A Taxonomy of Intrusion Response Systems

Natalia Stakhanova Samik Basu Johnny Wong

Department of Computer Science

Iowa State University

Ames, IA 50011 USA

{*ndubrov, sbasu, wong*}@iastate.edu

Abstract

Recent advances in intrusion detection field brought new requirements to intrusion prevention and response. Traditionally, the response to an attack was manually triggered by an administrator. However, increased complexity and speed of the attack-spread during recent years showed acute necessity for complex dynamic response mechanisms. Although intrusion detection systems are being actively developed, research efforts in intrusion response are still isolated. In this work we present taxonomy of intrusion response systems, together with a review of current trends in intrusion response research. We also provide a set of essential fetures as a requirement for an ideal intrusion response system.

1 Introduction

Intrusion detection has been the center of intense research in the last decade owing to the rapid increase of sophisticated attacks on computer systems. It typically refers to a variety of techniques for detecting attacks in the form of malicious and unauthorized activity. In the event an intrusive behavior is detected, it is desirable to take (evasive and/or corrective) actions to thwart attacks and ensure safety of the computing environment. Such counter-measures are referred to as *intrusion response*. Although intrusion response component is often integrated with the intrusion detection system (IDS), it receives considerably less attention than IDS research owing to the inherent complexity in developing and deploying response in an automated fashion. As such, traditionally, triggering an intrusion response is left as a part of the administrator's responsibility requiring high-degree of expertise. In recent years, some commercial IDS made available a small set of automated responses, such as blocking and logging actions [2]. However, with the increase in the complexity of intrusions and with it IDSs, necessity for complex dynamically triggered response strategies becomes obvious.

In this paper we attempt to provide a taxonomy of intrusion response and a review of the current status of the existing intrusion response systems (IRS) classified according to the presented taxonomy. By devising this classification we aim

- *to give researchers a better understanding of the problem.* This taxonomy gives a brief comprehensive overview of the intrusion response field.
- *to expose unexplored areas in the field.* Comparative study of the classifications of existing work shows the research ”gap” in the current state-of-art IRSs. This provides useful insights in the requirements of better and viable intrusion response mechanisms and opens avenues of future research.
- *to provide a foundation for organizing research efforts in the field of intrusion response.* To the best of our knowledge a comprehensive and systematic research classification of intrusion response systems does not exist. Some research works on classifications of IDS mention response mechanisms[11, 24, 3], but do not directly focus on the response part of IDS and therefore, lack the necessary depth. The goal of this paper is to provide a complete taxonomy of the existing intrusion response systems accompanied by representative examples. This paper is the first attempt to organize existing research efforts in this area which as we hope will be extended by other researchers in the future.

It should be noted that the presented taxonomy discusses advantages and limitations of the described response techniques merely to draw researchers' attention to these areas and not to advocate for any particular response mechanism. The reminder of the paper is organized as follows: We present a taxonomy of intrusion response mechanisms in Section 2, followed by the review of the existing research efforts in Section 3. Section 4 discusses current state of the intrusion response field and finally, Section 5 concludes the paper.

2 Taxonomy of Intrusion Response Systems

The general problem of constructing a novel taxonomy is the lack of common terminology. In these cases researchers tend to resort to a descriptive explanation or known term that have meaning close to the described phenomena. Since we face the same vocabulary problem, we attempt to find new terms for the newly described classifications while using terms that were already established in the field.

