Global Information Assurance Certification Paper

Timothy W. Muchow
GSEC Practical Assignment version 1.4b
Option 1

Closing in on Intrusion Management

Introduction

Adding an Intrusion detection system (IDS) to your network sounds simple enough. Just turn it on and you'll know about every intruder that tries to do evil on your treasured network and you'll be in position to stop his deed. Anyone who's activated an IDS already is laughing. Default installations simply feed too much information to the viewing analyst. Even well tuned IDS' create overwhelming amounts of data. If the network you are overseeing is for a large or enterprise sized corporation the volume is mountainous.

Arcsight, is one tool attempting to help security analysts find the events of importance, (the needle in the haystack) hidden within this wall of data. Peter Stephenson refers to this as Intrusion Management and notes:

> "*The Intrusion Management model is based upon the assumption that the ultimate goal of the information protection process must be three-fold. First, the process seeks to protect information assets from compromise. Second, the process must recognize that compromise is inevitable and that measures must be taken in advance to facilitate a means for investigating the compromise and recovering, managing and protecting the evidence of the compromise for future use in legal proceeding. Finally, the process must provide feedback that can speed response to a compromise and generate information that can be used to prevent similar compromises in the future*"[1]

This practical will discuss using Arcsight features to work towards the second piece of this management goal. Reducing false positive events and closing the gap between event and response. To keep things simple, source event examples will be limited to SNORT, "*The Open Source Network Intrusion Detection System*"[2]. A short review of snort basics is included as a foundation for how Arcsight interacts with the IDS data. This interaction is critical to successful integration of the pieces.

---

[1] Stephenson, P. (2000) "Intrusion Management: A top Level Model for Securing Information Assets in an Enterprise Environment" EICAR 2000 Best Paper Proceedings, 288 URL: http://www.conference.eicar.org/past_conferences/2000/papers/Tuesday/Security%20Trust%20and%20E-commerce/other/Stephenson.pdf 26 September 2000

[2] Created by Martin Roesch, and available @ http://www.snort.org/

Snort

Snort is free, This is only one reason for its huge popularity. It is also easy to install, well documented and numerous newsgroups are out there, with help for any question or problem you may encounter. New exploits are often quickly addressed, by Security the community at large. Newsgroup members finding a viable new rule, created by and tested by Snort users just like you, are shared openly. There's even a Corporate version, Sourcefire Network Sensor[3], by Sourcefire, with full support .

Snort is a signature based IDS. It only sees what it knows to look for and alerts each time it sees a matching signature. Snort looks at a packet of data and either alerts or discards the packet. Each alert adding to the data that needs to be sorted through. Each new signature adds to the problem, As Arnt Brox, Chief CEO of ProseqAS notes:

> "*it means that the greater the number of signatures searched for, the correspondingly higher the probability of identifying more false positives, especially on smaller and simpler signatures*"[4]

Reducing these false alerts is just one of the Arcsight features we'll look at.

Output from Snort can be either ASCII or binary (binary output is generally preferred, it allows for review packet), and with optional plug-ins output to a database is possible. Database output is key to incorporating Arcsight, regardless of IDS system being used as a source. Once data is placed in the database Arcsight is able to scrounge for details needed to make a match

Snort uses a user configurable set of rules to tell it what to look for inside network data traffic (the signature). Once Snort recognizes a match for an active rule, an alert is generated and logged to either the log file or database, from this point forward we'll assume the database is active. Rules can be based on nearly anything contained within the network packet. Content, (anywhere within the payload), Protocols, Flags (**Urg**ent, **Ack**nowledge, **Push, Reset, Syn**chronize, **Fin**ish), Time to live (TTL), fragmentation, data payload size etc., all can be used by Snort as rule triggers. A configurable option allows for the capture of a portion or the entire data packet (I'd recommend capturing whatever your hardware will let you, there's always a cost either to the CPU cycles or your storage space so this varies by installation) A typical Snort rule looks like this example from the Snort rule set version 1.9

