Seldom cry wolf: Tuning out false positives on Network Intrusion Detection Systems

Paul Leitao

Seldom cry wolf: Tuning out false positives on Network Intrusion Detection Systems

GIAC Security Essentials
Certification (GSEC)
Practical Assignment
Version 1.4b

Option 2 - Case Study in
Information Security

Submitted by: Paul Leitao on 15 November 2004
Location: SANS Downunder 2004, Melbourne Australia

Table of Contents

1 Abstract

No group of words sound more exciting to a technical security practitioner than Network Intrusion Detection Systems. These words invoke mental images of cyber clashes with hackers, forensic investigations and do-or-die incident handling exercises. The truth, however, is far less glamorous. In my experience, Intrusion Detection System (NIDS) management usually consists of carrying out less high profile tasks such as system patching, signature updates and, of course, false positive identification and tuning. Just after my attendance at SANS Downunder 2004 in Melbourne, Australia, one of the major projects that I was deployed on was NIDS tuning for a financial services organization. Having carried out this function in the past and being well aware of the problems associated with this exercise, I resolved to design and implement a tuning methodology which would make the tuning process less painful. The following document describes the tuning methodology design and implementation steps. It provides a step-by-step process of deployment within a medium size organization (all IP ranges have been changed to protect the innocent of course).

The paper will focus on providing a methodology that may be used as a starting point to identify and minimize false positives. The network infrastructure which was used for this project will be described and three (3) false positive tuning sample exercises will be also provided. It is important to note that not all networks are created equal and what may work well in one situation may not work well in another. Therefore, it is important to be flexible and tailor the methodology as required. Additionally, the NIDS product used in this particular financial organization is Symantec ManHunt. However, the methodology described in this paper can be applied to most popular NIDS products. It is my hope that this paper will provide a structured and stable way for NIDS analysts to tune out false positives in their respective systems. By doing so, they will have more time to carry out the fun activities such as cyber clashing with hackers, performing forensic investigations and engage in do-or-die incident handling exercises.

2 Before

2.1 The Existing Situation

The last project I worked on prior to my attendance at the SANS Downunder 2004 conference was a Network Intrusion Detection Systems (NIDS) deployment at the financial services organization I currently work for. For the purposes of this document, we shall from here on in refer to this organization as All Mine Finance or AMF. The project went smoothly (or as smoothly as these things can go) and 4 NIDS sensors were deployed to 4 business critical DMZs. Additionally, data stores, management consoles and workstations were also deployed and administrator training was carried out. All in all, a well deployed, well administered infrastructure was put in place where nothing existed before (please refer below for NIDS infrastructure details).Paul goes to SANS. Paul enjoys SANS. Paul returns to work.

Upon my return to AMF, I was informed that the next step of the NIDS road map was to be implemented. This consisted of tuning out a large percentage of false positives reported by the NIDS sensors. I say a large percentage because I believe that you can never eradicate all false positives but you can minimise them.

So what is a false positive? My definition of a false positive is an event flagged by a NIDS as an attack which, upon closer inspection, is not an attack at all.

It was decided that this portion of the NIDS deployment was to be handled as a separate project led, managed and executed by yours truly. This stance was initially adopted during the NIDS project planning phase so as not to delay the initial NIDS deployment. The timeframe for this project was 3 months, give or take a month.

Fresh out of the SANS Downunder 2004 conference, I hoped to draw on some of the course material covered to bring this project to a successful and timely completion. Additionally, I started working on a false positive identification and tuning methodology which I hoped would help me take some of the guess work out of the tasks involved.

2.2 Risks

One of the first things I did was identify potential risks associated with the NIDS tuning project. These are detailed below.

- **Too much traffic noise for analysts to investigate**

A hot DMZ is both an exciting and a potentially overwhelming place to be in. Sometimes a NIDS picks up so many events that it's hard to filter out the real false positives. This could result in project completion delays due to an overwhelming number of events to investigate. Additionally, NIDS Analysts could miss events of real significance.

If an analyst sees the same false positive over and over (and have analysed and dismissed it a few times), they can become de-sensitized to that false positive. This means that when that false positive is an actual attack e.g. same attack, different source IP address, the analyst may not pay the deserved attention to the event.

- **Lack of resources to carry out tuning and analyst in-experience**

AMF does not have many trained NIDS analysts. This means that there could potentially not be enough trained personnel available to carry out the tuning.

- **Lack of identifiable false positives within project time frame**

Once in a while, the Gods of Info Sec smile upon you and give you a little present like no false positives. This allows you to focus on managing confirmable incidents. However, be weary. This is often a wolf in sheep's clothing. Just because you can't see it does not mean it's not there. No one is that lucky.

