2008

The Methodology for Evaluating Response Cost for Intrusion Response Systems

Christopher Roy Strasburg
Iowa State University, cstras@iastate.edu

Natalia Stakhanova
Iowa State University

Samik Basu
Iowa State University, sbasu@iastate.edu

Johnny S. Wong
Iowa State University, wong@iastate.edu

Recommended Citation

Strasburg, Christopher Roy; Stakhanova, Natalia; Basu, Samik; and Wong, Johnny S., "The Methodology for Evaluating Response Cost for Intrusion Response Systems" (2008). *Computer Science Technical Reports*. 199.
http://lib.dr.iastate.edu/cs_techreports/199

The Methodology for Evaluating Response Cost for Intrusion Response Systems

Abstract

Recent advances in the field of intrusion detection brought new requirements to intrusion prevention and response. Traditionally, the response to the detected attack was selected and deployed manually, in the recent years the focus has shifted towards developing automated and semi-automated methodologies for responding to intrusions. In this context, the cost-sensitive intrusion response models have gained the most interest mainly due to their emphasis on the balance between potential damage incurred by the intrusion and cost of the response. However, one of the challenges in applying this approach is defining consistent and adaptable measurement of these cost factors on the basis of requirements and policy of the system being protected against intrusions. In this paper we present a structured methodology for evaluating cost of responses based on three factors: the response operational cost associated with the daily maintenance of the response, the response goodness that measures the applicability of the selected response for a detected intrusion and the response impact on the system that refers to the possible response effect on the system functionality. The proposed approach provides consistent basis for response evaluation across different systems while incorporating security policy and properties of specific system environment. We demonstrate the advantages of the proposed cost model and evaluate it on the example of three systems.

Disciplines

Information Security | OS and Networks

The Methodology for Evaluating Response Cost for Intrusion Response Systems

Chris Strasburg
Department of Computer Science
Iowa State University, USA
cstras@cs.iastate.edu

Natalia Stakhanova
Faculty of Computer Science
University of New Brunswick, Canada
natalia@unb.ca

Samik Basu
Department of Computer Science
Iowa State University, USA
sbasu@cs.iastate.edu

Johnny S. Wong
Department of Computer Science
Iowa State University, USA
wong@cs.iastate.edu

ABSTRACT

Recent advances in the field of intrusion detection brought new requirements to intrusion prevention and response. Traditionally, the response to the detected attack was selected and deployed manually, in the recent years the focus has shifted towards developing automated and semi-automated methodologies for responding to intrusions. In this context, the cost-sensitive intrusion response models have gained the most interest mainly due to their emphasis on the balance between potential damage incurred by the intrusion and cost of the response. However, one of the challenges in applying this approach is defining consistent and adaptable measurement of these cost factors on the basis of requirements and policy of the system being protected against intrusions.

In this paper we present a structured methodology for evaluating cost of responses based on three factors: the *response operational cost* associated with the daily maintenance of the response, the *response goodness* that measures the applicability of the selected response for a detected intrusion and the *response impact on the system* that refers to the possible response effect on the system functionality. The proposed approach provides consistent basis for response evaluation across different systems while incorporating security policy and properties of specific system environment. We demonstrate the advantages of the proposed cost model and evaluate it on the example of three systems.

Keywords

intrusion response assessment, cost-sensitive intrusion response

1. INTRODUCTION

The proliferation of complex and fast-spreading intrusions against computer systems brought new requirements to intrusion detection and response, demanding not only advances in intrusion detection mechanisms but also the development of sophisticated and automated intrusion response systems.

The majority of existing automatic intrusion response systems rely on the mapping of attacks to pre-defined responses [6, 7]. These approaches allow the system administrator to deal with intrusions faster and more efficiently. However, they lack flexibility mainly because few of these systems take into account intrusion cost factors.

In recent years the trend toward cost-sensitive modeling of response selection became more apparent [5, 1, 3, 12, 9]. The primary aim for applying such models is to balance intrusion damage and response cost to ensure adequate response without sacrificing the normal functionality of the system under attack. However, defining accurate measurement of these cost factors is one of the challenges in using cost-sensitive modeling approach.

Driving Problem. The traditional approach to the analysis of response cost is based on a manual assessment of various factors such as response effectiveness, probability of failure, expected duration of the response effect, etc [4]. This approach is highly accurate as it involves expert knowledge and judgment, however it often introduces a significant delay in reacting to the failure, and thus is not always appropriate in critical environments and unsuitable for automated response systems.

Some of the existing models supporting automatic response selection introduce response cost, along with intrusion damage cost, as one of the factors in the intrusion severity assessment for the attacked system and selection of the suitable response strategy. However, they generally do not agree on what constitutes the response cost and how it can be measured. Some suggest that response cost includes the labor cost of personnel involved in response deployment and criticality of the attack [5], others see response cost as a measure of response effectiveness to a detected attack and its disruptiveness to legitimate users [3].

Such disagreement primarily results from the lack of a consistent and standardized approach to measuring response cost. The goal of this work is to identify metrics representing intrusion response cost and develop a structured methodology for evaluating these metrics. Below we outline several key considerations surrounding the intrusion response cost followed by the primary contributions of this paper.

Solution Methodology. One of the challenging problems in the context of intrusion response systems is to identify whether or not a response should be deployed, in other words, what is the best suited action when an intrusion is detected. This problem primarily stems from the fact that though responses are deployed with the goal of countering an intrusion, they may not only fail but can also lead to undesired effects on the system. Thus, often the primary criteria in response selection mechanisms are the expected effectiveness of the response against the intrusion and its potential negative impact on the system.

The effectiveness of a response refers to both the ability of the response to prevent or mitigate damage from the intrusion and the coverage of the response, i.e., the number of intrusions it can potentially address. One of the intuitive ways to measure the effect of the response is to consider the system resources affected by the intrusion and protected by the response.

