AVOIDIT: A Cyber Attack Taxonomy

Chris Simmons, Charles Ellis, Sajjan Shiva, Dipankar Dasgupta, Qishi Wu
Department of Computer Science
University of Memphis
Memphis, TN, USA
{cbsmmons, ceellis, sshiva, ddasgupta, qishiwu}@memphis.edu

***Abstract*—Cyber attacks have greatly increased over the years, where the attackers have progressively improved in devising attacks towards a specific target. To aid in identifying and defending against cyber attacks we propose a cyber attack taxonomy called AVOIDIT (Attack Vector, Operational Impact, Defense, Information Impact, and Target). We use five major classifiers to characterize the nature of an attack, which are classification by attack vector, classification by attack target, classification by operational impact, classification by informational impact, and classification by defense. Our fifth category, classification by defense, is used to provide the network administrator with information of how to mitigate or remediate an attack. Contrary to the existing taxonomies, our taxonomy efficiently classifies blended attacks. Our taxonomy is applied using an application approach with pabulum to educate the defender on possible cyber attacks.**

Keywords - taxonomy; cyber attack taxonomy; vulnerability; computer security; cyberspace

I. INTRODUCTION

Cyber attacks have created a global threat, both in defending local and global networks. Attacks are becoming more sophisticated and possess the ability to spread in a matter of seconds. It is essential to provide tools necessary in detecting, classifying, and defending from various types of attacks. A variety of taxonomies aim at classifying vulnerabilities or attacks, but to date they have limitations in providing a defense strategy that can be used in a local application setting. This can be due to the enormous possibilities of defense strategies. We believe that coupling a defense mechanism with an attack taxonomy would enable a network administrator to not only understand the vulnerability, but also the strategy needed to mitigate and/or remediate the potential exploitation. Limitations exist toward providing defense strategies within an attack taxonomy. This presents an invaluable research area focused on the information a network administrator can apply when attempting to defend the network against cyber attacks. We propose a solution that addresses the shortcomings of existing taxonomies.

There is a deficient standard when disseminating vulnerability information, making it difficult for analysis with multiple vulnerabilities for potential defense. Landwehr et al. [1] state a taxonomy is most useful when it classifies threats in scope that correspond to potential defenses. This taxonomy differs from previous taxonomies, as it aids a defender to not only identify attacks, but also defense measures to mitigate and remediate attack vulnerabilities. One approach to gaining insight into attacker's target is to consider the attack paths, or combination of exploits [2]. AVOIDIT intends to provide a defender with vulnerability details to what encompasses an attack and any impact the attack may have on a targeted system. A blended attack exploits one or more vulnerabilities to perform an attack against a target [3]. AVOIDIT is able to classify blended attacks by providing the ability to label various vulnerabilities of an attack in a tree-like structure.

People question the impact a cyber attack has once its target is compromised. AVOIDIT provides useful information to the network administrator. We provide a mean to classify vulnerabilities that lead to cyber attacks with methods to mitigate and remediate vulnerabilities to help alleviate the impact of a successful exploitation. Avoiding the attack could simply require defending against propagation or further damage once an attack is identified. In order to better grasp this scenario, we provide several representative examples of attacks and how our proposed taxonomy successfully classifies well known attacks with defensive strategies.

Our paper is organized as follows: In Section 2 we survey previous attack taxonomies. In Section 3, we highlight requirements for a taxonomy and propose AVOIDIT a cyber attack taxonomy. In Section 4, we use well known attacks to compare previous taxonomies with AVOIDIT and show how our taxonomy is able to classify a vast majority of attacks. In Section 5, we show how AVOIDIT can be applied as an organizational element within a network setting. In Section 6, we present limitations along with areas for continued research and Section 7 we conclude this paper.

II. A BRIEF SURVEY OF ATTACK TAXONOMIES

Kjaerland [4] proposed a taxonomy of cyber-intrusions from Computer Emergency Response Team (CERT) related to computer crime profiling, highlighting cyber-criminals and victims. In this research, attacks were analyzed using facet theory and multidimensional scaling (MDS) with Method of Operation, Target, Source, and Impact. Each facet contains a number of elements with an exhaustive description. Kjaerland uses these facets to compare commercial versus government incidents. Kjaerland's taxonomy focuses on the motive of the attacker in an attempt to quantify why the attack takes place, and where the attack originated. Her taxonomy contains some limitations as she provides a high level view to the methods of

operation without providing more details to the methods that can be used in identifying attack inception.