- **System down time**

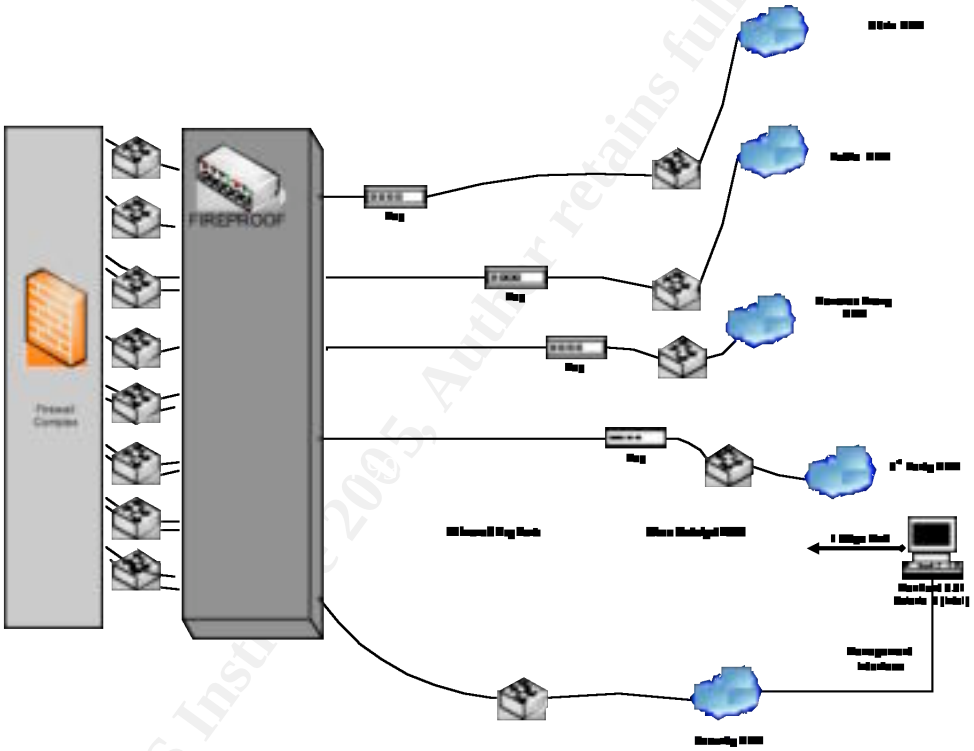
If the NIDS system is not up and running, you can't monitor. If you can't monitor, you can't identify false positives. Simple as that. Down time could occur due to a number of factors ranging from scheduled maintenance to Denial of Service attacks on the NIDS itself.

2.3 The Infrastructure

The following section details the ManHunt NIDS architecture as deployed at AMF. As shown in the figure below, all transmit and receive data is tapped from the network links connecting the Radware Fireproof systems to their respective DMZ switches. The implementation sees data from four (4) different DMZs being tapped and aggregated on a single Cisco Catalyst 2950 switch. Each of the tapped ports is mirrored to a single gigabit port on the 2950 switch that is monitored by a Manhunt sensor.

The management interface is used by security staff to manage the Manhunt node using the Manhunt Console software to be installed on nominated I.T. Security PC's.

For the purposes of this document, we will focus on tuning out false positives on the public DMZ traffic.



2.4 Traffic Flow

One of the most important things to determine when carrying out NIDS tuning is traffic flow. Traffic flow is the direction and paths used by network traffic moving to and from different parts of your network infrastructure including servers, workstations, networks segments, routers and switches. The type and flow of traffic within your organisations' infrastructure should be policy driven. By this I mean that your organisation's security policy should state what kind of traffic should flow to and from the different network segments.

For example, e-mail traffic coming from the Internet into your organization might flow from the Internet, into an e-mail relay server in your public DMZ segment and then sent into another e-mail server located in your internal network segment. The flow of this exchange would look something like this :

Internet --→ DMZ --→ Internal network

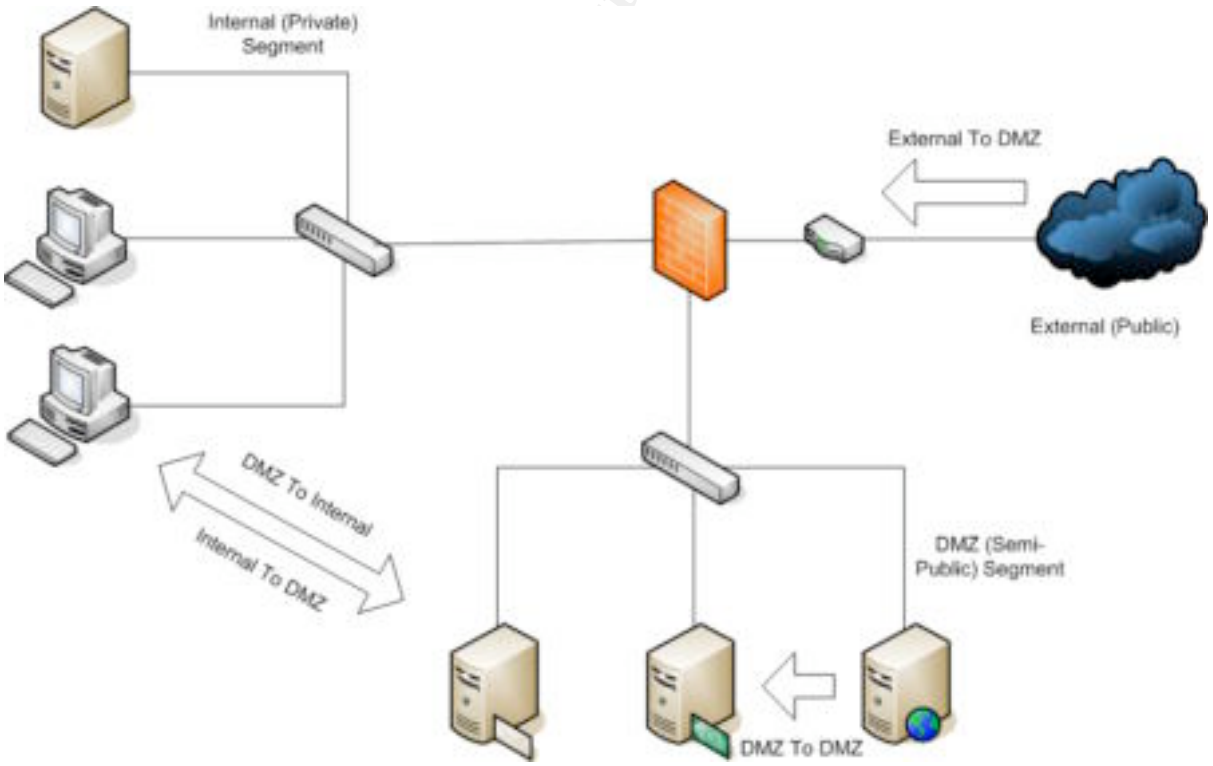
Outbound e-mail being sent from your organization and across the Internet to another entity might take the reverse path as such :

Internal network --→ DMZ --→ Internet

For the purposes of the NIDS tuning methodology detailed in the sections below, the following traffic flow paths for AMF were determined in order provide a starting point for the NIDS tuning project :

- **External to DMZ -** This is Internet traffic originating from the Internet and hitting our public DMZ such as e-mail or external clients surfing to our Web site.
- **DMZ to DMZ -** This is DMZ traffic amongst DMZ servers. Examples of this could be management traffic such as SNMP.
- **DMZ to Internal -** This would be DMZ to Internal traffic such as front end Web servers talking to back end database E-Commerce servers.
- **Internal to DMZ -** This would be internal to DMZ traffic such as internal clients surfing the Web via a proxy in the DMZ
- **External to Internal –** This should not occur as the AMF firewall does not allow it.
- **Internal to External –** Again, this should not occur as the AMF firewall does not allow it.

