Visual Analysis of Corporate Network Intelligence: Abstracting and Reasoning on Yesterdays for Acting Today

D. Lalanne, E. Bertini, P. Hertzog, and P. Bados

Abstract This article proposes to go beyond the standard visualization application for security management, which is usually day-to-day monitoring. For this purpose, it introduces a pyramidal vision of the network intelligence and of the respective role of information visualization to support not only security engineers, but also analysts and managers. The paper first introduces our holistic vision and discusses the need to reduce the complexity of network data in order to abstract analysis and trends over time and further to convert decisions into actions. The article further introduces the analysis tasks we are currently tackling. The two following sections present two different ways to overview network data concentrating on specific dimensions of network security: user and application centric firstly, and alarm and temporal centric secondly. Finally this article concludes with the limitations and challenges introduced by our approach.

1 Introduction

Most of the visualization tools designed and implemented so far for the domain of corporate network security generally support day-to-day monitoring of network activities or high level security dashboards. However, numerous other user profiles and needs are related to the administration and analysis of a computer network in a company, and there is an increasing need to analyze and take decisions on this resource and its related information. In other words, not only the security team and the system/network engineers are nowadays interested in reflecting on the network

E. Bertini and D. Lalanne
DIVA/DIUF University of Fribourg CH-1700 Fribourg, Switzerland, e-mail: e.bertini@unifr.ch, d.lalanne@unifr.ch

P. Hertuog and P. Bados
NEXThink S.A. Parc Scientifique, PSE-B CH-1015 Lausanne, Switzerland, e-mail: p.hertuog@nexthink.com, p.bados@nexthink.com

topography, users and applications, but also higher level decision-makers such as the security architect, the chief security officer, the helpdesk, legal department or even employees farther away from the raw network resources, such as the business managers, the chief information officer and finally the chief executive officer.

In this position paper, we propose to consider network intelligence as a central resource of the company and consider the role of interactive visual tools for supporting not only daily monitoring but also other administrative activities related to a corporate network, which generally requires analysis over a longer period of time.

Most of the data related to a corporate network can be represented with the tuple *who* (users), *where* (hosts), *how* (applications/ports), *what* (alarms) and *when* (specific time). From our experience, the *when* parameter has a privileged role and can be taken into account at various levels of detail: day(s) perspective for monitoring or tracking activities; weeks, months for a deeper analysis in time; months, quarters or years for trending. In this article, we will emphasize on the analysis aspect since it is the middle layer standing between the security team and the management. We believe this middle layer can particularly be helpful for increasing knowledge and incrementally defining policies.

The various time levels mentioned above imply various user tasks and different levels of detail to be supported. Figure 1 represents our understanding of network administration, from day-to-day monitoring by system administrators to analysis and further trending by managers. Of course users can have various roles and are not stereotyped to one activity. To make it simpler, we define three main layers corresponding to three major user tasks:

1. *Monitoring*. This task is already well supported by visualization tools and allows tracking abnormalities on a corporate network and finding the related causes in order to take immediate actions to preserve the sake of the network.
2. *Analyzing*. This task stands in the middle layer of our pyramidal view. We believe this layer is particularly crucial and must be supported by visualization tools. There are numerous analysis tasks to support such as segmenting user types and applications through visual clustering, visualizing alerts over time as an

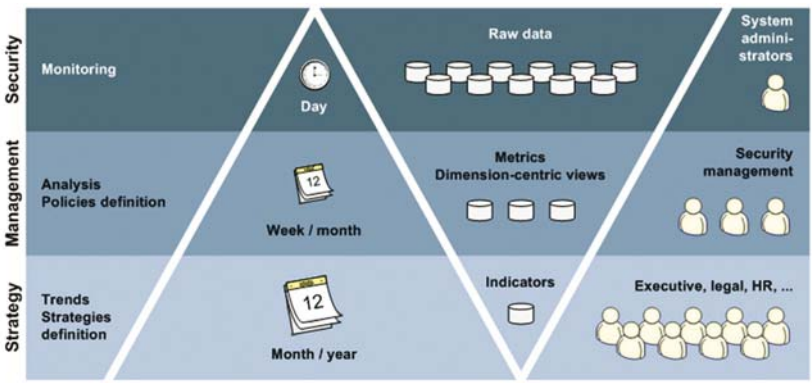

Fig. 1 Various levels of abstraction, various time granularities (day-to-day, month, year), and various roles of users, with different needs, are involved in network administration

entry point to understand relationships between network data, and also finding correlations, grouping similar elements, eliciting outliers.
3. *Trending*. Based on simple indicators, such as performance metrics, volume, license prices, etc. the management can observe its corporate IT evolution over time and can compare its own company with others in the field. The role of such kind of tools is the support for taking decisions and defining novel strategies, to be further converted in policies by its team members.

