LA-UR-11- 11-00221

Approved for public release;
distribution is unlimited.

Title: Graph Anomalies in Cyber Communication

Author(s): Scott A. Vander Wiel
Curtis B. Storlie
Gary Sandine
Aric Hagberg
Michael Fisk

Intended for: 2011 INFORMS Computing Society Conference,
Monterey, California

ABSTRACT

Graph Anomalies in Cyber Communication

Scott A. Vander Wiel
Curtis B. Storlie
Gary Sandine
Aric Hagberg
Michael Fisk

2011 INFORMS Computing Society Conference,
Monterey, California

Enterprises monitor cyber traffic for viruses, intruders and stolen information. Detection methods look for known signatures of malicious traffic or search for anomalies with respect to a nominal reference model. Traditional anomaly detection focuses on aggregate traffic at central nodes or on user-level monitoring. More recently, however, traffic is being viewed more holistically as a dynamic communication graph. Attention to the graph nature of the traffic has expanded the types of anomalies that are being sought. We give an overview of several cyber data streams collected at Los Alamos National Laboratory and discuss current work in modeling the graph dynamics of traffic over the network. We consider global properties and local properties within the communication graph. A method for monitoring relative entropy on multiple correlated properties is discussed in detail.

Graph Anomalies in Cyber Communication

Scott Vander Wiel[a], Curtis Storlie[a],
Gary Sandine[b], Aric Hagberg[b]
and Michael Fisk[c]

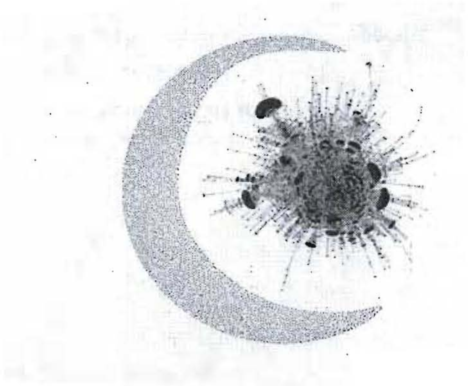
[a] *Statistical Sciences Group*
[b] *Applied Mathematics and Plasma Physics*
[c] *Information Sciences Group*

Los Alamos National Laboratory

January 10, 2011

Outline

Motivation
Temporal Communication Graphs
Modeling Dynamic Properties
Detecting Anomalies



January 27, 2010

ExxonMobil, **ConocoPhillips** and **Marathon Oil** executives were unwittingly duped by unsolicited e-mails carrying data-extracting malware .... This new form of corporate and, quite possibly, nation-sponsored espionage utilized custom spyware that is virtually undetectable by antivirus software applications used by the vast majority of large companies around the globe.

internetnews.com

January 15, 2010

Juniper Networks and **Symantec** ... are investigating a widespread cyber-espionage incident that has hit dozens of technology companies, including **Google** and **Adobe**.

Sources familiar with the situation say that 34 companies, most of them large Fortune 500 names, were hit by a sophisticated cyber attack, first uncovered by Google last month. The attackers used a previously unknown "zero-day" attack on Internet Explorer, and possibly other techniques, to break into company networks and steal sensitive information.

InfoWorld

2009

We regularly face attempts by others to gain unauthorized access ... by masquerading as authorized users or surreptitious introduction of software. These attempts ... are sometimes successful. ... We seek to detect and investigate these security incidents and to prevent their recurrence, but in some cases we might be unaware of an incident or its magnitude and effects.

***Intel** report to SEC*

Attack Detection and Response

Cyber Attack: misuse of a networked system such as

- penetration (*intrusion, exploitation*),
- remote command and control
- exfiltration of data
- denial of availability or integrity

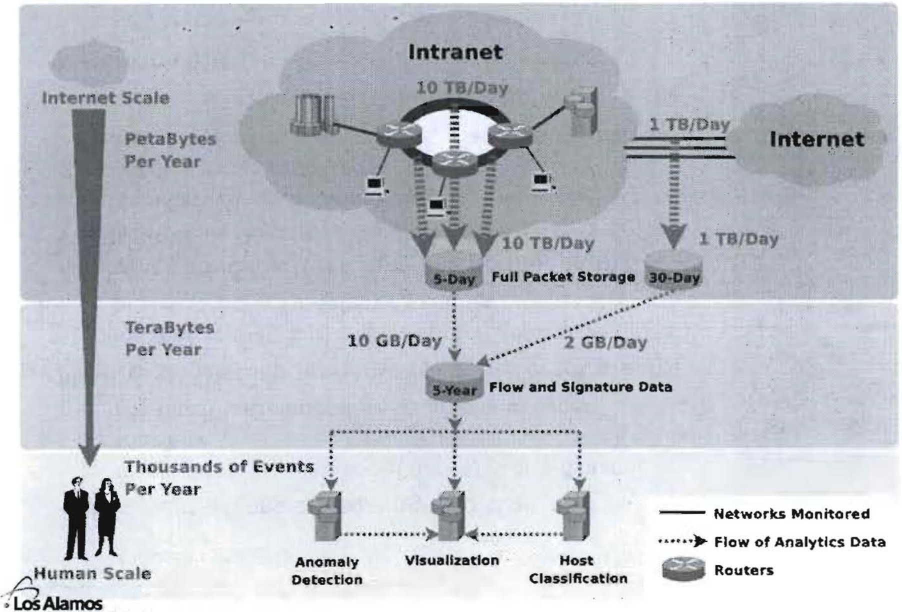
Goals: use observed sensor data to

- detect known attack methods and tools
- **detect unexplained patterns that could be attacks**
- prioritize responses based on likelihood of being an attack

New efforts to detect today's subtle cyber adversaries

- focus on invariants of attacker objectives rather than artifacts of specific attack technologies
- structural detection rather than simple rate-based detection

Scale of Data Collection at Los Alamos



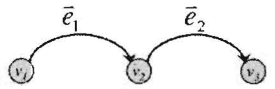
Temporal Communication Graphs

Many cyber data sets can (and should) be described as graphs

- vertices are hosts, users, etc.
- directed edges are communications
- events from heterogeneous sensors can be combined in one graph

A graph construction supports traditional local analysis while enabling new analysis

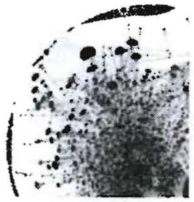
- exploitation is often a temporal path through the network
- long paths alter structural characteristics of the graph



$$\bar{e}_1 = \left\langle \begin{matrix} ([t_1,t_2],\ \underbrace{a,b,c,\ldots)}_{} \\ \ldots \end{matrix} \right\rangle$$

Attributes:

- protocol: tcp, udp, ...
- packet count, bytes, ...
- type: chat, web, ssh, ...
- probability of edge
- user type: scientist, admin, ...



