Chapter 5

Managing Snort Alerts

Scripts and Samples in this Chapter

- Gathering Snort Logs
- Building an Alerts Detail Report
- Building an Alerts Overview Report
- Managing Snort Rules

In This Toolbox

Snort is an open source intrusion detection system (IDS) used in a wide variety of network environments. Snort is a lightweight, but extremely powerful tool for detecting malicious traffic on your network. With a flexible and robust rules definition language, Snort is capable of detecting nearly any threat that crosses your network.

However, if you use Snort, you already know that reporting is not its strength. On a busy network you might find that it records tens or hundreds of thousands of suspicious events every day. Fortunately, Log Parser is a perfect match for Snort for managing intrusion detection logs.

Building Snort IDS Reports

An intrusion detection system is only valuable if you review and act on the data it produces. Unfortunately, sometimes an IDS will produce overwhelming amounts of data that make it difficult to process. To aid in our interpretation of the data, we can use Log Parser to take snapshots of our IDS logs and present them in different easy-to-read reports. In this chapter, we will build an example IDS report using nothing more than the power of Log Parser.

Gathering Snort Logs

To process the alert data, we first need a consistent method for gathering the data. Log Parser is an excellent method for managing Snort logs because you can query the file while Snort still has the log open. Many administrators schedule scripts to regularly cycle the Snort logs, but this requires stopping the service to release the file so a script can move it. Using Log Parser, we can use checkpoints to read the most recent data from the file.

Although Snort supports several output formats that Log Parser could use, I have found the CSV format most flexible and consistent. To configure Snort to use the CSV output format, simply add the following line in the snort.conf file:

output alert_csv: alert.csv default

This configures Snort to create a CSV log file named alert.csv in the configured logs directory using the default output fields. By default the CSV output processor includes these fields:

- timestamp
- sig_generator
- sig_id
- sig_rev
- msg
- proto
- src

- srcport
- dst
- dstport
- ethsrc
- ethdst
- ethlen
- tcpfags

- tcpseq
- tcpack
- tcplen
- tcpwindow
- ttl
- tos
- id

- dgmlen
- iplen
- icmptype
- icmpcode
- icmpid
- icmpseq

Snort CSV logs do not include a header row, so we need a separate file to name each column. In the file download for this chapter, I have included the file AlertHeader.csv to use for this purpose. To read CSV Snort alerts, you would use a command like this:

logparser.exe file:alert.sql -i:csv -headerRow:off -iHeaderFile:AlertHeader.csv iTsFormat:mm/dd/yy-hh:mm:ss

Note that we specify the CSV input format, but instead of using the header row, we specify a header file using the **iHeaderFile** option. We also specify the timestamp format so Log Parser can interpret that field as an actual time stamp rather than a string.

Νοτε

All of the queries in this chapter use the year in the timestamp date, which Snort does not log by default. To configure Snort to log the year, use the –y option when starting Snort. If your timestamps do not include the year, the queries in this chapter will return the error, "Semantic Error: argument of function TO_TIME must be a TIMESTAMP."

Building an Alerts Detail Report

In our IDS report we likely want to view summaries of the alert data such as:

- Most common alerts
- Most common source IP (Internet Protocol) addresses
- Most common target IP addresses

Using Log Parser's multiplex feature and template output format we can very easily create interactive HTML (Hypertext Markup Language) reports directly from the Snort logs.

Most Common Alerts

To begin our report, we will create HTML pages for the most common alerts. We will start with an index page showing the most common alert messages. Each line will have a link to a more detailed HTML page listing individual alerts with that message. The query for the index page is simple:

```
---Ch05Alerts-Index.sql---
SELECT DISTINCT
   sig id,
   msg,
   Count(msg) as Alerts
INTO report\alerts.html
FROM alert.csv
GROUP BY msg, sig_id
ORDER BY Alerts DESC
---Ch05Alerts-Index.sql---
   The key component here is the so much the query but the output template:
---Ch05Alerts-Index.tpl---
<LPHEADER>
   <html>
   <head>
     <meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
     <link rel="stylesheet" type="text/css" href="snort.css">
     <title>Snort Alert Messages</title>
  </head>
  <body>
  <h1>Snort Alerts Summary</h1><br/>
  <i>Created %SYSTEM TIMESTAMP% </i>
        <b>Signature</b>
           <b>Message</b>
           <b>Alerts</b>
        </t.r>
</LPHEADER>
<LPBODY>
        <a href=http://www.snort.org/snort-
db/sid.html?sid=%sig_id%> %sig_id%</a>
            %msg%
           <a href=alert\%sig_id%.html>&nbsp;%Alerts%</a>
        </LPBODY>
<LPFOOTER>
```

```
</body>
</html>
</LPFOOTER>
---Ch05Alerts-Index.tpl---
```

You can run the query using the output template using this command:

```
logparser.exe file:Ch05Alerts-Index.sql -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:tpl -tpl:Ch05Alerts-Index.tpl
```

Run this command and in a matter of seconds you should have a file named alerts.html that looks like the one shown in Figure 5.1. Note that the report lists the alerts in order, starting with the most common messages. If you click on the signature ID, it will jump to the reference page at www.snort.org. Note that the alert total is also a hyperlink, but we have not created that page yet. We now need to run another query to generate log details for each alert message.