The pyramidal view in Fig. 1 also indicates that the level of details of the network data is inversely proportional to the time range of observation, mainly for two reasons: the increasing amount of data to handle when considering larger periods of time, and the level of details necessary to take high level decisions. Because there is a large amount of raw data, aggregation is necessary in order to reduce complexity when shifting from day-to-day monitoring to larger periods of time. A top manager will need to see how simple indicators evolve during a full year, whereas an analyst who needs to apply network policies will need to analyze activities within a month, in order to adapt management strategies and decisions into actionable policies.

2 Background

In the recent years the use of visualizations as a means to monitor corporate networks and detect potential threats has grown in interest, and a good number of systems have been developed. Visualizations can be useful in network security, not only for monitoring, but also to analyze evolution in time of large quantities of data. When mapping data to visual features, it is possible to perceive complex patterns at a glance and to reduce the burden associated to the reasoning activity. Visualization works as an external memory, offloading cognitive resources and considerably increasing the efficiency and effectiveness of analysis (Allen et al., 2000; Card et al., 1999).

One way to classify the existing security visualization systems is according to the nature of their data source. Tools such as visFlowConnect (Yin et al., 2004), nVisionIP (Lakkaraju et al., 2004), RUMINT (Conti et al., 2005) or TNV (Goodall et al., 2005) manipulate network flows or the results of packet inspection. Applications like MieLog (Takada and Koike, 2002a) or Tudumi (Takada and Koike, 2002b) use logs collected directly on the endpoints. And finally, RainStorm (Abdullah et al., 2005), SnortView (Koike and Ohno, 2004), STARMINE (Hideshima and Koike, 2006), VisAlert (Yarden et al., 2005), or other visualization-based tools that use an hybrid approach (Hertzog, 2006), visualize alarms directly generated by IDS. Regardless the type of data utilized by the system, and thus the kind of supported tasks, all these systems share the same principle of using visualization to increase *situational awareness* and to make more effective and simpler the detection and comprehension of abnormal behaviors. Even if these systems also support the analysis of threats and the formulation of possible solutions, their main focus remains the monitoring task. The quality of a security visualization system is measured in its

ability to convey all the necessary information and to provide administrators with a sense of control of their network. For this specific reason, most of these systems further share a time perspective limited to day-to-day monitoring.

While ensuring a daily network safety is of paramount importance, we want to investigate the idea of taking into account longer time spans and to consider the analysis of a network not only in the perspective of monitoring but also of strategic analysis: understanding relationships between data, managing a segmented population of users and applications, observing the evolution of various indicators (overall risk, volume, number of licenses, etc.), and finally devising adequate network policies. A similar approach is presented in (D'Amico and Kocka, 2005) where different types of analyses, job functions and uses of visualization are described. The need to go beyond the day-to-day monitoring task is also acknowledged in the domain of *computer forensics* where the data collected over long time periods is necessary to deeply analyze some potentially criminal behaviors (Allen et al., 2000) but this domain does not cover the needs we try to address here: the main objective in forensics is to follow the paths of some criminal acts to find evidence of them; here we propose to look into the data to increase knowledge and take informed decisions.

As soon as we shift from the daily monitoring paradigm toward the extended time analytics paradigm, we face the problem of data explosion. Security visualization can profitably draw ideas from other fields of computer science like data warehousing, data mining, and visualization, where the problem of coping with large quantities of data to visualize trends and patterns has been largely investigated. Data reduction and summarization are particularly pertinent here. As an example, in business intelligence there is a long tradition of methods and techniques conceived to cope with millions of transactions accumulated everyday (Ramakrishnan and Gehrke, 2000). With such a volume of data produced at a constant rate it is mandatory to decide what to retain and what to discard, and also at what level of abstraction data must be represented. An interesting initial study going in this direction is used by Hierarchical Network Maps (Mansmann and Vinnik, 2006) a visualization tool for monitoring data traffic, where OLAP data cubes are used to represent data hierarchically and at different level of details. In data mining the are also several methods that might be useful to our purposes to: help reducing the amount of data, produce more abstract descriptions of the data, and discover hidden information (e.g., dimensional reduction, sampling, clustering, rules induction, classification) (Han and Kamber, 2000). Visualization then has as well an established set of tools and techniques to deal with large quantities of data and/or to produce effective visual abstractions such as: pixel-based visualizations (Keim, 2000) and visualization from data cubes (Stolte et al., 2002).