Properties of Cyber Graphs

Cyber event graphs are coming into their own

- much graph work on social networks and dynamic topology
- not much on temporal event graphs in cyber

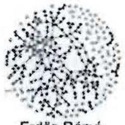
Need to understand properties of cyber event graphs

- improve relevance of random graph generative models
- provide a basis for detecting change

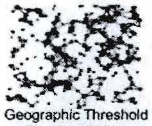
Anomalies are departures from the normal diurnal evolution of

- local activity on a node or edge
- distribution of subgraphs
- global properties (components, diameter, density, size)

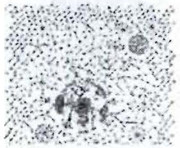
Hypothesis: a nosy intruder cannot avoid altering the graph structure



Erdös-Rényi Random Graph



Geographic Threshold Random Graph



5-min window of FTP traffic [Iliofotou 2009]

1-min window of SSH traffic (LANL)

Subgraph Statistics

Enumerate a set of subgraphs. Some options:

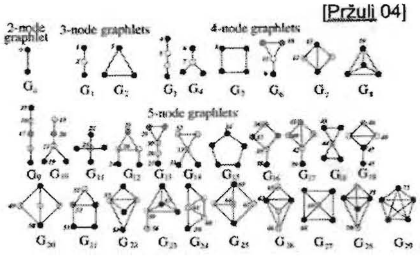
- connected components
- graphlets
- temporal walks (paths)
    - telescoping
    - consecutive
    - overlapping

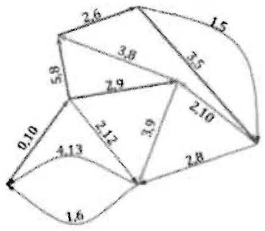
Measure subgraph features

- order
- diameter
- density

The distribution of features over subgraphs is a statistic of the full graph and evolves over time.

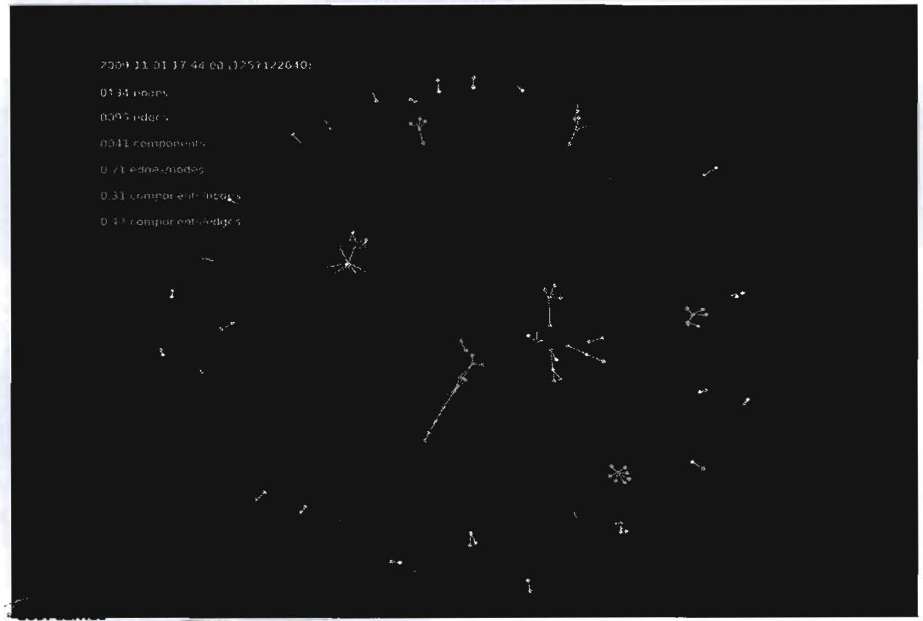
[Pržulj 04]





Path-based partitioning

SSH Graph from LANL's Network



Netflow-Type Data

Netflow is one type of data sampled from LANL's internal network by the routers. A Netflow record contains the following.

1. Source IP address
2. Source port
3. Destination IP address
4. Destination port
5. IP protocol
6. Ingress interface ID
7. IP Type of Service

In this talk we focus on SSH sessions (port=22) and build a graph for each ten-minute time interval in the month of November using the edges with connection times in the given interval

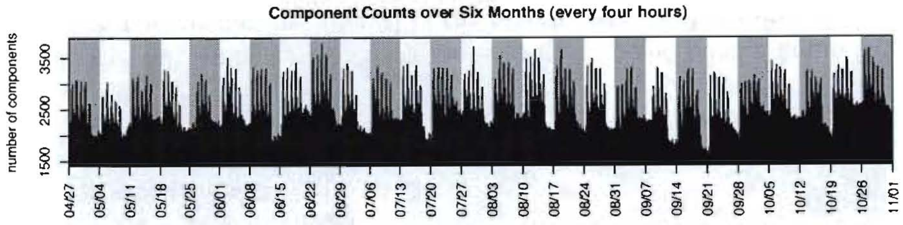
Summaries of SSH Graphs

Summarize each *connected component* by
order: O = number of nodes
diameter: D = greatest distance between any pair of vertices

Each ten-minute bin of time provides a set of (O,D) pairs from disconnected components

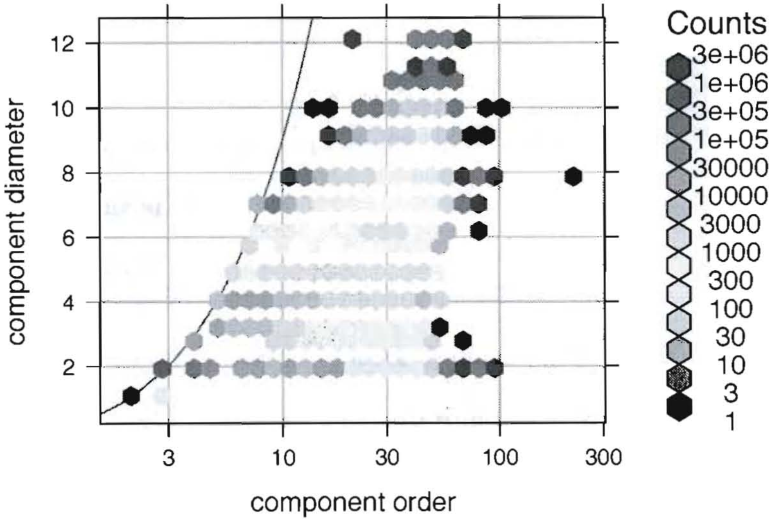
Future work: summarize using *graphlets* (arbitrary subsets of nodes and all edges between them)

Patterns over Time



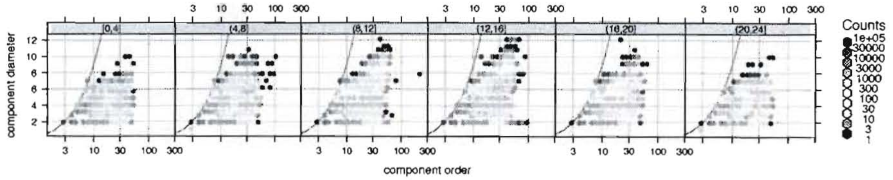
Numbers of components follow predictable daily and weekly patterns.