Figure 5.1 Snort Alert Messages Summary

Snort A	lert Messages - Microsoft Internet Explorer			
ile <u>E</u> dit	<u>V</u> iew F <u>a</u> vorites <u>T</u> ools <u>H</u> elp			1
Back	- 🕥 - 🖹 💈 🏠 🔎 Search 👷 Favorites 🧐 🙆 - چ 🖸	•		
dress	alerts.html		✓ → Go	Links »
Snort	2 Alerts Summary			
Signatur	e Message	Alerts		
1917	SCAN UPnP service discover attempt	3330		
972	WEB-IIS %2E-asp access	1206		
1054	WEB-MISC weblogic/tomcat .jsp view source attempt	576		
1245	WEB-IIS ISAPI .idq access	288		
1244	WEB-IIS ISAPI .idg attempt	288		
538	NETBIOS SMB IPC\$ share unicode access	216		
839	WEB-CGI finger access	180		
466	ICMP L3retriever Ping	180		
15	(http_inspect) OVERSIZE REQUEST-URI DIRECTORY	108		
853	WEB-CGI wrap access	72		
2382	NETBIOS SMB DCERPC NTLMSSP asn1 overflow attempt	72		
1201	ATTACK-RESPONSES 403 Forbidden	36		
1201 2404	ATTACK-RESPONSES 403 Forbidden NETBIOS SMB-DS Session Setup AndX request unicode username overflow attempt	36 36		
1201 2404 882	ATTACK-RESPONSES 403 Forbidden NETBIOS SMB-DS Session Setup AndX request unicode username overflow attempt WEB-CGI calendar access	36 36 36		
1201 2404 882 2	ATTACK-RESPONSES 403 Forbidden NETBIOS SMB-DS Session Setup AndX request unicode username overflow attempt WEB-CGI calendar access (http_inspect) DOUBLE DECODING ATTACK	36 36 36 18		

Creating a detail page for every individual message is surprisingly simple:

```
---Ch05Alerts-Detail.sql---
SELECT
sig_id,
TO_TIME(timestamp) AS Time,
msg,
proto,
src,
srcport,
```

dst, dstport, ethsrc, ethdst, ethlen, tcpflags, tcpseq, tcpack, tcplen, tcpwindow, ttl, tos, id, dgmlen, iplen, icmptype, icmpcode, icmpid, icmpseq INTO report\alert*.html FROM alert.csv ---Ch05Alerts-Detail.sgl---

This query takes advantage of Log Parser's multiplex feature and creates a unique output file for each unique value of **sig_id**, the first field in the query. It uses this value in place of the asterisk (*) in the filename specified on the INTO clause. Since we use the signature ID in the output filename, it is easy for us to link to those files from the main alert.html page.

For the alert detail pages, I wanted use a title at the top of the page showing the particular alert message, as shown in Figure 5.2. However, you cannot use field placeholders in the LPHEADER portion of the template file. For example, if I placed **%msg%** in the LPHEADER, each page would simply have %msg% as the title.

Figure 5.2 Detailed Alert Messages.

