Encounter Complexes For Clustering Network Flow

Leigh Metcalf, lbmetcalf@cert.org
Flocon 2015
Date

.

Network Flow

Clustering Network Flow for fun and profit!

Previously done for finding Trojans, Botnets, Spoofed flows…

But those methods use ‘known behavior’ to find repeats of that behavior.

Network Flow

The Encounter Complex uses no prior knowledge in its creation.

It is based on encounter traces, which occur when two nodes meet. We record the time and analyze the data.

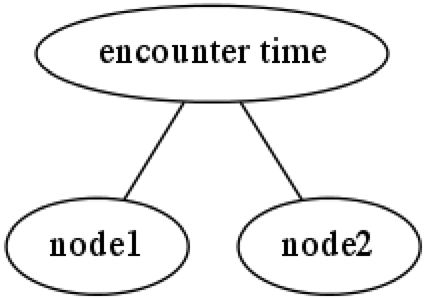
Encounter traces can include:

Two animals meet at a watering hole.

Two users use the same wireless node.

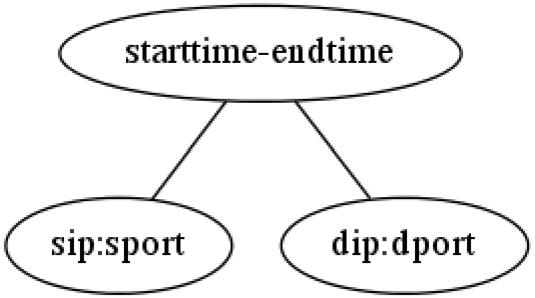
Encounter Trace

Encounter traces are defined as:



Encounter Trace for Network Flow

Defined as:



I maintain the time period rather than just a single moment in time.

Encounter Complex

Two traces have an edge between them if:

1. They share a node in common
2. The end of one occurs within Δ seconds of the start of the next

Encounter Complex

Let’s assume Δ=8

sIP:sPort	dIP:dPort	stime	etime
192.0.2.5:80	192.0.2.200:5265	1412870783	1412870880
192.0.2.199:5353	192.0.2.5:80	1412870885	1412871150

These two flows are connected since the first ends within 5 seconds of the second beginning.

Encounter Complex

Still assuming Δ=8

sIP:sPort	dIP:dPort	sTime	eTime
192.0.2.5:80	192.0.2.200:5265	1412870783	1412870880
192.0.2.199:5353	192.0.2.5:80	1412870885	1412871150
192.0.2.150:5353	192.0.2.3:25	1412870887	1412871175
192.0.2.5:80	192.0.2.205:5031	1412871160	1412871200

The third row does not share a node in common with the first two.

The second fails Δ=8 test, but would be part of the complex if Δ ≥ 10

Encounter Complex

We denote the Encounter Complex by G_Δ

Proposition:

If $\Delta \leq \Gamma$ then $G_\Delta \subseteq G_\Gamma$

This is clear because if two nodes are within Δ seconds of each other they are certainly within Γ seconds of each other.

Encounter Complex – Example

I used the LBNL data set.

- 11Gb of anonymized data
- Collected from October 2004 through January 2005
- Contains approximately 2.2 million flows
- Covers a wide variety of enterprise traffic

Encounter Complexes – Example

The time I chose had data from two sensors and contained:

- 47,834 network flows
- 1,423 IP addresses
- Average length of flow was 41.34 seconds
- Covered a little over an hour of traffic

Encounter Complex – Example

I created complexes for 7 values of Δ:

Δ	**Number of Graphs**	**Edges**	**Vertices**
1	6115	182,485	37,184
50	1498	3,681,789	40,623
100	891	6,769,521	40,763
200	695	12,551,635	40,807
300	597	18,325,825	40,822
400	537	23,605,755	40,831
Infinity	363	106,281,681	40,858

Encounter Complexes – Example

When Δ=infinity, there is quite a lot of work to be done creating the graph. It's essentially n^2 where n is the number of flows.

