Monitoring Network Traffic with Radial Traffic Analyzer

Daniel A. Keim Florian Mansmann Jörn Schneidewind Tobias Schreck

Databases and Visualization Group
University of Konstanz, Germany
{keim,mansmann,schneide,schreck}@inf.uni-konstanz.de

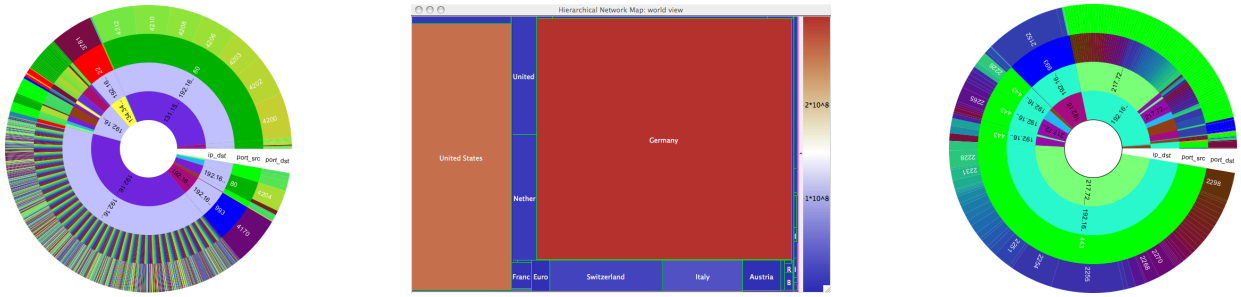

Figure 1: Radial Traffic Analyzer is a visual tool for interactive packet-level analysis of data flows in a computer network. The technique is useful to compare network load in a geographically aware display, to relate communication partners, and to identify the types of network traffic occurring at the considered network hosts.

Abstract

Extensive spread of malicious code on the Internet and also within intranets has risen the user's concern about what kind of data is transferred between her or his computer and other hosts on the network. Visual analysis of this kind of information is a challenging task, due to the complexity and volume of the data type considered, and requires special design of appropriate visualization techniques. In this paper, we present a scalable visualization toolkit for analyzing network activity of computer hosts on a network. The visualization combines network packet volume and type distribution information with geographic information, enabling the analyst to use geographic distortion techniques such as the *HistoMap* technique to become aware of the traffic components in the course of the analysis. The presented analysis tool is especially useful to compare important network load characteristics in a geographically aware display, to relate communication partners, and to identify the type of network traffic occurring. The results of the analysis are helpful in understanding typical network communication activities, and in anticipating potential performance bottlenecks or problems. It is suited for both off-line analysis of historic data, and via animation for on-line monitoring of packet-based network traffic in real time.

CR Categories: C.2.3 [Computer-Communication Networks]: Network Operations—Network Monitoring; I.3.8 [Computing Methodologies]: Computer Graphics—Applications;

Keywords: Visual Analytics, Network Traffic Monitoring, Information Visualization and Geography-based Solutions

1 Introduction and Background

Computer network infrastructures form the technical core of the Information Society. They transport increasing amounts of arbitrary kinds of information across arbitrary geographic distances. The *Internet* is the most successful computer network to date. It has fostered the implementation of all kinds of productive information systems not imaginable at the time it was originally designed. While the wealth of applications that can be built on top of the Internet infrastructure is merely unlimited, there are fundamental protocol elements which rule the way how information is transmitted between the nodes on the network. Based on these well-defined protocol elements, it is an interesting problem to devise tools for visual analysis of key network characteristics, thereby supporting the *network monitoring* application domain. Network monitoring in general is concerned with the surveillance of important performance metrics of networks to supervise network functionality, to detect and prevent potential problems, and to develop effective countermeasures for networking anomalies and sabotage as they occur.

In this paper, we consider the problem of visually analyzing important characteristics among the communication flows between hosts on the Internet. The communication data occurring is inherently complex as we have to deal with (a) large amounts of data (b) occurring in real-time, and which (c) potentially also contain complex interrelationships between the communication connections, which may furthermore (d) be varying in time. We tackle the problem by abstracting the Internet communication flow to the network (packet) level as defined by the *Open Systems Interconnection Reference* model of the International Organization for Standardization (ISO-OSI model). This model considers information flows on a network by means of packets (atomic information units) which are moved though the network from a given source host using a source port to (usually) one destination host using a destination port. Briefly, the Internet's *TCP/IP* suite of protocols implements methods to segment outbound data streams into packets which are combined at the receiving site to yield the original stream, thereby providing end-to-end connectivity. We recognize there are many options for characterizing and measuring network communication. E.g., it is possible to abstract the communication into such end-to-end connections, or go even further by analyzing the information *content* transported via such connections, like done in application-

