Adaptive Intrusion Response to Minimize Risk over Multiple Network Attacks*

Sapon Tanachaiwiwat, Kai Hwang, and **Yue Chen**

University of Southern California

Abstract: A new risk-minimizing security model is presented for distributed network intrusion detection with automated responses. The scheme is based on collective risk assessment of multiple attacks on a network. We propose a new adaptive alarm/defense framework in protecting network resources. We have tested the adaptive security framework on a small Linux cluster at USC. The desired security functions are distributed to routers, switches, firewalls, and hosts. Adaptive response strategies are suggested based on the alarm confidence, attack frequency, assessed risks, and estimated response costs. Dynamic security is achieved by frequent policy update against changing attack patterns or varying network conditions. Measured false alarms and detection rate are used in the armed response decision process. This paper reports the theoretical framework and experimental results obtained. Lessons learned from this network security project are presented.

Categories and Subject Descriptors: K.6.5 [Management of Computing and Information Systems]: Security and Protection, D4.6 [Operating Systems] Security and Protection-Invasive software (e.g., viruses, worms, Trojan horses), Security kernels, information and system security.

General Terms: Network Security, Intrusion Response System, Experimentation, and Performance

Additional Key Words and Phrases: Computer Security, intrusion detection, risk assessment, dynamic security, performance metrics, cost analysis, alarm confidence, residue risk, wasted response.

Table of Contents: (For reviewer's convenience, to be removed later)

Abstract 1
1. **Introduction 2**
2. **Network Intrusion Detection and Response 4**
   - 2.1 Dynamic Network Security
   - 2.2 Attack Types and Characteristics
3. **RADAR: Risk Assessment for Adaptive IDR 7**
   - 3.1 Modeling of The RADAR Framework
   - 3.2 Five IDS Performance Metrics
4. **Risk Assessment and Response Cost Analysis 14**
   - 4.1 Potential Damages and False Alarms
   - 4.2 Response Costs and Residue Risks
5. **Countermeasures and Response Strategies 17**
   - 5.1 Countermeasures Against Intrusive Attacks
   - 5.2 Adaptive Intrusion Response Strategies
   - 5.3 Thresholds for Selecting Response Strategies
6. **Experimental Results and Performance Effects 22**
   - 6.1 USC Experimental Setting
   - 6.2 Response Costs and Residue Risks
   - 6.3 Intrusion Response System Efficiency
7. **Conclusions and Lessons Learned 27**

References 28

---

* Manuscript submitted for exclusive review by ***ACM Transactions on Information and System Security,*** August 19, 2002.
Authors' address: S. Tanachaiwiwat, K. Hwang and Y. Chen, Internet Security and Pervasive Computing Lab., University of Southern California, Los Angeles, CA 90089-2560. Email: (**tanachai, kaihwang, yuec@usc.edu)**. All rights reserved by the co-authors and by the ACM.

1. Introduction

This paper presents a new network security model for dynamic *intrusion detection and response* (IDR) to multiple attacks amid heavy alarms. The effectiveness of this new model is proved by simulated experiments at USC Internet Security and Pervasive Computing Laboratory. Intranets or corporate networks are often exposed to the general public. The hackers often launch the attacks repeatedly until successful. There exists a high demand for tighter security with reasonable costs to cope with multiple and repeated attacks on networks, as indicated by Varadharajan and Katsavos [1996], Sandhu and Sumarati [1997], Kruegul et al [2002], and Ranum [2002]).

Most of today's IDR systems require human intervention in the loop. The slow response to multiple attacks may end up with serious damages beyond recovery. It becomes very difficult to restore to normal network services, if the risk level imposed cannot be reduced, minimized, or eliminated in timely manner (Shieh and Gligor [1997]). The newly proposed IDR scheme is aimed to achieve semi-automated intrusion response to real attacks amid false alarms. The *intrusion detection system* (IDS) raises the *alarm* as a result of the detection of an unknown *attack* launched by the hacker. The alarm raised and the attack incurred may not coincide all the time, depending on the accuracy of the IDS and of the attack type (Millen [2000]).

Extensive research has been reported on the design and evaluation of IDS in the past. Allen et al [2000] at CMU have assessed various intrusion detection technologies. The NSS Group in UK [2001] has evaluated various commercial IDS systems by major security companies. Axelsson [1999] has identified the fallacy and implementation barriers of IDS. High-performance and distributed IDSs have been studied by Malkhi and Reiter [1997], Foster, et al [1998], Seakar et al [1999], Carver and Pooch [2000], Ning, Jajodia, and Wang [2001], and Burroughs et al [2002]. McHugh [2000] has reviewed IDS evaluation from MIT Lincoln Lab (Haines et al [1999]).

A recent study on intrusion alert was reported by Cuppens and Miege [2002]. Gaffney et al [2002] have proposed a decision-theoretic approach to evaluating IDS. Ko and Redmond [2002] and Sheyner, et al [2002] have reported noninterference issues in intrusion detection. Stateful intrusion detection has been reported by Kruegul, et al [2002] and by Shieh and Gligor [1997]. We use existing IDS packages to construct an *intrusion response system* (IRS). The IRS launches countermeasures against the attacks. Before a fully automated IDR system appears in real-time applications, the proposed IRS scheme is useful to network administrators.

Network-based IDS is often installed to protect the corporate Intranets from known attacks under assumed network conditions. As the attack patterns or network conditions change with time, the network IDS may not be able to detect all intrusions successfully. In fact, false alarms may hide the attacks most of the time as pointed out by Newman [2002]. Consequently, the network may face the dilemma of launching ineffective responses to some attacks or wasting resources to deal with false alarms. Our new approach intends to overcome these difficulties.

We choose a dynamic approach to solving the network-security problems. Dynamic means the security policy will change in time. Recently, dynamic network security has been pushed by Meadows and McLean [1999] and by the NAI Labs and Boeing researchers. Petkac and Badger [2000] have studied the security agility in response to intrusion detection. Schnackenberg, et al [2000] have developed the IDIF protocol for building an

automated intrusion response infrastructure. The research reported here extends from our previous work on distributed firewalls and IDR systems (Hwang and Gangadharan [2001] and Hwang [2002]).

Risk assessment for network security was previously studied by Fisch [1996] and much earlier by Sage and White [1980]. We propose a new statistical scheme to assess the risk level of multiple or repeated attacks on an Intranet. Our scheme is called RADAR, standing for *Risk Assessment for intrusion Detection with Automated Response*. This scheme is based on a cost-efficient, alarm/defense matrix framework. This work extends the cost-sensitive IDR model proposed by Lee and Stolfo [2000]. Our framework was implemented with security functions distributed to routers, switches, firewalls, and hosts in an experimental network at USC.

Previously, excellent studies on intrusion response were reported by Carver and Pooch [2000], Lee and Stolfo [2000], Petkac and Badger [2000], Schnackenberg, et al [2000], Wang et al [2001], and West-Brown, et al [2002]. We introduce new concepts of alarm and defense matrices, residue risks, and wasted response to specify a dynamic IRS. We tested the RADAR framework with five attack types simulated on a small Linux cluster at USC. We report the performance with data collected from simulated security experiments conducted in Spring 2002. To benefit the readers, we summarize in Table 1 the notations used in this paper.

Table 1 Basic Notations, Definitions, and Key Equations in the Paper

Notation	Brief Definition	Notation	Brief Definition
F	**Attack frequency,** total number of attacks from multiple sources	G	**Alarm frequency,** total number of alarms raised by the IDS
$A = (a_{ij})$	**Alarm matrix,** where a_{ij} is the probability of an attack of type i detected as type j, Eq.(1)	H	**Detection hit rate,** the probability of detecting an attack hits correctly, Eq.(2)
S	**False alarm rate,** the probability of a confused or false-positive alarm raised, Eq.(5)	M	**Detection miss rate:** probability of misdetecting a real attack as no attack (**False negatives**), Eq.(3)
Q	**Alarm confidence**, the probability of an alarm correctly raised by IDS, Eq.(4)	I	**Alarm interval,** average time between adjacent alarms raised, equal to the total time divided by G.
E_{IDS}	**Intrusion detection system (IDS) efficiency** computed by Eq.(6)	E_{IRS}	**Intrusion response system (IRS) efficiency,** computed by Eq.(10)
R_{max}	**Maximum risk,** maximum damage assessed from multiple attacks	D_{hit}	**Hit damage,** true damages from all detection hits from overhead and delayed responses
$D_{confused}$	**Confused-alarm damage** from all confused alarms at off-diagonal entries in the alarm matrix	D_{miss}	**Miss damage** from all real attacks that were missed and reported at the last column of the alarm matrix
R_a	**Average response cost,** averaged over all possible responsive countermeasures	T_{total}	**Total observation window,** the total time of monitoring period
T_{ave}	**Time to respond** to an attack, averaged over all possible attacks and including all overhead time	W	**Wasted response,** from ineffective responses which were deployed too late or wrongly, Eq.(8)
$R_{residue}$	**Residue risk** after delayed response, confused alarms or misses, Eq.(9)	D	**Average damage** per attack, averaged over all possible attack types in the monitoring window

Because most IDSs are far from being perfect, we classify attacks into true versus false attacks. *True attacks* are real and may post risks. A *false attack* is not real and it may trigger the IDS to raise the wrong alarm, but it will not cause real damages. An alarm can be also true or false. A *true alarm* must coincide with a true attack. *False alarms* can be either positive or negative. A *false-positive alarm* corresponds to false attacks being wrongly detected by the IDS as a true attack. A *false-negative alarm* corresponds to a true attack being missed by the IDS as no attack.

Attacks are faced by the IDS as input. Alarms are generated by the IDS as output. The IRS sees only the alarms and is aware of the attack characteristics. We will deal with both attacks and alarms, but most the time alarm characteristics, such as the frequency, confidence, and impacts of alarms. The terms *risk, threat,* and *damage* will be used interchangeably. The only subtle difference between them is timing. The risks are created by attacks, some may cause real damages, if not timely suppressed. A risk may be eliminated, reduced, or minimized, if an intelligent IRS is designed to work with the IDS.

We will present the theoretical framework of the RADAR scheme first. Then we conduct the risk damage assessment. The assessed risk will be used to select the response strategies in subsequent sections. The effectiveness of the theoretical security model is verified by experimental results at the end. We will provide a deep analysis on IDS and IRS performance effects. The proposed RADAR and IRS techniques are useful to both security administrators and network IDS or IRS designers. Critical research issues are identified and lessons learned are summarized at the end.

2. Network Intrusion Detection and Response

We introduce below three key concepts in securing an Intranet built around a self-protecting LAN. First, we identify the security issues involved. Second, we describe the attack events as either hits (detected correctly) or miss (undetected), or false alarms by the IDS. Thirdly, we investigate response options to be adopted in building a cost-effective IRS.

2.1 Security Requirements in Networks

We desire to achieve dynamic security in a network, in which the protection can be distributed to all network nodes. This offers a distributed security architecture for protecting the network resources automatically, collectively, and cooperatively. Conventional networks are most passive, lacking such active intrusion detection and response capabilities.

Active networks are constructed with active nodes in switches, routers, firewalls, and hosts. Active nodes can execute special programs to achieve higher QoS (*quality of service*), such as network traffic control or security reinforcement. These active nodes perform customized computations on active messages flowing through them. Active messages contain packets, which encapsulate special active codes that are executed in every traversed node.

We consider security infrastructure that supports control of active codes. Active codes are supposed to be enabled by mobile codes for good causes. However, the hackers may inject faulty active codes to launch the attacks. Therefore, one of the major challenges in designing highly secure networks is to build security infrastructure that support the control of active codes. In particular, we focus on three countermeasures, namely port scanning, intrusion detection, and active response.

The IDR code should be designed active to offer the extensibility, effectiveness, utilization, and survivability. For simplicity, we will concentrate on a network with a single administration domain. We characterize in Figure 1 various attacks and possible responses of an IDR system. The purpose to clarify the cause-

consequence relationships in Table 2 among the event sets. These relationships will help introduce our RADAR model in Section 4, to perform the cost analysis in Section 5, and to present the experimental results in Section 6.