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 139 (msg:"NETBIOS nimda
.nws";content:"|00|N|00|W|00|S";flow:to_server,established;classtype:ba
d-unknown;reference:url,www.datafellows.com/v-
descs/nimda.shtml;sid:1294;rev:6;)[5]
```

[3] http://www.sourcefire.com/

[4] Brox, Arnt. Signature Based or Anomaly Based Intrusion Detection – The Practice and Pitfalls, 2 February 2002. URL: http://www.itsecurity.com/papers/proseq1.htm (30 September 2002)

[5] Snort users manual available @ http://www.snort.org/docs/writing_rules/

A detailed look shows that when Snort sees this data it

- Alerts with message… "NETBIOS nimda .nws",
- based on …TCP data,
- from ANY external host…. connecting to port 139 on,
- with the ACK flag (A) (and possibly others indicated by the + sign) and
- containing ASCII text "|00|N|00|W|00|S " (hex data would be bracketed between Pipes like this… "|data|").

The rule may even note where reference material on this exploit can be found (SID: # for SNORT signature Identifier[6], CID:# for Common Vulnerabilities and Exploits (CVE) entry or candidate number [7], or as in this example
www.datafellows.com/v-descs/nimda.shtml

Snort assigns a default priority level to most alerts based on groupings referred to as classtype these include:

High Priority Classifications - Priority 1
Medium Priority Classifications - Priority 2
Low Priority Classifications - Priority 3

More information on default priority is available in the Snort Users manual available at: http://www.snort.org/docs/writing_rules/chap1.html. These priority levels, and each vendors default priority levels are generally forwarded by Arcsight to the console. Each company looks at the priority of events differently. We'll look closer at this feature when we move on to ArcSight.
Snort provides options for tuning out the traffic (pass rules) that is normal and expected on your network. Configuring Snort so expected traffic does not cause alerts on every occurrence. Tuning the source of data that Arcsight is receiving is the first step in getting to the event of interest or our needle in the haystack

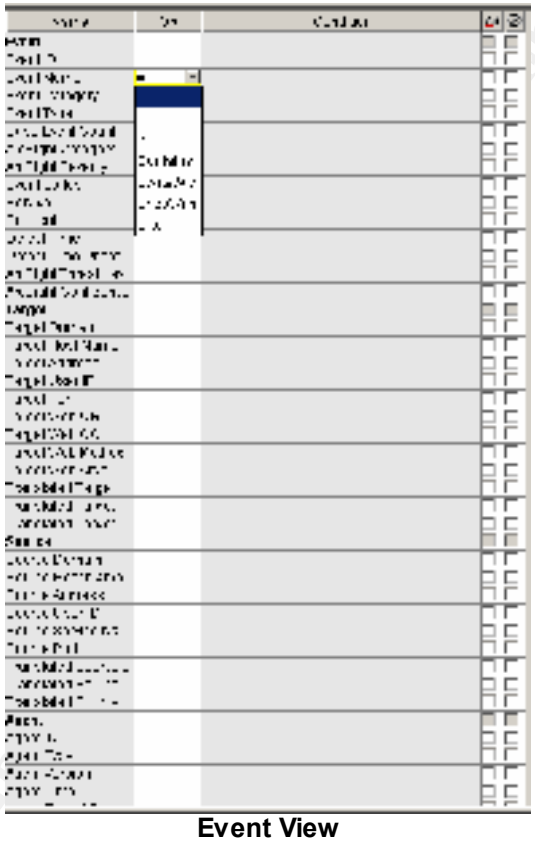
ArcSight Basics
ArcSight consists of three (3) pieces. The SmartAgent (agent), the ArcSight Manager (manager) and the ArcSight Console (console). The agent needs to be able to access the original data. In our example using Snort, The Snort Arcsight agent needs access to the database that the Snort sensor is forwarding alerts to, (pulling directly from the log files is an option, depending on source vendor). The Agent is designed specifically to deal with data from each vendor (separate agents are available for most of the Firewall systems, Network and Host based Intrusion systems out there, the complete list is available in PDF form from Arcsight @ http://www.arcsight.com/product.htm). The agent forwards the source alerts to the manager. The vendors default priority is forwarded from the agent

---

[6]Snort signature database @ http://www.snort.org/snort-db/

[7] Common Vulnerabilities and Exploits database @http://www.cve.mitre.org/

and becomes the ArcSight severity level for the event. Event severity levels may be modified at the agent allowing for custom settings.
