2016

A Framework for Selecting the Minimal Set of Preferred Responses to Counter Detected Intrusions

Maheedhar Gunasekharan
Iowa State University

A framework for selecting the minimal set of preferred responses to counter detected intrusions.

by

Maheedhar Gunasekharan

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Computer Science

Program of Study Committee:
Samik Basu, Major Professor
Johnny Wong
Wensheng Zhang

Iowa State University

Ames, Iowa

2017

DEDICATION

I would like to dedicate this thesis to my mother Manimegalai and to my father Gunasekharan without whose support I would not have been able to complete this work. I would also like to thank my friends and family for their loving guidance and financial assistance during the writing of this work.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my thanks to those who helped me with various aspects of conducting research and the writing of this thesis. First and foremost, Dr. Samik Basu for his guidance, patience and support throughout this research and the writing of this thesis. His insights and words of encouragement have often inspired me and renewed my hopes for completing my graduate education. I would also like to thank my committee members for their efforts and contributions to this work: Dr. Johnny. S Wong and Dr. Wensheng Zhang. I would additionally like to thank Dr.Ganesh Santhanam, Chris Strasburg for their guidance throughout my thesis research.

ABSTRACT

Over the past decades, cyber attacks have grown in frequency as well as in sophistication. Often, they elude the counter-measures that are in place due to inadequate expert man-power that is necessary to manually deploy the correct responses and maintain systems being compromised. We present a decision support framework to aid in timely deployment and maintenance of effective responses when intrusive or malicious behavior is detected.

The support framework has two specific objectives: to identify the best set of responses given the knowledge of the attack and the system being protected; and to identify the minimal set of responses that must be deployed. While appropriateness of responses is of utmost importance to safeguard systems from attacks, minimality in the number of responses, an important factor from the deployment and maintainability perspective, has often been discarded. Our framework leverages National Vulnerability Database as a source for information about the attacks, relies on the pre-specified expert knowledge about the responses that can adequately stop attack and takes into considerations the impact of an attack as well as responses on the system being protected in terms of well-studied CIA (Confidentiality, Integrity and Availability) vector.

We utilize Trade-off Enhanced Conditional Preference Network (TCP-net) to qualitatively represent and reason about the CIA priorities of the expert and model the problem of identifying minimal set of most effective responses into a search problem. The choice of TCP-net stems from the fact that the CIA priorities are typically qualitative in nature and it has been proven that quantification of priorities that are inherently qualitative can result in incorrect and often unexplainable results due to seemingly small perturbations in quantitative measures. Our TCP-net based computation can generate provably optimal solution where optimality corresponds to minimality of selected responses. While optimality is an important factor, the necessity for computing the solution efficiently cannot be overstated, particularly in the context where timeliness in response deployment is equally important. We investigate and evaluate several

heuristics with the goal of searching part of the potentially large solution space and compute a solution that is "close" to the optimal solution. We discuss the relative advantages and disadvantages of each heuristic, and present a specific one that is efficient in computing the optimal solution.

CHAPTER 1. INTRODUCTION

Over the past few decades we have seen cyberattacks grow sophisticated and elude the countermeasures that are in place. Although numerous efforts are being taken to strengthen our defenses, new attacks emerge every day that find loopholes in the defenses. In order to build an impenetrable cyber defense system, we need to have advanced intrusion detection and response mechanisms in place. Especially the intrusion response systems needs to be well adept and robust in responding to an attack that compromises the system. We propose an intrusion response mechanism that takes heterogenous distributed information coupled with up-to-date information on attacks from national vulnerability databases (NVD (2007)) to provide the most efficient responses. We also get CIA (Confidentiality, Integrity, Availability) triad preferences from the security administrators so that we can provide responses that uphold the interests of the organisation.

1.1 Challenges & Objectives

One of the challenges in selecting responses stems from the fact that detections are often imprecise (Axelsson (2000)) , i.e., an intrusive or anomalous behavior can be attributed to multiple attacks and, therefore, to the exploitation of multiple weaknesses. As a result, a large number of responses may be considered as valid and appropriate for the detected intrusive behavior. While it is important for security critical system to respond to all possible attack scenarios (Foo et al. (2005)), automated deployment is not likely to be effective because deployment of unnecessary responses can also harm the system in the same way as an attack. For instance, responses may unnecessarily update certain configurations or take away privileges, thus impacting the useful and normal behavior of the system leading to DOS (Toth and Kruegel (2002)).

While automation in deployment may be unwise, manual intervention in selecting responses even with expert knowledge can be time-consuming and overwhelming making it impractical for systems where fast and effective response deployment is a necessity. *Our objective is to develop an intrusion response system that would take into account the organization's Confidentiality, Integrity and Availability preferences and provide intuitive and justifiable guidance in response selection that would uphold those preferences to the best possible extent, at real time.*

1.2 Proposed framework

There are three primary objectives for our work.

1. *Ensure protection of the system from attacks*: The deployed responses must prevent the attack from having any negative impact on the system.

2. *Ensure minimal impact on the system due to respones deployments*: A response is essentially an action that limits the capability of the system which is vulnerable to attack . Hence automated responses present the risk of negatively impacting the system since they are deployed without any assessment of its effect on the system. We incorporate a decision support framework that ensures the responses deployed have the least impact on the CIA priorities of the organization.

3. *Ensure minimal overhead in administering responses by deploying minimal set of responses*: Cost of deployment is a factor that is often overlooked in automated intrusion response systems. Our heurisitic aims to deploy minimal number of responses which can effectively prevent the impact of a given set of weaknesses without compromising the system CIA preferences.

We present a framework that provides sound guidance for administering a minimal set of responses that are likely to address the detected intrusions and will least impact the system in a negative fashion. This decision framework has three primary components. First, it requires the input from the intrusion detection system in terms of weaknesses (w_i's) being exploited. For this, we use the National Vulnerability Database (NVD), a well-documented source of

attacks, vulnerabilities and impact of attacks. Second, the framework relies on the mapping of responses (r_j's) that can address the exploitation of the weakness ($w_i \mapsto \{r_{i1}, r_{i2}, \ldots, r_{in}\}$). And finally, the framework incorporates the knowledge of system preferences in terms of CIA-impact (Confidentiality, Integrity and Availability) and the corresponding influence on CIA by the detected weaknesses (available from NVD) and available responses (typically provided by system administrators) to assess the quality of responses being selected.

To illustrate the complexities in deciding the correct set of responses, consider a system where protecting confidentiality and integrity is more important than ensuring availability. Assume that the detection mechanism identified the exploitation(s) of two weakness w_1 and w_2. The exploitation of weakness w_1 may result in severe compromise of confidentiality and no other impact. The exploitation of weakness w_2, on the other hand, may result in moderate compromise of both confidentiality and integrity. Each weakness is associated with three effective responses: r_{i1}, r_{i2}, r_{i3} for $i \in (1, 2)$, each with its own value of negative impact on the system. For instance, r_{11} may have severe impact on the integrity of the data, while r_{12} may have moderate impact on the integrity and severe impact on the availability. The goal is to identify the best possible set of responses that can address exploitation(s) of both the weaknesses in a timely fashion. The trade-off in the selection will come into play due to the system requirement. In this example, availability will have the least priority compared to confidentiality and integrity. Even with such a small set of weaknesses and responses, the job can be unmanageable and error-prone for experts. Furthermore, the task becomes daunting as it comes with the added requirement to not get out-paced by the detection mechanism, which is typically automated.

We measure the impact of the weakness/response in terms of CIA. The important aspect of this is the measuring unit. The impact measures are not quantitative. It has been well-studied and recommended to not rely on quantitative measures for CIA vector. As a result, qualitative measures are considered, which may not produce single "best" solution but any optimal solution can be logically justified.

Finally, in addition to the quality of the selected responses in terms of their ability to stop intrusions without negatively impacting the system, it is also important to deploy minimal

number of responses. Minimality brings in new challenges in the section process. Often minimality can be in odds with the quality. Furthermore, minimality is a global property, which can be correctly computed only after considering all possible choices in the solution, which makes it difficult to ensure minimality in an efficient fashion.

1.3 Contribution

In light of the above, our work **contributes** to the response selection process in the following way:

1. We present Trade-off enhanced Conditional Preference network (TCP-net) to capture preferences over CIA attributes. The network unambiguously represents qualitative preferences between the attributes and the attribute values.

2. We model the method for identifying the responses as a search process and present 4 search heuristics that take into consideration the detected exploits, the mapped responses and the preference specifications to identify the deployable responses.