(O,D) Distribution Aggregated to a Full Month

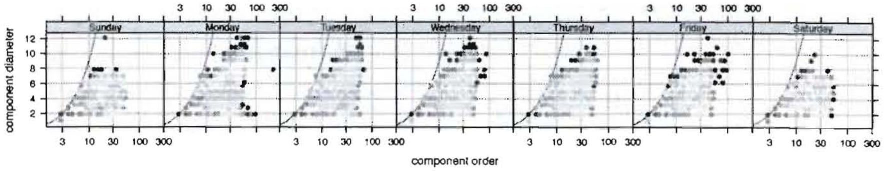


Daily and weekly trends in (O,D)

By hour of day



By day of week



A Dynamic Model for Multivariate Count Data

$$\begin{aligned}\mathbf{x} &= (\text{O,D})\\ \mathcal{X} &= \text{set of possible (O,D) pairs}\\ y_t(\mathbf{x}) &= \text{count of subgraphs with (O,D) equal } \mathbf{x}\end{aligned}$$

Observe $\{y_t(\mathbf{x}) : \mathbf{x} \in \mathcal{X}\}$ over a sequence of times $t = 1, 2, \ldots$.

Goal: monitor counts over time and alarm if observed counts depart from established patterns.

Model the counts at time t as

$$\begin{aligned}\{y_t(\mathbf{x}) : \mathbf{x} \in \mathcal{X}\} | M_t &\sim \text{Multinomial}(M_t, \{p_t(\mathbf{x}) : \mathbf{x} \in \mathcal{X}\}),\\ M_t &\sim \text{Poisson}(\lambda_t).\end{aligned}$$

M_t is the total count over the table.
Predict λ_t and $p_t(\mathbf{x})$ from data up to $t-1$.

Predict λ_t by Kernel Filtering Previous Counts, M_s

$$\hat{\lambda}_{t|t-1} = \frac{\sum_{d=0}^{\infty} \sum_{s=1}^{t-1} K_\lambda(t, s, d, \mathbf{h}_\lambda) M_s}{\sum_{d=0}^{\infty} \sum_{s=1}^{t-1} K_\lambda(t, s, d, \mathbf{h}_\lambda)},$$

The kernel function decomposes as

$$K_\lambda(t, s, d, \mathbf{h}_\lambda) = K_D(d, h_{\lambda,d}) K_B(t, s, d, h_{\lambda,b})$$

K_D reaches back to data on previous days

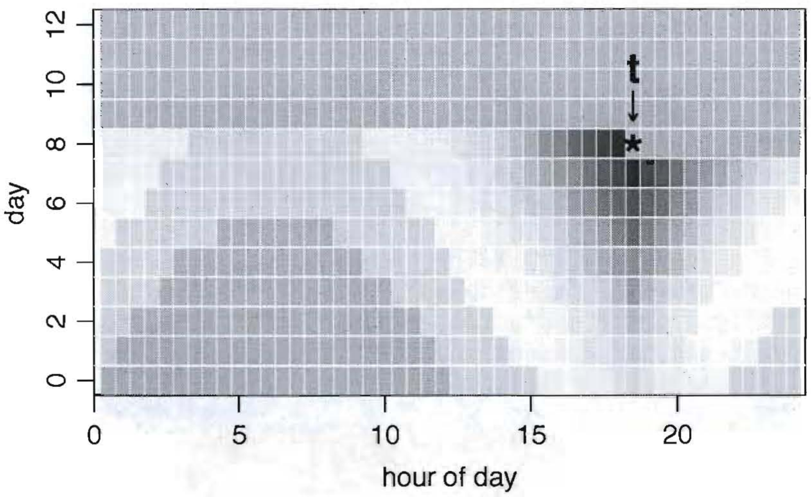
$$K_D(d, h_{\lambda,d}) = (1 - h_{\lambda,d})^d$$

K_B reaches back to data on previous time bins

$$\begin{aligned}K_B(t, s, d, h_{\lambda,b}) =&(1 - h_{\lambda,b})^{|t-d*T-s|}\\ &(T = 144 \text{ bins/day})\end{aligned}$$

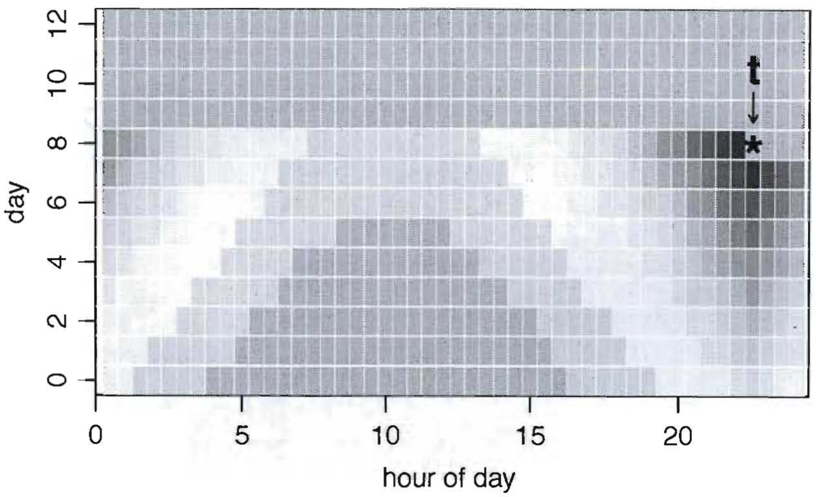
$\mathbf{h}_\lambda = (h_{\lambda,d}, h_{\lambda,b})$ are tuning parameters chosen by cross-validation.

The Kernel for Predicting λ_t from $M_1, \ldots, M_{t-1}$



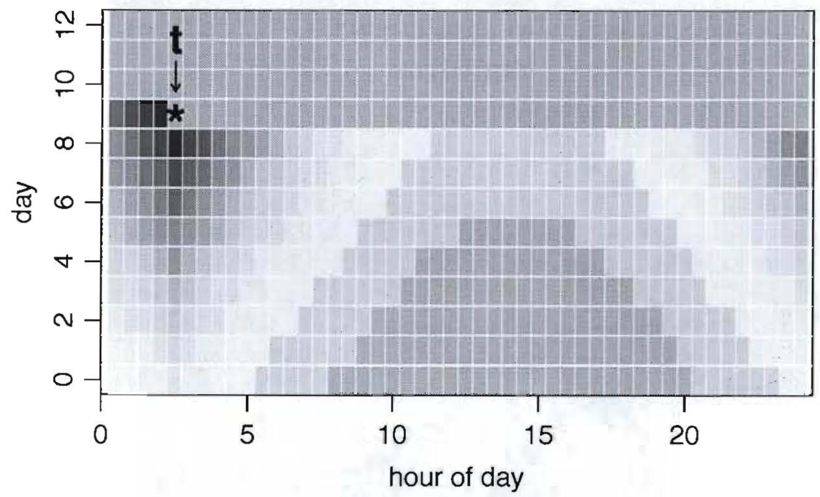
$\mathbf{h}_\lambda = (0.17, 0.13)$ — equivalent to 11 days × 14 half-hour bins

