# A taxonomy of intrusion response systems

Natalia Stakhanova*, Samik Basu and Johnny Wong

Department of Computer Science,
Iowa State University,
Atanasoff Hall, Ames, Iowa 50011, USA
E-mail: ndubrov@cs.iastate.edu
E-mail: sbasu@cs.iastate.edu
E-mail: wong@cs.iastate.edu
*Corresponding author

**Abstract:** Recent advances in the field of intrusion detection brought new requirements to intrusion prevention and response. Traditionally, the response to an attack is manually triggered by an administrator. However, increased complexity and speed of the attack-spread during recent years show 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 a taxonomy of intrusion response systems, together with a review of current trends in intrusion response research. We also provide a set of essential features as a requirement for an ideal intrusion response system.

**Keywords:** information and computer security; intrusion response; taxonomy.

**Reference** to this paper should be made as follows: Stakhanova, N., Basu, S. and Wong, J. (2007) 'A taxonomy of intrusion response systems', *Int. J. Information and Computer Security*, Vol. 1, No. 1/2, pp.169–184.

**Biographical notes:** Natalia Stakhanova has graduated with honours from Moscow Open Social University, Russia and is currently in the final stages of her PhD degree at Iowa State University, Ames, Iowa. Her primary research interests include specification-guided and anomaly-based intrusion detection and adaptive intrusion response mechanisms.

Samik Basu is an Assistant Professor in the Department of Computer Science at Iowa State University, Ames, Iowa. He received his PhD in Computer Science from the State University of New York at Stony Brook in 2003. His primary research area is formal verification of systems exhibiting infinite-state behaviour. He also actively participates in projects that involve application of formal (logic and/or automata-based) methods to intrusion detection, web service composition and diagnosis/synthesis of controllers for discrete event systems.

Johnny Wong is a Professor and Associate Chair of the Computer Science Department, Iowa State University in Ames, Iowa USA. He received his PhD in Computer Science from the University of Sydney, Australia. His research interests include operating systems, distributed systems, computer networks, multimedia systems, intrusion detection and protection, wireless ad-hoc networks and peer-to-peer systems. Most of his research projects are funded by government agencies and industries. Recently, he was involved in a project on Intelligent Multi-Agents for Intrusion Detection and Countermeasures funded by the US Department of Defense (DoD) and a project on Database Generating

Copyright © 2007 Inderscience Enterprises Ltd.

and X-Ray Displaying on the World Wide Web Applications funded by the Mayo Foundation. Currently, he is working on the NSF SFS Programme on Information Assurance and a research project on Endoscopic Multimedia Information Systems, all funded by the US National Science Foundation (NSF). He also worked on projects for Intrusion Detection Systems on Wireless Networks Security and Specification Based Intrusion Detection Systems. He has served as a member of programme committees of various international conferences on intelligent systems and computer networking. He has published over 100 papers in peer reviewed journals and conferences. He is a member of IEEE Computer Society and ACM.

## 1 Introduction

Intrusion detection has been at the centre of intense research in the last decade owing to the rapid increase of sophisticated attacks on computer systems. Typically, intrusion detection refers to a variety of techniques for detecting attacks in the form of malicious and unauthorised activity. When intrusive behaviour 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 the 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 part of the administrator's responsibility requiring a high-degree of expertise. In recent years, some commercial IDS have provided a limited set of automated responses, such as blocking and logging actions (TippingPoint, 2006). However, with the increase in the complexity of intrusions and, with it, the IDSs, the necessity for complex and automated response strategies has become 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 (IRSs) classified according to the presented taxonomy. By devising this classification we aim