   The heuristics do not explore the entire search-space for all possible set of responses (which is prohibitively expensive); instead they rely on local and history information in a search tree to identify the best extensions of the search path. We characterize the quality of the heuristic-results in terms of the difference between the heuristic-results and solutions resulting from exploring the entire search-space.

3. We present detailed experimental evaluation by considering different exploit scenarios. We show that the order in which the detected exploits are considered for response selection does not impact the final result, which makes our method appropriate in practice, where the response selection must work as and when exploits are detected.

4. We implement our method to realize a decision framework. Our framework is modular and can easily interface with different intrusion detection mechanisms that can identify the exploits as specified in NVD. Furthermore, being based on sound preference logic such as TCP-net, our framework can be also used to logically justify the selection of responses

in every scenario. This is important in two aspects: for postmortem analysis and for understanding and possibly updating preferences and response-mappings for future use.

1.4 Organization

The rest of the paper is organized as follows. In Section 2, we talk about the intrusion response mechanisms that are currently used, their advantages and disadvantages. We also hypothesize why our approach would overcome those disadvantages. In Section 3, we present our preference formalism including the dominance relation and analyze its properties. We present four algorithms for identifying the most preferred compositions and discuss their properties. We discuss the complexity of each of our algorithms. In Section 4, we present results of experiments that we performed to compare our algorithms in terms of the quality of solutions produced, performance and efficiency. In Section 5, we summarize our contributions and discuss the related and future work in this area.

CHAPTER 2. REVIEW OF LITERATURE

Selecting and deploying responses in the context of detected intrusion has remained one of the active areas of research both in academia and industry. We will discuss the contribution of the existing work in light of quality (adaptability, robustness) and degree of automation in response selection and deployment (Section 2.1). We will also present the need for preferences to soundly justify the selection process. In particular, we will discuss the appropriateness of qualitative preferences for response selection and present an overview of qualitative preferences (Section 2.2).

2.1 Intrusion response systems

Security frameworks rely on two specific mechanisms to protect systems from being compromised– intrusion detection and intrusion response. The former focuses on detecting the type, voracity and the cause of the intrusion (Kabiri and Ghorbani (2005); Lazarevic et al. (2005)), while the latter deals with identifying the possible responses to the attacks (Stakhanova et al. (2007); Shameli-Sendi et al. (2012)). In some frameworks, detection and response are tightly coupled to provide efficient feedback to the detection mechanism about the result of deployment of some responses and possibly re-assess the type and the cause of the intrusion (Rowe et al. (1999)). In others, the coupling is intentionally left loose to avoid incorrect over-fitting and to allow quick plug-n-play (Papadaki and Furnell (2006); Lewandowski et al. (2001)). We present a framework for response selection that can be used in conjunction with any detection mechanism that can identify information about the weaknesses (configurations, privileges) of the system being exploited by the detected attack(s).

Table 2.1 Some of the existing Intrusion response systems

Reference	IRS mechanism
Kanoun et al. (2010)	access control policies
Jahnke et al. (2007)	attack path of intrusion
Kheir et al. (2009)	minimal cost of response
Thames et al. (2008)	firewall
Zhang et al. (2003)	logs the attack data for forensics

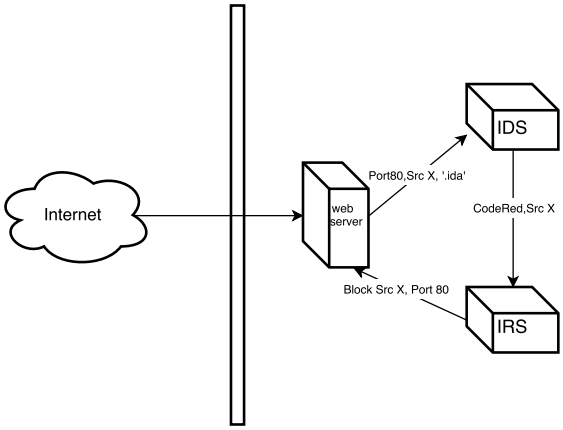

Figure 2.1 Code red attack is blocked successfully

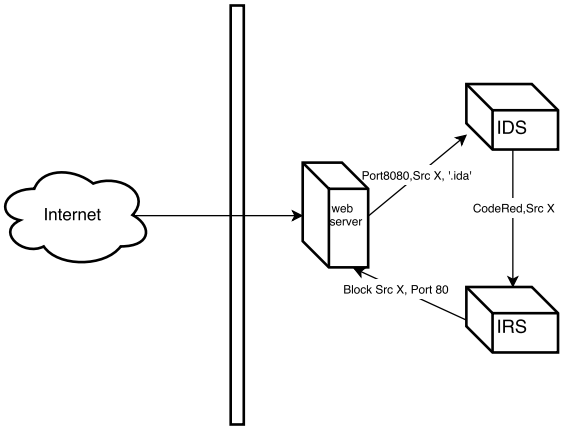

Figure 2.2 Code red attack succeeds

Surveys such as Stakhanova et al. (2007); Shameli-Sendi et al. (2012); Anuar et al. (2010) present several techniques and the mechanims they employ to react to an attack. These are tabularized in Table 2.1 . Some systems try to find the source of the intrusion by employing several counter-espionage mechanisms. For example, Wang et al. (2002) injects a watermark into the backward connection of the intrusion, and wake up and colloborate with intermediate routers along the intrusion path to determine the source of intrusion and respond to it. Lewandowski et al. (2001) specify how an effective IRS framework should be designed by coordinating information from multiple sources to deploy an effective defense against fast and distributed information attacks. Though the above systems were effective in a number of cases, the potential and effectiveness of these systems to provide automatic, pre-emptive, pro-active responses (Stakhanova et al. (2007)) were limited.

Bass (2000) paved the way for how an IDS must be designed in order to be most efficient at detecting sophisticated attacks by making use of information from a variety of sensors and

databases. We feel that this design needs to be extended to IRS as well. Most of the IRS engines which work along with IDS does not leverage the complete heterogenous set of information available to IDS while providing responses. Instead they provide responses that are static mappings between identified malicious event and response. For example , Lets consider the scenarios in Figure 2.1 and Figure 2.2. When an IDS detects a codeRed attack, it sends a notification to the IRS that codeRed has been identified. The IRS responds by blocking all network traffic on port 80. This should work most of the time (works in the scenario shown in Figure 2.1). But if the codeRed was actually happening on a different port (Figure 2.2), then the response would have no effect. Though every information about an attack is be available to the IDS, it is not communicated to the IRS along with the alert notification. Because of this, current IRS systems though good at responding to known attacks lack the capability to respond to unknown and sophisticated attacks.

Responses can either be signal-oriented , those which are targeted to thwart the impact of the malicious signals or they can be resource-oriented, those that are aimed at protecting a resource from any attack. We can think of signal as the cause of an attack and resource being compromised as the effect of an attack. Its easier and sensible to prevent the cause from have any effect at all rather than stopping the effect. And also due of the complex nature of the cyber attacks these days, we beleive that a signal-oriented approach would be more efficient. Consider the following example. A CodeRed attack happens via a GET request which contains code that exploits a buffer overflow vulnerability in the indexing software in Microsoft's Internet Information Server(IIS).This vulnerability allows the worm to run code from within the IIS server. A signal-oriented approach in this scenario detects the malicious signal and blocks communication on that port. Whereas, a resource-oriented approach needs to know that the particular resource is being attacked in order to deploy countermeasures to safeguard the resource from that attack. But this largely depends on the granularity of the definition of resource. In order to effectively thwart the attack, the entity under attack should have been defined as a resource. In case of CodeRed resource is the IIS server. And this might differ from attack to attack and that makes it difficult to design a resource-oriented approach.

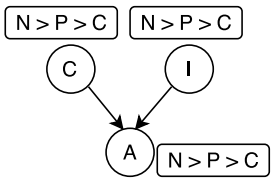

Figure 2.3 TCP-net

Unlike an IDS, the effectiveness of responding to an attack is not entirely dependant on the signals causing the attack but also dependant on the setup of the system environment. Hence we also use system specific knowledge of the organization's infrastructure while responding to the attacks. Through NVD we estimate how the attack would have manifested on a given infrastrucutre. We also get the CIA triad preferences from the security administrator in order to provide more appropriate mission sensitive responses. At the end of the process, we provide the security administrator of the organisation with a partial order of responses that will be ranked based on their effectiveness whilst preserving the preferences of the organisation.

2.2 Qualitative preferences

We measure the impact of the weakness/response in terms of CIA. The important aspect of this is the measuring unit. The impact measures are not quantitative. It has been well-studied and recommended to not rely on quantitative measures for CIA vector as (a) quantitative measures may incorrectly impose measurable degree of severities of impact and (b) seemingly small perturbations in the quantitative measures can result in a incorrect and often unexplainable results (Jansen (2009); Bartol et al. (2009)). As a result, qualitative measures are considered, which may not produce single "best" solution but any optimal solution can be logically justified.

