ELISHA: A Visual-Based Anomaly Detection System for the BGP Routing Protocol

S.T. Teoh, K.L. Ma, S.F. Wu
Computer Science Department
University of California, Davis
Davis, USA
{teoh, ma, wu}@cs.ucdavis.edu

D. Pei, L.Wang, L. Zhang
Computer Science Department
UCLA
Los Angeles, USA
{peidan, lanw, lixia}@cs.ucla.edu

A. Mankin, D. Massey, X. Zhao
Networking Group – East (NGE)
USC/ISI
Arlington, USA
{mankin, masseyd, xzhao}@isi.edu

R. Bush
Networking
AT&T
Bainbridge Island, USA
randy@psg.com

Abstract — ELISHA is a human-interactive visual-based anomaly detection system for handling faults and security attacks on the BGP (Border Gateway Protocol) routing protocol. A "fully automated" anomaly detection system for analyzing and correlating unknown attacks or faults is hard to build due to the consideration of effectiveness, coverage, and false positive. In this paper, we demonstrate that the ELISHA system can utilize human intelligence to effectively resolve two critical tasks related to the BGP MOAS (Multiple Origin AS) problem: (1) MOAS anomaly detection and analysis, (2) MOAS event correlation.

Keywords— BGP, MOAS, Information Visualization, Anomaly Detection, Event Correlation

I. INTRODUCTION

With the popularity of the Internet technology, unintentional faults and intentional intrusions directly on network protocols, such as routing protocols, have become a serious threat to our Internet-connected society. In 1997, a buggy Bay-Network BGP (Border Gateway Protocol) [2] router falsely de-aggregated thousands of network addresses which disabled the whole east coast Internet for up to 12 hours. In the same year, an implementation bug in OSPF (Open Shortest Path First) routing protocol [3] in most (if not all) commercial routers was discovered and it was also demonstrated how to utilize this bug to maliciously control an OSPF network [15,16,27] for one hour with only one single attack packet. Then, starting in 2000, different types of DDoS (Distributed Denial of Services) attacks were launched by "slaves" from unknown locations around the whole Internet. In the middle of 2001, worm attacks such as CodeRed and Nimda were spread around the Internet. The surprising observation was that these two worm instances affected not only the victim web servers, but also, possibly, the BGP protocol stability due to a report from Renesys [4].

Driven by these faulty or intrusive instances on the Internet infrastructure, many research teams have studied how to either design new protocols and/or enhance existing protocols such that the Internet can be more robust and fault/intrusion-tolerant. For instance, the Secure BGP (S-BGP) [5] protocol utilizes the PKI infrastructure to authenticate and authorize the route update messages in an inter-domain environment. A network protocol scrubber [6] or normalizer [7] enforces the protocol operations to strictly follow the "correct interpretation" of the original protocol specification. Self-stabilization network protocols have the capability to both locally detect certain incorrect behaviors and globally reset the network to a stable state automatically [28]. Very recently, the FNIISC team (UCLA, USC/ISI, AT&T, NCSU, and UCDavis) is investigating the possibility to statistically profile the "stable" BGP paths leading to critical DNS servers (i.e., root and gTLD) and to filter out "unusual" routes potentially being falsely injected by attackers or simply misbehaving BGP routers.

Another complementary approach to handle these Internet vulnerabilities is via ***an interactive process*** between network administrators/operators and network management systems with visualized network information. We believe that, at least in the short term, a network system with machine intelligence alone will have certain limitations in detecting and responding to novel attacks/faults targeting on the Internet infrastructure itself. For instance, given a colorful image of some BGP routing data, an experienced human operator can discover numerous facts about the Internet instantly, while an analysis program must already have the mechanisms built in to achieve the same results. One of the most difficult tasks in intrusion detection is "event correlation," but via human visualization, this task might be much more plausible. Furthermore, due to the complexity and size of the Internet, it is already a very difficult task to evaluate Internet's "health" condition and clearly identify the root causes of some observed symptoms. Without a comprehensive understanding of the Internet, it is not certain that some new Internet protocols, architecture, or enhancements will be effective in responding to the problems we have today.

In this paper, we describe our design, implement, and evaluate an "experimental" visual anomaly detection system "ELISHA", which will allow a network operator to interactively monitor and diagnose the network. Our goal is to build an intrusion detection system combining both machine and human intelligence to support an “interactive” analysis and correlation process.

II. Related Works: Visualization of Network Data

Information visualization [12.25.26] has emerged as one of the most active areas of computer science research in recent years due to the explosive growth of the World Wide Web. In contrast to scientific visualization, which is primarily about transforming numerical data defining physical structures or phenomena in three spatial dimensions into pictures, information visualization generally maps very large amount of textual, symbolic, or relational data into spatial forms that can be displayed graphically. Data visualization exploits the human vision system to help us explore and better communicate with others particular aspects of the data under study.

For instance, the SeeNet system [9,10] from AT&T visualizes important network performance parameters such as the sizes of flows, link capacities, link/node utilization, and various timing information. The network dynamic information is graphically displayed and animated such that the human operators can answer questions such as whether some parts of the network have been damaged during a disaster event. Furthermore, SeeNet has a set of tunable parameters to allow the users to dynamically focus on different aspects of the data. Similarly, the SUNY-Albany and U. of Idaho team [17] developed a visual intrusion detection system to identify malicious behaviors such as illegal login attempts and port probing. Also, Fortier and Shombert from Averstar built a system to visualize firewall log files to detect attacks in the application layer such as SMTP or WWW. Other visual-based systems for network, telecommunication, and security can be found in [11,18,19,20,21,22,23].