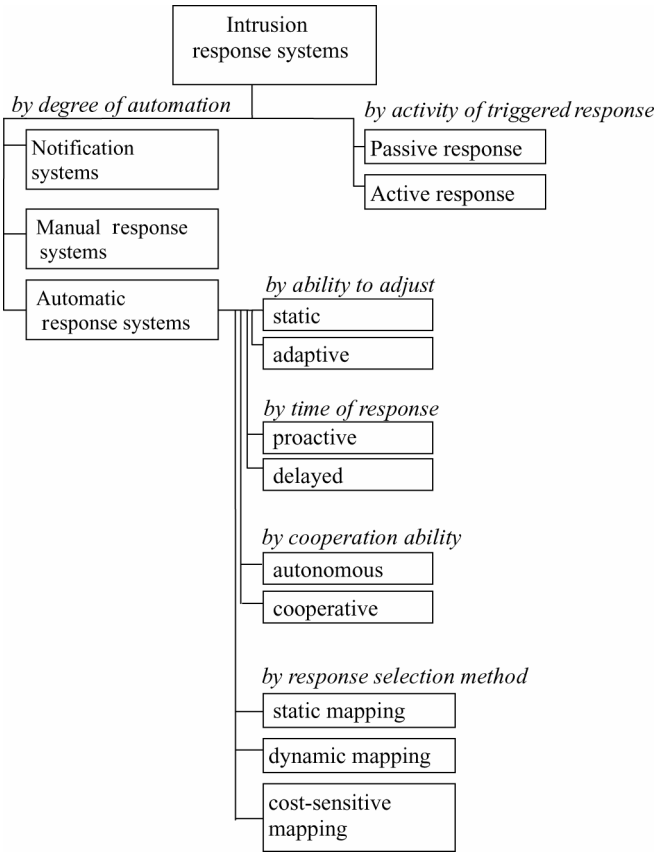
- *to give researchers a better understanding of the problem*. This taxonomy provides a brief comprehensive overview of the intrusion response field.
- *to expose unexplored areas in the field*. Comparative study of the classifications of existing work shows the research 'gap' in the current state-of-art IRSs. This provides useful insight into the requirements of better and viable intrusion response mechanisms and opens avenues of future research.
- *to provide a foundation for organising research efforts in the field of intrusion response*. A comprehensive and systematic classification of intrusion response systems does not exist to the best of our knowledge. Some research on classification of IDS mentions response mechanisms (Axelsson, 2000; Toth, 2003; Kabiri and Ghorbani, 2005), but does not directly focus on the response part of IDS and therefore, lacks the necessary depth. The goal of this paper is to provide a complete taxonomy of existing intrusion response systems accompanied by representative examples. This paper is the first attempt to organise 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 organised 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 has meaning close to the described phenomena. Since we face the same vocabulary problem, we attempt to find new terms for newly described classifications while using terms that are 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 minimise 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 minimise the damage done by the attacker and/or attempt to locate or harm the attacker. The majority of the existing IDS provide passive response. Among 20 IDS evaluated by Axelsson (2000), 17 systems supported passive response while only three 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.

**Table 1** List of common passive and active intrusion responses

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

†Borrowed from (Wang, Reeves and Wu, 2001).

- *Level of automation*: The classification according to the level of automation has been presented in early works by many authors (Carver, Hill and Surdu, 2000; Ragsdale et al., 2000; Toth and Kruegel, 2002). 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 categorisation, also includes additional principles that emphasise the differences among various existing approaches.

  *Notification systems*: Notification systems mainly provide information about the intrusion, which is then used by the system administrator to select an intrusion response. The majority of existing IDSs provide notification response mechanisms.

*Manual response systems*: Manual response systems provide a higher degree automation than notification-only systems and allow the system administrator to launch an action from a pre-determined 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 an automated decision-making process. Although today intrusion detection systems are greatly automated, automatic intrusion response support is still very limited.

- *Ability to adjust*

  *Static*: The majority of IRSs 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 until the moment when a considerable number of intrusions exposes the inadequacy of the 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 the system to dynamically adjust the response selection to the changing environment during the attack time. Adaptation capability can be represented in many 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 an IDS that falsely flagged normal activity as intrusive or due to an IRS that triggered an inappropriate response.