The manager converts forwarded alerts to an easily viewed format. Adding each detail of the event to Arcsight event fields. These fields will include a unique identifier, event name (alert "message" from our snort example), severity (priority level from our Snort example), categories (used to group related events from different vendors), and others. The original IP header is decoded by Arcsight and added to the new event. This decoding simplifies viewing so that details like source and target IP address' are easily viewed . Other fields may include ports used, protocol type. The event view below shows some of the detail available



Event View

Now that the manager has the data, the event is then forwarded, real time (I know, NOT real time, just close as these tools allow) to the console and to a database connected to the manager, where this data can be securely stored, reviewed or compared with other events. Once stored the data can be retrieved using ArcSight queries. These allow for queries based on a single piece of data or its absence or combinations of data (query availability varies by field…for example: time based queries only apply to time fields). The following list of queries is current in ArcSight 2.0.0

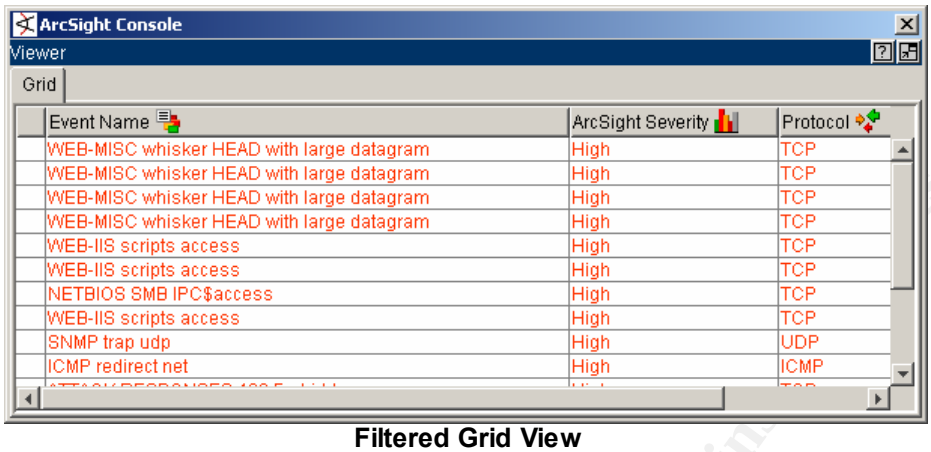
Equal (=)	Looks for exact match of single piece of data, (an IP address, etc)
Not Equal (!=)	Anything that does NOT match, a single piece of data (NOT this address)

In (IN)	Looks for any match from within a group
Contains	Looks for partial match (such as any event that contains "Nimda")
Between	Time based, allows for hourly, daily, weekly, monthly queries
Subnet	Within subnet (10.10.10.0/24 for example)
Like	
Starts With	Matches
Ends With	Contains
On	Exact time matches

These queries are the meat and potatoes of ArcSight. The Arcsight rule generators and report generators are all built on these queries

The last piece is the Console. The concept is really pretty simple. There's too much information. We (the Intrusion analyst) need some way of managing of all this information. Arcsight Manager is a correlation engine. It stores the data, then lets you decide what is of value. The console is the presenter of this information. The console lets the viewing analyst decide what information is presented and how. The consoles key pieces include The Grid (used to view multiple events, either as they occur, or close), event viewer (used to view the details of the event, including the payload, if available), reports (allows the analyst to collect historical data based on any field data is collected on and present this in chart or table forms), rules (rules look for combined events, or correlations and forward to the grid or perform an action like paging out each time this is detected) and dashboards (these visually present issues or changes). Using filters, rules and dashboards in combination to point the analyst quickly at likely valid events is where the meat of this discussion lies.

The Grid.
