Gaining Insight Through Security Visualization

OSSIR Paris
2010/01/12

Saâd Kadhi -- HAPSIS
Agenda

★ Introducing Security Visualization
★ From Data to Graphs
★ Firemen For Firewalls
★ Visual Vulnerability Management
★ A Few Things To Know
Introducing Security Visualization
What Is Security Visualization?

★ "A picture is worth a thousand log records" (Raffael Marty)
★ It’s a process
★ Generating a picture (or graph) from log records (or security events in the broader sense)
From Events to Picture

★ SecViz takes security events as input and (should) produce a worthy visual representation

★ A worthy visual representation is a visual representation that provides insight and support decision-making
From Data To Insight

★ Visualization allows us to move from data to information and then move from information to insight

★ And insight is paramount! (Cluebats have yet to be invented)

1. cluebat
   A metaphorical bat used to 'beat some sense into' someone who is blatantly stupid

   some guy just tried to install 'crack_hotmail_passwords.exe' and he wonders why his machine is full of crap. someone needs to beat him with a cluebat.
Ever tried to analyze a log file of 529083 lines to try to understand why there's a sudden surge of tcp/25 connections that are about to take down your front-line defenses?

No? Then be my guest...
But Seriously, Why?

★ Because of the human visual system!

★ Pattern seeker

★ Massive, high-bandwidth, parallel processor

★ The human brain has a hard time processing text
More Sources, More Data, More Everything

★ Databases, documents, emails, websites...

★ Huge amount of data in this information-oriented era (and growing...)

★ We need new ways of sorting this mess out
Visualization Can Be One Answer

★ Display relevant information graphically to aid in understanding the data

★ Discover "hidden" relationships

★ Analyze a large amount of data very quickly
SecViz Mantra

Large Amount of Data

Application Logs
DNS Logs
Firewall Events
IDS/IPS Events
Network Events

Overview
Zoom and Filter
Details on-demand
Insight
From Data To Graphs
It’s Not a Perfect World
(far from it...)

- Visualization of data is not a straightforward process
- ... well, not always
- First, we need to define the problem and the objective (very, very clearly)
It’s Not a Perfect World (reloaded)

★ We also need to think about some choices to make: color assignments, type of graph to use...

★ These choices depend on the problem and the objective
Yes, We Need a Process

1. Define the problem
2. Assess available data / data sources
3. Parse/Filter data
4. Transform to visual representation
5. View visual representation
6. Interpret and decide
Process, Pictured
Define The Problem

★ What are you looking for?

★ What are you trying to find an answer for?

★ Example: Who is trying to connect to my SSH server?
Assess Available Data / Data Sources

★ What data is available? Log files?

★ Do we need any additional data

★ Example: /var/log/auth.log + GeoIP information
Parse & Filter

★ Parse and filter the data / data sources to extract the necessary information

★ The information needs to be normalized in order to be fed to the graph generating tool
Parse & Filter Example

Jan 11 12:06:00 skin sshd[8846]: error: PAM: authentication error for illegal user test1 from 220.162.241.11
Jan 11 12:06:00 skin sshd[8846]: Failed keyboard-interactive/pam for invalid user test1 from 220.162.241.11 port 45239 ssh2
Jan 11 12:06:20 skin sshd[8851]: Invalid user ts from 59.108.230.130
Jan 11 12:06:23 skin sshd[8853]: Invalid user ts from 59.108.230.130
Jan 11 12:06:25 skin sshd[8855]: Invalid user ts from 59.108.230.130
Jan 11 12:06:27 skin sshd[8857]: Invalid user ts from 59.108.230.130
Jan 11 12:06:29 skin sshd[8859]: Invalid user ts from 59.108.230.130
Jan 11 12:06:32 skin sshd[8861]: Invalid user ts from 59.108.230.130
Jan 11 12:06:34 skin sshd[8863]: Invalid user ts from 59.108.230.130
Jan 11 12:06:37 skin sshd[8865]: Invalid user ts from 59.108.230.130
Jan 11 12:06:39 skin sshd[8867]: Invalid user ts from 59.108.230.130
Jan 11 12:06:41 skin sshd[8869]: Invalid user ts from 59.108.230.130
Jan 11 12:06:43 skin sshd[8871]: Invalid user ts from 59.108.230.130
Jan 11 12:06:46 skin sshd[8873]: Invalid user ts from 59.108.230.130
Jan 11 12:06:48 skin sshd[8875]: Invalid user teamspeak from 59.108.230.130
Jan 11 12:06:50 skin sshd[8877]: Invalid user teamspeak from 59.108.230.130
Jan 11 12:06:53 skin sshd[8879]: Invalid user teamspeak from 59.108.230.130
Jan 11 12:06:55 skin sshd[8881]: Invalid user teamspeak from 59.108.230.130
Jan 11 12:06:57 skin sshd[8883]: Invalid user teamspeak from 59.108.230.130
Jan 11 12:07:00 skin sshd[8885]: Invalid user teamspeak from 59.108.230.130
Jan 11 12:07:02 skin sshd[8887]: Invalid user ts1 from 59.108.230.130
Jan 11 12:07:05 skin sshd[8889]: Invalid user ts1 from 59.108.230.130
Jan 11 12:07:07 skin sshd[8891]: Invalid user ts2 from 59.108.230.130
### Parse & Filter Example

