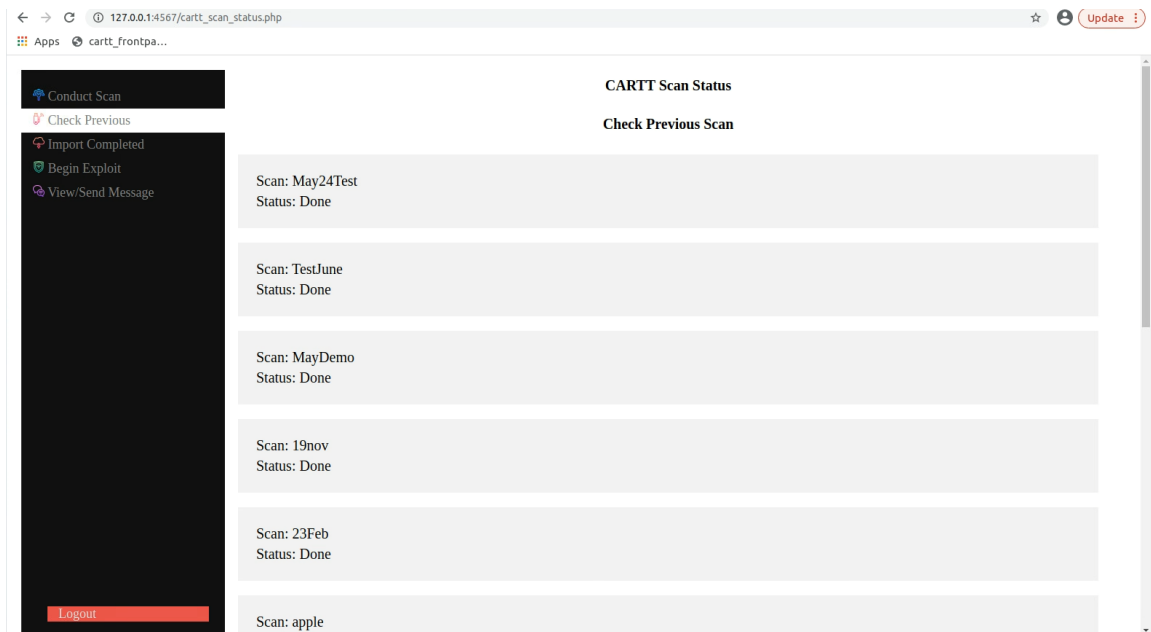


Figure 28. New CARTT Check Previous Scan Page

## 8. Import Completed Scan

This action button allows the operator to import existing completed OpenVAS scans into an MSF workspace. The design maintains the same name for the workspace as that of the scan name for simplicity and ease of use for the operator, as shown in Figure 29. This option arms the operator with the ability to restart previously configured scans without reentering any previous input parameters by selecting the button to import.

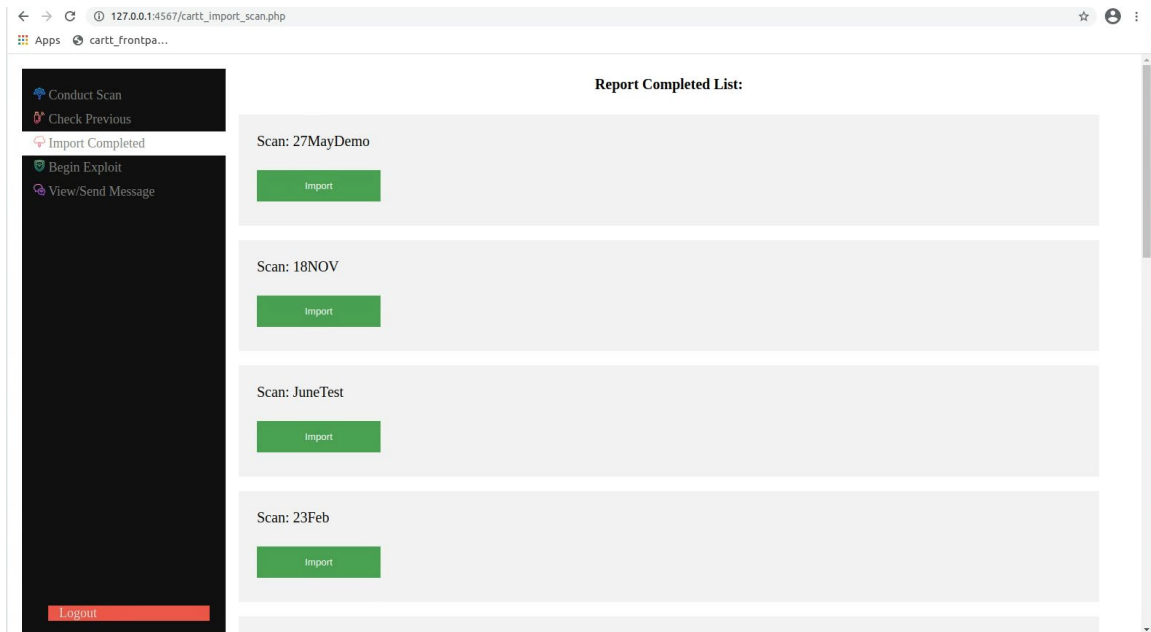


Figure 29. New CARTT Import Completed Page

## 9. Begin Target Exploitation

This action button provides the operator the capability to interact with the MSF workspace so that they can configure and test the available modules as exploit. The operator is given the list of hosts available within the IP space that was previously enter. After selecting the host's IP address for vulnerability analysis, CARTT will display the number of vulnerabilities associated with that host. Currently, CARTT only shows the common vulnerabilities and exposure (CVE).