- *Time instance of the response*

  *Proactive (preemptive)*: Proactive response systems allow to foresee the incoming intrusion before the attack has effected the resource. Such prediction is generally hard and often relies on probability measures and analysis of current user or system behaviour. Proactiveness of the response also requires that the detection and response mechanisms are tightly-coupled such that responses can be fired when the attack is identified. Although proactive detection of the attack and early response is a desirable feature, it is often hard to guarantee 100% correctness of the triggered response action. The trade-off between the correctness of the attack detection and a 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 the IDS or the full match of the intrusive trace with an existing attack signature. Although, the majority of 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 make the system incapable of rolling back to a safe state.

  Generally, the delayed response leaves more time for 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 many researchers (Fisch, 1996; Bishop, 2003) as *incident prevention* and *intrusion handling* of intrusion response, respectively. Proactive response is merely an incident prevention that takes place before an 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 defence and repair, often systems fall back into one of these approaches.

- *Cooperation capabilities*

  *Autonomous*: Autonomous response systems handle intrusions independently at the level they are 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 and so on.

  *Cooperative*: Cooperative response systems refer to a set of response systems that combine efforts to respond to an intrusion. Cooperative systems can consist of many autonomous systems that are capable of detecting and responding to intrusions locally, however the final or additional, response strategy is determined and applied globally. Often, network IRS are built in such a cooperative manner. Such systems achieve better performance in terms of response speed and contained damage volume. 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. (Toth and Kruegel, 2002; Toth, 2003). The authors noted that the majority of the existing approaches use static mapping tables or rule-based dynamic engines, which we define as static and dynamic mapping approaches, respectively.

  *Static mapping*: Static mapping systems are essentially automated manual response systems that map an alert to a pre-defined response. For example, detecting an attack on a host can trigger the dropping of 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 represent 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 and Kruegel, 2002), this approach seems to be infeasible for large systems where the number of threat scenarios to be analysed 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 setting of dynamic mapping 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 unauthorised 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 compare to static mapping approach. For example, attack alerts 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 potentially be exploited by an adversary, it provides much more fine-grained control over the 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 characterising the 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 might not always be suitable, as more effective solutions based on relative measurements can be applied (Peltier, 2001). The relative measurements can be contracted based on organisation security policies, risk factors, etc. (Lee et al., 2002). One of the downsides of this approach is the need to update cost factor values over time. In most cases this is done manually which 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. The considered approaches are summarised in Table 2.

### *3.1 Static vs. Adaptive*

The response models proposed by Foo et al. (2005) and Carver et al. (2000) are examples of adaptive approaches. AAIRS, due to (Carver and Pooch, 2000; Carver, Hill and Surdu, 2000; Ragsdale et al., 2000), provides adaptation through confidence metric associated with each IDS and through a success metric corresponding to the response component of the system. The confidence metric indicates the ratio of false positive alarms to the correct number of intrusions generated by each IDS employed by the system. Similarly, the success metric quantifies response actions and response plans that were more successful in the past.

**Table 2** Classification of the surveyed systems

| *IRS* | *Year published* | *Response selection* | *Response time* | *Adjustment ability* | *Cooperation ability* |
|---|---|---|---|---|---|
| DC&A (Fisch, 1996) | 1996 | dynamic mapping | delayed | static | cooperative |
| CSM (White et al., 1996) | 1996 | dynamic mapping | delayed/proactive | static | autonomous |
| EMERALD (Porras and Neumann, 1997) | 1997 | dynamic mapping | delayed | static | cooperative |
| BMSL-based response (Bowen et al., 2000; Uppuluri and Sekar, 2001) | 2000 | static mapping | delayed† | static | autonomous |
| SoSMART (Musman and Flesher, 2000) | 2000 | static mapping | delayed‡ | static | cooperative |
| pH (Somayaji and Forrest, 2000) | 2000 | static mapping | delayed | static | autonomous |
| Lee's IRS (Lee et al., 2002) | 2000 | cost-sensitive | delayed | static | autonomous |
| AAIRS (Carver et al., 2000; Ragsdale et al., 2000) | 2000 | dynamic mapping | delayed | adaptive | autonomous |
| SARA (Lewandowski et al., 2001) | 2001 | static/dynamic mapping¶ | delayed | static | cooperative |
| CITRA (Schnackenberg et al., 2001) | 2001 | static/dynamic mapping | delayed | static | cooperative |
| TBAIR (Wang, Reeves and Wu, 2001) | 2001 | static/dynamic mapping | delayed | not defined§ | cooperative |
| Network IRS (Toth and Kruegel, 2002) | 2002 | cost-sensitive | not defined†† | static | cooperative |
| Specification-based IRS (Balepin et al., 2003) | 2003 | cost-sensitive | delayed | static | autonomous |
| ADEPTS (Foo et al., 2005) | 2005 | cost-sensitive | proactive | adaptive | autonomous |
| FLIPS (Locasto et al., 2005) | 2005 | static mapping | proactive | static‡‡ | autonomous |