<table>
<thead>
<tr>
<th>SrcIP; NumConn</th>
</tr>
</thead>
<tbody>
<tr>
<td>148.233.140.193;1</td>
</tr>
<tr>
<td>190.34.172.5;1</td>
</tr>
<tr>
<td>193.27.193.74;1</td>
</tr>
<tr>
<td>200.13.253.122;1</td>
</tr>
<tr>
<td>204.213.57.35;1</td>
</tr>
<tr>
<td>212.243.41.9;3</td>
</tr>
<tr>
<td>220.162.241.11;2</td>
</tr>
<tr>
<td>58.247.222.163;1</td>
</tr>
<tr>
<td>58.60.106.24;1</td>
</tr>
<tr>
<td>59.108.230.130;38</td>
</tr>
<tr>
<td>94.23.203.221;2345</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SrcIP; NumConn; CountryISO; CountryName</th>
</tr>
</thead>
<tbody>
<tr>
<td>148.233.140.193;1; MX; Mexico</td>
</tr>
<tr>
<td>190.34.172.5;1; PA; Panama</td>
</tr>
<tr>
<td>193.27.193.74;1; SE; Sweden</td>
</tr>
<tr>
<td>200.13.253.122;1; CO; Colombia</td>
</tr>
<tr>
<td>204.213.57.35;1; US; United States</td>
</tr>
<tr>
<td>212.243.41.9;3; CH; Switzerland</td>
</tr>
<tr>
<td>220.162.241.11;2; CN; China</td>
</tr>
<tr>
<td>58.247.222.163;1; CN; China</td>
</tr>
<tr>
<td>58.60.106.24;1; CN; China</td>
</tr>
<tr>
<td>59.108.230.130;38; CN; China</td>
</tr>
<tr>
<td>94.23.203.221;2345; FR; France</td>
</tr>
</tbody>
</table>
What properties do we need in the resulting graph? (i.e. choosing the right graph)

How about color, size, shape?

How about scale, layout, zooming in/out?

It’s time to introduce graph types!
Graph Types

★ There are far too many types (and variations)

★ Of particular interest are: pie charts, bar charts, histograms, link graphs and Treemaps
**Pie Charts**

- Well you know about these... your boss (and salespeople) crave them

- Compare single-dimensional values as parts / % of a whole

- Only a small number of different values at a time
Pie Charts

Source IPs for denied SSH logins

- 98%
- 0%

- 148.233.140.193
- 190.34.172.5
- 193.27.193.74
- 200.13.253.122
- 204.213.57.35
- 212.243.41.9
- 220.162.241.11
- 58.247.222.163
- 58.60.106.24
- 59.108.230.130
- 94.23.203.221
Bar Charts

★ Used to show the frequency of one-dimensional values

★ Each bar represents a value

★ The bar's height represents the frequency count
Bar Charts

Source IPs for denied SSH logins

NumConn

<table>
<thead>
<tr>
<th>Source IP</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>148.233.140.193</td>
<td>2250</td>
</tr>
<tr>
<td>193.27.193.74</td>
<td></td>
</tr>
<tr>
<td>204.213.57.35</td>
<td></td>
</tr>
<tr>
<td>220.162.241.11</td>
<td></td>
</tr>
<tr>
<td>58.60.106.24</td>
<td></td>
</tr>
<tr>
<td>94.23.203.221</td>
<td></td>
</tr>
<tr>
<td>190.34.172.5</td>
<td></td>
</tr>
<tr>
<td>200.13.253.122</td>
<td></td>
</tr>
<tr>
<td>212.243.41.9</td>
<td></td>
</tr>
<tr>
<td>58.247.222.163</td>
<td></td>
</tr>
<tr>
<td>59.108.230.130</td>
<td></td>
</tr>
</tbody>
</table>
★ **Histograms**