The Kernel for Predicting λ_t from $M_1, \ldots, M_{t-1}$



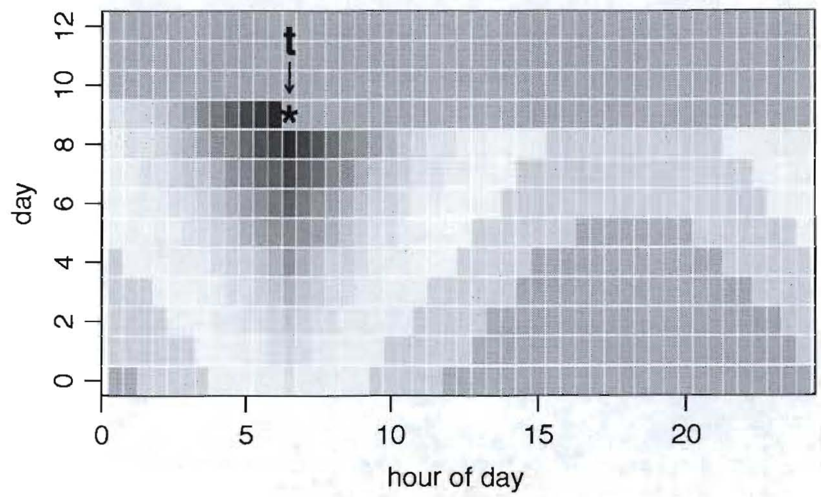
$\mathbf{h}_\lambda = (0.17, 0.13)$ — equivalent to 11 days × 14 half-hour bins

The Kernel for Predicting λ_t from $M_1, \ldots, M_{t-1}$



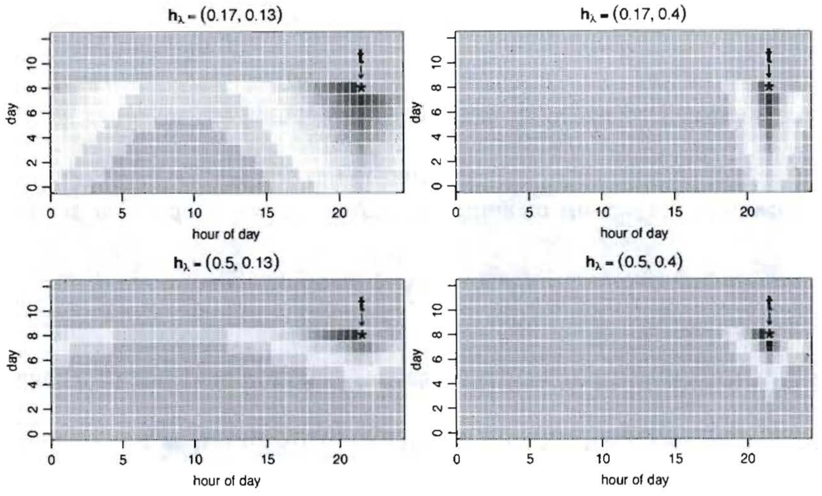
$\mathbf{h}_\lambda = (0.17, 0.13)$ — equivalent to 11 days × 14 half-hour bins

The Kernel for Predicting λ_t from $M_1, \ldots, M_{t-1}$



$\mathbf{h}_\lambda = (0.17, 0.13)$ — equivalent to 11 days × 14 half-hour bins

Adjusting the Tuning Parameters



$h_{\lambda,d}$ adjusts height; $h_{\lambda,b}$ adjusts width

Predict $p_t(\mathbf{x})$ by Kernel Filtering

Filter previous empirical (O,D) distributions

$$\{\tilde{p}_s(\mathbf{x}) \equiv y_s(\mathbf{x})/M_s : \mathbf{x} \in \mathcal{X}\} \qquad (s < t)$$

to predict $p_t(\mathbf{x})$ as

$$\hat{p}_{t|t-1}(\mathbf{x}) = \frac{\sum_{d=0}^{\infty}\sum_{s=1}^{t-1}\sum_{\mathbf{w}\in\mathcal{X}} K_P(t,s,d,\mathbf{x},\mathbf{w},\mathbf{h}_p)\tilde{p}_s(\mathbf{w})}{\sum_{d=0}^{\infty}\sum_{s=1}^{t-1}\sum_{\mathbf{w}\in\mathcal{X}} K_P(t,s,d,\mathbf{x},\mathbf{w},\mathbf{h}_p)},$$

Choose tuning parameters $\mathbf{h}_p = [h_{p,d}, h_{p,b}, h_{p,1}, \ldots, h_{p,q}, h_{p,\delta}]$ by cross validation.

Predict $p_t(\mathbf{x})$ by Kernel Filtering (cont.)

The kernel decomposes as

$$K_P(t,s,d,\mathbf{x},\mathbf{w},\mathbf{h}_p) = K_D(d,h_{p,d})K_B(t,s,d,h_{p,b})K_W(\mathbf{x},\mathbf{w},\mathbf{h}_p)M_t^{-1}$$

with day (K_D) and time-bin (K_B) kernels as before and now a *within* (O,D) kernel

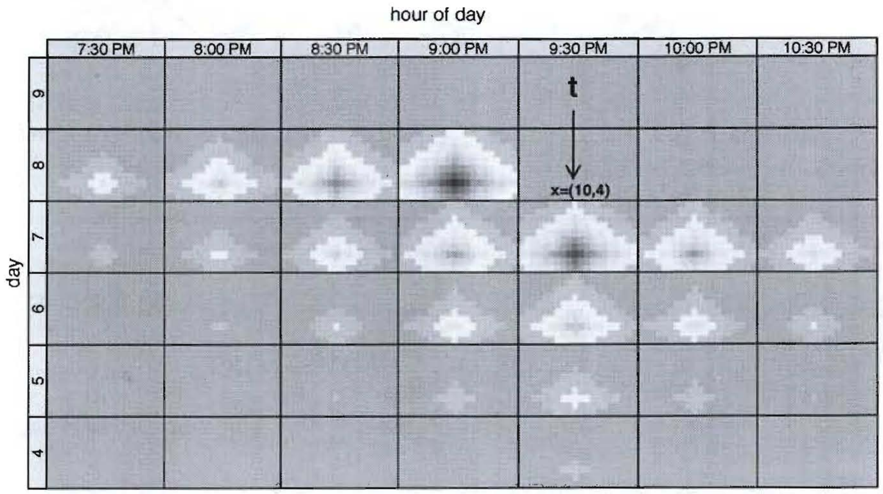
$$K_W(\mathbf{x},\mathbf{w},\mathbf{h}_p) = \prod_{j=1}^{2}(1-h_{p,j}h_\delta^{x_j})^{|x_j-w_j|}.$$

Parameters $h_{p,j}$ reduce kernel weights according to the distance between a given (O,D) pair ($\mathbf{w}$) and the target pair ($\mathbf{x}$)

Parameter h_δ makes the kernel wider for larger x_j, where probabilities are likely smaller.

M_t^{-1} weights by inverse variance.