Instead of visualizing the performance data under a whole network system in SeeNet, in this paper, for the purpose of fault/intrusion detection and analysis, we focus on visualizing the behaviors of individual network protocols such as the BGP routing protocol in depth. A fundamental contribution here is that, while a system like SeeNet can identify a particular link is congested, our work will have the capability to move further into the information related to the congestion. More specifically, our system will potentially help a human operator to quickly further nail down exactly which network protocols are responsible to the congestion and/or which network entities are possibly misbehaving under the protocol specification.

The Cichild system from NLANR (National Laboratory of Applied Network Research) is yet another very interesting network data visualization system. In Cichild, raw packet data are represented graphically via different methods such as Bar charts and Vertex-Edge graphs in 2D and 3D. The NAM (Network Animator) system [24] from USC/ISI and UCLA provides a 2D-only graphical interface to network traces under a NS2 environment. Similar to SeeNet, Cichild and NAM are both designed to study network performance (TCP and wireless, e.g.), and has not focused on analyzing and correlating events for the purpose of intrusions and faults. We believe that it is critical not only to display the information graphically but also to consider the relationship between the graphical representation and the systematic and interactive process for network attack and fault analysis.

III. Visualization Design

A. *Graph Drawing and Visualization*

A carefully designed visual representation (e.g., [25.26]) of data can help us perform more effective anomaly analysis on network protocols. However, because of the intrinsic complexity of the data collected from the Internet, deriving a spatial mapping for the data that is cognitively useful for anomaly analysis is a considerable challenge. We determine that studying network protocol data is fundamental to anomaly analysis. The visualization component of the proposed research will focus on the development of appropriate 2D and 3D graph drawing and visualization technology for each individual network protocol and possibly their correlation. Other visual metaphors that are complementary to graph visualization will also be investigated.

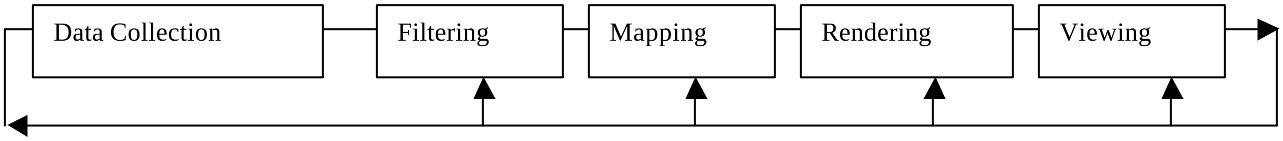
Graph is one of the most fundamental data structures in computer science. Research in graph drawing has been conducted in several diverse areas, including graph theory, layout algorithms, and user-graph interaction. Previous research efforts were mainly made to improve clarity of 2D graph while the development of 3D graph drawing algorithms has been rather sparse. We intend to develop data mapping and graph drawing algorithms that meet the following requirements:

a) Interactive navigation
b) Focus with context
c) Information augmentation
d) Scalability

Existing graph visualization methods are lacking one or more of the above capabilities. Interactivity is the key to effective data visualization. Our visual-based analysis system must be highly interactive. If computational and storage requirements would prohibit from interactive viewing, an incremental, hierarchical navigation capability should be offered to the user. When displaying a graph, the limited computer screen space forces us to be selective on the level and extent of the graph displayed. The key is to maximize the utilization of the screen space available to draw the viewer’s attention to a particular aspect of the graph while providing enough context information to direct further navigation. A graph representation allows us to see topology information. Frequently, it is helpful to also augment the graph visualization with other information such as activities between individual nodes. In particular, this information augmentation can help perceive correlation in both spatial and temporal domains. Finally, a scalable visualization solution is required to cope with the data volume, the complex relationships, the limited screen space, and limited computing resources.

B. *Integration of Visual-based Analysis and Correlation*

The graph representation of the data is displayed through the proposed user interface. The user interacts with graphs to reveal suspicious patterns. The process of visualization is an inherently iterative one consisting of multiple steps. A standard visualization process is depicted in the following figure:



In this research, we pay special attention to the filtering and mapping steps, which prepare the collected data for rendering and viewing. Generally, the filtering step:

a) Removes "noise" from the raw data,
b) Reduces the data to a more manageable size, or
c) Enhances particular aspects of the data.

In the context of anomaly analysis, its job is to extract and organize particular aspects of the data (e.g., the Origin AS changes in BGP routing) for the subsequent steps. The mapping step transforms the filtered data into a collection of graphics entities with appropriate properties (e.g., colors, transparency, and texture) for rendering.

The data collection step, if done continuously in a real-time monitoring setting, can present tremendous research challenges in data management and retrieval, which are beyond the scope of this proposal. We thus choose to mainly address the theoretical formulation of measuring schemes, data abstraction and encoding, scalable visual representations, and interaction mechanisms required to establish effective visual-based evaluation and response strategies.

To summarize, our proposed approach has the following four advantages. First, the ELISHA system integrates the capability of cognitive pattern matching. Second, the visual/graphical information from ELISHA triggers the human's intelligence and memory to reason and analyze the observed situation. A human operator will obtain experience via this process and be trained to more accurately identify the problems quickly. Third, the interactive monitoring and analysis process provides a feedback loop from the human back into the ELISHA system. Finally, the human expert can provide an in depth explanation about the potential problems using the annotated images and/or animations derived. The template is designed so that author affiliations are not repeated each time for multiple authors of the same affiliation. This template was designed for two affiliations.

IV. BACKGROUND: THE MOAS CONFLICTS IN BGP

In the following sections, we will use the BGP MOAS conflicts as an example to convey our initial design and prototype of ELISHA. Therefore, in this section, we briefly describe what MOAS is.