★ **Histograms look like bar charts**

★ **Bar charts are not suitable for continuous data while histograms are (ex. number of logins on any given day)**

★ **We can group thousand of values**
Link Graphs

★ Best-suited for visualizing relationships

★ two dimensions (ex. source IP, destination IP)

★ three dimensions (ex. source IP, destination port, destination IP)
Link Graphs
Link Graphs
TreeMaps

★ Best-suited for visualizing multi-dimensional, hierarchical data

★ Use size and color to encode specific properties

★ Extremely practical for visualizing large data sets
Interpret & Decide

★ So, what is the answer to the initial problem?

★ What actions shall be performed (if any)?
Firemen For Firewalls
Firewall Overload

★ On Aug 27th, 2009, Internet-facing firewalls got overloaded all of a sudden

★ legitimate traffic came to a halt

★ After some time, the firewalls felt better
The Problem

★ An MRTG-like graph tells us there is a spike in 25/tcp (SMTP?) connections to our mail relays

★ What happened?

★ What can we do to prevent it from happening again?
Available Data / Data Sources

★ Firewall log files
★ Mail relay log files
★ GeoIP
Firewall Log Files

★ Very valuable
★ Date and time, action, source IP, source port, destination IP, destination port, protocol...
★ The spike was seen for about 2h
★ That’s a 110MB, 529083 lines file
Mail Relay Log Files

★ Worthless

★ It took a rocket scientist (well, almost...) to figure out that these “best-of-breed” anti-spam appliances don’t record incoming connections but incoming connections once they passed the first stage of SPAM clearance!
The Quest Begins

★ How to make something out of that 110MB firewall log file?

★ Which IP connected to our mail relays, how many times, to which country does it belong and is it a legitimate MTA?

★ Secondary mail relays? botnet?...
Parse and Filter

★ A quick "grep 1 sort -u" etc... gives us some initial information

★ 529083 lines translate into 125859 unique source IPs (uh oh...)

★ A Perl script tells us 119812 IPs made 10 connections or less
On The Way To Visualization

★ We want to see visually the source IPs and the number of connections each one made

★ What graph type shall we choose?
Preparing The Transformation

★ We need to normalize the data

★ Depending on the TreeMap tool, we must either use a specific format (TM3 files for HCIL TreeMap) or a more general-purpose one (CSV,...)

★ CSV is a good choice (and HCIL TreeMap is a nice piece of bloatware)
The log file we received was in TXT

Quite trivial to parse and transform into CSV

We can also add GeoIP information
From TXT Log File To CSV

"Number" "Date" "Time" "Interface" "Origin" "Type" "Action" "Service" "Source Port" "Source" "Destination" "Protocol" "Rule" "Rule Name" "Current Rule Number" "User" "Information" "Product"
"13" "27Aug2009" "13:58:28" "eth-s1p4c0" "mainsite-fies1p4-c003" "Log" "Accept" "tcp-25" "1538" "117.204.18.22" "mainsite-mailrelay1-dmz" "tcp" "17" "" "17-dn"
"service_id: tcp-25" "VPN-1 Power/UTM"
"15" "27Aug2009" "13:58:28" "eth-s1p4c0" "mainsite-fies1p4-c003" "Log" "Accept" "tcp-25" "35187" "76.215.109.27" "mainsite-mailrelay2-dmz" "tcp" "17" "" "17-dn"
"service_id: tcp-25" "VPN-1 Power/UTM"
"24" "27Aug2009" "13:58:28" "eth-s1p4c0" "mainsite-fies1p4-c003" "Log" "Accept" "tcp-25" "4823" "200.43.109.166" "mainsite-mailrelay1-dmz" "tcp" "17" "" "17-dn"
"service_id: tcp-25" "VPN-1 Power/UTM"