Another factor characterizing the response is its potential effect on the system. While the responses are deployed against a detected intrusion, they often alter the state of the system negatively affecting system resources and leading to damage. For example, complete network isolation of a Web server in response to an SQL attack, although effectively stops the intrusion, also results in unavailability of service. While such an intrusion response might be desirable in a security-critical system, it is unacceptable in a service-oriented setting.

Although, the response effect on the detected intrusion and its impact on the system resources are the primary characteristics considered for intrusion response, there is a third component that often remains behind the curtains, an operational cost of the response in the form of administrator time and additional system resources (i.e., storage, network bandwidth, etc.) required for response setup and processing. While this cost does not directly affect the attacked system or the intrusion, it can significantly contribute to the decision of which response to deploy.

In light of the above, we present a structured system independent methodology for the evaluation of responses' cost based on the three parameters: (a) *the response goodness in addressing the detected intrusion*(s) which includes the effectiveness of the response and its coverage capability, (b) *the damage incurred by a response on the system* and (c) *the operational cost of a response on a given system.*

Within this methodology, we propose to assess response impact with respect to resources of the affected system. Our model takes into account the relative importance of the system resources determined through the review of the system policy goals according to the following categories: *confidentiality*, *availability* and *integrity.*

One of the important steps in this process is the analysis of the system resources that includes the enumeration of the available system resources and their classification. This not only reveals the underlying ranking of the resources but also provides a better understanding of the response impact.

Based on this analysis, the evaluation algorithm assesses the *response goodness* in terms of the resources protected by the response and the *response damage* in terms of the resources impaired by the action. In addition to this, the proposed model incorporates the *operational cost* of the response defined with respect to human effort required, necessary system resources and involved monetary expenses.

This methodology does not substitute the response selection process in case of detected intrusion, but rather allows to evaluate the available responses on the consistent basis. We have implemented the proposed response cost evaluation methodology as semi-automated tool that can be employed to guide system administrators during the response selection process.

Contributions. The main contributions can be summarized as follows:

1. **Structured and comprehensive methodology for assigning response costs**: the proposed model presents the first roadmap for defining a standardized metric for response cost evaluation.

2. **System independent evaluation model:** the proposed model is adaptable to different environment settings, ie. systems with widely varying operational requirements.

3. **Consistent response metrics**: the proposed evaluation metrics are defined in terms of the system resources that bring a common ground to the assessment process.

4. **Adaptable evaluation metrics**: the response metrics are quantified with respect to the security policies and properties of the specific system. Thus, the computed costs can be effortlessly adjusted as and when the system requirements are modified.

5. **Practical applicability**: the advantages of the proposed model are demonstrated on the example of three systems.

Organization. The remainder of the paper is organized as follows. A brief overview of related work is given in Section 2. Section 3 outlines the response cost evaluation approach. Sections 4 and 5 provide the details of the response goodness and response system impact evaluation. Experimental setup and results are given in Section 6. Section 7 concludes the paper with our future work.

2. RELATED WORK

A number of techniques aiming at enhancing intrusion response automation were proposed and deployed over the last five years. A comprehensive review of this research work is given by Stakhanova et al. [10]. Comparatively the field of the response cost assessment has received considerably less attention.

The approach to intrusion response proposed by Lee et al. [5] is based on a combination of intrusion detection and response factors. Three cost factors were identified: *operational cost* which includes the cost of processing and analyzing data for detecting the intrusion, *damage cost* which assesses the amount of damage that could potentially be caused by the attack and *response cost* which characterizes the operational cost of reaction to the intrusion. These factors present the foundation of the intrusion cost model, i.e. total expected cost of intrusion detection, and consequently provide a basis for the selection of an appropriate response.

Another approach called ADEPTS, Adaptive Intrusion Response using Attack Graphs, proposed by Foo et al. [3], employs attack graphs to identify the actions required to achieve possible attack targets in a distributed system, and consequently, to show the objectives of suitable responses. Attacker goals are expressed as end states in the attack graph with intermediate steps leading to the fulfillment of those goals. Responses are selected to frustrate attack goals based on *effectiveness* to that particular attack in the past, *disruptiveness* to legitimate users and the *confidence level* which indicates the probability that the attack is actually taking place.

Models proposed by Toth and Kruegel [11], Balepin et al. [1] and Jahnke [4] not only consider costs and benefits of the responses, but also introduce a link between the cost of responses and the system resources in the network.

The approach proposed by Toth and Kruegel is a network-based response mechanism that builds a dependency tree of the resources on the network. The proposed algorithm for optimal response selection takes into account the penalty cost of a resource being *unavailable* and the *capability* of a resource that indicates the resource performance if the specified response strategy is triggered, compared to the situation when all resources are available. Clearly, the set of responses with the least negative impact on the system (lowest penalty cost) is chosen to be applied in response to the detected intrusion.

An approach proposed by Jahnke [4] also attempts to quantify response measures based on modeling system resources as a dependency graph. Although this method requires careful graph construction and validation, it allows automatic assessment of the *response success* computed through the change of the availability of resource nodes in the graph and *required effort or cost* defined as the amount of instances to be modified for deploying the selected response.

A similar approach, for host-based intrusion detection and response, was proposed by Balepin et al. [1] . The local resource hierarchy is modeled as a directed graph where the nodes represent specific system resources and the edges are the dependencies between them. Each node is associated with a list of responses that can be applied to restore working state of resource in case of an attack. A particular response for a node is selected based on (a) *the cost of the response*, the sum of the resources that will be affected by the response action, (b) *the benefit of the response*, the sum of the nodes, previously affected by intrusion that will be restored to working state, and (c) *the cost of the resource*, the quantification of the importance of the resource.

While these approaches include response cost as a necessary measurement in the response selection process, each computes response cost using different techniques. In this work we attempt to generalize the response cost metric and address the following:

- *The existing models are system dependent.* Most of these approaches focus on networks of systems [3, 11, 4], with only few considering host-based response [5, 1]. However, with the exception of [1], they are all specifically designed to reflect characteristics of the considered system. This significantly limits the applicability of the models to varying system constraints, and thus, their practicality. In this work we propose a common methodology for response cost estimation that is adaptable to different environment settings.

