Intelligent Network Management
Using Graph Differential
Anomaly Visualization

Qi Liao

Department of Computer Science
Central Michigan University
Network Management

What is going on in the network?

Internet

Public servers

DMZ

Private servers

Applications

Data

Wireless Users

Wired Users

Enterprise
Security Management

- Needs of Network Manager
  - Health check
  - Situation awareness
  - Accountability / Forensics
  - Troubleshoot

- Challenges
  - Huge amount of data
  - Complexity
  - Dynamics
  - Gap: daily monitoring ↔ operational interpretation
Network Anomaly

- Network anomaly is useful in many areas of network management.
- Some examples of “easy” anomalies:
  - Readings from sensor network
  - DoS attack
  - Port scanning
  - Packet headers match a pattern
- More *general* (harder) anomalies:
  - Stealthy
  - Less traffic
  - Given only a time-series of network graphs, can we detect abnormal changes and find the underlying causes?
Graph Diff. Anomaly Visualization

My network at time $i$  

Spatial anomalies

My network at time $j$

How similar / different?

Temporal anomalies
Differential Anomaly Visualization

- Graph *differential anomaly visualization (DAV)* framework
  - Whole graphs
  - Nodes and edges
  - Communities (subgraphs)
  - More tolerant to the *dynamics* of network.
- Effectively visualizes the *dynamics* and *abnormal changes* among the heterogeneous, time-series network graphs.
Monitoring Where, Who, and What

- Need finer granularity than raw network connectivity
- Two important enterprise network components
  - **Who** (users) are responsible
  - **What** (applications) are running on the network.

- **CONTENT** vs. **CONTEXT**
  - Associated with each network connection
  - Users, applications, parameters, file accesses, etc.
Local Context

Bigger picture: what is happening on the network

4/17/2012

Department of Computer Science
Michigan University

Central

8
Most existing tools show this view
Web traffic in, web traffic out, DNS, Active Directory
Network flows – Who and what?

Network Context Graphs

R3208.orange.fr

qliao

www.cmich.edu

lab01.cps.cmich.edu

rmcfall admin

firefox

IIS

apache

nessus
Data Collection Agent

- Gathers context from local hosts
  - **who** (users), **what** (applications), **when** (time), **where** (hosts)
- Built-in system tools (free and robust)
  - **who**, **where**
  - **what**
  - **who**, **what**, **where**
  - **when**
  - `netstat`, `ps`, `lsof`, `diff`

- Easy to deploy (no change to existing systems)
- Lightweight
  - CPU < 2%
  - Bandwidth (1000 hosts: 240 Kbps = 0.2% of 100Mbps)
  - Disk (1GB/host/year)
HUA Graph View

Graph controls

Monitored hosts

External Domains

Apps

Users

Sort by degrees, weights, names

Node selection
Bipartite graphs

- The general **HUA connectivity graphs** can be separated into *(multi-)*bipartite graphs.

<table>
<thead>
<tr>
<th>src host</th>
<th>dst host</th>
</tr>
</thead>
<tbody>
<tr>
<td>host:iss-node030.cse.nd.edu_L</td>
<td>domain:128.105.175.0_R</td>
</tr>
<tr>
<td>host:iss-node032.cse.nd.edu_L</td>
<td>host:iss-node007.cse.nd.edu_R</td>
</tr>
<tr>
<td>host:cclweb03.cse.nd.edu_L</td>
<td>domain:64.12.30.0_R</td>
</tr>
<tr>
<td>host:cclweb00.cse.nd.edu_L</td>
<td>host:loco27.cse.nd.edu_R</td>
</tr>
<tr>
<td>host:129.74.153.243_L</td>
<td>host:loco01.cse.nd.edu_R</td>
</tr>
<tr>
<td>host:cwrl-c0-15.cse.nd.edu_L</td>
<td>host:cwrl-c0-22.cse.nd.edu_R</td>
</tr>
<tr>
<td>host:cclscratch00.cse.nd.edu_L</td>
<td>host:bartok.helios.nd.edu_R</td>
</tr>
<tr>
<td>host:cclws00.cse.nd.edu_L</td>
<td>host:loco21.cse.nd.edu_R</td>
</tr>
<tr>
<td>host:classical.cselab.nd.edu_L</td>
<td>host:iss-node006.cse.nd.edu_R</td>
</tr>
<tr>
<td>host:chamber.cselab.nd.edu_L</td>
<td>domain:207.171.185.0_R</td>
</tr>
<tr>
<td>host:thermometer.cse.nd.edu_L</td>
<td>host:cclws03.cse.nd.edu_R</td>
</tr>
<tr>
<td>host:cclsun12.cse.nd.edu_L</td>
<td>domain:64.124.109.0_R</td>
</tr>
<tr>
<td>host:cwrl-c0-1.cse.nd.edu_L</td>
<td>domain:141.161.133.0_R</td>
</tr>
<tr>
<td>host:129.74.154.230_L</td>
<td>host:cwrl-c0-9.cse.nd.edu_R</td>
</tr>
<tr>
<td>host:cwrl-c0-2.cse.nd.edu_L</td>
<td>domain:205.188.211.0_R</td>
</tr>
<tr>
<td>host:sc0-03.cse.nd.edu_L</td>
<td>host:msvpn-p1.cc.nd.edu_R</td>
</tr>
<tr>
<td>host:sc0-04.cse.nd.edu_L</td>
<td>host:styx.cse.nd.edu_R</td>
</tr>
<tr>
<td>host:cse-ibm-02.cse.nd.edu_L</td>
<td>host:confucius.helios.nd.edu_R</td>
</tr>
</tbody>
</table>
K-partite graphs

