SCIT and IDS Architectures for Reduced Data Ex-filtration

Ajay Nagarajan[1] and Arun Sood[1, 2]
[1]*International Cyber Center and Department of Computer Science, George Mason University, Fairfax, VA*
[2]*SCIT Labs, Clifton, VA*

Abstract

Today's approach to security is based on perimeter defense and relies heavily on firewalls, Intrusion detection systems (IDS) and Intrusion prevention systems. Despite years of research and investment in developing such reactive security methodologies, our critical systems remain vulnerable to cyber attacks. In our approach we assume that intrusions are inevitable and our effort is focused on minimizing losses. Towards this end we have introduced a recovery based limited exposure time system called Self Cleansing Intrusion Tolerance (SCIT). In this paper, we investigate architectures that combine SCIT architecture with existing IDS approaches. The effectiveness of SCIT and IDS security architectures in terms of minimizing data ex filtration losses is analyzed using decision trees and the results of Monte Carlo simulation is presented.

1. Introduction

The variety and complexity of cyber attacks is increasing. Verizon 2009 Data Breaches Investigation Report [1] shows that customized malware is difficult to detect and data ex-filtration often occurs over a period of days, weeks and months. The attackers' strong motivation leads to organized and targeted cyber attacks. The current intrusion detection and prevention approaches are reactive in nature and inadequate to prevent all attacks. We conclude that intrusions are inevitable, and have adopted an intrusion tolerance approach. In [4, 8] we have introduced Self Cleansing Intrusion Tolerance (SCIT) approach. SCIT is a recovery driven intrusion tolerance system that makes the attacker work harder by reducing the server's exposure time to the internet.

More recently, a combination of reactive and proactive systems has been proposed [6]. We see such hybrid approaches, with multiple layers of defense as a desirable approach to protecting the cyber infrastructure. In this paper, we explore the usefulness of adding IDS systems to an intrusion tolerance approach. Specifically, in this paper we study a combination of IDS and SCIT architectures. We compare 4 architectures: (1) Network IDS only; (2) SCIT only; (3) Network IDS + Host IDS; (4) Network IDS + SCIT. From the view point of reducing data ex-filtration we discover that Network IDS + SCIT is the preferred solution.

The rest of the paper is divided into 6 sections. In the next section we discuss recent reports to motivate this study. Section 3 provides an introduction to SCIT and how it reduces losses. Section 4 presents the methodology utilized in this paper to gauge the effectiveness of a security strategy. Section 5 gives an overview of various security architectures compared in this paper along with decision trees representing their functionality. Section 6 gives an account of the Monte-Carlo simulation, the parameters used and the results obtained.

2. Motivating Examples

In reports of recent breaches, it has become clear that intruders were in the system for long periods. Not only did the IDS/IPS fail to prevent the intrusion, these systems were not able to detect the presence of the intruder. To illustrate this point, we refer to the following data breach reports:

- Verizon DBIR [1] focuses on 90 studies conducted in 2008. 285 million consumer records were compromised. Some of the parameters we use in this paper are derived from this report. The average Intruder Residence Time (time between system compromise and breach containment) was more than 28 days and on average 675 records were compromised per day.
- Network Solutions breach [12] of June - July 2009 resulted in 600,000 records compromised and the breach was detected after 2 months.
- Wyndham Hotels breach [11] was detected in January 2010, with an estimated start date of October 2009.

From these incidents, we conclude that any strategy that will shorten the duration of the breach would lead to better protection of data files. Consequently, in our analysis we focus on the estimated records ex-filtrated because of malicious activity.

3. SCIT Framework

In [4] we presented SCIT, an intrusion tolerant technique that provides enhanced server security. SCIT research has focused on critical servers that are most prone to malicious attacks. The technique involves multiple virtual instances of a server. These are rotated and self-cleansed periodically irrespective of the presence or absence of intrusions. Self-cleansing refers to loading a clean image of the server's OS and application into the Virtual Machine. Rotation here refers to the process of bringing an exposed virtual server off-line, killing it, restarting it and in the meanwhile, bringing another virtual server online to assure availability. By doing so, in the event of an intrusion, the intruder is denied prolonged residence on the server. Once the virtual server's exposure time to the Internet is completed, the virtual server instance is automatically rotated by a controller. This virtual instance of the server is what is referred to as virtual server throughout this paper.