Hansman and Hunt [6] proposed a taxonomy with four unique dimensions that provide a holistic classification covering network and computer attacks. Their taxonomy provides assistance in improving computer and network security as well as consistency in language with attack description. The first dimension being attack vector is used to classify the attack. The second dimension classifies the target of the attack. The third dimension consists of the vulnerability classification number, or criteria from Howard's taxonomy [9]. The fourth and final dimension highlights the payload or effects involved. Within each dimension various levels of information are provided to supply attack details. Hansman et al. mentioned the need of future work to improve classifying blended attacks, which is a limitation within their taxonomy. Another limitation is the lack of vulnerability information, which prohibits capturing information to aid in protecting a system from attacks.

Mirkovic and Reihner [10] offer a comprehensive taxonomy of Distributed Denial of Services (DDoS) attack and defense mechanisms in aim to classify attacks and defense strategies. This research highlight features of attack strategies, where the strategies are imperative in devising countermeasures. Mirkovic and Reihner's taxonomy of DDoS attacks is categorized by Degree of Automation, Exploited Weakness, Source Address Validity, Attack Rate Dynamics, Possibility of Characterization, Persistent Agent Set, Victim Type, and Impact on Victim. These categories are used to examine the exploitation, the victim impact, and characteristics with exploiting a DDoS attack. In addition to classifying DDoS attacks, Mirkovic and Reihner developed a taxonomy of DDoS defenses consisting of Activity Level, Cooperation Degree, and Deployment Location. The combination classifying DDoS attacks and defenses within a taxonomy provides communication of threats to foster cooperation between researchers for discussing solutions.

Lough [8] proposed an attack-centric taxonomy called VERDICT (Validation Exposure Randomness Deallocation Improper Conditions Taxonomy). Lough focuses on four major causes of security errors: Improper Validation, Improper Exposure, Improper Randomness, and Improper Deallocation. He labels these four characteristics with a prefix of "Improper" with attacks being thought of as improper conditions. Validation refers to improperly validating or unconstrained data, which also includes physical security. Exposure involves the improper exposure of information that could be used directly or indirectly for the exploitation of a vulnerability. Randomness deals with the fundamentals of cryptography and the improper usage of randomness. Deallocation is the improper destruction of information, or residuals of data, which also includes dumpster diving. He uses one or more of these characteristics to describe vulnerability within a system. Hansman and Hunt [6] describe Lough's taxonomy as lacking pertinent information that would be beneficial for knowledge bodies, such as CERT, to classify day-to-day attacks and issuing advisories. Lough's taxonomy lacks the classification to the type of attack, such as worms, Trojans, viruses, etc.

Howard [9] provides an incident taxonomy that classifies attacks by events, which is an attack directed at a specific target intended to result in a changed state. The event involves the action and the target. He highlights all steps that encompass an attack and how an attack develops. The attack consists of five logical steps an attacker performs to achieve an unauthorized result. Those steps are: tools, vulnerability, action, target, and unauthorized result. The tool refers to the mechanism used to perform the attack; the vulnerability is the type of exploit used to perform attack. The action refers to the method used by the attacker to perform the attack (i.e. Probe, Scan, Authenticate, etc.). The target is the intention the attack is attempting to compromise, and the unauthorized result is the change state caused due to the attack. Although Howard presents a useful taxonomy that provides an informative baseline for cyber intrusions, he lacks the details needed for thorough insight into the attack.

III. Our Proposed TAxonomy: AVOIDIT

A taxonomy defines what data is to be recorded and how like and unlike samplings are to be distinguished [1]. In developing a successful taxonomy, there are requirements that should be observed for universal acceptance. In this paper we analyze previous taxonomies and highlight valuable aspects that are needed to create a complete useful taxonomy [8,9]. These requirements include the following:

Accepted – builds on previous work that is well accepted.

Mutually exclusive – each attack can only be classified into one category, which prevents overlapping.

Comprehensible – clear and concise information; able to be understood by experts and those less familiar.

Complete/exhaustive – available categories are exhaustive within each classification, it is assumed to be complete.