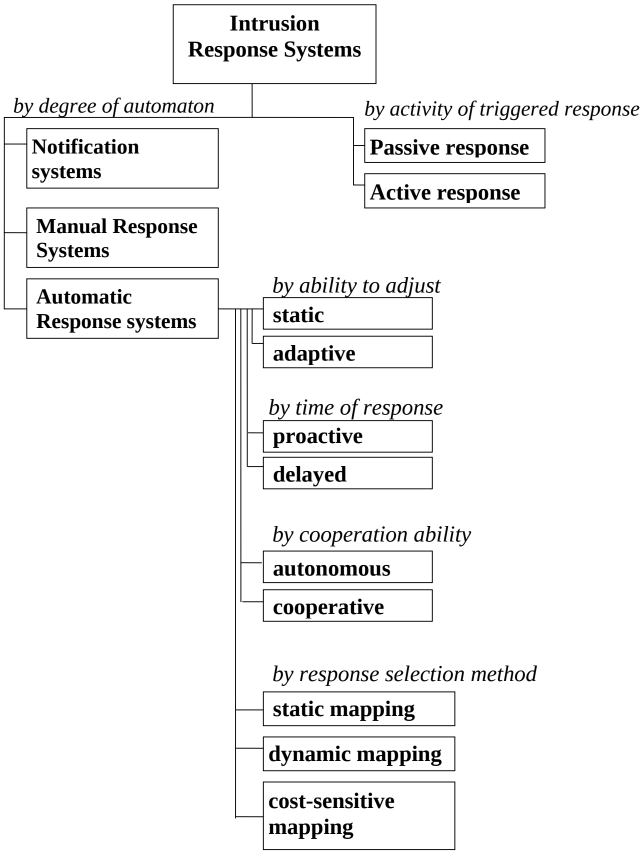

Figure 1: Taxonomy of the intrusion response systems

The proposed taxonomy is given in Figure 1. In the reminder of this section we provide details on each of the categories in the given classification. Intrusion response systems can be classified according to the following characteristics:

- *Activity of triggered response*
  - **Passive:** Passive response systems do not attempt to minimize damage already caused by the attack or prevent further attacks. Their main goal is to notify the authority and/or provide attack information.
  - **Active:** As opposed to passive systems, active systems aim to minimize the damage done by the attacker and/or attempt to locate or harm the attacker.

Passive	**Active**
Administrator notification:	Host-based response actions:
generate alarm	deny full/selective access to file
(through email, online/pager notification, etc.)	delete tampered with file
generate report	allow to operate on fake file
(can contain information about one intrusion	restore tampered with file from backup
such as attack target, criticality, time,	restrict user activity
source IP/user account, description of suspicious packets, etc.	disable user account
as well as intrusion statistics for some period of time)	shutdown compromised service/host
such as number of alarms from each IDS,)	restart suspicious process
attack targets grouped by IP etc)	terminate suspicious process
enable additional IDS	disable compromised services
enable local/remote/network activity logging	abort suspicious system calls
enable intrusion analysis tools	delay suspicious system calls
backup tampered with files	
trace connection for information gathering purposes	Network-based response actions:
	enable/disable additional firewall rules
	restart targeted system
	block suspicious incoming/outgoing network connection
	block ports/IP addresses
	trace connection to perform attacker isolation/quarantine[1]
	create remote decoy[1]

Table 1: List of common passive and active intrusion responses.

Majority of the existing intrusion detection systems provide passive response. Among 20 IDS evaluated by Axelsson [3], 17 systems supported passive response while only 3 systems were designed to mitigate the damage or harm the attacker. Table 1 gives an overview of the passive and active approaches used in the existing response systems.

- *Level of automation* The classification according to the level of automation has been presented in early works by several authors [25, 7, 19]. However, employing only these categories gives a very broad view of the response systems and hence does not provide enough information about the existing research efforts. The taxonomy presented here, on this categorization, also includes additional principles that emphasize differences between various existing approaches.

  – **Notification systems:** Notification systems mainly provide information about the intrusion which is then used by system administrator to select an intrusion response. Majority of the existing IDS provide notification response mechanism.

[1] Borrowed from [27]

- **Manual response systems:** Manual response system provides higher degree automation than notification-only systems and allows system administrator to launch an action from a predetermined set of responses based on the reported attack information.
- **Automatic response systems:** As opposed to manual and notification approaches, automatic response systems provide immediate response to the intrusion through automated decision making process. Although intrusion detection systems are greatly automated nowadays, automatic intrusion response support is still very limited.
  - *Ability to adjust*
    - **Static:** Majority of the IRS are static as the response selection mechanism remains the same during the attack period. These systems can be periodically upgraded by the administrator, however, such support is manual and often delayed till the moment when considerable amount of intrusions exposes the inadequacy of current response mechanism. Although this approach takes a conservative view of the system and environment, it is simple and easy to maintain.
    - **Adaptive:** The adaptability of the response is an ability of system to dynamically adjust response selection to the changing environment during the attack time. Adaptation capability can be represented in several ways including (a) adjustment of system resources devoted to intrusion response such as activation of additional IDS, or (b) consideration of success and failure of responses previously made by the system. The latter can refer to both detection and response mechanisms. Failure of the response can be due to the mistake of IDS that falsely flagged normal activity as intrusion or due to the mistake of IRS that triggered an inappropriate response.
  - *Time instance of the response*
    - **Proactive (preemptive):** Proactive response system allows to foresee the incoming intrusion before the attack has effected the resource. Such prediction is generally hard and often relies on the probability measures and analysis of current user/system behavior. Proactiveness of the response also requires that the detection and response mechanisms are tightly-coupled such that responses can be fired as soon as a likelihood of attack is identified. Although proactive detection of the attack and early response is a desired feature, it is often hard to guarantee 100% correctness of the triggered action. The trade-off between the cor-

rectness of the attack detection and timely response to the possible attack is an inherent characteristic of intrusion response systems.