The ArcSight Grid is the main viewer of events. These events can come from real time or a replay (or query) of past events, stored in the database. Further the Grid being viewed may filter the events so that only the events desired are seen. Filter options may include events of a single type such as our Nimda alert noted earlier or may be based on the ArcSight severity. The example below is of a filtered Grid showing only Very High or High ArcSight severity events

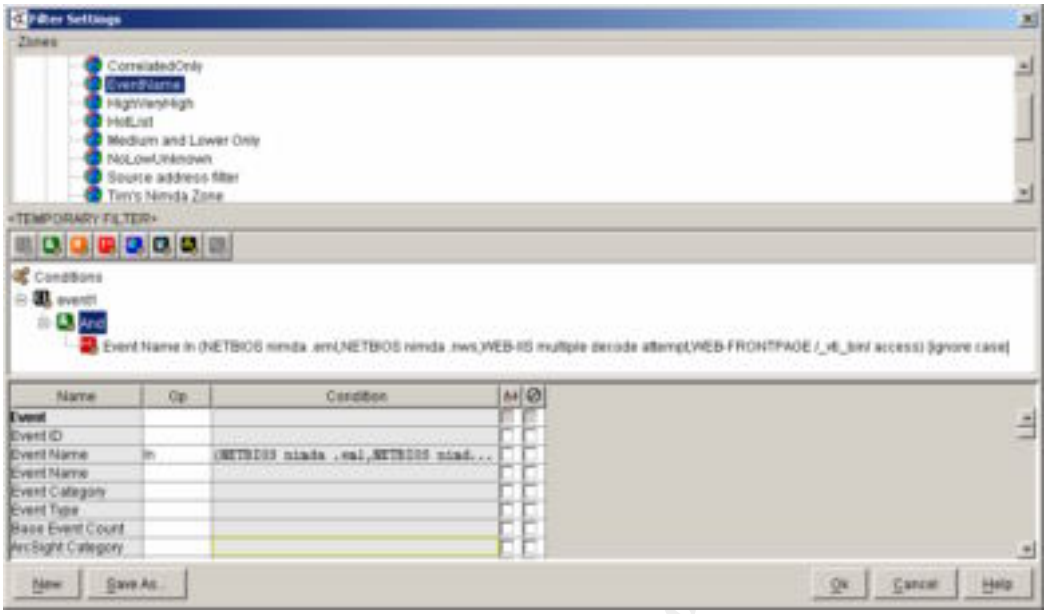


Filtered Grid View

The filter used the create this display looks like

Very High and High Filter View

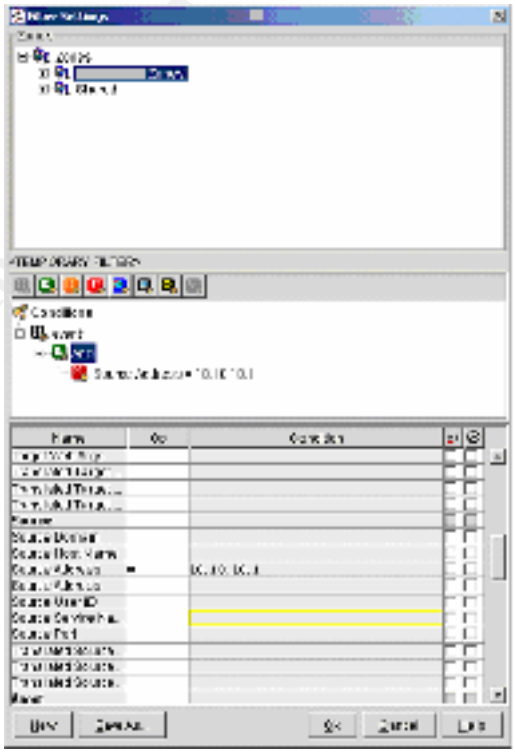
In our Nimda example from Snort a filter might look like



Nimda Filter View

The above filter is based on details of the Nimda Virus taken from CERT® Advisory CA-2001-26 Nimda Worm[8]

or a specific suspected intruder



[8] http://www.cert.org/advisories/CA-2001-26.html

IP Address filter

While attempting to reduce the volume of data being viewed, the analyst may set filters so only one or two of the Nimda related events are seen on his Grid. Though this would reduce the volume, it doesn't reduce false alerts and leaves possible gaps. The chance of an infected host being missed still exists. Though filtering is a great feature, it requires that you know what to filter for. What event is involved? What the IP address of the target host or source address of the intruder is? These are very likely to not yet be known. Waiting until an event of interest occurs then querying for the related events based on its details, may lead the analyst to a target (our needle) on occasion. There is a large delay between the developing steps…We're still looking to close the time gap and reduce looking at false alerts. As David Blackman suggests:

> *"IDS must evolve beyond its point product tradition and encompass a new level of management capabilities. Companies have to focus on managing events, correlating them, and responding to them in real time."*[9]

Concentrating our efforts to real issues as they occur is our focus and Arcsight has additional tools to help us close in on this goal.

Arcsight Rules

ArcSight's ability to correlate or associate data is at the core of reducing the volume of or the false alerts the analyst spends his time on. ArcSight uses devices it call Rules to construct these correlations. Creating a rule involves choosing data from more than one related event or event type and linking these events. The rules can be very simple or extremely complex.

Taking a look back on our Nimda example, the CERT advisory (CERT® Advisory CA-2001-26 Nimda Worm **)** indicates a combination of file propagation and a series of footprints or signatures left by the Nimda Virus that are associated. I've captured a number of these (though not all for simplicity) and the relating Snort alert generated when the sensor sees this Nimda data traffic

Snort Alert-*"*WEB-IIS CodeRed v2 root.exe access*"*
Footprint
GET /scripts/root.exe?/c+dir

Snort Alert-"*WEB-IIS cmd.exe access"*
Footprint
GET /c/winnt/system32/cmd.exe?/c+dir
GET /d/winnt/system32/cmd.exe?/c+dir
GET /scripts/..%5c../winnt/system32/cmd.exe?/c+dir

Snort Alert-*"WEB-FRONTPAGE /_vti_bin/ access"*
Footprint
GET /_vti_bin/..%5c../..%5c../..%5c../winnt/system32/cmd.exe?/c+dir

---