3 On the Need to Support Visual Analysis

Figure 2 details the middle layer presented in the introduction and particularly emphasizes on the gap between the usual daily network monitoring and the very high level of decision making. Our claim is that this gap is currently not well bridged

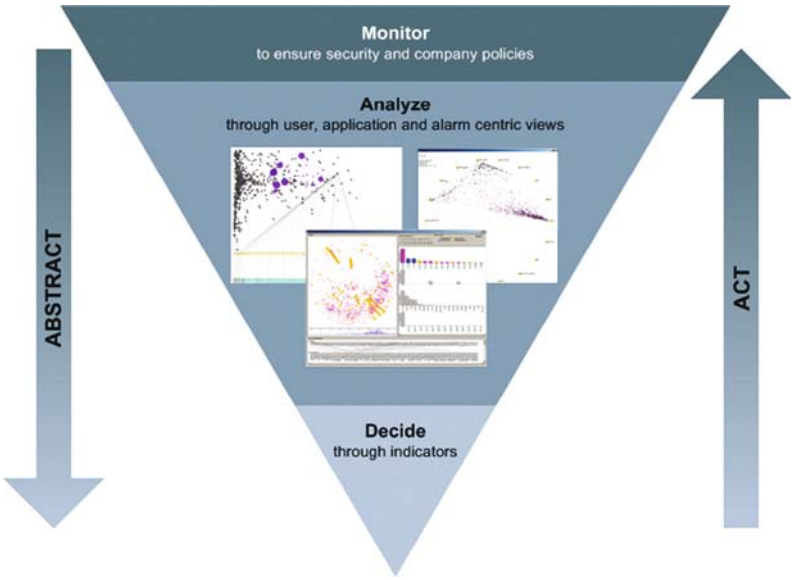

Fig. 2 Visual interactive tools are critical not only to support analysts in abstracting indicators to help decision makers defining strategies but also to convert high level strategic decisions into actionable policies

by any interactive tools, neither in the bottom-up direction, nor in the top-down. The bottom-up bridge should support the analysis of raw data in order to abstract indicators for the top management to easily take decisions, whereas the top-down bridge should ensure that the causes of the discovered trends are elicited, and the taken decisions are, for instance, converted by the analyst into actionable network policies.

Monitoring network data over time can be useful to observe how the network evolves according to taken actions ("*what if I add this policy, or dispense this awareness program to my users*?"). Visualization tools can help observing evolution and development in order to compare network status before and after specific intentional actions. Although it is hard to infer a strict causality between a specific action and the resulting network evolution, correlations between actions and impacts on the network can be clearly brought to light. Furthermore, we believe giving analysts the opportunity to annotate the network at specific times, or to label applications or users, could be of great value for assessing the impact of network policies or management strategies (control volume, optimize number of licenses, etc.). In a longer term, we could envision supporting prediction of network evolution on simple indicators, such as network overall performance based on customizable features.

To wrap it up, our belief is that abstracting network data in time and reducing complexity for the sake of understanding open various opportunities to increase knowledge, support high level reasoning, devise policies and strategies, and monitor

specific events, performance and risk over time in order to finally increase the overall security level. Our pyramidal view further sustains the idea of supporting collaboration between layers and people within an IT service, not limited to security.

3.1 Types of Analyses

In our view the analytical process can have a strong impact both on the lower and higher layers, interacting with each with different tools and purposes. Two types of analyses can take place.

- *Explorative (from monitoring to deciding)* – its main purpose is to find something useful or interesting without having a completely formulated and explicit goal. We see this type of analysis originating from the middle of the pyramid, thus mainly performed by security analysts as a way to inform administrators (below) and managers (above).
- *Explicative (from decision to action)* – its main purpose is to explain trends or behaviors of interest observed in the system. This is the type of analysis that is currently performed at the monitoring level to: (1) understand what is the real danger represented by the event; (2) understand the origin of the event. The same type of analysis however can be scaled to longer time spans and adopted by the higher levels of the pyramid. As an example, current trending tools, commonly found in dashboards, permit to see trends (mainly variations in time) but fail to explain why such trends take place. Being able to explain them might increase the ability to devise clever policies.

Explicative analysis takes requests from the outside to return explanations. Explorative analysis stems from an inner and less focused effort to analyze what happens in the other layers and to inform them when some useful knowledge is produced. Both types of analyses are useful and implemented only in few systems. We believe that the approach we propose can support both these tasks and increase the distribution of knowledge across all the layers within an organization or company.

3.2 Analysis Tasks

In our attempt to understand the kind of analytical tasks that can be performed in network security as soon as we abstract away from the view of "pure" network monitoring we have devised a list of possible tasks. This is certainly not exhaustive but is useful to understand what kind of knowledge might be generated. The following tasks are also the ones supported by the software tools and prototypes we have developed so far and from which we have gained most of our experience. In describing these tasks we consider that the following data on network's traffic is available: source and target hosts, applications, ports, and user IDs and that some

form of alert system is in place. Following are the tasks we have isolated from our own experience:

- *Segmentation (who does what)* – the network can be seen as a place with actors (users) who exploit some resources and generate traffic and events. In this context one is interested to know who does what to segment the population or the resources according to the traffic or events they generate. As an example, in our SpiralView (presented below), it is possible to see which users, with what resources, generate some specific types of alarms: these elements are "segmented" in terms of alarm types. Many other methods of segmentation can be imagined.
- *Correlation, clustering, and outlier detection (building profiles)* – what is really difficult from the perspective of a network analyst is to summarize in few elements what are the typical behaviors/habits taking place in a network. As an example, it is certainly true that there will be groups of users who use the same set of applications in more or less the same manner and that spotting them would be useful to build user profiles and thus to devise specific policies for specific groups. In this task we consider all types of analyses permitting to identify homogeneous groups of resources that explain some relevant behaviors observed in the network. Another example would be to see if there are any emerging patterns between source hosts and target hosts, i.e., if there are groups of source hosts who usually connect to the same set of target hosts, and so on. It is worth to note that in the effort to find consistent groups able to expose some patterns, we often find outliers, i.e., elements behaving as no other elements in the network. Often these are at least as informative as groups, typically exposing malfunctions, potential threats or poor network management.
- *Alerts as entry point to the whole population (normal vs. abnormal behavior)* – if a system is equipped with some sort of alert generation system (e.g., an IDS) it is true that the whole network traffic can be split into two wide categories: resources involved at least once in suspicious behaviors and the others. If we look at the network through this lens we can recognize interesting opportunities. One is to use the resources involved in suspicious behaviors as an entry point to the whole population. One can isolate suspicious behaviors originating, e.g., from a group of alarms and see if there are other similar behaviors which do not generate alarms. Alternatively, one can compare the typical traffic of a resource and discriminate the traffic that generates alarms to better understand its nature.
- *Tracking and evolution* – While it is always possible to consider the data under inspection as the whole data accumulated so far (besides the obvious computational and scalability problems), we noticed that there is an interest per se in comparing the state of the network before and after some specific moments in time (e.g., the application of a new policy) or even to visualize the evolution through animated visualizations. To this end, it is also important to provide analysts with powerful annotation and tracking tools that permit to easily find their elements of interest and compare their status at different times.

The analysis vision proposed in this article might sound ambitious and we do not plan to solve it in one shot. In the rest of the article, we present three applications that take place in the middle layer of our pyramids and support various analysis tasks corresponding to the ones presented above. Those applications support the idea of reducing network data complexity through visualizations manipulating a reduced number of dimensions, that we call dimension-centric views, enabling to explore network data through specific facets. Further, we believe that there is an adequacy facet/task, i.e., some dimensions are more adequate for supporting some tasks. The following visual applications use a proprietary engine developed by NEXThink S.A.[1], which, differently from most existing engines, is able to convey, other than traditional data (such as IP addresses, ports, etc.), information about applications (in terms of binaries in opposition to protocols or services in most systems) and users (e.g., Windows SID).

4 User and Application Centric Views of the Corporate Network

Classifying users and applications within a company is a big challenge. The goal of this analysis is to visually represent the pertinent information to help the administrator answer this kind of questions:

- What is the typical user in my company? What is her/his typical behavior?
- How many different groups of users do I have in my network?
- Which is the typical profile of a "marketing" user? Has one user a behavior similar to this typical profile?
- Can we cluster applications and identify the family of an unknown one?
- Is there any differentiating pattern between users involved in alarms and users never involved?

We do not intend to completely solve those questions in the following subsections but rather to propose initial designs of dimension-centric visualizations to explore potential capabilities of tools addressing such problems. It is important to make clear from the beginning that some of these prototypes represent only initial designs and therefore their completeness or final effectiveness is not in question here. The following visualizations address similar problems from opposite point of views: while the RadViz aims at plotting similarities, the OriginalityView aims at plotting the uncommon. These two visualizations are examples illustrating the underlying adequacy between task, data facet, and visualization techniques.

4.1 The RadViz: Visually Grouping Similar Objects

In order to visually group similar users, and thus to find profiles, we use a customized version of RadViz (also known as StarCoordinates) (Hoffman et al., 1999;

[1] http://www.nexthink.com

Kandogan, 2001), a common technique to visualize *n*-dimensional datasets to find clusters and outliers. A number of anchors equal to the number of data dimensions are laid out in a circular manner. Each data item is connected to each anchor through a spring whose force is proportional to the value of the given dimension for the given data item. The dots occupy the position where the sum of the forces is equal to zero. As a result, the data points that share common combinations of values across all the dimensions occupy similar positions on the screen, thus segmenting the datasets in groups.

We apply this design to our particular case to find users who have similar application usage behaviors. In our design each user is a data item and each dimension an application with values corresponding to the user-application pair network usage (measured in terms of number of connections). An anchor can represent a class of applications, e.g., browser, email, multimedia, etc., or directly an application, and each dot is a single user in the network. The design is particularly suitable for the task because the visual technique is known to scale well as the number of data dimensions (i.e., applications) increases.

In our custom design we use a bivariate color scheme to distinguish between users who generated alarms, with red hues and increasing brightness as the number of total alarms increase, and users who never generated alarms, with a green hue. The size of dots is proportional to total network activity, i.e., big dots high activity, small dots low activity. The anchors also convey useful information. Their size is proportional to the number of users who use the application and its color to the total activity. In a single view we can isolate groups of users, and compare which applications they use, their activity level, and their alarm level. We can also spot the applications (anchors) that are most used and therefore that influence most the placement of dots on the screen.