  - **Delayed:** The response action is delayed until the attack has been confirmed. Such assurance may be provided through the confidence metrics of IDS or full match of the intrusive trace with an existing attack signature. Although, majority of the existing systems use delayed response approach, it may not be suitable for safety-critical systems. For example, for systems relying on checkpoints as fault tolerance mechanism, a delayed response might lead to inability of a system to roll back to the safe state.

    Generally, the delayed response leaves more time to the attacker, consequently allowing more damage to occur and therefore putting the greater burden of system recovery on the system administrator.

  The proactive and delayed intrusion responses have also been considered by several researchers [9, 5] as *incident prevention* and *intrusion handling* of intrusion response respectively. Proactive response is merely an incident prevention that takes place before attack has succeeded, while delayed response is intrusion handling that is performed after the intrusion and includes actions to restore system state. While these two steps should be performed sequentially to provide full system defense and repair, often systems fall back into one of approaches.

∗ *Cooperation capabilities*

  - **Autonomous:** Autonomous response systems handle intrusion independently at the level it was detected. As such, a host-based IDS detecting an intrusion on a single machine will trigger a local response action such as terminating a process, shutting down the host, etc.

  - **Cooperative:** Cooperative response systems refer to a set of response systems that combine efforts to respond to an intrusion. Cooperative system can consist of several autonomous systems that are capable of detecting and responding to intrusions locally, however the final or even additional response strategy is determined and applied globally. Often network IRS are built in such cooperative manner. It allows to achieve better performance in terms of speed of the response and volume of the contained damage. Although cooperative systems provide more effective response than autonomous systems alone, they are also more complex requiring strong coordination and communication among their components.

∗ *Response selection mechanism.* A step into distinguishing various response selection principles was taken by Toth et al.[25, 24]. The authors noted that majority of the existing approaches use static mapping tables or rules based dynamic engines which we define as static and dynamic mapping approaches.

- **Static mapping:** Static mapping systems are essentially automated manual response systems that map an alert to a predefined response. For example, alert about attack on a host can trigger dropping incoming/outgoing network packets. These systems are easy to build and maintain. However, they are also predictable and therefore, vulnerable to intrusions, in particular, denial-of-service attacks. Another weakness of the static mapping systems is their inability to take into account the current state of the whole system. In static mapping systems the triggered response actions present isolated efforts to mitigate the attacks without considering current condition and the impact on other services and system in general. Additionally, as it has also been noted by Toth[25], this approach seems to be infeasible for large systems where the volume of threat scenarios to be analyzed and the constant changes in system policies make the process of building such decision tables cumbersome and prone to errors.

- **Dynamic mapping:** Dynamic response mapping systems are more advanced than static mapping systems as the response selection is based on the certain attack metrics (confidence, severity of attack, etc). In the dynamic mapping setting an intrusion alert is associated with a set of response actions. The exact action is chosen in real-time based on the characteristics of the attack. Generally the selection mechanism for an alert can be presented by a set of ”if-then” statements. For example,

```
if unauthorized user gains access to the password file then
        if confidence of attack is greater than 50% then
                disable user account and restore password file from backup
        if confidence of attack is smaller than 50% then
                give a fake password file
```

  Generally, by adjusting attack metrics we can provide more flexibility in intrusion response selection. For example, attacks with low confidence and severity level can be ignored; moderately severe intrusions with low certainty can be traced while high severity attacks can be responded with appropriate actions. Although this approach can still be potentially exploited by an adversary, it provides much more fine-grained control in response to an attack.

- **Cost-sensitive:** Cost-sensitive response systems are the only response systems that attempt to balance intrusion damage and response cost. The optimal response is determined based on the cost-sensitive model that incorporates several cost and risk factors. Usually these factors are divided into factors related to the intrusion such as damage cost and factors characterizing response part such as response action cost. Accurate measurement of these factors is one of the challenges in using these cost models. Numeric values such as monetary values, probabilistic measurement or percentages that correspond to some objective metrics are not always suitable, as more effective solution based on relative measurements can be applied [17]. The relative measurements can be contracted based on organization security policies, risk factors, etc[12]. One of the downsides of this approach is the necessity to update cost factor values with time. In most cases it is done manually which also puts additional burden on the system administrator.