## I. CHAPTER SUMMARY

This chapter discussed the research design/methodology used to develop and improve the web-based CARTT application. The simplistic design focused on the three CARTT user categories, and primarily the operator. The design used the heuristic evaluation and usability methods to improve the UI and UX.

## **IV. USABILITY TESTING AND EVALUATIONS**

The User Interface test and evaluation were based on the quality measures discussed in Chapters 2 and 3. The heuristic evaluation focused on usability criteria to evaluate the interface design, possible interactive framework errors, button labels, etc. The designer completed testing during the design and development phases but expanding the testing and evaluation to actual users of the tool gave us the chance to improve the design and logic to fit the users' needs.

This chapter will begin with a presentation and discussion of the scenario design that was used to develop the task list that users were asked to complete. The task list was then used to develop the usability specification that, for each task, identified what usability attribute was being tested and what measurement(s) were used to assess if the specification has been met. The scenario testing describes the methodology and details of exactly what participants were instructed to do during the user test. Finally, the user testing data is presented.

### **A. SCENARIO TESTING**

The operator user scenario was created to analyze the operator error rate while trying to accomplish each task in the new CARTT GUI. This test captured the users' satisfaction as they executed each task.

The task analysis provided an in-depth understanding of the importance, frequency, and sequence of actions that a user must execute to complete specific critical tasks within CARTT. Most of the operations and objectives in CARTT are designed around the operator primarily, and the administrator secondarily. Prior to test and evaluate the efficiency, effectiveness, and user satisfaction for these two roles, five volunteers were used for this study. Three are Computer Science, one Cyber Systems Operations, and one National Security Affairs, all students at Naval Postgraduate School (NPS). All were in or beyond their fourth quarter of education at NPS.

## **1. User Briefing**

The briefing language below was used to make sure there was consistency in the testing process. The users needed to understand that we were evaluating the application interface and the focus was not on their ability.

Thank you for agreeing to help us test our interface for a new CARTT application for the following tasks: account creation, configuration, checking previous scan, started importing new scans and viewing and sending messages. We are going to give you a set of tasks we would like you to execute on the prototype to determine the effectiveness of the interface. This is an evaluation of the interface, not of you, so if you have difficulties with any of the tasks, it is because our interface is lacking and is in no way a reflection on you.

For each task, a scenario was read to the test user to provide context, and then they were given the opportunity to execute the scenario tasks. We also provided them the opportunity to ask questions and request clarification before the execution of each task.

## **B. SCENARIO DESIGN**

The test questions were generated based on the application requirements and design to evaluate the UI for all the features to support the functionality, usability, interface, performance, and security features. All five test subjects were graduate students at the Naval Postgraduate School who were familiar with the importance of network vulnerabilities. An experiment record sheet was also used to record the number of unexpected user errors during each task and how satisfied they were with the design.

### **1. High Level Task List**

A task analysis provides an in-depth understanding of the importance, frequency, and sequence of actions that a user must execute to complete certain critical tasks within CARTT. The list provided below identifies only the top-level tasks, not how they are accomplished. These key tasks focus on the operator because most of the operation and the objectives in CARTT are designed around the operator.

Seven high level key tasks were selected as an experimental test and goals.

Key tasks:

- Create an account
- Configure and conduct a scan
- Check status of a previous scan
- Start a previously configured scan
- Import a completed scan
- Begin exploit process
- View/Send messages

## **2. User Scenarios**

The user scenarios were tightly centered on the three types of CARTT users (operator, administrator, and commander). The scenarios were designed to visualize and apply possible solutions captured during the experience as they proceed toward the completion of each task during the experiment.

Scenario 1: As an operator (Tier 2), perform the following task:

- Task 1: Create an account (2 min)
- Task 2: View/Send Messages (2 min)
- Task 3: Based on the message you received from the commander, check the network vulnerabilities within the assigned network and send a message to the commander with your findings. (5 min)
- Task 4: Configuration and Conduct a Scan (2 min)
- Task 5: Check status of a previous scan (2 min)
- Task 6 Start a previously configured scan (2 min)
- Task 7: Import a Completed Scan (2 min)
- Task 8: Begin Exploit Process (3 min)
- Task 9: Request to unlock your account (1 min)

Scenario 2: Log in as the administrator with the provided credential, perform the following task (1 min)

- Task 1: Unlock an account (1 min)
- Task 2: Approve an account (1 min)
- Task 3: Delete an account (1 min)

Scenario 3: Log in as the commander with the provided credential, perform the following task (1 min)

- Task 1: send an email to the Tier 2 with the instruction to conduct a scan with the Scan Name: provided. (2 min)

### C. USABILITY SPECIFICATION

The usability scale specification for the operator is shown in Table 1; it is a comparison between the tasks that the operator performed in the original CARTT design and the target level achieved by the test operator using the expanded graphical UI. This process was used to aim for a target level of efficiency by removing redundant clicks.