level firewalls. We here focus on visualizing packet level communication properties, as the packet level defines a simple data structure in terms of source and targets of hosts and ports. From its port information, we can usually conclude the type of *service* addressed by the packet, e.g., port 80 usually indicates WWW traffic, port 22 indicates Secure Shell (SSH) Traffic, and so on. We therefore feel that in combination with the compact data structure given at the packet layer in the ISO-OSI model, this level is a viable option to consider for visual network communication monitoring.

Based on the IP packet data structure, in this paper we apply two different layout techniques to visualize packet-based distribution information of communication of a network. The visualization is based on the packet attributes source and destination of Internet hosts (IP-addresses) and corresponding port numbers.

We build hierarchic radial layouts visualizing the distribution of a given communication volume along the main four packet-based attributes. The basic idea of this approach is to provide a radial hierarchical layout, to visually represent the frequent patterns in a high level view, and to allow the user to get details on demand by providing drill down and selection capabilities. Combining the radial layouts with an appropriate colormap, the user gets a compact informative summary over the packets inbound and outbound with respect to a given host on a network. We complement the radial network packet layouts by a second layout technique where we leverage a *Treemap* [19] like rectangular layout technique to visualize the geolocation of packets as derived from their respective IP-addresses. We discuss results of the application of the two approaches on a real-world data set collected at a workstation of one of our department members, and also from the root Internet gateway of our institution. The results demonstrate the usefulness of the techniques for analyzing packet-level network traffic characteristics present on a local user's workstation, and also from the gateway perspective. The tool is useful in discovering interesting distribution information like the pattern and sizes of traffic between outside networks and a given local system. Also, the types of services the users utilize can be readily perceived, making possible surveillance of compliant usage of the network by the users. The technique may also be useful for instructional usage like teaching practical aspects of the TCP/IP protocol within the ISO-OSI reference model.

The remainder of this paper is structured as follows. Previous work is discussed in the next section. In Section 3, we briefly describe the architecture employed for our experiments. Section 4 introduces the *Radial Traffic Analyzer* layout scheme, which in Section 5 is combined with a geospatial layout technique for enhanced data representation. Section 6 discusses use cases of the techniques, and Section 7 gives some preliminary, informal evaluation we collected from the experiments performed on our data. Finally, Section 8 concludes and outlines future work in the area.

2 Related Work

Visual support for network monitoring has recently gained momentum, as documented by the CSS Workshop on Visualization and Data Mining for Computer Security in 2004 (VizSEC/DMSEC 2004) and by the Workshop on Visualization for Computer Security held in conjunction with the 2005 IEEE Visualization conference. First results have been presented there; still it is an intriguing endeavor to design visual analysis tools for network monitoring which has just yet begun.

To display IP-related events such as port scans, errors, or IDS alerts, Lau [16], for example, presented the *Spinning Cube of Potential Doom*. The visualization is based on a rotating 3D cube used as 3D scatterplot. However, 3D scatterplots are difficult to interpret on a 2D screen introduce overlay problems.

The glyph-based security visualizaton [14] as a user-centered approach offers a visual interface to assign variables of network statistics and intrusion detection data sets to visualization attributes, ultimately leading to a glyph visualization of the past events or the current situation. On the one hand, this approach is very flexible, but on the other hand many possible parameter settings make choosing a good visualization a difficult task.

Other research focuses on placing IP addresses as pixels on the screen, grouping them using rings according to trust levels and balancing the pixel distribution within (cf. [9]). This approach is certainly more powerful in displaying many different IP addresses. However, the visual correlation of those IP addresses with the other dimensions of the data becomes difficult. *IDS Rainstorm* [6] also uses small visual units such as pixels to show an overview utilizing several axes for the whole local IP address space. After zooming into regions of interest, lines appear and link the pictured incidents to other characteristics of the data set. This linking in detail views is also utilized in other applications like *TNV* [11] or the *VisAlert* W^3 tool [10]. In contrast to these methods, we try to bring together the complementing pieces of information through extensive use of the visualization attribute position. Different variables of the data set are mapped to rings (see Figure 2), and the positioning scheme makes analysis of a single data item easy by following a straight line from the center to the outer ring. Furthermore, sorting and grouping operations are applied to bring similar data tuples close to each other. As a perfect arrangement taking into account all attributes of the data set is not possible, we use color in order to visually link identical data characteristics.