Kernel for Estimating $p_t(10,3)$



Anomaly Detection

At time $t-1$, a predictive distribution for the next set of counts is

$$y_t(\mathbf{x}) \sim \text{Poisson}(\hat{\lambda}_{t|t-1}\hat{p}_{t|t-1}(\mathbf{x})).$$

These are independent over $\mathbf{x} \in \mathcal{X}$ under the multinomial model.

We use exceedance probabilities to detect unusually large subgraph counts relative to predictions:

$$U_t(\mathbf{x}) \equiv \Pr(\text{Poisson} \geq y_t(\mathbf{x}))$$

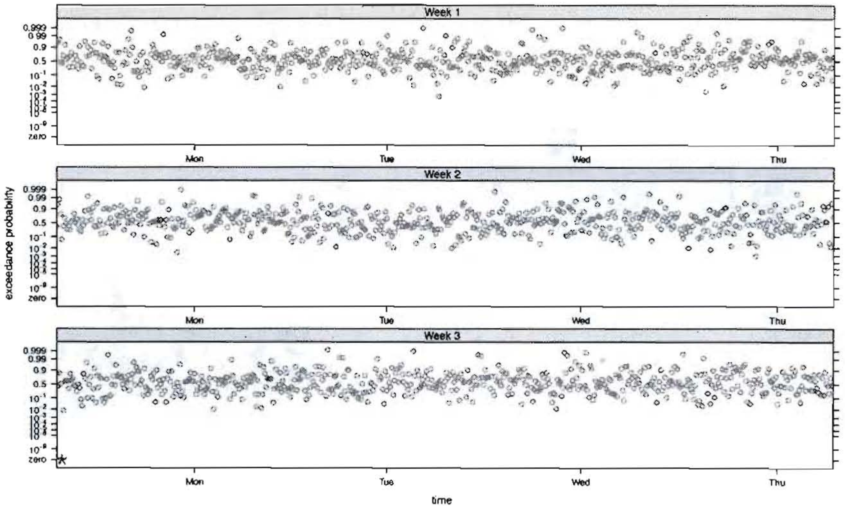
The joint exceedance probability

$$U_t = \prod_{\mathbf{x}\in\mathcal{X}} U_t(\mathbf{x})$$

has a known distribution (negative log-gammma if randomization is used), providing a reference to calculate p-values for joint exceedance.

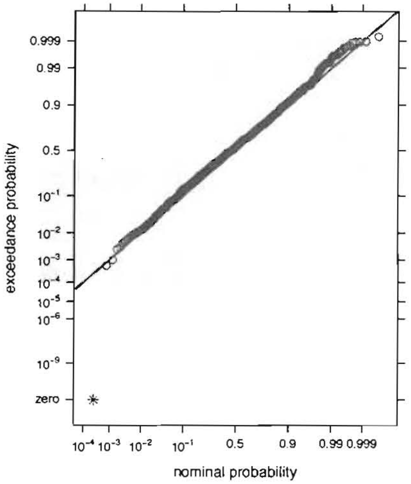
P-Values for Joint Exceedance

Three weeks of monitoring SSH graphs every ten-minutes.



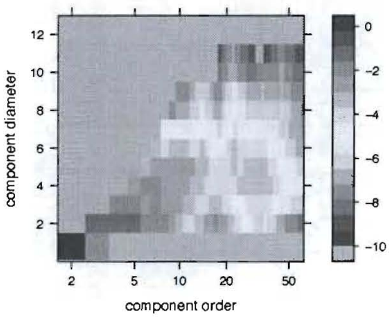
Distribution of P-Values for Joint Exceedance

A QQ plot shows we have a reasonable reference model for monitoring.

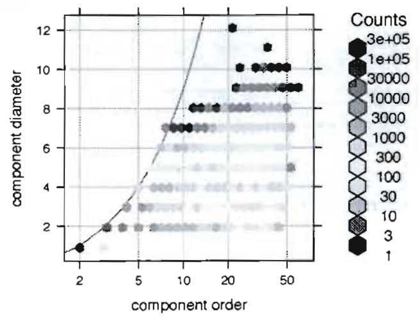


The Anomaly: Compare $\hat{p}_t(\mathbf{x})$ to Data

Log$_{10}$ estimated probabilities



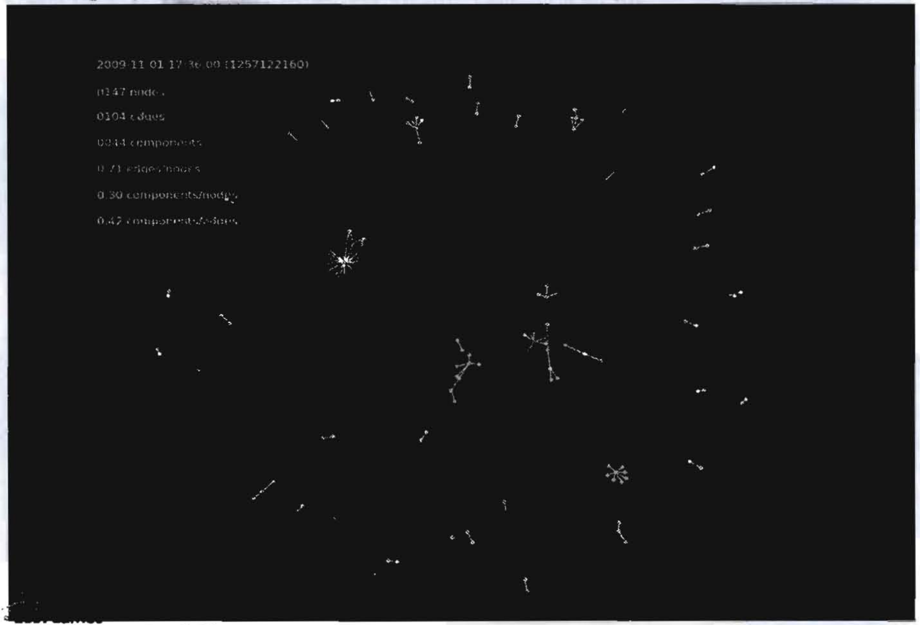
Histogram for 4 to 8 PM



One subgraph with (O,D) = (21, 12) is highly unusual at this time of day (but not at other times).

More broadly distributed anomalies can also be detected.