Table 1. Operator Usability Scale

| Task                                   | Usability Attribute | Value to be Measured                               | Previous Level* | Current Average Level† |
|----------------------------------------|---------------------|----------------------------------------------------|-----------------|------------------------|
| <b>Create an Account</b>               | Initial impression  | Interview question                                 | N/A             | 1                      |
| <b>Configured Scan</b>                 | Initial impression  | Interview question                                 | 4               | 1                      |
| <b>Conduct a Scan</b>                  | Initial performance | Clicks/taps                                        | 3               | 3                      |
| <b>Conduct a Scan</b>                  | Number of errors    | Any suboptimal click/tap resulting in backtracking | 2               | 1                      |
| <b>Conduct a Scan</b>                  | Efficiency          | Clicks/taps (assuming you must edit everything)    | 3               | 1                      |
| <b>Check status of a previous scan</b> | Initial impression  | Interview question                                 | 3               | 1                      |

| <b>Task</b>                               | <b>Usability Attribute</b> | <b>Value to be Measured</b>                        | <b>Previous Level*</b> | <b>Current Average Level†</b> |
|-------------------------------------------|----------------------------|----------------------------------------------------|------------------------|-------------------------------|
| <b>Check status of a previous scan</b>    | Initial performance        | Clicks/taps                                        | 2                      | 1                             |
| <b>Check status of a previous scan</b>    | Number of errors           | Any suboptimal click/tap resulting in backtracking | 2                      | 0                             |
| <b>Check status of a previous scan</b>    | Efficiency                 | Clicks/taps (assuming you must edit everything)    | 3                      | 1                             |
| <b>Start a previously configured scan</b> | Initial impression         | Interview question                                 | 4                      | 1                             |
| <b>Start a previously configured scan</b> | Initial performance        | Clicks/taps                                        | 3                      | 1                             |
| <b>Start a previously configured scan</b> | Number of errors           | Any suboptimal click/tap resulting in backtracking | 2                      | 0                             |
| <b>Start a previously configured scan</b> | Efficiency                 | Clicks/taps (assuming you must edit everything)    | 3                      | 1                             |
| <b>Import a completed scan</b>            | Initial impression         | Interview question                                 | 4                      |                               |
| <b>Import a completed scan</b>            | Initial performance        | Clicks/taps                                        | 3                      | 1                             |
| <b>Import a completed scan</b>            | Number of errors           | Any suboptimal click/tap resulting in backtracking | 2                      | 0                             |
| <b>Import a completed scan</b>            | Efficiency                 | Clicks/taps (assuming you must edit everything)    | 3                      | 1                             |

\*For the “Previous Level,” after testing against the previous design and procuring user feedback, we used the associated values from the usability attributes as a benchmark to ensure we made positive progression with the new design.

†For the interview/survey questions, we assumed a 5-point Likert scale.

For each participant in the usability study, the following steps were taken:

(1) Participant was briefed as to the purpose of the study and their role, and then given brief instructions as per the user scenario.

(2) Participant executed each task in the sequence, and during task execution their performance was recorded, and data was collected as outlined in the usability specification (see Table 2 below).

(3) Lastly, the data were analyzed to determine if usability specifications have been met with the new UI.

### **1. Data Collection**

For the data collection, I presented the subjects using the 5-point Likert as an evaluation scale. On the five-point difficulty/success scale, each rating represented the following:

- 1 = least difficult (task completed successfully with no error)
- 2 = minor difficulties (task completed after short, unsuccessful clicks/short detour)
- 3 = moderate difficulties (task completed by self and after multiple unsuccessful clicks/long detour)
- 4 = major difficulties (task completed with the assistance and after numerous unsuccessful clicks/long detour)
- 5 = most difficult task or give up

Each user was interviewed immediately following each task. At the end of each scenario, the subjects were asked how satisfied they felt and what changes they would suggest to improve the GUI.

## **D. RESULTS**

After getting users' responses on usability testing for the operator about the current CARTT user interface, it was essential to know about their overall satisfaction on each task

performed during the experiment. Table 2 shows the results from the five student subjects on how satisfied they were completion of each task and common trends. Table 3 shows the results of the administrators' tasks, and Table 4 shows the results of the commanders' tasks.