🗿 WEB-IIS 🤋	62E-as	o access -	Microsof	ft Internet Explo	rer											
Ele Edit (<u>V</u> iew F	<u>a</u> vorites <u>T</u>	ools <u>H</u> el	p												RU.
③ Back • ③ · 🖹 🗿 🟠 🖉 Search 👷 Favorites 🛷 🙆 • 🧕																
Address 🖉 🕻	:\Progra	m Files\Snort	bin log \Re	eport\report\alert\97	2.html								✓ →	Go Li	nks »	Ð.
Alert: WEB-IIS %2E-asp access Created 2004-11-18 11:49:16																
Back to aler	ts inde>	(-						a.
timestamp	proto	src	srcport	dst	dstport	ethsrc	ethdst	ethlen	tcpflags	tcpseq	tcpack	tcplen t	cpwindow	tti to	3 ID	d
15:13:07	TCP	10.8.0.72	1913	3.1/9.100.233	80	0:90:48:21:36:16	0:9:58:39:87:14	0x18B	***AP***	UX414ABAED	0xB814AFD5		0x40B0	128 0	55238	1.2
15:18:13	TOP	10.8.0.72	1978	3.179.100.233	80	0:90:48:27:36:76	0:9:58:39:87:F4	UXIC6	AP	UX6E/D8A9A	UXF593EF97		0X40B0	128 0	5/902	
15:21:46	TCP	10.8.0.72	1997	3.179.100.233	80	0:90:48:21:36:16	0:9:58:39:87:F4	Ow1E2	888AD888	0x9132E162	OWE4ER10R2		0x40B0	128 0	61/13	1
15:23:17	TCP	10.0.0.72	2034	3.179.100.233	00	0:90:48:27:36:76	0.9:58:39:87:F4	Ow1E3	888AD888	OXIIDIIE66	0xE4FB10B3		0x4080	120 0	65161	1
15:23:45	TCP	10.0.0.72	2040	3.179.100.233	90	0:90:48:21:36:16	0:9:58:39:87:F4	0x1E4	888AD888	0x4FAE205B	0xD450269E		0x4080	128 0	65170	1
15:25:40	TCP	10.0.0.72	2040	249 22 121 140	90	0:90:48:21:36:16	0:9:58:39:87:F4	0v2BE	888AD888	0x4FAE203B	0x0430269E		0x4080	128 0	501	1
15-25-11	TCP	10.0.0.72	2049	249.22.121.140	90	0.00.48.25.36.56	0.9.58.39.87.64	0x254	***AD***	0x00425746	0x280514CA		0x4080	120 0	642	2
15-25-12	TCP	10.0.0.72	2040	249.22.121.140	90	0.00.48.25.36.56	0.9.58.39.87.64	0×352	***AD***	0x304237A0	0x200019649		0x4080	120 0	692	2
15-25-15	TCP	10.8.0.72	2049	249.22.121.140	80	0:90:48:25:36:56	0.9:58:39:87:F4	0x2FB	***AD***	0xEDC4ECA6	0x47111421		0x4080	128 0	886	
15:25:17	TCP	10 8 0 72	2047	249 22 121 140	80	0:90:48:25:36:56	0-9-58-39-87-F4	0x358	***AP***	0xEDC4EE6B	0x47117C45		0x40B0	128 0	1017	ŝ
15-25-18	TCP	10 8 0 72	2051	249 22 121 140	80	0:90:48:25:36:56	0-9-58-39-87-F4	0x358	***AD***	0x41C0C403	0xC8E20EE8		0x40B0	128 0	1047	\$
15:25:18	TCP	10.8.0.72	2052	249.22.121.140	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x3E8	***AP***	0x82B86D66	0xD42E0688		0x40B0	128 0	1063	1
15:25:18	TCP	10.8.0.72	2053	249.22.121.140	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x3E8	***AP***	0xB67A47C7	0x76C7ECC4		0x40B0	128 0	1068	1
15:25:18	TCP	10.8.0.72	2054	249.22.121.140	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x3E8	***AP***	0x5C7E7E2A	0x7DE68E72		0x40B0	128 0	1082	1
15:25:19	TCP	10.8.0.72	2055	249.22.121.140	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x3F8	***AP***	0x2D66108E	0x5C73DD08		0x40B0	128 0	1097	1
15:25:21	TCP	10.8.0.72	2056	249.22.121.140	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x35B	***AP***	0x67962C8C	0x6D2533EA		0x40B0	128 0	1174	ε
15:25:21	TCP	10.8.0.72	2052	249.22.121.140	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x3F8	***AP***	0x82B86D66	0xD42E0688		0x40B0	128 0	1188	1
15:25:21	TCP	10.8.0.72	2057	3.179.5.21	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x2B4	***AP***	0xAF4CF59E	0xAAA7F2AB		0x40B0	128 0	1215	E
15:25:21	TCP	10.8.0.72	2058	249.22.121.140	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x25A	***AP***	0x93B6B9A9	0x6F5BCDC4		0x40B0	128 0	1220	5
15:25:22	TCP	10.8.0.72	2059	249.22.121.140	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x259	***AP***	0xD6494891	0x6B9A84CF		0x40B0	128 0	1251	5
15:25:22	TCP	10.8.0.72	2060	249.22.121.140	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x24D	***AP***	0xBC86F2FC	0xD3B123A6		0x40B0	128 0	1283	5
15:25:25	TCP	10.8.0.72	2060	249.22.121.140	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x24D	***AP***	0xBC86F2FC	0xD3B123A6		0x40B0	128 0	1399	5
15:25:26	TCP	10.8.0.72	2061	3.179.5.21	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x369	***AP***	0xE6640008	0x5C354142		0x40B0	128 0	1430	8
<	****	** * * **		· · · · · · · · · · · · · · · · · · ·	~~			· ····	*******							>
🙆 Done													Зм	y Compu	ter	

Nevertheless, sometimes a little creativity can make up for Log Parser's limitations. To write the individual titles, I actually run the query twice, once to write the headers and again to write the detail information. For the first pass I use this query and template file:

```
---Ch05Alerts-DetailHeader.sql---
SELECT DISTINCT
    sig_id,
    msg
INTO report\alert\*.html
FROM alert.csv
---Ch05Alerts-DetailHeader.sgl---
---Ch05Alerts-DetailHeader.tlp---
<LPBODY>
       <html>
       <head>
        <meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
       <link rel="stylesheet" type="text/css" href="..\snort.css">
       <title>%msg%</title>
       </head>
       <body>
       <b><font face="Arial" size="3">Alert: %msg%</font></b><br/>>
        <i>Created %SYSTEM_TIMESTAMP% </i>
        <a href=..\alerts.html>Back to alerts index</a>
</LPBODY>
```

```
---Ch05Alerts-DetailHeader.tlp---
```

Run the query with this command:

```
logparser.exe file:Ch05Alerts-DetailHeader.sql -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:tpl -tpl:Ch05Alerts-DetailHeader.tpl
```

The query first pulls the **sig_id** and **msg** and creates a unique file for each **sig_id** and writes a header using **msg** in the LPBODY. At this point we have a set of files with nothing but a header. Next, we run another set of queries using Ch05Alerts-Detail.sql and this template:

```
---Ch05Alerts-Detail.tpl---
<LPHEADER>
              <b>timestamp</b>
                   <b>proto</b>
                   <b>src</b>
                   <b>srcport</b>
                   <b>dst</b>
                   <b>dstport</b>
                   <b>ethsrc</b>
                   <b>ethdst</b>
                   <b>ethlen</b>
                   <b>tcpflags</b>
                   <b>tcpseq</b>
                   <b>tcpack</b>
                   <b>tcplen</b>
                   <b>tcpwindow</b>
                   <b>ttl</b>
                   <b>tos</b>
                   <b>id</b>
                   <b>dgmlen</b>
                   <b>iplen</b>
                   <b>icmptype</b>
                   <b>icmpcode</b>
                   <b>icmpid</b>
                   <b>icmpseq</b>
              </LPHEADER>
<LPBODY>
                %time%
                     %proto%
                    <a href=..\src\%src%.html>%src%</a>
                    %srcport%
                    <a href=..\dst\%dst%.html>%dst%</a>
```

```
  %dstport%
                   %ethsrc%
                   %ethdst%
                  %ethlen%
                   %tcpflags%
                   %tcpseq%
                   %tcpack%
                   %tcplen%
                   %tcpwindow%
                   %ttl%
                   %tos%
                  %id%
                  %dgmlen%
                  %iplen%
                  %icmptype%
                  %icmpcode%
                  %icmpid%
                  %icmpseq%
                 </t.r>
</LPBODY>
<LPFOOTER>
        </body>
    </html>
</LPFOOTER>
---Ch05Alerts-Detail.tp---
```