Every virtual server is rotated through 4 states as shown in Figure 1. Exposed state is the state in which the virtual server is on-line. If the exposed virtual server is busy processing an earlier query, the new incoming requests are put in a queue. The queries that are in the queue of a virtual server and are not processed during its exposed state are processed in its quiescent state. In quiescent state, no incoming queries are accepted. The virtual server is killed and restarted in the Stop / Start state. A virtual server in live-spare state suggests that it's ready to go on-line.

We use VMware in our implementation, though the SCIT approach is not reliant on this virtualization approach. The SCIT Controller ensures the constant rotation of the virtual servers.

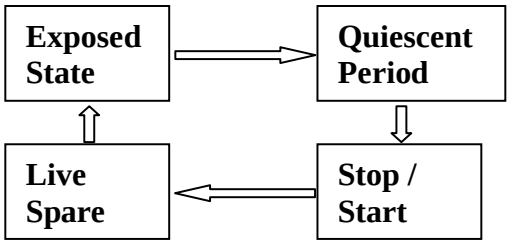


Figure1: SCIT State Diagram

4. Methodology to calculate data-ex filtration costs

4.1. Overview

We consider four SCIT / IDS architectures. Two alternatives are standalone – NIDS only and SCIT only. In PCI DSS[9] and in DODi 8500.2[10], host IDS are suggested in addition to Network IDS, thus we consider NIDS + HIDS systems. Finally, we treat NIDS and SCIT. To evaluate the potential losses from each of these systems we follow the approach of [7]. We develop decision trees that represent the functionality of respective security architectures. The conditional probabilities in the decision trees help characterize their security properties. These decision trees are translated into decision guidance systems (DGS) by modeling them on Gnumeric - an open-source spreadsheet software suitable for Monte Carlo simulation. We have 4 DGS' - one each for NIDS, SCIT, NIDS + HIDS, NIDS + SCIT architectures.

The DGS built on top of the decision tree using Gnumeric takes incoming traffic (in terms of queries) as input and divides the traffic into 4 categories: Confirmed Intrusion (CI), Non-intrusions (NI), False Alarms (FA) and Missed Intrusions (MI). Gnumeric's inbuilt Monte-Carlo simulation capabilities are used to generate incoming network traffic. In the case of Intrusions and Missed Intrusions, there would be an Intruder Residence time (IRT) associated with it. Section 6 expands on IRT and how it is modeled in the simulation. Using this IRT and the parameters from Verizon DBIR [1] from section 2, data ex filtration costs in terms of records compromised are calculated.

4.2. Assumptions

In our analysis we assume that

- In the malicious data ex-filtration process, records are stolen at a uniform rate.
- No records are stolen if the IDS correctly identifies an intrusion.

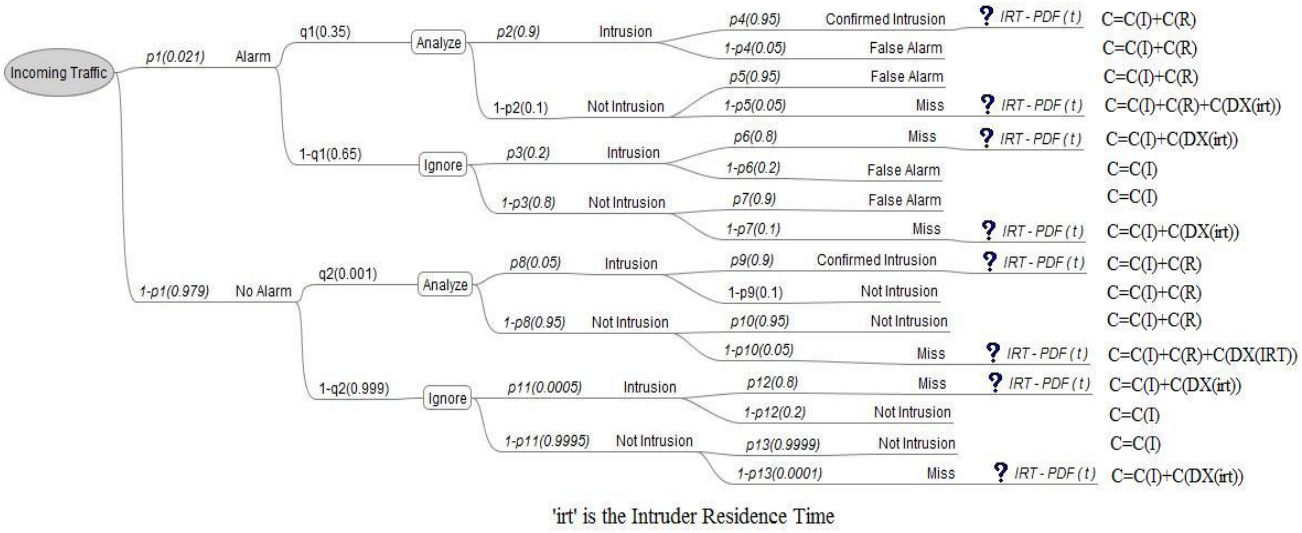
There is a constant cost associated with:

- Performing Intrusion Detection on a single query (incoming traffic) --- C(I)
- SCIT processing of a query (incoming traffic) --- C(T)
- Responding to one intrusion alarm --- C(R)

Since our objective is to characterize the effectiveness of the security architecture in terms of least data ex filtrated, we ignore the constant costs. However, there is provision in the decision guidance systems to include these costs if need be.

5. SCIT / IDS Scenarios

Each of the four SCIT / IDS architectures are considered and are explained briefly. Decision tree representations of each of the architectures are discussed. The decision trees provide a mechanism to estimate costs associated with each of the outcomes (Confirmed Intrusion (CI), Non-intrusions (NI), False Alarms (FA) and Missed Intrusions (MI)). This helps us get a better idea of data ex-filtration costs suffered in each of the IDS



In all the decision trees, (p1...pn) and (q1...qn) represent conditional probabilities.

Figure 2: NIDS Decision Tree

and / or SCIT scenarios. We emphasize that no loss occurs in the case of confirmed intrusion, since IDS detects those.

A number of probability values (p1…p34); (q1…q6) make up the following decision trees, however, it's interesting to note that not all of them contribute equally in determining the outcome. For example, sensitivity analysis performed on the NIDS decision tree suggests that each of the possible outcomes (CI, NI, FA and MI) are most sensitive to change in the value of p1. They are less sensitive to change in the values of q1 & q2. They are least sensitive to change in the values of p4…..p13.

5.1. NIDS

In this case, we consider a stand-alone independent Network Intrusion Detection System (NIDS) security architecture. The decision tree in Figure 2 represents NIDS functionality and its effectiveness in finding intrusions and minimizing data ex filtration. In Figure 2, values within braces next to the probability variables represent respective values considered to perform Monte-Carlo simulation. For instance, p1 (0.021) indicates that a value of 0.021 has been utilized for probability variable 'p1' in the simulation. Entire incoming traffic is monitored by the NIDS. Based on what it sees, there is a probability *p1* of NIDS triggering an alarm and a probability *1-p1* of NIDS determining the traffic to be safe. In case of an Alarm, a probability *q1* is associated with initiating a response and a probability *1-q1* associated with ignoring the Alarm. For instance, intrusions with severity (1, 2) are responded to and alarms with low severity ratings (3 to 6) are ignored. Such decisions are often made in security operations centers because of manpower limitations and the large number of alarms generated by the IDS.

In the case of responding to an alarm and analyzing it, there is a probability *p2* that the alarm ends up being categorized as an intrusion and a probability *1-p2* of it being safe. Again, no security procedure in place is ideal, there is an error rate associated with it. For example, traffic which is categorized as an intrusion, in reality could be an intrusion (confirmed intrusion) with a probability of *p4* or could be a false alarm (error on NIDS's part) with a probability of *1-p4*. A similar explanation follows anything that is categorized as a non-intrusion. On ignoring an Alarm, incoming traffic is let through without further analysis. This traffic in reality could be an intrusion (error on system administrator's part – ignoring the alarm) or a non-intrusion (error on NIDS'