Unambiguous – involves clearly defined classes, with no doubt of which class an attack belongs.

Repeatable – the classification of attack should be repeatable.

Terms well defined – categories should be well defined, and those terms should consist of established terminology that is compliant within the security community

Useful – the ability to be used and gain insight into a particular field of study, particularly those having great interest within the field of study.

Applying these requirements for a complete taxonomy, we propose AVOIDIT. AVOIDIT provides, through application, a knowledge repository used by a defender to classify vulnerabilities that an attacker can use. Fig. 1 provides an overview of our proposed taxonomy, which provides details to support comprehending each attack classification and how a variety of attacks are represented in each category.

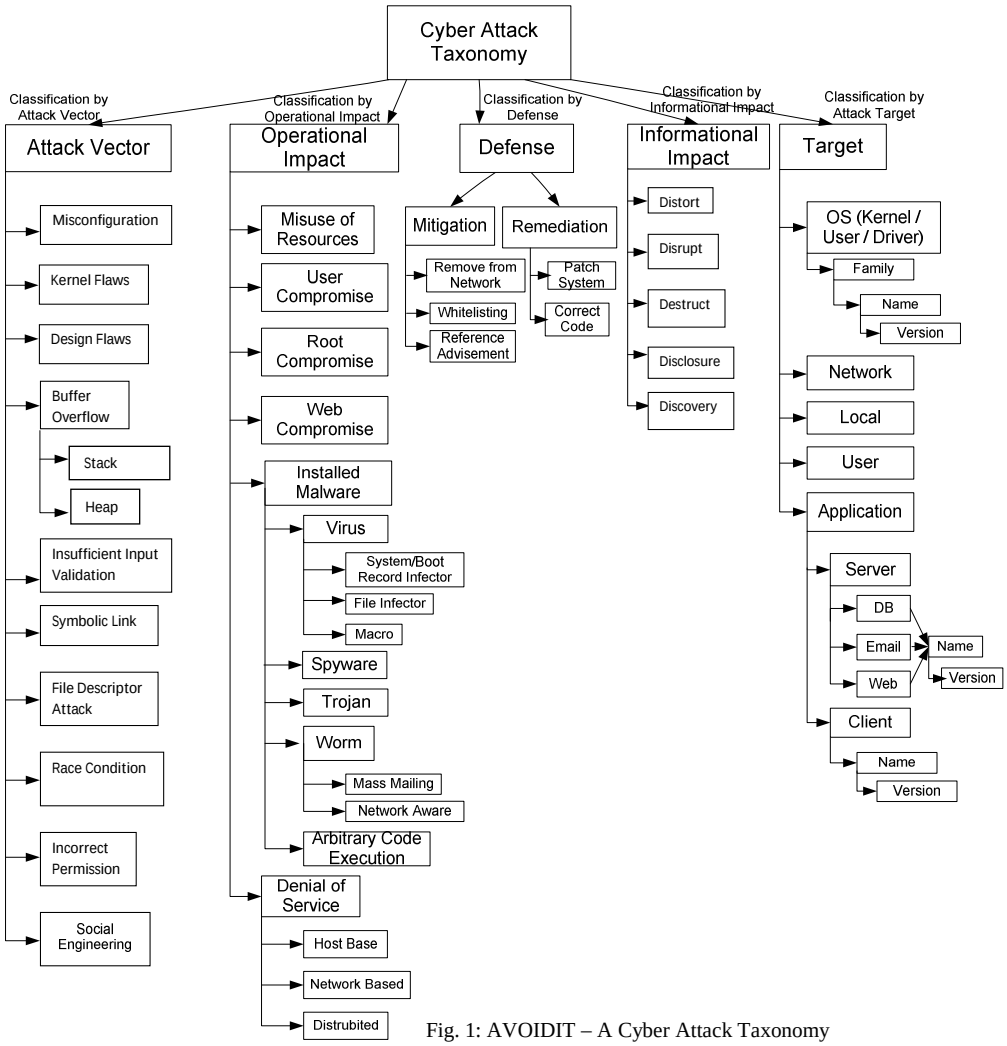

Fig. 1: AVOIDIT – A Cyber Attack Taxonomy

A. Classification by Attack Vector

When an attack takes place, there is a possibility it uses several vectors as a path to a full blown cyber attack. An attack vector is defined as a path by which an attacker can gain access to a host [7]. This definition includes vulnerabilities, as it may require several vulnerabilities to launch a successful attack. In this section we list several vulnerabilities that are used to render a majority of attacks.