Table 2. Operator Tasks

|                                                                                                        |                                                                                                                                                                                                                  |
|--------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Task 1.1 Create an Account                                                                             |                                                                                                                                                                                                                  |
| Average Satisfaction                                                                                   | 1, 1, 1, 2, 1 with an average of $\approx 1$                                                                                                                                                                     |
| Most Common Remarks                                                                                    | The account type selection should not be set to administrator as the first option for the user. The user should be forced to select from the current dropdown menu.                                              |
| Task 2.1 Configuration and Conduct a Scan                                                              |                                                                                                                                                                                                                  |
| Average Satisfaction                                                                                   | 1, 1, 1, 2, 1 with an average of $\approx 1$                                                                                                                                                                     |
| Most Common Remarks                                                                                    | Entering the data requested was made easy but it took too long to display, and the user was not sure if the task requested was submitted.                                                                        |
| Task 2.2. Enter an IP address and Subnet for the Target network as well as a unique name for the scan. |                                                                                                                                                                                                                  |
| Average Satisfaction                                                                                   | 1, 1, 1, 2, 1 with an average of $\approx 1$                                                                                                                                                                     |
| Most Common Remarks                                                                                    | Entering the data requested was made easy but it took too long to display, and the user was not sure if the task requested was submitted.                                                                        |
| Task 2.3. Enter the name of a scan to check its progress                                               |                                                                                                                                                                                                                  |
| Average Satisfaction                                                                                   | 1, 1, 2, 2, 1 with an average of $\approx 2$                                                                                                                                                                     |
| Most Common Remarks                                                                                    | Entering the data requested was made easy but it took too long to display, and the user was not sure if the task requested was submitted.                                                                        |
| Task 2.4. Please select the IP address of the target host.                                             |                                                                                                                                                                                                                  |
| Average Satisfaction                                                                                   | 1, 1, 1, 1, 1 with an average of $= 1$                                                                                                                                                                           |
| Most Common Remarks                                                                                    | Entering the data requested was made easy but it took too long to display, and the user was not sure if the task requested was submitted.<br><br>A selection of the host for further analysis and display of the |

|                                                                               |                                                                                                                                                                            |
|-------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                                                               | vulnerability vs. manually entering or copy and paste may not be the best option because of the increase of errors.                                                        |
| Task 3.1. Check status of a previous scan                                     |                                                                                                                                                                            |
| Average Satisfaction                                                          | 1, 1, 1, 1, 1 with an average of = 1                                                                                                                                       |
| Most Common Remarks                                                           | Requesting visual feedback in the “Scan Status” page, e.g.: green check marks for successfully completed scan, red “X” for failed scan, or yellow for pending/in progress. |
| Task 4.1. Start a previously configured scan                                  |                                                                                                                                                                            |
| Average Satisfaction                                                          | 1, 2, 2, 2, 1 with an average of $\approx 2$                                                                                                                               |
| Most Common Remarks                                                           | Entering the data requested was made easy but it took too long to display, and the user was not sure if the task requested was submitted.                                  |
| Task 5.1. Import a Completed Scan                                             |                                                                                                                                                                            |
| Average Satisfaction                                                          | 1, 2, 2, 2, 1 with an average of $\approx 2$                                                                                                                               |
| Most Common Remarks                                                           | Entering the data requested was made easy but it took too long to display, and the user was not sure if the task requested was submitted.                                  |
| Task 5.2 Please enter the CVE of a Vulnerability                              |                                                                                                                                                                            |
| Average Satisfaction                                                          | 1, 2, 2, 2, 1 with an average of $\approx 2$                                                                                                                               |
| Most Common Remarks                                                           | Requesting to be able to preselect the CVE displayed for the configuration process for exploitation.                                                                       |
| Task 5.3 Copy and paste a module into the text field to construct an exploit. |                                                                                                                                                                            |
| Average Satisfaction                                                          | 1, 2, 2, 2, 1 with an average of $\approx 2$                                                                                                                               |
| Most Common Remarks                                                           | Requesting to be able to preselect the CVE displayed for the configuration process for exploitation.                                                                       |
| Task 6.1. Begin Exploit Process                                               |                                                                                                                                                                            |
| Average Satisfaction                                                          | 1, 2, 2, 2, 2 with an average of $\approx 2$                                                                                                                               |
| Most Common Remarks                                                           | Requesting to be able to preselect the module displayed for exploitation attempt and validation vs. copy and paste in order to minimize errors.                            |

|                              |                                                                                                                     |
|------------------------------|---------------------------------------------------------------------------------------------------------------------|
| Task 7.1. View/Send Messages |                                                                                                                     |
| Average Satisfaction         | 1, 1, 1, 2, 1 with an average of $\approx 1$                                                                        |
| Most Common Remarks          | Requesting to view only email assigned to the organization so that they can focus only on their organization tasks. |