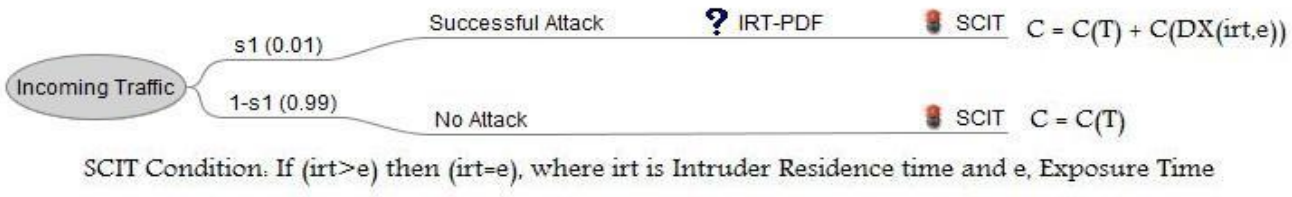
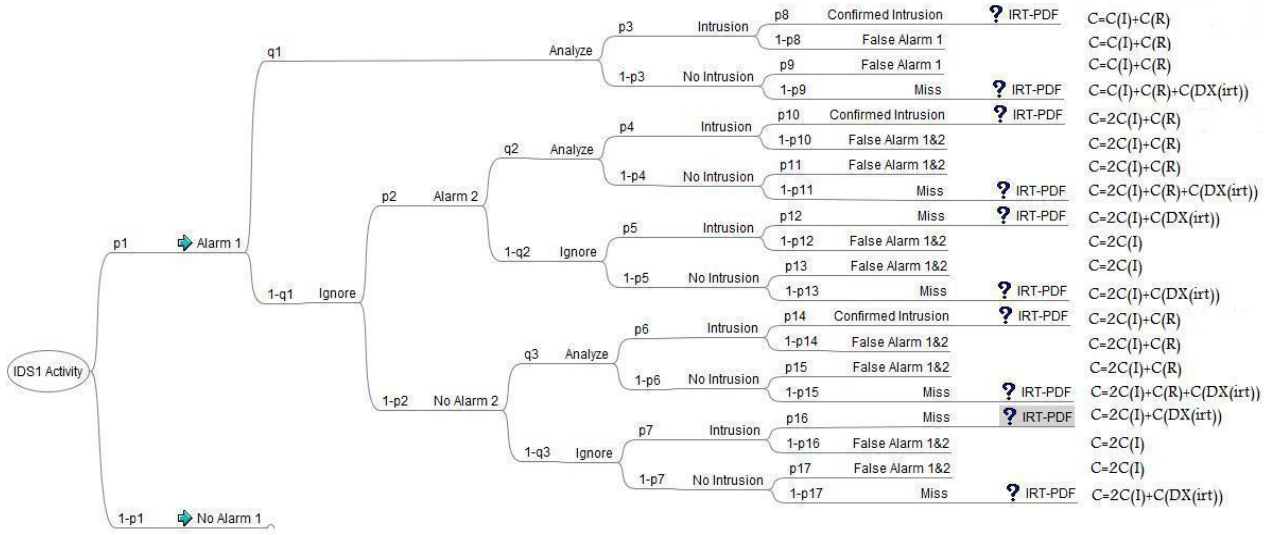


Figure 3: SCIT Decision Tree



'irt' is the Intruder Residence Time

Only one half of the IDS-IDS decision tree is represented above. The ' No Alarm 1 ' case follows a similar decision pattern to that of ' Alarm 1 ' case with respective probabilities (q4...q6) and (p18....p33). Here (q1...q6) represent probabilities associated with actions and (p1....p33) represent probabilities associated with uncertainties.

Figure 4: NIDS – HIDS Decision Tree

part). In this case, intrusions are characterized as Misses and non-intrusions as False Alarms.

In the case of a No-Alarm; the system administrator can still opt to analyze the traffic just to make sure the system is functioning the way it is supposed to. This could be on the basis of his / her suspicion or could be a random check to determine if all things are well. The procedure that follows is similar to the one discussed in the case of an Alarm.