Refer to the diagram below for additional information.



3 During

The first phase of the project was scheduled to start on the second week of August 2004. I say the first phase because tuning false positives on your NIDS is an on-going process, not a completion target. New attacks bring with them new NIDS signatures and filters and these bring with them potentially new false positives. It is a never ending process.

The following section details the methodology which I used to tune some of the false positives out of the NIDS. Most literature on the Internet regarding this subject was a little fuzzy and placed most of the work on the analyst's previous experience in carrying out NIDS tuning. I decided to put together a flowchart based approach to determining whether identified events were indeed false positives or confirmed attacks on AMF's I.T. infrastructure. This methodology is described in the sections below along with some example applications of this methodology.

3.1 The road map

Prior to starting the tuning process, the following project objectives were determined:

1. Tune out the maximum number of false positives from the NIDS system monitoring the 4 DMZs.
2. Create a tuning methodology which could not only be used for this project but could also be used by less experienced AMF NIDS analysts as a starting point for on-going NIDS tuning. This methodology would provide a template for NIDS Analysts to continue false positive identification and eradication and AMF.
3. Minimize the possibility of NIDS Analyst de-sensitization. This occurs when an analyst sees the same false positive over and over (and have analyzed and dismissed it a few times); they can become de-sensitized to that false positive. This means that when that false positive is an actual attack e.g. same attack, different source IP address, the analyst may not pay the deserved attention to the event.
4. Respond and communicate with AMF's Incident Management Team (IMT) when confirmed attacks were detected.

Additionally, the following requirements were determined:

- An experienced NIDS analyst should carry out the tuning processed (guess who the bunny that got picked for that job was ?)
- The analyst must have NIDS visibility of all four monitored DMZs at all times. This means minimal system down time.
- Prior to implementation of NIDS filters to block the false positives, another NIDS analyst would have to concur with the decision. This two-person control would ensure that the false positive was validated as such.
- As no system is 100% bullet proof, ensure an effective incident response path has been implemented.

3.2 Implementation

Day 1 arrives. I get to work, grab the strongest cup of coffee I can find, pump some Pixies through the head-phones and carry out the following tasks :

1. Log into the NIDS console and print out a report detailing the detected events for the past 24 hours.
2. If not many events were detected, I would apply the methodology detailed below to all events one by one and identify any false positives which were present. If the detected event count was high, I would determine events of interest by applying the following criteria :
    - Target – Potential, concentrated attacks targeting business critical entities as detailed in previous risk assessments.
    - Timing – Multiple attacks either originating from the same IP address or network or multiple attacks targeting business critical infrastructure.
3. Once I flagged the events of interest, I would apply the tuning methodology detailed below.
4. If a false positive was identified, I would enter it into the false positive register (see Appendix A below), commence the two-person validation process and create the tuning filter if required. If a confirmed attack was discovered, I would kick the incident handling process into gear.

Sounds easy, right ? Well, it wasn't. Anyone who has ever had to tune a NIDS will tell you that it's never as simple as that. Although this routine would and will be carried out over and over during the next 3 months, there are a great many instances in which nothing is black and white. Sometimes you have to rely on your knowledge and experience to make a judgement call. Having said that, the methodology detailed in the next section has helped me develop a starting point for false positive identification and analysis.

3.3 The tuning methodology

As any NIDS analyst will tell you, tuning out false positives is probably one of the hardest and most un-glamorous tasks which have to be carried out during a NIDS deployment. This is perhaps because structured methodologies to fast track this process are few and far between. Usually, the analyst has to step through every event flagged and make a judgment call based on his/her experience as to whether to create a tuning filter or not. One of the requirements of this project as stated above was to design a false positive tuning methodology that could be used as a starting point for less experienced NIDS analysts. The following section details the proposed tuning process to minimize false positives within AMF's NIDS infrastructure. It is important to note that the methodology detailed below is very much based on traffic flow. By this I mean that certain components should talk to each other in certain ways whilst others shouldn't. Source and destination IP addresses play a big part in this methodology. Target ports are also important but are left to be investigated further during the "Real Attack" condition testing of the equation. This is because false positives can occur on open ports. A port does not have to be closed on a destination host for an event to be a false positive. For example, if I had a condition like "IF Dest_port = Not Open THEN FP=True", I'd be

leaving myself open to incorrectly rejecting a potential false positive.

Not all false positives can be eradicated, but following a structured tuning methodology will result in the removal of a large number of false positives. However, this can only be accomplished over a period of time. It is not an over night achievement.

Additionally, NIDS tuning is closer to art than science. For this project, I had to not only understand TCP/IP but also AMF's network layout, server and desktop environments, attack methodologies and communications protocols.