- Misconfiguration - Misconfiguration - An attacker can use a configuration flaw within a particular application to gain access to a network or personal computer to cause a variety of attacks. Settings that are improperly configured, usually default settings, are an easy target for an attacker to exploit [5].
- Kernel Flaws - An attacker can use a kernel flaw within an operating system, which is the core code of an operating system, to gain certain privileges to exploit a vulnerability within the operating system.
- Buffer Overflow - Buffer overflow is caused when a piece of code does not adequately check for appropriate input length and the input value is not the size the program expects. Cowan [11] describes a buffer overflow when a buffer with weak or no bounds checking is populated with user supplied data. An attack can exploit a buffer overflow vulnerability leading to a possible exploitation of arbitrary code execution, often of privileges at the administrative level with the program running [5]. Buffer Overflow can occur in both stack and heap memory locations. A buffer overflow constitute majority of attacks [11]. A heap buffer overflow occurs in the heap data area, which is dynamically allocated by the application running [6].
- Insufficient Input Validation - A program fails to validate the input sent to the program from a user [5]. An attacker can exploit an insufficient input validation vulnerability and inject arbitrary code, which commonly occurs within web applications.
- Symbolic Links - A file that points to another file [5]. An attacker can exploit a symbolic link vulnerability to point to a target file for which an operating system process has write permissions.
- File Descriptor - A file that uses numbers from a system to keep track of files, as opposed to file names

[5]. Exploitation of a file descriptor vulnerability allows an attacker the possibility of gaining elevated privileges to program related files.

- Race Condition - Occurs when a program attempts to run a process and the object changes concurrently between repeated references allowing an attacker to gain elevated privileges while a program or process is in privilege mode [5].
- Incorrect File/Directory Permission - An incorrect permission associated to a file or directory consists of not appropriately assigning users and processes [5]. Exploiting this vulnerability can allow a multitude of attacks to occur.
- Social Engineering – The process of using social interactions to acquire information about a victim or computer system. These types of attacks provide quick alternatives in disclosing information to assist an attack that in normal circumstances may not be available.

B. Classification by Operational Impact

Classification by Operational Impact involves the ability for an attack to culminate and provide high level information known by security experts, as well those less familiar with cyber attacks. We provide a mutually exclusive list of operational impacts that can be categorized and concisely presented to the public.

- Misuse of Resources - An unauthorized use of IT resources [4]. We can extend this definition to consider any IT related function that require a certain privilege and those privileges are converted into an abusive action.
- User Compromise - A perpetrator gaining unauthorized use of user privileges on a host, as a user compromise [4].
- Root Compromise - Gaining unauthorized privileges of an administrator on a particular host [4]. We shall extend this notion slightly by including any elevated privileges above a normal user including administrative and/or root level privileges to a particular system.
- Web Compromise - A website or web application using vulnerabilities to further an attack [4]. An attack can occur through a web compromise, usually via cross site scripting or sql injection.
- Installed Malware - Exploiting some vulnerability an attack can be launched via user installed malware, whether user installed or drive-by installation. Installed malware can allow an adversary to gain full control of the compromised systems leading to the exposure of sensitive information or remote control of the host.
    - Virus - A form of installed malware, where Hansman and Hunt[6] describes a virus as a piece of code that will attach itself through some form of infected files, which will self-replicate upon execution of program. Types of viruses include boot record infectors, file infectors, and macros.
    - Spyware - A type of malware program that is covertly installed and infects its target by collecting information from a computing system without owner's consent.
    - Trojan - A benign program to the user that allows unauthorized backdoor access to a compromised system. A common way to introduce a victim into a multitude of attacks.
    - Worms – A self-replicating computer program. A considerable threat to the internet today. Worms do not require human intervention to propagate as it is a self-replicating program that spreads throughout the network. Worms include mass mailing and network aware worms.
    - Arbitrary Code Execution - Involves a malicious entity that gains control through some vulnerability injecting its own code to perform any operation the overall application has permission [13].
- Denial of Service - Denial of Service (DoS) is an attack to deny a victim access to a particular resource or service, and has become one of the major threats and rated among the hardest Internet security issues [13]. In this section we will provide details into the types of DoS attacks.
    - Host Based - A Host based DoS aims at attacking a specific computer target within the configuration, operating system, or software of a host. These types of attacks usually involved resource hogs, aimed at consuming up all resources on a computer; crashers, which attempts to crash the host system [6].
    - Network Based - A Network based DoS targets a complete network of computers to prevent the network of providing normal services [13]. Network based DoS usually occur in the form of flooding with packets [6], where the network's connectivity and bandwidth are the target [13].
    - Distributed - A Distributed Denial of Service (DDoS) is becoming more popular as an attacker's choice of DoS. A distributed denial of service uses multiple attack vectors to obtain its goal [10].