This time when we run the query, we use this command:

```
logparser.exe file:Ch05Alerts-Detail.sql -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:tpl -tpl:Ch05Alerts-Detail.tpl -fileMode:0
```

Note that I used **-fileMode:0** in the command to instruct Log Parser to append to the files rather than overwrite them. In this pass, Log Parser will take the files already containing titles and append the alert details as shown earlier in Figure 5.2. The two-pass approach will obviously slow down report creation time with very large alert files, but it is still surprisingly effective for most purposes.

Τιρ

If you find that the two-pass approach is too slow, another option is to use Log Parser as a COM component in a script and process each output record individually.

Alerts by IP Address

Each IP address in the alerts report shown in Figure 5.2 is a clickable hyperlink that leads to a detail page showing all alerts for that IP address. Using a process similar to that used previously for the alert messages, I created a summary page (Figure 5.3), and detail pages (Figure 5.4) using a two-pass approach. I repeated this process for both source and destination IP addresses to produce a fully interactive HTML IDS report. At this point, you can run the entire report with these Log Parser commands:

```
logparser.exe file:Ch05Alerts-Index.sgl -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:tpl -tpl:Ch05Alerts-Index.tpl
logparser.exe file:Ch05Alerts-DetailHeader.sql -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:tpl -tpl:Ch05Alerts-DetailHeader.tpl
logparser.exe file:Ch05Alerts-Detail.sql -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:tpl -tpl:Ch05Alerts-Detail.tpl -fileMode:0
logparser.exe file:Ch05SrcIP-Index.sgl -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:tpl -tpl:Ch05SrcIP-Index.tpl
logparser.exe file:Ch05SrcIP-DetailHeader.sql -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:tpl -tpl:Ch05SrcIP-DetailHeader.tpl
logparser.exe file:Ch05SrcIP-Detail.sql -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:tpl -tpl:Ch05SrcIP-Detail.tpl -fileMode:0
logparser.exe file:Ch05DstIP-Index.sql -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:tpl -tpl:Ch05DstIP-Index.tpl
logparser.exe file:Ch05DstIP-DetailHeader.sgl -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:tpl -tpl:Ch05DstIP-DetailHeader.tpl
logparser.exe file:Ch05DstIP-Detail.sql -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:tpl -tpl:Ch05DstIP-Detail.tpl -fileMode:0
```

Figure 5.3 Snort Alerts by Destination IP Address

Snort Alerts - Microsoft Internet Explorer		
Eile Edit View Favorites Tools Help		AU
🔇 Back 🔹 🕥 · 🖹 💈 🏠 🔎 Search 🤺 Favorites 🤗	8· 🎍 🖬 🗸 💭	
Address 🖉 C: \Program Files \Snort\bin \log \Report\veport\dst.html		🕶 🛃 Go 🛛 Links 🎽 📆 🕶
Snort Alerts by Destination IP Address Created 2004-11-18 12:19:29		
Destination IP Address	Alerts	
239.255.255.250	3330	
249.22.121.140	828	
63.241.72.111	378	
209.239.57.147	288	
69.20.62.196	216	
192.168.70.201	216	
60.154.80.250	198	
3.179.100.233	180	
164.82.201.36	180	
192.168.70.17	162	
209.73.83.85	72	
209.239.57.99	72	
66.135.208.226	54	
192.168.70.49	54	
192.168.10.252	54	
10.37.16.51	36	
209.20.231.199	36	
3.179.5.21	36	
209.52.17.116	36	
192.168.70.238	36	
69.20.118.37	36	
66.135.208.101	18	
66.135.192.88	18	
66.135.202.140	18	
63.215.198.192	18	
33.113.198.252	18	
e		😼 My Computer