To identify such an optimal set of responses, it is necessary to have formal framework to represent and reason with the preferences over values of the CIA attributes of the targeted weaknesses and responses, as well as relative importance over them. We use the qualitative preference formalism TCP-net Brafman et al. (2006) to represent and reason with the qualitative preferences of the admin.

Each node in the TCP-net is labeled with an attribute and annotated with the preference over its values (intra-attribute preference). The edges between the nodes denote relative im-

portance: the attribute at the destination node is relatively more important than the one at the source node.

2.2.1 Semantics of TCP-nets

The semantics for TCP-nets is based on the "ceteris paribus" interpretation of its statements Boutilier et al. (1999). In the context of preferences over CIA attributes, this interpretation is as follows. Consider two weaknesses (or responses) with CIA impacts represented by α and β respectively, each of which is a triple containing the values of C, I and A attributes (a triple is a generic term, and can be used to denote the CIA attributes of a weakness or a response). We refer to the value of the attribute X in α by $\alpha(X)$. β is said to be `preferred` to α, denoted by $\beta \succ \alpha$, if and only if the valuations of a subset of β's attributes can be changed to obtain another triple β', such that the following two conditions hold.

1. For some attribute, $X \in \{C, I, A\}$, $\beta(X) \succ_X \beta'(X)$; and for all attributes Y s.t. $Y \rhd X$, $\beta(Y) = \beta'(Y)$ or $\beta(Y) \succ_Y \beta'(Y)$
2. $\beta' \succ \alpha$ or $\beta' = \alpha$

Note that the above is a recursive definition of the preference relation $\succ$ between triples (Rule 2 requires β' to be preferred to α). Hence in general, checking whether β is preferred to α involves searching for the existence of a sequence of triples β'_i $(i = 1..n)$ such that $\beta \succ \beta'_1 \succ \ldots \succ \beta'_n = \alpha$.

In our example, the response `strengthenFirewall` is preferred to `blockUserAccount` because the valuation of `strengthenFirewall`, namely (N, P, P) can be first changed to β'_1 =(P, P, P) (confidentiality impact is changed from none to partial), and then again to β'_2 =(P, P, N), which is the valuation of `blockUserAccount`.

2.2.2 Non-dominated set

Given a set of CIA triples and TCP-net preferences over them, the semantics of TCP-net induces a partial ordering over the triples Brafman et al. (2006). This ordering can be represented as a directed graph over the triples, such that there is an edge from one

triple α to another β whenever $\beta \succ \alpha$. The set of most preferred triples with respect to the TCP-net is then the roots of this induced graph, and called the *non-dominated set* of triples. In other words, a triple β is in the non-dominated set if there is no triple α such that $\alpha \succ \beta$. For example, suppose that the admin wants to find the most-preferred response(s) from the set {`blockUserAccount`, `performForensics`, `temporaryShutdown`}. The preferences induced by the TCP-net in Figure 2.3 over this set are: (a) `temporaryShutdown` $\succ$ `performForensics`, and (b) `blockUserAccount` $\succ$ `performForensics`. Hence, the non-dominated set is {`temporaryShutdown`, `blockUserAccount`}.

2.3 Cyber attacks

Reports of threats predictions, (mca (2016); cnb (2015)) cite that ransomware is one of the most prominent and dangerous threats to both organizations and personal computers. Ransomware is an interesting example to look at because most IRS have failed to stop this attack. Ransomware has been so detrimental and elusive because detecting it would require monitoring of both system level and network level events. Therefore, In order to successfully respond to these threats the IRS engine needs to use information from various heterogeneous distributed agents and thereby provide more reliable IRS service. This is a important goal to be achieved for any security-decision systems which must identify,track and respond to complex multiple threats. For example, the input into such systems would consist of numerous distributed packet sniffers, log files, SNMP traps, user profile databases ,system messages and operator commands. This would aid in the IRS to serve more efficient and appropriate responses. New strains of ransomware are being released into internet every day which makes it increasingly difficult to respond to it. In order to be better equiped to handle such threats , we need to utilise some kind of shared data model which would allow the response engine to reason about events in a cooperative way. We know that multi-million dollar efforts are being taken by number of cybersecurity organisations worldwide to provide up-to-date information on attacks and vulnerabilities. We propose a federative data model for security systems that utilises such shared NVD databases to query and assert knowledge about security incidents and the context in which they occur.

We feel that such sophistication is necessitated by the complex and advanced nature of cyber attacks these days which always seems to be one step ahead of the cybersecurity efforts.

CHAPTER 3. METHODS AND PROCEDURES

The main motive of our work is to develop an autonomous intrusion response system that would take into account the preferences of the organization and deploy responses that would uphold these given preferences, and effectively counter the weakness at minimal cost. In order to acheive this, we need to be able to compare the preference values of a given entity(weakness or attack) and be able to optimally derive a set of responses that are minimal in size, best in terms of preferential quality and effectively stop a given set of attacks. In this chapter,we will dive deep into the methods we have used to get this desired result.

3.0.1 Illustrative Example.

Consider a system administrator (admin) in charge of securing a mission critical network. At any point in time, the admin observes a set of attacks on the network, and has to apply a set of responses to address the weaknesses exploited and mitigate potential adverse security impacts to the system and the contained data. We measure security impacts to a system in

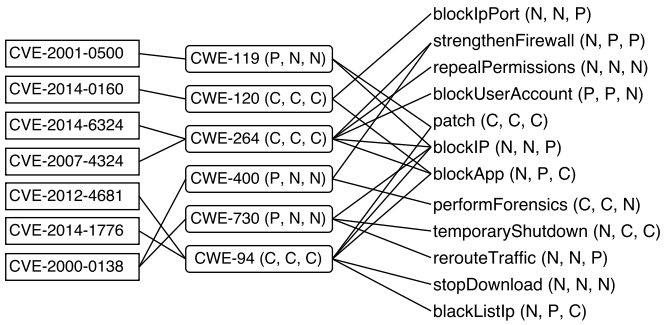

Figure 3.1 Example scenario showing a set of attacks observed (CVE's), the corresponding weakness (CWE's), and the responses that can address the respective weaknesses. The CIA impacts of weaknesses and responses are given in parentheses; N = none, P = partial, C = complete.

terms of the classical CIA model: confidentiality (C) is impacted when sensitive information is leaked to attackers, integrity (I) is impacted when the attacker is able to tamper with sensitive information, and availability (A) is impacted when the attacker can cause a kind of denial of service.

Each attack exploits one or more weaknesses in the system. The NVD database is an index of weaknesses and their adverse impacts on a system. In particular, NVD provides information on whether each weakness partially (P), completely (C), or does not at all (N) adversely impact the system with respect to the CIA attributes. Similarly, the admin may have at his disposal one or more responses to address each weakness. Note that applying a response may also adversely impact the system, as it may restrict certain operations by genuine users (e.g., blocking an IP will have an availability impact on the system). Hence, similarly to NVD provided data for weaknesses, the admin can associate each response with a CIA impact value of P, C, or N.

Consider the attack scenario in Figure 3.1, where an admin seeks a set of responses (subset from the right hand side) to address a set of attacks (left hand side) that exploit certain weaknesses (middle column) in the system. The CIA impacts of the weaknesses and responses are depicted alongside their names. For this example, the natural preferences over the impact values of C, I and A attributes are used – N $\succ_X$ P, P $\succ_X$ C, and the relative importance preferences stated by the admin are: $C \rhd A$ and $C \rhd I$, i.e., confidentiality is more important than availability and integrity. Multiple different sets of responses can be identified to address the attacks being detected. Graphically, a solution set consists of responses such that each attack has a path to some response in the set (Figure 3.1). The objective is to identify the solution that is most preferred and contains the smallest number of responses.

To identify such an optimal set of responses, it is necessary to have formal framework to represent and reason with the preferences over values of the CIA attributes of the targeted weaknesses and responses, as well as relative importance over them. We use the qualitative preference formalism TCP-net to represent and reason with the qualitative preferences of the admin.

The TCP-net representation of the admin preferences in this examples is shown in Figure 3.2.

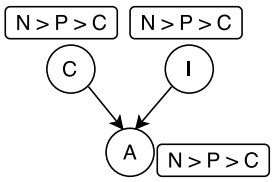

Figure 3.2 TCP-net

In addition to ordering individual responses per the TCP-net, it is necessary to order sets of responses, as a given attack scenario may demand multiple responses to be applied. For example, if CWE-119 and CWE-120 are both exploited in an attack, the admin can choose from four possible sets of responses, namely pairs with one element from each of the sets {`patch`, `blockIP`} and {`blockIpPort`, `blockApp`}.