C. Classification by Defense

We extend previous attack taxonomy research to include a defense classification. In this section we highlight several strategies a defender can employ to remain vigilant in defending against pre- and post- attacks. We provide the possibility of using both mitigation and remediation when

classifying attack defenses, as an attack could be first mitigated before a remediation can occur.

- Mitigation - Prior to vulnerability exploitation or during an attack, there are several steps a defender can use to mitigate damage an attack has caused, or has the potential to cause. An example can involve an installation of a worm that propagate over the network, one instance could be to remove a set of hosts from the network and route traffic, while the administrator works on removal of the worm. Mitigation involves lessening the severity of the attack.
    - Remove from Network - The ability of an administrator to remove infected hosts preventing further damage. As the example described above, a particular worm may reside in a network and begins propagation.
    - Whitelisting - A list of permissible connections that are known to the defender. An attack could be directed at a particular software, which may reside on predetermined port.
    - Reference Advisement - Notes provided by the defender to mitigate an attack, or a vulnerability/vendor database reference number used to alleviate a vulnerability or attack.
- Remediation - In the presence or prior to vulnerability exploitation, there are resolution steps that are available to a defender to prevent an attack. Remediation would involve taking the appropriate steps to correct the situation prior to or during an exploitation.
    - Patch System - Applying patches the vendor has released due to some vulnerability within software in use. When a vulnerability or attack is present, on various cases, a defender fails to utilize the patches a vendor provides.
    - Correct Code - Steps within an organization to release a code patch to a specific application that will close the potential for an attacker to exploit.

D. Classification by Informational Impact

An attack on a targeted system has potential to impact sensitive information in various ways. A committed resource must be able defend information warfare strategies in an effort to protect themselves against theft, disruption, distortion, denial of service, or destruction of sensitive information assets [12]. In this section we classify an attacks impact, or the effect on information and define the criteria used.

- Distort - A distortion in information, usually when an attack has caused a modification of a file. When an attack involves distort, it is a change to data within a file, or modification of information from the victim [4].
- Disrupt - A disruption in services, usually from a Denial of Service. When an attack involves disrupt, it is an access change, or removal of access to victim or to information [4].
- Destruct - A destruction of information, usually when an attack has caused a deletion of files or removal of access. Destruct is the most malicious impact, as it involves the file deletion, or removal of information from the victim [4].
- Disclosure - A disclosure of information, usually providing an attacker with a view of information they would normally not have access to. Kjaerland [4] describes disclosure as unauthorized disclosure of information, with the possibility of leading to other compromises.
- Discovery - To discover information not previously known. For example, when a scanning tool probes for information, the information discovered can be used to launch an attack on a particular target.

E. Classification by Attack Target

Various attacks target a variety of hosts, leaving the defender unknowingly susceptible to the next attack.

- Operating System (Kernel / User / Driver) - Responsible for the coordination of activities and the sharing of resources of a computer. An attack can be formulated to target vulnerabilities within a particular operating system.
- Network - Target a particular network or gain access through a vulnerability within a network or one of the network protocols [6].
- Local - An attack targeting a user's local computer.
- User - An attack against a user is an attack to retrieve a user's personal information.
- Application – An attack towards specific software. An application can be either client or server. A client application is software that is available to aid a user performing common tasks. A server application is software designed to serve as a host to multiple concurrent users.