In Figure 1, the attacks and alarms form the attack set. This set is observed during a given time window. From the IDS point of view, we classify these attacks into two major categories: The *detected set* consisting of all attack hits and true alarms and the *miss set* consisting of all false alarms or misses. The attack hits are those that have been correctly detected by the IDS as true attacks of certain types. These attacks will cause true damages, once they take effect. A false-positive attack is a real attack or a no-attack, which is wrongly detected by the IDS as a different attack type. A false-negative attack is a real attack that the IDS fails to detect its occurrence and label them as no-attacks. We will consider all types of attacks in our study.

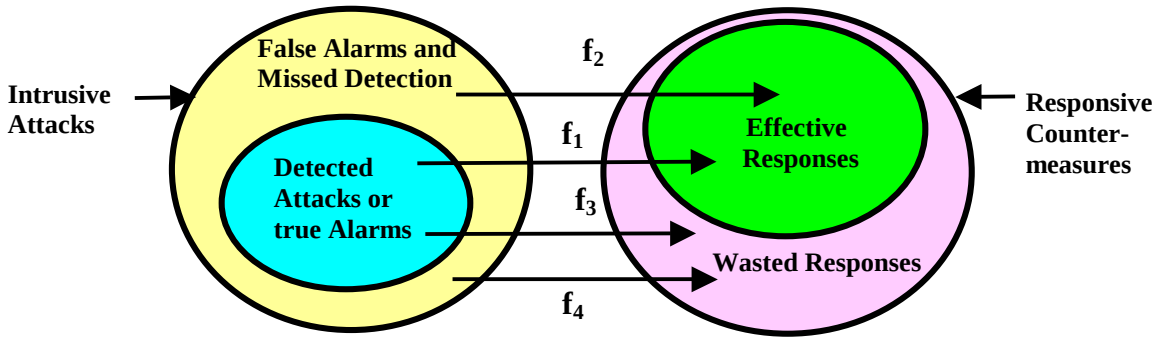

Figure 1 Set-theoretic relationships between intrusive attacks and possible responses, where mappings f_1 and f_2 are desired, but f_3 and f_4 are not wanted.

Response Options: To counteract attacks, we consider both effective and wasted responses. The purpose is to paint a clear picture between malicious attacks and cost-effective responses. The relationship between attacks and responses are represented by 4 mappings shown by arrows in Fig.1. Mappings $\mathbf{f_1}$ and $\mathbf{f_2}$ are needed to implement in our IDS and IRS designs. But mappings $\mathbf{f_3}$ and $\mathbf{f_4}$ are not desired, because they cause waste. This is a fundamental understanding for a security administrator to take timely and effective countermeasures to stop the attacks.

In Figure 1 and Table 2, we show two mappings from the attack set to two categories of response subsets: effective versus wasted responses. An ideal IDR system should be able to counter all detected attacks and alarms as in the mappings $\mathbf{f_1}$ and $\mathbf{f_2}$. An ordinary IDR system responds effectively to only detected attacks but not to false alarms or misses. The response to false alarms may be wasted as seen in $\mathbf{f_4}$. The worst IDR system cannot cope with either attack hits or false alarms properly, thus causing a significant resource waste as seen by the mappings $\mathbf{f_3}$ and $\mathbf{f_4}$. We try to avoid the wasted responses through risk assessment.

Our ultimate goal is to build an automatic IDR system to enforce dynamic security in a network domain. All effective response to an attack requires countermeasures by hardware, operating system, run-time support, etc. Those wasted responses should be minimized or eliminated. Based on today's security requirements, the mapping $\mathbf{f_1}$ is a necessity. The mapping $\mathbf{f_2}$ is rather difficult to realize with today's technology. The best shot is to enforce $\mathbf{f_1}$ completely and to support $\mathbf{f_2}$ partially. The wasted responses may not be avoidable. The total wastes in mappings $\mathbf{f_3}$ and $\mathbf{f_4}$ are not acceptable in a poor IDR design.

Table 2 Ratings of Network Intrusion Detection and Response Systems

Quality and Efficiency of IDS, IRS, and IDR	Effective Response to Attack Hits, (Mapping f_1)	Effective Response to False Alarms and Misses (Mapping f_2)	Wasted Response to Attack Hits (Mapping f_3)	Wasted Response to False Alarms and Misses (Mapping f_4)
Ideal design	**Yes**	**Yes**	**No**	**No**
Ordinary IDR	**Yes**	**No**	**No**	**Yes**
Poor IDR	**Yes**	**No**	**Yes**	**Yes**

2.2 Attack Types and Characteristics

We study four types of network attacks: *Probe, U2R, R2L and DoS* (Haines J. et al.[2001]). **Probe** is the unauthorized intruder's attempt to scan the opening ports (services), OS and Web server programs. This attack would be the first step that leads to another type of attack. After the attacker learns the weakness of the target system, they will know exactly what to exploit the vulnerabilities or even crash the whole system. Presently, available probe tools range from simply scanning ports to reporting everything about the hosts.

U2R (*User-to-root***)** is the attack that users trying to gain privilege of root account. This attack type could cause catastrophic damage since root can modify and delete everything in the system. **R2L (***Remote-to-Local***)** is the attack that intruders use to seek the way to gain access to the system. Since the first step to gain root access usually start with having local user account and crack the root password later. The last type is **DoS (***Denial of Service***)**. The DoS attack aims to disrupt the system services. This is normally accomplished by flooding the system resources to deny other services.

In Table 3, we summarize 5 attack programs: Nimda, CodeRed, Queso, Smurf and BIND that we have applied in attack generation. These programs are chosen in our running example to demonstrate how to apply our RADAR model to repel multiple attacks with minimized risk.

Nimda is identified as U2R or R2L. It is the mass-mailing worm that utilizes multiple methods to spread itself The worm can send it self out by e-mail, search open network shares, attempt to copy itself to unpatched or already vulnerable Microsoft IIS Web Server. The worm will create the Guest account to Windows NT, 2000 and add this account to the "Administer" group. This would enable the intruder to execute the arbitrary command in the local system. Moreover the high scanning rate of Nimda worm would result in bandwidth DoS condition on network of infected machine. Details of the Nimda Worm and its recent attack incidences can be found in the web site: http://www.symantec.com/avcenter/venc/data/w32.nimda.a@mm.html

CodeRed is also another type of Internet worm. It spreads itself by submitting crafted HTTP request to vulnerable IIS Microsoft web server. This worm will deface every webpage of infected machine. Furthermore the exploit that this worm use could make the intruder to execute the arbitrary command in the local system and intruder can fully gain control of infected machine. High random IP scan of this worm can also significantly degrade the network performance. Additional information on the CodeRed program and its recent attack incidences can be found in the web site: http://www.symantec.com/avcenter/venc/data/codred.worm.html

Queso is the Probe attack. It sends a packet to a host and examines the response. Since different operating system would give different response. The attacker will learn the type of operating system (e.g. Linux kernel 2.4.1). The result of this attack would help intruder to select the proper tools or exploited to that specific machine or modify the tools to suit the targets. For details of Queso, visit the site, http://www.cert.org/incident_notes/IN-98.04.html

Smurf is a DoS attack. Ping's packet returns to IP of the targeted machine. This ping is issued to the entire IP broadcast address. This case, every machine will respond to the bogus ping packets and reply to the targeted machine. The targeted machine will be flooded with those responses. This attack causes significant degradation in network performance or even the collapse of the entire network. For details of Smurf, visit the web site: http://securityresponse.symantec.com/avcenter/venc/data/smurf.dos.attack.html

BIND is another DoS attack, which exploits the vulnerabilities in BIND (Berkeley Internet Name Domain System) software. This attack results in shutdown of DNS. The web browser will not be able to locate the websites as a consequence of BIND attack.. For updated incidence report of the BIND attacks, readers are advised to visit the web site: http://securityresponse.symantec.com/avcenter/security/Content/ 2002.06.06.html

Table 3 Five Attacks Types Tested and Their Characteristics

Types	Vulnerable OS	Attack Impacts
Nimda	Windows NT, 2000, XP	Intruders gain control of the local system (R2L, U2R). High scanning rate causes bandwidth depletion (DoS).
CodeRed	Windows 2000, XP, Cisco	Deface webpage, install a backdoor, and cause bandwidth depletion.
Queso	Linux, Sun Solaris, Cisco	A scan attack to explore OS victim. Intruders gain knowledge of how to exploit the targeted system
Smurf	Solaris, Linux	Send forged ICMP echo request packets to flood the victim with DoS causing network to collapse
BIND	Berkeley Internet Name Domain System	BIND server down, unable to locate websites (DoS).

3. RADAR: Risk Assessment for Adaptive IDR

In this section, we introduce the RADAR framework. Then we define the key parameters used in modeling the security system. Basic IDS performance metrics are specified for use in subsequent sections.

3.1 Modeling of The RADAR Framework

As shown in Figure 2, the RADAR framework is built with three functional blocks: the IDS (*intrusion detection system*), the RAS (*risk assessment systems*), and the IRS (*intrusion response system*). The IDS is used to detect intrusions. It will raise the alarm after detecting an attack. The RAS is used to assess the risk levels and estimate potential damages. Risk comes from two different sources: the real attacks and the false alarms. When the IDS reports an attack, we need to estimate how serious the risk or the damage might be. Furthermore, the RAS must be able to distinguish various forms of false alarms or miss detections.

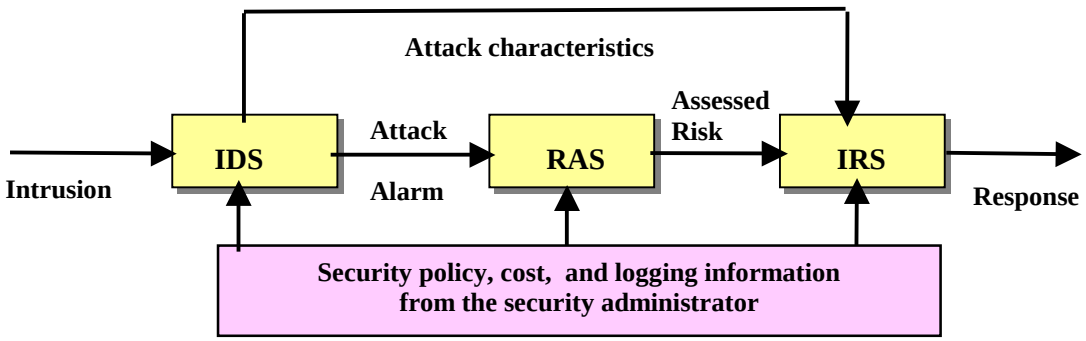

Figure 2 RADAR system for protecting a network from attacks (IDS: intrusion detection system, RAS: risk assessment system, and IRS: intrusion response system)

We define *risk* from an attack in terms of the potential *damage* the attack may cause to the network hosts or to other resources (such as servers and database) within the network domain. As mentioned earlier, we lump all possible loess into a single dollar amount for the sake of clarity and simplicity to convey the main ideas in our RAS and IRS development. Our RADAR system contributes mostly in the RAS and IRS subsystems. We do not claim much new work in the IDS area.

The RADAR scheme is designed to analyze the network IDS reports and to assess the potential damages or residue risk after countermeasures are enabled. Several response strategies are suggested, based on the alarm confidence, attack frequency, potential damages, and response costs assessed. Measured false alarms and detection rates are used in the armed response decision process.

We use a simple running example to introduce all important concepts and procedures involved. However, all formula derived are for general case of $\boldsymbol{n}$ attack types and the IRS is built with $\boldsymbol{m}$ response features. For clarity to illustrate the new ideas, we introduce the theoretical RADAR framework with a simple running example of $\boldsymbol{n}$ = 5 attack types. Alarm matrix and damage vector are used to formulate the risk-assessment and RADAR system. It should be mentioned that none of the existing IDS is perfect. They cannot detect all the attacks correctly without making mistakes or missing some attacks amid large amount of false alarms.