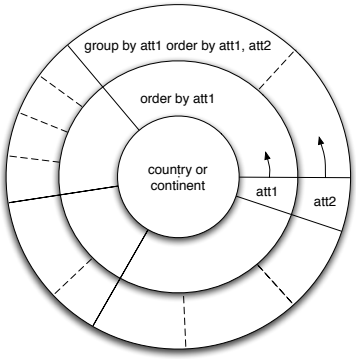

Figure 2: Design ratio of RTA

Our approach offers easy-to-understand metaphors like one rectangle for each country scaling its area according to its traffic or one ring for each attribute of the data set as well as intuitive interaction capabilities.

3 Software Architecture

In our analysis, we focus on the network layer of the internet protocol stack. The network layer provides source and destination IP addresses, whereas the transport layer provides source and destination ports. Additionally, we collect information about the used protocols, (mostly TCP and UDP) as well as the payload (transferred bytes). In short, we store a tuple $t = (time, ip_{src}, ip_{dst}, port_{src}, port_{dst}, protocol, payload)$ for each transferred packet. For matters of simplicity, we restrict ourselves to UDP (used by connection-less services) and TCP packets (used by connection-oriented services). To capture network packets, we use the packet capturing libraries *libpcap* [4] and *WinPcap* [5], as well as the Java wrapper *JPcap* [1] to access the libraries using a Java interface.

To store and retrieve real-time network statistics from a local PC in a convenient way, we employ a *SQLite* database [3], which pro-

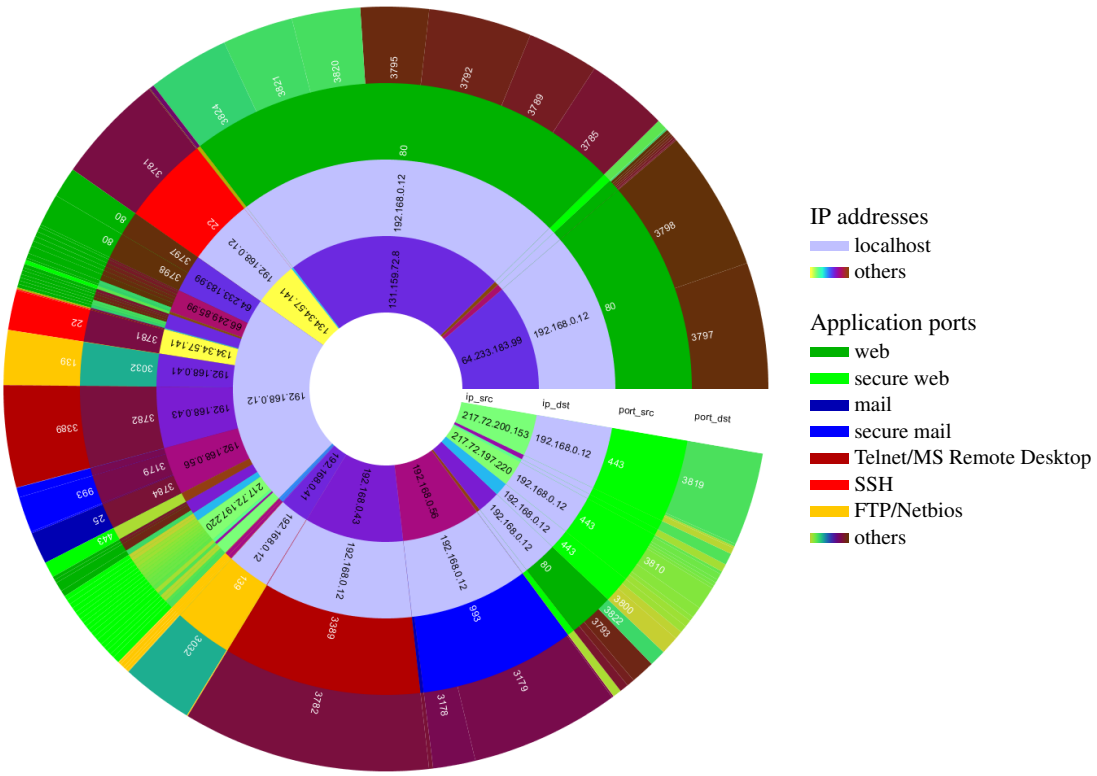

Figure 3: RTA display showing the distribution of network traffic of a local computer. We maintain an overview by grouping the packets from inside to outside. The inner two circles represent the source and destination IP addresses, the outer two circles represent the source and destination ports. Traffic originating from the local computer can be recognized by the lavender colored circle segment in the inner ring. Traffic to this host can be recognized by the lavender colored segments on the second ring. Normally, ports reveal the application type of the respective traffic. This display is dominated by web traffic (port 80 - colored green), remote desktop and login applications (port 3389 - red, port 22 - bright red) and E-mail traffic (blue).