†Although not clearly described, the approach can be extended to proactive response.

‡Although use of *case-based reasoning* technique can be adjusted to recognize repetitive attacks in advance.

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

§Proposed work only describes the general principles of the framework.

††The paper only presents an algorithm for evaluation of response impact.

‡‡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 the adaptive response selection category.

A similar adaptation concept based on the feedback is presented in ADEPTS (Foo et al., 2005). In this case, an 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 the delayed response category. One of the solutions in these models is suspending the suspicious processes until the intrusion has been confirmed (Somayaji and Forrest, 2000; Balepin et al., 2003). Such suspension can be temporary until a further response strategy is formulated (Balepin et al., 2003) or permanent if the system decides to abort the delayed program (Somayaji and Forrest, 2000). Another approach in delayed response is allowing the execution of the suspicious behaviour until the observed pattern has matched an intrusive signature (White et al., 1996; Wang, Reeves and Wu, 2001).

A rare example, presented in recent work by Foo et al. (2005), investigates a proactive approach to response deployment. The proposed system employs *an intrusion graph* (I-Graph) to model attack goals and consequently to determine the 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 spreading based on the alarm confidence values. Finally, appropriate response actions are deployed targeting identified attack goals.

Another proactive handling of response, FLIPS, was recently proposed by Locasto et al. (2005). Feedback Learning Intrusion Prevention System (FLIPS), is based on a technique Selective Transactional Emulation (STEM) (Sidiroglou et al., 2005) that creates a unique environment for emulation of selected application pieces prior to their real execution. Using this approach for code injection attacks, malicious code can be recognised within a few bytes and prevented from execution.

The Cooperating Security Manager system (CSM) proposed by White et al. (1996), although not specifically designed to be proactive, can yield proactive reaction to intrusive behaviour 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, a notified host can take a proactive action to prevent it. Such situations arise when an attacker attempts to gain unauthorised access to an account by trying different passwords. However, instead of checking all possible passwords on one machine, the attacker moves to a different host after each failed attempt. While several unsuccessful logins can raise an alarm, a single attempt will not be significant enough to be flagged as suspicious. Therefore, reporting such activity to other CSM hosts allows them to detect and prevent this attack.

### *3.3 Autonomous vs. Cooperative*

There are several examples of cooperative response systems in the published literature. One such example, Survivable Autonomic Response Architecture (SARA)

(Lewandowski et al., 2001) was developed as a unified approach to coordinate fast, automatic responses. 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 defence? Thus, each host of the system can be equipped with an 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 (1997). The framework introduces a layered approach allowing the deployment of independent monitors through different abstract layers of the network. The response component of the framework is represented by a resolver that is responsible for analysing 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 (Schnackenberg et al., 2001) provides an example of cooperative agent-based system. This architecture utilises a neighbourhood structure where the information about detected intrusion is propagated back through the neighbourhood to the source of the attack and submitted to a centralised authority. The centralised 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. In contrast, the CSM system (White et al., 1996) (see Section 3.2) is a distributed IDS equipped with autonomous response mechanism. 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 systems include (Bowen et al., 2000; Somayaji and Forrest, 2000; Uppuluri and Sekar, 2001). 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 the static mapping category and automatically respond to an intrusion by tracing it back to the source and applying pre-determined response actions (Schnackenberg, Djahandari and Sterne, 2000; Wang et al., 2001). Although automated, these approaches have the spirit of notification intrusion response systems as they mainly report about the intrusion source.