www.syngress.com

2.111 - Microsoft Internet Explorer											
jew Favorites Tools Help											1
🕥 · 🖹 🖉 🏠 🔎 Search 🔆 Favorites	Ø	8.3	0.								
:\Program Files\Snort\bin\jog\Report\report\dst\63.241.72.111.h	.tml								✓ ⇒	Go Links »	Đ
Alerts for destination IP: 63.241.72.111 Created 2004-11-18 12:19:31											
nation IP index											
msg	proto	src	srcport	dstport	ethsrc	ethdst	ethien	tcpflags	tcpseq	tcpack	tc
WEB-IIS %2E-asp access	TCP	10.8.0.72	2227	80	0:90:4B:2F:36:F6	0:9:5B:39:87:F4	0x2B4	AP+++	0xF7B18DBE	0x5BBA3D96	
WEB-IIS %2E-asp access	TCP	10.8.0.72	2231	80	0:90:4B:2F:36:F6	0:9:5B:39:B/:F4	0x369	***AP***	0xC4296033	0x7F659DAA	
WEB-IIS %2E-asp access	TCP	10.8.0.72	2354	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x2B4	***AP***	0xDF706403	0xF971E60B	
WEB-IIS %2E-asp access	TCP	10.8.0.72	2358	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x369	***AP***	0x38A6EF07	0x6608227	
WEB-IIS %2E-asp access	TCP	10.8.0./2	2437	80	0:90:4B:2F:36:F6	0:9:5B:39:8/:F4	0x2A4	***AP***	0x9E4B8ECB	0x330305DE	
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3898	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x28A	***AP***	0xEESDE77F	0xASE4DA0B	1
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3901	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x309	***AP***	0x44E32019	0xC7127C3D	4
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3903	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x2EC	***AP***	0xA163ECE1	0x5461464F	
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3904	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x39F	***AP***	0x14210772	0x748F2DDF	
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3909	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x353	***AP***	0xED2D135B	0xAEBE9961	
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3913	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x3C3	***AP***	0x87E0DAD6	0x71E72A91	
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3920	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x325	***AP***	0xF56B40CB	0xE2BC37	
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3921	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x3EA	***AP***	0x91FC7438	0x8166FF10	
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3927	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x33D	***AP***	0x507031F1	0xB197F470	
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3928	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x40A	***AP***	0xA919407F	0xC7807A43	
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3931	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x359	***AP***	0x5187349D	0xBB886268	
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3934	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x386	***AP***	0x30E0C6C7	0x557FF7F3	
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3935	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x443	***AP***	0xE6D3AE82	0xBAD62746	1
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3936	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x35B	***AP***	0x1B6C48D	0xA061AC02	
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3938	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x59A	***A****	0x664E63AA	0x3E1C122	
WEB-MISC weblogic/tomcat .jsp view source attempt	TCP	10.8.0.72	3938	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x59A	***A****	0x664E63AA	0x3E1C122	
WEB-IIS %2E-asp access	TCP	10.8.0.72	2227	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x2B4	***AP***	0xF7B18DBE	0x5BBA3D96	ù
WEB-IIS %2E-asp access	TCP	10.8.0.72	2231	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x369	***AP***	0xC4296033	0x7F659DAA	
WEB-IIS % 2E-aco access	TCP	10.8.0.72	2354	80	0:90:4B:2F:36:F6	0:9:5B:39:B7:F4	0x2B4	***AP***	0xDF706403	0xF971E60B	
WED-113 /02E-03P 000033											
WEB-IIS %2E-asp access	TCP	10 8 0 72	2358	80	0-90-48-2F-36-F6	0-9-58-39-87-F4	0×369	***7D***	0v3846EE07	0v6608227	
	Program FlexShortban Usp (Program Isp (Program FlexShortban Usp (Program FlexShortban Usp (Program FlexShortban Usp (Program Isp (P	Program Files (Snort/bin (log) Report Yeard (State), 2:41.72.111.html Program Files (Snort/bin (log) Report Yeard (State), 2:41.72.111.html destination IP: 63.241.72.111 Hi1-18 12:19:31 inston IP: Index Test (Snort/bin (log) Report Yeard (State), 2:41.72.111.html destination IP: 63.241.72.111 Hi1-18 12:19:31 inston IP: Index Test Test (Snort/bin (log) Report Yeard (State), 2:41.72.111.html destination IP: 63.241.72.111 Hi1-18 12:19:31 inston IP: Index Test Test	Program Files (Snort/bin (log) Report Veport/Str(53.241.72.111.htm) Program Files (Snort/bin (log) Report Veport/Str(53.241.72.111.htm) destination IP: 63.241.72.111 Harting Strain	Program Files/Snort/bin/log/Report/teport/dat/63.241.72.111.html Program Files/Snort/bin/log/Report/teport/dat/63.241.72.111.html destination IP: 63.241.72.111 f-11-81 22:19:31 inston IP: index Program Files/Snort/bin/log/Report/teport/dat/63.241.72.111.html destination IP: 63.241.72.111 f-11-81 22:19:31 inston IP: index WEB-115 %2E-asp access TCP 10.8.0.72 2237 WEB-115 %2E-asp access TCP 10.8.0.72 2354 WEB-115 %2E-asp access TCP 10.8.0.72 2354 WEB-115 %2E-asp access TCP 10.8.0.72 3961 WEB-115 %2E-asp access TCP 10.8.0.72 3963 WEB-115 %2E-asp access TCP 10.8.0.72 3973 WEB-115 %2E-asp access TCP 10.8.0.72 3973 WEB-115 %2E-asp access TCP 10.8.0.72 3933 WEB-115 %2E-asp access	Formation Topologic Program Files (Snort/bin log) Report Yeport (St 63.241.72.111.htm) Program Files (Snort/bin log) Report Yeport (St 63.241.72.111.htm) destination IP: 63.241.72.111 F11-18 12:19:31 instance Program Files (Snort/bin log) Report Yeport (St 63.241.72.111.htm) destination IP: 63.241.72.111 F11-18 12:19:31 instance Program Files (Snort/bin log) Report Yeport (St 63.241.72.111.htm) destination IP: 63.241.72.111 F11-18 12:19:31 instance PED-115 %26:-sap access TCP 10.8.0.72 2344 WEB-115 %26:-sap access TCP 10.8.0.72 2344 WEB-115 %26:-sap access TCP 10.8.0.72 3981 WEB-115 %26:-sap access TCP 10.8.0.72 3981 WEB-MISC weblogic/tomeat.jsp view source attempt TCP 10.8.0.72 3981 WEB-MISC weblogic/tomeat.jsp view source attempt TCP 10.8.0.72 3991 WEB-MISC weblogic/tomeat.jsp view source attempt TCP 10.8.0.72 3921 WEB-MISC weblogic/tomeat.jsp view source attempt TCP 10.8.0.72 3921 WEB-MISC weblogic/tomeat.jsp view so	Revines Dots Left Program FlexSport/bm/log/Report/report/dst/83.241.72.111.html Image: Constraint of the system of	Favorities Tools Life Program FlexSport/tem/log/Report/teport/stat/s3.241.72.111.html Image: Sport/tem/log/Report/teport/stat/s3.241.72.111.html destination IP: 63.241.72.111 Fragmen FlexSport/tem/log/Report/teport/stat/s3.241.72.111.html WEB-115 %2E-sep access TCP 10.8.0.72 2231 80 0.990482.F3.561F6 019583997.F4 WEB-115 %2E-sep access TCP 10.8.0.72 2338 80 0.990482.F3.561F6 019583997.F4 WEB-115 %2E-sep access TCP 10.8.0.72 2381 80 0.990482.F3.561F6 019583997.F4 WEB-115 %2E-sep access TCP 10.8.0.72 2381 80 0.990482.F3.561F6 019583997.F4 WEB-MISC weblog/tomat.jsp view source attempt TCP 10.8.0.72 2381 80 0.990482.F3.561F6 019583997.F4 WEB-MISC weblog/tomat.jsp view source attempt TCP	Text Favorites Total Text Program FlexStoroTbin (log)Report Yeport(skip3.241.72.111.html) Image: Control (log)Report Yeport(skip3.241.72.111.html) Image: Control (log)Report Yeport(skip3.241.72.111.html) destination IP: 63.241.72.111 Fridge m FlexStoroTbin (log)Report Yeport(skip3.241.72.111.html) Image: Control (log)Report Yeport(skip3.241.72.111.html) destination IP: 63.241.72.111 Fridge m FlexStoroTbin (log)Report Yeport(skip3.241.72.111.html) Image: Control (log)Report Yeport(skip3.241.72.111.html) WEB-115 %26E-sep access TCP 10.8.0.72 2227 00 0.99948.27.364F6 0.99548.27.364F6 0.99548.27.364F6 <t< td=""><td>Texa Provides Search Program Files (Snort/bin (sop Report Veport) 54:53.241.72.111.html destination IP: 63.241.72.111 Files (Snort/bin (sop Report Veport) 54:53.241.72.111.html Image: Snort/bin (sop Report) 74:53.241.72.111.html destination IP: 63.241.72.111 Files (Snort/bin (sop Report) 74:53.241.72.111.html Image: Snort/bin (sop Report) 74:53.545.545.545.545.545.545.545.545.545.5</td><td>Terret Terret Terret</td><td>Term Term <t< td=""></t<></td></t<>	Texa Provides Search Program Files (Snort/bin (sop Report Veport) 54:53.241.72.111.html destination IP: 63.241.72.111 Files (Snort/bin (sop Report Veport) 54:53.241.72.111.html Image: Snort/bin (sop Report) 74:53.241.72.111.html destination IP: 63.241.72.111 Files (Snort/bin (sop Report) 74:53.241.72.111.html Image: Snort/bin (sop Report) 74:53.545.545.545.545.545.545.545.545.545.5	Terret Terret	Term Term <t< td=""></t<>