In cases of Missed Intrusion traffic, damage is done to the system. In these cases, an intruder remains in the system for IRT duration of time causing damage, where IRT is the intruder residence time. In the simulation, we use the IRT-Probability Density Function (Section 6) to estimate IRT. In this scenario the amount of damage that could be caused to the system is unbounded, since IRT is unbounded.

5.2. SCIT

The security architecture in this case consists of a standalone SCIT system. There is no intrusion detector in the system. In other words, all potential attacks are successful since there are no IDS / IPS to check for them. Figure 3 represents SCIT's decision tree. The incoming traffic is classified as either being a successful attack or not. This is not done by the system since SCIT treats all incoming traffic in the same manner. There is a probability ‘s1’ associated with the incoming traffic being an attack and a probability ‘1-s1’ associated with it being safe traffic. In the case of an attack, the intruder remains in the system for IRT duration of time causing damage.

In the case of incoming traffic being safe, there is no IRT associated with it. Estimation of IRT is provided in section 6. In the case of an attack, estimated cost is C(T)+C(DX(irt,e)), where C(T) is the cost of SCIT implementation and C(DX(irt,e)) is the cost of data ex filtrated by the intruder in IRT duration of time. Since SCIT is in place, IRT can never be greater than SCIT's exposure time 'e'. And so the maximum possible damage that can be caused to the system by the intruder is now C(DX(e)) where ‘e’ is the Exposure Time. In the case of safe (no attack) traffic, estimated cost is C(T) and no data loss occurs.

5.3. NIDS + HIDS

This architecture is an extension of NIDS. An additional layer of security in the form of Host IDS (HIDS) is added to the system. NIDS+HIDS systems could either have two IDS's running in parallel or have one followed by the other. We will consider our NIDS+HIDS to be serial, with the NIDS tuned to the network needs, and HIDS tuned to the specific needs of the host. The first IDS (NIDS) performs its task exactly in the manner illustrated in the case of NIDS in section 5.1. If IDS 1 does not trigger an alarm or if IDS 1 alarm is ignored then IDS 2 (HIDS) is run to see if it triggers an alarm (Note, there is a small probability ‘q4’ of system administrator analyzing the traffic even though IDS 1 does not trigger an alarm. IDS 2 is not run in these cases). This adds another layer of security in the sense that IDS 2 could pick up an intrusion that IDS 1 had missed. According to [7], unless one of the IDS’ is worthless, it is