3 Examples

In this section we will discuss the existing intrusion response systems in relation to the proposed taxonomy.

3.1 Static vs. Adaptive

The response models proposed by Foo et al. [10] and Carver et al.[8, 7, 19] are examples of an adaptive approach. AAIRS, due to [8, 7, 19], provides adaptation through confidence metric associated with each IDS and through success metric corresponding to the response component of the system. The confidence metric indicates the rate of false positive alarms to correct number of intrusions generated by each IDS employed by the system. Similarly, the success metric indicates response actions and response plans that were more successful in the past.

Similar adaptation concept based on the feedback is presented in ADEPTS [10]. In this case effectiveness index, a metric showing effectiveness of a response action against particular attack, is decreased if the action fails. While ADEPTS supports automatic update of the response effectiveness metric, AAIRS requires system administrator intervention after each incident.

Unlike these two solutions, other models considered in Table 2 offer no adaptation support in response mechanism.

3.2 Proactive vs. Delayed

Among the existing response systems presented in the literature, the majority fall into delayed response category. One of the solutions in these models is suspension of the suspicious processes until the intrusion has been confirmed [4, 23]. Such suspension can be temporal until further response strategy is formulated [4] or permanent until the system decides to abort delayed program [23]. Another approach in delayed response is allowing the execution of the suspicious behavior until the observed pattern has matched an intrusive signature [29, 27].

A rare example presented in recent work by Foo et al. [10] investigates a proactive approach to response deployment. The proposed system employs *an intrusion graph* (I-Graph) to model attack goals and consequently to determine possible spread of the intrusion. The mechanism maps alarms provided by the involved IDS to I-Graph nodes and estimates the likelihood of the attack spread based on the alarm confidence values. Finally, appropriate response actions are deployed targeting identified attack goals.

Another proactive handling of response was recently proposed by Locasto et al.[14]. FLIPS, intrusion prevention system, is based on STEM technique[22] that allows to create unique environment for emulation of selected application pieces prior to their real execution. Using this approach for code injected attacks, malicious code can be recognized within a few bytes and prevented from execution.

The Cooperating Security Manager system (CSM) proposed by White et al.[29], although not specifically designed to be proactive, can yield proactive reaction to intrusive behavior in certain cases. This is a distributed approach that combines individual hosts equipped with CSM. While each host performs a local intrusion detection, it is also responsible for notifying other CSMs about suspicious activity. Clearly, instead of waiting for intrusive activity from a user, notified host can take a proactive action to prevent it. An example of such situation is when attacker attempts to gain unauthorized access to an account by trying different passwords. However, instead of checking all possible passwords on one machine, attacker moves to a different host after each failed attempt. While several unsuccessful logins can raise an alarm, single attempt will not be significant enough to be flagged as suspicious. Therefore, reporting such activity to other CSM hosts allows to detect this attack.

3.3 Autonomous vs. Cooperative

There are several examples of the cooperative response systems in the published literature. One such example, Survivable Autonomic Response Architecture (SARA) [13] was developed as unified approach to coordinate fast automatic response. It consists of several components that function as sensors (information gathering), detectors (analysis of sensor data), arbitrators (selection of appropriate response actions) and responders (implementation of response). These components can be arranged among participating machines in a manner that provides the strongest defense. Thus, each host of the system can be equipped with arbitrator which can provide local intrusion response and at the same time participate in a global response selection strategy.

Another cooperative model is EMERALD - a distributed framework for network monitoring, intrusion detection and automated response proposed by Porras and Neumann[18]. The framework introduces a layered approach allowing to deploy independent monitors through different abstract layers of the network. The response component of the framework is represented by the *resolver* that is responsible for analyzing attack reports and coordinating response efforts. While *resolvers* are responsible for response strategy on their local level, they are also able to communicate with resolvers at other EMERALD layers, participating in global response selection.

The Cooperative Intrusion Traceback and Response Architecture (CITRA) presented in [21] provides an example of cooperative agent-based system. This architecture utilizes neighborhood structure where the information about detected intrusion is propagated back through the neighborhood to the source of the attack and submitted to the centralized authority. The centralized authority, referred to as Discovery Coordinator, finally determines an optimal system response. While the Discovery Coordinator is responsible for coordinating global response, local CITRA agents can issue a local response action on a local intrusion detection report.