Quadripartite graph

Info-gain

Critical path

Hosts
- host: cclsun08.cse.nd.edu_L
- host: 129.74.153.206_L
- host: jupiter.cse.nd.edu_L
- host: 129.74.153.243_L
- host: vault.cse.nd.edu_L
- host: 129.74.154.204_L
- host: cclweb02.cse.nd.edu_L
- host: 129.74.154.253_L
- host: cclweb03.cse.nd.edu_L
- host: bender-wire.cse.nd.edu_L
- host: coldb.cse.nd.edu_L
- host: saturn.cse.nd.edu_L
- host: cclscratch00.cse.nd.edu_L
- host: bootleg.cselab.nd.edu_L
- host: cclscratch01.cse.nd.edu_L
- host: cclscratch02.cse.nd.edu_L
- host: cclsun00.cse.nd.edu_L
- host: cclsun01.cse.nd.edu_L
- host: cclsun02.cse.nd.edu_L
- host: cclsun03.cse.nd.edu_L
- host: cclsun04.cse.nd.edu_L
- host: cclsun05.cse.nd.edu_L
- host: cclsun06.cse.nd.edu_L
- host: cclsun07.cse.nd.edu_L

Users
- user: slu5
- user: usr25
- user: mallaspa
- user: glmsey
- user: usr33
- user: tracdar
- user: romariti
- user: cc1
- user: mabrec2
- user: ychen12
- user: usr32
- user: condor
- user: dchen
- user: diclesiak
- user: usr99
- user: hwang6
- user: jkjackale
- user: thromens
- user: pbui
- user: mcracker
- user: mmcmier
- user: mnelson3
- user: ovandak

Applications
- app: MATLAB
- app: acoread
- app: amandad
- app: bash
- app: firefox-bin
- app: bonobo-activation-server
- app: java
- app: catalog_save
- app: gaim
- app: gzip
- app: ssh
- app: weather-applet
- app: chim_server
- app: clock-applet
- app: condor
- app: python
- app: httpd
- app: parrot
- app: sendmail
- app: condor_negot
- app: condor_green
- app: condor_q
- app: condor_schedd
- app: condor_shadow
Similarity Graphs (app)

Local users (root)

Ent. users (condor)

Parrot

java

condor_shadow

applications

# users
Visual Analysis for Network Management

Data mining / machine learning

- Automatic
- Algorithmic, analytic methods

Visualization

- Manual
- Interactive visual exploration
- Bring in domain knowledge from experienced managers.

4/17/2012 Department of Computer Science Central Michigan University
Differential Anomaly Visualization

- What are the changes?
- What are the variance and invariance?
- How similar (different) from day-to-day network activities?
- What changes are normal / abnormal?
- How to quantify and visualize the evolution of changes?

Dynamic and noisy data (hosts, users, applications)

Differential Visualization

Insights (variants, invariants, abnormal behaviors, root causes ...)

4/17/2012
Hierarchical DAV

(overview + context)

Whole Graphs

Nodes / Edges

Communities
Graph Diff. Anomaly Visualization

My network at time $i$

My network at time $j$

Spatial anomalies

How similar / different?

Temporal anomalies
Graph Properties

Graph sizes

Cluster coefficients

Graph diameters

Degree distributions

Graph distances

Graph variance scores
Graph Similarity

- General graph *isomorphism*

A more complex example
Graph distance

- **Edit distance**: number of operations required to transform one into the other.
- **Graph Edit Distance** (GED) [Bunke07] to measure the graphs’ similarities.
- Maximum common subgraphs (MCS) based:

\[
d(g_1, g_2) = 1 - \frac{|mcs(g_1, g_2)|}{\max(|g_1|, |g_2|)}
\]

- Graph edit distance (GED) based:

\[
d(g_1, g_2) = \frac{|g_1| + |g_2| - 2|mcs(g_1, g_2)|}{|g_1| + |g_2|}
\]
Minimum common supergraphs (*MCP*)

Maximum common subgraphs (*MCS*)

Median Graph (*MG*)
Differential visualization

New (appear)