Some of the risk from attacks can be completely removed without causing any damage, if the IRS is properly designed and timely responds to the attacks. However, almost none of the existing IRS was designed fully automatic. Still human must be involved in the intrusion response loop. Thus many responses are either ineffective or too slow to stop the damage. For this reason, residue damages may still exist even after the IRS takes actions. We will analyze all the design and cost-effectiveness tradeoffs. Our work extends beyond the damage cost, and response cost suggested by Lee and Stolfo [2000].

Alarm Matrix: The detection results of an IDS are represented by an *alarm matrix* .

$$\boldsymbol{A} = (\boldsymbol{a}_{ij}) \qquad \textbf{(1)}$$

As shown in Figure 3, $\boldsymbol{A}$ is a square matrix of order $\boldsymbol{n}$+1, where $\boldsymbol{n}$ is the number of different attack types. For $1 \leq i, j \leq \boldsymbol{n}$, the i-th row corresponds to attack type i. The j-th column corresponds to alarm type j detected by the IDS. The matrix element $\boldsymbol{a}_{ij}$ is equal to the number of times that attack type **i** has been detected and triggered

the alarm type j. The alarm matrix is generated by real-world attack/alarm distributions. The attacks are indicated by the rows and alarms by columns.

On Figure 3, the *detection hits* are represented by i = j along the diagonal elements (marked as the ***x*** subdomain). These are the attacks that are correctly alarmed or detected by the IDS. A *confused alarm* occurs when attack type i is wrongly detected as type j, corresponding to all off-diagonal elements ($i \neq j$) in the ***z*** domain**.** A *false negative alarms* correspond to all miss detection at the last column marked as the ***y*** domain. The *false-positive alarms* are at bottom row with i = n+1 for no attacks marked as the ***u*** domain.

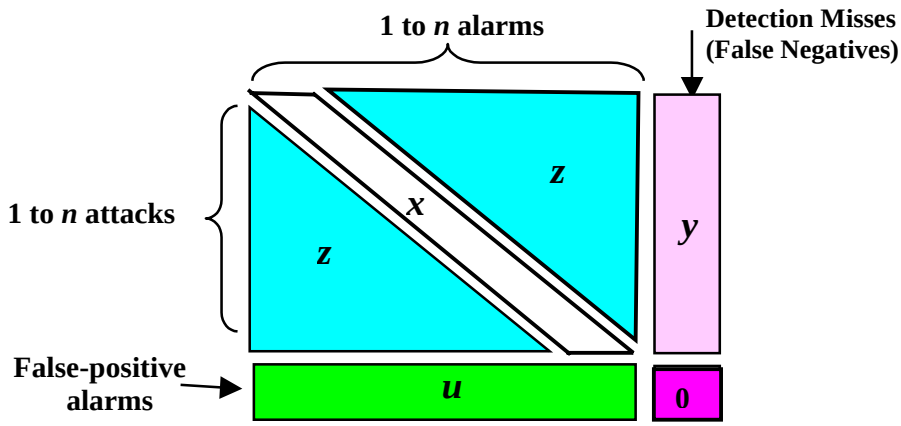

Figure 3 Alarm matrix with 4 subdomains: ***x*** for detection hits, ***y*** for detection misses or false-negative alarms, ***z*** for confused alarms, and ***u*** for false-positive alarms

We define 4 subdivided areas in the alarm matrix as follows. Various combinations of the four-tuple: (***x, y, z, u***) simply reflect different attack/alarm patterns encountered.

$x = \sum_{i=1}^{n+1} a_{ii}$ is the summation of all detection hits along the diagonal of the alarm matrix ***A,***

$y = \sum_{i=1}^{n+1} a_{i,n+1}$ is the summation of all misses or false-negative alarms in the last column,

$z = \sum_{j=1}^{n} \sum_{i=1}^{n+1} a_{ij}$ $(i \neq j)$ is the summation of all confused alarms in the off-diagonal entries , and

$u = \sum_{j=1}^{n} a_{n+1,j}$ is the summation of false-positive alarms in the bottom row of the matrix.

Running Example:

For illustrative purpose, we tested our RADAR model with simulation experiments over the following alarm matrix resulted from data entries extracted from an IDS evaluation report by MIT Lincoln Lab (Haines, et al [2000]). This matrix corresponds 5 attack types in Table 3. The matrix entries

correspond to a month-long monitoring of a simulated network domain on the Linux cluster built at USC Laboratory as explained in Section 6.1.

$$\boldsymbol{A} = \begin{pmatrix} 16 & 0 & 2 & 0 & 0 & 22 \\ 0 & 13 & 0 & 1 & 0 & 10 \\ 2 & 0 & 41 & 3 & 0 & 19 \\ 0 & 0 & 0 & 8 & 0 & 24 \\ 0 & 0 & 0 & 0 & 2 & 4 \\ 5 & 1 & 4 & 5 & 0 & 0 \end{pmatrix}$$

Many of the matrix elements are zero, meaning that no attacks or no alarms raised. Most non-zero entries are along the diagonal with $\boldsymbol{x}$ = 80, meaning a high detection hit rate. The confused alarms are signaled by $\boldsymbol{z}$ = 8 off-diagonal entries of the $\boldsymbol{A}$ matrix. Our alarm matrix is very similar in concept to the *confuse matrix* defined by the MIT group. We have $\boldsymbol{u}$ = 15 false-positive alarms at the bottom row and the detection miss set $\boldsymbol{y}$ = 79 in the rightmost column of $\boldsymbol{A}$. ■

Attack Frequency: The *attack frequency,* $\boldsymbol{F}$, is the summation of all real attacks within a monitoring period: $\boldsymbol{F} = \sum_{i=1}^{n} \sum_{j=1}^{n+1} a_{ij} = x + y + z$. In our example, $\boldsymbol{F}$ = 167 attacks that were launched in two weeks. Note that the false-positive alarms in the subset $\boldsymbol{u}$ are not part of the attack frequency. This measure is often used in IDS studies.

Alarm Frequency: The *alarm frequency* is defined the summation of all raised alarms during a monitoring period. One can calculate $\boldsymbol{G} = \sum_{i=1}^{n+1} \sum_{j=1}^{n} a_{ij} = u + x + z$. In our example, $\boldsymbol{G}$ = 103 alarms were observed in our experiments conducted in two weeks. Note that the misses or false-negative alarms in the subset $\boldsymbol{y}$ are not part of the alarm frequency. This measure is more often used in IRS design. Since our study is focused on intrusion response, we will use the alarm frequency more often than using the attack frequency in subsequent sections.

According to a recent Internet Security Threat Reports (Riptech [2002]), the average number of attack is more than 700 attacks over a six-month period. Our $\boldsymbol{F}$ and $\boldsymbol{G}$ in the running example are fairly close to the reality. These two frequencies have very different meanings. $\boldsymbol{F}$ reflects the number of real attacks, either detected or missed. The $\boldsymbol{G}$ reveals the total number of attacks and false alarms observed. The relative magnitude of the two varies with different IDSs used. An IDS may be cautiously designed to over kill with many false alarms such that $\boldsymbol{G} > \boldsymbol{F}$. An ineffective IDS may have missed many attacks such that $\boldsymbol{G} < \boldsymbol{F}$. Our running example falls into the second case with mixed distribution in detection hit rate and miss rate as shown in subsection 3.2.

Alarm Probability: From the alarm matrix $\boldsymbol{A}$, we can define a similar matrix to indicate the probability of each attack event a_{ij} as $\boldsymbol{P} = (p_{ij})$. $\boldsymbol{P}$ is a square matrix of order $\boldsymbol{n}$+1. The matrix element p_{ij} is the probability of an attack type **i** detected as type **j**. Thus, we have $p_{ij} = a_{ij} / f_i$, where $f_i = \sum_{j=1}^{n+1} a_{ij}$. Note that the entry $p_{n+1,n+1} = 0$, because nothing is detected for no attacks. This implies that IDS should be smart enough not to treat a no-attack as a false

alarm. The *probability matrix* represents the past attack statistics. This matrix entries are often used in building an IRS, when new attack statistics are unknown to the designer. But as the system gets deployed, the designer may want to update the design parameters often to reflect more recent attacking incidences.

3.2 Five IDS Performance Metrics

The quality of an IDS is rated by five measures: the *detection hit rate, miss rate, false- alarm rate, alarm confidence,* and *IDS efficiency* as given below. All 5 measures are based on a newly introduced alarm matrix ***A,*** which is generated by the IDS after monitoring the network for a fixed period.

Detection Hit Rate: For attack type-**i**, denote h_i as the probability that the attack is correctly detected by the IDS. For each attack type i = 1, 2, …, ***n,*** we define $h_i = \dfrac{a_{ii}}{\sum_{j=1}^{n+1} a_{ij}} = p_{ii}$

By this definition, the hit rates are revealed by the diagonal elements in the probability matrix. In order to see the average behavior of an entire attack set, we define the *average attack hit rate* ***H*** across all attack types:

$$H = x / F = \sum_{i=1}^{n} f_i h_i / F \qquad (2)$$

Detection Miss Rate: Denote m_i as the probability that the IDS fails to detect an attack and regard it as no attack. For i = 1, 2, … , ***n***, we can compute $m_i = \dfrac{a_{i,n+1}}{\sum_{j=1}^{n+1} a_{ij}} = p_{i,\,n+1}$

We define the *average miss rate* by the following ratio for the entire attack set

$$M = y / F = \sum_{i=1}^{n} f_i m_i / F \qquad (3)$$

Alarm Confidence: This is a quality indicator of the IDS system applied. For an alarm of type j, we denote q_j as the percentage of the alarm being correctly raised by the IDS. For each alarm type j = 1, 2, **…, n,** we have $q_j = \dfrac{a_{jj}}{\sum_{i=1}^{n+1} a_{ij}}$ **.** We define the *alarm confidence* of the entire attack set by the following value, averaged over ***G*** alarms.

$$Q = x / G = \sum_{j=1}^{n} a_{jj} q_j / G \qquad (4)$$

False-Alarm Rate: Denote s_j as the probability of a false alarm on attack type-j from another attack type. For j = 1, 2, …, ***n***, we compute $s_j = 1 - q_j$**.** We define the *average false alarm rate* by:

$$S = (z + u)/G \qquad (5)$$

Note that ***Q*** = 1 **-** ***S.*** This term ***Q*** is conceptually different from the *detection hit rate* ***H.*** Since both ***H*** and ***Q*** deal with same hit subset ***x***, we realized the relationship ***x = HF = QG*** between them. The alarm confidence is

sensitive to attack frequencies of all types, while the hit rate is fixed with particular attack type. We use ***Q***, only if we know the distribution of real attack types

Figure 4 shows the detection rates, the miss rates, false-alarm rates, and false-positive rates for the running example. The x-axis shows 5 attack types and the y-axis shows the IDS performance rates. Three rates are plotted for each attack program. We plot the average performance at the right end. The 3 bars for average IDS performance reflect the average behavior of the entire attack set as listed in Table 3. Since our effort is to repel multiple attacks at the same time. The average behavior will be more useful in designing an IRS.

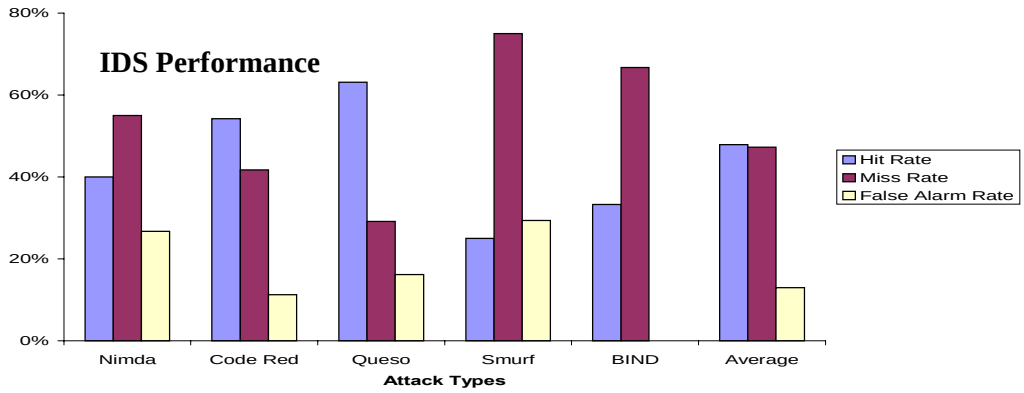

Figure 4 Intrusion detection rate, miss rate, and false alarm rate measured for 5 attack types as well the average intrusive behavior

Among the 5 attack programs, the detection rates vary from 25% to 65%, meaning some programs are more accurately detected than others. The Smurf and BIND are more difficult to detect, because of high miss rates experienced. The miss rates are between 30% and 80%, which implies that the IDS we used are far from being perfect. The false-alarm rates of all 5 attack programs are rather low, less than 5%. In fact, many of the confused alarms are zero in the example matrix. This has contributed to the low false-alarm rates. Figure 5 plots the alarm confidence of each attack type.