We use the approach developed in Santhanam et al. (2011) to compare sets of attribute values, summarized as follows. First, we obtain the *aggregated valuation* of the set of responses with respect to C, I and A as a function of the impact values of the corresponding attribute in each constituent response. Specifically, we consider the worst valuation of attribute X amongst all the responses in the set as the valuation of X for the set. For example, the aggregate valuation of the set {`blockIP`, `blockApp`} is (N, P, C) because `blockApp` has the worst valuation for the CIA attributes, similarly, any response paired with `patch` has an aggregate valuation of (C, C, C). We then use the aggregated valuation of the sets of responses to compare and choose the most preferred according to the admin's preferences. Hence, in the above example, the set {`blockIP`, `blockIpPort`} would be the most preferred, which is preferred to the set {`blockIP`, `blockApp`}. The other two response sets containing `patch` would be the least preferred for the admin because they have the worst possible valuation. The concept of non-dominated set defined over individual triples can also be applied to sets of responses by comparing sets of responses with respect to their aggregated valuation. In this example, however, the non-dominated set is a singleton, i.e., {{`blockIP`, `blockIpPort`}}.

3.1 Preferential Selection and Minimization

Our primary objective is to select a set of responses for a given set of the detected attacks. The attacks result from the exploiting certain system configurations (weaknesses). The selection

process takes into consideration the system preferences in terms of CIA and obtains the set of responses that will not only mitigate the weaknesses but also will be most preferred as per the system preferences. As noted before, considering the preferences and priorities of the system is important, as there are many different ways in which weaknesses can be addressed, which can result in different sets of responses. Another important aspect in response selection is the number of responses necessary to address the detected attacks; smaller number of responses are likely to be preferred than larger.

Therefore, in terms of computation, our objective can be realized by (a) exploring the solution space where each solution is defined by the set of responses that can mitigate the weaknesses causing the detected attacks, and (b) identifying the ones that are most preferred in terms of the CIA preferences and size. Note that, the number of solutions in the solution space can be very large depending on the number of possible responses being considered; and there can be multiple valid solution that are equally preferred (or indistinguishable in terms of preferences).

It is immediate that exploring the entire solution space is impractical as the likely overhead in terms of time will be prohibitively large. Therefore, it is important to identify methods and heuristics that do not explore the entire solution space but are still capable in computing results "close" to the valid solutions. We will discuss 4 such heuristics and characterize them in terms of the quality of the results computed by them. The quality of a solution will capture its degree of difference with the best solution (computed by exploring the entire solution space).

3.1.1 Search Tree and Heuristics

In the following, we describe the basic outline for a generic heuristic method and then discuss the application of the heuristics. The solution space can be viewed as a tree of responses; we will refer to this as the *search tree*. At every level (except the root) of the search tree, we consider a weakness being addressed. For each weakness, there can be many responses, which represent the branching factor for that level. For instance, if at level 1, the weakness w_1 can be addressed by three different responses: r_{11}, r_{12}, r_{13}, then the number of branches from the root of the search tree is 3, each branch leading to a node corresponding to the response. Therefore, the nodes

in the tree are choice points and the choices are branches corresponding to the selection of a response. If at level 2, the weakness w_2 can be addressed by two responses: r_{21}, r_{22}, then for every branch of the level 1 (which represents the selection of a specific response), there will be two branches at level 2. A solution is one where at least one response has been selected in every level of the search tree. Figure 3.5 illustrates one such search tree.

There are three aspects in any heuristics that computes the solution in this search tree: (a) the order in which the weaknesses are considered, (b) the order in which the responses at different level are considered, and (c) the context in which these orders are considered in the selection. While the aspects (a) and (b) are immediate, the aspect (c) denotes the extent to which the selection at the i-level can be impacted by those made at $j \neq i$ levels.

3.1.1.1 `GreedyCIA`.

In this heuristic, the weakness are considered in the order they are detected to be exploited (in essence, in some random order). The best response with respect to CIA preferences are selected at each level. The primary reason for the greedy approach is efficiency. Once a choice is made at a particular level, other choices at that level or at levels above it are never explored.

3.1.1.2 `wCoverage▶rCIA`.

In this heuristic, the weakness are ordered in terms of the strength/coverage of the responses that can address them. That is, if the responses for weakness w_i (level i) can address n weaknesses, if the responses for weakness w_j (level j) can address m weaknesses, and if $n > m$, then i must be higher than j. The intuition behind such ordering is to increase the likelihood of selecting responses with higher coverage, thus minimizing the total number of responses in the solution.

The response selection utilizes the system preferences with respect to the CIA values. The heuristics relies on the history of selection (see aspect (c) above)—at every level an ordered list of partial solutions are maintained, and most preferred among the partial solutions is considered. Note that, the partial solutions in the ordered list may not address the same set of weaknesses. Further note that, each partial solution has a CIA value computed using worst

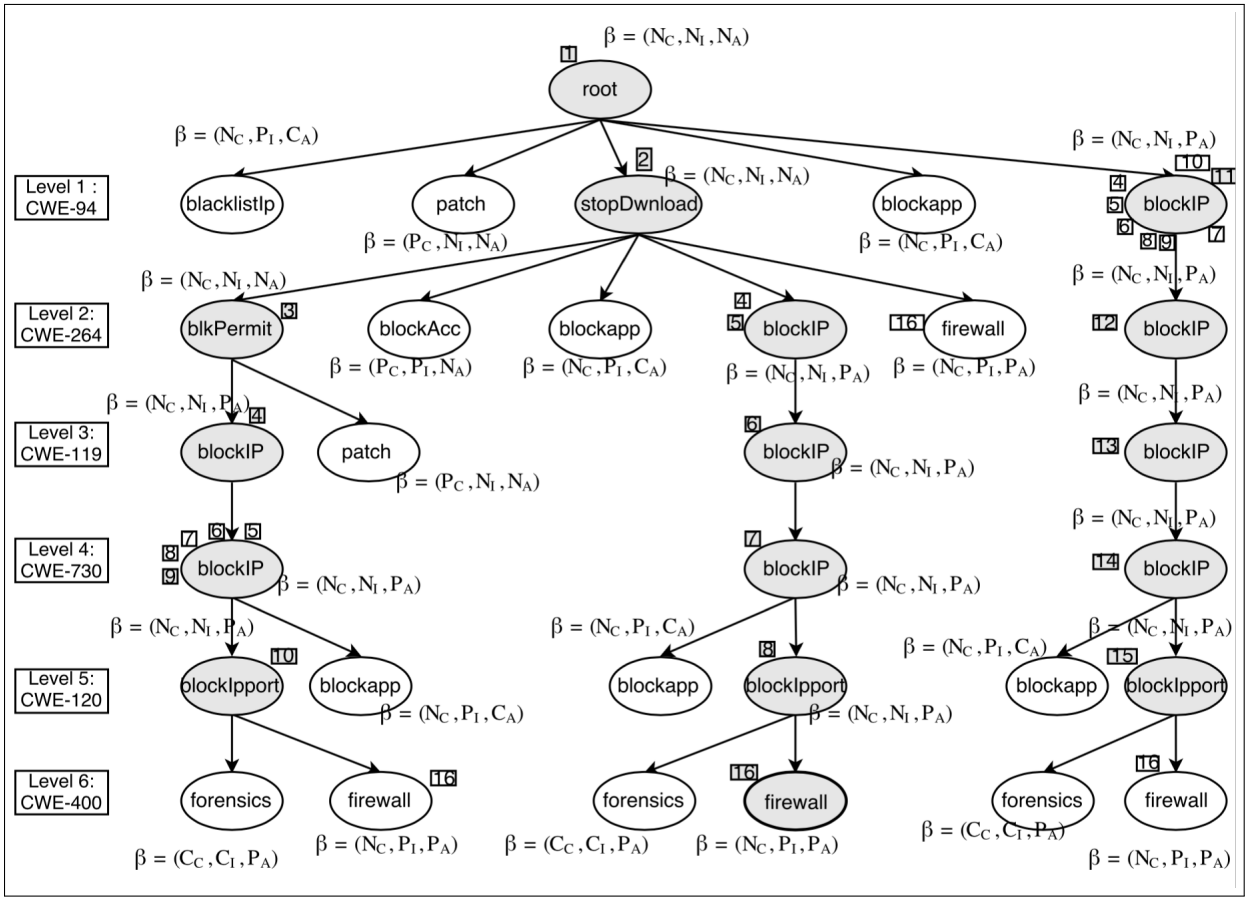

Figure 3.3 Result of applying `wCoverage▶rCIA`

valuation (see Section 2.2); we will denote this value as the β-value. If there are two partial solutions that have the same β-value or their β-values are indistinguishable with respect to the system preferences, then their mutual order is chosen randomly.