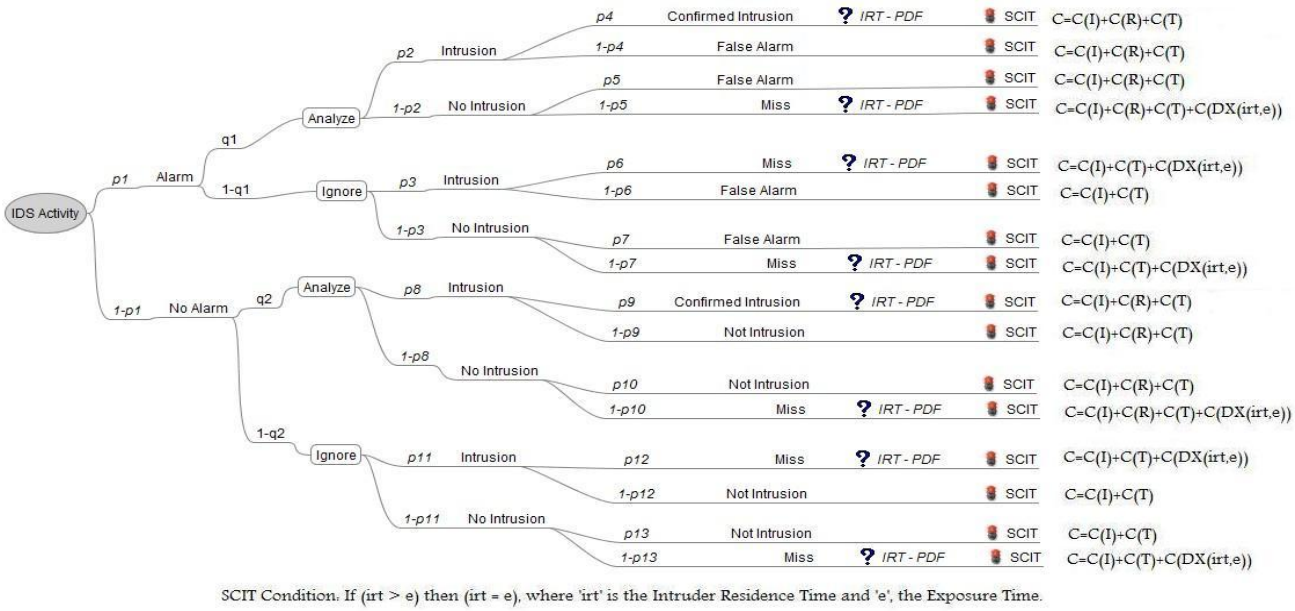


Figure 5: NIDS – SCIT Decision Tree

better to use both in combination than to use single IDS. They suggest that since there is no incremental cost to getting IDS2 report, the expected cost from using an IDS composed of two independent detectors is the same regard-less of whether the response decision is made sequentially or in parallel. In a serial IDS-IDS setup, it is advisable to have the better performing IDS as IDS 1.

5.4. NIDS+SCIT

The system here is an extension of a previous case, NIDS. An additional layer of security - SCIT - is added to the NIDS. In cases where an intruder resides on the system for IRT duration of time, SCIT comes into play. As pointed out, in the case of NIDS, potential damage that can be caused to the system is unbounded. This is primarily because IRT remains unbounded in NIDS. On adding SCIT, IRT is no longer unbounded. SCIT introduces a metric called 'Exposure Time'. Since SCIT is pro-active and performs self-cleansing after time 'e', where 'e' is the Exposure Time; an upper bound is set on IRT. With SCIT the maximum damage C (DX (irt)) that can be caused to the system is C (DX (e)) since (irt <= e). NIDS+SCIT performs better than standalone SCIT since NIDS helps identify certain intrusions before they can cause damage and have to be tolerated.

6. Monte Carlo Simulation

Methodology as presented in section 4 was followed to perform the Monte-Carlo Simulation. The decision trees represented above are captured in the simulation. The values used for the probabilities have been chosen on the basis of discussions with experienced managers. Certain assumptions were made in the process of simulating the decision trees based on these discussions: A) There are nearly twice as many False Alarms as Confirmed Intrusions and B) Out of the 50,000 incoming queries – 500 are potential attacks (as shown in Figure 3). Once the decision trees are incorporated in the Gnumeric spreadsheet format with all probability values plugged in, the inbuilt Monte-Carlo simulation feature in Gnumeric can be used to simulate the incoming traffic. Table 1 summarizes the parameters used in the simulation. Primary objective of the simulation was to compute a mean / total damage cost (in terms of records lost) in each of the SCIT / IDS cases given incoming traffic of 50,000 queries.

The Intruder residence time used in the simulation is modeled as a Pareto distribution. We assume IRT can take values between 0 hours and 2 months with mean being 48 hours. As compared to the examples in Section 2, this is a very conservative choice. Using the 28 days average, noted in Section 2, would be even more advantageous to SCIT. This average is incorporated in Intruder Residence Time Probability Density Function (IRT-PDF), which gives a relation between IRT values and their respective probabilities of occurrence.

6.1 Probability values chosen for the simulation

The values of (q1...q2) and (p1...p13) are the same for NIDS and NIDS+SCIT. These values are presented in Figure 2 within parenthesis next to respective variables. In the case of SCIT, probability values are presented in Figure 3. In case of NIDS + HIDS, the probability values

are given below – variables followed by their value:

q1 (0.35) | q2, q5 (0.1) | q3, p7 (0.01)
p8, p9 (0.95) | p18, q4, q6, p23 (0.001) | p33 (0.9999)
p1 (0.021) | p2,p6,p22,p19 (0.05) | p5,p21 (0.3)
p4,p12,p14,p20,p28,p30 (0.8) | p16,p32 (0.7)
p17,p3,p10,p11,p13,p15,p24,p25,p26,p27,p29,p31 (0.9) |

6.2. Results of the Simulation

Data loss measured in number of records is our metric for assessing effectiveness of security architecture. The results in Table 2 show data ex filtration costs in records. This table shows that the potential for damage is high for NIDS only and NIDS + HIDS alternatives. The records ex-filtrated are about the same for both scenarios. If SCIT is deployed then the ex-filtration losses are significantly reduced. The loss rate is dramatically impacted by the exposure time chosen. To illustrate this feature, we have reported the result for the case of 4 minute and 4 hour exposure times[1]. The best scenario is a combination of NIDS and SCIT. For NIDS+SCIT (ET 4 minutes) the records lost are less than 0.16% of the NIDS only loss and 0.19% of NIDS+HIDS loss.