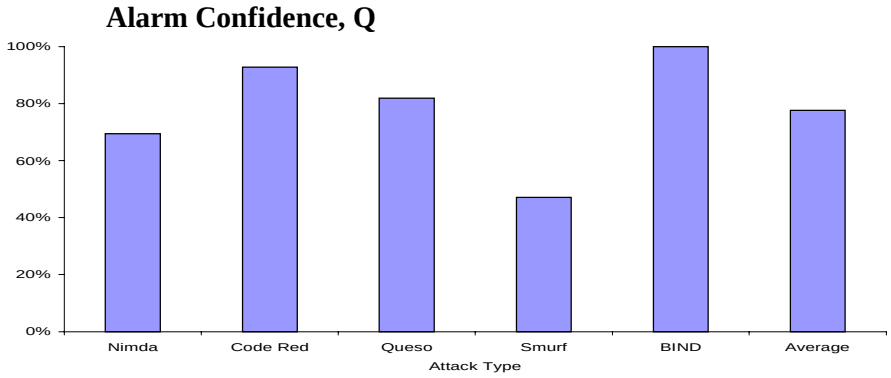

Figure 5 Alarm confidence for 5 attack types and the average behavior of the entire attack set in the running example

Based on the above results, we are now ready to derive an expression for the *IDS efficiency* in terms of the hit rate $\boldsymbol{H}$, alarm confidence $\boldsymbol{Q}$, and miss rate $\boldsymbol{M}$ defined in Eqs.(2), (3), and (5), respectively.

Theorem 1: (IDS Efficiency)

Considering all possible attacks and false alarms raised during an observation period, we derive the ***IDS efficiency,*** $\boldsymbol{E_{IDS}}$, as follows:

$$E_{IDS} = x/(u+x+y+z) = \left(\sum_{i=1}^{n+1} a_{ii}\right) \Big/ \left(\sum_{i=1}^{n+1}\sum_{j=1}^{n+1} a_{ij}\right) = QH/(H+QM) \qquad \textbf{(6)}$$

Proof: According to definitions given in Eqs. (2, 3, 5), we have

$$E_{IDS} = x/(u+x+y+z) = x/(G+y) = 1/(G/x + y/x) = 1/[\,1/Q + (y/F)/(x/F)\,]$$

$$= 1/(1/Q + M/H) = QH/(H+QM) \qquad \textbf{Q.E.D.}$$

In our running example, we have $\boldsymbol{E_{IDS}}$ = 80/182 = 44%. In Figure 6, we plot the IDS efficiency for three attack/alarm patterns, represented by three (***u, x, y, z)*** combinations. The top curve corresponds to the use of a high-hit IDS with ***M/H*** = 0.1, the middle one for an average IDS with ***M/H*** = 0.5, and the lower curve for using an IDS with a high-miss rate and high-alarm rate of ***M/H*** = 0.9. This example shows that the IDS efficiency varies between 20% and 90%, depending the detection rate and alarm confidence encountered.

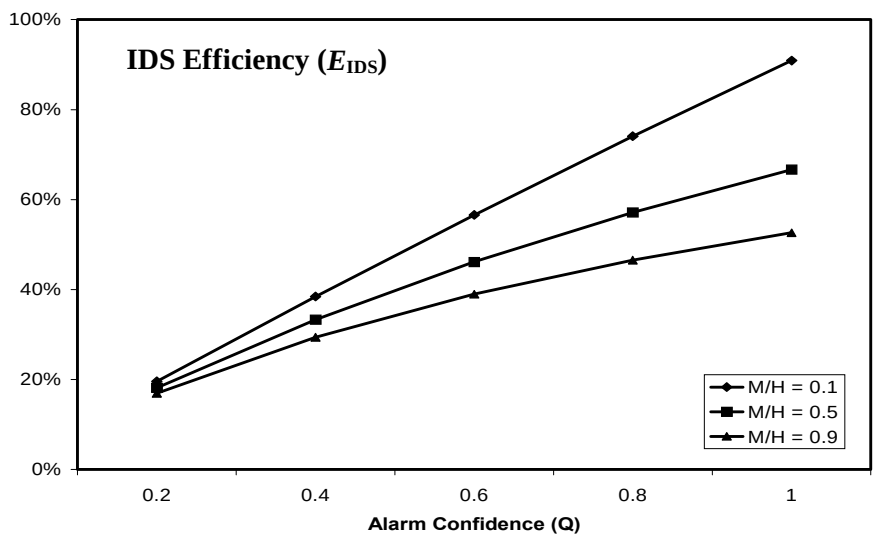

Figure 6 Plot of the IDS efficiency with respect to three distinct IDS reports under different attack/alarm distributions

If the IDS raises alarms very often on every suspicious packets, the false alarm rate will increase. The detection rate will also increase. On the other hand, If the IDS raises alarms only after sufficient evidence is available, the detection rate will suffer. In this case, the higher is the alarm confidence, the lower is the IDS to raise false alarms. In general, the quality of the IRS is built on top of the quality of the IDS. The two subsystems must be used jointly hand-in-hand to achieve high IDR performance

4. Risk Assessment and Response Cost Analysis

In this section, we assess the risk level from multiple attacks according to the potential damages from both real attacks and false alarms. Then we analyze the corresponding response cost factors. These cost models are useful in selecting the effective response mechanisms to reduce the risks to the minimal level, limited primarily by the efficiency of the IDS and IRS used.

4.1 Potential Damages and False Alarms

To consider the damages from each attack type, one must consider not only technical impacts to the system operations, but also the added costs in loss of operational services and replacement or repairing of hardware and software resources, among many other factors. For simplicity in illustrating the main ideas of risk assessment and strategic response actions to be taken, we will lump all the damage factors into a single dollar amount.

The damage dollar amount varies with attack time, attack types, attack frequency, system platforms, response time, and network conditions. The potential damages from attacks to a network are estimated by a damage vector of n+1 components, $(\boldsymbol{d_1}, \boldsymbol{d_2}, \ldots, \boldsymbol{d_n}, \mathbf{0})$, where $\boldsymbol{d_i}$ is the potential unit damage cost by a single attack of type i, exclusively**.** The last zero implies no damage from the case of no intrusions. We will assess the risk level using an *average damage* $\boldsymbol{D} = \left(\sum_{i=1}^{n} \boldsymbol{d_i}\right) / \boldsymbol{n}$ over all attack types.

The damage is estimated by a monetary loss from the particular attack type, if the attack is successfully carried out. The actual dollar amount may vary from system to system, because it may include all possible loss of resources, operation time, and recovery costs, etc. For simplicity, we lump all damage costs into a single dollar amount for each attack type. We will consider a damage vector ($4000, $7500, $5000, $3000, $1500, 0) in the running example. The average damage $\boldsymbol{D}$ = $21,000 / 5 = $4,200.

Three attack damage classes are specified below. We show below how to estimate the damage costs associated with the 5 attack types in the running example.

Maximum Risk: This amounts the total damage by multiplying the unit attack damages by the corresponding attack frequencies. The *maximum risk* $\boldsymbol{R_{max}} = \sum_{i=1}^{n}\sum_{j=1}^{n+1} \boldsymbol{d_i a_{ij}} \cong \boldsymbol{D \times F}$**.** In our running example, we calculated the $\boldsymbol{R_{max}}$ = $770,000. In the following, we subdivide the total risk into several components, based on attack and alarm categories encountered.

Attack Hit Damage: This corresponds to the true damage from hit attacks correctly detected by the IDS. The hit damage, $\boldsymbol{D_{hit}}$, is defined by the following summation of products: $\boldsymbol{D_{hit}} = \sum_{i=1}^{n} \boldsymbol{a_{ii} d_i} \cong \boldsymbol{xD}$. Note that the approximated value is based on risk assessment of the entire attack set. An IRS could suppress this damage partially or fully, depending on the response time and countermeasures applied (Cohen [1999]). The full damage could be experienced if no effective or no timely countermeasure is taken.

Confused-Alarm Damage: This amounts the damage from the confused alarms that are incorrectly detected by the IDS. We calculate this damage by: $D_{confused} = \sum_{j=1}^{n} \sum_{\substack{i=1 \\ (i \neq j)}}^{n} a_{ij} d_i \cong zD$. Note that the false-positive alarms in the last row of the alarm matrix does not cause any damage, since no real attacks were launched. Only false-positive miss alarms may cause the following miss damage.

Miss Damage: This amounts the damage from real attacks that are missed by the IDS as no attacks. The entries in the last column of the alarm matrix contribute to this damage. It is calculated by $D_{miss} = \sum_{i=1}^{n+1} a_{i,n+1} d_i \cong yD$. Since the IDS fails to detect these attacks, the IRS cannot repel these attacks at all. Most likely these damages will run through its entire course to become residue damage even after certain countermeasure is taken.

4.2 Response Costs and Residue Risks

The main purpose of risk assessment is to enable the cost-effective selection of the appropriate response mechanisms to cope with the anticipated attacks. First, we define several fundamental terms used to specify the response patterns. Then, we analyze the response costs item by item and also study the average behavior over the observation period.

Total Response Cost: Let R_a be the average response cost over all attack types based on selected defensive mechanisms in the deployment vector**.** The total response cost C_{total} covers all response costs caused by all attack type. This C_{total} is measured by $C_{total} \cong R_a G$ in dollar amount. Not that this total cost includes both effective and wasted components as shown in Figure 1.

Alarm Intervals: Let T_{total} be the total observation period, within which a total of G alarms were raised by the IDS. The average alarm interval, $I = T_{total} / G$ is averaged over all alarms raised. The relative magnitude of H and Q determines the relative magnitude of G and F. Our running example has the average alarm interval I = (30 days × 24 hrs) / 103 alarms = 6.01 hours between alarms for a total monitoring period of one month.

Time to Respond is the average time T_{ave}, that the IRS responds to any detected attack. The faster the IRS responds, the better chance to suppress the threat. There are two response categories: automated versus manual responses. Automated response does not require human intervention, while the manual response requires the security officer to make decisions. Assume the percentage of automated response as q and the percentage of manual response $1 - q$. Denote the average time to respond with an automated response as T_a and that of a manual response as T_m. Then we calculate T_{ave} as follows:

$$T_{ave} = q\, T_a + (1 - q) T_m \qquad (7)$$

Within the alarm interval, the T_{ave} corresponds to the overhead time to respond to an attack. During this time, the response is still in preparation and potential damage may occur. Beyond the overhead time, the defense will be established to repel the attacks.

Theorem 2: (Wasted Response)

The *wasted response* W is a fraction of the total response cost $R_a G$ ***that*** is not effective as a result of improper response by the IRS. The fractional waste is caused by the undesired mappings f_2 and f_4 illustrated in Fig.1. We compute W by:

$$W = R_a G \left[(G T_{ave} / T_{total}) Q + 1 - Q \right] \quad \textbf{(8)}$$

where R_a is the average response cost, G is the alarm frequency, T_{ave} is the average response time, Q is the alarm confidence, and T_{total} is the total time of the observation window.

Proof: The $R_a G$ is the total response cost. The ratio $G T_{ave} / T_{total}$ indicates the total overhead within the observation window. During these periods, the countermeasures are not effective yet. The first term is the waste caused by the alarm confidence Q in Eq.(5). In other words, the first waste term is caused by the undesired mapping $\mathbf{f_3}$ in Figure 1. The second waste term is caused by confused and false-positive alarms rated by $S = 1 - Q$ in Eq. (4). This corresponds to the undesired mapping $\mathbf{f_4}$ in Figure 1. **Q.E.D.**

We will plot the experimental results on the wasted response in Section 6.2. Next, we study the intrusive behavior and its consequence of residue risk, if the responsive countermeasures are not immediately deployed and some damage may be already done, if the action is too slow to become effective.