The Internet is made of thousands of Autonomous Systems (ASes), loosely defined as a connected group of one or more IP prefixes which have a single and clearly defined routing policy [1]. BGP (Border Gateway Protocol) [2] is the standard inter-AS routing protocol. A BGP route lists a particular prefix (destination) and the path of ASes used to reach that prefix. The last AS in an AS path should be the origin of the BGP routes. A **M**ultiple **O**rigin **A**utonomous **S**ystem (**MOAS**) conflict occurs if a prefix appears to originate from more than one ASes. More precisely, suppose prefix $\boldsymbol{d}$ is associated with AS paths $\boldsymbol{asp}_1 = (\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_n)$ and $\boldsymbol{asp}_2 = (\boldsymbol{q}_1,\boldsymbol{q}_2, \ldots, \boldsymbol{q}_m)$. We say a MOAS conflict occurs if $\boldsymbol{p}_n <> \boldsymbol{q}_m$.

A MOAS conflict is either valid or invalid. A MOAS conflict is valid if each originating AS can directly reach the prefix. On the other hand, if one of the origin ASes cannot reach the prefix, then we say it is an invalid MOAS conflict. The problem of detecting invalid origins is complicated by BGP operational practices. RFC1930 [1] recommends that each prefix originates from a single AS, but this is not a requirement. Valid operational policies can cause a single prefix to originate from multiple Autonomous Systems. In general, a BGP router has no way to determine whether a MOAS conflict is the result of a fault, an attack, or a legitimate operational policy.

Faulty aggregation or de-aggregation can cause MOAS conflicts. In faulty aggregation, an AS advertises an aggregated prefix, even though some of more specific prefixes are not reachable by the AS. A MOAS conflict occurs if this aggregated route is also generated by other ASes. Packets that use the faulty aggregated route will travel to the faulty AS and then may not be able to reach all the more specific prefixes. On the other hand, for example, On April 25th, 1997, a severe Internet outage occurred when one ISP falsely de-aggregated most of the Internet routing table and advertised the prefixes as if they originated from the faulty ISP. The falsely originated prefixes resulted in MOAS conflicts, which caused serious impacts on Internet routing.

From Geoff Huston's BGP Table Statistics website, starting on 2/18/2001, a daily count of MOAS conflicts has been tracked using data from some ISPs and from the Oregon Route Views Server. On 04/19/2001, the website switched to tracking MOAS conflicts on a bi-hourly instead of daily basis. In our joint work with UCLA and USC/ISI [13,14], MOAS conflict data was obtained from Internet routers and the raw data was analyzed, without any sophisticated visualization tools, based on the total number of conflicts, duration of the conflicts, and the prefix length. Both the number of MOAS conflicts and distribution of the duration of MOAS conflicts were different than what we anticipated.

The BGP route for a prefix (destination) includes an AS path. The last AS along the path to the prefix is considered to be the origin AS. We examined the AS paths that led to the same prefix but ended in different origin ASes. We primarily used data from the Oregon Route Views server to obtain the BGP routes and AS paths. Currently, the Oregon Route Views server peers with 54 BGP routers in 43 different ASes. Each peer exports its BGP routing table to the Route Views server.

The Oregon Route Views data is particularly attractive because it provides data from a number of different vantage points. The data obtained from a particular local point, such as in an individual ISP, may show a smaller number of MOAS conflicts since fewer potential AS paths may be visible at that point in the network. For example, at a randomly selected time, the Oregon Route Views server observed 1364 MOAS conflicts, but three other individual ISPs observed 30, 12, and 228 MOAS conflicts during the same period. This only means that fewer MOAS conflicts were visible to these ISPs and even the number of MOAS conflicts observed from the Oregon Route Views Server may underestimate the total number of MOAS conflicts.

To obtain a relatively complete view, we used “archived Oregon Route Views” data from both NLANR and PCH-net. NLANR archived the Oregon Route Views data on a daily basis from 11/08/1998 to 03/16/2001. PCH-net archived the Oregon Route Views data on a daily basis from 03/16/2001 to the present. The MOAS conflicts are identified by prefixes only no matter whether a MOAS conflict was conflicted by same set of origin ASes or the conflict was continuous. Overall 38225 conflicts were observed over 1279 days, and there is a significant increase from 683 conflicts in 1998 to 1294 conflicts in 2001 alone.

While many research scientists (including us) have studied and analyzed MOAS data collected from different sources, the conclusions we have drawn so far include:

a) Unusual amount of MOAS conflicts occurred occasionally, and we know some of these instances are due to faulty BGP routers.
b) In some instances, after carefully analyzing the raw data, we might identify which AS contributed most of the MOAS conflicts, but we do not know in general what types of faults they were.

V. THE DESIGN OF “ELISHA/MOAS”

We have built a prototype ELISHA system to support an interactive process for analyzing, as an example, the BGP MOAS (Multiple Origins Autonomous Systems) conflicts. Although the result here is preliminary, we have clearly seen some great advantages in using ELISHA to analyze and correlate network protocol information. For instance, while the system was initially designed to analyze the MOAS problem, we actually discovered network problems outside the scope of MOAS. In this section, we will first briefly describe the design of ELISHA. Then, we will show a few scenarios on using ELISHA to not only detect the problem but also quickly nail down the root cause to the detected problem.