The communications paths which were addressed during the NIDS tuning process are :

- **External to DMZ (EXTERNAL to AMF_Public_DMZ)**
- **DMZ to DMZ (AMF_Public_DMZ to AMF_Public_DMZ)**
- **DMZ to Internal (AMF_Public_DMZ to AMF_INTERNAL_NETWORK)**
- **Internal to DMZ (AMF_INTERNAL_NETWORK to AMF_Public_DMZ)**

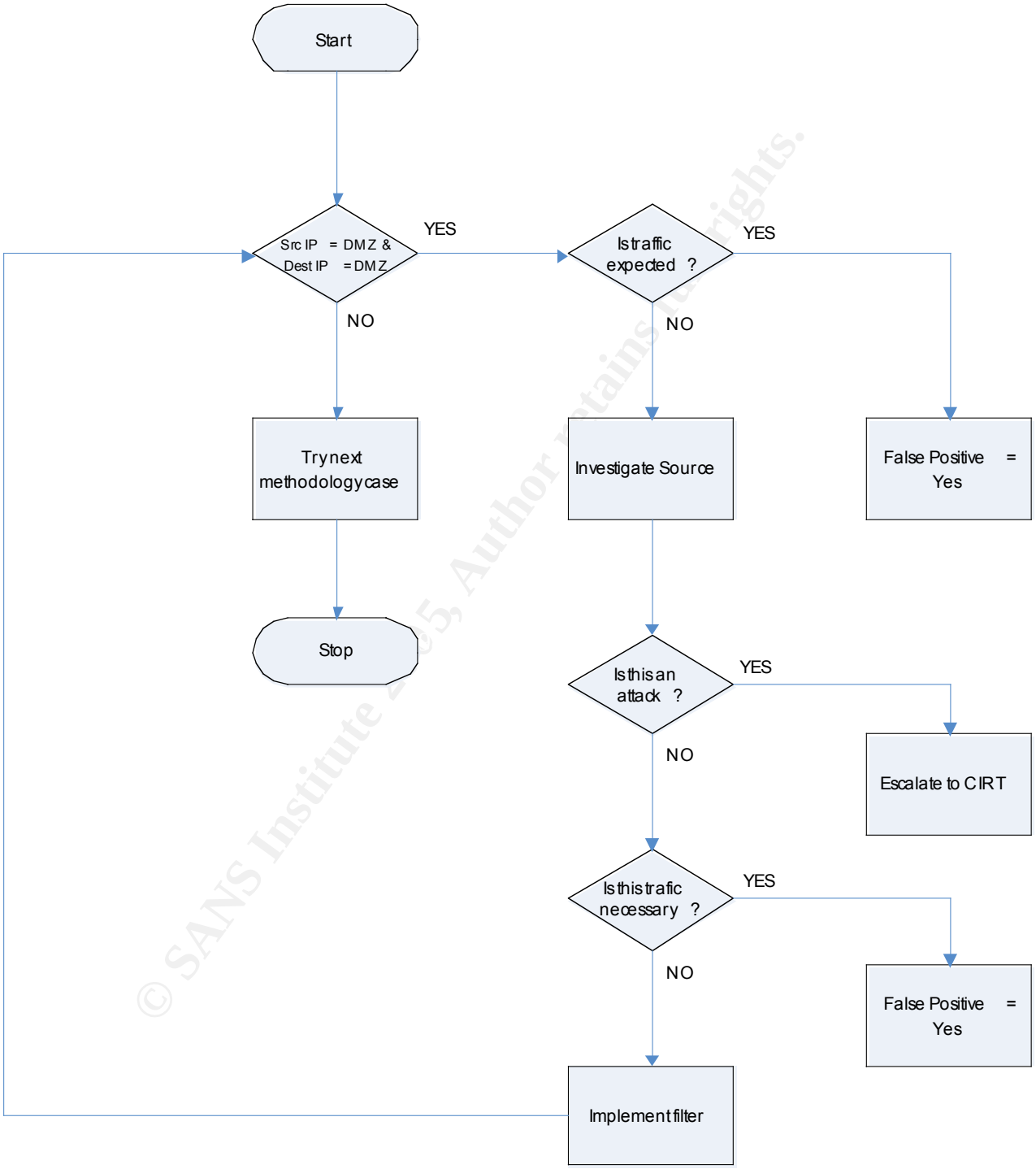
Prior to commencing, I ensured the NIDS was logging all traffic. Once this was confirmed the following process was applied :

1. Log into the ManHunt console.
2. Generate an activity report for the last 24 hours OR view activity for the last 24 hours.
3. Gather the following details from each event of interest:
    - **Source IP address**
    - **Destination IP address**
    - **Event type**