Lemma 1: (Residue Risk)

This risk is caused by the remaining damage after applying a set of intrusion countermeasures. This is a measure of the remaining risk level after the effect of the IRS. When the response time is longer than the alarm interval, i.e. $T_{ave} > I$, we know that $R_{residue} = R_{max.}$ When $T_{ave} < I$, we can estimate the $R_{residue}$ by considering the following three components terms:

$$R_{residue} = (T_{ave} / I) D_{hit} + D_{confused} + D_{miss} \cong (G T_{ave} / T_{total}) xD + zD + yD \quad \textbf{(9)}$$

Proof: By Eq. (7) and definition of alarm interval I, the ratio $T_{ave} / I = G T_{ave} / T_{total}$ represents the proportion of the alarm interval I, attributed to the overhead time in the intrusive response process. During this overhead period, the network system is defenseless and thus the attack hits may cause real damage, even the response will be in place in a later time. Thus the first residue-risk term is attributed to the attack-hit damage. The second term is the residue risk caused by the confused false alarm. The third residue risk is caused by detection misses. The proof is complete by substituting the three damages to end with the above approximated estimation of the residue risk. **Q.E.D.**

Theorem 3: (IRS Efficiency)

The *IRS efficiency* indicates the quality of the IRS used. The IRS takes appropriate countermeasures against the incoming attacks amid false alarms raised by the IDS. Based on Lemma 1 and the maximum risk $R_{max} \cong DF$. We obtained the following expression for $E_{IRS} = 1 - (R_{residue} / R_{max})$, provided $Q > E_{IDS}$.

$$E_{IRS} \cong 1 - [(T_{ave}/I)Dx + (Dz + Dy)]/DF$$

$$= H[1 - T_{ave}/I] = H[1 - G(T_{ave}/T_{total})] \quad \textbf{(10)}$$

Proof: If the alarm confidence is lower than the IDS efficiency, the IRS will not perform a satisfactory job. From Theorem 1 (Eq.6), we know that the provision $Q > E_{IDS}$ is always true. Because $I = T_{total}/G$, $y + z = F - x$, and from Eq. (2), we have

$$E_{IRS} \cong 1 - [(T_{ave}/I)Dx + (Dz + Dy)]/DF = 1 - (T_{ave}/T_{total})GH - (F - x)/F$$

$$= 1 - (T_{ave}/T_{total})GH - 1 + H = H[1 - T_{ave}/I] = H[1 - G(T_{ave}/T_{total})] \quad \textbf{Q.E.D.}$$

In a way, the IRS efficiency is equal to the detection hit rate minus the effect of response overhead. This efficiency measure is thus heavily related to the detection hit rate H, the alarm frequency G, and the overhead proportion $G(T_{ave}/T_{total})$. The overhead term represents the proportion of penalty from delayed response over G alarms raised. The above approximation will be used to plot the IRS efficiency in Section 6. Different response strategies may lead to different levels of residue risk and thus end up with different IRS efficiency.

For example, when the Smurf attack is detected, the damage has already taken place. Thus the residue risk remains very high, regardless of the countermeasure applied. On the other hand, the risks from the Nimda and CodeRed in our experiments can be effectively suppressed with minimal residue risk, if the IRS respond within a sufficiently small time. These concerns will be answered by the numerical results in Section 6.

5. Countermeasures and Response Strategies

The adaptive intrusion response strategies are introduced in this section. Fist, we identify the available response mechanisms for handling different attacking types. Then we outline the strategy selection procedure based on network conditions and risk assessment results from previous sections.

5.1 Countermeasures Against Intrusive Attacks

This section demonstrates the response options that can effectively counter the incidents. Table 4 summarizes the options under consideration. Each response mechanism is designed to counter the attacks differently. Some responses are more effective if it responds immediately. If the attacker tries to send the worm to a target machine, immediate dropping of suspicious packets will prevent the system from being infected.

In Table 4, we list a number of known intrusion response mechanisms, commonly available or desired for building an IDS, an IRS, or a combined IDR system. Some of these mechanisms have been selected to implement the proposed RADAR system. The mechanisms are identified by functional names in the second column and by code names at the left. Other response mechanisms, such as tracing back to the origin of an attack (Wang et al. [2001]), are not included in Table 4, since these mechanisms have not been commercially implemented yet.

If we need to verify the effectiveness of a response mechanism, we must test the mechanism on a target system, such as checking the file integrity, etc. This testing approach is rather expensive to implement, because it

consumes significant amount of resources including the CPU cycles and memory usage. To eradicate the infected files, we need to know the result of previous actions (virus checking, modification check, etc.). Then we can proceed with deleting some files or removing the virus. To fully utilize the available mechanisms, we characterize intrusion response mechanisms into three categories:

- **Immediate response (IR):** This means every time the IDS detects an attack; it will respond with the selected mechanisms immediately.
- **Delayed response (DR)**: This means every time the IDS detects an attack; it will respond with the selected mechanisms in pre-specified time window later after the event.
- **Conditioned response (CR):** This means if IDS detects an attack, the response will be launched only if certain conditions must be met first. A **CR** can be either an **IR** or a **DR.**

Only a brief description of the functionality of each mechanism is given. The requirements refer to levels of implementation, such as at the kernel level, or user space, or at application levels. The response category is labeled as IR, DR, or CR as defined above. Representative software packages are identified for each mechanism at the right column. For the 5 testing attack types, the mechanisms applied are summarized below.

Nimda and CodeRed: Because Nimda and CodeRed are the combination of worm, virus and Trojan horse implementing backdoor software; they are able to cause serious damages. For examples, the following countermeasures are often deployed to repel these two attack types.

1. Using removal tool or manually remove.
2. Disable or block access to Web service until a patch available
3. Applying patch or isolate the infected computer quickly.
4. Train employees not to open unexpected attachment.
5. Reinstall the OS or restore files from the backup.
6. Change all the passwords, examine log files and footprints left by intruder.

If one respond to the intrusive activities earlier enough, one can kill the suspected process and block the intruders by updating the firewall rules. The countermeasures for remaining attack types are briefed below.

Queso: This program attacks OS weakness by observing the response pattern of TCP packet. To counter this kind of probe attack is not an easy task. You need to modify the OS system source code or change the OS parameters such as the stack finger print characteristics [hacking exposed]. This may solve the problem but it will violate the RFC 1644 (TCP extensions for transaction). The other option is to employ automated responses. The automated responses will be triggered by using signature base intrusion detection such as snort and when it find the matched pattern, the system can block the intruder's IP by using firewall, or snipping the session by using TCP Reset packet, or dropping the packets, or even alter the routing table.

Smurf: This is a DoS attack and it uses forged targeted IP address and sent to broadcast address. Thus this case we cannot just automated blocking the IP. First of all we need to disable the response to ICMP and disable directed broadcast IP. If you are the amplifier site and this attack is detected during its activity, we can easily drop the

packets before it will hurt the targeted machine. Being the victim host, one can also limit the ICMP traffic by notifying the Internet Service Provider.

BIND: This is another DoS attack, which can bring down the DNS service. The only available response mechanism to repel this attack is to upgrade some specific libraries. Details can be found in the web sit: http://securityresponse.symantec.com/avcenter/security/Content/ 2002.06.06.html

Table 4 Responsive Countermeasures to Repel Attacks with Minimal Risks

Code Name	Response Mechanisms	Brief Functional Description	Implementation Requirements	Software Packages
A	IP Blocking	Using IDS or firewall to block the suspect IP	Kernel (IR)	Snort, Port Sentry, CISCO IDS
B	Dropping Packets	Using IDS or firewall to drop the intrusion signature matched packets	Kernel (IR)	Hogwash
C	Reroute	Using firewall to modify routing table to inexistent IP (Black hole)	Kernel (IR)	Port Sentry
D	Killing process	Killing the suspected process or all running process	Kernel or User (CR)	Shell
E	Reboot	Rebooting the attacked machine	Kernel or User (CR)	Shell or Manual
F	Shutdown	Shutting the attacked machine	Kernel or User (CR)	Shell or Manual
G	TCP Reset	Sending reset packets to bring down the session either to sender or both destination and sender	Kernel (IR)	Port Sentry, CISCO IDS
H	Disable NIC	Isolating the victim machine by disable network interface card	Kernel or User (CR)	Shell or Manual
I	Delete files	Deleting the suspected files (modified or infected files).	Kernel or User (CR)	Shell
J	Run virus check	Using virus check software to confirm the status of system	Application or User (DR)	Norton Anti-Virus, McAfee
K	Run file integrity check	Using file integrity check software to confirm the status of system	Kernel or User (DR)	Tripwire
L	Turn off the services	If vulnerable software is running, it will be turned off	Kernel or User (CR)	Shell
M	Reinstall OS	After detected as infected or mostly modified, reinstall the OS.	User (CR)	Manual
N	Applying patch for OS	Updating the OS by applying patch from vendor	User (CR)	Manual
O	Reinstall Software	Upgrade the vulnerable software to the newest version.	User (CR)	Manual
P	Applying patch for Software	Instead of reinstall the whole software package we can also applying patch to current version.	User (CR)	Manual
Q	Change all passwords	Changing all passwords compromised systems	User (CR)	Manual
R	Format the hard disk	Most extreme to remove all the malicious codes in the system.	User (CR)	Manual
IP: Internet Protocol, IDS: Intrusion Detection System, IR: Immediate Response, DR: Delayed Response, CR: Conditioned Response, OS: Operating System, NIC: Network Interface Circuit.				

5.2 Adaptive Intrusion Response Strategies

Four intrusion response strategies are specified below. The selection of a particular strategy depends on the meeting of certain conditions as illustrated in Figure 7. The idea is to make the selection of response strategy based on the alarm frequency and maximum risk estimated. Three important parameters are used in the selection process, namely the IDS efficiency, the alarm frequency, and assessed risk level. Three thresholds: α, β, γ are chosen, respectively. We show the selection flow chart and then specify the 4 strategies below, separately.

When the IDS efficiency is lower than a minimum threshold α. Everything is manually operated. Special IDS refinement is thus needed to improve the accuracy. We choose α = 40% as the minimum of IDS efficiency that can be tolerated. This choice is close to the IDS efficiency experienced in our running example. When the alarm frequency is high such as β = 25 alarms per week, one should apply Strategies A or B. Strategies C and D should be applied only when the attack frequency is lower than a preset threshold β. For high-risk situation, strategy A or C should be applied. Otherwise, use strategies B or D. The maximum risk threshold we choose is γ = \$150,000, less than one fifth of the maximum risk R_{max} = \$770,000 in our running example.