A. Types of Origin AS Changes in BGP

From the raw BGP data collected, we can produce a set of “Origin AS Change (OASC)” events. Each Origin AS Change (OASC) event contains the following five attributes:

a) ***Prefix*** is the IP prefix whose Origin AS has changed.
b) ***AS-before*** is a list of the associated AS(es) before the change.
c) ***AS-after*** is a list of the associated AS(es) after the change.
d) ***Date*** is the date on which the change occurred.
e) ***Type*** is the type of a OASC change event.

Furthermore, Origin AS changes (OASCs) are classified into 4 main types and then further classified into 8 types in total. The 4 main classes are:

a) ***B***-type: An AS announces a more specific prefix out of a larger block it already owns. This might be OK due to possibly traffic engineering.
b) ***H***-type: An AS announces a more specific prefix out of a larger block belonging to another AS. In other words, this AS punches a hole on prefix addresses belonging to others.
c) ***C***-type: An AS announces a prefix previously owned by another AS.
d) ***O***-type: An AS announces a prefix previously not owned (and therefore owned by ICANN by default).

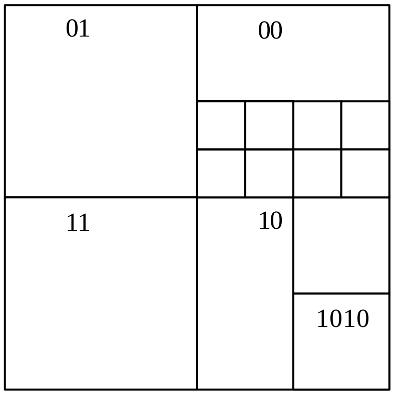
The C-type and O-type changes are further classified by whether they involve Single Origin AS (SOAS) or MOAS:

a) **CSM**: ***C***-type change from SOAS to MOAS
b) **CSS**: ***C***-type change from SOAS to SOAS
c) **CMS**: ***C***-type change from MOAS to MOAS
d) **CMM**: ***C***-type change from MOAS to MOAS
e) **OS**: ***O***-type change involving SOAS
f) **OM**: ***O***-type change involving MOAS
g) **H**: ***H***-type change always involving another AS (being punched a hole)
h) **B**: ***B***-type change always involving itself only

The color associated with each of the eight types will be used in interpreting the screen snapshots of the ELISHA prototype later.

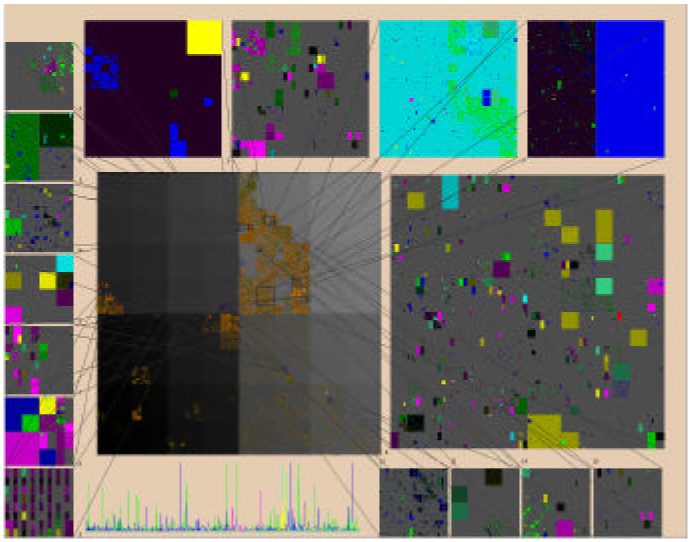
B. Representing IP Address Prefixes

In BGP/MOAS, 2 key concepts are IP address prefix and Autonomous systems. We will first describe our quad-tree representation of IP address prefixes.



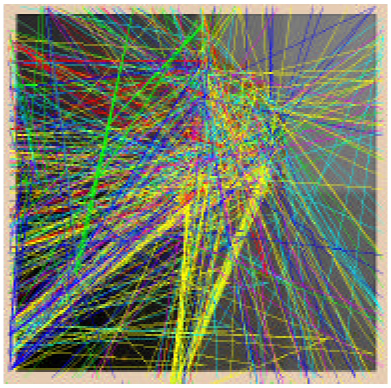
In our prototype, each IP prefix maps to one pixel on a square. The mapping is done in a traditional quad-tree manner as shown below. In a quad-tree, a square is repeatedly subdivided into 4 equal squares. In mapping a 32-bit prefix to a square, we start with the first two most significant bits of the address to place the IP address in one of the 4 squares in the second level of the quad-tree. We then use the next two most significant bits to place the IP prefix in the appropriate third level square within this square. We do this repeatedly until we can place the prefix in a square the size of a single pixel. The prefix is mapped to that pixel.

As an example, the following is the visualization of data for 416 days up till February 19, 2001. The main window shows the quad-tree mapping of the entire space of 32-bit IP address. A pixel is colored yellow if an Origin AS Change occurred on the current day (February 19, 2001), and colored brown to green if a change occurred on a previous day (January 1, 2000 through February 18, 2001). In the windows showing detail, a square is used to depict each change, with hue determined by the type of the change, brightness determined by how long ago the change occurred (present day data shown the brightest), and size determined by the mask of the prefix. The background of the main window is shaded according to the IP prefix the pixel represents. The brighter the pixel, the larger the IP prefix represented.



Due to the limitations of a computer screen space, we use a 512x512 pixel square to represent the entire 32-bit IP prefix space. With only 512x512 pixels, even though many IP prefixes map to the same pixel, we found that this is sufficient in spreading out the IP addresses in BGP/MOAS data. IP prefixes sharing similar more significant bits would be in close proximity on the screen. With an additional level of zooming into a portion of the data, we can view individual IP prefixes as shown above. In the detail windows, each IP prefix is shown as a square or a rectangle. The size of the rectangle indicates the size of the block of IP addresses; prefixes with a smaller mask get mapped to larger rectangles.