All of the cooperative approaches to response selection and deployment tend to be distributed network-oriented systems. While CSM system [29], discussed in the previous section, presents an example of autonomous response system, it is a distributed IDS. CSM system allows hosts to share information and detect intrusive user activity in a cooperative manner, however the response actions are determined and deployed by each machine locally.

Other examples of autonomous response system include [23, 6, 26]. These are host-based systems specifically oriented to handle local intrusion detection and response.

3.4 Static mapping vs. Dynamic mapping

Most tracing techniques fall into static mapping category and automatically respond to an intrusion by tracing it back to the source and applying predetermined response actions [28, 20]. Although automated, these approaches have a spirit of notification intrusion response systems as they mainly report about the intrusion source.

Several recent tracing mechanisms take one step further offering a combination of static and dynamic mapping techniques [27, 21]. TBAIR [27] framework suggests to trace the intrusion back to the source host and dynamically select the suitable response such as remote blocking of the intruder, isolation of the contaminated hosts, etc.

Similar approach was taken by CITRA [21]. This framework integrates network-based intrusion detection, security management systems and network infrastructure (firewalls, routers) to detect the intrusion, trace it back to the source and coordinate local response actions based on the attack report. The response mechanism is based on two factors: certainty and severity of the intrusion. While certainty represents the likelihood that reported event is an intrusion, severity defines potential damage to the system and is mainly based on the policy of the particular site. Depending on the reported certainty and severity values, a response action is chosen from a predetermined set.

While these dynamic techniques rely on the underlying predefined set of responses, as opposed to static mapping techniques, the actual action is determined dynamically based on additional factors specific to the current intrusion attempt (intrusion confidence and severity).

Based on agent architecture SoSMART approach [15] is an example of statically mapped response selection system. User-designed incident cases mapped to the appropriate responses present an available set of response actions. In addition to this response decision set, SoSMART model employs a case-base reasoning (CBR) as an adaptation mechanism that matches current system state to the situations previously identified as intrusive. Based on the past experience an additional set of responses can be selected and deployed. Dynamic addition of the new cases allows CBR system to evolve over time.

The next two discussed approaches also offer static mapping response selection mechanism as they rely on the deployment of the prespecified response actions. Authors of [6, 26] proposed an approach to intrusion detection and response based on the specifications of normal behavior expressed in BMSL (Behavioral Monitoring Specification Language). BMSL specifies system behavior in a finite state machine automata fashion and augments each intrusion specification path with a response action. This action can be represented by invocation of a response function, assignment to a state variable or a set of rules for process isolation.

The pH system developed by Somayaji and Forrest [23] is an intrusion detection and response system. Its detection component is based on the normal behavioral profile of the system consisting of N-gram sequences of system calls. Sequences of calls deviating from the normal behavior are considered anomalous and can be either aborted or delayed. Although, these two response actions are simple and computationally not expensive, authors acknowledge that they are not suitable for all applications and additional response might need to be considered.

3.5 Dynamic mapping vs. Cost-sensitive

CSM [29] and EMERALD [18] are dynamic mapping systems. In both approaches the selection of the response strategy is based on confidence information about detected intrusive behavior produced by the detection component and severity metrics associated with an attack.

Another dynamic mapping technique specifically aimed at intrusion damage control and assessment, DC&A, is proposed by Fisch[9]. DC&A tool contains two primary components: *damage control processor* responsible for actions necessary to reduce or control the damage done by the intruder while the intrusion is still in progress and *damage assessment processor* that performs post-attack measures aimed at system recovery. A specific response action to an intrusion is selected by damage control unit based on a suspicion level of user's activity provided by IDS and from the responses available for the given suspicion level. If user's suspicion level increases with time a different response action can be later selected. After intruder leaves the system, damage assessment processor will determine necessary actions to restore original system state based on final suspicion level associated with the intruder. For example, the assessment procedure can include analysis of log files followed by replacement of the stolen files from backup storage.

One of the most complex dynamic mapping approaches is Adaptive, Agent-based Intrusion response system based on agent architecture (AAIRS) [8, 7, 19]. Framework agents represent the layers of the response process. Intrusion alarms are first processed by the Master analysis agent which computes confidence level and classifies the attack as new or ongoing. This classification is mainly based on the preset decision tables. This information is then passed to the Analysis agent which generates action plan based on the response taxonomy. Authors proposed 6-dimentional taxonomy [8]: timing, type of attack, type of attacker, degree of suspicion, attack implications and environmental constraints. Finally, the Tactics agent decomposes the response plan into specific actions and invokes the appropriate components of the response toolkit. This work mainly presents a foundation for intrusion response system as no specific techniques or algorithms necessary for AAIRS are provided.