Old (disappear)

Spatio-temporal dynamics

Invariance

Show / Hide
Differential visualization

Old (disappear) + Invariance
Link Anomalies

- Not exactly *link prediction* problem.
  - Common neighbors assumption
  - Known nodes only assumption
  - Non-dynamic assumption
- Proof-of-concept
  - Non-linear weighting frequency function

\[
P(L_i) = \frac{\sum_{t=1}^{N} w(t) \cdot d_{i,t}}{\sum_{t=1}^{N} w(t)}, \quad d_{i,t} \in \{0,1\}
\]

- Probability of \(i\)-th link to appear
- Whether \(i\)-th link appears at time \(t\)
- Non-linear time weighting function

- Can take inputs from future link anomaly algorithms
Link Anomalies Visualization

RED: Type-I anomaly: should appear but did not appear
BLUE: Type-II anomaly: should not appear but appeared
Link Anomalies Visualization
Link Anomalies Visualization

Should appear
Community-based DAV

- **Intermediate** similarity metric

**Community membership changes**

- **COARSE**
  - Graph property changes
- **FINE**
  - Node / edge changes

Susceptible to the dynamics of graphs

**Balance** of granularity and complexity
Intra-graph clusters visualization

1) firefox
2) httpd web
3) desk apps
4) Condor research computing

Walktrap
[Pons:2006]
Community-based DAV

- Graphs changes via community similarity
  - Similar to Rand Index [Rand71]

\[
dist(C_1, C_2) = 1 - \frac{SS + DD}{SS + SD + DD + DS}
\]

- Flexibility
- Suitability for highly dynamic networks

Nodes consistently belong to the same (or different) communities

changes are normal
Community-based DAV (example)

Anomaly caused by a spike of community changes at time 8 and 9

Walktrap
Community-based DAV (MDS view)

Nodes that are farther away indicate *anomalous* user behaviors.

Graph/communities
Communities of a User Similarity Graph

Grad students community

Condor community

Time: 8
Communities of a User Similarity Graph

**Grad students community**

**Time: 9**

**Users change community membership**

**Condor community**
Conclusion

- Network (security) management is hard.
  - Large scale, heterogeneity, dynamics, complexity
- Anomaly detection and analysis is important yet challenging.
- We developed a novel hierarchical graph differential anomaly visualization (DAV) framework
  - Combines automated graph data mining and manual exploration.
  - At different levels: Graphs, Nodes/Edges, Communities
- Completeness
  - Overview vs. Details-on-demand
  - Exact changes vs. Dynamic churns
  - Detection vs. root causes
- DAV: intelligent, time-efficient management alternative.
More info visit http://cps.cmich.edu/liao1q

Thank You!
Questions

System Messages:

ERROR: file C:\\\NetBeans\\\Lockdown\\\GUI_DATA\graphHT\1_1_2009--1_17_2010_HUA\HH_dg_w_1263186000.attr is invalid.
java.lang.NullPointerException
Read C:\\\NetBeans\\\Lockdown\\\GUI_DATA\graphHT\1_1_2009--1_17_2010_HUA\HH_dg_w_1263186000.ght.
ID: 1_HH_dg_w_1263186000: G=(V,E) directed weighted graph, |V|=478 |E|=1556. No cluster.
Total graphs read: 2
Inferred START/END time range: Sun Jan 10 00:00:00 EST 2010 -- Mon Jan 11 00:00:00 EST 2010
(Graph --> xml): Wrote GUI_DATA\graphML\1_HH_dg_w_1263099600.xml
Prefuse graph created: colorByNodeTypes_animatedView_ID: 0_HH_dg_w_1263099600: G=(V,E) directed weighted graph, |V|=373 |E|=1117. No cluster.
Thread(Plot graph): Total processing time 0 seconds.

(Graph --> xml): Wrote GUI_DATA\graphML\MinCommonSupgraph_0_HH_dg_w_1263099600_clone -- 1_HH_dg_w_1263186000_clone.xml
Prefuse graph created: colorByNodeTypes_animatedView_ID: MinCommonSupgraph 0_HH_dg_w_1263099600_clone -- 1_HH_dg_w_1263186000_clone: G=(V,E) directed weighted graph, |V|=530 |E|=1740. No cluster.
Thread(Plot graph): Total processing time 5 seconds.

(Graph --> xml): Wrote GUI_DATA\graphML\MinCommonSupgraph_0_HH_dg_w_1263099600_clone -- 1_HH_dg_w_1263186000_clone.xml
Prefuse graph created: colorByNodeTypes_animatedView_ID: MinCommonSupgraph 0_HH_dg_w_1263099600_clone -- 1_HH_dg_w_1263186000_clone: G=(V,E) directed weighted graph, |V|=530 |E|=1740. No cluster.
Thread(Plot graph): Total processing time 2 seconds.