The user can interact with the visualization in many ways. The applications/anchors can be filtered out/in to select interesting subsets. A small 2D scatter plot supports the user in this task displaying the anchors on a space where usage and number of users are mapped to the axes. The user can select subsets of anchors and use them to see how the population maps to the selected resources. The anchors can also be moved around the circle and their force can be increased/decreased in order to discriminate between clusters and isolate possible outliers (similar techniques are described in (Kandogan, 2001)).

By selecting one or multiple dots, it is possible to activate a bar chart that compares the usage profile of the selected users. This feature enables to understand what is the usage profile represented by the cluster and how the items inside relate each to another. As an example, in Fig. 3 we have selected the applications with the highest ratio between total activity and number of users. Four applications are markedly influential: *iexplore.exe* (Internet Explorer), *nlnotes.exe* (Lotus Notes), *ldiscn32.exe* (LANDesk Management Suite), *amclient.exe* (LANDesk Application Management Client). *nlnotes.exe* is the one that attracts most of the users (bottom right), as clearly shown on the figure. We can see at least three major groupings: users who mostly use *nlnotes.exe* (bottom right), users using mostly *iexplore.exe* and *ldiscn32.exe*, and those using mostly *ldiscn32.exe* and *amclient.exe*. We can also see that there is a big

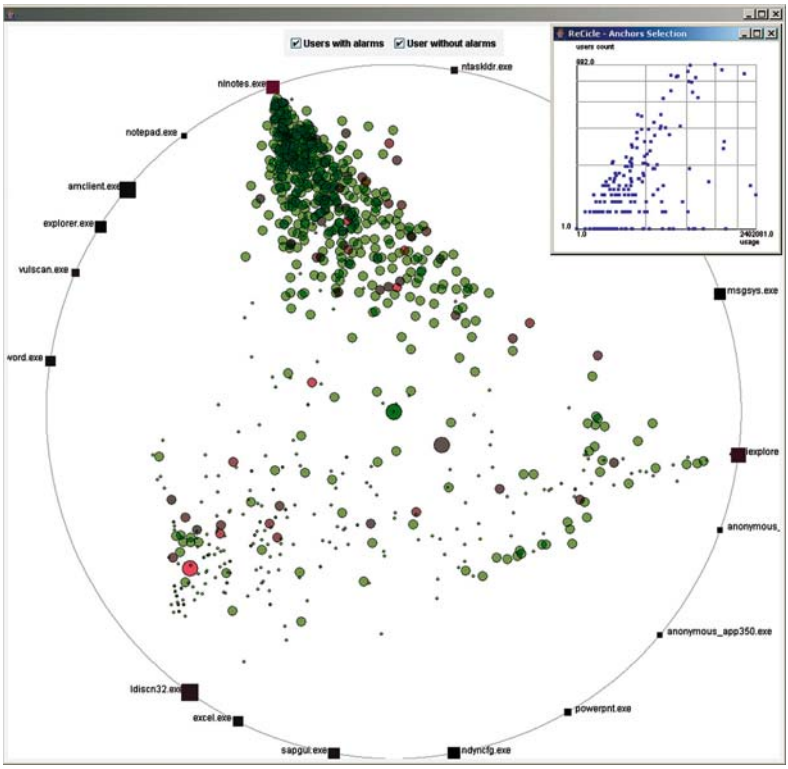

Fig. 3 The RadViz application: plotting similarities

red user on the bottom, quite distant from any other user. This represents the system user that comprehensibly is very distinct from the others: very high number of connections and alarms and an original composition of applications.

There are many others patterns that can be found using the interactive capabilities offered by the tool. Here we just want to give a glimpse of how this visualization can help in spotting groups of users and therefore in building profiles. However, the same technique can reasonably be applied to other combinations of network entities, e.g., to see how users use target hosts, or how applications use network ports. This kind of activity can increase the knowledge of a security analyst and help her/him in formulating novel strategies to apply.

4.2 The OriginalityView: Plotting the Uncommon

The OriginalityView (Fig. 4) exploits the, so called, originality metric as a way to discover original users, detect outliers, and segment user population according to their usage of applications. The originality metric, that we adapted from the