Compared to the amount of work published on static and dynamic response selection mechanisms, the category of cost-sensitive selection is relatively small.

The approach to intrusion response proposed by Lee at al. [12] is based on a cost-sensitive modeling of the intrusion detection and response. Three cost factors were identified: *operational cost* that includes the cost of processing and analyzing data for detecting intrusion, *damage cost* that assesses the amount of damage that could potentially be caused by attack and *response cost* that characterizes the operational cost of reaction to intrusion. These factors present the foundation of intrusion cost model, i.e total expected cost of intrusion detection, and consequently provides a basis for a selection of an appropriate response.

Graph-based approach called ADEPTS, Adaptive Intrusion Response using Attack Graphs, as discussed in the previous section, is proposed by Foo et al. [10]. Modeling intrusion using graph approach allows to identify possible attack targets and consequently shows objectives of suitable responses. The response actions for the affected nodes in the graph are selected based on the effectiveness of this response to the particular attack in the past, the disruptiveness of the response to the legitimate users and confidence level that indicates the probability that real intrusion is taking place.

Models proposed by Toth and Kruegel [25] and Balepin el al. [4] not only consider costs and benefits of the response actions, but also attempt to model dependencies between services in the system. Such modeling reveals priorities in response targets and evaluates the impact of different response strategies on dependent services and system.

The approach proposed by Toth and Kruegel [25] is a network-based response mechanism that builds dependency tree of the resources on the network. The proposed algorithm for optimal response selection takes into account *a penalty cost* of a resource being unavailable and *capability of a resource* that indicates the resource performance if specified response strategy is triggered, compared to the situation when all resources are available. Clearly, the set of response actions with the least negative impact on the system (lowest penalty cost) is chosen to be applied in response to the detected intrusion.

Similar approach, based on host-intrusion detection and response, was proposed by Balepin el al.[4]. In this system, local resource hierarchy is represented by a directed graph. Nodes of the graph are specific system resources and graph edges represent dependencies between them. Each node is associated with a list of response actions 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 *the cost of the response action* (sum of the resources that will be affected by the response action), *the benefit of the response* (sum of the nodes, previously affected by intrusion and restored to working state) and *the cost of the node or resource*.

4 Discussion

Development of effective response mechanism for potential intrusions is inherently complex due to the requirement to analyze a number of "unknown" factors in various dimensions: intrusion cause/effect, identification of optimal response, state of the system, maintainability etc. As such, it is absolutely necessary to have a complete understanding of problems that needs to be addressed for developing a smart and effective response system.

This paper presents a comprehensive discussion of various issues of intrusion response methods and classifies different techniques along with their detailed comparison revealing the corresponding advantages and disadvantages. An overview of the research and development of intrusion response system in the last decade is given in Table 2 and can be summarized as follows:

IRS	Year published	Response Selection	Response time	Adjustment ability	Cooperation ability
DC&A [9]	1996	dynamic mapping	delayed	static	cooperative
CSM [29]	1996	dynamic mapping	delayed/proactive	static	autonomous
EMERALD [18, 16]	1997	dynamic mapping	delayed	static	cooperative
BMSL-based response [6, 26]	2000	static mapping	delayed[1]	static	autonomous
SoSMART[15]	2000	static mapping	delayed[2]	static	cooperative
pH [23]	2000	static mapping	delayed	static	autonomous
Lee's IRS [12]	2000	cost-sensitive	delayed	static	autonomous
AAIRS[8, 7, 19]	2000	dynamic mapping	delayed	adaptive	autonomous
SARA [13]	2001	static/dynamic mapping[3]	delayed	static	cooperative
CITRA [21]	2001	static/dynamic mapping	delayed	static	cooperative
TBAIR [27]	2001	static/dynamic mapping	delayed	not defined[4]	cooperative
Network IRS [25]	2002	cost-sensitive	not defined[5]	static	cooperative
Specification-based IRS [4]	2003	cost-sensitive	delayed	static	autonomous
ADEPTS [10]	2005	cost-sensitive	proactive	adaptive	autonomous
FLIPS [14]	2005	static mapping	proactive	static[6]	autonomous

Table 2: Classification of the surveyed systems.