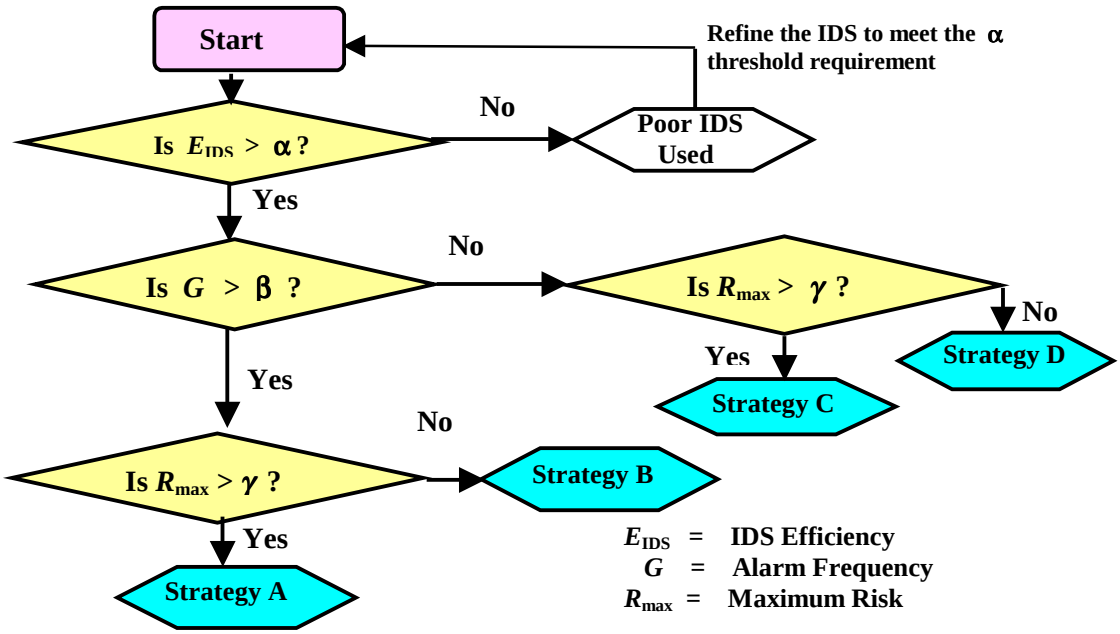

Figure 7 Selection among four intrusion response strategies where α, β, and γ are performance thresholds set by the IRS or by the security officer

Strategy A: In the worst scenario, we need to install as much automated response features as possible. Thus we need to respond to simultaneous attacks effectively. This strategy is the most expensive one to implement. In general, this strategy demands more than 80% automated responses and less than 20% manual responses.

Strategy B: If the total risk is low but with high attack frequency, then we should enforce more automated responses (IR/DR). This requires less sophisticated implementation and thus cost less than Strategy A. It is designed to achieve 60% automated responses and 40% manual responses.

Strategy C: The security plan for strategy C is less sophisticated than strategies A and B. Since the attack frequency is low, we use less automated responses. This strategy requires smaller cost to implement. It is designed to have 40% automated responses and 60% manual responses as shown in Table 5.

Strategy D: This is the least risky case where we still need to handle it properly. We focus more on effectiveness and the trace of the attack sources, since this scenario may trigger higher risk in the future. Strategy D require the security officer to conduct almost 100% of the operations, thus it requires the least cost to implement

5.3 Thresholds for Selecting Response Strategies

The conditions in selecting among these intrusive response strategies are summarized in Table 5. Strategy A uses most automated defense features. Strategy B applies comparable automated and manual features. Strategy C uses fewer automated then manual features. Strategy D uses all manual features. Among these, Strategy A is the one that can respond very fast but very costly. Strategies B and C are comparable in response time and costs. Strategy D is rather slow but costs significantly lower.

Table 5 Selection of Automated and Manual Features in Intrusion Response Strategies

Response Strategy	Strategy A	Strategy B	Strategy C	Strategy D
Selection Condition	High alarm frequency and high risk level	High alarm frequency and low risk level	Low alarm frequency and high risk level	Low alarm frequency and low risk level
Automated Features	A, B, C, D, E, F, G, H, I, J, K, L	A, B, C, G, J, K	A, B, C, G	None
Manual Features	None or very few	E, F, H, I, M, N, O, P, Q, R	E, F, H, I, J, K, M, N, O, P, Q, R	A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R

Illustrative Example (Continued):

Figure 7 outlines the decision process to select the appropriate strategy. We set the thresholds α = 40% for an average IDS, β = 700 alarms per 6 months based on the Ripteck report [2002], and a high-risk threshold γ = \$200,000 based on our running example. To respond to 5 attack types, we select from 18 possible mechanisms listed in Table 4. Assume T_a = 5 seconds and T_m = 2 hours. The following average response times are resulted from the application of the 4 response strategies, respectively.

T_{ave} = 0.8 × 5 + 0.2 ×7200 = 1,444 sec or 24 minutes for Strategy A.

T_{ave} = 0.6 × 5 + 0.4 ×7200 = 2,883 sec or 48 minutes for Strategy B.

T_{ave} = 0.4 × 5 + 0.6 ×7200 = 4,322 sec or 1 hour 12 minutes for Strategy C.

T_{ave} = 0.0 ×5 + 1.02 ×7200 = 7,200 sec or 2 hours for Strategy D.

With our running example, Strategy A is used. We obtain the following IDS efficiency, residue risk, and IRS efficiency from Eqs.(6), (9) and (10), respectively. The residue risk is rather high due to the high miss rates in our running example. For this reason, the low IRS efficiency is expected, since many of the attacks were either wrongly alarmed or missed by the IDS.

E_{IDS} = 80 / (80 + 79 + 8 + 15) = 0.44 = 44%

$R_{residue}$ = (0.057 × \$393,500) + \$40,500 + \$336,000 = \$399,079

E_{IRS} = 1 - (\$399,079 / \$770,000) = 48.17% ■

Effects of Selection Thresholds: In Figure 8, we plot on a 3-D bar diagram to show how the alarm frequency threshold β and the maximum risk threshold γ affect the choice of response strategies. Strategy A is applied to yield the highest efficiency with (β, γ) = (40 alarms, \$200,000). Strategy B gives the second highest efficiency with (β, γ) = (40 alarms, \$800,000). Strategy C is next with (β, γ) = (120 alarms, \$200,000). Finally, Strategy D gives the lowest efficiency with the threshold pair (β, γ) = (120 alarms, \$800,000). The alarm frequency and the β threshold are all measured in one-month period.

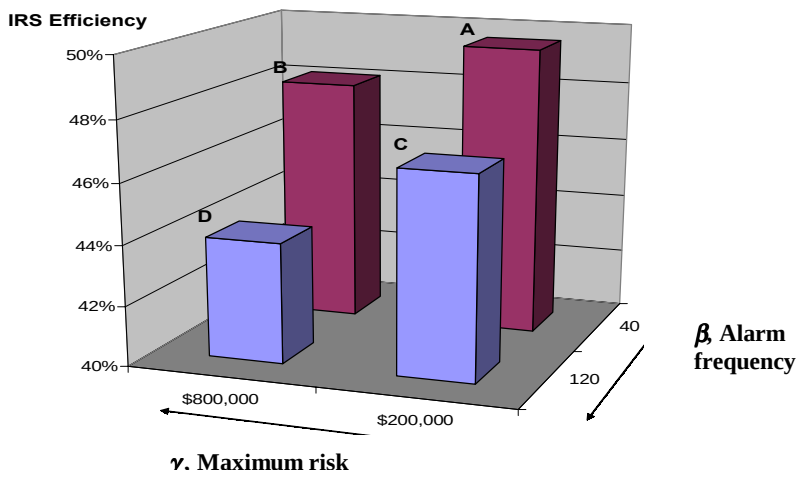

Figure 8 Effects of the thresholds (β , γ) to select the response strategy for the running example with G = 103 alarms/month and R_{max} = $770,000

Obviously, there exists a tradeoff between the two thresholds. The choice of depends on the alarm frequency G often encountered. The threshold depends on the maximum risk level R_{max} that we can be tolerated. The switching between strategies should be done dynamically with the variation of G and R_{max}. The low IRS efficiency is directly related to the low IDS efficiency or the low detection hit rate experienced.

6. Experimental Results and Performance Effects

Experimental results are presented below to prove the effectiveness of the RADAR framework for intrusion detection with semi-automatic responses. We focus on several performance factors: strategy selection, response time, alarm rate, residue risks, detection misses, and IRS efficiency.

6.1 Experimental Setting at USC

Figure 9 shows the experimental setting of distributed micro-firewalls installed in a PC cluster at USC. The security policy is distributed to all cluster nodes under the coordination of a policy manager. A gateway firewall is installed at the front-end, filtering packets between the cluster network and the external world of the Internet. The distributed IDS and firewall architecture was previously reported in Hwang and Gangadhran [2001].

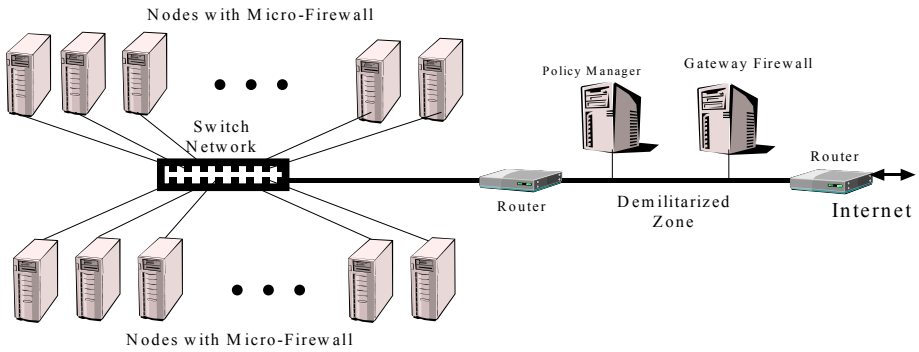

Figure 9 PC cluster built at USC for implementing distributed IDS and firewalls with adaptive intrusion response in the RADAR scheme

Individual cluster nodes act as detectors and enforcers of the central security policy. All network nodes work together to form a single security domain. A micro-firewall was built with IPtable on each node by Linux kernel extensions. We have also used existing IDS packages such as the Snort and LogSentry. This testbed was specifically designed with software to simulate the multiple-attack network environment.

We have evaluated SOAP, Agelets, RMI, CORBA, AC, FTP, HTTP, and Email for message exchange protocols. For the experiments reported here, the XML was used for both IDS reporting and security policy update. The distribution of security functions removes the bottleneck constraints of a single gateway firewall. This simulated security architecture was designed to repel both external intrusions and insider attacks.

6.2 Response Costs and Residue Risks

In this section, we study the risk and cost factors related to IRS efficiency. In Figure 10, the total response cost (C_{total}) is represented by the bar height. Both total cost and wasted response increase linearly with the alarm frequency (***G***). On the hand, the wasted response cost, ***W*** shown in the blue section at the lower bar, decreases with the alarm confidence (***Q***) experienced. The C_{total} for 200 alarms is about twice of that for 100 alarms raised.

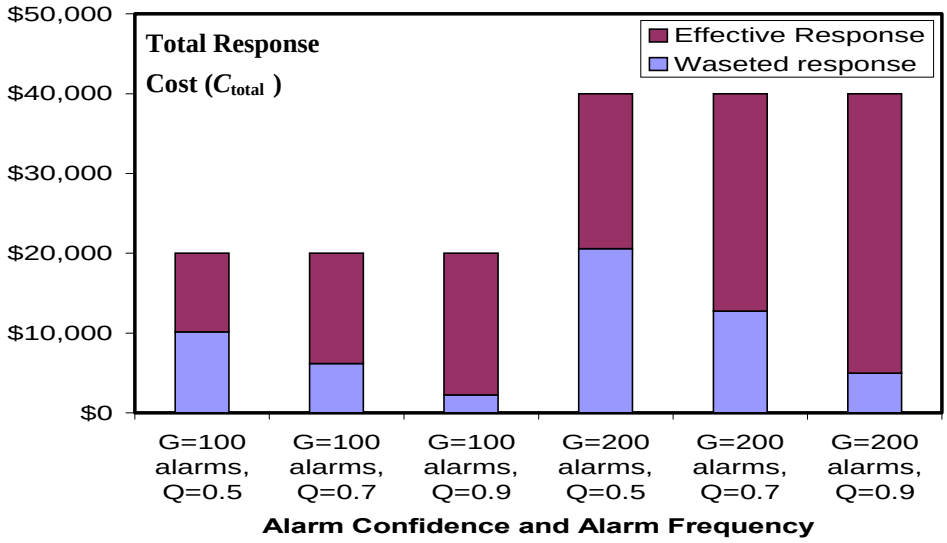

Figure 10 Breakdown of the total response cost into effective and wasted parts under various alarm confidence and frequencies

The effective part of the response is shown by the brown upper section. This part increases with both frequency and confidence of the alarms raised. The purpose of this cost breakdown is to reveal the level of wasted response. For the low alarm rate (***Q*** = 100), the waste is in the range 15-45% of the total cos. For a higher alarm rate (***G*** = 200), the waste rate did not increase much, even the total response cost is doubled.