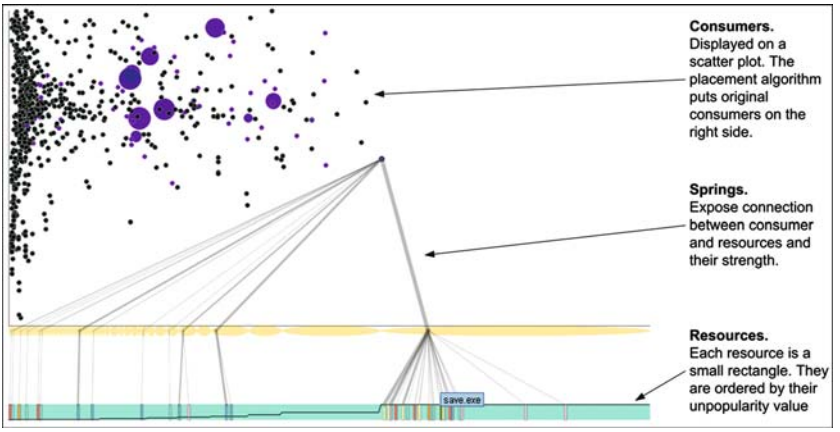

Fig. 4 The OriginalityView: eliciting outliers

standard TF.IDF used in information retrieval (Salton and Buckley, 1988), measures the weight of an application to segment a population of users. In other words, it measures how important an application is to a user in respect to the overall population. This originality metric takes into account the global use of an application, over the number of users that employ it. The corresponding OriginalityView consists of two main parts: the top is a scatter plot of consumers (users in this case) and the bottom an axis of resources (applications in this case). The small colored rectangles in the bottom area are the applications, colored according to the application category they map to (e.g., browser, file transfer, spyware), and ordered by their unpopularity value. The users are represented by the colored dots and their placement on the horizontal axis depends on their barycenter: each user is virtually connected through springs to all the applications he uses and attracted by them with a force proportional to originality value. When a user is selected, the visualization exposes his connections with applications through lines whose width is proportional to the originality value, which represents the force of the spring. This way it explains what makes a user original and draw his profile by highlighting the resources he uses. The given design leaves at least three relevant visual properties available for use. Key visual features like *y*-axis position, size and color can be mapped to other relevant parameters to investigate correlation with the originality. In our figure, as an example, we mapped the *y*-axis to average number of network connections (that is how active a user is), size to the level of access privileges (e.g., system administrator up to standard user), and color to estimated risk. Interestingly, purple users, the one with more associated risk, tend to be also original. Investigating more in details their usage pattern is very beneficial in terms of security findings.

Besides the complexity of the underlying placement algorithm and the absence of a real metric space on the horizontal axis, the final visual result is easy to understand: (1) original users and unpopular applications are on the rightmost side, (2) the lines connecting a selected user with the applications explain her/his originality and usage

profile. Even if the tool has not been evaluated through any formal user study, we gained some evidence of its understandability by showing it to stakeholders in the network security domain: the design principle was very easily understood and the interface can be operated with ease.

5 Alarm/Event Centric Views

The SpiralView is our first attempt to build a tool to analyze network data in large periods of time, in order to address various analysis tasks such as tracking and evolution, segmentation of users, and overview of the network considering a limited proportion of the population, as an entry point to the overall population. In its current implementation, the SpiralView is first of all a tool designed to observe alarms over time, and in an extended way, all types of events, as long as they are aggregated. Alarms are a particular case of events. Events are objects in time or instantiations of properties in objects. In principle, any temporal record on the network can be considered as an event. We also consider events being any known actions that could have an impact on a corporate network such as an awareness program given to a group of employees, or new security policies, etc. While the SpiralView can be used to monitor the network, its primary purpose is to support the analyst in reasoning about how the network evolves and in taking informed decisions on how to administrate it. The focus is shifted from day-to-day monitoring, as a way to spot dangerous events and react, to the analysis of extended periods of time to devise policies that improve the network's behavior. Examples include: better targeted awareness programs, restriction or relaxation of network constraints, redefinition of access rules. To this end, the system also allows to attach notes to alarms or specific moments in time to remember when some strategies have been implemented.

All alarms generated in the system in the last k months are displayed, starting from the oldest in the center up to the most recent in the outer ring. The spiral shape has the following advantages over other time-based visualizations: (1) it can present data sequentially; (2) it exposes periodic behavior through radial alignments of objects; (3) it assigns more space to recent alarms. The perception of time periods in the spiral is extremely important. We decided to use a daily period as a default (that is, one ring represents one day) because this is the most natural way to see alarms in time from the point of view of an administrator, other layouts are available however (e.g., week layout) and might be used to expose periodic behaviors at different time scales. Certain types of network alarms, in fact, tend to be clustered around specific times in the day. The spiral thus follows a 24 h period, starting at midnight in the top, following with 6 am in the right, noon in the bottom, 6 pm in the left. The color of alarms represents their type because it is the most important information administrators use to discriminate between alarms, and corresponds to the same colors displayed in the bar charts for a ready correlation. Their size is mapped to the severity that is the second most important information.