Table 3. Administrator Tasks

|                            |                                                                                                                                                                                                                                  |
|----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Task 1 Unlock Account      |                                                                                                                                                                                                                                  |
| Average Satisfaction       | 1, 1, 1, 2, 1 with an average of $\approx 1$                                                                                                                                                                                     |
| Most Common Remarks        | Requesting that the user get a notification message that the account has been unlock.                                                                                                                                            |
| Task 2 Approve Account     |                                                                                                                                                                                                                                  |
| Average Satisfaction       | 1, 1, 1, 2, 1 with an average of $\approx 1$                                                                                                                                                                                     |
| Most Common Remarks        | Requesting that the user get a notification message that the account has been approved.                                                                                                                                          |
| Task 3 Delete Account      |                                                                                                                                                                                                                                  |
| Average Satisfaction       | 1, 1, 1, 2, 1 with an average of $\approx 1$                                                                                                                                                                                     |
| Most Common Remarks        | Requesting a second warning page or popup before the delete is final. This is a critical task therefore 2 warnings should be appropriate. Requesting that the user get a notification message that the account has been deleted. |
| Task 4 Add Organization    |                                                                                                                                                                                                                                  |
| Average Satisfaction       | 1, 1, 1, 1, 1 with an average of $\approx 1$                                                                                                                                                                                     |
| Most Common Remarks        | Requesting a drop-down menu instead of user entering the five digits. The current design increases the number of potential error entry.                                                                                          |
| Task 5 Delete Organization |                                                                                                                                                                                                                                  |
| Average Satisfaction       | 1, 1, 1, 1, 1 with an average of $\approx 1$                                                                                                                                                                                     |
| Most                       | Requesting that the user get a notification message that the                                                                                                                                                                     |

|                |                           |
|----------------|---------------------------|
| Common Remarks | account has been deleted. |
|----------------|---------------------------|

Table 4. Commander Tasks

|                       |                                                                                                                                                                    |
|-----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Task 1 Access Account |                                                                                                                                                                    |
| Average Satisfaction  | 1, 1, 1, 1, 1 with an average of $\approx 1$                                                                                                                       |
| Most Common Remarks   | Change the UIC to command name.                                                                                                                                    |
| Task 2 Send Message   |                                                                                                                                                                    |
| Average Satisfaction  | 1, 1, 1, 1, 1 with an average of $\approx 1$                                                                                                                       |
| Most Common Remarks   | Requesting that the user get a notification message that the account has been send and that the user received the message, and the tasking is in work in progress. |

From the collected data, the completion of each task was analyzed. The tasks were divided by success level criteria, as shown in Tables 5 and 6, using the 5-point Likert scale scoring method.

Table 5. Task Success Rate Criteria

| Success rate criteria | Classification                                                                       |
|-----------------------|--------------------------------------------------------------------------------------|
| 1                     | Task completed successfully with no error                                            |
| 2                     | Task completed after short unsuccessful clicks/short detour                          |
| 3                     | Task completed without assistance and after multiple unsuccessful clicks/long detour |
| 4                     | Task completed with assistance after multiple unsuccessful clicks/long detour        |
| 5                     | Task incomplete                                                                      |

Table 6. High-Level Task Success Results

| Task         | Task 1.1                                     | Task 2.1                                     | Task 3.1                                     | Task 4.1                                     | Task 5.1                                     | Task 6.1                                     | Task 7.1                                     |
|--------------|----------------------------------------------|----------------------------------------------|----------------------------------------------|----------------------------------------------|----------------------------------------------|----------------------------------------------|----------------------------------------------|
| Success Rate | 1, 1, 1, 2, 1 with an average of $\approx 1$ | 1, 2, 1, 2, 1 with an average of $\approx 1$ | 1, 1, 1, 1, 1 with an average of $\approx 1$ | 1, 2, 1, 1, 1 with an average of $\approx 1$ | 1, 2, 1, 1, 1 with an average of $\approx 2$ | 1, 2, 2, 2, 1 with an average of $\approx 2$ | 1, 1, 1, 1, 1 with an average of $\approx 1$ |

In Table 6, the number of failed tasks was not used in this analysis because completion of tasks was the focus of this analysis. Thus, failed tasks did not have any significance during the test and evaluation.

We determined that calculating the average completion time of each task was not a valid way to measure effectiveness or user satisfaction for this study since some scenarios took very long to complete after the user had completed all tasks. For instance, the processes to “Conduct a Scan” can take several hours to complete when the operator must scan for vulnerabilities on a large operational network, and the time completion cannot be used as a measure of user ability or satisfaction.

## 1. Result Analysis and Discussion

Based on the requirement set for the usability testing of CARTT, the success rate measurement was evaluated and analyzed. All the tasks were accomplished, and the type and the number of errors were primarily related to the virtual environment and not the UI design. In general comments, the subjects were not very satisfied with the network latency during the testing. The users got impatient waiting for a network response, which led to them clicking multiple times on an execution button.

The success rate was associated with the simplicity of the page and that most of the data were started locally with the help of PostgreSQL. Another factor contributing to the success rate is the familiarity with the user input requested and the page’s layout. Users noticed the usefulness of the left-side vertical navigation bar that provided quick access to the critical tasks and showed the actual content and effectiveness of the site readily available.