Figure 11 plots the residue risk as a function the average response time. For our running example of 5 attack types, the residue risk increases linearly with the response time until reaching the 3-hour limit. In other words, the residue risk gets saturated at an upper limit determined by the maximum risk. The risk growth in time doubles from the lowest cost for immediate response to the maximum cost of $770,000 with a 3-hour response time. The lowest residue risk is attributed mainly to miss damage and confused alarm damage.

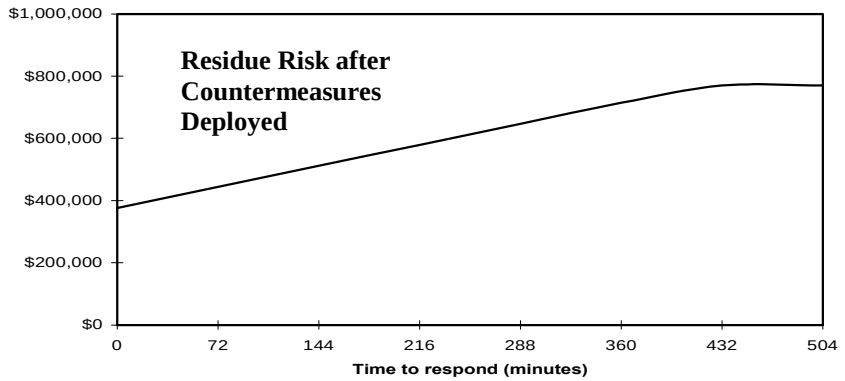

Figure 11 Variation of the residue risk with respect to response time

Figure 12 shows the breakdown of the residue risk by hit rate, miss rate, and false alarm under the strategy A. We consider four IDS reporting conditions: *high hit, high miss, high false alarms,* and *the mixed report* in our running example. These are demonstrated in four bars. As expected the case of high hit results in the least residue risk. Both high-miss and high-alarm rates lead to high risks. The running example correspond a mixed case of both high hit and high miss situation.

In the case of a high hit report, the residue risk is largely reduced to less than $200,000 on the leftmost bar, far less than the maximum risk of $770,000. The residue risk is most due to missed (brown section) and false-alarm (yellow section) damages in this case. In the second bar from the left, the high-miss dominates the most increase in residue risk. No hit damage (blue section) is noticed.

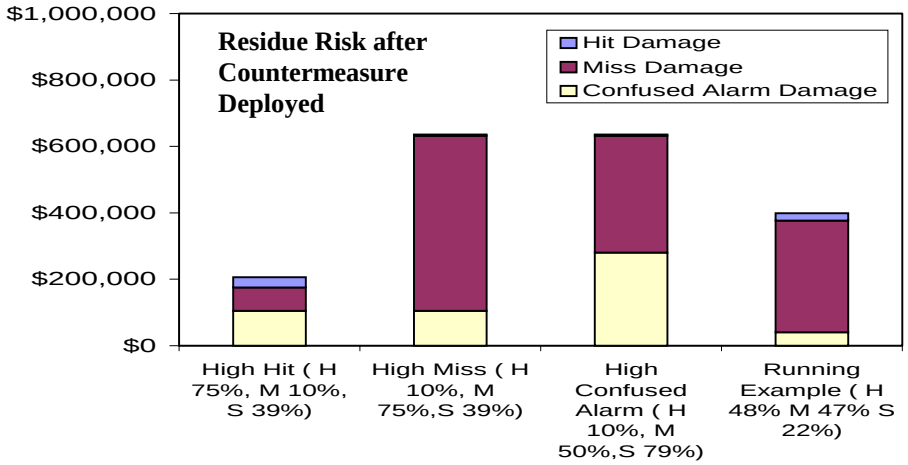

Figure 12 Residue risks under four risk assessment reports: high hit, high miss, confused alarm, and mixed rating in the running example

In the third bar, the effects of confused alarm increase to a larger proportion. The total risk remains very high (about 85% of the maximum risk) in the two middle cases. The rightmost bar corresponds to a mixed case of our running example. The miss penalty contributes the most to the total residue risk. The message being conveyed here is that increasing the detection hit rate and reducing the misses and false alarms are the two keys to cut down the total residue risk, after the responsive countermeasures are taking effect.

6.3 Intrusion Response System Efficiency

Effects of detection hit rate and alarm frequency on the IRS efficiency are revealed below. Figure 13 shows the relative performance of four intrusion response strategies for the running example. We plot the IRS efficiency with respect to three response time settings. Strategy A has the highest chance to stop the attack before it can cause any real damage. This is reflected by the highest IRS efficiency (48%) experienced. At the other extreme, strategy D falls to a low efficiency of 10% for a slow response time of about 6 hours.

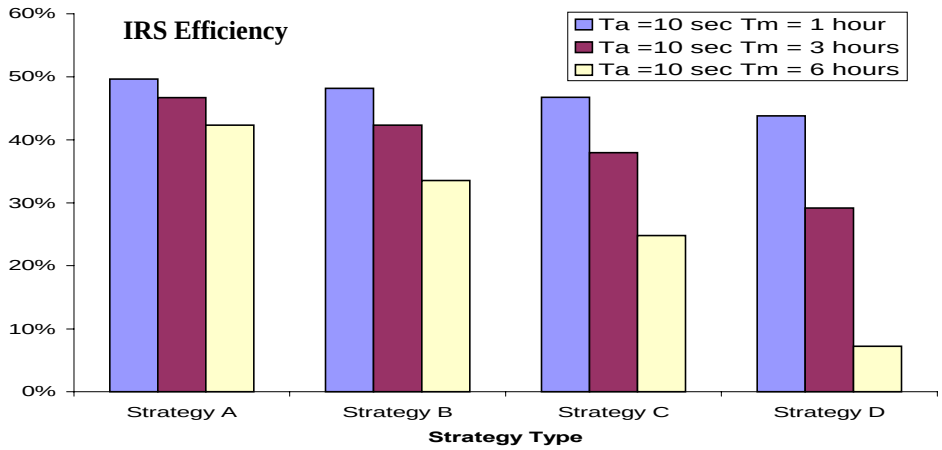

Figure 13 Variation of the IRS efficiency with response time and strategies applied for the running example with a mixed attack/alarm distribution

The performance of Strategies B and C lies in the middle. The performance gap between the 4 strategies gets widened as the response time increases. The difficulty and cost in implementing different strategies should be a factor in the selection process. The importance is to select the most cost-effective strategy. Inadequacy and over-kill are both unacceptable. In general, the low IRS efficiency is primarily caused by the slow response, confused alarm rate, and high miss rate. From the above results, we realize that all four strategies have comparable performance with a short response time of 1 hour (purple bars). For a response in 3 hours, only strategies A, B, and C are acceptable (brown bars). For long response time exceeding 3 hours, only strategy A and B are acceptable.

In Figure 14, we plot the effects of alarm interval $\boldsymbol{I}$ on the IRS efficiency. We have defined $\boldsymbol{I = T_{total}/G}$. With a fixed $\boldsymbol{T_{total}}$, the alarm interval varies inversely with the frequency. We scale the time from 24 minutes to 6.6 hours. Based on Eq.(10), the IRS efficiency of the 4 response strategies is plotted. The worst case comes when the average response time $\boldsymbol{T_{ave} = I}$. The efficiency reduces to zero in this case. Therefore, all 4 efficiency curves start at their respective $\boldsymbol{T_{ave}}$ value on the x-axis. The IRS efficiency converges to a saturated value equal to the detection hit rate H**,** as the alarm interval approaches infinity.

In the ranking order of **A, B, C,** and **D**, the 4 strategies increase in efficiency as the interval $\boldsymbol{I}$ increases. The growth rates also follow the same ordering. In other words, Strategy A has the fastest growth rate and the D the lowest. For example, the 4 strategies have efficiencies equal to 45%, 35%, 27%, and 16%, respectively, if the alarm interval gets prolonged to 192 minutes or more than 3 hours. The key message is to make the alarm interval sufficiently long to yield certain level of response efficiency for the strategy applied. For a fixed observation period, this means the IDS should raise fewer alarms, unless necessary.

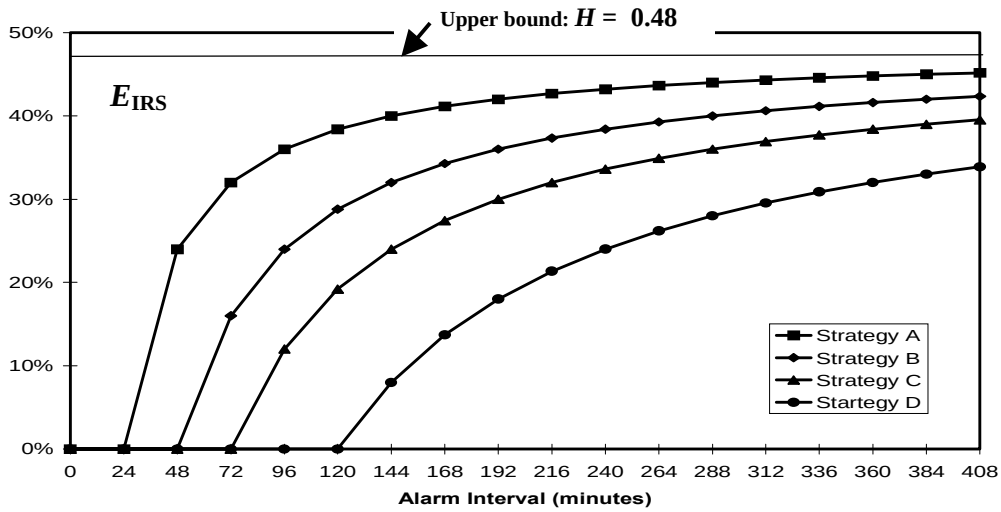

Figure 14 Effect of alarm interval on the IRS efficiency under 4 response strategies for the running example with a detection hit rate *H* = 0.48

In Figure 15, we show the effects of the detection hit rate ***H*** on the E_{IRS}. Three curves are shown corresponding to three attack frequencies from low to high within an observation period of one month. Based on the response strategy selection in Figure 7 and the parameters appeared in Eq.(10), we know that the IRS efficiency is affected by many factors including the IDS efficiency, the average response time T_{ave}, the total observation time T_{total}, the residue risk, the attack hit rate ***H,*** and the alarm frequency ***G.*** This plot reveals the negative effect of low hit rate on the IRS efficiency.

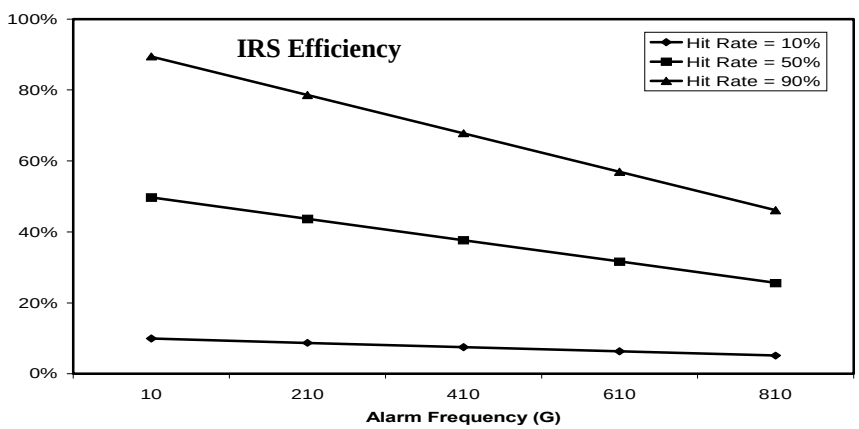

Figure 15 Effects of alarm frequency on the IRS efficiency under three representative attack detection rates

The IRS efficiency increases linearly with the alarm confidence. The IRS efficiency drops quickly with the increase of the residue risk. However, the IRS efficiency does improve with the response investment. This implies that the more security countermeasures it takes, the lower risk the protected network may face. With a short response time, the IRS efficiency thus depends heavily on the IDS efficiency.

7. Conclusions and Lessons Learned

To sum up, we highlight the research contributions from this work and present the lessons learned. We make a few suggestions for further research efforts. The primary contribution lies in the development of the RADAR model for risk assessment and for selection of intrusion response strategies as the threat pattern changes.