Table1: Parameters used in the simulation

Simulation metrics	Value (units)
Number of queries used	50,000
Query Inter Arrival Time	10 ms to 18 ms
Intruder Residence Time (IRT)	0 minutes to 2 months
Mean IRT (modeled as Pareto distribution) against respective probabilities of occurrence.	48 (hrs)
Exposure time of SCIT (ET)	Case 1: 4 (hrs) Case 2: 4 (minutes)
Mean number of records stolen per day	675.4 records / breach
Mean number of records stolen per hour	28.15 records / breach

Table 2: Results of the Monte-Carlo simulation

Case	Total Damage (records)	No. of Breaches	Mean Damage (records/breach)
NIDS	245,962 (100%)	192	1,281
SCIT: ET 4h SCIT: ET 4m	55,364 (23%) 1,015 (0.4%)	508 508	109 2
NIDS+HIDS	210,578 (86%)	164	1,284
NIDS+SCIT: (ET 4h) NIDS+SCIT: (ET 4m)	20,931 (9%) 383 (0.16%)	191 191	110 2

[1] The prototypes that we have built have an Exposure Time (ET) of 1 minute, but in this analysis we take a higher ET to show the effectiveness of SCIT architecture.

7. Conclusion:

The SCIT architecture provides a robust security mechanism that guarantees certain security properties by limiting the exposure time. An important advantage of SCIT compared to IDS solutions is that SCIT does not generate false alarms, and thus can help reduce the intrusion alerts management costs. Thus SCIT also provides administrative and economic benefits which make it a reasonable choice to be included in security architecture. In particular, this is expected to be of interest in environments where technical skills are limited. Examples of such environments are found in military tactical settings, in remote and rural locations, small organizations and in newly emerging countries. The simulation studies presented suggest that a combination of an NIDS with SCIT on host servers provides a robust architectural solution in the face of new attacks.

Acknowledgement

This research was partially supported by NSF grant OISE - 0940922.

References

[1] Verizon 2009 Data Breach Investigations Report http://www.verizonbusiness.com/resources/security/reports/2009_databreach_rp.pdf
[2] Organically assured and survivable information systems. http://www.tolerantsystems.org
[3] Paulo E. Veríssimo et al. "Intrusion-Tolerant Middleware: The Road to Automatic Security". *IEEE Security & Privacy*, 2006.
[4] Yih Huang, David Arsenault, and Arun Sood. "Incorruptible System Self-Cleansing for Intrusion Tolerance". *Performance, Computing, and Communications Conference, IPCCC 2006*.
[5] Rabih Zbib et al. "Intrusion Tolerance in Distributed Middleware". *Information Systems Frontiers 6:1, 2004*
[6] Paulo Sousa et al. "Resilient Intrusion Tolerance through Proactive and Reactive Recovery". *13th IEEE International Symposium on Pacific Rim Dependable Computing*, 2007.
[7] Jacob W Ulvila, John E Gaffney Jr "Evaluation of Intrusion Detection Systems" , Journal of Research of the National Institute of Standards and Technology 2003
[8] Anantha K Bangalore, Arun K Sood "Securing web servers using Self-Cleansing Intrusion Tolerance", Second International Conference on dependability , 2009)
[9] PCI DSS Compliance Standards (Requirement 11.4) https://www.pcisecuritystandards.org/pdfs/pci_audit_procedures_v1-1.pdf
[10] DoDi 8500.2 Information Assurance Implementation http://www.dtic.mil/whs/directives/corres/pdf/850002p.pdf
[11] Wyndham Hotels Security Breach http://www.wyndhamworldwide.com/customer_care/data-claim.cfm
[12] Network Solutions Security Breach http://www.careandprotect.com/