- *The existing models only partially outline the factors contributing to response cost.* While most of these approaches use the concept of response benefit or effectiveness as a factor related to the response's ability to mitigate the intrusion damage, the operational cost of the response is not included. Our model incorporates three major components that define response cost: response goodness, response impact on the system and response operational cost.

- *The existing models lack consistency with each other.* Each model approaches response cost evaluation from a different perspective. Foo et al. [3] measures the response effectiveness against a detected attack based on the past experience. Lee et al. [5] relates the response cost to the required labor efforts, while the works by [11], [1]. and [4] consider response cost in association with the system resources, but with varying evaluation methods. [1] measures the response cost as the sum of manually assigned costs of affected resources. [11] calculates response cost as a function of system capability reduction, while [4] essentially extends the idea in [11] by adding a fine-grained quantification of system resource unavailability. The emerging theme in these works effectively establishes the idea of employing system resources to estimate response cost, and based on this promising trend we work to build a structured and consistent methodology for response cost assessment based on the resources of the system. Thus our model can be viewed as a generalization of the existing approaches.

Our aim is to preserve the strengths of these works while avoiding their disadvantages. Toward this end, the proposed model identifies the major factors that constitute the response cost. We also provide a comprehensive step-by-step methodology for the automatic and consistent assessment of these factors, accounting for system-specific confidentiality, reliability and availability requirements.

3. RESOURCE COST EVALUATION MODEL: OVERVIEW

The evaluation of the intrusion response cost is performed in three dimensions: *the operational cost* (OC) of a response in a given environment, *the response goodness* (RG) with respect to detected intrusion(s) and *the response impact on the system*(RSI).

The operational cost of a response measures various aspects of the response associated with its daily maintenance. The response goodness provides a measure of the ability of the corresponding response to mitigate damage caused by the intrusion to the system resources. Finally, the impact of a response on the system quantifies the negative effect of the response on the system resources and is estimated independently from the response success or failure in countering the intrusion(s).

Intuitively, the combination of the OC and the RSI constitutes the penalty associated with the response, while the RG is the benefit of this response measure. One simple cost model describing the overall measure of response cost RC is:

$$\mathtt{RC} = \frac{\mathtt{OC} + \mathtt{RSI}}{2} - \mathtt{RG} \tag{1}$$

3.1 Methodology For Response Cost Evaluation

Figure 1 presents the overview of steps for evaluating the response cost RC (following Equation 1). In the following, we discuss each step in detail.

Step 1: System classification . The first step in quantifying the cost of a response involves determining the characteristics of the computing environment where the response will be deployed.

Generally, the systems can be classified according to the security policy goals of the organization into two broad categories:

- **open access systems** have minor or no restrictions. Example of such systems are the public networks provided at airport, city, etc. Other examples would include public web servers, DNS servers, or e-mail services offered for general public consumption.
- **limited access systems** are systems that require rigorous authentication and can be further classified according to their primary emphasis as follows:
    - **safety-critical systems** emphasize the necessity of service availability and, in case of failure, require a safe degradation of services they provide.
    - **security-critical systems** focus on security, and thus, have data confidentiality and integrity as their primary concerns.
    - **business-critical systems** are a combination of safety and security-critical systems. The primary goal of a business-critical system is profit or business security. As such, safety-critical requirements like service availability and system performance are tightly coupled with equally valuable requirements of data confidentiality and integrity.

The above classification provides important insights to the risks that each class of systems can tolerate, and therefore helps in measuring the cost for various types of intrusion damages. For example, a public Web server and a financial processing system will have different sensitivities toward data confidentiality and availability. Even within a system class, security priorities can differ. For instance, a business critical Web server used to accept product orders may have very different requirements from the one used to process payroll.

Step 2: The system policy goals. The determination of the importance of the system policy goals, and subsequently, the assessment of the potential risks are the responsibilities of the organization to which the system belongs. It is usually a manual process consisting of an informal series of questions such as “Will data be exposed?”, “How critical is the confidentiality of the data?”, “How concerned are we with data integrity?”, “Will service availability be impacted?”, etc. This provides an ad-hoc relative assessment of the system goals for the organization. Based on the above observation, system policy goals can be classified as follows:

1. *Confidentiality* refers to the imposed restrictions on information flows, e.g., restricted access to data.

2. *Integrity* is a guarantee of the consistency and accuracy of the information or the system computing environment as a whole.

3. *Availability* indicates the requirement of (functionality, storage etc.) service and information availability upon request.

These categories of system goals are ranked according to their importance (a value between 0 for *no importance* and 1 for *absolute importance*) in a particular system type (safety-critical, security-critical, etc.). These decisions can be based on monetary values or other established business metrics for the cost of failure to meet system goals (e.g., the estimated dollar cost of a confidentiality breach). In the case of a classified data processing system (a security-critical system), for instance, data confidentiality may be a 1, indicating the absolute importance of this security facet for this system.

1: **The system classification:**
-*identify the type of the system according to the security priorities*
2: **The system policy goals:**
-*assign weights to system policy goals for the system*
3: **The system resources:**
-*enumerate resources available on the given system*
-*determine the resource importance for each system policy category*
-*compute the overall resource weight for the system policy*
4: **The response taxonomy:**
-*identify the responses suitable for the system*
5: **The response operational cost:**
-*assess the operational cost of the responses*
6: **The response goodness:**
-*assess the goodness of the responses*
7: **The response impact on the system:**
-*compute the impact of the available responses on the given system*

Figure 1: The methodology for intrusion response cost evaluation.

Step 3: System resources. Responses are reactions to the intrusions and are directed to protect the system resources threatened by an attack. System resources can be broadly viewed as the system assets (e.g., host, network, etc.), services provided by the system (e.g., FTP, HTTP, file system, etc.) and users served by the system.