For instance, consider a situation where at level i, r_{i_1} and r_{i_2} are the possible choices and due to better CIA values r_{i_1} is chosen for further exploration in the search tree. At level $j = i + 1$, the choices are r_{j_1} and r_{j_2}, and the CIA values associated with these responses are such that the combined CIA value of the pairs r_{i_1}, r_{j_1} and r_{i_1}, r_{j_2} are both worse than the CIA value for r_{i_2}; and specifically, CIA value of r_{i_1}, r_{j_1} is worse that r_{i_1}, r_{j_2}. In this case, the order list of partial solutions at level j will have r_{i_2} proceeded by the pairs r_{i_1}, r_{j_1} followed by r_{i_1}, r_{j_2}, and search tree will explored by considering the choices r_{j_1} and r_{j_2} in the context of r_{i_2}.

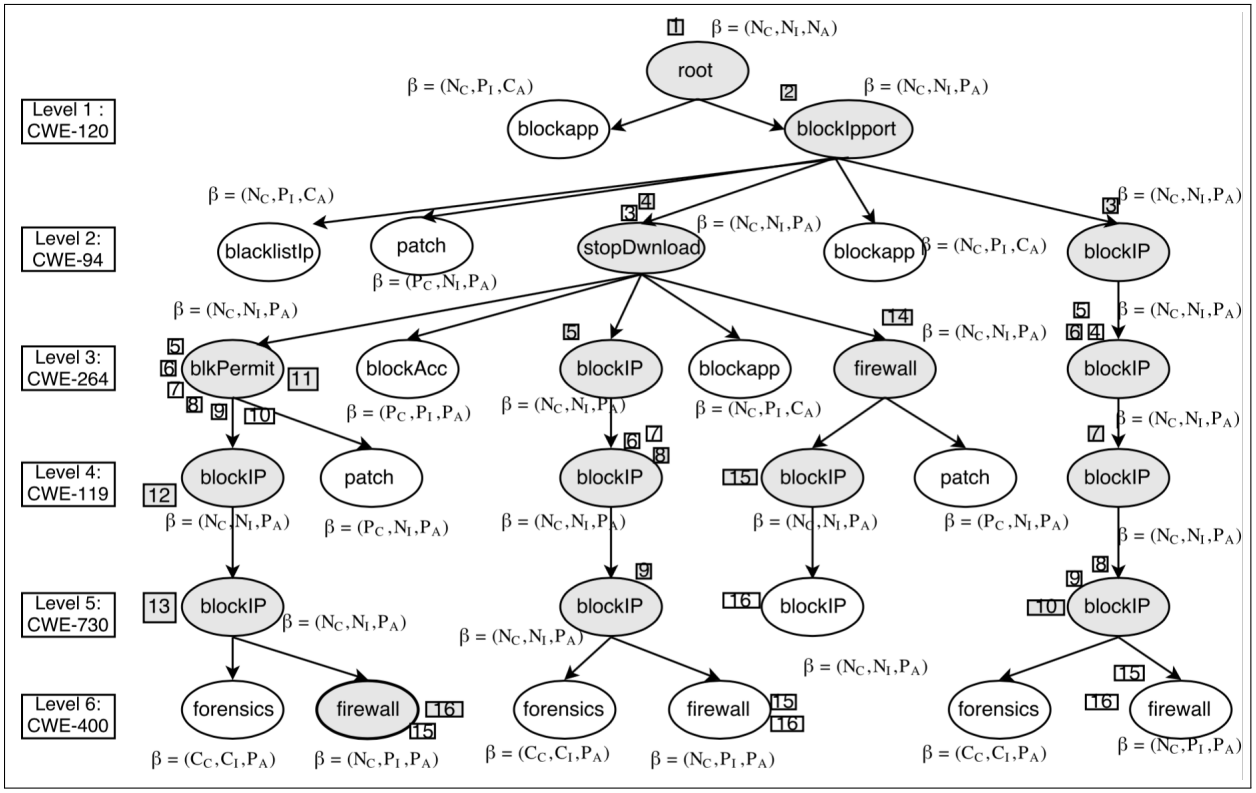

Figure 3.4 Result of applying `rCIA`

3.1.1.3 `rCIA`.

In this method, there is no specific order in which the weaknesses are considered. The responses are selected using the same process as in `wCoverage`▶`rCIA`.

3.1.1.4 `rCIA`▶`rCoverage`.

In this method, there is no specific order in which the weaknesses are considered. The responses are selected using the same process as in `wCoverage`▶`rCIA` with an augmentation to take into consideration the coverage aspect of each response. The augmentation involves the ordering of partial solutions that are indistinguishable (see Section 2.2) based on their β-values and preferences over them. In such a case, the partial solution whose coverage size is larger is placed higher in the order. Intuitively, this is likely to lead to smaller size solutions eventually.

Note that the coverage measure comparison of the responses is a context-sensitive information requiring the knowledge of weaknesses being considered. A response may have higher

coverage than another as the former is applicable in addressing larger number of weaknesses than the latter; but that does not imply that the former is a better choice than the latter in any context. For instance, consider the situation where r_1 can address w_1, w_2, r_2 can address w_3 and the weaknesses being detected is w_3.

In this heuristic, we assume knowledge of weaknesses that we have already considered in the search tree (history context) and also use the knowledge of weakness that will be considered in one step (look-ahead context). The reason for limiting the look-ahead context is to ensure that the response selection at each step does not necessarily require the complete knowledge of all weaknesses; thus making the heuristics applicable on-the-fly as and when the detection engine provides information about weaknesses.

It can proved that in all the proposed heuristics other than the `GreedyCIA`, the most preferred (or more precisely, one of the non-dominated) solution will be computed. Intuitively, the proof relies on the fact that the search path that can produce a preferred partial solution, will be always explored. When the CIA value of the responses in a partial solution in a search path falls below (in terms of system preferences) the CIA value of response in a partial solution in another path, the search progresses along the latter and stores the current search path. This technique is based on the algorithm proposed by Santhanam et al. Santhanam et al. (2008) for efficient composition of services using qualitative preferences. Note that, the heuristics (other than `GreedyCIA`) has different characteristics in terms of the size of the solution. We claim that `rCIA▶rCoverage` is best suited to obtain a solution that is most preferred and is likely to be the smallest. The following example illustrates the search process for `rCIA▶rCoverage`.

Example. Consider the search tree in Figure 3.5 resulting from the weaknesses presented at each level and the system preference that confidentiality is relatively more important than both integrity and availability. Each level is associated with a weakness and the branches at the levels correspond to candidate responses mapped to the weakness at that level. Each node is annotated with the β-value representing the CIA value of the partial solution obtained. The nodes are also annotated with a number; the number refers to the step at which the node is considered to be part of the partial solution. The root, therefore, is annotated with β-value