vides a thin implementation on the database side. To better serve the performance requirements of monitoring large networks, we decided to integrate a second database interface for a *PostgreSQL* database [2]. For the analysis of larger data sets, a more intelligent preprocessing is employed by merging individual packets to sessions to significantly reduce the database size. The easiest way to do this preprocessing is to take advantage of the knowledge implemented in commercial routers by exporting their packet statistics functionalities which group matching outgoing and incoming packets into one connection.

Usually, the data to be examined is abundant and the normal daily patterns conceal exceptional traffic patterns. Therefore, filters are crucial for the task of finding malfunctions and threats within the information infrastructure. In our tool, we implemented rules to discard "ordinary" traffic (e.g., web traffic), but also to select just certain subsets of the traffic (e.g., traffic on ports used by known root-kits). In the course of the visual analytics process, the user interactively applies, combines, and refines these automatic analysis methods to confirm or reject hypotheses about the data in her or his search for insight.

4 Radial Traffic Analyzer

The visualization metaphor of the Radial Traffic Analyzer (RTA) consists of concentric rings subdivided into sectors and is very close to the *Solar Plot*, *Sunburst* and the *Interring* [7, 21, 23]. Roots of the utilized radial layout are discussed in previous work of ours (cf. [8]).

As users might tend to minimize eye movements, the cost of sampling will be reduced if items are spatially close (cf. [22], p. 156). We therefore choose a radial layout for RTA, place the most important attribute (as chosen by the user) in the inner circle, and arrange the values in ascending order, to allow better comparisons of close and distant items. The subdivision of this ring is conducted according to the proportions of the measurement (i.e. number of packets or connections) using an aggregation function over all tuples with identical values for this attribute. Each further ring displays another attribute and uses the attributes of the rings further inside for grouping and sorting, prioritized by the order of the rings from inside to outside as illustrated in Figure 2.

In the default configuration, we use four of these rings. The visualization is to be read from inside to outside, starting from the innermost ring for the source IP addresses, the second ring for the destination IP addresses, and the remaining two rings for the source and the destination ports, respectively. In Figure 3 beginning on the right, we map the fractions of the payloads for each group of network traffic counter-clockwise on the rings while sorting the groups according to ip_{src}, ip_{dst}, $port_{src}$, and $port_{dst}$. Beginning with grouping the traffic according to ip_{src}, we add another grouping criteria for each ring further outside. This results in a finer subdivision of each sector on the next ring.

To facilitate a better understanding of the rings, sectors representing identical IP addresses (inner two rings) are drawn in the same color, ports (outer two rings) respectively. To further enhance the coloring concept, we created a mapping function for ordinal attributes that maps a number x (i.e., the port number, or IP address number) to the indices of an appropriate colormap: $c(x) = x \ mod \ n$ (n: number of distinct colors used). Prominent