One of the initial steps in the process of computing a response impact measure is the enumeration of the resources available in the considered system. The importance of a resource depends on the system policy goals which in turn depend on the type of system. For example, for a simple Web server, availability is an important policy goal and accordingly important resources will include HTTP. Therefore, the resources are assigned weights according to their importance for each system policy goal for a specific system. The overall weight of the system resource, denoted by $W_{\texttt{SR}}$, is computed as a combination of the resource importance for each policy goal category $\texttt{SRimportance}_i$ (i is the policy category index) and the system specific category weight $\texttt{PolicyCategoryWeight}_i$ (weight of the i-th policy category index):

$$W_{\texttt{SR}} = \sum_i [\texttt{SRimportance}_i \times \texttt{PolicyCategoryWeight}_i] \quad (2)$$

To illustrate this process, lets consider the example of the network interface resource and its importance for each policy category for an open access system (i.e., a public Web server):

Policy Goal		**Resource Importance**
Category	**Weight**	*Network Access*
Data confidentiality	0	0.1
Data availability	1.0	1.0
Data integrity	0.7	0.1

Following Equation 2, the weight of the network interface resource for the system policy is computed as follows:

$$W_{\texttt{NetworkAccess}} = 0 \times 0.1 + 1.0 \times 1.0 + 0.7 \times 0.1 = 1.07$$

Step 4: Taxonomy of responses. Once the system goals are identified, its resources are enumerated and their importance is quantified based on the system goals, the next step is to identify the set of responses that are suitable for a system. Generally, the responses are deployed to either counter possible attacks and defend the system resources or regain secure system state. Thus, the selection of applicable responses primarily depends on the identified system resources. In this work, we identify the set of appropriate responses based on the general taxonomy of intrusion response actions developed by [8].

Steps 5-7: Assessment of response cost. One of the challenges in assessing response cost is to accurately define numeric values. Assigning monetary values to reflect response cost, although provides concrete metric, is not always possible. More effective solution can be provided with the use of the relative measurements constructed based on system-specific policies.

The assessment of *response operational cost* is generally independent from the system policy and includes the cost for the setup and deployment of the response, and data processing overhead needed to analyze the result of response. For example, "the system logging" response is fairly easy to setup. However, it requires significant storage resources and often incurs high processing overhead. Broadly, the involved operational expenses can be classified on the basis of three requirements: *human resources* which refer to administrator time, *system resources*, which include storage, network bandwidth, processor time, etc., and *direct expenses* which include data processing fees by a third party, subscription service fees, cost of components replacement, etc. Determining these factors is a manual process that involves expert knowledge and a high degree of judgment.

At the same time the other two factors: the applicability of the selected responses for a detected intrusion (response goodness) and the assessment of the possible response effects on the system (response system impact), can be only evaluated in the context of the system.

Thus, the proposed computation model for these factors in-

1: **The applicable responses for intrusions:**
-*determine suitable responses for deployment attack signatures*
2: **The response goodness:**
-*compute the goodness measures for responses in terms of known attacks*

Figure 2: Intrusion response goodness estimation steps.

tegrates the relative impact of the intrusion response with the environmental factors of the system. The proceeding sections elaborate on our model to measure the response goodness and impact of the response on the system.

4. ASSESSING THE RESPONSE GOODNESS

The steps for evaluating the response goodness are presented in Figure 2.

Step 1: Applicable responses. . Often the detection mechanism of the intrusion detection system (IDS) provides administrators with a set of alerts indicating potential attacks rather than a specific intrusion. When this situation arises, the response needs to be deployed preemptively on the basis of high likelihood of possible intrusions. In these cases, the response is evaluated based on the number of possible intrusions it can potentially address, and consequently, the number of resources that can be protected by the response. In practice, the applicability of responses to potential attacks can be determined through the analysis of the existing intrusion signatures in the IDS.

Step 2: Response goodness. . The response goodness includes a review of the availability of the system resources involved in the intrusion. For example, an alert triggered on TFTP traffic on port 69 should not be accounted for in the response goodness assessment if TFTP protocol is not currently supported.

The goodness of the response $\mathtt{R}_i$ where $i \in [1 \dots m]$ (m different responses) against the intrusion $\mathtt{I}_j$ potentially affecting n system resources $\mathtt{SR}_1^j, \mathtt{SR}_2^j, \dots \mathtt{SR}_n^j$ is computed as follows:

$$\mathtt{RG}_{\mathtt{R}_i}(\mathtt{I}_j) = \sum_{k \in [1 \dots n]} \mathtt{Avail}(\mathtt{SR}_k^j) \times \mathtt{W}_{\mathtt{SR}_k} \qquad (3)$$

where $\mathtt{Avail}(\mathtt{SR}_k^j)$ is a binary value that denotes the availability of k-th system resource that can be affected by $\mathtt{I}_j$ and $\mathtt{W}_{\mathtt{SR}_k}$ is the resource weight (as computed by Equation 2 in Section 3.1). To ensure the consistency of the computed metric, $\mathtt{RG}$ values are normalized within a range of $[0, 1]$ by dividing individual $\mathtt{RG}_{\mathtt{R}_i}(\mathtt{I}_j)$ by the normalization term, $\mathtt{MAX}(\mathtt{RG}(\mathtt{I}_j))$ which is the maximum $\mathtt{RG}$ value computed for available responses for the intrusion $\mathtt{I}_j$, i.e.,

$$\begin{aligned} &\mathtt{MAX}(\mathtt{RG}(\mathtt{I}_j)) = \mathtt{RG}_{\mathtt{R}_l}(\mathtt{I}_j) \text{ such that} \\ &\qquad l \in [1 \dots m] \ \wedge \ \forall i \in [1 \dots m] : \mathtt{RG}_{\mathtt{R}_i}(\mathtt{I}_j) \leq \mathtt{RG}_{\mathtt{R}_l}(\mathtt{I}_j) \end{aligned}$$

In the rest of the paper, we will refer to $\mathtt{RG}_{\mathtt{R}_i}(\mathtt{I}_j)$ to mean its normalized valuation.

5. ASSESSING THE RESPONSE IMPACT ON THE SYSTEM

The step-by-step process for evaluation of the response impact on the considered system is presented in Figure 3.