Several recent tracing mechanisms take an additional step by offering a combination of static and dynamic mapping techniques (Schnackenberg et al., 2001; Wang, Reeves and Wu, 2001). The TBAIR (Wang, Reeves and Wu, 2001) framework traces the intrusion back to the source host and dynamically selects the suitable response such as remote blocking of the intruder, isolation of the contaminated hosts, etc.

A similar approach was taken by CITRA (Schnackenberg et al., 2001). 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 pre-determined set.

While these dynamic techniques rely on an underlying pre-defined 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).

The SoSMART approach (Musman and Flesher, 2000), based on an agent architecture, is an example of a 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, the 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 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 approaches also offer static mapping response selection mechanism as they rely on the deployment of the pre-specified response actions. Bowen et al. (2000) and Uppuluri and Sekar (2001) proposed an approach to intrusion detection and response based on the specifications of normal behaviour expressed in Behavioural Monitoring Specification Language (BMSL). BMSL specifies system behaviour 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 as a set of rules for process isolation.

The pH system developed by Somayaji and Forrest (2000) is an intrusion detection and response system. Its detection component is based on the normal behavioural profile of the system consisting of N-gram sequences of system calls. Sequences of calls deviating from the normal behaviour are considered anomalous and can be either aborted or delayed. Although, these two response actions are simple and computationally inexpensive, the authors acknowledge that they are not suitable for all applications and additional responses might need to be considered.

### *3.5 Dynamic mapping vs. Cost-sensitive*

CSM (White et al., 1996) and EMERALD (Porras and Neumann, 1997) are dynamic mapping systems. In both approaches the selection of the response strategy is based on confidence information about detected intrusive behaviours 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 (1996). The DC&A tool contains two primary components: a *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 a *damage assessment processor* that performs post-attack measures aimed at system recovery. A specific response action to an intrusion is selected by the damage control unit based on the suspicion level of a user's activity provided by the IDS and from the responses available for the given suspicion level. If the user's suspicion level increases

with time, a different response action can be later selected. After the intruder leaves the system, the 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 the Adaptive, Agent-based Intrusion Response System based on an agent architecture (AAIRS) (Carver and Pooch, 2000; Carver, Hill and Surdu, 2000; Ragsdale et al., 2000). Framework agents represent the layers of the response process. Intrusion alarms are first processed by the Master analysis agent, which computes a confidence level and classifies the attack as new or ongoing. This classification is mainly based on preset decision tables. This information is then passed to the Analysis agent, which generates an action plan based on the response taxonomy. Authors proposed a 6-D taxonomy (Carver and Pooch, 2000) – 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 et al. (2002) is based on a cost-sensitive modelling of intrusion detection and response. Three cost factors were identified: *operational cost* that includes the cost of processing and analysing data for detecting the intrusion, *damage cost* that assesses the amount of damage that could potentially be caused by the attack and *response cost* that characterises 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.

A graph-based approach called ADEPTS, Adaptive Intrusion Response using Attack Graphs, as discussed in the previous section, is proposed by Foo et al. (2005). Modelling intrusions using graphs allows ADEPTS 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 legitimate users and the confidence level that indicates the probability that a real intrusion is taking place.

Models proposed by Toth and Kruegel (2002) and Balepin et al. (2003) not only consider costs and benefits of the response actions, but also attempt to model dependencies among services in the system. Such modelling 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 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 a *penalty cost* of a resource being unavailable and *capability of a resource* that indicates the resource performance if the specified response strategy is triggered, compared with 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.

A similar approach, based on host-intrusion detection and response, was proposed by Balepin et al. (2003). 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 analyse 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 necessary to have a complete understanding of the problems that need to be addressed for developing a smart and effective response system. We summarise below the overview of research and development of intrusion response systems in the last decade (Table 2).