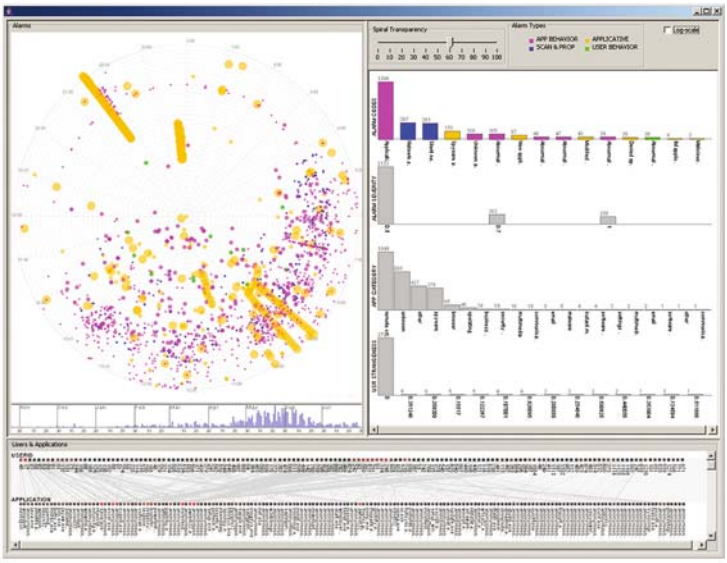

Fig. 5 The SpiralView: Analyzing alarms in time as an entry point to deeper analysis of relationships between network resources

The tool is provided with additional views, coupled with the spiral, to visualize related network resources and attributes. Their design is based on simple interactive bar charts and a custom user/application view which we chose because of their familiarity and ease of use. The bar charts measure the number of alarms falling in each category. As an example, the top bar chart in Fig. 5 presents the number of alarms pertaining to each alarm type category (e.g., network scanning, malicious activities, etc.). The user can select a single or a combination of bars, similarly to brushing histograms (Spence and Tweedie, 1998), to make queries and filter out alarms that are not within specified categories. The tool implements a two-way interaction mechanism. Selection on bar charts filters out and thus segments the set of alarms according to categories of interest. At the same time, interaction with alarms in the spiral enables to select groups and see how they map onto network resources. The spiral is further coupled with a time histogram at its bottom, which is used to convey aggregate data about how the total number of alarms evolves over time. The histogram is also used to select a time period in the spiral and zoom on it. We have also implemented an animated zoom that supports the user in understanding the change of view. When zooming in, each alarm is moved along a radial path and the substrate changes (e.g., the distance between rings grows) to reflect the change in time resolution.

Finally, thanks to annotation capabilities the spiral also serves as a communication tool between administrators and as a tool to keep track of the manual interventions made on the network. Indeed, the analyst can annotate it in order to label alarms or specific times to measure the effectiveness of deliberate interventions.

For instance, an administrator can enter an annotation on the fly, explaining the origin of the highlighted group of alarms and also marking the action undertaken on the engine or on the network to relax this type of alarms. This capability is extremely important in that it permits to remember when certain actions took place and thus to compare the status of the system before and after an intervention. Since the primary purpose of the system is to permit long term analysis and policies' assessment, with annotations not only it is possible to devise new strategies but also to check if and how new rules have changed the network's behavior and to share this knowledge between stakeholders.

The SpiralView's design is the result of various interactions with network security analysts from private companies who already use NEXThink's engine for more than a year. Taking into account real world tasks, they provided us valuable feedbacks on the usefulness of our visualizations and on their usability. The major concern we had to face with this design is the visualization performance and its complexity. Finer grain modifications have been made recently to improve the interaction and readability of the actual SpiralView (zooming mechanisms, brush/link with histograms, etc.) but its performance remains an issue to be solved. More information about the SpiralView can be found in Bertini et al. (2007).

6 Limitations and Challenges

The approach we propose in the article opens numerous challenges and limits to overcome. Visually abstracting relevant information to support trending by managers is not a trivial task. Finding the right balance between usability and completeness, implies finding the right level of abstraction/aggregation (visual vs. data aggregation), but also producing computationally reactive interactive visualizations. For instance, when developing our SpiralView we had to face usability problems, due to the computational time to interact with the view in production with thousands of users. We believe these problems can be often solved with user evaluations of the prototypes produced, and dynamic adaptation of the views to the network topography. However these approaches are time consuming efforts. We believe this problem can be bypassed by limiting the spectrum of an application, with fewer dimensions, and supporting only specific tasks. Similarly, finding the ways to produce fruitful exploration (actionable insights) is a very challenging problem since exploration by definition leads to unknown discoveries, outcomes and thus actions to be taken. For this reason, we believe supporting network administration is an incremental process, that should be supported with simple customizable tools that can be assembled in a toolkit, among which the final user can pick the tools of interest to build her/his own environment. Other challenges include finding communication bridges among the layers of the pyramid so that the different types of users within a company can exchange their findings and decisions. We currently believe that annotations are the best way to capture, store and exchange information, but

more formal representations might be found to be able to search and retrieve this type of information.

7 Conclusion

This article presents a pyramidal approach to network management, showing various levels of time and data granularity, in order to support various types of users: system engineers, analysts and managers. The paper first introduces our vision and discusses the need to reduce the complexity of network data to abstract analysis and trends in time and further to convert decisions into action. The article further introduces envisioned analysis tasks and presents two different ways to support them and to overview network data both concentrating on specific dimensions: users and applications (resources centric) firstly, and alarms (events centric) secondly. The paper presents in particular three visual tools built to support different tasks. We believe these tools are still too close from the tasks performed usually by system administrators and much more efforts are necessary in order to develop tools that support higher level roles and activities such as trending. Even though our approach opens numerous challenges, our major objective with this article is to support the idea that visualization tools can be useful for a broader range of applications related to the administration of a corporate network, that at the end will benefit not only to define adapted security policies but also to improve the understanding of the network usage within the company. Finally, our technical goal is to build a visual customizable toolbox, to bridge the gap between daily network monitoring and strategic trending and decision making.