Step 1: Response Impact on System Resources. The impact of a response is evaluated based on the defined system goals and their importance. The impact assessment process for a specific response includes three steps. First, identify the system resources affected by each response. Second, for each resource, order the responses on the basis of how they are affecting the resource. Finally, compute the negative impact of the responses on the associated resource using the ordering obtained above. Eventually, the impact of a response on the system as a whole will be an aggregation of the response's impact on the resources present in the system.

For each response we determine the system resources it may affect. For instance, *blocking a specific subnet* can protect the network interface resource and also disrupt legitimate user activities. After all responses are categorized within the considered system resource, we independently evaluate each system resource. All responses affecting the resource are ordered or ranked based on their relative impact on the considered resource, from the greatest impact to the least impact, and assigned an index $i \in [0 \dots (m-1)]$, where m is the total number of responses in the list corresponding to a particular resource. A response with rank i has more impact on the corresponding resource than the response with rank j $(i < j)$. These ranks are based on historical data and/or the expertise of the system administrator. We quantify the impact using the rank as follows:

$$\mathtt{Impact}_{\mathtt{R}_i,\mathtt{SR}} = 1 - \frac{i}{m} \qquad (4)$$

where R_i is the i-th ranked response. The resultant valuation is between $\frac{1}{m}$ and 1. To illustrate this process, lets consider the example of the *network interface* resource. The set of available responses are ranked according to their impact and the corresponding impact quantification is computed as follows:

Rank i	**Responses for** $\mathtt{SR}$ $(\mathtt{R}_i)$	$\mathtt{Impact}_{\mathtt{R}_i,\mathtt{SR}}$
0.	Complete network isolation	1 - 0/5 = 1.0
1.	Network isolation: block subnet	0.8
2.	Terminate process	0.6
3.	Delay suspicious process	0.4
4.	Deploy intrusion analysis tools	0.2

Generally, the values determined as a result of ranking are dependent on the characteristics of system environment. Thus, changes in the environment, i.e., modifications in the software usage, addition of network equipment, new knowledge or skills gained by the administrator, etc., can affect the order and relative severity of the responses. Thus, as the settings of the environment change, these values may be manually adjusted to more accurately reflect relative damage on the system resources.

For instance, automatically restarting terminated processes

1: **Response Impact on System Resources:**
-*identify the system resources affected by each response*
-*order responses for each system resource based on their relative impact on that resource*
-*for each system resource assign numeric value to responses according to their place in that list*
2: **The Overall Response System Impact:**
-*compute the impact values for the available responses on the given system*

Figure 3: Intrusion response system impact assessment process.

may justify changing the network interface impact rating of *Terminate process* to 0.2. In addition, a uniform distribution of responses according to their order may not always be appropriate. For instance, if two responses affect a resource in an identical fashion, then manual adjustment may be necessary. Note that these adjustments are made on an environmental and technical basis only, independent of the policy implications.

Step 2: The Overall Response System Impact Computation. The overall impact of the response measure is estimated based on the weight of the system resource for a specific system policy (Equation 2 in Section 3) and the impact value of the response for that resource (Equation 4). The overall rating of the response $\mathtt{R}_i$ on the system, the response system impact, denoted by $\mathtt{RSI}_{\mathtt{R}_i}$, is computed as follows:

$$\mathtt{RSI}_{\mathtt{R}_i} = \sum_{\mathtt{SR}} \mathtt{Impact}_{\mathtt{R}_i,\mathtt{SR}} \times \mathtt{W}_{\mathtt{SR}} \qquad (5)$$

Similar to RG valuations (Equation 3 in Section 4), we normalize $\mathtt{RSI}_{\mathtt{R}_i}$ using the maximum valuation of RSI for any response.

While manual assignment of some values is inevitable, these abstractions allow an expert to focus separately on the technical nature of the responses and the high-level goals of the system. In many cases, two different individuals or groups are uniquely qualified to make the respective technical and policy based decisions. As such if the environment changes, the system administrator can modify the high-level system goals while a technical specialist adjusts response damage factors based on changes to the system or network environment. Such separation of concern reduces the decision complexity, and therefore, the risk of human error.

6. EXPERIMENTAL RESULTS

6.1 Case Study

We have implemented the proposed response cost evaluation methodology as semi-automated tool that relies on administrator to provide system resource values and ranking of the responses according to their impact on resources. Using this tool we have analyzed the response costs of the following three systems: a public web server, security critical system and a user desktop workstation.

A public web server example. A public web server can be classified as an *open access system* (see Step 1 in Section 3.1) and is characterized by a high priority of service availability and integrity with a low focus on confidentiality. A typical example of such a system is an informational web page for a small business. To reflect the priorities of such a server, the following system policy goal values were pre-assigned (see Step 2 in Section 3.1):

availability: 1, confidentiality: 0 and integrity: 0.7

The set of server resources used in our case study includes *file system*, *network interface*, *processor* and *system memory.* The corresponding resource weights for the system policy goals are given in Table 1 (following Step 3 in Section 3.1).

Table 2 provides the details of the responses. The response goodness and response system impact values are computed from the resource impact along with the resource weights from Table 1. The operational cost value for each category is directly assigned based on the maximum expected cost associated with the response. For example, `Delay suspicious process` may not cause any additional labor, but it complicates the follow-up debugging and analysis thus resulting on labor weight =0.6. The `normalized total` of `OC` value is the sum of the category values normalized by the maximum total over all responses.

To evaluate our response cost model, we used a subset of the response measures generated by [8]. Specifically, the following set of responses was considered:

- `Allow read/write dummy version of file`: This response either makes a dummy copy of a file, or filters the file "on the fly" (i.e., a process dynamically blocks certain information from being read from or written to the file) to ensure that only safe content is provided for reading or writing. This action primarily protects the file system resource by preserving integrity of and confidentiality of files. It also provides indirect protection to the CPU, memory, and network interface resources which may be affected by reading or executing malicious content which would have been written to a file. However, deploying this response also changes the system interactions with files, causes additional load on the processor, and may require memory to store modified versions of the file.