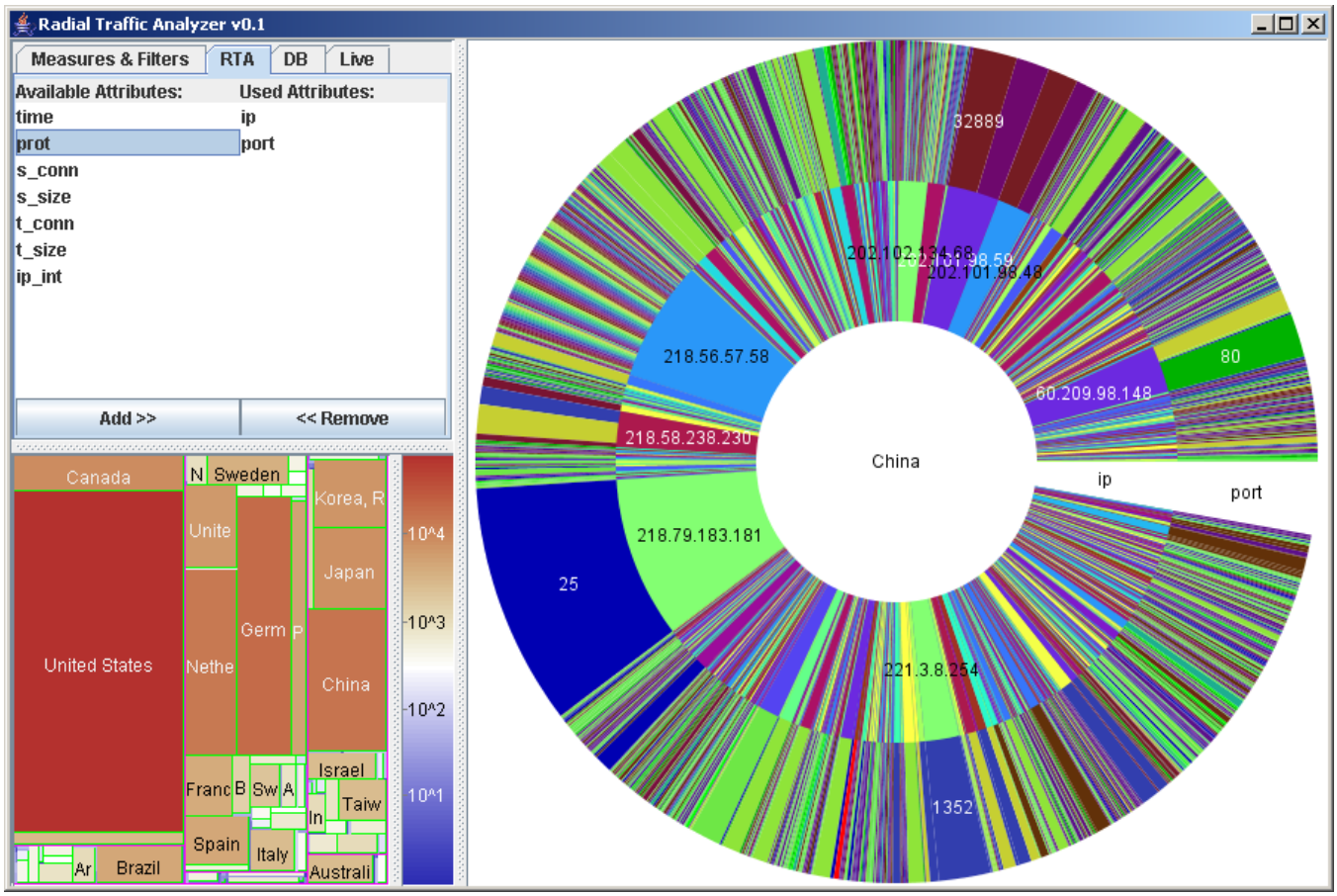

Figure 4: Integrated System View of RTA: On the bottom left attributes of the data set can be added as additional rings. In this case, traffic from China was selected in the HistoMap visualization (bottom left) which shows the country-wise proportions of filtered network traffic. In this case the accumulated number of failed connections from inbound traffic of our University gateway on 11/29/2005 was employed. A port scan from host 218.56.57.58 is visible as well as a large amount of failed attempts to open SMTP connections (email delivery) from host 218.79.183.81.

ports (e.g., HTTP=80, SMTP=25, etc.) are mapped to colors that do not show up in our colormap for easier identification. This mapping function facilitates to correlate close IP addresses or ports. To differentiate between traffic that is transferred over an unsecured and a secured channel, we modify the brightness of the color (i.e., HTTP/80 = green, HTTPS/443 = light green, etc.). To map numeric attributes (e.g., number of connections, time, etc.) to color, it makes more sense to normalize the data values and then map them to a colormap with light to dark colors or vice versa. Different colormaps were used for the attributes, and should clarify the comparability of rings. An IP address appearing as a sending host in the innermost circle and reappearing as a receiving host in the second circle should be colored identically, whereas this color should then not be used for a port. We further elaborate on these aspects in Section 6.

The main bottleneck of the technique is display space. Rings further outside show more detailed information while consuming more display space at the same time. Depending on the question at hand, different grouping is useful and is done by assigning the chosen dimension (i.e., source IP, destination IP, source port, destination port) to the inner rings. On the one hand, a grouping according to the hosts might be useful when determining high-load hosts communicating on different ports, while on the other hand a grouping according to the target ports clearly reveals the load of each type of traffic. To compensate for the strict importance rating according to the inner circles, the positioning and thus importance within the sorting order can be interactively changed using drag & drop mouse interaction.