IV. TAXONOMY COMPARISON

In this section we use previous taxonomies described in Section 2 to compare AVOIDIT with past computer attacks and vulnerabilities. This section will highlight how our cyber attack taxonomy successfully captures vulnerability attack information and provide a defender with countermeasures that can be efficient in preventing or assuaging successful attacks.

A. SQL Slammer

This section provides details into the SQL Slammer worm. Slammer was able to perform 55 million scans per second and compromised ninety percent of vulnerable hosts in 10 minutes [3]. Table 1 classifies the SQL Slammer worm.

Table 1. Slammer Attack Classification

LOUGH

Name	Improper Validation	Improper Exposure	Improper Randomness	Improper Deallocation
Slammer	X	X		

HOWARD

Name	Tools	Vulnerability	Action	Target	Unauthorized Result
Slammer	Script	Configuration, Design	Prob, Modify	Network	Corruption of Information

HANSMAN

Name	1st Dimension	2nd Dimension	3rd Dimension	4th Dimension
Slammer	Network-Aware Worm	MS SQL Server 2000	CAN-2002-0649	Stack Buffer Overflow & UDP packet flooding DoS

AVOIDIT

Name	Attack Vector	Operational Impact	Informational Impact	Defense	Target
Slammer	Misconfiguration	Installed Malware: Worm: Network Aware	Discovery	Mitigation: Whitelisting CAN-2002-0649	Network
Slammer	Buffer Overflow	Installed Malware: Worm: Network Aware	Distort	Remediation: Patch System	Application

In table 1 Lough's taxonomy is too general to provide useful information in describing the attack; Howard's taxonomy provides preliminary information. Hansman and Hunt's taxonomy is able capture more detail in comparison to Howard. Our taxonomy provides information on what caused the worm infection, and possible defense strategies a network administrator can use to reduce the malware's ability to further propagate and cause damage. Using AVOIDIT, if the first insertion was alleviated, the Slammer worm would not be able to spread.

B. Microsoft RPC Stack Overflow

In 2008, a Windows Server service Remote Procedure Call (RPC) stack buffer overflow vulnerability [14] was exploited and is currently "in the wild". This RPC service provides print support and network pipe sharing were other users were able to access services over a network. The notable Conficker or Downadup attacks use these vulnerabilities to perform attacks on vulnerable systems. Table 2 classifies the RPC buffer overflow.

Table 2. RPC Stack Overflow Classification

LOUGH

Name	Improper Validation	Improper Exposure	Improper Randomness	Improper Deallocation
MS RPC Stack Overflow	X	X		

HOWARD

Name	Tools	Vulnerability	Action	Target	Unauthorized Result

MS RPC Stack Overflow	**Script**	**Design**	**Modify**	**Process**	**Increased Access**

HANSMAN

Name	1st Dimension	2nd Dimension	3rd Dimension	4th Dimension
MS RPC Stack Overflow	**Stack Buffer Overflow**	**Windows Server**	CVE-2008-4250	**Corruption of Information**

AVOIDIT

Name	Attack Vector	Operational Impact	Informational Impact	Defense	Target
MS RPC Stack Overflow	**Buffer Overflow: Stack**	**Installed Malware: ACE**	**Distort**	**Mitigation:** Reference Advisement VU#827267 **Remediation: Patch System**	**OS: Windows Server**
Gimmiv.A	**Buffer Overflow: Stack**	**Installed Malware: Trojan**	**Disclosure**	**Mitigation:** Reference Advisement Microsoft **Remediation: Patch System**	**OS: Windows: Server**
Conficker	**Buffer Overflow: Stack**	**Installed Malware: Worm**	**Disrupt**	**Mitigation:** Reference Advisement Microsoft **Remediation: Patch System**	**OS: Windows: Server, 2000, XP**

Classifying the buffer overflow vulnerability using Lough or Howard's taxonomy, we are unable to view the details, and unable to aid in defending against the vulnerability exploit. Using Hansman and Hunt's taxonomy, we may have been able to classify the attack, but the variations of the vulnerability the various attacks exploited are not present. With this particular vulnerability exploitation, you can view AVOIDIT as being able to thoroughly classify the vulnerability, potential blended attacks, and attack variations that specifically exploited the Windows buffer overflow vulnerability.