- `Backup tampered with files`: Creating a simple backup of files that may be tampered with is an effective way to protect file system data, and possibly intrusion evidence as well. While having virtually no load on the system, this response does carry an operational cost in both resources and labor as it requires storage re-

Resource	Policy Goals		Resource Weight	Overall Resource Weight (W)
	Category	Weight		
`File system`	availability	1.0	0.7	1.4
	confidentiality	0.0	0.2	
	integrity	0.7	1	
`Network interface`	availability	1.0	1.0	1.07
	confidentiality	0.0	0.1	
	integrity	0.7	0.1	
`Processor`	availability	1.0	1.0	1.21
	confidentiality	0.0	0	
	integrity	0.7	0.3	
`System memory`	availability	1.0	1	1.49
	confidentiality	0.0	0.6	
	integrity	0.7	0.7	

Table 1: System Resources

sources and human effort to clean out the repository or restore files.

- `Delay suspicious process`: Delaying a suspicious process can disrupt the timing of an intrusion, often preventing it from functioning. This gives time for deploying additional response measures and offers protection to all system resources by providing a safe time window to complete critical tasks. However, the temporal nature of this response adds considerable complexity to the environment for follow-up forensic analysis, incurring labor operational cost, and also affecting both the processor and, in the event of a network process, network interface.

- `Deploy intrusion analysis tools`: Performing detailed intrusion analysis in an automated manner provides additional recovery and forensics information, and can be used to support better automated decision making. However, it will not offer any system resource protection and can cause additional processor load and network traffic delay due to packet interception and analysis. This response also tends to generate a large amount of data requiring labor to analyze and a significant amount of storage, which is reflected in a high labor and resource values of operational cost.

- `Detailed logging`: This response provides detailed information about system events, giving a substantial advantage in recovery and forensic analysis, and it carries virtually no negative impact to the system. However it does not provide any direct system resource protection. The operational cost of this response involves significant human efforts required to analyze the log files, and considerable storage resources, so it carries a high `OC` value in both labor and resources.

- `Disallow access to file`: This is one of the most effective measures against malicious file access. Similar to `Allow read/write dummy version of file` response, this action protects all considered resources. Due to its predictable behavior and simplicity in implementation, the response impact on the file system and processor is significantly lower. However, the response adds the risk of completely blocking access to a file, and thus, can disrupt the legitimate activity of users. Because the response is likely to require administrator attention, there is a labor operational cost associated with it.

- `Network isolation from specific subnet`: Filtering network traffic from a suspect subnet without completely disrupting traffic flow, prevents further exploitation of the system and effectively protects system resources from attack. At the same time, this response impacts the network interface for the legitimate traffic from that subnet. Eventually this response is likely to carry an operational cost associated with the additional labor for analysis of the blocking action.

- `Complete network isolation`: Filtering all network traffic prevents any network-based attack from being effective, thus, protecting all system resources other than networking, which is made useless. From the operational cost perspective, this action will result in additional labor cost from analyzing the block filter and determining the conditions of network access restoration.

- `Process isolation: different environment`: Process isolation in a different environment (e.g., to a virtual machine, a sandbox) provides many benefits to intrusion detection and response essentially allowing accurate detection of the attack and approximation of the potential damage. However, providing such a simulated environment requires a significant amount of processing power and memory. It also carries significant operational costs through additional resources for storage and operation of the parallel environment, direct monetary expenses if a third-party service is used, and administrator time needed to maintain the virtual environment. Thus the overall operational cost for this response is ranked the highest.

- `Terminate process`: Terminating the suspicious process potentially prevents malicious instructions from being executed entirely. When deployed on fine-grained level (e.g., killing suspicious `MySQL` threads in the event of an SQL-Injection attack), this response can be targeted to minimize the system impact on access. However, it still carries a large risk of preventing legitimate access to the system, especially in case of critical services. From the operational cost point of view, debugging processes killed in the past can be tedious and may require significant labor and system resources.

Response	System Resource			Protected resources	Operational cost	
	Name	Rank	Impact		Category	Weight
Allow read/write dummy version of file	file system	0/2	1	1	direct	0
	network interface	n/a	0	1	labor	0.2
	processor	3/5	0.4	1	resources	0
	system memory	0/2	1	1	normalized total	0.1
Backup tampered with files	file system	n/a	0	1	direct	0
	network interface	n/a	0	0	labor	0.1
	processor	n/a	0	0	resources	0.3
	system memory	n/a	0	0	normalized total	0.2
Delay suspicious process	file system	n/a	0	1	direct	0
	network interface	3/5	0.4	1	labor	0.6
	processor	0/5	1	1	resources	0
	system memory	n/a	0	1	normalized total	0.3
Deploy intrusion analysis tools	file system	n/a	0	0	direct	0
	network interface	4/5	0.2	0	labor	0.9
	processor	1/5	0.8	0	resources	1
	system-memory	n/a	0	0	normalized total	0.95
Detailed logging	file system	n/a	0	0	direct	0
	network interface	n/a	0	0	labor	0.8
	processor	n/a	0	0	resources	0.666
	system memory	n/a	0	0	normalized total	0.733
Disallow access to file	file-system	1/2	0.5	1	direct	0
	network interface	n/a	0	1	labor	0.5
	processor	n/a	0	1	resources	0
	system memory	n/a	0	1	normalized total	0.25
Network-isolation: from specific subnet	file system	n/a	0	1	direct	0
	network interface	1/5	0.8	1	labor	0.7
	processor	n/a	0	1	resources	0.1
	system memory	n/a	0	1	normalized total	0.4
Complete network isolation	file system	n/a	0	1	direct	0
	network interface	0/5	1	0	labor	1
	processor	n/a	0	1	resources	0.1
	system memory	n/a	0	1	normalized total	0.55
Process isolation: different environment	file system	n/a	0	1	direct	0.7
	network interface	n/a	0	1	labor	0.3
	processor	2/5	0.6	1	resources	1
	system memory	1/2	0.5	1	normalized total	1
Terminate process	file system	n/a	0	1	direct	0
	network interface	2/5	0.6	1	labor	0.4
	processor	4/5	0.2	1	resources	0.1
	system memory	n/a	0	1	normalized total	0.25