It does contain all of the other graphs though…

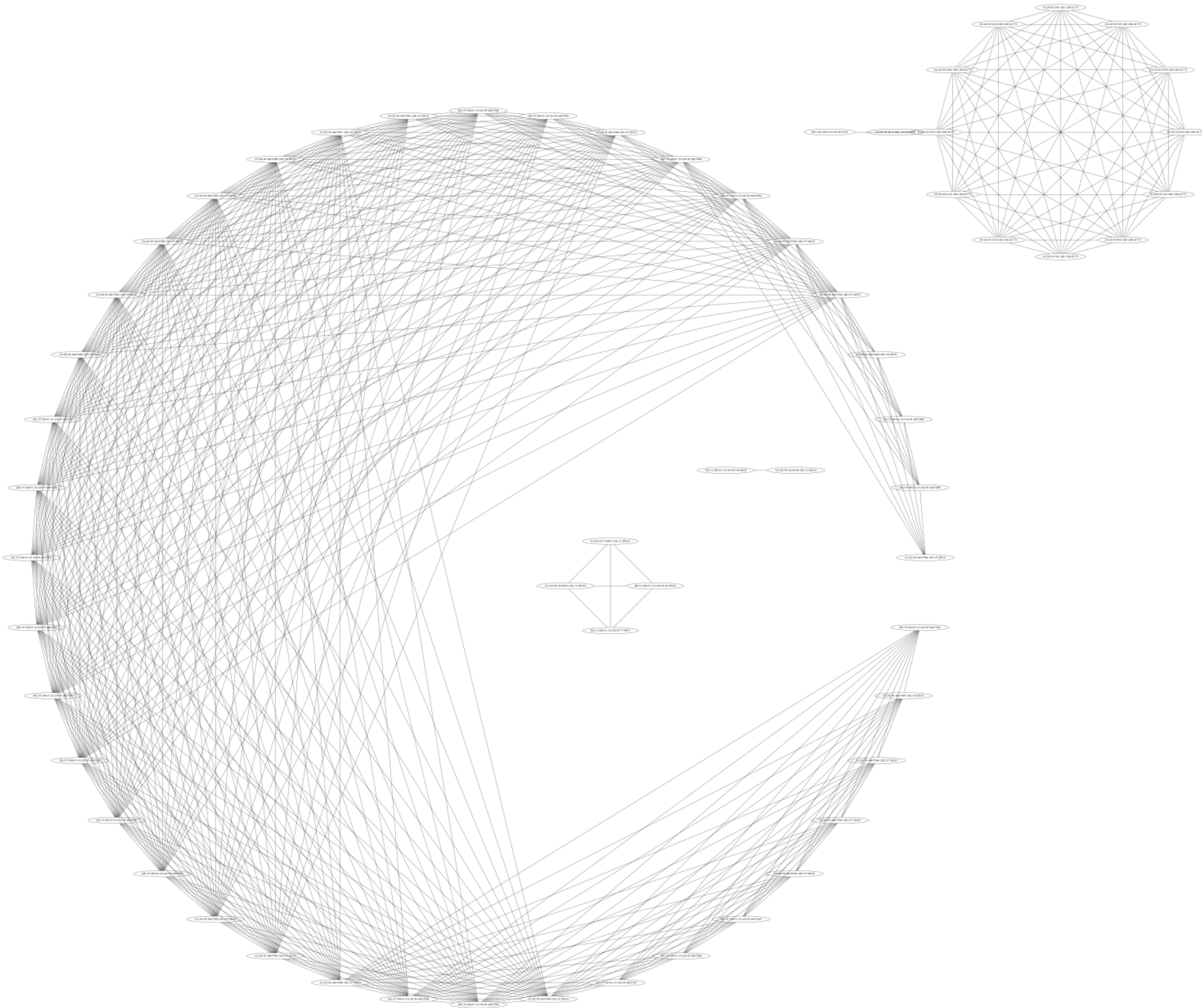
Encounter Complexes – Example

Analyzing it by visualization isn't very useful.

When Δ=1 there are 6115 graphs to analyze…

Four of which are on the next slide.

Encounter Complexes – Example



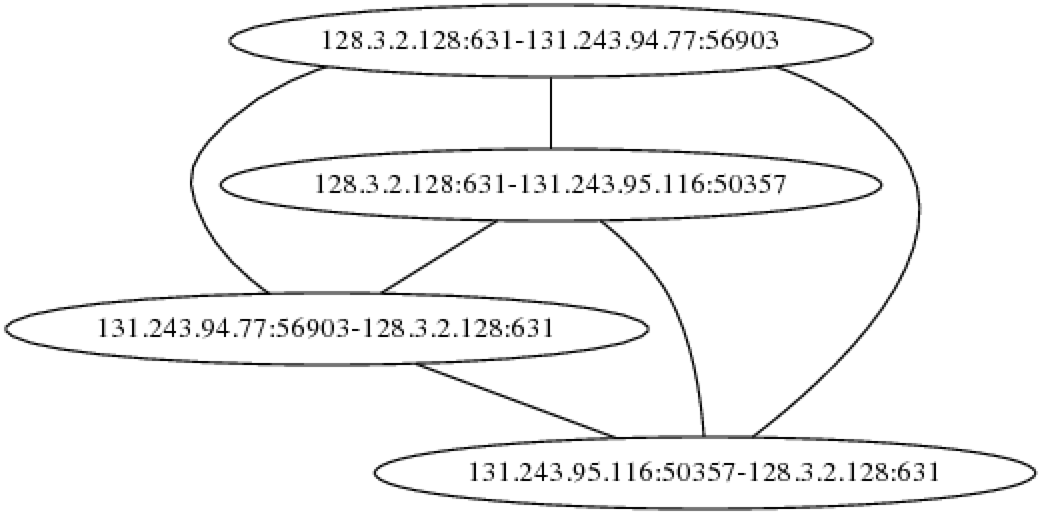
Encounter Complexes -- Example

We can analyze them by looking at two things:

- The vertex with the highest degree
- The node that is most prevalent through the graph

Encounter Complexes -- Example

This graph has 128.3.2.128:631 as the most common node…. Could be printing!



Clustering the Clusters

We've created graphs from the network flow…

…now we want to cluster those graphs

- Similar traffic!
- Fewer things to look at!
- Everyone wins!

Clustering the Clusters

There are three steps to creating the clusters:

1) Group together those graphs with similar port
   - Look at the vertex with the highest degree and consider the ports there.

2) Now refine those clusters by putting together graphs with a similar amount of vertices
   - Where ‘similar amount’ means within 10%

Digression into Graph Theory

Graph isomorphisms are an NP complete problem

(This means it is impossible in a reasonable amount of time)

Graph similarity has as many methods as mathematicians working on the problem

…so of course, I came up with my own method.

Digression into Graph Theory

The degrees of vertices within the graph in an encounter complex are a measure of similarity within that graph

The higher the degree, the more similar the vertex is to other vertices in the graph

Digression into Graph Theory

Method:

Given two graphs G_1 and G_2 create a sorted degree vector for each graph. (That is, put all of the degrees of each graph in a vector then sort it.)

If one vector is shorter than the other, pad that one with zeroes until they match in size.

Digression into Graph Theory

We can have two graphs with the same number of edges, vertices and cycles that have different degree vectors.

Example: A graph with 7 edges, 6 vertices had 2 cycles.

[3, 3, 2, 2, 2, 2]

[5, 3, 2, 2, 1, 1]

Both valid degree vectors for this graph.

Digression into Graph Theory

Once you have the two vectors, use the Pearson coefficient as a distance measure.

Pearson measures the linear dependence between the two vectors.

Digression into Graph Theory

It is also possible to have two graphs that are distinctly different but the Pearson coefficient of the degree vectors is 1.

Digression into Graph Theory

Example:

A graph with 42 edges, 10 vertices and 33 cycles:

[9, 9, 9, 9, 9, 9, 8, 8, 7, 7]

A graph with 29 edges, 10 vertices and 20 cycles:

[7, 7, 7, 7, 7, 7, 5, 5, 3, 3].

Pearson coefficient is 1 in this case.

These two graphs are modelling similar behavior

Clustering the Clusters

Last step of refinement:

3) Two graphs are in the same cluster if their Pearson coefficient is greater than 0.9

Encounter Complexes – Example

Δ	Clusters	Number of Clustered Graphs
1	99	756
50	29	193
100	6	32
200	4	11
300	6	14
400	3	6
infinity	0	0

Encounter Complexes -- Example

For Δ=1 I found a cluster with 47 graphs.

All of these graphs had sIP:50122 in common.

50122 can be used for:

SAP, Symantec and SSH forwarding

Without more information, I don't know much… other than they have common activity across the graphs

Encounter Complexes – Example

Another cluster had 50 graphs.

- Port 80 is common across the cluster
- But no common node

We found similar web traffic patterns

Comparing Encounter Complexes

Clustering the clusters works well when looking at a single complex…

…What if I compare two complexes?

Comparing Encounter Complexes

I chose a second time period from the LBNL data.

Contained:

- 127,223 flows
- 4,490 IP addresses
- A little over an hour of data

I created a complex where Δ=1

- Contained 14,676 components

Comparing Encounter Complexes

I then compared the two complexes using the criteria listed before:

1. Similar port
2. Similar size
3. Pearson measurement of degree vectors > 0.9

Comparing Encounter Complexes

I found 63 clusters when I compared the two graphs containing a total of 2,087 subgraphs.

I examined one cluster that contained 8 subgraphs

2 from one encounter complex

6 from the other encounter complex

The common port was 427 but the destination IP address varied

Future Work

- Bytes!
  - Weight the encounter complexes with the bytes transferred in the process
- Protocol!
  - Label the encounter complexes using the protocols in the flow
- Persistent Homology
  - Apply this to the infinity graphs to compare encounter complexes

Questions/comments?