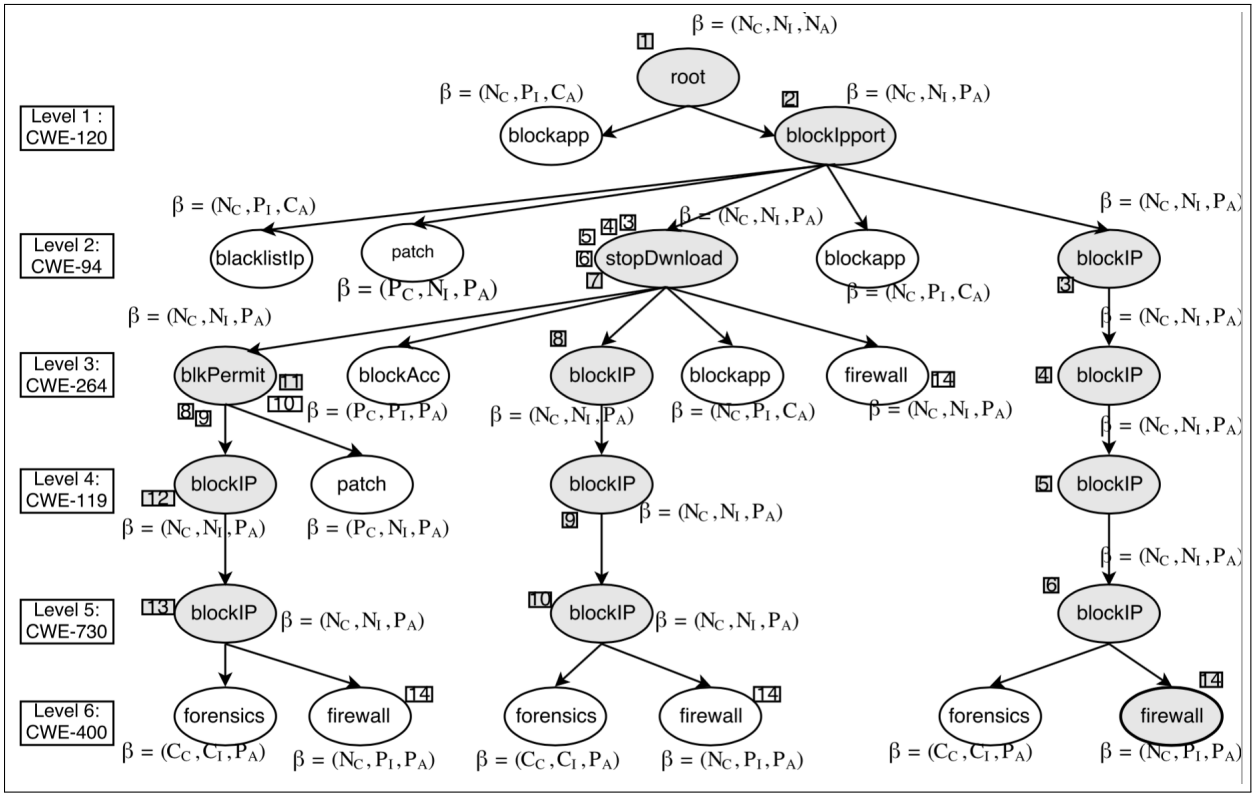

Figure 3.5 Result of applying `rCIA▶rCoverage`

(N_C, N_I, N_A) denoting the partial solution (at this point no responses are selected) does not negatively impact the confidentiality, integrity and availability. At Level 1, the weakness being considered is `CWE-120` (note that in `rCIA▶rCoverage` we do not consider any specific ordering of weaknesses) and there are two possible responses that can address them. The β-values for each of the selection indicates that `blockIpport` is a better choice and as a result, the search tree is expanded following this choice. At Level 2, both `stopDwnload` and `blockIP` have the same β-value. As both these responses are considered, they are annotated with the number 3 indicating that at step 3 these responses have the same β-value. As per the heuristics, the tie is broken using the coverage of these responses over known weaknesses including `CWE-264` (with one-step look-ahead), and `blockIP` is selected as the response at levels 2 and 3 (at this point). Proceeding further, at steps 4, 5, and 6 `blockIP` has coverage over weaknesses at levels 3, 4 and 5, thus maintaining the same β-value. In all these steps, selection of `blockIP` is preferred to `stopDwnload` because of the coverage value. However, at step 7, the partial solution of responses at level 6 involving `firewall` has a β-value, which is less preferred than

the partial solution computed (and not explored) at level 2 involving `stopDwnload`. As a result, our heuristic, proceeds to explore the search tree starting from `stopDwnload` at level 2.

At step 8, There are two partial solutions at level 3 that includes `blkPermit` or `blockIP` that has the same β-value. Amongst them since `blockIP` has a known high coverage (since we have already have knowledge of all detected weaknesses until level 6), that node is chosen for further expansion through steps 9, 10 till a similar situation arises where the partial solution of responses at level 6 have a β-value that is less preferred to the β-value of the partial solution at level 2 involving `blkPermit`.

Hence at step 11, the `blkPermit` node is expanded through steps 12, 13 up to level 6. Now interestingly at step 14, we are faced with three alternatives to choose from, that are all indistinguishable from each other:

{`blockIpport`, `blockIp`, `firewall`}

{`blockIpport`, `stopDwnload`, `blockIp`, `firewall`}

{`blockIpport`, `stopDwnload`, `blkPermit`, `blockIp`, `firewall`}

Now the heuristic looks for the solution that has covered the highest number of weaknesses among the options. This is still a tie since every option covers 6 weaknesses. Then the heuristic identifies the solution that has used minimal number of responses to achieve that coverage . This results in the final solution set consisting of (`blockIpport` , `blockIP` and `firewall`), thus ensuring both quality in terms of system preferences and minimality in terms of response set.

Eventhough `rCIA` and `wCoverage`▶`rCIA` provide results that are indistinguishable in terms of systems preferences to `rCIA`▶`rCoverage`, it is seen in the Figure 3.4 and Figure 3.3 that several decisions made by the former pair of heuristics respectively, were not able to provide the same quality in terms of minimality and time taken as that of the `rCIA`▶`rCoverage`. This is because they make random decisions whenever there is a tie in selecting the node to expand.

CHAPTER 4. EXPERIMENTAL EVALUATION

4.0.1 Experimental setup

We have conducted experiments using as inputs the attacks (CVE-ids), associated weaknesses (CWE-ids) and their CIA impacts from the NVD. The responses are targeted at stopping a weakness from being exploited. Note that, different attacks may exploit the same weakness(es) and hence instead of mapping the responses to possible attacks, it is more intuitive and valuable to map the responses to the weakness that may be exploited. In our experiments, we do not consider uncertainty and assume that any response mapped to a weakness will be able to stop the attacks that solely rely on exploiting those weaknesses. However, as has been noted before, any response, in addition to minimizing the negative impact of allowing an attack to go unmanaged, may incur some negative impact due to its own deployment. Our objective in the experiments is to present the solutions that takes into consideration the trade-off between deploying different set of responses.

There are two different experiments. In the first experimental setup shown in table 4.1, we have used two sets of 12 different system preferences. For each data set, we have considered 7 different attacks with 6 associated weaknesses, and 12 relevant responses. Each weakness has at least one response mapped to it and the maximum number of responses associated to mitigating a weakness is 5. In the second experimental setup we used a subset of size 4 from the above set of weaknesses and ran the same tests. This is shown in table 4.3. The CIA metrics for the attacks/weakness are obtained from the NVD, while the negative impact for deploying the responses are assigned based our understanding and knowledge of the responses. In some cases, the same response may be deployable to address multiple weaknesses; the corresponding negative impact (in terms of CIA) of the response, therefore, can be context-dependent (context

being the weakness(es) it is addressing). In our experiments, we have considered the worst values for the CIA for a response, if it has two or more sets of CIA values representing its negative impact.

4.0.2 Evaluation criteria

We have evaluated the solutions (set of responses) obtained by applying the four heuristics in two dimensions: the quality of the solutions in terms of the CIA values (preferences with respect to the system preferences) and the size of the response set (smaller being better). Tables 4.1 and 4.3 presents the experimental results. The first column presents the system preferences. For instance, the first row in both datasets presents the preference: $C \rhd A \rhd I$ denoting that protecting confidentiality is more important that ensuring availability, which is more important than ensuring integrity. The second column presents the size of the result-set (number of responses necessary) for the "base case"—the base case corresponds to the ideal situation where all possible response-sets are considered by exploring the entire search tree and the best (in terms of system preferences) and the smallest set is selected. So, for the $C \rhd A \rhd I$, the size of the most preferred set of responses is 3. The subsequent columns present the solutions corresponding to our heuristics; for each heuristic the size of the result-set is presented along with whether or not the result-set is as preferred (column title `comp?` short for comparable) as the base case results. Note that, due to the qualitative nature of preferences (see Section 2.2), more than one result may have the same quality with respect to the system preferences.