SSH graph from LANL's Network During the Anomaly



2009-11-01 17:36:00 (1257122160)

0147 nodes

0104 edges

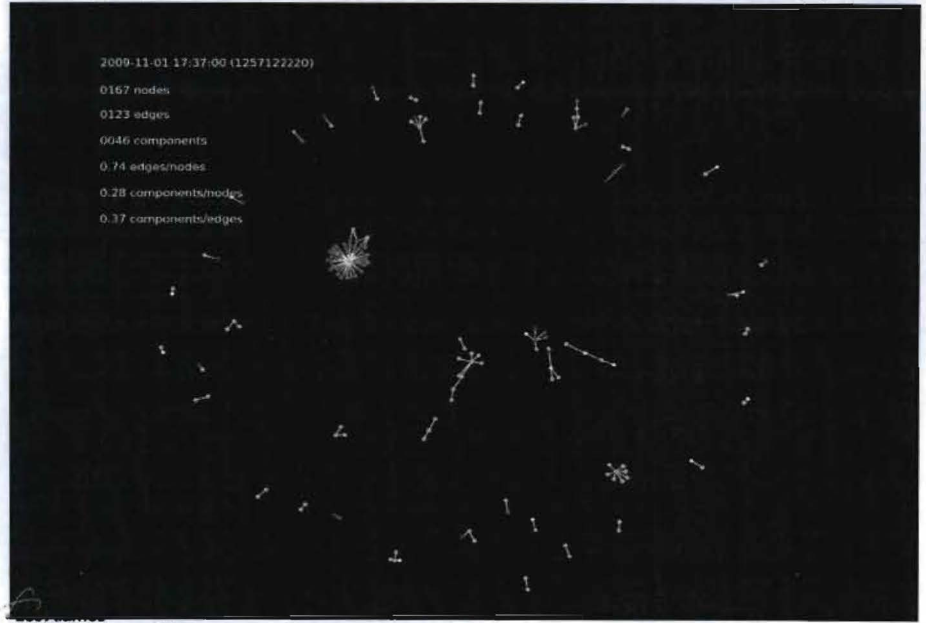
0044 components

0.71 edges/nodes

0.30 components/nodes

0.42 components/edges

SSH graph from LANL's Network During the Anomaly



2009-11-01 17:37:00 (1257122220)

0167 nodes

0123 edges

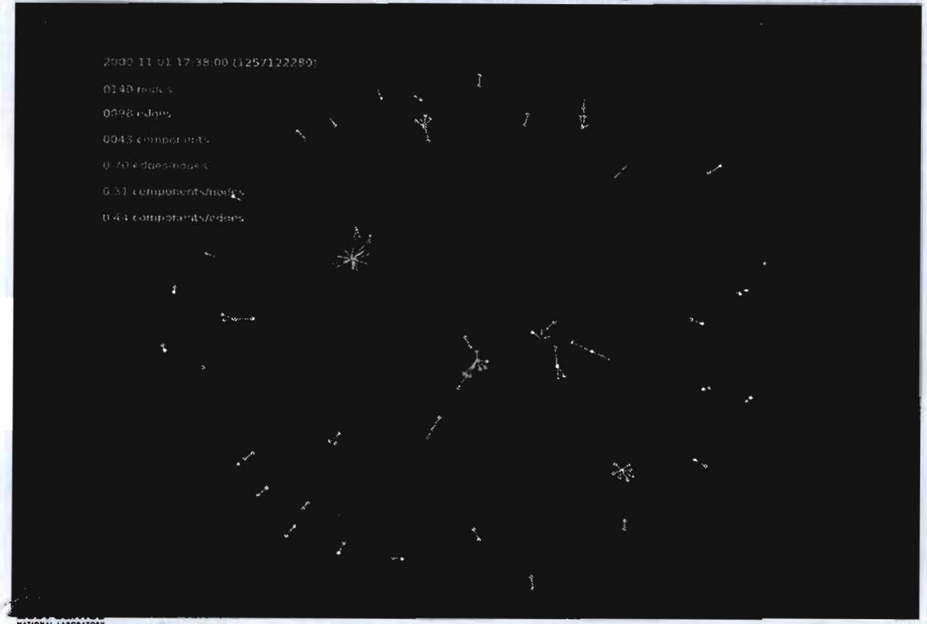
0046 components

0.74 edges/nodes

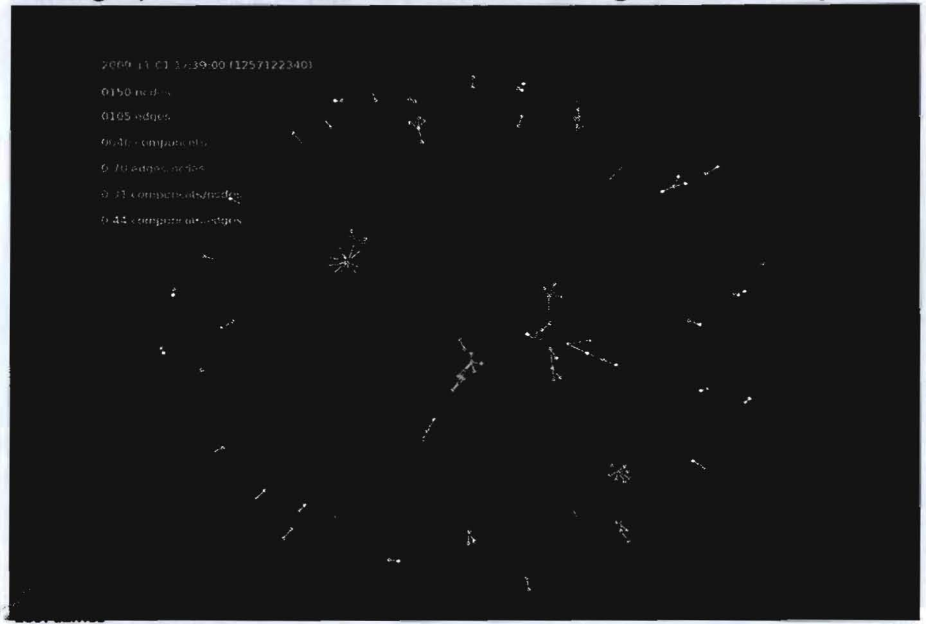
0.28 components/nodes

0.37 components/edges

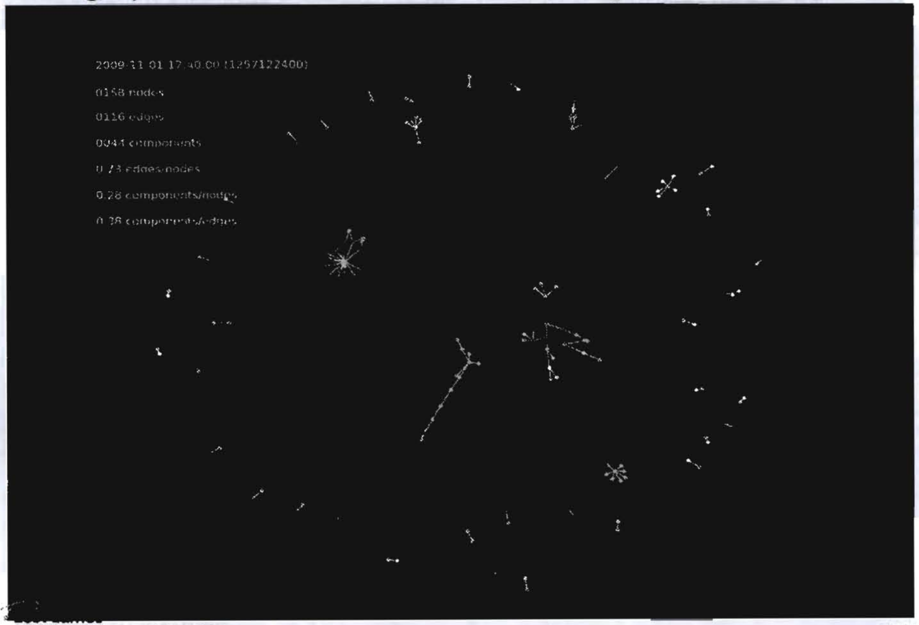
SSH graph from LANL's Network During the Anomaly