4. Once this information was gathered, I applied the following methodology :

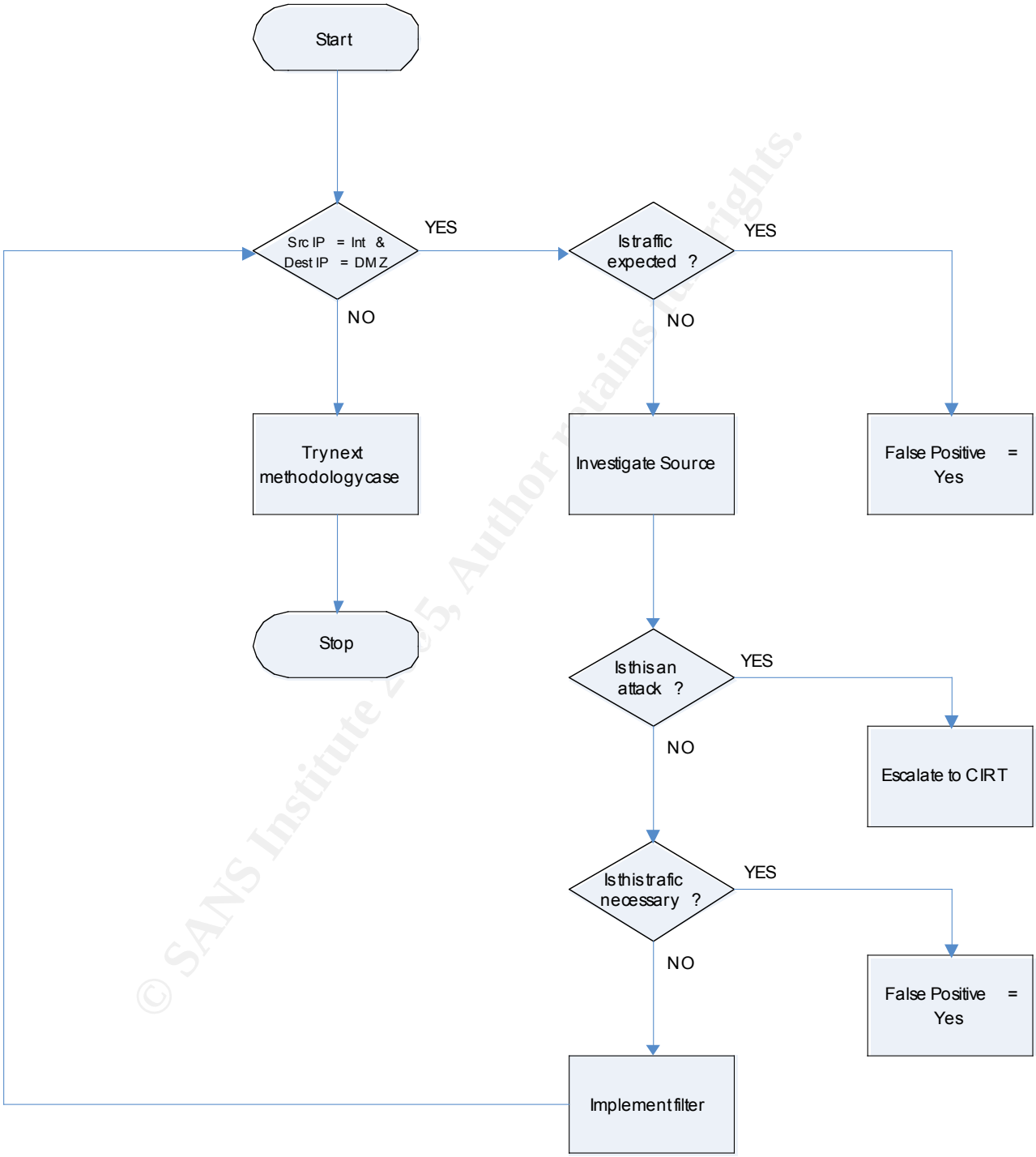
DMZ-To-DMZ

Case 1. DMZ entities talking to other DMZ entities. In this case, AMF DMZ servers talk to each other, either for management purposes (i.e. SNMP) or due to compromise (i.e. compromised server trying to compromise other servers).



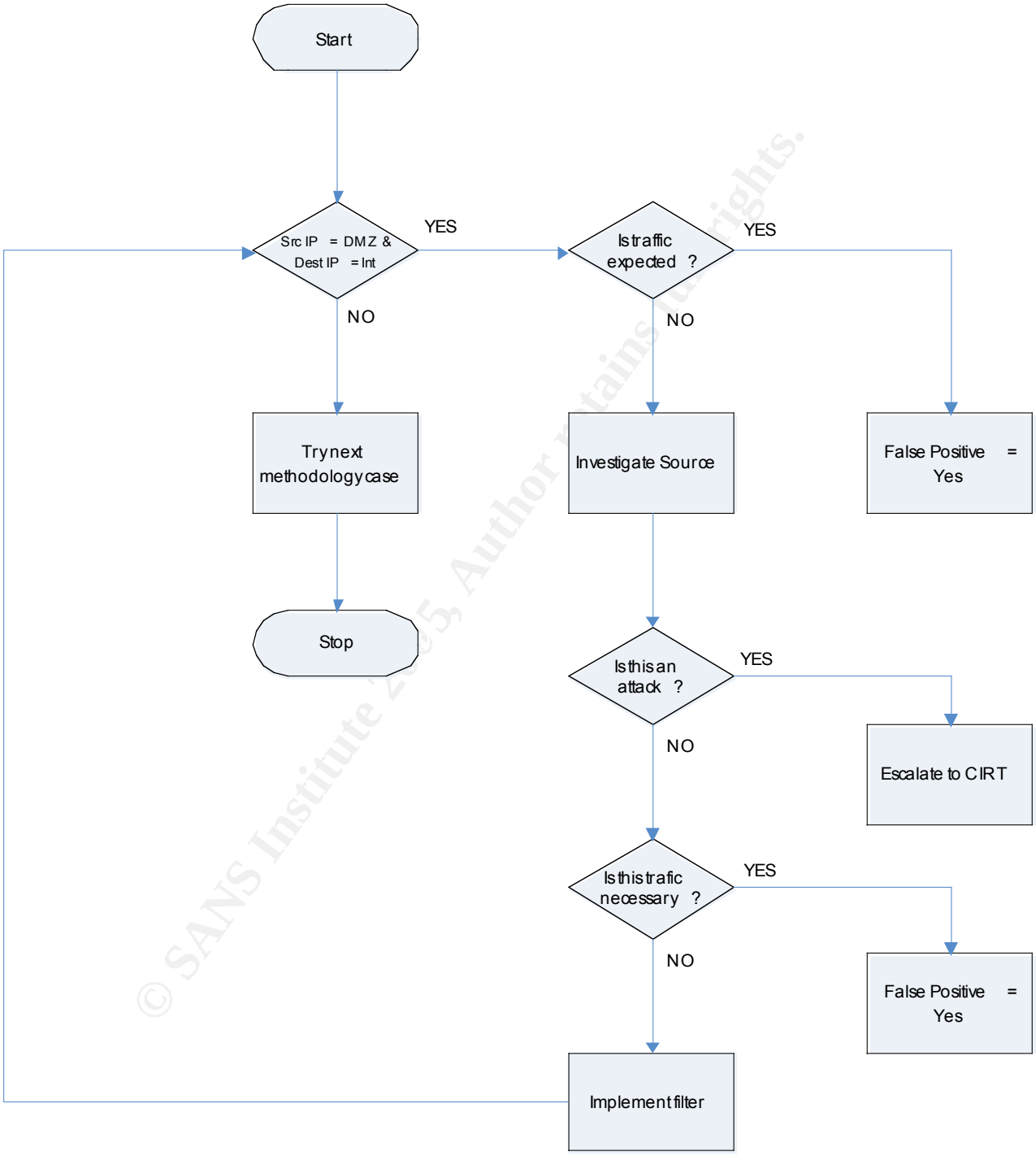
INT-To-DMZ

Case 2. Internal entities talking to DMZ entities. In this case, internal resources might talk to DMZ resources such as a desktop accessing a proxy server for Internet access or an SSH client transferring new content to a Web server for publication.