- Recent years have seen increased interest in developing cost-sensitive modeling of response selection. The primary aim for applying such a model is to ensure adequate response without sacrificing the normal functionality of the system under attack. Our survey shows that though a number of response frameworks often offer facilities responsible for these mechanisms, very few works provide the detailed algorithms.

- In terms of response-deployment time, majority of proposed frameworks conservatively invoke responses once the existence of intrusion is a certainty. Though this reduces false-positive response, delayed responses can potentially expose systems to higher level of risk from intrusions with no mechanism for restoring system to its pre-attacked state. Therefore, a few research effort developed proactive response mechanisms to enable early response to intrusions, notably, most of them appeared just recently. It should be also mentioned that developing an optimal proactive response mechanism is difficult as it can prohibitively increase false positives.

- Another elusive characteristic of response systems is adaptiveness. It is a powerful feature required to ensure normal functionality while still providing effective defense against intrusive behavior, and to auto-

[1] Although not clearly described, the approach can be extended to proactive response.

[2] Although use of *case-based reasoning* technique can be adjusted to recognize repetetive attacks in advance.

[3] The authors also mention application of more complex response strategies based on some decision-making process.

[4] Proposed work only describes the general principles of framework.

[5] The paper only presents an algorithm for evaluation of response impact.

[6] Although the approach is called ”hybrid adaptive intrusion prevention”, adaptiveness mainly refers to the detection of future attacks based on the feedback, and hence does not fall into adaptive response selection category

matically deploy different responses on the basis of the current system state. At the same time, adaptiveness brings system into the higher level of complexity and poses new questions such as ”How can we automatically classify a response as success or failure? If the response has failed how can we determine whether the system state changed due to triggered (failed) response or continuance of the attack? How can we separate the beginning of new intrusion and continuance of the old attack?” As such, very few of the existing response mechanisms incorporate adaptation.

- Finally, we have seen the presence of both cooperative and autonomous response systems. Typically, host-based intrusion response techniques are autonomous while cooperative methods are deployed in network IDS. Although techniques presented here are existing research efforts, several commercial products with limited automatic response support are also available today[2, 1]. While the research approaches employ a range of different response selection principles, commercial tools provide only static mapping response as simplest and easily maintainable solution.

An ideal intrusion response system. In light of the above discussion, we see the following features as requirements for an ideal intrusion response system. We claim that these requirements will be the driving factor in identifying various future avenues of research in this domain.

- **Automatic**. The volume and the intensity of the nowadays intrusions require rapid and automated response. The system must be reliable to run without human intervention. Human supervision often brings a significant delay into intrusion handling; the response system alone should have means to contain incurred damage and prevent harmful activity. Although complete automation may not be achievable in practice due to presence of newer and novel intractable intrusions, significant reduction of human effort and expert knowledge is desirable.

- **Proactive**. The modern software systems are built on multiple heterogenously-developed components that have complex interactions with each other. Because of these interactions, intrusions are likely to spread faster in the system, causing more damage. Proactive approach to response is the most practical in intrusion containment.

- **Adaptable**. The presence of multiple components, that constitute a software system, also results in a dynamic environment owing to the complex interactions between components. As such, intrusive behavior can affect systems in a way which is unpredictable. The intrusion response system should be equipped with means to recognize and react to the changes in the dynamic environment.

- **Cost-sensitive**. Response to intrusions in dynamic and complex systems require careful consideration of the trade-offs among cost and benefits factors. A simple basic response action triggered every time certain symptom is observed might be a wasteful effort and can cause more damage.

5 Conclusion

In this paper we presented taxonomy of the intrusion response systems. The proposed taxonomy provides an insight into this important field allowing us to see several unexplored areas for research. This work is the first attempt to organize existing knowledge and provides a foundation for further research in this area. We hope that our work will inspire active research of intrusion response methods and will be further extended by other researchers.

References

[1] Dynamic intrusion response (DIR). Available from ”http://www.enterasys.com/”.

[2] TippingPoint intrusion prevention systems. Available from ”http://www.tippingpoint.com”.

[3] S. Axelsson. Intrusion detection systems: A survey and taxonomy. Technical Report 99-15, Chalmers Univ., March 2000.

[4] I. Balepin, S. Maltsev, J. Rowe, and K. Levitt. Using specification-based intrusion detection for automated response. In *Proceedings of the 6th Int’l Symp on Recent Advances in Intrusion Detection*, 2003.