[9] Blackman, David, Intrusion Detection is failing: Enter Intrusion Management. 29 July 2002 URL: http://www.itsecurity.com/papers/pentasafe1.htm (28 September 2002)

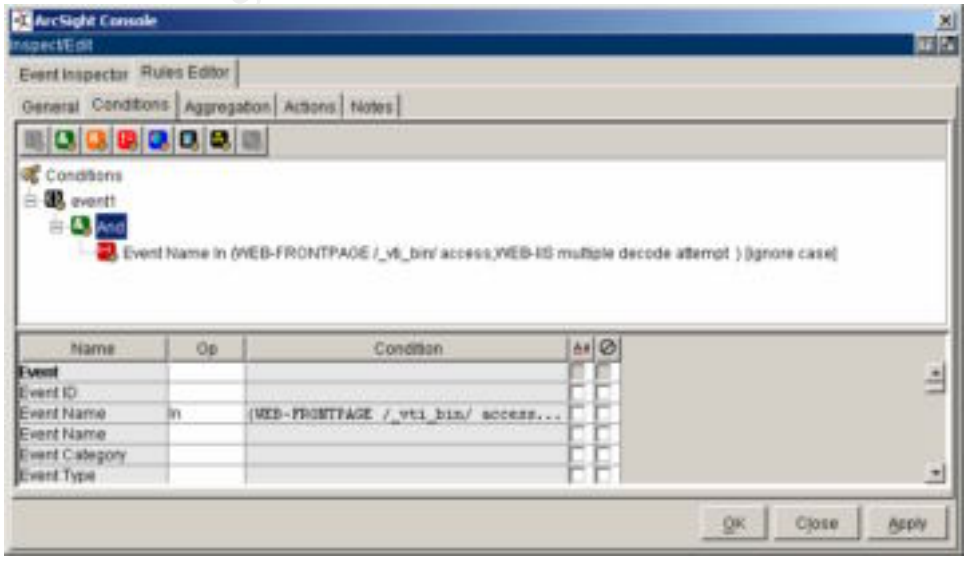
Snort Alert-*"WEB-IIS multiple decode attempt"*
GET /_mem_bin/..%5c../..%5c../..%5c../winnt/system32/cmd.exe?/c+dir
GET
/msadc/..%5c../..%5c../..%5c/..\xc1\x1c../..\xc1\x1c../..\xc1\x1c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%5c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%2f../winnt/system32/cmd.exe?/c+dir

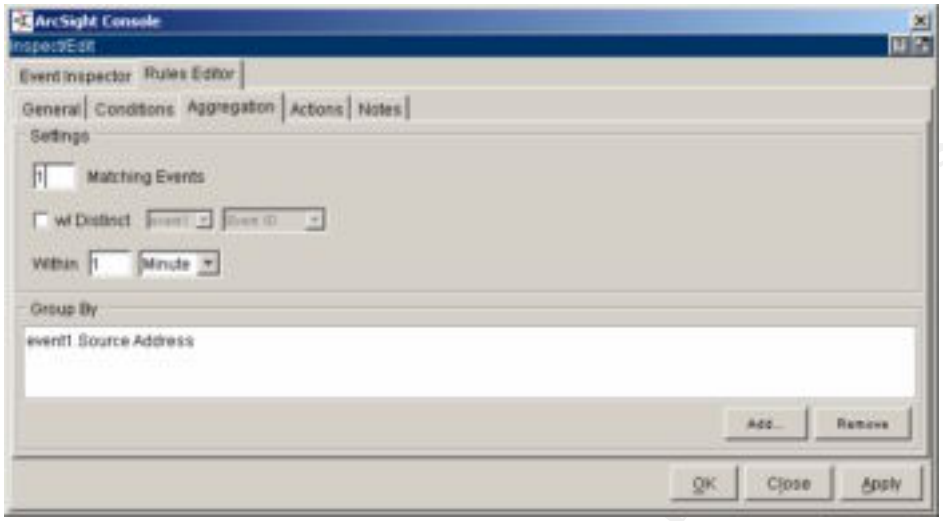
Snort Alert-*"NETBIOS nimda .eml"*
Propagated files
*.eml

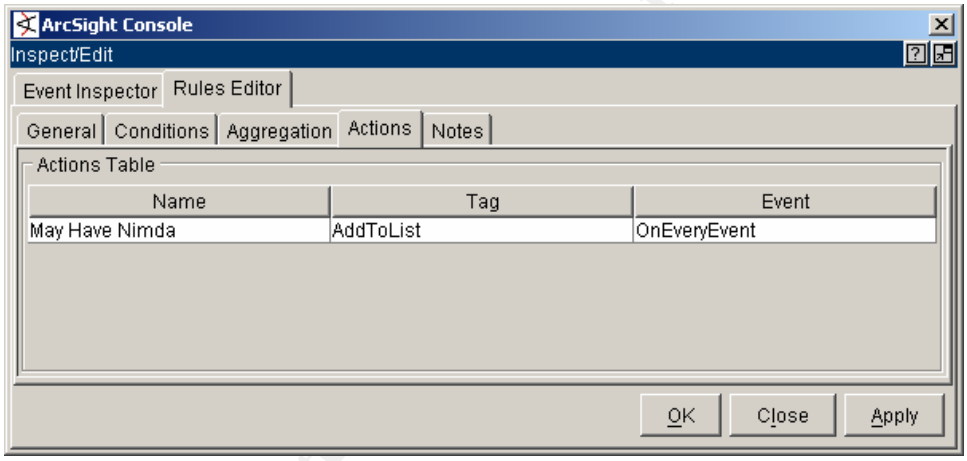
Snort Alert-*"NETBIOS nimda .nws"*
Propagated files
*.nws

Alone one of these events does not definitively indicate a compromised host, allowing for false positives. Seeing more than one Nimda related event within a configured time period, would be an issue worth the analyst looking into.
Putting the details we know about Nimda together. We can now build a rule. The first two alerts may indicate a host probing for the backdoor left by CodeRed version II, so these alerts may indicate CodeRed infection instead of Nimda. Because these are not Nimda specific we'll leave these out of our rule. After all we're trying to get only Nimda hosts.
The next pair may indicate the host is propagating files caused by the Virus. Our rule will add the source address of these events to an Active list (our list will contain this address for the next 48 hours) we'll call this list "Could be Nimda" here's the main pieces creating this rule:

- Event names for either WEB-FRONTPAGE /_vti_bin/ access or WEB-IIS multiple decode attempt
- On each event grouped by source address
- Added to list "Could be Nimda"

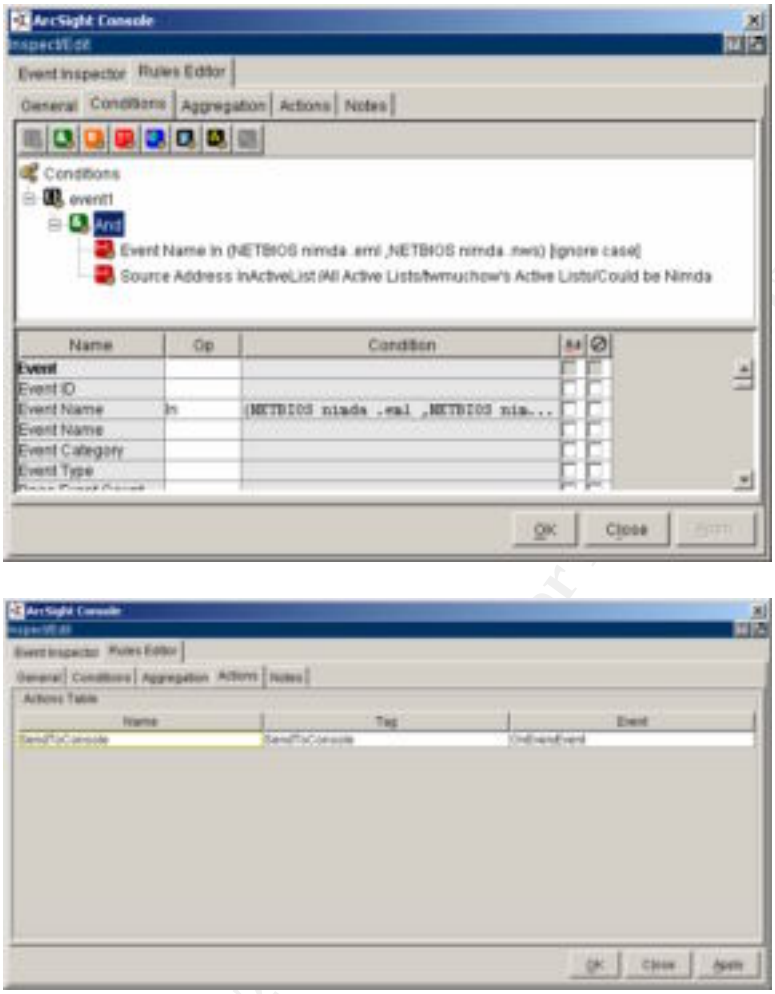






Nimda Rule part 1

Next a rule is built using the last pair of alerts that may indicate a compromised host sending files with either the .eml or .nws file extensions.



Nimda Rule part 2

This rule will become active anytime an event for either the .eml or .nws is created. ArcSight will compare the source address of the event with address' in the "May have Nimda" list if a match is found a correlated event named "Nimda Compromised Host" is forwarded to the Grid with the details.

Once a new rule is activated, every time the Arcsight correlation engine encounters the first piece of a rule statement, ArcSight will keep the event in memory, for the configured time window, looking for a match to the next piece of the rule. If it finds a match (or matches), the correlated rule is fired. Our rule forwards notification to the Grid in the form of an event (the type of notification is also configurable, options include: forwarding to console, paging, emailing operator or running a script). Correlated rules appear on the Grid with a lighting bolt added, allowing the analyst to distinguish incoming correlated events from single events. Events causing the match are also captured with the correlated event for easy viewing by the Intrusion analyst. Arcsight comes with a large collection of rules, preconfigured, awaiting activation. Use only the rules that apply to your network, after testing has proven their effectiveness, as Marcus J. Ranum, CTO, NFR security, Inc notes:

> *"As with every new technology, a certain amount of skepticism is essential- results that sound too good to be true probably are. System that boast amazing performance most likely only achieve those results under carefully constructed conditions"*[10]

Once the effectiveness of each rule is proven (monitor for the next week, if , as in our Nimda example, no valid infections are missed, your good to go), the analyst can then set filters eliminating all related single events from the active Grid. Only correlated events need the analyst's attention.

Use of rules for validation of events, as in our nimda rule example is only one way to utilize the rule feature to focus our efforts. Every network operates in a certain way. We'll refer to this as "normal". Determining what is normal for your network, can be accomplished by profiling its behavior, David J. Marchette suggests,

> *"Profiling the activity on the network is the act of collecting statistics that give a summary of the kinds of activities that are naturally occurring on the network. This gives a picture of the normal traffic on the network, which can be used to detect intrusions as deviations from this normal behavior"*[11]

Create your own network profile of normal by looking at your networks behavior for time intervals, such as hourly, daily or weekly. Averaging the data, noting spikes or changes in this data may lead you to create correlated rules that seek changes or anomalies in your normal traffic. Dipankar Dasgupta and Hal Brian relate that:

> "*Anomaly detection is performed by detecting changes in the patterns of utilization or behavior of the system. This type of intrusion detection is performed by building a statistical model that contains metrics derived from system operation and flagging as intrusive any observed metrics that have a significant statistical deviation from the model*"[12]

A word of caution about anomaly profiling comes from David Newman, Joel Snyder and Rodney Thayer:

> " *alarms from anomaly-based systems are only as useful as the baseline with which they're compared. An anomaly-based system might characterize a network already rife with attacks as "normal" and thus miss*

---

[10] Ranum, Marcus J. "Experiences Benchmarking Intrusion Detection Systems" Dec 2001. URL:http://www.snort.org/docs/Benchmarking-IDS-NFR.pdf (28 September 2002)

[11] Marchette, David J. Computer Intrusion Detection and Network Monitoring, Springer –Verlag New York, Inc, 2001. 109

[12] Dasgupta, Dipankar and Brian, Hal "Mobil Security Agents for Network Traffic Analysis" date unknown URL: http://issrl.cs.memphis.edu/DISCEX-II_Dasgupta.pdf (26 (September 2002)

future intrusions"[13]

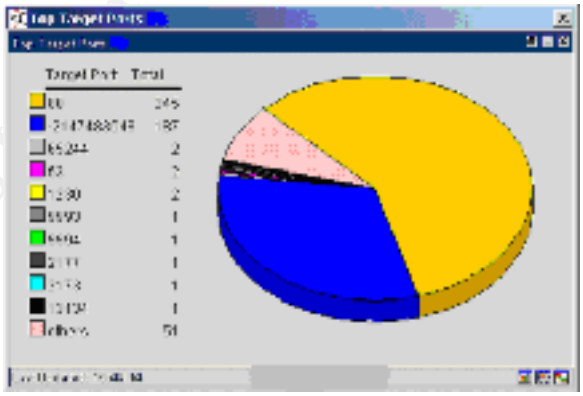
A simple example is to build a model based on the total number of events (or total web related events if that is your key business) logged by each sensor in use, daily or hourly peaks should be noted over a week's time (for our example). Then create a rule that forwards a correlated event to the console whenever these expected peaks are exceeded.