Figure 5.4 IP Address Details

Building an Alerts Overview Report

Now that we have a detailed alerts report, we might want to build a summary index page. This page should include links to the detailed reports and also display graphs and short summaries of the data to get a quick overview of the network. Figure 5.5 shows the final report.

Figure 5.5 Snort Alerts Summary

🗿 Snort Alerts S	ummary - Mic	rosoft Internet Ex	plorer							
Eile Edit View	Favorites To	ols <u>H</u> elp								R.
🕒 Back * 🕑	· × 2	Search	🖈 Favorites 🧔		-					
Address C: Prog	ram Files\Snort\b	oin Vog Report (report (r	ndex.html					*	→ Go Link	s " 🐑 -
									_	~
Snort A	erts Su	mmarv								
Alerts by Message	Top Alerts									
Alerts by	Signature	Message			Alerts		- UT			
Source IP	1917	SCAN UPnP servi	ce discover attempt		3330				1917	
	972	WEB-IIS %2E-as	p access		1206				9 72	
Alerts by Destination IP	1054	WEB-MISC weblo	gic/tomcat .jsp viev	source attempt	576				1054	
b estimation xi	1245	WEB-IIS ISAPI .	dq access		288			}	1245	
	1244	WEB-IIS ISAPI .	dq attempt		288				1244	
	839	WEB-CGI finger	C\$ share unicode at	cess	180			1	I 538	
	466	ICMP L3retriever	Ping		180				8 39	
	15	(http_inspect) O\	ERSIZE REQUEST-L	RI DIRECTORY	108				466	
	853	WEB-CGI wrap a	ccess		72					
Links	Top Source	e IP Addresses		Alerts Per Hour			Top Destinati	on Ports		
www.snort.org			III 192 168 70 219	750			445 - TCP			
Search Ports		W	10.8.0.72	500			- ICMP			
			102 168 70 17	500-		1	120 TCD			
			E 102.168.70.201	250			135-104			
			102 168 70 247	mm	mmm	MULIA	80 - TCP		_	
			■ 192.100.70.247 ■ 402.468.70.22	0			1900 - UDP			
			III 192.168.70.33	-250			0	1000 2000	3000	4000
				n			L			
81.								6	1	
E Doue									3 my compute	

www.syngress.com

Since the summary report includes only one table of data with multiple graphs, I placed the entire HTML markup in a single template file, Ch05Summary-Index.tpl. This template creates the HTML for the report that includes references to the individual graphs. The query for the top alerts is similar to that used for the alerts index page, but only includes the top 10 records:

```
---Ch05Summary-Index.sql---
SELECT TOP 10
sig_id,
msg,
Count(msg) as Alerts
INTO report\index.html
FROM alert.csv
GROUP BY msg, sig_id
ORDER BY Alerts DESC
---Ch05Summary-Index.sgl---
```