Table 2: Response Characteristics

Response	**Operational Cost**	**System Resource Impact**	**Response Goodness**	**Response Cost**
`Disallow access to file`	0.25	0.207	1	-0.771
`Terminate process`	0.25	0.262	1	-0.744
`Network isolation from specific subnet`	0.4	0.254	1	-0.673
`Delay suspicious process`	0.3	0.485	1	-0.607
`Allow read/write dummy version of file`	0.1	1	1	-0.45
`Complete network isolation`	0.55	0.317	0.793	-0.359
`Process isolation: different environment`	1	0.436	1	-0.282
`Backup tampered with files`	0.2	0	0.271	-0.171
`Detailed logging`	0.733	0	0	0.367
`Deploy intrusion analysis tools`	0.95	0.35	0	0.65

Table 3: Response Actions Evaluation Results: a public web server

Analysis of response cost. Table 3 demonstrates the response cost `RC` calculation for the set of responses, based on Operational Cost `OC`, Response System Impact `RSI` and Response Goodness `RG`. Generally, the response cost can be viewed as the potential risk of using the response in the system. As such low cost values indicate the responses that are more beneficial for the system, and thus, preferred in case of attack.

For our evaluation we considered response cost measurements in the context of an SQL Injection attack. The attack takes advantage of database vulnerabilities in the application layer. Using incorrect query input through abuse of webpage form fields, or submission of custom HTTP queries, the attacker can perform arbitrary commands on the system [2]. One of the dangers of this attack is the potentially damaging effect on all resources of the system.

As the results show, the most beneficial response in a public web server system is the *Disallow access to file* response (`RC` $=-0.771$). While it has less potential to cause major file system problems and to affect service availability, it still provides a protection to resources from malicious content uploaded to the system.

On the other hand, the responses *Detailed intrusion analysis tools* (`RC` =0.65) and *Detailed-logging* (`RC` =0.367) are ranked as the most costly actions. Though having little negative impact on the system resources and practically no protection against the intrusion, both responses incur high operational cost. Their response cost `RC` valuation reflects this situation.

Another example of operational cost impact can be noted between *Complete network isolation* and *Process isolation different environment* responses. While process isolation is considered more effective based on the protected system resources, network isolation is preferred. This is primarily due to the significant operational cost of process isolation. From a system administrator perspective, it is much easier to analyze a network filter than a complex simulated environment for a process. In addition, the possibility of data integrity loss is much lower due to blocking network access than it is due to specific process migration to different environment.

The response with the largest direct impact to the system, based on a resource analysis, is *Allowing read/write access to a dummy version of the file.* Due to its processor load, the memory requirements, and the potential for file system inconsistencies, this response is the most resource intensive to deploy. However, its low operational cost and universal effectiveness at protecting system resources places it in the middle of the response list. While this response has a smaller impact on data availability than blocking file access does, it has a larger impact on integrity. With the system integrity value of 0.7, malicious updates of the data need to be stopped even at a high cost in other areas. Thus, blocking access to a file entirely is preferred over restricted access.

Considered in the context of a public access system undergoing a powerful SQL Injection attack, the evaluation results show that a relative assessment of responses based on high-level system policy goals and system resources allows administrators to effectively quantify response cost parameters in a way that supports automated analysis and selection of responses against the detected intrusions.

The current evaluation of the responses is based on the analysis of the calculated response cost values in the context of one SQL Injection attack. In the intrusion detection and response system, the evaluation of the applicable response actions is performed through various response selection techniques which alone with the response cost, consider factors such as the likelihood of detected intrusions, the potential damage incurred by the attacks to the system, etc.

Response cost for other types of systems. Similarly to a public web server, we have analyzed the cost of the same response set for two other systems: a classified research system and a user desktop workstation.

A security critical mainly prioritizes data confidentiality and integrity, putting less emphasis on system performance, human resources, and storage. One example of such system is a CIA database server.

On the other hand, *a user desktop workstation* is a typical user station that has high data availability, moderate system performance and data integrity, with data confidentiality and human resources being less critical. Usually the

Response	Classified system				A user desktop station			
	Operational Cost	**System Resource Impact**	**Response Goodness**	**Response Cost**	**Operational Cost**	**System Resource Impact**	**Response Goodness**	**Response Cost**
Disallow access to file	0.25	0.205	1	-0.772	0.25	0.198	1	-0.776
Terminate process	0.25	0.142	1	-0.804	0.25	0.254	1	-0.748
Network isolation: from specific subnet	0.4	0.139	1	-0.73	0.4	0.249	1	-0.676
Delay suspicious process	0.3	0.258	1	-0.721	0.3	0.46	1	-0.62
Allow read/write dummy version of file	0.1	1	1	-0.45	0.1	1	1	-0.45
Complete network isolation	0.55	0.174	0.865	-0.503	0.55	0.311	0.794	-0.364
Process isolation: different environment	1	0.37	1	-0.315	1	0.436	1	-0.282
Backup tampered with files	0.2	0	0.319	-0.219	0.2	0	0.262	-0.162
Detailed logging	0.733	0	0	0.367	0.733	0	0	0.367
Deploy intrusion analysis tools	0.95	0.186	0	0.568	0.95	0.331	0	0.64

Table 4: Response Actions Evaluation Results for Various Systems

latter two priorities are provided by a dedicated support group.

Table 4 demonstrates the response cost calculation for these systems. The ranking of the responses among considered systems has of a lot of similarities: the responses *Detailed intrusion analysis tools* and *Detailed-logging* are considered as the most costly actions, while the *Disallow access to file* and *Terminate process* are ranked as the most beneficial responses. At the same time, there are some interesting differences that should be noted.