To help you determine what is anomalous behavior Arcsight includes reporting tools that allow for queries that retrieve historical data. Creating report queries uses the same logic used in filter and rules creation and can be customized to seek data from any time period desired. These reports present the data in Adobe Acrobats[14] PDF format.

Dashboards

The last feature we'll look at is the ArcSight Dashboards tool. Dashboards allow for near real time analysis of event data. Each dashboard is comprised of a data source, sampling choices and display options. Bar and Pie charts are the most utilized. Viewing the events causing the current display is as simple as double clicking the desired data. A dashboard filter is created and a replay is fed to the grid for review. With a sampling interval set down to once a minute or less, you can close the time gap to where an active response is still possible, such as blocking the intruders IP on your Firewall or shutting down the rogue modem he's sneaking in on. A display of the most active TCP ports for the last 10 minutes could display

- Top 10 ports being used
- Total count for the interval
- A spiffy pie chart

And might look like



[13] Newman, David - Snyder Joel and Thayer Rodney "Crying wolf: False alarms hide attacks" Network World Magazine 24 June 2002 URL: http://www.nwfusion.com/techinsider/2002/0624security1.html (25 September 2002)

[14] http://www.adobe.com/main.html

Dashboard View

As always, there's always a cost to setting the sampling too low, load on the CPU both on the ArcSight manager's server and console workstation increases as the sampling frequency increases. So the more powerful your systems are the more flexibility you'll have.

Putting it together

The job of ArcSight is to get usable information to the Intrusion analyst, in as timely a manner as possible, allowing for Intrusion Management. The huge volume of information that must be sorted through before valid events are identified must be reduced. Reducing useless data or false positives at the source by tuning to eliminate unnecessary events is the first step. Further reduction of the volume can be achieved using the ArcSight rules feature combined with event filtering. The examples provided are quite simple for demonstration purposes. ArcSight rules of very complicated combinations are possible, so complex rules characterizing intricate exploit features are possible. Utilizing a thorough understanding of exploits based on all its features and signatures to create ArcSight rules that capture validated events will increase the likelihood of an intrusion being detected and responded to. Great sources for information on the characteristics of exploits include the Snort signature database[15], CVE[16] and CERT[17]. Building rules that look for more than just a signature event, looking at each and every event generated by the source (Snort in our example) may not be necessary. This allows the Intrusion analyst to filter related single events concentrating on the correlated events that have increased probability of valid intrusion. Dashboards are the final piece. These provide visual display of the data being received by the ArcSight manager directly to the console and can alert the Intrusion analyst to changes or anomalies in the current data flow. Giving near real time data to the Intrusion analyst and the ability to review then act was our goal from the beginning.