## **2. Conclusions and Recommendations**

We leveraged the diagnostic evaluation criteria above, necessitated the use of five test subjects to find and fix problems associated with the application. The corrective actions required to address the deficiencies identified in and during the usability analysis were implemented during the design phases. The test and evaluation were conducted to assess the quantitative (task accomplished in a scenario and errors) and qualitative (user satisfaction) CARTT's expanded graphical user interface and user experience. The user satisfaction was likely due to very slow performance due to the use of the virtual machine to do the CARTT test and evaluation. We could leverage jQuery or similar technologies to provide visual indications and feedback, but we cannot change the server response speed. Popular frameworks are PHPUnit and Codeception and the use of bare-metal server may help to isolate this issue.

## **E. CHAPTER SUMMARY**

This chapter focused on testing and evaluating CARTT functionality with the new UI and user satisfaction while interacting with the GUI. During the experiment, the five subjects felt that CARTT had overall performance issues due to back-end bugs. The ten usability heuristics used to design the UI should continue to be used for the UI, emphasizing error prevention and recognition. CARTT implementation and testing design difficulties were also addressed in this chapter.

## **V. CONCLUSIONS AND FUTURE WORK**

### **A. SUMMARY**

The primary goal of this thesis was to extend CARTT development by integrating a GUI for accessing open-source penetration testing software tools and techniques to uncover network vulnerabilities and launch exploits against those vulnerabilities. The main objective was to improve the aesthetics of CARTT and improve responsiveness and efficiency for improving usability and the user experience.

This research has proven that we can fully automate the specific process to scan devices and detect vulnerabilities on a targeted network without the extensive training required to perform DOD red teaming. The work continues to demonstrate that we can fully automate open-source network scanning and vulnerability assessment tools to address the number of DOD red teams available and the ill-proportioned cybersecurity team to support the ever-growing DOD digital footprint. A usability testing was performed to assess the GUI performance using some of Nielsen's ten usability heuristics as a parameter. Some modifications were recommended by the test subjects' feedback to improve the usability of CARTT.

### **B. CONCLUSION**

The CARTT GUI was designed for non-expert users with little to no experience and knowledge of using tools such as OpenVAS and MSF. The expanded GUI for web-based CARTT provides its users with the essentials tools necessary to conduct vulnerability scans by automating tasks that require knowledge and experience in red teaming, in particular to scan the vulnerabilities on an organization's targeted IT infrastructure. The user interface plays a fundamental role in software use and utility. The user experience is based on the exact amount of information required by the user. The design of the user interface can have a significant positive or negative impact on the user experience. Nielsen's ten usability heuristics can be used as a strategy to successfully design user interfaces. We address the specific research questions for this thesis in the following sections.

## **1. Primary Research Question**

*How can a web-based graphical user interface be developed for CARTT to allow non-expert users to build, integrate, and launch exploits using Cyber Red Team tools and techniques?*

We successfully developed the web-based graphical user interface for CARTT, and tested the web-based GUI to evaluate the site's functionality, effectiveness, and efficiency. We also focused on learnability, memorability, minimizing errors, and improving user satisfaction based on the new interface.

## **2. Second Research Question**

*How can CARTT safely integrate automation into a client/server architecture without causing system issues during automated vulnerability scan?*

To make sure CARTT is fully functional, we used subjects to test its end-to-end functionality. CARTT users, operators, administrators, and commanders could log in securely, configure scan on a remote target host, conduct a vulnerability scan, check the status of the previous scans, import the last scan, and send messages between users. CARTT users must know essential network information related to the target host, such as IP address, subnet mask, and see the name of previous scans.

## **C. FUTURE WORK**

This thesis focused on improving the user experience by expanding the graphical user interface, making the layout consistent, and remove redundant functions on the web browser. CARTT's capabilities and user experience should continue to expand because of growing cybercriminals looking to penetrate systems, networks, services, and facilities.

The following are some recommendations for future work to improve CARTT further.

### **1. Improve the Backend of the User Interface**

There are many possible ways to improve the compliance of UI design and improve the software quality. The code may require some refactoring around the

Model View Controller or similar architecture. This will help separate the user interface from the business logic and perform unit testing on the business logic. Due to the constant involvement of UI design, a system could be developed to review, check for any errors, and check for any design features that are too odd for CARTT (i.e., colors, notation consistency, shapes, etc.).

## **2. CARTT User Expansion**

The login could include an additional user as “standard.” The user could automatically scan their own device to assess any vulnerabilities that could allow unauthorized users to access their computer. The standard user should have the capability to raise an alert to their network administrator to patch these vulnerabilities in order to increase cybersecurity.

## **3. Improve the Scan Result**

CARTT could provide the high-risk level classification of the network vulnerabilities discovered. The display of the level of severity and how a problem could be fixed would be helpful information for the operator as for the commander. The classification of the severity, number, and types of vulnerabilities could also be displayed in a comprehensive and graphical report. This display can become very useful when the users are scanning an extensive network and could help provide an overview of the network’s health.