C. *Relationship Between Prefix and AS*



To represent the relationship between IP address prefix and different ASes, we place 4 lines surrounding the IP square, and an AS number is mapped to a pixel on one of the 4 lines. A line is then drawn from an IP address to an AS number if there is an Origin AS change involving that IP address and that AS number. This mapping takes advantage of the user's acute ability to recognize position, orientation and length. This figure shows the visualization of the IPAddrPrefix-AS relationship of Origin AS Changes of “a typical day” (April 5, 2001). The color of each line represents one of the eight different OASC types.

Since there are more AS numbers than pixels, more than one AS number maps to a pixel. Again, we provide zooming features for the user to differentiate between AS numbers, which map to the same pixel in the main display. The lines representing changes for the AS in focus is shown with brighter and more saturated colors than other changes. This effectively highlights the AS, fading the other changes into the background.

D. *Animation and Other Features*

For the time dimension, ELISHA shows one day's data at a time, and allows the user to animate the visualization (each frame showing consecutive day's data). With this ``movie" display, the user can detect temporal patterns. To assist our memory of patterns from previous days, we allow a user-defined window of a certain number of days prior to the currently shown date. Data from these previous days are displayed, but with darker, less saturated colors, so that the current day's data stands out.

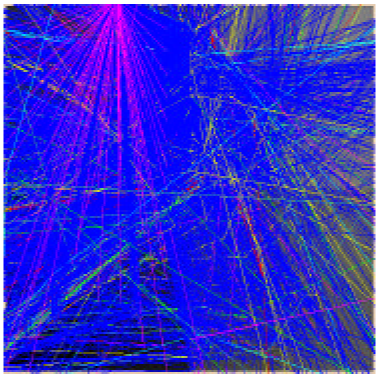
For the convenience of the user, we also provide textual display of the IP address or AS number represented by the pixel clicked by the user. Another feature for convenience is a slider bar to tell the date of the current data shown. The user can click on the bar to choose the date to show. A simple plot of the total number of changes of each type on each day is shown with the bar.

By choosing parameters like what IP prefixes to zoom in on, which AS numbers to focus on, which type of changes to view etc., the user follows an interactive process to navigate abstract information in different levels of details. Depending on the combination of chosen parameters, the user can see the overall pattern of the data, or the user can focus attention on very specific parts of the data. Different choices would reveal different anomalies and information.

VI. DETECTING AND ANALYZING ANOMALIES WITH “ELISHA/MOAS”

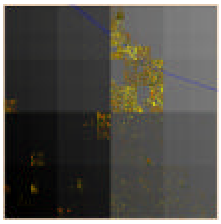
In this section, we present a few examples of using ELISHA/MOAS interactively to identify and analyze MOAS problems.

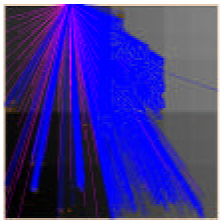
A. *Interactive Visual Analysis: “What went wrong on August 14, 2000?”*

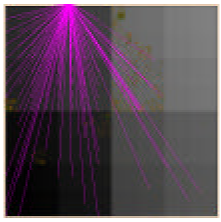


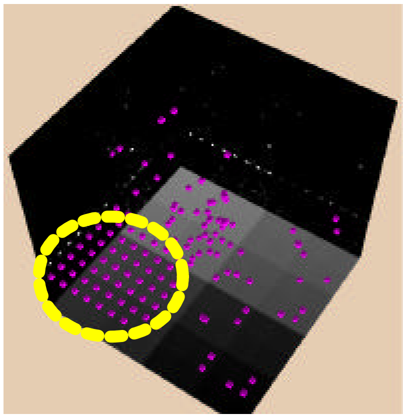
On August 14, 2000, when we turned on all eight OASC types and monitored on all ASes, we could tell something was obviously wrong from the following screenshot. Since the amount of “**blue**” was abnormally large, **our initial hypothesis** was that, due to configuration errors possibly, one or more ASes punched a large number of holes on the IP address prefixes belonging to other innocent ASes.

Immediately, to verify our hypothesis, we used the “AS detail” feature in ELISHA to select only one single H type OASC event and displayed the ASes being involved. The rationale is that, if a very small number of ASes are the root causes for the aggressive Hole-punching problem, selecting one of such event would lead us to one of the “trouble makers.” The figure below on the left shows that after we selected one AS randomly (AS-11724 in our example). In the same figure, the “**solid blue**” line connects the victim AS (AS-11724) and the prefix address (207.50.48.0/21) being hole-punched, while the “**dash blue**” line connects the attacking AS (AS-7777) and a subset of the prefix address (207.50.53.251/32). In other words, AS-7777 would attract all the traffic toward 207.50.53.251 from AS-11724, which supposed to be the true owner. Immediately, we know that the potential attacker (or faulty BGP router) was from AS-7777. Now, we can use the features in ELISHA to only select the OASC events related to AS-7777, and we have the middle snapshot. In fact, after focusing on AS-7777, we can easily validate via ELISHA that this AS was the only AS causing H type events on August 14, 2000. However, we also observed certain amount of OS type events (the “**pink**” links), and more interestingly, it seems to us that the pattern of the **pink** links was regularly distributed across a region of IP address prefixes that has never been used or allocated in Internet as in the right figure.