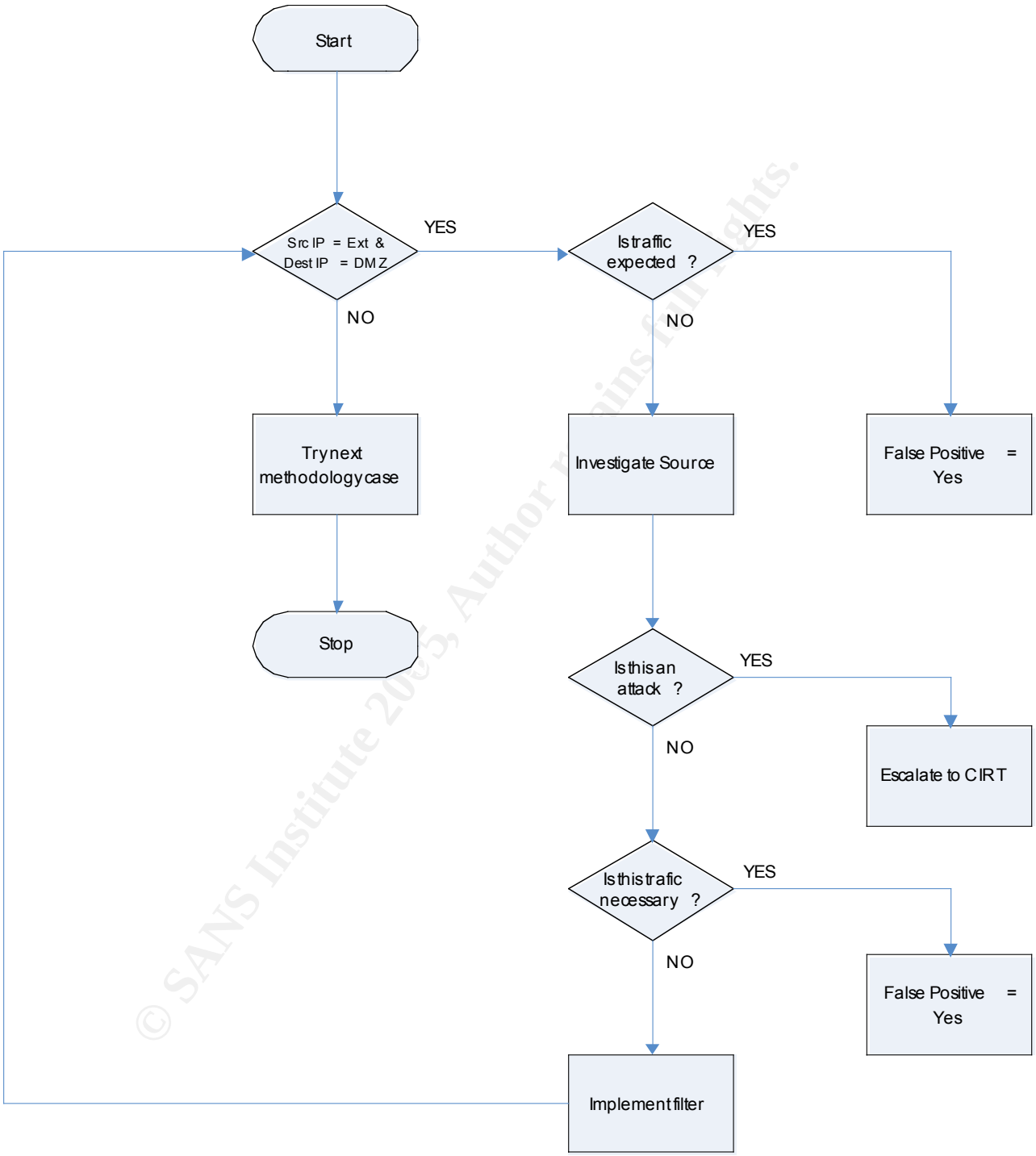
DMZ-To-INT

Case 3. Traffic from the DMZ to the Internal network. In this case, traffic flows from the DMZ entities to components on the internal network segment(s). Examples of this might be security and management agents talking to back end management servers or Web server front ends talking to backend database systems.



EXT-To-DMZ

Case 4. External (public) talking to DMZ servers. Any traffic allowed through your firewall to your public facing DMZ fits in here.



5. Have another NIDS Analyst validate the false positive
6. Document false positive in the False Positive Register and create a filter in ManHunt to filter out this Event.

Refer to the next section for examples of how to apply the methodology. Real world IP addresses have been changed to protect both the innocent and the (possibly) guilty.
NOTE: Again, it is important to reiterate that although this methodology will work well for most false positives, it is critical that the NIDS analyst utilize some discretionary judgment and treat each potential false positive on a case by case basis.

3.3.1 Example #1 : Ping Scan

The following section applies the methodology detailed above to a ping scanning attack. This scan was initiated from within AMF's public DMZ and targeted every other server in AMF's public DMZ. However, all is not what it seems. The output has been shortened for brevity's sake. There were approximately 2,500 entries in the original incident.