The query for the pie graph is similar, but does not include the actual message and this time processes all records:

```
---Ch05Summary-GraphTopAlerts.sql---
SELECT
sig_id,
Count(msg) as Alerts
INTO report\AlertsTopAlerts.gif
FROM alert.csv
GROUP BY sig_id
ORDER BY Alerts DESC
---Ch05Summary-GraphTopAlerts.sql---
```

Finally, there are three queries for the remaining graphs:

```
---Ch05Summary-GraphTopSrcIPs.sql---
SELECT
    src,
    Count(msg) as Alerts
INTO report\AlertsTopSrcIPs.gif
FROM alert.csv
GROUP BY src
ORDER BY Alerts DESC
---Ch05Summary-GraphTopSrcIPs.sql---
SELECt
    Count(*) as Alerts
USING QUANTIZE(timestamp,360) as Hour
INTO report\AlertsByHour.gif
```

www.syngress.com

FROM alert.csv

```
GROUP BY Hour
---Ch05Summary-GraphAlertsPerHour.sql---
SELECT TOP 5
STRCAT(STRCAT(TO_STRING(dstport),' - '), proto) AS Destination,
Count(*) as Alerts
USING dst as DestinationPort
INTO report\AlertsTopDstPorts.gif
FROM alert.csv
GROUP BY Destination
ORDER BY Alerts DESC
---Ch05Summary-GraphTopDstPorts.sql---
```

Finally, we can generate the entire index page with these commands:

```
logparser.exe file:Ch05Summary-Index.sql -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:tpl -tpl:Ch05Summary-Index.tpl
logparser.exe file:Ch05Summary-GraphTopAlerts.sql -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:chart -chartType:Pie3D -groupSize:350x190 -
values:OFF -chartTitle:"" -categories:OFF
logparser.exe file:Ch05Summary-GraphTopSrcIPs.sql -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:chart -chartType:Pie -groupSize:300x150 -
values:OFF -chartTitle:"" -categories:OFF
logparser.exe file:Ch05Summary-GraphAlertsPerHour.sql -i:csv -iHeaderFile:AlertHeader.csv
-iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:chart -chartType:Smoothline -
groupSize:300x150 -values:OFF -chartTitle:"" -categories:OFF
logparser.exe file:Ch05Summary-GraphTopDstPorts.sql -i:csv -iHeaderFile:AlertHeader.csv -
iTsFormat:mm/dd/yy-hh:mm:ss -headerRow:off -o:chart -chartType:BarStacked -
groupSize:300x150 -values:OFF -chartTitle:""
```

The final result is a fully interactive IDS report using nothing more than Log Parser.

NOTE

You can create the entire report shown here using the CreateReport.cmd batch file included with the file download for this chapter.

Managing Snort Rules

Log Parser complements Snort with more than just reporting. Log Parser's powerful parsing features make it an ideal tool for managing and updating Snort rule definitions.

Snort's intrusion detection engine is nothing more than a network sniffer. The rule definitions are what give it the ability to identify attack patterns. For Snort to be most effective, you should customize the rules for your particular environment. You should also keep it up to date with the most recent rule definitions. Snort uses dozens of rule files, each categorized by attack category. You might find it useful to combine these rules into a single file for easier management. Because Log Parser can easily read and parse all files in a directory, rule management is simple.

The following query reads all rule files in a directory and sorts them by the signature ID, removing any duplicate rules you might have:

```
---Ch05RulesBvSID.sql---
SELECT DISTINCT
  TO INT(EXTRACT VALUE(Params, 'sid')) AS SID,
  Rule
USING
  Field1 AS Rule,
REPLACE_STR(REPLACE_CHR(SUBSTR(Rule,ADD(INDEX_OF(Rule,'('),1),LAST_INDEX_OF(Rule,')')),':'
,'='),'; ','&') AS Params,
INTO all.rules
FROM *.rules
ORDER BY SID
---Ch05RulesBySID.sql---
---Ch05RulesBySID.tpl---
<LPHEADER>
#_____
# Snort Rules sorted by SID
      Generated %SYSTEM TIMESTAMP%
# by %USERDOMAIN%\%USERNAME%
#_____
</LPHEADER>
<LPBODY>%Rule%
</LPBODY>
---Ch05RulesBySID.tpl---
   Run this query with the following command:
logparser file:Ch05Rulesbysid.sql -i:tsv -headerRow:off -lineFilter:-# -o:tpl -
tpl:Ch05RulesBySID.tpl
```

Note in the query that I did not use SUBSTR and INDEX_OF to extract the **sid** value, but rather used the replace functions to make the parameters look like a URL (Uniform Resource Locator) query string. This allows me to use the EXTRCT_VALUE functions to easily grab any value I want from the rule. Note also that I used a template file rather than outputting directly to a file. This is so I can include a comment header but also because I want to sort by SID, but not include the value in the output. You cannot sort by a field specified in the USING clause.