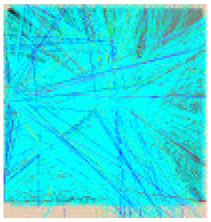


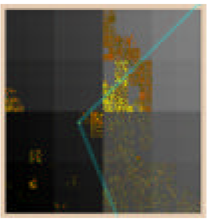
To validate further about what exactly is going on, we used our 3D representation to analyze all the **pink** links. With the left figure, from the left middle part (circled by a thick yellow dash line), it is very clear and interesting that AS-7777 announced prefix addresses forming a grid in the unused IP address space. Based on the location and shape of the Grid and the raw events, we concluded that AS-7777 falsely announced from 65.0.0.0/8 to 126.0.0.0/8 plus many others.

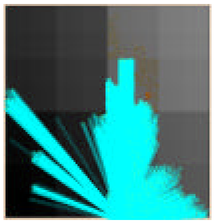
Please note that the discovery of the **pink** grid is trivial by human, if the visual orientation is right. However, the same task would be very difficult for a fully automated intrusion detection system to reveal this type of facts unless the pattern matching mechanism for grids has been included in advance. Certainly, this case shows the limitation of the traditional intrusion detection systems in handling “unknown/new/novel” attacks, while the ELISHA visual-based anomaly detection system has a very good chance to catch them. In the case of August 14, 2000, with a few clicks interactively, ELISHA not only helped detect the problem, but also, quickly nailed down the trouble source, AS-7777. Furthermore, via visualization, it even can tell the details about the errors from AS-7777.

B. *Interactive Visual Correlation: “What was AS-15412 doing in April 2001?”*

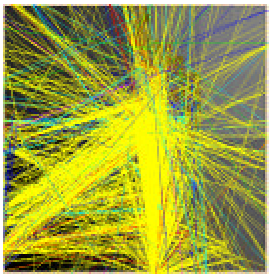
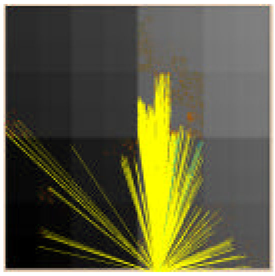
Earlier we showed that on April 5, 2001, things looked “normal”. But, on April 6, the next day, we observed a unusual amount of “**skyblue**” – the CSM events. A CSM event indicates that a particular IP address prefix was originated by one AS yesterday, but more than one ASes are claiming it today. With the possibility of multi-homing and private AS numbers, a small amount of the “**skyblue**” CSM events is probably OK, but the snapshot on the left below is visually abnormal.





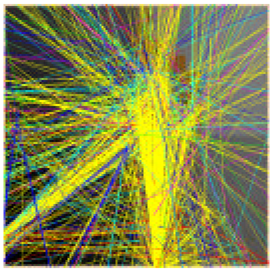
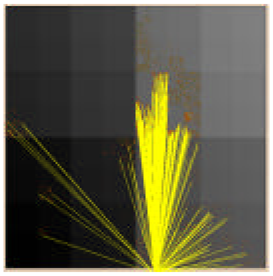


Again with a simple interactive analysis process, very quickly we identified that AS-15412 is the only AS injecting all the “**skyblue**” CSM conflicts on April 6, 2001. In the middle snapshot above, it shows that AS-15412 is conflicting with a victim AS-10132, which is the Eastar Technology Center in Hongkong. (We randomly picked one of the victims and we used the “whois” program to find out more information from the whois server, whois.radb.net.) Furthermore, after we animated the data only related to AS-15412, we found something interesting: AS-15412 not only injected thousands of CSM “**skyblue**” events on April 6, but also, from April 7 to 12, it introduced thousands of CMS “**yellow**” events. This indicates that, right after the CSM mistakes on April 6, 2001, the system administrator responsible for the AS-15412 problem started to “correct” the problems by withdrawing the bad announcements, which caused a storm of CMS events during next 6 days. The 16 small figures below shows OASC events from April 7, 2001 to April 14, 2001, i.e., two for each day. The left one (with more colorful lines) represents OASC from all ASes, while the right one includes only the OASC events connecting to AS-15412, a small ISP (Internet Service Provider) called FLAG Telecom Global Internet in London, UK.

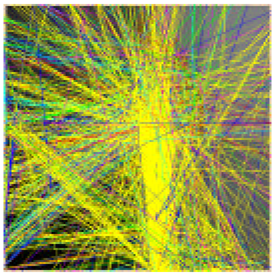
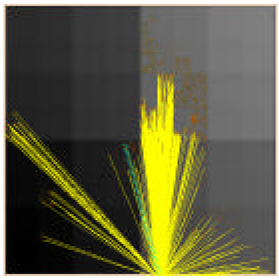
04/07/2001 all

04/07/2001 15412

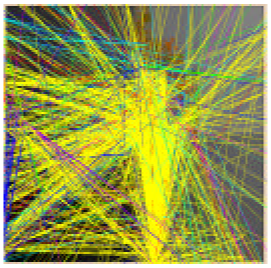
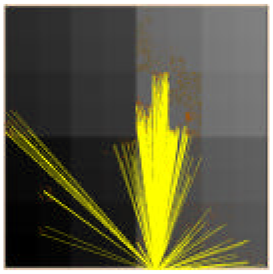
04/08/2001 all

04/08/2001 15412

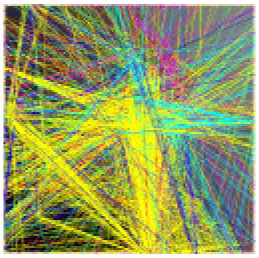
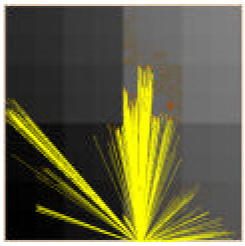
04/09/2001 all