Summary of Contributions: Four major research contributions are identified below:

- Proposed a new mathematical model, the RADAR scheme, for risk assessment with armed intrusive response in LANs, clusters, and Intranets, using newly defined alarm matrix, damage and cost vectors, all illustrated with a simple running example.
- Introduced the new security concepts of alarm confidence, alarm rates, and residue risk. The IDS efficiency and IRS efficiency are revisited with new definitions, that can cover attack patterns amid alarms in terms of the 4-tuple (u, x, y, z), risk assessment, and minimized damages. These are useful tools for designing and evaluation of automated IDR systems in the future.
- Developed four dynamic intrusion response strategies, based on dynamically adjusting the thresholds on the IDS efficiency, alarm frequency, and total response cost. Strictly speaking, we have only achieved a semi-automatic response to multiple attacks within a short time period.
- A systematic procedure is provided for selection of intrusion response strategy, based on risk assessment and cost analysis. This procedure is adaptively implemented with respect to changing attack/alarm patterns and varying network conditions.

Lessons Learned: Several unique lessons are learned from the research findings reported here. These lessons are reported for the first time from our research group at USC. We also compare our findings with several reports from other research groups.

- Using the newly introduced alarm matrix, we revealed the fundamental differences between attacks and alarms. Our RADAR solution is meant to reveal instead of hiding attacks amid heavy alarms. Alarms and attacks are distinguished with fine lines of being true or false, distribution dependence, alarm frequency, and alarm confidence, etc.
- The IDS sees the attacks amid alarms. How to make the alarm accurate is the job of the IDS. The IRS sees only the alarms, not the attacks. In this sense, the new concepts of alarm confidence (Eq.5) wasted response Eq.(8), and residue risk (Eq.9) are useful to both IDS and IRS designers.. The IRS and IDS must be built hand-in-hand to provide a total network security solution.
- We learned that the accuracy, quality, and effectiveness of the IDS have fundamental impact towards the development of an effective intrusion response system (IRS). The IRS cannot reduce or minimize the risk alone, if a poor IDS is used in the first place. Other factors affecting the IRS efficiency include the attack detection rate, alarm frequency, and response overhead, etc.

- The experimental research findings verified the belief that collective risk assessment is the right approach to building cost-sensitive intrusion detection and response systems. Without the risk assessment, the IRS strategy cannot be optimized with minimized risks. These findings further reinforce the response cost models developed by Lee and Stolfo [2000].
- Our research findings suggest the tradeoffs exist between residue risks and the total response costs over multiple network attacks. Larger-scale security benchmark experiments are very much needed to improve the QoS of any distributed IDS and IRS in real network environments. This may demand more integration of the two subsystems over an extended monitory period.

Research Suggestions: We suggest the following directions for further R/D efforts. Essentially, we suggest more intensive collaborations between University and industrial research groups.

- For the perfection of the newly proposed RADAR system, the integration of the distributed IDS and IRS components must be continuously tested through extensive security benchmark experiments over larger network domains, attack sets, and observation window than small-scale experiments conducted at USC Labs. Currently, we are exploring industrial counterparts to join in this effort.
- The new concepts introduced for local-area network security could be extended for protecting wide-area Intranets, the Internet, or web services. The emphasis should be focused on automated intrusion detection and responses in production network environments. The ultimate goal should be set for full automation with very short intrusion response time.
- At the time of this reporting, the proposed intrusion response scheme was only tested through simulated experiments at USC Labs. To repel a large number of network attacks over an extended observation time window, future collaborative R/D should be beamed to prolong the observation window and to shorten the response time to any number of attacks. That challenge should involve both University and industrial research groups to work together.

References:

ALLEN, J., CHRISTIE, A., FITHEN, W., MCHUGH, J., PICKEL, J., and STONER, E. 2000. State of the Practice of Intrusion Detection Technologies. *Technical Report* CMU/SEI-99-TR-028, Carnegie Mellon Univ., Software Engineering Inst, Pittsburgh.

AXELSSON, S. 1999. The Base-rate Fallacy and its Implications for the Difficulty of Intrusion Detection. In *Proceedings of the 6th ACM conference on Computer and communications security,* 1-7.

BURROUGHS, D.J. WILSON, L.F. and CYBENKO, G.V. 2002 Analysis of Distributed Intrusion Detection Systems using Bayesian Methods Performance, *IEEE Int'l Computing, and Communications Conference,* 2002. 21st 329 -334.

CARVER, C. A. and POOCH, U. W. 2000. An Intrusion Response Taxonomy and its Role in Automatic Intrusion Response In *Proc. of IEEE Workshop on Information Assurance and Security,* US Military Academy, West Point, New York.

COHEN, F. 1999. Simulating Cyber Attacks, Defenses, and Consequences. *White Paper*. Fred Cohen & Associates. http://www.all.net/journal/ntb/simulate/simulate.html

CUPPENS, F. and MIEGE, A. 2002. Alert Correlation in a Cooperative Intrusion Detection Framework. In *Proceedings of 2002 IEEE Symposium on Security and Privacy* 187 - 200

FISCH, E. A., 1996. Intrusion Damage Control and Assessment: A Taxonomy and Implementation of Automated Responses to Intrusive Behavior. *Ph.D. Thesis,* Texas A&M University, College Station, TX.

FOSTER, C. KESSELMAN, G. TSUDIK and TUECKE, S. 1998. Security Architecture for Large-Scale Distributed Computations *ACM Conference on Computer and Communications Security.*

GAFFNEY, J.E., and Jr. ULVILA, J.W. 2002 Evaluation of intrusion detectors: a decision theory approach In *Proceedings of 2002 IEEE Symposium on Security and Privacy*, 50 – 61.

HAINES, J. W., LIPPMANN, R. P., FRIED, D., J., TRAN, E., BOSWELL, S. and ZISSMAN, M., A. 1999, DARPA Intrusion Detection Evaluation: Design and Procedures. *Technical Report*, MIT Lincoln Laboratory .

HWANG, K. and GANGADHARAN, M. 2001, Micro-Firewall for Dynamic Security with Distributed Intrusion Detection", *IEEE Int'l Symposium of Network Computing and Applications*, Cambridge, MA.

HWANG, K. 2002, Wireless PKI and Proactive Intrusion Response to Secure Intranets and M-Commerce, Keynote Address, *Third International Conference on Parallel and Distributed Computing, Applications, and Technologies*, (PDCAT 2002), Kanazawa, Japan, Sept. 4, 2002.

KO, C. and REDMOND, T. 2002, Noninterference and intrusion detection. In *Proceedings of 2002 IEEE Symposium on Security and Privacy*, 162 – 172

KRUEGUL, C., VALEUR, F., VIGNA, G. and KEMMERER, R. 2002, Stateful intrusion detection for high-speed networks. In *Proceedings of 2002 IEEE Symposium on Security and Privacy*, 162 – 172

KUMAR, S. and SPAFFORD, E., 1994, An Application of Pattern Matching in Intrusion Detection, , The COAST *Tech. Report,* Dept. of Computer Science Purdue University, June 17, 1994,

LEE, W. and STOLFO, S. J. 2000 A framework for constructing features and models for intrusion detection systems *ACM Trans.s on Information and System Security,* November 2000, Volume 3 Issue 4, 227 – 261.

MALKHI, D. and REITER, M. 1997. Unreliable Intrusion Detection in Distributed Computations. In *Proceeding of the 10th Computer Security Foundations Workshop* (CSFW97)

McHUGH, J. 2000. Testing Intrusion Detection Systems**:** a Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory. *ACM Transactions on Information and System Security,* November 2000 Volume 3 Issue 4, 262 – 294.

MEADOWS, C. and McLEAN, J. 1999. Computer Security and Dependability: Then and Now", in *Computer Security, Dependability, and Assurance: From Needs to Solutions*, Edited by Paul Ammann, Bruce Barnes, Sushil Jajodia, and Edgar Sibley, IEEE Computer Society Press,166-170.

MILLEN, J. K. 2000 Survivability Measure. *Technical report*, SRI International Computer Science Laboratory.

NEWMAN, D. 2002. Crying Wolf: False Alarms Hide Attacks. Network World, Inc. Web site: www.nwfusion.com/techinsider/20020624security1.html.

NING, P., JAJODIA, S. and WANG, X. S. 2001. Abstraction-based Intrusion Detection in Distributed Environments. *ACM Transaction on Information and System Security (TISSEC),* Volume 4, Issue 4, 407-452

NSS Group. 2001. Intrusion Detection Systems. *Group Test Report* (Edition 2), Cambridgeshire, England.

PETKAC, M. and BADGER, L. 2000. Security Agility in Response to Intrusion Detection, *16th Annual Computer Security Applications Conference*, New Orleans, Louisiana

RANUM, M. J., 2002. Experiences Benchmarking Intrusion Detection Systems. NFR Security Technical Publication. http://www.nfr.com/publications/

RIPTECH, 2002. Internet Security Threat Report, Volume I January 2002 http://www.riptech.com/securityresources/istreports.html

SAGE, P. and WHITE, E. B. 1980. Methodologies for Risk and Hazard Assessment: A Survey and Status Report. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-10(8), 425-445.

SANDHU, R. S. and SAMARATI, P. 1997 Authentication, Access Control, and Intrusion Detection, *IEEE Communications*, vol. 32, no. 9, 40-48 (under the title "Access Control: Principles and Practice")

SCHNACKENBERG, D., DIAHANDARI, K. and STERN, D. 2000. Infrastructure for Intrusion Detection and Response", *Proceedings. of the DARPA Information Survivability Conference and Exposition* (DISCEX), Hilton Head, S.C. Jan. 25-27.

SEKAR, R., GUANG, Y. VERMA, S. and SHANBHAG T. 1999. A High-performance Network Intrusion Detection System. In *Proc. of the 6th ACM conference on Computer and communications security*, 8 – 17.

SHEYNER, O., HAINES, J., JHA, S., LIPPMANN, R. and WING, J.M. 2002 Noninterference and intrusion detection. In *Proceedings of 2002 IEEE Symposium on Security and Privacy*, 162 – 172

SHIEH, S. and GLIGOR, V. D. 1997. On a Pattern-Oriented Model for Intrusion Detection. *IEEE Transaction on Knowledge and Data Engineering*, Volume 9, Number 4, 661-667

VARADHARAJAN, V. and KATSAVOS, P. 1996. Security Issues in Emerging High Speed Networks. In *Proc. of the 19th National Information System Security Conference NISS96*.

WANG, X., REEVES, D. and WU, F. 2001. Tracing Based Active Intrusion Response. *Journal of Information Warfare*, 1(1), Teamlink, Australia, 50-61.

WEST-BROWN, M. J., STIKVOORT, D. and KOSSAKOWSKI, K. 2002. *Handbook for Computer Security Incident Response Teams,* CMU/SEI-98-HB-001, CMU Software Engineering Institute.

Biographical Sketches

Sapon Tanachaiwiwat is presently pursuing Ph.D. degree in the Electrical Engineering Department at the University of Southern California. He has received his B.S. degree in Electrical Engineering from the Mahidol University in Thailand and M.S. degree in EE from USC. His current research interest includes Internet security and distributed Intrusion detection and response. He can be reached at **tanachai@usc.edu.**

Kai Hwang is a Professor of Electrical Engineering and Computer Science and the Director of the Internet Security and Pervasive Computing Lab at the University of Southern California. An IEEE Fellow, he specializes in computer architecture, parallel processing, Internet security, and pervasive computing. Dr. Hwang has published numerous papers and books in these areas. Presently, he leads a research group at USC developing new Internet security architectures, distributed intrusion detection and response systems, and distributed software RAID for scalable cluster computing. He can be reached at **kaihwang@usc.edu**

Yue Chen is presently pursuing his Ph.D. degree at the Computer Science Department at the University of Southern California. He has received his M.E. and B.E. degree in Electrical Engineering from University of Science and Technology of China. His current research interest includes Internet security, trust model evaluation, and software issues in trust engineering. He can be reached at **yuec@usc.edu.**