As soon as many different circle segments are drawn, some segments become too small to plot labels into. Therefore, we cut long labels and employ Java tooltip popups showing the complete label and additional information like the host name for a given IP address, and the possible application programs corresponding to the respective ports (see [12]). As filtering is an often used task, a simple mouse click triggers a filter that discards all traffic with the chosen attribute values. Detailed information about the data tuples represented through a circle segment is accessible using a popup menu.

Transferred bytes is not the only available measure when analyzing network traffic. When investigating failed connections, for example, the measure transferred bytes would not show the data tuples of interest on the ring, as they all have 0 bytes for the attribute. In this situation, the measure number of connections would be useful to correctly size the circle segments.

Experts often compare transferred bytes to the count of sessions on a set of active hosts. High traffic with only few sessions is considered to be a download ressource, whereas medium traffic on many sessions it typical for more medium-bandwith applications like WWW.

The RTA display is flexible to display many different data sets and can be adjusted to the data at hand on the fly. An example is to configure the inner two rings with the source and target IP

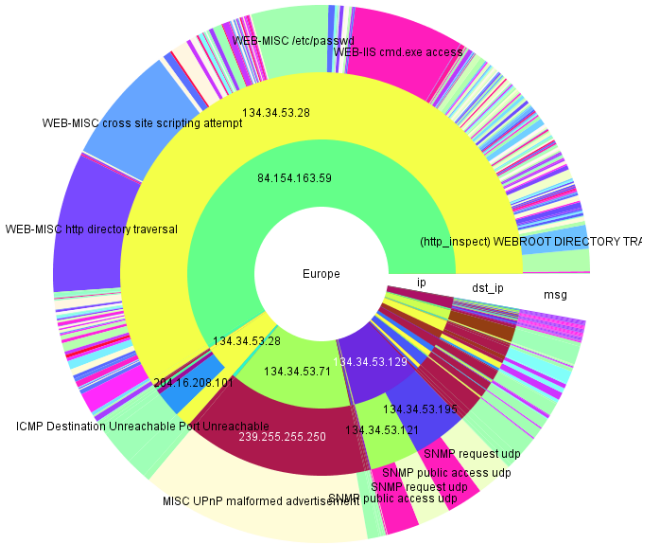

Figure 5: Displaying security alerts from the intrusion detection system *Snort*. After discarding ICMP Router Advertisements, ping and echo alerts, we can clearly see that host 134.34.53.28 (green) was attacked by 84.154.163.59 using various methods (outer ring).

addresses and the outer ring with security alerts generated by an intrusion detection (IDS) system (see Figure 5). Alternatively, one can extend the IP address dimension through the use of associated higher-level network attributes (e.g., IP network block, autonomous system, etc.) to investigate whether e.g., a denial of service (DOS) attack originates from a certain network block, or to assess the danger of a virus spread from neighboring autonomous systems.

5 Combining RTA with Geospatial Displays

To retrieve a country name for a given IP address, we use Maxmind's GeoIP Database [18], which claims to assign 99% (95% in the non-commercial version) of all IP addresses correctly to a country. After having evaluated this geo-location information, we use the HistoMap algorithm [13] to partition the available display space into rectangles. Each rectangle represents a country, and is scaled such that its size proportionally represents the traffic volume inbound (or outbound, repsectively) to (from) the given country. We adapted the HistoMap algorithm in such a way that it not only approximately preserves spatial relations of neighboring continents and countries, but also optimizes the output rectangles for squareness. This is done by preferring rectangle splits in either horizontal or vertical direction based on a test whether the resulting rectangles are more square-like than when performing the split in the other orientation.

On a click on one of the squares in the HistoMap display (see Figure 4, bottom left), the RTA display shows detailed traffic statistics in the main view. Drill-down and roll-up functionalities provide aggregates of the traffic data for each continent or a detail view for each country and are triggered by mouse wheel interaction. Coloring is done using a logarithmic scale as network traffic characteristics feature high variances. We tried to directly draw the RTA displays into the rectangles, but this resulted in heterogenous scales across the whole display, as longer rectangles offer less space than equally sized squared ones.

6 Results and Findings

We found out that our tool is useful to observe network traffic characteristics over time. By using a time frame up to the current moment in which we group the captured packets, we can display a smooth transition by continuously updating the screen. In Figure 6 one can see a series of RTA displays to observe changes in network traffic. There are three different modes to visualize network traffic, namely (1) to aggregate all traffic and continually add the new traffic, (2) to specify a time frame in which one measures the traffic and continually drop the old traffic, and (3) to always display the same amount of traffic by specifying a flexible time frame.