- Recent years have seen increased interest in developing *cost-sensitive modelling* 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 offer facilities responsible for these mechanisms, very few works provide the detailed algorithms.
- In terms of *response-deployment* time, the majority of the proposed frameworks conservatively invoke responses once the existence of intrusion is a certainty. Though this reduces false-positive responses, delayed responses can potentially expose systems to a higher level of risk from intrusions with no mechanism for restoring system to its pre-attacked state. Therefore, a few research efforts 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 diffcult 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 defence against intrusive behaviour, and to automatically deploy different responses on the basis of the current system state. At the same time, adaptiveness brings a system into a higher level of complexity and poses new questions such as '*How can we automatically classify a response as a success or a failure? If the response has failed, how can we determine whether the system state changed due to a triggered (failed) response or a continuance of the attack? How can we separate the beginning of a 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 (DIR, 2006; TippingPoint, 2006). While the research approaches employ a range of different response selection principles, commercial tools provide only static mapping response as a simple, easily maintainable solution.

## 5 Conclusion

This paper presents a taxonomy of intrusion response systems accompanied with representative examples demonstrating the current state of the intrusion response systems and classifies different techniques along with their detailed comparison revealing the corresponding advantages and disadvantages. The taxonomy also provides an insight to the requirements of an ideal intrusion response system and opens avenues of future research in this domain. Specifically, we see the following features as desirable for an ideal intrusion response system:

- *Automatic*. The volume and the intensity of intrusions today 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 novel intractable intrusions, significant reduction of human effort and expert knowledge is desirable.
- *Proactive*. Modern software systems are built on multiple heterogeneously-developed components that have complex interactions with each other. Because of these interactions, intrusions are likely to spread rapidly, causing more damage. A 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 among components. As such, intrusive behaviour can affect systems in a way, which is unpredictable. The intrusion response system should be equipped with means to recognise and react to 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 benefit factors. A simple response action triggered every time certain symptom is observed might be a wasteful effort and can cause more damage.

The outlined requirements form the foundation of comprehensive intrusion response system and will be the driving factors in identifying various future avenues of research in this domain.

## References

Axelsson, S. (2000) 'Intrusion detection systems: a survey and taxonomy', *Technical Report 99–15*, Chalmers University.

Balepin, I., Maltsev, S., Rowe, J. and Levitt, K. (2003) 'Using specification-based intrusion detection for automated response', Paper presented at the *6th International Symposium on Recent Advances in Intrusion Detection*, pp.136–154. In proceedings.

Bishop, M. (2003) *Computer Security: Art and Science*. UK: Addison-Wesley Publishing Co.

Bowen, T., Chee, D., Segal, M., Sekar, R., Shanbhag, T. and Uppuluri, P. (2000) 'Building survivable systems: an integrated approach based on intrusion detection and damage containment', Paper presented at the *IEEE DARPA Information Survivability Conference and Exposition (DISCEX I)*, Vol. 2, Hilton Head Island, SC, USA, pp.84–99. In proceedings.

Carver, C. and Pooch, U. (2000) 'An intrusion response taxonomy and its role in automatic intrusion response', Paper presented at the *2000 IEEE Workshop on Information Assurance and Security*. In proceedings.

Carver, C., Hill, J.M. and Surdu, J.R. (2000) 'A methodology for using intelligent agents to provide automated intrusion response', Paper presented at the *IEEE Systems, Man, and Cybernetics Information Assurance and Security Workshop*, pp.110–116. In proceedings.

DIR (2006) 'Dynamic intrusion response (DIR)', Available at: http://www.enterasys.com

Fisch, E. (1996) *A Taxonomy and Implementation of Automated Responses to Intrusive Behaviour*, PhD thesis, Texas A&M University.

Foo, B., Wu, Y.-S., Mao, Y.-C., Bagchi, S. and Spafford, E.H. (2005) 'ADEPTS: adaptive intrusion response using attack graphs in an e-commerce environment', Paper presented at the *2005 International Conference on Dependable Systems and Networks*, pp.508–517. In proceedings.