04/09/2001 15412

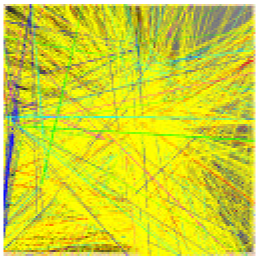
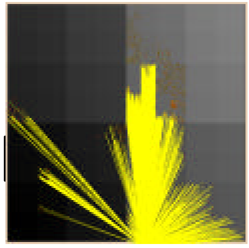
04/10/2001 all

04/10/2001 15412

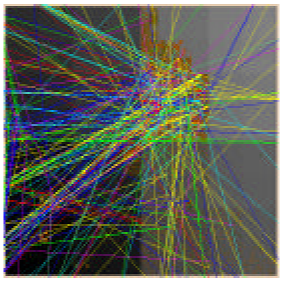
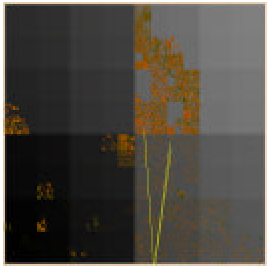
04/11/2001 all

04/11/2001 15412

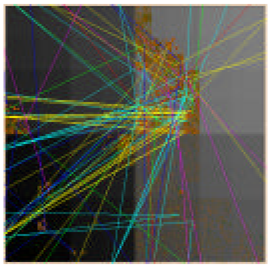
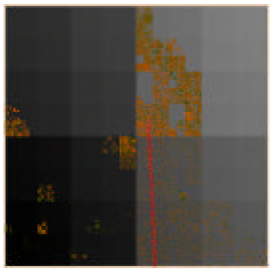
04/12/2001 all

04/12/2001 15412

04/13/2001 all

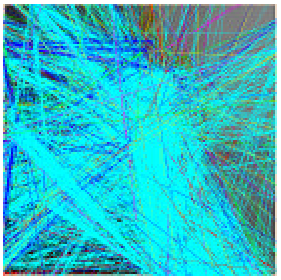
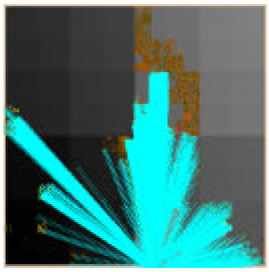
04/13/2001 15412

04/14/2001 all

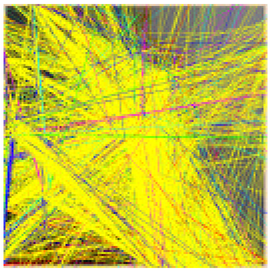
04/14/2001 15412

But, what amazed us, then, was a few days later, on April 18, 2001 (4 days later), AS-15412 caused exactly the same mistake again, and the “shape” is exactly the same as the one on April 6. The difference though is that this time it only took them one day to fix all the problems.

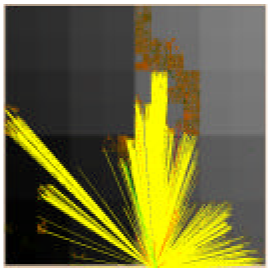



04/18/2001 all

04/18/2001 15412



04/19/2001 all



04/19/2001 15412

Please note that, via this example, we again observe the big advantage of integrating machine and human intelligence. From human's point of view, the correlation between the CSM (**skyblue**) OASC events and the CMS (yellow) events is very clear. After watching the animation and interactively identifying the AS causing the problem, we will treat, for example, the big yellow screens on both April 12 and 19 are "OK" probably because we know AS-15412 over at UK was trying to fix the problems they created. But, to draw the same conclusion without the right visual abstraction or without any human intelligence is very hard to achieve. On those two days (April 12 and 19 in 2001), without the right correlation as we shown here, thousands of false alarms might have been reported.

VII. Summary about the ELISHA/MOAS Experiment

Via the preliminary ELISHA/MOAS prototype and the experience in using the program to analyze the BGP routing data on the Internet, we have demonstrated the great potential in applying visualization techniques to critical problems in fault and intrusion detection on network protocols:

A. An interactive process approach to handle "unknown/novel" attacks/faults

In a large complex system, it is impossible to consider all possible attacks or faults. It is also very difficult to pre-design and pre-implement a set of mechanisms to detect and respond to problems not being seriously considered before. However, the human intelligence (such as the security instance response team) can certainly complement an intrusion detection system, but we must have an effective interactive process to follow in order to resolve problems correctly and quickly.

B. A visual approach to analyze and correlate events

In handling millions of events from a large complex distributed system, "false alarms" and "event correlation" become two most critical issues (or technical bottlenecks). First, the proposed visual system, ELISHA, will provide a global and abstract picture about the activities on the network. Therefore, the human operator will be given not only a hugh set of events but also the context and the relations among the events graphically. An experienced ELISHA user can then justify the validity of a reported attack instance based on his/her comprehensive awareness of the target system. On the other hand, if he/she is not certain about the situation, then the interactive process should guide him/her to navigate more information from ELISHA to reduce the potential false positive. Second, in our preliminary experiment, we found that visual event correlation might be very plausible and effective. With the right type of abstraction and animation, a human operator can quickly correlate a set of reported events and provide a valid explanation about what is going on.

Acknowledgment

This research is funded by DARPA under the fault Tolerant Networking program.

References

[1] "Guidelines for creation, selection, and registration of an Autonomous System (AS)" by John Hawkinson and Tony Bates, rfc1930, IETF.