We also applied our tool for detecting port scans within a large data set, and the results were visually conspicuous (cf. Figure 4) and intuitively recognized as scans: Due to the sorting order, the whole spectrum of colors from the colormap appears several times on the second ring. This visualizes that a continuous range of ports has been probed, which is typical for a port scan. Network traffic of "normal" applications varies the used source ports only infrequently, and just a few target ports are normally employed.

Another possibility is to scale the radius of the circles according to the traffic load they represent. In this way, the network monitoring analyst gets a visual clue on the load situation. However, the major drawback of this possibility is that the display might become too small to analyze because of strong variations in the network traffic. We therefore discarded this option and do not present results on it here.

7 Evaluation

According to the feedback we got out of in a limited, informal user study we performed with a number of our undergraduate students, the mapping of network data to a radial layout makes intuitive sense and offers an effective overview of the composition of network communication in terms of network packets. It was recognized that the technique is applicable to small data sets captured on a local computer, as well as to traffic monitored on the university gateway after intelligent preprocessing (we obtained anonymous, cumulated statistics). However, the technique cannot show all details due to the visual limitations inherent in radial layouts. We can compensate for the shortcoming by discarding some obvious traffic, such as web and mail traffic, and by offering fast interactive filtering capabilities.

The application of geographic distortion techniques appears to be useful especially in static displays. Due to the restrictions of the applied geographic distortion technique, unwanted discontinuities in the positioning of geographic items were recognized when their proportions changed. We see further optimization potential by applying different distortion techniques.

8 Conclusions

The main contribution of this paper is the adaption and application of radial and rectangular layout techniques to the domain of network traffic monitoring on the ISO-OSI packet level. We presented the Radial Traffic Analyzer which is capable of visually monitoring network traffic, relating communication partners and identifying the type of traffic being transferred. Statistics about the network traffic were captured, stored and grouped in order to present them in a meaningful way. The RTA display is perfectly suitable to show grouped information in the inner circles while presenting related detail information on the outer circles. It is complemented by appropriate interaction techniques like hints on mouse-over, drag & drop to adapt the order of the rings, filtering using clicks and details accessible via a popup menu.

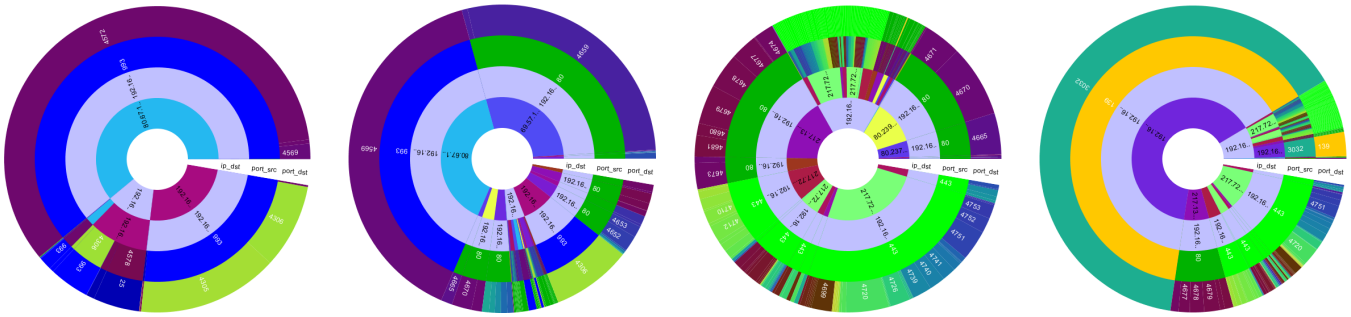

Figure 6: Animation over time: a) The user first checks her email (blue) on two different mail servers, and then sends out one email using an unsecured channel (dark blue). b) She then surfs on some web pages (port 80, dark green). As one can see, the blue mail traffic is still visible in the bottom left corner. c) Afterwards, the user logs into her online banking account using HTTPS (bright green). d) Finally, a large file is accessed on the local file server using the netbios protocol (orange)

By using a time frame, we are capable of continuously monitoring network traffic. Due to the applied grouping characteristics, changes within the visualization are smooth in many realistic scenarios. The use of a spatial visualization which enables grouping of network traffic according to the geographic sources is a further feature.

The Radial Traffic Analyzer is not only suitable for monitoring purposes, but also to understand networking concepts in the scope of education.