IP Addr;NumConn;Country (ISO);Country (Name);
110.10.163.173;2;KR;Korea, Republic of;
110.10.249.70;4;KR;Korea, Republic of;
110.10.50.208;2;KR;Korea, Republic of;
110.11.217.209;2;KR;Korea, Republic of;
110.11.27.99;1;KR;Korea, Republic of;
110.12.108.16;4;KR;Korea, Republic of;
110.12.148.248;1;KR;Korea, Republic of;
110.12.84.77;2;KR;Korea, Republic of;
110.137.108.101;1;ID;Indonesia;
110.137.110.115;4;ID;Indonesia;
110.137.111.167;1;ID;Indonesia;
110.137.111.67;1;ID;Indonesia;
110.137.160.14;6;ID;Indonesia;
110.137.160.47;6;ID;Indonesia;
110.137.161.37;2;ID;Indonesia;
110.137.166.231;2;ID;Indonesia;
Viewing The Results

★ To put it otherwise, let’s load the 125k line CSV in the Macrofocus TreeMap tool (way better than HCIL TreeMap)

★ We need to fiddle a bit with the color, shape, grouping etc. to get the best from our data
Doesn't sound like legitimate business, does it?

25/tcp probe of all those 125k unique IPs: 96912 filtered, 7836 open, 21098 closed

Only 6.23% answer on 25/tcp
Visual Vulnerability Management
A Real Situation

★ Company with many business units located worldwide

★ In order to keep the attack surface as small as possible, a vulnerability discovery service is offered

★ Regular vulnerability scanning
★ Local contacts in business units have (very) limited time and sometimes basic security knowledge

★ Vulnerability scanner reports only on highly-critical, remotely-exploitable vulnerabilities
Deliverables

★ After each scan campaign, the local contacts receive HTML (yes, Web 2.0-style!) reports, scoring, and Excel spreadsheet giving an overview of which assets are more or less vulnerable.

★ But a spreadsheet is still text...
Taking It To The Next Level

★ We need to prioritize actions more efficiently

★ i.e. concentrate efforts on the more valuable assets with the highest number of vulnerabilities

★ Let’s get visual with TreeMaps!
Enriching Data

★ The vulnerability scanner gives us the number of vulnerabilities for each asset

★ We need to add the "business value"

★ Scale from 1 to 3. The higher the value, the more valuable an asset is
In The Beginning There Was Text

<table>
<thead>
<tr>
<th>Asset IP addr</th>
<th>Scan Results</th>
<th>Business Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>x.y.2.23;2;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.5.131;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.35.4;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c.d.28.41;2;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a.b.80.112;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a.b.43.166;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.5.132;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.35.5;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.105.20;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a.b.43.167;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.5.133;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a.b.1.38;1;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.35.6;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a.b.80.161;1;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.105.21;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>e.f.1.35;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c.d.28.42;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a.b.80.114;1;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a.b.43.168;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.5.134;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a.b.1.39;1;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.35.7;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.75.2;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.105.22;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a.b.8.160;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.5.135;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.75.3;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.105.23;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a.b.8.161;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a.b.43.76;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.35.9;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.105.24;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a.b.80.117;0;1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x.y.125.10;0;1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
And a TreeMap Appeared
A Few Things To Know
Still a Young Field

★ SecViz is not really mature at this point

★ It picked up some momentum in 2007 and some active research is being conducted since then
Common Pitfalls

★ There are very few industrial-grade tools

★ Time spent parsing, filtering and normalizing data can be a hurdle

★ The problem of filtering too much / not enough
Tools of The Trade

★ Data capture: tshark

★ Classic Unix tools (grep, sed, awk, perl, ruby...)

★ Linkgraphs: AfterGlow, GraphViz

★ TreeMaps: Macrofocus TreeMap

★ DAVIX Linux Distribution
Reference

★ SecViz Web community

★ Conti G., Security Data Visualization.

Thanks!

★ OSSIR

★ HAPSIS

★ and of course to you for listening to my babble
Get The Slides

★ You can get the slides from the OSSIR website

★ Questions ? Comments ?

› saad.kadhi@hapsis.fr
★ Saâd Kadhi, I.S. Security Consultant, HAPSIS (http://www.hapsis.fr/)
License

★ Creative Commons Attribution-NonCommercial 3.0

▶ [http://creativecommons.org/licenses/by-nc/3.0/](http://creativecommons.org/licenses/by-nc/3.0/)