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF REFERENCES

- [1] Followup Audit on Corrective Actions Taken by DOD Components in Response to DOD Cyber Red Team-Identified Vulnerabilities and Additional Challenges Facing DOD Cyber Red Team Missions, DODIG-2020-067. Department of Defense, Washington, DC, USA, 2020
- [2] J. Plot, Red “Team in a box (RTIB): Developing automated tools to identify, assess, and expose cybersecurity vulnerabilities in Department of the Navy systems,” M.S. thesis, Dept. of Comp. Sci., NPS, Monterey, CA, USA, 2019. [Online]. Available: <http://hdl.handle.net/10945/62832>
- [3] P. Edwards, “Cyber automated red team tool” M.S. thesis, Dept. of Comp. Sci., NPS, Monterey, CA, USA, 2019. [Online]. Available: <http://hdl.handle.net/10945/64145>
- [4] J. Berrios, “A client/server model for automated red teaming,” M.S. thesis, Dept. of Comp. Sci., NPS, Monterey, CA, USA, 2021 [Online]. Available: <https://calhoun.nps.edu/handle/10945/66584>
- [5] P. A. Grassi, M. E. Garcia, and J. L. Fenton, “Digital identity guidelines: revision 3,” June 2017, NIST Special Publication 800–63-3 [Online]. Available: <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-3.pdf>
- [6] The Penetration Testing Execution Standard, “Main page,” 16 August 2014. [Online]. Available: [http://www.pentest-standard.org/index.php/Main\\_Page](http://www.pentest-standard.org/index.php/Main_Page)
- [7] M. Johnson, “DISA Strategic Communication and Public Affairs,” Defense Information Systems Agency, 20 May 2019. [Online]. Available: <https://disa.mil/NewsandEvents/2019/automated-tools-cyber-threats-analysis>
- [8] Director, Operational Test & Evaluation (DOT&E), “Cyber Assessments.” FY19 Cybersecurity, 30 January 2020. [Online]. Available: <https://www.dote.osd.mil/Portals/97/pub/reports/FY2019/other/2019cyber-assessments.pdf?ver=2020-01-30-115600-800>
- [9] J. Schab, “Tackling DOD Cyber Red Team Deficiencies Through Systems Engineering, 2017.” [Online]. Available: <https://www.sans.org/reading-room/whitepapers/testing/tackling-dod-cyber-red-team-deficiencies-systems-engineering-38020>

- [10] R. S. Cohen, “DOD’s Cyber ‘Red Teams’ Stressed as Security Tests Grow,” Air Force Magazine, 04 Feb 2019 [Online]. Available: <https://www.airforcemag.com/dods-cyber-red-teams-stressed-as-security-tests-grow>
- [11] Mimecast, “The State of Email Security 2020.” Accessed 24 March 2020. [Online]. Available: [https://www.mimecast.com/globalassets/cyber-resilience-content/the\\_state\\_of\\_email\\_security\\_report\\_2020.pdf](https://www.mimecast.com/globalassets/cyber-resilience-content/the_state_of_email_security_report_2020.pdf)
- [12] Schneider Downs, “Purple Team Assessment.” Accessed 11 Dec 2020 [Online]. Available: <https://www.schneiderdowns.com/cybersecurity/services/purple-team-assessment>
- [13] Department of Defense Cyber Red Team Certification and Accreditation, CJCSM 6510.03, Joint Chiefs of Staff, Washington, DC, USA, 2013 [Online]. Available: <https://www.jcs.mil/Portals/36/Documents/Library/Manuals/m651003.pdf?ver=2>
- [14] Cyberspace Operations, Joint Publication 3-12, Joint Chiefs of Staff, Washington, DC, USA, 2018 [Online]. Available: [https://www.jcs.mil/Portals/36/Documents/Doctrine/pubs/jp3\\_12.pdf](https://www.jcs.mil/Portals/36/Documents/Doctrine/pubs/jp3_12.pdf)
- [15] Computer Fraud and Abuse Act of 1986, 2020. 18 U.S.C. § 1001. 1986 [Online]. Available: <https://www.congress.gov/99/statute/STATUTE-100/STATUTE-100-Pg1213.pdf>
- [16] J. Petters, “What is Metasploit? The Beginner’s Guide,” blog, March 29, 2020. [Online]. Available: <https://blogvaronis2.wpengine.com/what-is-metasploit/>
- [17] Rapid7, “Using the Metasploit Web Interface,” Accessed 09 December 2020 [Online]. Available: <https://docs.rapid7.com/metasploit/metasploit-web-interface-overview/>
- [18] A. Shanley and M. N. Johnstone, “Selection of penetration testing methodologies: A comparison and evaluation,” Australian Information Security Management Conference, Edith Cowan University, Perth, Western Australia, 2015. [Online]. Available: <https://ro.ecu.edu.au/cgi/viewcontent.cgi?article=1181&context=ism>
- [19] OpenVAS by Greenbone, “Open Vulnerability-Assessment Scanner,” Accessed 11 November 2020 [Online]. Available: <https://www.openvas.org/index.html>
- [20] Web Technology, “Usage Statistics and Market Share of Server-side Programming Languages for websites,” Accessed 11 November 2020 [Online]. Available: [https://w3techs.com/technologies/overview/programming\\_language](https://w3techs.com/technologies/overview/programming_language)
- [21] PHP, “What do I need?” Accessed 19 January 2021 [Online]. Available: <https://www.php.net/manual/en/tutorial.requirements.php>