V. AVOIDIT CLASSIFICATION STRUCTURE

In this section we were able to classify a multitude of vulnerabilities and attacks. AVOIDIT benefits from the ability of being able to classify attacks in a tree-like structure, providing the ability to classify the allusive blended attack. Predecessors [4, 6] state that providing a tree-like structure is a solution to solving the blended attack, but claim this particular structure can become unorganized. We provide our taxonomy in a tree-like structure to successfully classify common vulnerabilities and cyber attacks to provide defenders with the needed information to defend their networks. Table 3 provides insight into how a searchable schema can be obtain we classify attacks using a tree-like structure, which enable a searchable schema. By using a parent-child relationship, AVOIDIT is able to display how multi-staged attacks can be captured, classified, and disseminated.

Table 3. Cyber Attack Classifications Structure

ID	Parent	Name	Attack Vector	Operational Impact	Defense	Informational Impact	Target
001		Slammer	Misconfiguration	Worm:NetworkAware	Mitigation: Whitelisting Remediation : Patch System	Discovery	Network

002	001	Slammer	Buffer Overflow	Installed Malware: Worm: NetworkAware	Remediation : Patch System	Distort	Application
003		Zotob	Buffer Overflow	Installed Malware: Worm	Remediation : Patch System	Distort	OS
004	003	Zotob	BoF: Stack	Installed Malware: Worm	Remediation : Patch System	Distort	Local
008		SamyXSS	Design Flaw	Web Compromise	Remediation : Correct Code	Disrupt	User
009		DebianAdmin	Kernel Flaw	Root Compromise	Remediation : Patch System	Disclosure	OS
010	009	DebianAdmin	Kernel Flaw	DoS	Mitigation: RA	Distort	OS
011		Yamanner	Social Engineering	Web Compromise	Mitigation: RA	Disclosure	Application: Server: Email
012	011	Yamanner	Design Flaw	Installed Malware: Worm: MassMailing	Mitigation: RA	Disrupt	User
013		MS RPC Stack Overflow	Buffer Overflow: Stack	Installed Malware: ACE	Mitigation: Reference Advisement VU#827267 Remediation: Patch System	Distort	OS: Windows Server
014	013	Gimmiv.A	Buffer Overflow: Stack	Installed Malware: Trojan	Mitigation: RA Microsoft Remediation: Patch System	Disclosure	OS: Windows: Server
015	013	Conficker	Buffer Overflow: Stack	Installed Malware: Worm	Mitigation: RA Microsoft Remediation: Patch System	Disrupt	OS: Windows: Server, 2000, XP

VI. AVOIDIT Applied in a Network

In this section we show how AVOIDIT can be used within cyber security to support a defender against malicious attackers.

AVOIDIT is intended to be used in multiple aspects of a network defense policy. It can be used to store event notifications within a database to educate administrators of attack frequency. The network administrator can also use an AVOIDIT organized knowledge repository in order to locate strategies that are appropriate for securing their network against vulnerabilities that can be exploited and used for unauthorized access. AVOIDIT used in a network defense strategy can improve the overall level of security. Our taxonomy can be used by applications that can offer a multitude of functions. The most obvious of these is that the taxonomy can be used to provide a defender with information related to the commonality, frequency, and vendor response pertaining to an event in which a vulnerability was exploited. This information will then be used to identify and implement defense measures. Previous taxonomies in Section 2 lack the structure of useful information to classify attacks through vulnerabilities that can be used in an application to assist a defender against an attack. Our taxonomy provides a more apparent approach to educate the defender on possible cyber attacks using vulnerability details. AVOIDIT will be used in a future game theoretic defense system to capture vulnerability information to provide a network administrator with a solution when defending against cyber attacks [15]. Until now, previous attack taxonomies have not been applied in a defense model, thus through application, our taxonomy presents a better approach in capturing and disseminating valuable information in defending a network against cyber attacks.

VII. AVOIDIT Limitations

Attacks have become increasingly present in the cyber world, and being able to provide the ability to prevent all attacks is extremely difficult. In this section we will highlight some of the limitations of AVOIDIT.

A. Lack of Defense Strategies

The defense strategies in our taxonomy present a defender with an appropriate starting point to mitigate and/or remediate an attack. The plausible defenses are enormous, so the proposed taxonomy provides a high level approach to cyber

defense. Although AVOIDIT is extensible, more research is needed to provide an exhaustive list of possible defense strategies for each vulnerability exploited.