[2] "A Border Gateway Protocol 4 (BGP-4)" by Y. Rekher and T. Li, IETF Internet draft, draft-ietf-idr-bgp4-17.txt.

[3] "OSPF Version 2" by J. Moy, rfc2328, IETF, April 1998.

[4] "Global Routing Instabilities during Code Red II and Nimda Worm Propagation" by James Cowie, Andy Ogielski, BJ Premore and Yougu Yuan, NANOG, http://www.renesys.com/projects/bgp_instability/, 19 September 2001.

[5] "Secure Border Gateway Protocol (Secure-BGP)" by Stephen Kent, Charles Lynn, and Karen Seo, in IEEE Journal on Selected Areas in Communications Vol. 18, No. 4, April 2000, pp. 582-592.

[6] "Transport and Application Protocol Scrubbing" by G. R. Malan, D. Watson, F. Jahanian, & P. Howell, *in Infocomm'2000.*

[7] "Network Intrusion Detection: Evasion, Traffic Normalization, and End-to-End Protocol Semantics" by M. Handley, C. Kreibich and V. Paxson, in Proc. USENIX Security Symposium 2001.

[8] "*Security Architecture for the Internet Protocol"by* S. Kent, R. Atkinson, RFC-2401, Internet Society, Network Working Group, Nov. 1998.

[9] "Navigating Large Networks with Hierarchies" by Stephen G. Eick and Graham J. Wills, in Proc. IEEE Conf. Visualization.

[10] "Visualizing Network Data" by R.A. Becker, S.G. Eick, A.R. Wilks, in IEEE Transactions on Visualization and Computer Graphics.

[11] "An Eye on Network Intruder-Administrator Shootouts" by Luc Girardin, in Proceedings of the Workshop on Intrusion Detection and Network Monitoring (ID'99). USENIX Assoc, Berkeley, CA, USA.

[12] "Graph Visualization and Navigation in Information Visualization: a Survey" by Ivan Herman, Guy Melançon, M. Scott Marshall, in IEEE Transactions on Visualization and Computer Graphics.

[13] "Improving BGP Convergence through Consistency Assertions" by D. Pei, X. Zhao, L. Wang, D. Massey, A. Mankin, S. F. Wu, and L. Zhang, to appear in INFOCOMM, June, 2002, New York.

[14] "*An Analysis of BGP Multiple Origin AS (MOAS) Conflict*" by X. Zhao, D. Pei, L. Wang, D. Massey, A. Mankin, S.F.Wu, L. Zhang, in ACM SIGCOMM Internet Measurement Workshop, pp.31-35, November 1-2, 2001, San Francisco.

[15] "*Real-Time Protocol Analysis for Detecting Link-State Routing Protocol Attacks*" by H.Y. Chang, S. F. Wu, and Y.F. Jou, in ACM Transaction on Information System Security, Vol. 4, No. 1, pp.1-36, 2001.

[16] "A *Property-Oriented Fault Detection Approach for Link State Routing Protocol*" by F. Wang, F. Gong, and S.F. Wu, in 9th IEEE International Conference on Computer Communication and Networks (ICCCN'00), pp.114-119, October, 2000, Las Vegas, NV.

[17] "Intrusion and Misuse Detection in Large-Scale Systems" by R. Erbacher, K. Walker, and D. Frincke, in IEEE Computer Graphics and Applications, Jan/Feb, 2002, 38-48.

[18] "Network Performance Visualization: Insight Through Animation" by Brown J.A., McGregor A.J., Braun H-W, in PAM2000 1.

[19] "CyberNet: A framework for managing networks using 3D metaphoric worlds" by P. Abel, P. Gros, D. Loisel, C Russo Dos Santos.

[20] "Managing networks through a virtual world" by Crutcher L., Lazar A., Feiner S., and Zhou M, in *IEEE Parallel and Distributed Technology*, 3(2), Summer 1995, 4-13

[21] "WWW-based 3D Distributed, Collaborative Virtual Environment for Telecommunication Network Management"by Kahani, M., Beadle, H., in Proc. Australian Telecommunication Networks and Applications Conference (ATNAC'96), December 1996, pp. 483-488.

[22] "VENoM - Virtual Environment for Network Monitoring" by Cubeta, J., Egts, D., in http://www.nrl.navy.mil/CCS/people/cubeta/venom/paper.html.

[23] "Visualizing Large Telecommunication Data Sets" by Koutsofios, E., North, S., Keim, D., in IEEE Computer Graphics and Applications, Mai/June 1999 pp 16-19.

[24] " Network Visualization with Nam, the VINT Network Animator" by D. Estrin, M. Handley, J. Heidemann, S. McCann, Y. Xu, B, Yu, in IEEE Computer, Nov. 2000.

[25] "Visualization Exploration and Encapsulation via a Spreadsheet-Like Interface", by T.J. Jankun-Kelly and Kwan-Liu Ma, IEEE Transactions on Visualization and Computer Graphics, 7(3), July-September 2001, pp. 275--287.

[26] "Image Graphs - A Novel Interface for Visual Data Exploration" by Kwan-Liu Ma, in Proceedings of Visualization '99 Conference, October 1999.

[27] "An Experimental Study of Insider Attacks for the OSPF Routing Protocol" by B. Vetter, F. Wang, S.F. Wu, IEEE ICNP'97, May 1997.

[28] "Closure and Convergence: A Foundation for Fault-Tolerant Computing", by A. Arora and M. Gouda, in IEEE Transactions on Software Engineering, Special Issue on Software Reliability, Vol. 19, No. 3, pp. 1015-1027, November 1993.