4.0.3 Observation

The key observations from the experiments are as follows. `GreedyCIA`, as expected, is the fastest running heuristic. However, it cannot consistently compute the results that are either the smallest or the most preferred. This method may be applicable in less critical systems where the system preferences are inconsequential. `wCoverage`▶`rCIA` and `rCIA` are comparable in terms of time and the quality of results. In the former case, coverage is considered indirectly in terms of ordering of weaknesses; however, that does not impact the quality of results in

Table 4.1 Experimental Results

Sys. preference	BaseLine	GreedyCIA			wCoverage▶rCIA			rCIA			rCIA▶rCoverage		
	size	size	comp?	time	size	comp?	time	size	comp?	time	size	comp?	time
Dataset 1													
$C \rhd A \rhd I$	3	6	yes	0.07	3	yes	3.12	5	yes	3.53	3	yes	1.28
$A \rhd C \rhd I$	3	6	no	0.06	5	yes	5.00	5	yes	4.12	3	yes	2.21
$C \rhd A$	3	6	yes	0.07	5	yes	2.56	4	yes	2.37	3	yes	1.01
$A \rhd C$	3	6	no	0.07	5	yes	5.45	5	yes	7.18	3	yes	2.34
$C \rhd A; C \rhd I$	3	6	yes	0.08	3	yes	4.37	5	yes	6.32	3	yes	1.09
$A \rhd C; A \rhd I$	3	6	no	0.07	3	yes	2.32	3	yes	4.45	3	yes	0.58
$I \rhd A \rhd C$	3	6	no	0.10	4	yes	5.04	3	yes	5.12	3	yes	1.11
$I \rhd C \rhd A$	3	6	yes	0.05	3	yes	2.56	4	yes	4.56	3	yes	1.36
$A \rhd C; I \rhd C$	3	6	no	0.07	4	yes	6.51	5	yes	6.57	3	yes	2.01
$C \rhd I; A \rhd I$	3	6	yes	0.08	4	yes	4.32	4	yes	8.12	3	yes	2.21
$I \rhd C; I \rhd A$	3	6	yes	0.07	5	yes	2.01	3	yes	3.12	3	yes	0.24
$A \rhd I$	3	6	yes	0.09	4	yes	3.25	4	yes	4.11	3	yes	1.00
Dataset 2													
$C \rhd A \rhd I$	3	5	yes	0.07	4	yes	2.48	3	yes	3.55	3	yes	0.30
$A \rhd C \rhd I$	4	6	yes	0.08	5	yes	3.49	4	yes	4.21	4	yes	1.03
$C \rhd A$	3	6	no	0.07	3	yes	7.59	4	yes	8.11	3	yes	3.10
$A \rhd C$	4	6	yes	0.06	5	yes	4.12	4	yes	2.27	4	yes	1.11
$C \rhd A; C \rhd I$	3	5	yes	0.06	5	yes	3.14	3	yes	3.11	3	yes	0.40
$A \rhd C; A \rhd I$	4	6	yes	0.08	4	yes	4.32	4	yes	5.46	4	yes	2.10
$I \rhd A \rhd C$	4	6	yes	0.07	4	yes	1.43	4	yes	2.45	4	yes	0.57
$I \rhd C \rhd A$	3	6	no	0.06	4	yes	1.35	3	yes	3.10	3	yes	1.10
$A \rhd C; I \rhd C$	4	6	yes	0.07	5	yes	3.00	4	yes	5.46	4	yes	2.23
$C \rhd I; A \rhd I$	3	6	no	0.06	3	yes	4.32	4	yes	7.23	4	yes	2.56
$I \rhd C; I \rhd A$	3	6	yes	0.07	5	yes	2.31	4	yes	4.54	3	yes	1.45
$A \rhd I$	3	6	no	0.07	4	yes	4.22	5	yes	8.35	4	yes	3.22

Table 4.2 Experimental Results

Sys. preference	BaseLine	GreedyCIA			wCoverage▶rCIA			rCIA			rCIA▶rCoverage		
	size	size	comp?	time	size	comp?	time	size	comp?	time	size	comp?	time
Dataset 1													
$C \rhd A \rhd I$	2	4	yes	0.01	4	yes	0.21	2	yes	0.02	2	yes	0.01
$A \rhd C \rhd I$	2	4	no	0.01	4	yes	2.19	2	yes	0.03	2	yes	0.02
$C \rhd A$	2	4	yes	0.03	3	yes	0.31	2	yes	0.16	2	yes	0.06
$A \rhd C$	2	4	no	0.07	4	yes	1.37	2	yes	0.06	2	yes	0.06
$C \rhd A; C \rhd I$	2	4	yes	0.05	4	yes	0.17	2	yes	0.08	2	yes	0.06
$A \rhd C; A \rhd I$	2	4	no	0.06	2	yes	2.02	2	yes	0.09	2	yes	0.08
$I \rhd A \rhd C$	2	4	no	0.05	4	yes	1.22	2	yes	0.10	2	yes	0.08
$I \rhd C \rhd A$	2	4	yes	0.05	2	yes	0.25	2	yes	0.12	2	yes	0.06
$A \rhd C; I \rhd C$	2	4	no	0.05	2	yes	1.19	2	yes	0.09	2	yes	0.14
$C \rhd I; A \rhd I$	2	4	yes	0.04	4	yes	0.56	2	yes	0.25	2	yes	0.14
$I \rhd C; I \rhd A$	2	4	yes	0.06	3	yes	1.22	2	yes	0.08	2	yes	0.13
$A \rhd I$	2	4	yes	0.04	2	yes	0.20	2	yes	0.25	2	yes	0.12
Dataset 2													
$C \rhd A \rhd I$	2	4	yes	0.01	4	yes	0.10	4	yes	0.03	4	yes	0.05
$A \rhd C \rhd I$	2	4	yes	0.01	3	yes	0.31	2	yes	0.04	2	yes	0.02
$C \rhd A$	2	4	yes	0.07	3	yes	0.30	3	yes	0.42	4	yes	0.10
$A \rhd C$	2	3	yes	2.27	5	yes	4.12	2	yes	0.01	2	yes	0.02
$C \rhd A; C \rhd I$	2	4	yes	0.06	4	yes	0.18	2	yes	0.09	2	yes	0.06
$A \rhd C; A \rhd I$	2	4	yes	0.04	4	yes	1.39	2	yes	0.02	2	yes	0.01
$I \rhd A \rhd C$	2	4	yes	0.05	3	yes	0.43	2	yes	0.02	2	yes	0.03
$I \rhd C \rhd A$	2	4	no	0.05	2	yes	0.46	2	yes	0.31	2	yes	0.43
$A \rhd C; I \rhd C$	2	4	yes	0.08	2	yes	1.33	2	yes	0.03	2	yes	0.04
$C \rhd I; A \rhd I$	2	4	no	0.06	2	yes	1.18	4	yes	0.42	2	yes	0.04
$I \rhd C; I \rhd A$	2	4	yes	0.05	4	yes	0.46	2	yes	0.16	2	yes	0.05
$A \rhd I$	2	4	no	0.05	4	yes	0.26	2	yes	1.16	2	yes	0.04

Table 4.3 Summary of Experimental Results

Sys. preference	GreedyCIA			wCoverage▶rCIA			rCIA			rCIA▶rCoverage		
	size	comp.?	time	size	comp.?	time	size	comp.?	time	size	comp.?	time
Dataset 1												
$C \rhd A \rhd I$	0%	60%	0.01	0%	100%	0.21	70%	100%	0.02	100%	100%	0.01
$A \rhd C \rhd I$	0%	0%	0.01	0%	100%	2.19	70%	100%	0.03	100%	100%	0.02
$C \rhd A$	0%	80%	0.03	40%	100%	0.31	80%	100%	0.16	100%	100%	0.06
$A \rhd C$	0%	10%	0.07	0%	100%	1.37	70%	100%	0.06	80%	100%	0.06
$C \rhd A; C \rhd I$	0%	30%	0.05	0%	100%	0.17	100%	100%	0.08	100%	100%	0.06
$A \rhd C; A \rhd I$	0%	60%	0.06	80%	100%	2.02	100%	100%	0.09	100%	100%	0.08
$I \rhd A \rhd C$	0%	30%	0.05	0%	100%	1.22	90%	100%	0.10	90%	100%	0.08
$I \rhd C \rhd A$	0%	90%	0.05	100%	100%	0.25	60%	100%	0.12	100%	100%	0.06
$A \rhd C; I \rhd C$	0%	100%	0.05	90%	100%	1.19	70%	100%	0.09	100%	100%	0.14
$C \rhd I; A \rhd I$	0%	80%	0.04	0%	100%	0.56	90%	100%	0.25	100%	100%	0.14
$I \rhd C; I \rhd A$	0%	70%	0.06	20%	100%	1.22	100%	100%	0.08	100%	100%	0.13
$A \rhd I$	0%	50%	0.04	80%	100%	0.20	90%	100%	0.25	90%	100%	0.12
Dataset 2												
$C \rhd A \rhd I$	10%	90%	0.01	0%	100%	0.10	20%	100%	0.03	90%	100%	0.05
$A \rhd C \rhd I$	10%	90%	0.01	40%	100%	0.31	40%	100%	0.04	100%	100%	0.02
$C \rhd A$	10%	90%	0.07	60%	100%	0.30	50%	100%	0.42	100%	100%	0.10
$A \rhd C$	10%	60%	2.27	0%	100%	4.12	90%	100%	0.01	80%	100%	0.02
$C \rhd A; C \rhd I$	10%	100%	0.06	0%	100%	0.18	70%	100%	0.09	100%	100%	0.06
$A \rhd C; A \rhd I$	10%	100%	0.04	0%	100%	1.39	80%	100%	0.02	100%	100%	0.01
$I \rhd A \rhd C$	10%	90%	0.05	50%	100%	0.43	90%	100%	0.02	100%	100%	0.03
$I \rhd C \rhd A$	10%	40%	0.05	90%	100%	0.46	70%	100%	0.31	80%	100%	0.43
$A \rhd C; I \rhd C$	10%	0%	0.08	80%	100%	1.33	80%	100%	0.03	100%	100%	0.04
$C \rhd I; A \rhd I$	10%	20%	0.06	70%	100%	1.18	30%	100%	0.42	100%	100%	0.04
$I \rhd C; I \rhd A$	10%	30%	0.05	0%	100%	0.46	10%	100%	0.16	100%	100%	0.05
$A \rhd I$	10%	70%	0.05	0%	100%	0.26	70%	100%	1.16	100%	90%	0.04