SSH graph from LANL's Network During the Anomaly

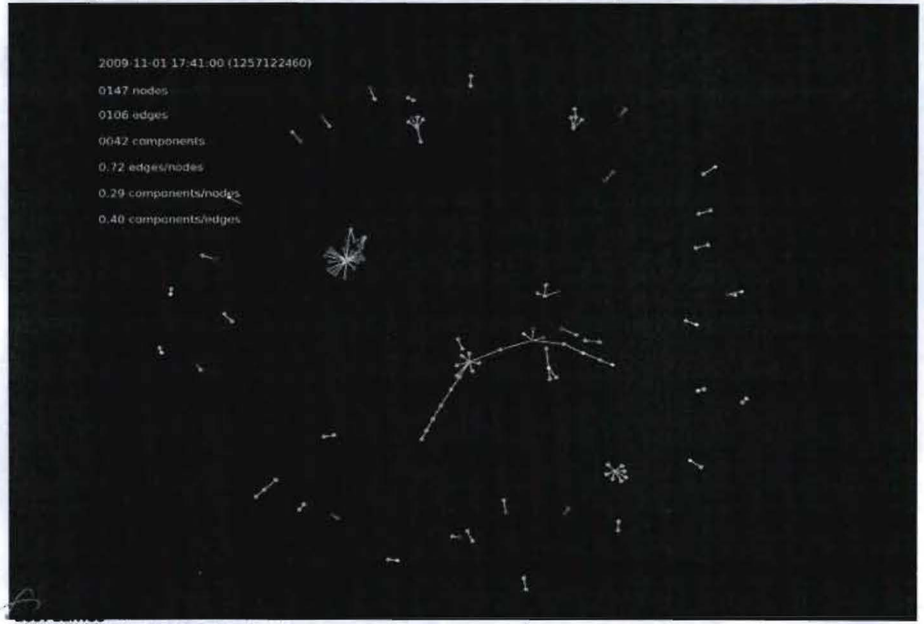


SSH graph from LANL's Network During the Anomaly



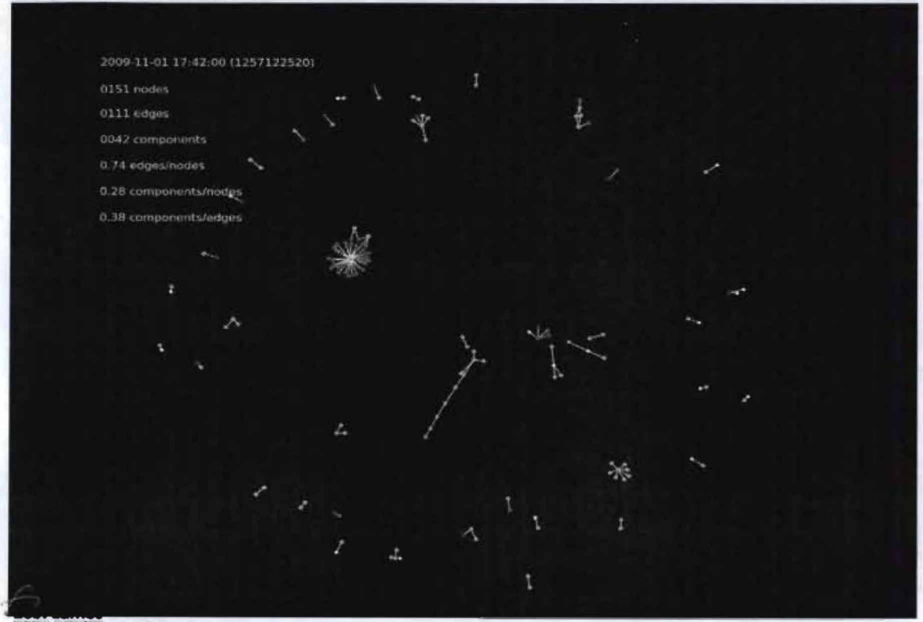
```
2009-11-01 17:40:00 (1257122400)
0158 nodes
0116 edges
0044 components
0.73 edges/nodes
0.28 components/nodes
0.38 components/edges
```

SSH graph from LANL's Network During the Anomaly



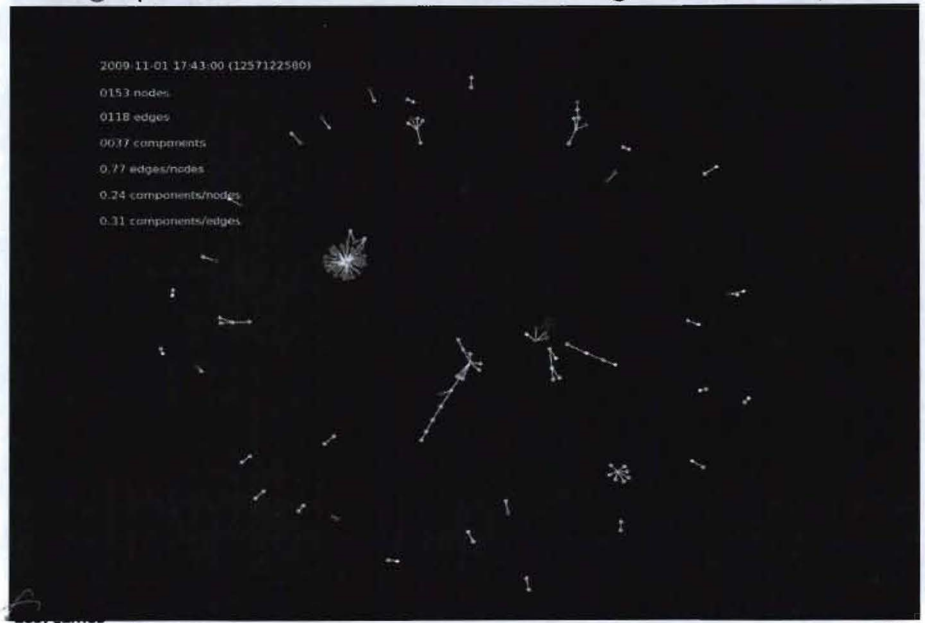
```
2009-11-01 17:41:00 (1257122460)
0147 nodes
0106 edges
0042 components
0.72 edges/nodes
0.29 components/nodes
0.40 components/edges
```