Acknowledgements We would like to particularly thank Florian Evéquoz for his brilliant implementation of the OriginalityView and the Swiss Innovation Promotion Agency CTI/KTI for financing this project.

References

Abdullah, K., Lee, C., Conti, G., Copeland, J.A., Stasko, J. (2005) IDS RainStorm: visualizing IDS alarms. Proceedings of the IEEE Workshops on Visualization for Computer Security

Allen, J., Christie, A., Fithen, W., Mchugh, J., Pickel, J., Stoner, E. (2000) State of the practice of intrusion detection technologies. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA. CMU/SEI-99-TR-028,CMU/SEI

Bertini, E., Hertzog, P., Lalanne, D. (2007) SpiralView: towards security policies assessment through visual correlation of network resources with evolution of alarms. IEEE Symposium on Visual Analytics Science and Technology (VAST)

Card, S.K., Mackinlay, J.D., Shneiderman, B. (1999) Readings in Information Visualization: Using Vision to Think. Morgan Kaufmann, Los Altos, CA

Conti, G., Grizzard, J., Ahamad, M., Owen, H. (2005) Visual exploration of malicious network objects using semantic zoom, interactive encoding and dynamic queries. Proceedings of the IEEE Workshops on Visualization for Computer Security

D'Amico, A., Kocka, M. (2005) Information assurance visualizations for specific stages of situational awareness and intended uses: lessons learned. Proceedings of the IEEE Workshops on Visualization for Computer Security

Goodall, J.R., Lutters, W.G., Rheingans, P., Komlodi, A. (2005) Preserving the big picture: visual network traffic analysis with TN. Proceedings of the IEEE Workshops on Visualization for Computer Security

Han, J., Kamber, M. (2000) Data Mining: Concepts and Techniques. Morgan Kaufmann, Los Altos, CA

Hertzog, P. (2006) Visualizations to improve reactivity towards security incidents inside corporate networks. Proceedings of the Third International Workshop on Visualization for Computer Security

Hideshima, Y., Koike, H. (2006) STARMINE: a visualization system for cyber attacks. Proceedings of the Asia Pacific Symposium on Information Visualisation – Vol. 60

Hoffman, P., Grinstein, G., Pinkney, D. (1999) Dimensional anchors: a graphic primitive for multidimensional multivariate information visualizations. Proceedings of the 1999 Workshop on New Paradigms in Information Visualization and Manipulation

Kandogan, E. (2001) Visualizing multi-dimensional clusters, trends, and outliers using star coordinates. Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining

Keim, D.A. (2000) Designing pixel-oriented visualization techniques: theory and applications. IEEE Transactions on Visualization and Computer Graphics **6**: 59–78

Koike, H., Ohno, K. (2004) SnortView: visualization system of snort logs. Proceedings of the 2004 ACM Workshop on Visualization and Data Mining for Computer Security

Lakkaraju, K., Yurcik, W., Lee, A.J. (2004) NVisionIP: netflow visualizations of system state for security situational awareness. Proceedings of the 2004 ACM Workshop on Visualization and Data Mining for Computer Security

Mansmann, F., Vinnik, S. (2006) Interactive exploration of data traffic with hierarchical network maps. IEEE Transactions on Visualization and Computer Graphics **12**: 1440–1449

Ramakrishnan, R., Gehrke, J. (2000) Database Management Systems. McGraw-Hill, New York, USA

Salton, G., Buckley, C. (1988) Term-weighting approaches in automatic text retrieval. Vol. 24. Pergamon Press, New York, pp. 513–523

Spence, R., Tweedie, L. (1998) The attribute explorer: information synthesis via exploration. Interacting with Computers **11**: 137–146

Stolte, C., Tang, D., Hanrahan, P. (2002) Query, analysis, and visualization of hierarchically structured data using Polaris. Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining

Takada, T., Koike, H. (2002a) MieLog: a highly interactive visual log browser using information visualization and statistical analysis. Proceedings of the 16th USENIX Conference on System Administration

Takada, T., Koike, H. (2002b) Tudumi: information visualization system for monitoring and auditing computer logs. Sixth International Conference on Information Visualisation (IV'02)

Yarden, L., Jim, A., Shaun, M., Stefano, F. (2005) Visual correlation for situational awareness. Proceedings of the Proceedings of the 2005 IEEE Symposium on Information Visualization

Yin, X., Yurcik, W., Treaster, M., Li, Y., Lakkaraju, K. (2004) VisFlowConnect: netflow visualizations of link relationships for security situational awareness. Proceedings of the 2004 ACM Workshop on Visualization and Data Mining for Computer Security