Using this same technique, you might find it useful to create a rules reference page. This query reads all rule definitions and generates the HTML reference page shown in Figure 5.6:

```
---Ch05RulesRef.sgl---
SELECT DISTINCT
   TO_INT(EXTRACT_VALUE(Params, 'sid')) AS SID,
         EXTRACT_VALUE(Params, 'classtype') AS Category,
         REPLACE_CHR(EXTRACT_VALUE(Params, 'msg'), '"', '') AS Message,
         Rule
USING
        Field1 AS Rule,
REPLACE_STR(REPLACE_CHR(SUBSTR(Rule, ADD(INDEX_OF(Rule, '('), 1), LAST_INDEX_OF(Rule, ')')), ':'
,'='),'; ','&') AS Params
INTO Rules.htm
FROM *.rules
ORDER BY SID
---Ch05RulesRef.sql---
---Ch05RulesRef.tpl---
<LPHEADER>
        <html>
        <head>
                <meta http-equiv="Content-Type" content="text/html; charset=windows-
1252">
                <title>Snort Rules Reference</title>
                <style>
<!--
H1 {
        font : bold 14pt Verdana, Geneva, Arial, Helvetica, sans-serif;
        color : #4A4322;
        }
TD {
        COLOR: Black; FONT: 11px Verdana, arial, geneva, helvetica, sans-serif;
                        border : 0px solid #EBE7D3;
                        vertical-align : top;
                        background-color : #EBE7D3;
                        }
TH {
        COLOR: Black; FONT: 11px Verdana, arial, geneva, helvetica, sans-serif;
        background-color : #9F9B64;
        text-align : left;
}
-->
</style>
        </head>
        <body>
        <h1>Snort Rules Reference</h1><br/>
```

```
<b>Signature</b>
                    <b>Message</b>
                    <b>Category</b>
               </LPHEADER>
<LPBODY>
               <a href=http://www.snort.org/snort-
db/sid.html?sid=%SID%> %SID%</a>
                  %Message%
                    %Category%
               </LPBODY>
<LPFOOTER>
         </body>
     </html>
</LPFOOTER>
---Ch05RulesRef.tpl---
```

Run this query with the following command:

```
logparser file:ch05RulesRef.sql -i:tsv -headerRow:off -lineFilter:-# -o:tpl -
tpl:Ch05RulesRef.tpl
```

Figure 5.6 Snort Rules Reference

🗿 Snort Ri	iles Reference - Microsoft Internet Explorer			.ox
<u>File</u> Edit	<u>View Favorites Tools H</u> elp			AV.
G Back	🕥 · 📓 🕼 🔎 Search ☆ Favorites 🤗 🔗	• 🗟 💿 • 📄		
Address 🖉	C:\Program Files\Snort\rules\Rules.htm		✓ → Go Li	nks » 🐑 🕇
Snort	Rules Reference			
Signature	Message	Category		
103	BACKDOOR subseven 22	misc-activity		
104	BACKDOOR - Dagger_1.4.0_client_connect	misc-activity		
105	BACKDOOR - Dagger_1.4.0	misc-activity		
106	BACKDOOR ACKcmdC trojan scan	misc-activity		
107	BACKDOOR subseven DEFCON8 2.1 access	trojan-activity		
108	BACKDOOR QAZ Worm Client Login access	misc-activity		
109	BACKDOOR netbus active	misc-activity		
110	BACKDOOR netbus getinfo	misc-activity		
111	BACKDOOR netbus getinfo	misc-activity		
112	BACKDOOR BackOrifice access	misc-activity		
113	BACKDOOR DeepThroat access	misc-activity		
114	BACKDOOR netbus active	misc-activity		
115	BACKDOOR netbus 2 active	misc-activity		
116	BACKDOOR BackOrifice access	misc-activity		
117	BACKDOOR Infector.1.x	misc-activity		
118	BACKDOOR SatansBackdoor.2.0.Beta	misc-activity		
119	BACKDOOR Doly 2.0 access	misc-activity		
120	BACKDOOR Infector 1.6 Server to Client	misc-activity		
121	BACKDOOR Infector 1.6 Client to Server Connection Request	misc-activity		
122	BACKDOOR DeepThroat 3.1 System Info Client Request	misc-activity		
124	BACKDOOR DeepThroat 3.1 FTP Status Client Request	misc-activity		
125	BACKDOOR DeepThroat 3.1 E-Mail Info From Server	misc-activity		
126	BACKDOOR DeepThroat 3.1 E-Mail Info Client Request	misc-activity		
127	BACKDOOR DeepThroat 3.1 Server Status From Server	misc-activity		~
🖉 Done			😼 My Comput	ter .:

Log Parser has a powerful feature that allows you to parse files directly from a URL. You can use this feature to download new Snort rule definitions. The following command downloads the newest Bleeding Snort rules from www.bleedingsnort.com.

logparser "SELECT Field1 INTO bleeding.rules FROM http://www.bleedingsnort.com/bleedingall.rules" -i:tsv -headerRow:off -o:tsv -headers:off

Of course, once you download the rules, you can merge and sort them as shown previously in this chapter.

Final Touches

As you can see, Log Parser is a powerful addition to Snort, both for reporting and for rules management. In very little time and using nothing more than Log Parser itself, you can create elaborate interactive HTML reports to view and drill down through thousands of IDS alerts. With Log Parser's charting features, you can view the data from different angles to help identify high priority alerts.