Date/Time	Source IP	Destination IP	Protocol	ManHunt Event Type
12/072004 12:30	10.xxx.xxx.xxx	10.xxx.xxx.xxx	ICMP	ICMP_Enum_Scan

Application of the proposed methodology provides us with the following results.

Source IP : **10.xxx.xxx.xxx**
Destination IP : **10.xxx.xxx.xxx (entire network segment)**
Attack : **ICMPEnum Scan**

What we have here is a situation where the event's source host had some management software installed on it which by default sends out ICMP type 8 packets (Echo Request) to all other IP addresses on the same segment. These servers would then send ICMP type 0 packets (Echo Reply) back. The software did this every minute or so to see if the other servers were up. This created a situation where the NIDS flagged an event because the characteristics of the ICMP packet matched the following event characteristics in the ManHunt NIDS:

> ***ICMPENUM Scan****ICMPENUM_SCAN*
> ***General*** *This signature detects a scan of your network performed with the icmpenum tool.*
> ***Details*** *Icmpenum is a proof-of-concept tool used to demonstrate possible distributed attacking concepts.*
> ***Affected*** *ICMP*
> ***Possible False Positives*** *There are no known false positives associated with this signature.*

As can be seen above, the packets sent by the management software matched the same characteristics as the ICMPEnum tool. Inspection of the source server revealed that ICMPEnum was not installed, only this new management software. Once the software was re-configured, the pings stopped and so did the associated NIDS events.

3.3.2 Example #2 : FTP Brute Force attack

Web and FTP server attacks are quite common on the Internet. The following attack targeting an FTP server in AMF's public DMZ appears to be malicious but it's not. The output has been shortened for brevity's sake. There were approximately 230 entries in the original incident.

Date/Time	Source IP	Source Port	Destination IP	Destination Port	Protocol	TCP Flags	ManHunt Event Type
24/09/2004 10:09	203.xxx.xxx.xxx	5428	165.xxx.xxx.xxx	21	TCP	***AP***	FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5398	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5405	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5409	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5411	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5437	165.xxx.xxx.xxx	21	TCP	***AP***	FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5398	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5405	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5409	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5411	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5428	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5439	165.xxx.xxx.xxx	21	TCP	***AP***	FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5398	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5428	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5437	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5405	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5409	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5411	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5446	165.xxx.xxx.xxx	21	TCP	***AP***	FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5398	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5405	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5409	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5411	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5428	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES
24/09/2004 10:09	203.xxx.xxx.xxx	5437	165.xxx.xxx.xxx	21	TCP		FTP_TOO_MANY_TRIES

Application of the proposed methodology provides us with the following results.

Source IP : **203.xxx.xxx.xxx**
Destination IP : **165.xxx.xxx.xxx**
Attack : **FTP Brute Force Attack**

This event required some investigation. What was discovered was quite simple. The FTP account's password had recently been changed without notification. An automated file transfer process initiated by a business partner was trying to log into the FTP server with the old password and upload some files. Due to the password change, the login kept failing and triggering the NIDS events we see above. Although this was a false positive, additional issues of change control and change notification were raised with the relevant department, as it demonstrated a breakdown in business security policy.

3.3.3 Example #3: Cisco DoS attack

Denial of Services attacks are some of the scariest attacks to deal with because they target the availability of a system (or lack thereof). The following Cisco DoS attack appears to be the real deal but it's actually something totally different.

Date/Time	Source IP	Source Port	Destination IP	Destination Port	Protocol	TCP Header Flags	ManHunt Event Type
24/09/2004 15:12	203.xxx.xxx.xxx	7161	165.xxx.xxx.xxx	20	TCP	***A**S*	CISCO_CAT_DOS

Let's apply the methodology detailed above to the details above to determine if this is a false positive or not.

Source IP : **203.xxx.xxx.xxx**
Destination IP : **165.xxx.xxx.xxx**
Attack : **Cisco Catalyst DoS**

Close inspection of the attack signature shows that the Cisco Catalyst attack targets port 7161 on Cisco Catalyst Switches. What we have here is an FTP data stream where the client's ephemeral source port selected just happened to coincide with port 7161. When the NIDS parses the packet and discovers this port, it does not look at the packet payload for additional signs of attack. Instead, it just thinks that a Cisco Catalyst DoS is in progress (based on the port number) and raises an event. An important point to make here is that the less specific an event is, the greater the possibility of it being a false positive. As an example, a NIDS signature that only detects on port number (such as the Cisco Dos one above) has a greater chance of generating a false positive that a signature that flags an event based on port number and payload content.

As can be seen, this case is a false positive. However, some additional investigation had to take place prior to reaching this decision. What we see here is that there was no attack to stimulate the response. Instead, what triggered the NIDS was the response itself, the characteristics of which just happened to match a pre defined signature.

We can see from the previous examples that the methodology works well. However, investigation on a case by case basis still has to be carried out by the NIDS analyst in order to determine if the event is a true false positive or not.

4 After

The following section details the post-project implementation debrief and benefits. Whilst the project was personally rewarding and educational, a lot of hard work was carried out to ensure that the project finished on time and met the required objectives.

4.1 Post implementation diagnosis

So how successful was the project and the associated methodology ? Overall a high degree of success was achieved. Approximately, 65% of false positives flagged by the NIDS were identified and addressed. Additionally, two new NIDS analysts are currently using and refining the proposed methodology for on-going false positive tuning. The following benefits of the NIDS tuning project were identified post-implementation :

- 65% of false positives flagged by the NIDS were identified and addressed.
- Increased ability to focus on confirmed attacks.
- Decrease in NIDS noise generated by false positives.
- NIDS analyst education fast tracking due to methodology implementation. However, on-going refinements to the methodology will be carried out.
- Less burden placed on incident handling team to have to investigate potential attacks which turn out to be false positives.
- Greater understanding of AMF's Internet footprint, network infrastructure and implemented communications protocols.