[5] M. Bishop. *Computer Security: Art and Science*. Addison-Wesley Publishing Co., 2003.

[6] T. Bowen, D. Chee, M. Segal, R. Sekar, T. Shanbhag, and P. Uppuluri. Building survivable systems: An integrated approach based on intrusion detection and damage containment. In *Proceedings, IEEE DARPA Information Survivability Conference and Exposition (DISCEX I)*, 2000.

[7] C. Carver, J. M. Hill, and J. R. Surdu. A methodology for using intelligent agents to provide automated intrusion response. In *Proceedings of the IEEE Systems, Man, and Cybernetics Information Assurance and Security Workshop, West Point, NY, June 6-7, 2000*, pages 110–116, 2000.

[8] C. Carver and U. Pooch. An intrusion response taxonomy and its role in automatic intrusion response. In *Proceedings of the 2000 IEEE Workshop on Information Assurance and Security*, June 2000.

[9] E. Fisch. *A Taxonomy and implementation of automated responses to intrusive behavior*. PhD thesis, Texas A&M University, 1996.

[10] 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 2005 International Conference on Dependable Systems and Networks*, pages 508–517, 2005.

[11] P. Kabiri and A. A. Ghorbani. Research on intrusion detection and response: A survey. In *International Journal of Network Security*, volume 1, 2005.

[12] W. Lee, W. Fan, M. Millerand, S. Stolfo, and E. Zadok. Toward cost-sensitive modeling for intrusion detection and response. In *Journal of Computer Security*, volume 10, 2000.

[13] S. M. Lewandowski, D. J. V. Hook, G. C. O'Leary, J. W. Haines, and L. M.Rossey. SARA: Survivable autonomic response architecture. In *DARPA Information Survivability Conference and Exposition II*, 2001.

[14] M. E. Locasto, K. Wang, A. D. Keromytis, and S. J. Stolfo. FLIPS: Hybrid adaptive intrustion prevention. In *Recent Advances in Intrusion Detection (RAID)*, 2005.

[15] S. Musman and P. Flesher. System or security managers adaptive response tool. In *DARPA Information Survivability Conference and Exposition II*, 2000.

[16] P. G. Neumann and P. A. Porras. Experience with EMERALD to date. In *First USENIX Workshop on Intrusion Detection and Network Monitoring*, pages 73–80, 1999.

[17] T. R. Peltier. *Information Security Risk Analysis*. Auerbach Publications, 2001.

[18] P. Porras and P. Neumann. EMERALD: event monitoring enabling responses to anomalous live disturbances. In *Proceedings of the 1997 National Information Systems Security Conference*, 1997.

[19] D. Ragsdale, C. Carver, J. Humphries, and U. Pooch. Adaptation techniques for intrusion detection and intrusion response system. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics at Nashville, Tennessee*, pages 2344–2349, 2000.

[20] D. Schnackenberg, K. Djahandari, and D. Sterne. Infrastructure for intrusion detection and response. 2000.

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

[22] S. Sidiroglou, M. E. Locasto, S. W. Boyd, and A. D. Keromytis. Building a reactive immune system for software services. In *Proceedings of the USENIX 2005 Annual Technical Conference*, 2005.

[23] A. Somayaji and S. Forrest. Automated response using system-call delay. In *Proceedings of the 9th USENIX Security Symposium*, 2000.

[24] T. Toth. *Improving Intrusion Detection Systems*. PhD thesis, Technical University of Vienna, 2003.

[25] T. Toth and C. Kruegel. Evaluating the impact of automated intrusion response mechanisms. In *ACSAC '02: Proceedings of the 18th Annual Computer Security Applications Conference*, 2002.

[26] P. Uppuluri and R. Sekar. Experiences with specification-based intrusion detection. In *RAID ’00: Proceedings of the 4th International Symposium on Recent Advances in Intrusion Detection*, 2001.

[27] X. Wang, D. S. Reeves, and S. F. Wu. Tracing based active intrusion response. In *Journal of Information Warefare*, volume 1, 2001.

[28] X. Wang, D. S. Reeves, S. F. Wu, and J. Yuill. Sleepy watermark tracing: an active network-based intrusion response framework. In *Sec ’01: Proceedings of the 16th international conference on Information security: Trusted information*, pages 369–384, 2001.

[29] G. White, E. Fisch, and U. Pooch. Cooperating security managers: A peer-based intrusion detection system. In *IEEE Network*, volume 10, pages 20–23, 1996.