Kabiri, P. and Ghorbani, A.A. (2005) 'Research on intrusion detection and response: a survey', *Int. J. Network Security*, Vol. 1, pp.84–102.

Lee, W., Fan, W., Millerand, M., Stolfo, S. and Zadok, E. (2002) 'Toward costsensitive modelling for intrusion detection and response', *Journal of Computer Security*, Vol. 10, pp.5–22.

Lewandowski, S.M., Hook, D.J.V., O'Leary, G.C., Haines, J.W. and Rossey, M.L. (2001) 'SARA: Survivable autonomic response architecture', Paper presented at the *DARPA Information Survivability Conference and Exposition II*, Vol. 1, pp.77–88. In proceedings.

Locasto, M.E., Wang, K., Keromytis, A.D. and Stolfo, S.J. (2005) 'FLIPS: Hybrid adaptive intrusion prevention', Paper presented at the *International Symposium on Recent Advances in Intrusion Detection (RAID '05)*. In proceedings.

Musman, S. and Flesher, P. (2000) 'System or security managers adaptive response tool', Paper presented at the *DARPA Information Survivability Conference and Exposition II*, Vol. 2, p.1056. In proceedings.

Peltier, T.R. (2001) *Information Security Risk Analysis*. New York: Auerbach Publications.

Porras, P. and Neumann, P. (1997) 'EMERALD: event monitoring enabling responses to anomalous live disturbances', Paper presented at the *1997 National Information Systems Security Conference*. In proceedings.

Ragsdale, D., Carver, C., Humphries, J. and Pooch, U. (2000) 'Adaptation techniques for intrusion detection and intrusion response system', Paper presented at the *IEEE International Conference on Systems*, *Man, and Cybernetics*, pp.2344–2349. In proceedings.

Schnackenberg, D., Djahandari, K. and Sterne, D. (2000) 'Infrastructure for intrusion detection and response', Paper presented at the *IEEE DARPA Information Survivability Conference and Exposition*, pp.3–11. In proceedings.

Schnackenberg, D., Holliday, H., Smith, R., Djadhandari, K. and Sterne, D. (2001) 'Cooperative intrusion traceback and response architecture citra', Paper presented at the *IEEE DARPA Information Survivability Conference and Exposition*, pp.56–68. In proceedings.

Sidiroglou, S., Locasto, M.E., Boyd, S.W. and Keromytis, A.D. (2005) 'Building a reactive immune system for software services', Paper presented at the *USENIX 2005 Annual Technical Conference*, pp.149–161. In proceedings.

Somayaji, A. and Forrest, S. (2000) 'Automated response using system-call delay', Paper presented at the *9th USENIX Security Symposium*, pp.185–198. In proceedings.

TippingPoint (2006) 'TippingPoint intrusion prevention systems', Available at: http://www.tippingpoint.com

Toth, T. (2003) *Improving Intrusion Detection Systems*, PhD thesis, Technical University of Vienna.

Toth, T. and Kruegel, C. (2002) 'Evaluating the impact of automated intrusion response mechanisms', Paper presented at the *18th Annual Computer Security Applications Conference (ACSAC '02)*, p.301. In proceedings.

Uppuluri, P. and Sekar, R. (2001) 'Experiences with specification-based intrusion detection', Paper presented at the *4th International Symposium on Recent Advances in Intrusion Detection (RAID '00)*, Springer-Verlag, London, UK, pp.172–189. In proceedings.

Wang, X., Reeves, D.S. and Wu, S.F. (2001) 'Tracing based active intrusion response', *Journal of Information Warefare*, Vol. 1, pp.50–61.

Wang, X., Reeves, D.S., Wu, S.F. and Yuill, J. (2001) 'Sleepy watermark tracing: an active network-based intrusion response framework', Paper presented at the *16th International Conference on Information Security: Trusted Information (Sec '01)*, pp.369–384. In proceedings.

White, G., Fisch, E. and Pooch, U. (1996) 'Cooperating security managers: a peer-based intrusion detection system', *IEEE Network*, Vol. 10, pp.20–23.