4.2 Residual risk

As detailed in the Before section above, a number of risks to the project were identified during the planning phase. This section details how the implementation of the project impacted these risks. In all cases, the project implementation had an impact on each risk, either through risk eradication or risk minimisation.

- **Lack of resources to carry out tuning and analyst in-experience**

  Luckily, resources were available to carry out the project. These resources included not only NIDS analysts and trainees but also system administrators which provided access to the required servers and desktops for investigation proposes.

- **Lack of identifiable false positives within project time frame**

  No problems here. The saying "if you build it, they will come" certainly applied to AMF's systems. The NIDS generated events from external sources, internal sources and sources within the public DMZ itself.

- **NIDS down time**

  NIDS down time did not impact the project. This was perhaps because there was none. Due to ManHunt's architecture, signature updates, patch updates and log rotation do not require the system to be rebooted or brought offline at any time. However, this does not mean that it may not have to be done in the future.

Therefore, a project is currently underway to build redundancy into the existing NIDS infrastructure.

4.3 On-going tuning

The technological landscape is ever changing and evolving. As previously mentioned, on-going NIDS tuning is an activity which must continue to be carried out indefinitely. This must happen because false positives will continue to be flagged as more technology is developed, deployed and changed and more attacks are developed and deployed to compromise this technology. AMF is no exception to this rule. Post project, an on-going NIDS tuning methodology has been implemented. The tasks which are carried out are pretty much the same as those carried out during the project which are detailed in previous sections. The two big differences with on-going tuning are i) implementation of the proposed methodology and ii) NIDS analyst education. These are detailed below.

Methodology – The methodology proposed above has now been fully implemented at AMF. NIDS analysts utilise it as part of their daily tasks to tune new false positives out of the NIDS system. One of the best parts of having other NIDS analysts using the methodology is that tweaking of the methodology can be carried out, ensuring that it evolves to meet the changing needs of the organisation. Two items that have already been amended are incorporation of multiple DMZ functions and greater packet payload investigation.

NIDS Analyst education – One of the benefits of the project was that it provided the less knowledgeable NIDS analysts with a greater understanding of NIDS event identification, investigation and tuning. For some, the learning curve was quite large. Although TCP/IP, system and network administration and other disciples were skill sets located within the group, the ability to investigate and respond to incidents was something which had to be learnt from scratch for some. Additionally, the project ensured that a thorough understanding of AMF's existing network, server and desktop infrastructure was acquired by all involved through investigations of flagged events.

The on-going tuning process is scheduled to be reviewed in six (6) months time. This will ensure that the processes for on-going NIDS tuning are as functionally appropriate then as they were at the time of their implementation during the project.

4.4 Summary

All in all, the project was very successful. The original objectives were met within the required timeframe and some additional, positive side effects were also experienced, the main two being additional analyst training and better understanding of AMF's infrastructure. In summary, the following project outcomes were experienced :

- Decline in false positives by approximately 65%.
- Implementation of a NIDS false positive tuning methodology.
- Increase in NIDS analyst training and network knowledge.
- Better understanding of traffic hitting AMF's networks.

In summary, I hope you found this paper both entertaining and educational. If nothing else, I hope the reader takes away with them some ideas for use as a starting point to tuning false positives out of their Network Intrusion Detection Systems.

Appendix A – False Positive Register

The following appendix details the False Positive Register. The False Positive Register details the NIDS signatures that have been flagged as false positives along with other false positive relevant information such as date and time analysed and source and destination IP addresses and ranges. The NIDS Analyst carrying out the tuning exercise fills in the false register once the false positive detected has been validated by a second NIDS Analyst. The register allows security staff to keep track of false positives detected.

Date/Time	Event Name	# of events	SourceIP	DestIP	Traffic Direction	Filter in place (Y/N)
12/072004 12:30	ICMP_ENUM_SCAN	2,500	10.xxx.xxx.xxx	10.xxx.xxx.xxx	DMZ-to-DMZ	**Y**
24/09/2004 10:09	FTP_TOO_MANY_TRIES	230	203.xxx.xxx.xxx	165.xxx.xxx.xxx	External-to-DMZ	**Y**
24/09/2004 15:12	CISCO_CAT_DOS	1	203.xxx.xxx.xxx	165.xxx.xxx.xxx	External-to-DMZ	**Y**

Appendix B - References

Internet

Brook, Jon-Michael C. "Network IDS: To tailor, or not to tailor." 6 March 2002. URL: http://www.sans.org/rr/papers/index.php?id=357 (24 Oct. 2004).

Timm, Kevin. "Strategies to Reduce False Positives and False Negatives in NIDS." SecurityFocus InFocus. 11 September 2001. URL: http://www.securityfocus.com/cgi-bin/sfonline/infocus.pl?id=1463 (24 Oct. 2004).

Ranum, Marcus J. "False Positives: A User's Guide to Making Sense of IDS Alarms." February 2003. URL: http://www.icsalabs.com/html/communities/ids/whitepaper/FalsePositives.pdf (24 Oct. 2004).

Dubrawsky, Ido and Saville, Rolland. "SAFE: IDS Deployment, Tuning, and Logging in Depth." The *SAFE: A Security Blueprint for Enterprise Networks*. URL: http://www.cisco.com/en/US/netsol/ns340/ns394/ns171/ns128/networking_solutions_white_paper09186a00801bc111.shtml (24 Oct. 2004).

Symantec Corporation. "Symantec ManHunt 3.0 Event Reference." 2004. URL: ftp://ftp.symantec.com/public/english_us_canada/products/manhunt/3.0/manuals/Symantec_ManHunt_Event_Reference_20041014.pdf (25 Oct. 2004).

Books

Northcutt, Steven. Network Intrusion Detection An Analyst's Handbook. Indianapolis: New Riders Publishing, 1999. 89 – 92.