B. Physical Attack Omission

Physical attacks are an important aspect in achieving security. While it is necessary to understand physical attacks, our proposed taxonomy focuses on cyber attacks. Further research can be done to include the physical aspect of cyber security, which may include the end hosts of an attack.

VIII. CONCLUSION

This paper provides a cyber attack taxonomy that enhances the cyber security industry. AVOIDIT will classify attacks by attack vectors, operational impact, defense, informational impact, and target. This classification scheme will aid a defender in protecting their network by providing vital attack information. It is presented in a tree-like structure to neatly classify common vulnerabilities used to launch cyber attacks.

We are aware of the possibility of new attack manifestation, therefore AVOIDIT could be extended to include new categories within each classification. AVOIDIT will provide a defender with the appropriate information to make an educated decision in defending against cyber attacks. Creative approaches to defending attacks will become available and providing an extensible taxonomy able to capture new defenses is imperative to defense. We believe AVOIDIT provides a foundation for the cyber security community and provide the ability to continuously grow as attacks and defenses become more sophisticated. In future work, to build a Game Theoretic Defense System, we will investigate the applicability of AVOIDIT in determining the action space of the attacker [15].

ACKNOWLEDGMENT

This work is supported by the Office of Naval Research (ONR) under grant N00014-09-1-0752.

REFERENCES

[1] Landwehr, Carl E., Bull, Alan R., McDermott, John P., Choi, William S., “A Taxonomy of Computer Program Security Flaws, with Examples”. ACM Computing Surveys, 26,3 (Sept. 1994).

[2] S. Noel, S. Jajodia, B. O’Berry, M. Jacobs, “Efficient Minimum-Cost Network Hardening via Exploit Dependency Graphs,” in Proceedings of the 19th Annual Computer Security Applications Conference, Las Vegas, Nevada, December 2003.

[3] David Moore, Vern Paxson, Stefan Savage, Colleen Shannon, Stuart Staniford, and Nicholas Weaver. Inside the slammer worm. In IEEE Security and Privacy, volume 1, 2003.

[4] Kjaerland, M., “A taxonomy and comparison of computer securityincidents from the commercial and government sectors”. Computers and Security, 25:522–538, October 2005.

[5] Scarfone, K., Souppaya, M., et al., “Technical Guide to Information Security Testing and Assessment”. NIST (Sept. 2008) http://web.nvd.nist.gov/view/vuln/detail?execution=e7s1

[6] Hansman, S., Hunt R., “A taxonomy of network and computer attacks”. Computer and Security (2005).

[7] Attack Vector. Retrieved June 19, 2009. http://searchsecurity.techtarget.com/dictionary/definition/1005812/attack-vector.html

[8] Lough, Daniel. “A Taxonomy of Computer Attacks with Applications to Wireless Networks,” PhD thesis, Virginia Polytechnic Institute and State University, 2001.

[9] Howard, John D. and Longstaff, Thomas A. “A Common Language for Computer Security Incidents,” Technical report, Sandia National Laboratories, 1998.

[10] Mirkovic, J., and Reiher, P. “A Taxonomy of DDoS Attack and DDoS Defense Mechanisms. In ACM CCR (April 2004).

[11] Cowan, C., F. Wagle, Calton Pu, S. Beattie, and J. Walpole. 1999. "Buffer overflows: attacks and defenses for the vulnerability of the decade." 2.

[12] Cronin, B. and Crawford, H. "Information warfare: Its Application in military and civilian contexts," Information Society, volume 15, pp. 257-263, 1999.

[13] C. Douligeris and A. Mitrokotsa, “DDoS Attacks and Defense Mechanisms: Classification and State-of-theart,” Comp. Networks, vol. 44, 2004, pp. 643–66.

[14] Porras, Phillip, Saidi, Hassen and Yegneswara, Vinod. An Analysis of Conficker's Logic and Rendezvous Points. Malware Threat Center. SRI International Technical Report, February 2009.

[15] Shiva, S., Dasgupta, D., Wu, Q. “Game Theoretic Approaches to Protect Cyberspace,” Office of Naval Research, Grant Number N00014-09-1-0752, 2009.