Conclusions

Intrusion Detection is NOT the save all answer to protecting your network. As Peter Coffee recommends:

> ""*When driven by top-down commitment from enterprise management, the resulting culture of vigilance will leave fewer vulnerabilities to find, repair and monitor and will drastically reduce the costs that arise during and after successful attacks. Vigilance is the longest-term—but highest-*

[15] http://www.snort.org/snort-db/

[16] http://www.cve.mitre.org/cve/

[17] http://www.cert.org/nav/index_red.html

payback—element in an IT security strategy"[18]

Implementing strong security practices and, educating users on security procedures. Follow up with supporting tools like secure software applications and operating systems. Finally adding Firewalls and Intrusion detection systems and security personnel will still not stop all intrusions. ArcSight, with well implemented rules and dashboards may give your intrusion analyst a chance at finding the intruder buried in your data fast enough for Intruder management instead of crisis management.

List of References

[1] Stephenson, P. (2000) "Intrusion Management: A top Level Model for Securing Information Assets in an Enterprise Environment" EICAR 2000 Best Paper Proceedings, 288 URL: http://www.conference.eicar.org/past_conferences/2000/papers/Tuesday/Security%20Trust%20and%20E-commerce/other/Stephenson.pdf 26 September 2000

[2]Created by Martin Roesch, and available @ http://www.snort.org/

[3] http://www.sourcefire.com/

[4] Brox, Arnt. Signature Based or Anomaly Based Intrusion Detection – The Practice and Pitfalls, 2 February 2002. URL: http://www.itsecurity.com/papers/proseq1.htm (30 September 2002)

[5] Snort users manual available @ http://www.snort.org/docs/writing_rules/

[6]Snort signature database @ http://www.snort.org/snort-db/

[7] Common Vulnerabilities and Exploits database @http://www.cve.mitre.org/

[8] **http://www.cert.org/advisories/CA-2001-26.html**

[9] Blackman, David, Intrusion Detection is failing: Enter Intrusion Management. 29 July 2002 URL: http://www.itsecurity.com/papers/pentasafe1.htm (28 September 2002)

---

[18] Coffee, Peter "5 Steps to Enterprise security, Step 5:Vigilance" 10 December 2001 URL: http://www.eweek.com/article2/0,3959,6698,00.asp (20 September 2002)

[10] Ranum, Marcus J. "Experiences Benchmarking Intrusion Detection Systems" Dec 2001. URL:http://www.snort.org/docs/Benchmarking-IDS-NFR.pdf (28 September 2002)
[11] Marchette, David J. Computer Intrusion Detection and Network Monitoring, Springer –Verlag New York, Inc, 2001. 109

[12] Dasgupta, Dipankar and Brian, Hal "Mobil Security Agents for Network Traffic Analysis" date unknown URL: http://issrl.cs.memphis.edu/DISCEX-II_Dasgupta.pdf (26 (September 2002)

[13] http://www.adobe.com/main.html

[14] http://www.snort.org/snort-db/

[15] http://www.cve.mitre.org/cve/

[16[http://www.cert.org/nav/index_red.html

[17] Coffee, Peter "5 Steps to Enterprise security, Step 5:Vigilance" 10 December 2001 URL: http://www.eweek.com/article2/0,3959,6698,00.asp (20 September 2002)