SSH graph from LANL's Network During the Anomaly



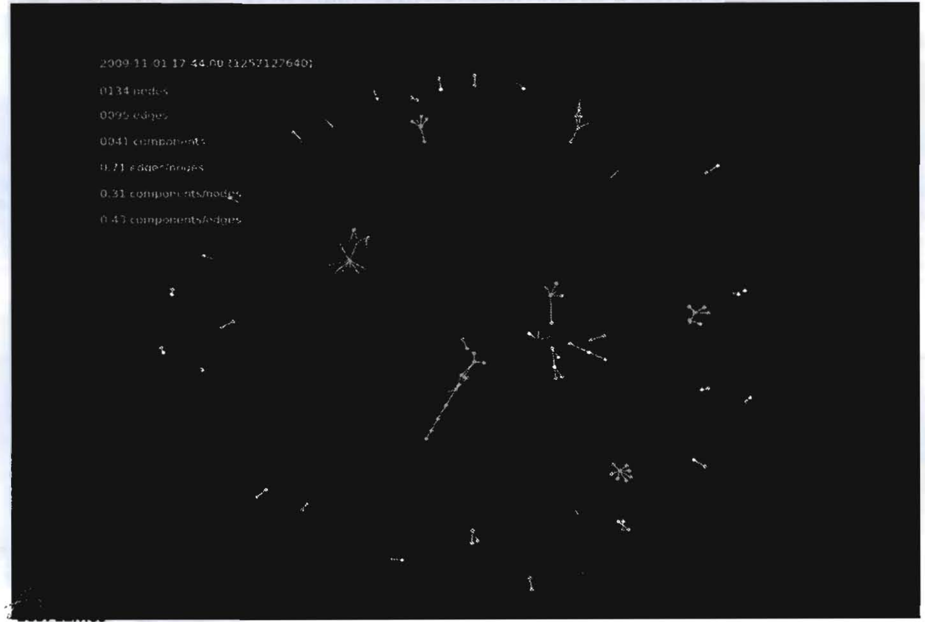
```
2009-11-01 17:42:00 (1257122520)
0151 nodes
0111 edges
0042 components
0.74 edges/nodes
0.28 components/nodes
0.38 components/edges
```

SSH graph from LANL's Network During the Anomaly



```
2009-11-01 17:43:00 (1257122580)
0153 nodes
0118 edges
0037 components
0.77 edges/nodes
0.24 components/nodes
0.31 components/edges
```

SSH graph from LANL's Network During the Anomaly



2009-11-01 17:44:00 (1257122640)

0134 nodes

0095 edges

0041 components

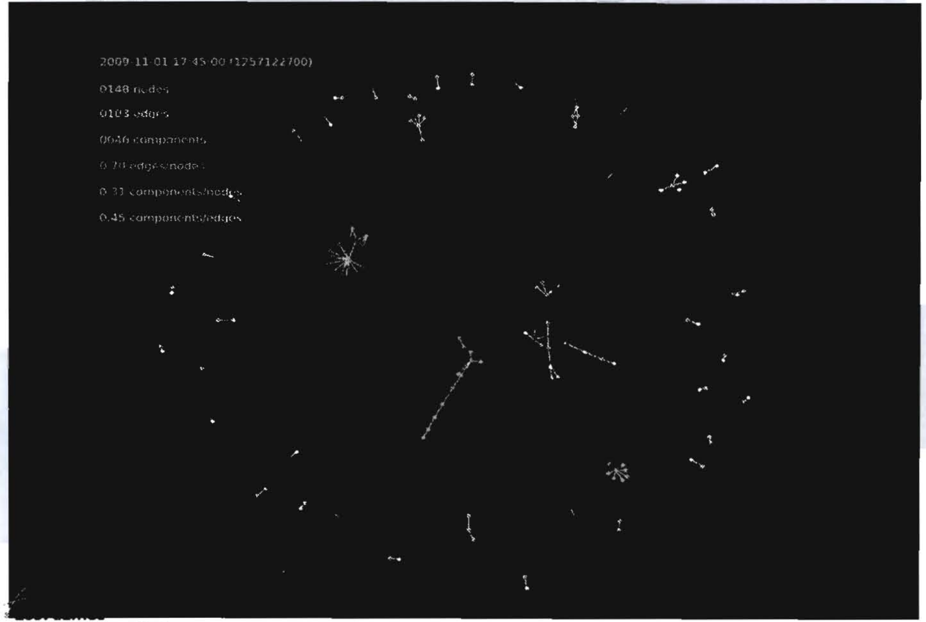
0.71 edges/nodes

0.31 components/nodes

0.43 components/edges

[redacted]

SSH graph from LANL's Network During the Anomaly



2009-11-01 17:45:00 (1257122700)

0148 nodes

0103 edges

0046 components

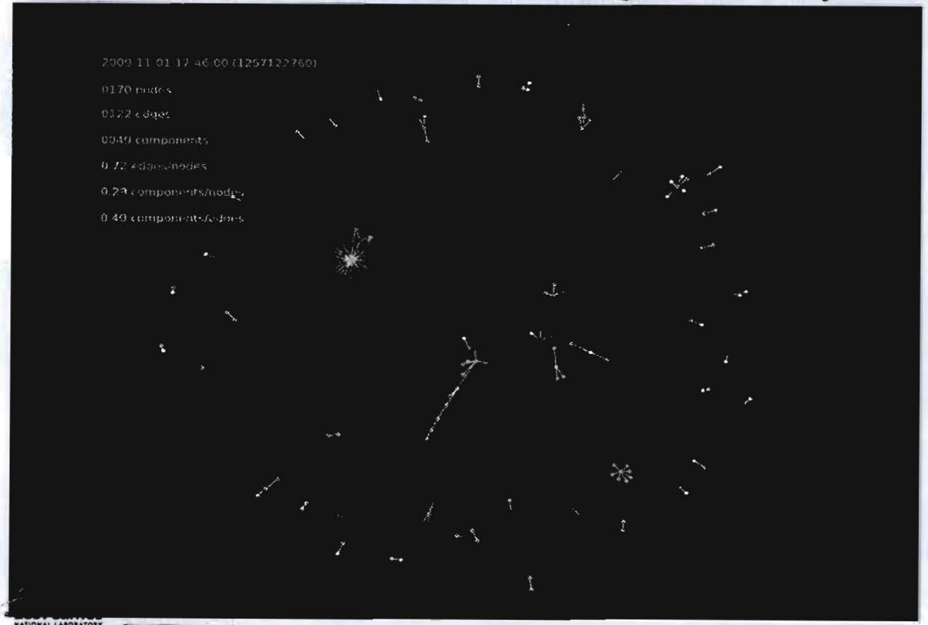
0.70 edges/nodes

0.31 components/nodes

0.45 components/edges

[redacted]

SSH graph from LANL's Network During the Anomaly



```
2009 11 01 17 46:00 (1257122760)
0170 nodes
0122 edges
0049 components
0.72 edges/nodes
0.29 components/nodes
0.40 components/edges
```

Conclusions & Further Work

- Monitoring subgraph properties can signal changes in network behavior.
- The dynamic kernel filter effectively smooths data from recent times in the target day from other recent days.
- The approach can be extended to incorporate
  - additional subgraph properties (eg., number of edges)
  - time constraints on paths (eg., telescoping times)
  - local communication patterns (eg., busy edges vs. quiet ones)
  - additional services (SSH is only illustrative)