terms of minimizing the size of the solution. In the latter case, the coverage is not considered as a criteria for response selection and weaknesses do not have any specific ordering. In short, the experiments reveal that the ordering of weaknesses does not impact the quality of the solution in terms of CIA preferences. Finally, the `rCIA▶rCoverage` method has taken less time than both `wCoverage▶rCIA` and `rCIA`, and also performs better in terms of quality of the results. The reason for taking less time can be attributed to the fact that indistinguishability between partial solutions during exploration are taken care of using coverage, which reduces the breadth of exploration in the search tree. As a result, the number of responses is minimal because coverage is directly considered to guide the search path. Therefore, in general, we can conclude that `rCIA▶rCoverage` is likely to outperform the other heuristics as per the evaluation criteria.

CHAPTER 5. SUMMARY AND DISCUSSION

We present a road-map for incorporating qualitative preferences and priorities in selecting responses. We present four heuristic methods to realize the road-map and present experimental evaluations characterizing the quality of the results in terms of the CIA values and the size. The methods can be applied on-the-fly, i.e., as when weaknesses are detected. This is because the order in which weaknesses are considered does not impact the optimality of the solution in terms of the CIA value. We have shown that `rCIA▶rCoverage` method is likely to be the best heuristic.

As part of future work, we plan to further develop the framework with clearly defined interfaces to allow easy plug-n-play with existing intrusion detection engines. Additionally, we plan to include context-sensitive information in response selection process. One of the context-sensitive information that we have ignored (for simplicity) involves the parameters of responses.

For instance, the response `blockIp` requires the parameter, an `IP` address, to be effective. Once such parameters are considered, then the minimality of the solution will not only depend on the type of the responses in the solution set but also the parameters used in the responses—`blockIp` applied on two different `IP` address will have to be counted as a two different response. This will be a straightforward extension to our methods.

Another type of context information involves the CIA values representing the negative impact of the responses. In our current setting, we have considered that the negative impact of a response is the worst possible value it can have with respect to CIA. As a result, when comparing two or more responses, our comparison uses these worst values. However, one can also consider comparing responses in the context of the weaknesses for which they are being considered. Such context-sensitivity is likely to bring in dependency of the heuristics on the

ordering of weaknesses (a priori knowledge of all detected weaknesses) and therefore, may not be practical in general. We plan to investigate extensions of our methods that can incorporate this level of context-sensitivity without compromising on the quality of results and the efficiency of the computation.

BIBLIOGRAPHY

(CNBC cybersecurity threat report 2015). Cnbc cybersecurity threat report for 2015. http://www.cnbc.com/2014/12/19/top-5-cyber-security-risks-for-2015.html.

(Mcafee threat predictions 2016). Mcafee threat predictions for 2016. http://www.mcafee.com/us/resources/reports/rp-threats-predictions-2016.pdf.

Anuar, N. B., Papadaki, M., Furnell, S., and Clarke, N. (2010). An investigation and survey of response options for intrusion response systems (irss). In *Information Security for South Africa (ISSA), 2010*, pages 1–8. IEEE.

Axelsson, S. (2000). The base-rate fallacy and the difficulty of intrusion detection. *ACM Transactions on Information and System Security (TISSEC)*, 3(3):186–205.

Bartol, N., Bates, B., Goertzel, K. M., and Winograd, T. (2009). Measuring cyber security and information assurance: a state-of-the-art report. *Information Assurance Technology Analysis Center IATAC*.

Bass, T. (2000). Intrusion detection systems and multisensor data fusion. *Communications of the ACM*, 43(4):99–105.

Boutilier, C., Brafman, R. I., Hoos, H. H., and Poole, D. (1999). Reasoning with conditional ceteris paribus preference statements. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 71–80. Morgan Kaufmann Publishers Inc.

Brafman, R. I., Domshlak, C., and Shimony, S. E. (2006). On graphical modeling of preference and importance. *Journal of Artificial Intelligence Research*, 25:389–424.

Foo, B., Wu, Y.-S., Mao, Y.-C., Bagchi, S., and Spafford, E. (2005). Adepts: adaptive intrusion response using attack graphs in an e-commerce environment. In *Dependable Systems and Networks, 2005. DSN 2005. Proceedings. International Conference on*, pages 508–517. IEEE.

Jahnke, M., Thul, C., and Martini, P. (2007). Graph based metrics for intrusion response measures in computer networks. In *Local Computer Networks, 2007. LCN 2007. 32nd IEEE Conference on*, pages 1035–1042. IEEE.

Jansen, W. (2009). Directions in security metrics research. national institute of standards and technology. *Computer Security Division*.

Kabiri, P. and Ghorbani, A. A. (2005). Research on intrusion detection and response: A survey. *IJ Network Security*, 1(2):84–102.

Kanoun, W., Cuppens-Boulahia, N., Cuppens, F., and Dubus, S. (2010). Risk-aware framework for activating and deactivating policy-based response. In *Network and System Security (NSS), 2010 4th International Conference on*, pages 207–215. IEEE.

Kheir, N., Debar, H., Cuppens-Boulahia, N., Cuppens, F., and Viinikka, J. (2009). Cost evaluation for intrusion response using dependency graphs. In *Network and Service Security, 2009. N2S'09. International Conference on*, pages 1–6. IEEE.

Lazarevic, A., Kumar, V., and Srivastava, J. (2005). Intrusion detection: A survey. In *Managing Cyber Threats*, pages 19–78. Springer.

Lewandowski, S. M., Van Hook, D. J., Leary, G. C., Haines, J. W., and Rossey, L. M. (2001). Sara: Survivable autonomic response architecture. In *DARPA Information Survivability Conference & Exposition II, 2001. DISCEX'01. Proceedings*, volume 1, pages 77–88. IEEE.

NVD (2007). NVD database. https://nvd.nist.gov.

Papadaki, M. and Furnell, S. (2006). Achieving automated intrusion response: a prototype implementation. *Information management & computer security*, 14(3):235–251.

Rowe, J., Schnackenberg, D., Darby, D., Levitt, K., Wee, C., Klotz, D., and Schatz, J. (1999). Intrusion detection and isolation protocol: Automated response to attacks. In *Recent Advances in Intrusion Detection*.

Santhanam, G. R., Basu, S., and Honavar, V. (2008). Tcp- compose–a tcp-net based algorithm for efficient composition of web services using qualitative preferences. In *International Conference on Service-Oriented Computing*, pages 453–467. Springer.

Santhanam, G. R., Basu, S., and Honavar, V. (2011). Representing and reasoning with qualitative preferences for compositional systems. *Journal of Artificial Intelligence Research*, 42:211–274.

Shameli-Sendi, A., Ezzati-Jivan, N., Jabbarifar, M., and Dagenais, M. (2012). Intrusion response systems: survey and taxonomy. *Int. J. Comput. Sci. Netw. Secur*, 12(1):1–14.

Stakhanova, N., Basu, S., and Wong, J. (2007). A taxonomy of intrusion response systems. *International Journal of Information and Computer Security*, 1(1-2):169–184.

Thames, J. L., Abler, R., and Keeling, D. (2008). A distributed firewall and active response architecture providing preemptive protection. In *Proceedings of the 46th Annual Southeast Regional Conference on XX*, pages 220–225. ACM.

Toth, T. and Kruegel, C. (2002). Evaluating the impact of automated intrusion response mechanisms. In *Computer Security Applications Conference, 2002. Proceedings. 18th Annual*, pages 301–310. IEEE.

Wang, X., Reeves, D. S., Wu, S. F., and Yuill, J. (2002). Sleepy watermark tracing: An active network-based intrusion response framework. In *Trusted Information*, pages 369–384. Springer.

Zhang, F., Zhou, S., Qin, Z., and Liu, J. (2003). Honeypot: a supplemented active defense system for network security. In *Parallel and Distributed Computing, Applications and Technologies, 2003. PDCAT'2003. Proceedings of the Fourth International Conference on*, pages 231–235. IEEE.