Terminate process has a significantly lower system impact on the security critical system than it does on the user desktop since the primary affect resource here is availability. Consequently, this response will be chosen first on a security critical system, but second on a user desktop, where *Disallowing access to a file* is preferred. Intuitively, it is less disrupting for a normal user to have a blocked access to one file, while being able to continue work, than to have request process repeatedly killed. On the other hand, a security critical system puts a priority on protecting system data from unauthorized access, and thus, *Terminate the process* is preferred response here.

Similarly, *Complete network isolation* response is more costly compare to *Allow read/write access to a dummy version of a file*, since it is less costly restrict file access than to completely isolate a user who relies on the network services.

Other interesting thing between these two systems is that the response goodness is higher overall for the security critical system, while the system resource impact tends to be lower. This is mainly due to the fact that most of the considered responses affect resource availability more than confidentiality. Thus, the higher priority of the availability is on a system, the higher cost of the responses will tend to be. However, since the intrusion potentially affects confidentiality, the goodness of the responses tends to be higher on the security critical system, consequently, putting more pressure on deploying responses that are more likely to stop the attack.

6.2 Practical exercise

To evaluate the practical value of our approach, we conducted an experiment where we asked system administrators to rank the set of response actions using their traditional methods according to responses' priority to be deployed on the system in the case of an SQL injection attack.

In the experiment we offered four types of system: *public web server*, *classified*, *medical data repository* and *receptionist workstation* and a set of responses described in Table 2. We recruited 9 system administrators with different level of expertise (5 experts and 4 with moderate level of expertise).

The motivation for the experiment was to evaluate the consistency of the response cost assessment using our methodology in comparison with the traditional approach primarily based on the manual selection of responses according to the administrator expertise.

Surprisingly, the results showed a substantial variability in the response ranking among administrators. The rank order correlation coefficients between any two rankings are in the range of $\{-0.74, 0.15\}$. This means that ranking is not consistent neither among experts, nor among administrators with moderate expertise level, and consequently, varis from the ranking determined by our approach. As one of the responders noted, the response ranking provided by our method characterized a smooth process for system administrators to follow during an attack, while his personal response preference is an overreaction to the situation.

This provides strong testimony that even experienced administrators need a standardized metric for evaluating intrusion responses that would allow to assess the costs involved in each response deployment in a consistent manner. Our approach can be employed by system administrators to guide them through the response selection process

7. CONCLUSION AND FUTURE WORK

In this paper we have presented a comprehensive and structured methodology for evaluation of response cost. The proposed model identifies three main components that constitute response cost, namely, response operational cost, the response goodness in mitigating the damage incurred by the detected intrusion(s) and the response impact on the system.

These response metrics provide a consistent basis for evaluation across systems, while allowing the response cost to be adapted with respect to the security policy and properties of specific system environment. This approach takes advantage of the accuracy inherent in expert assignment of values, and combines it with a structured calculation of relative values, resulting in flexibility and consistency. Importantly, this approach is practically implementable in a real-world environment, making response cost assessment accessible to system administrators with a range of system expertise.

The presented work is the initial step in the direction of establishing standardized response metrics which opens a wide field for future research avenues. One direction we plan to explore is the role of the individual system resource characteristics in the overall resource value assessment. A second potentially fruitful direction is the comparison of cost factors with economic principles, allowing the application of established economic cost models. Automated refinement of response impact on system resources based on past deployment is another possibility of future work, as is a more structured and rigorous approach to assigning and scaling. Finally, we also plan to focus on experimentation with real system settings which might give a deeper insight into evaluation process advantages.

8. REFERENCES

[1] I. Balepin, S. Maltsev, J. Rowe, and K. Levitt. Using specification-based intrusion detection for automated response. In *Proceedings of the International Symposium on Recent Advances in Intrusion Detection*, 2003.

[2] S. Boyd and A. Keromytis. SQLrand: Preventing SQL injection attacks. In *Proceedings of the 2nd Applied Cryptography and Network Security Conference*, pages 292–304, 2004.

[3] B. Foo, Y.-S. Wu, Y.-C. Mao, S. Bagchi, and E. H. Spafford. ADEPTS: Adaptive intrusion response using attack graphs in an e-commerce environment. In *Proceedings of the International Conference on Dependable Systems and Networks*, pages 508–517, 2005.

[4] M. Jahnke, C. Thul, and P. Martini. Graph based metrics for intrusion response measures in computer networks. In *Proceedings of the 32nd IEEE Conference on Local Computer Networks*, pages 1035–1042, Washington, DC, USA, 2007. IEEE Computer Society.

[5] W. Lee, W. Fan, M. Miller, S. J. Stolfo, and E. Zadok. Toward cost-sensitive modeling for intrusion detection and response. *J. Comput. Secur.*, 10(1-2):5–22, 2002.

[6] D. Schnackenberg, H. Holliday, R. Smith, et al. Cooperative intrusion traceback and response architecture CITRA. In *Proceedings of the IEEE DARPA Information Survivability Conference and Exposition (DISCEX I)*, 2001.

[7] A. Somayaji and S. Forrest. Automated response using system-call delay. In *Proceedings of the USENIX Security Symposium*, 2000.

[8] N. Stakhanova. *A framework for adaptive, cost-sensitive intrusion detection and response system.* PhD thesis, Iowa State University, 2007.

[9] N. Stakhanova, S. Basu, and J. Wong. A cost-sensitive model for preemptive intrusion response systems. In *Proceedings of the 21st International Conference on Advanced Networking and Applications*, pages 428–435, Washington, DC, USA, 2007. IEEE Computer Society.

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

[11] T. Toth and C. Kruegel. Evaluating the impact of automated intrusion response mechanisms. In *Proceedings of the Annual Computer Security Applications Conference*, 2002.

[12] Y.-S. Wu, B. Foo, Y.-C. Mao, S. Bagchi, and E. Spafford. Automated adaptive intrusion containment in systems of interacting services. In *To appear in Journal of Computer Networks*, 2007.