- [22] MySQL, “What is MySQL?” Accessed 11 November 2020 [Online]. Available: <https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html>
- [23] S. Verma, “Top 5 databases for PHP web application development,” Veon Consulting, 19 May 2014 [Online]. Available: <https://www.veonconsulting.com/top-databases-for-php/>
- [24] A. Komninos, “An Introduction to Usability,” Accessed 24 November 2020 [Online]. Available: <https://www.interaction-design.org/literature/article/an-introduction-to-usability>.
- [25] S. Krug, Usability testing on 10 cents a day in Don’t Make Me Think, Revisited: A Common Sense Approach to Web Usability, 3rd Ed., New Riders, 2014, Available: O’Reilly
- [26] G. A. Miller, “The magical number seven, plus or minus two: Some limits on our capacity for processing information,” *Psychological Review*, no. 101(2), 1994 pp. 343–352.
- [27] H. A. Simon, *Administrative behavior: a study of decision-making processes in administrative organizations*, New York, NY, USA: Macmillan, 1947.
- [28] M. Soegaard, *The Basics of User Experience Design*. Interaction Design Foundation, 2020 [Online]. Available: <https://www.socialsavvi.com/wp-content/uploads/2018/10/The-Basics-of-user-experience-design.pdf>
- [29] Minimum Security Requirements for Federal Information and Information Systems, FIPS PUB 200, Gaithersburg, MD, USA, 2006. [Online]. Available: <https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.200.pdf>
- [30] U.S. General Services Administration, Technology Transformation Services, “User Interface Design Basics.” Accessed 20 November 2020. [Online]. Available: <https://www.usability.gov/what-and-why/user-interface-design.html>
- [31] National Institute of Standards and Technology “Computer Security Resource Center (CSRC),” Accessed 26 November 2020 [Online]. Available: <https://csrc.nist.gov/glossary>
- [32] Positive Technologies, “Web application vulnerabilities and threats: statistics for 2019,” Accessed 25 November 2020 [Online]. Available: <https://www.ptsecurity.com/upload/corporate/ww-en/analytics/web-vulnerabilities-2020-eng.pdf>
- [33] OWASP, “Access Control.” Accessed 21 January 2021 [Online]. Available: [https://owasp.org/www-community/Access\\_Control](https://owasp.org/www-community/Access_Control)

- [34] S. Nagpure and S. Kurkure, “Vulnerability Assessment and Penetration Testing of Web Application,” in 2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA), PUNE, India, pp. 1– 6, 2017
- [35] Software Testing Help, “Web Application Testing Complete Guide (How To Test A Website),” July 31, 2021. [Online]. Available: <https://www.softwaretestinghelp.com/web-application-testing>
- [36] National Institute of Standards and Technology, NIST Special Publication 800–63B Washington, DC, USA, 2017 [Online]. Available: <https://doi.org/10.6028/NIST.SP.800-63b>
- [37] E. Wong, “User Interface Design Guidelines: 10 Rules of Thumb,” User Interface Design, September 2020. [Online]. Available: <https://www.interaction-design.org/literature/article/user-interface-design-guidelines-10-rules-of-thumb>
- [38] U.S. Web Design System (USWDS), “Data visualizations.” Accessed 20 November 2020 [Online]. Available: <https://designsystem.digital.gov/components/data-visualizations>
- [39] J. R. Kingsburg and A. D. Andre “A comparison of three-level menu navigation structures for web design,” presented at the PROCEEDINGS of the Human Factors and Ergonomics Society 48th Annual Meeting, San Jose State University, San Jose, USA, 2004 [Online]. Available: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1005.5726&rep=rep1&type=pdf>
- [40] J. Nielsen, “10 Usability Heuristics for User Interface Design,” Nielsen Norman Group, November 15, 2020. [Online]. Available: <https://www.nngroup.com/articles/ten-usability-heuristics>
- [41] B. Brumfield, G. Cox, N. Delgado, M. Puleio, K. Shifflett, D. Smith and D. Taylor, “Project Silk: Client-Side Web Development for Modern Browsers, 1st ed.,” Microsoft patterns & practices, 2013, pp. 51–75.
- [42] J. Kibby, “How Can a Mobile-Freindly website Boost User Engagement,” Griffon Webstudios, 15 April 2018 [Online]. Available: <https://griffonwebstudios.com/mobile-friendly-website-boost-user-enagement>
- [43] C. Zapata, “Evaluating UX by Measuring Task Efficiency,” UX Magazine, 13 June 2013 [Online]. Available: <https://uxmag.com/articles/evaluating-ux-by-measuring-task-efficiency>

## **INITIAL DISTRIBUTION LIST**

1. Defense Technical Information Center  
Ft. Belvoir, Virginia
2. Dudley Knox Library  
Naval Postgraduate School  
Monterey, California