For future work, we plan to make our tool publicly available and extend its interactivity. We intend to combine the RTA display with our *Hierarchical Network Map* [17] which places autonomous systems (internet backbone systems) and networks within the country nodes of a HistoMap. Like shown in [15], we plan to extract rules from the insight gained through interaction with out tool to enhance future discovery of attacks and intrusion using rule-based intrusion detection systems like snort [20]. Furthermore, we want to research zoom regions within RTA to show details without prior filtering.

Acknowledgement

We thank Barbara Loehle for providing data as well as Christian Panse and Mike Sips for their valuable input. The work was partially funded by the German Research Foundation (DFG) under grant GK-1042 “Explorative Analysis and Visualization of Large Information Spaces”, University of Konstanz, Germany.

References

[1] JPcap. http://netresearch.ics.uci.edu/kfujii/jpcap/doc/index.html.

[2] PostgreSQL. http://www.postgresql.org/.

[3] SQLite. http://www.sqlite.org/.

[4] tcpdump and libpcap. http://www.tcpdump.org/.

[5] WinPcap. http://www.winpcap.org.

[6] Kulsoom Abdullah, Chris Lee, Gregory Conti, John A. Copeland, and John Stasko. Ids rainstorm: Visualizing ids alarms. In *Proc. IEEE Workshop on Visualization for Computer Security (VizSEC)*, October 2005.

[7] Mei C. Chuah. Dynamic aggregation with circular visual designs. In *1998 IEEE Symposium on Information Visualization (InfoVis '98), 19-20 October 1998, Research Triangle Park, NC, USA, Proceedings*, pages 35–43, 1998.

[8] M. Sips D. Keim, J. Schneidewind. Fp-viz: Visual frequent pattern mining. In *Poster Paper, IEEE Symposium on Information Visualization (InfoVis 2005), Minneapolis, MN, USA, October 23-25*, 2005.

[9] Glenn A. Fink and Chris North. Root polar layout of internet address data for security administration. In *Proc. IEEE Workshop on Visualization for Computer Security (VizSEC)*, October 2005.

[10] Stefano Foresti, James Agutter, Yarden Livnat, and Shaun Moon. Visual correlation of network alerts. *IEEE Computer Graphics and Applications*, 26(2):48–59, March/April 2006.

[11] John R. Goodall, Wayne G. Lutters, Penny Rheingans, and Anita Komlodi. Preserving the big picture: Visual network traffic analysis with tnv. In *Proc. IEEE Workshop on Visualization for Computer Security (VizSEC)*, October 2005.

[12] Internet Assigned Numbers Authority. TCP and UDP port numbers. http://www.iana.org/assignments/port-numbers.

[13] Daniel A. Keim, Florian Mansmann, Christian Panse, Joern Schneidewind, and Mike Sips. Mail explorer - spatial and temporal exploration of electronic mail. In *Proc. Eurographics/IEEE-VGTC Symposium on Visualization (EuroVis 2005), Leeds, United Kingdom June 1st-3rd*, 2005.

[14] Anita Komlodi, Penny Rheingans, Utkarsha Ayachit, and John R. Goodall. A user-centered look at glyph-based security visualization. In *Proc. IEEE Workshop on Visualization for Computer Security (VizSEC)*, October 2005.

[15] Kiran Lakkaraju, Ratna Bearavolu, Adam Slagell, William Yurcik, and Stephen North. Closing-the-loop in nvisionip: Integrating discovery and search in security visualizations. In *Proc. IEEE Workshop on Visualization for Computer Security (VizSEC)*, October 2005.

[16] Stephen Lau. The spinning cube of potential doom. *Communications of the ACM*, 47(6), 2004.

[17] Florian Mansmann and Svetlana Vinnik. Interactive exploration of data traffic with hierarchical network maps. 12(6), November/December 2006. to appear.

[18] Maxmind LLC. GeoIP Country Database. http://www.maxmind.com/.

[19] Ben Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Transactions on Graphics*, 11(1):92–99, 1992.

[20] Sourcefire. Snort. http://www.snort.org/.

[21] J. T. Stasko and E. Zhang. Focus + context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In *Proceedings of the IEEE Symposium on Information Visualization*, 2000.

[22] Colin Ware. *Information Visualization, Perception for Design*. Academic Press, 2000.

[23] Jing Yang, Matthew O. Ward, Elke A. Rundensteiner, and Anilkumar Patro. Interring: a visual interface for navigating and manipulating hierarchies. *Information Visualization*, 2(1):16–30, 2003.