Visual Spam Campaigns Analysis using Abstract Graphs Representation[*]

Orestis Tsigkas
Centre of Research &
Technology Hellas
Information Technologies
Institute
Thessaloniki, Greece
torestis@iti.gr

Olivier Thonnard
Symantec Research Labs
2229 Route des Crètes
Sophia Antipolis, France
Olivier_Thonnard@symantec.com

Dimitrios Tzovaras
Centre of Research &
Technology Hellas
Information Technologies
Institute
Thessaloniki, Greece
tzovaras@iti.gr

ABSTRACT

In this work we present a visual analytics tool introducing a new kind of graph visualization that exploits the nodes' degree to provide a simplified and more abstract, yet accurate, representation of the most important elements of a security data set and their inter-relationships. Our visualization technique is designed to address two shortcomings of existing graph visualization techniques: scalability of visualization and comprehensibility of results. The main goal of our visual analytics tool is to provide security analysts with an effective way to reason interactively about various attack phenomena orchestrated by cyber criminals. We demonstrate the use of our tool on a large corpus of spam emails by visualizing spam campaigns performed by spam botnets. In particular, we focus on the analysis of spam sent in March 2011 to understand the impact of the Rustock takedown on the botnet ecosystem. As spam botnets continue to play a significant role in the worldwide spam problem, we show with this application how security visualization based on abstract graphs can help us gain insights into the strategic behavior of spam botnets, and a better understanding of large-scale spammers operations.

Keywords

information visualization, network security, attack campaigns, security intelligence

1. INTRODUCTION

Cyber security analysts should be aware as quickly as possible of the emerging strategies used in cyber-crime and how these evolve over time. Increasing our understanding of ongoing and emerging attack campaigns has also the potential for improving our defense and remediation mechanisms, thanks to a better understanding of attackers behaviors and tactics. Towards this goal, security visualization, when done appropriately, can help achieve an increased level of *situation perception*, for example by effectively highlighting various attack campaigns orchestrated by the same team of attackers. Similarly, data *clustering* can be an effective step towards analysing data in an exploratory way. Identifying clusters in a security data set can provide interesting viewpoints on the phenomena being analyzed [22].

By grouping patterns (observations, data items, or feature vectors) into homogeneous clusters, clustering can shed some light on the root causes of attack phenomena. In security and threat monitoring, there is little prior information available about the data (i.e., no “ground truth”), and the decision-maker (or the analyst) must make as few assumptions about the data as possible. It is under these restrictions that a clustering methodology is particularly appropriate for the exploration of inter-relationships among the security events to make an assessment of their underlying structure.

However, the effectiveness of both analytical clustering and visualization techniques is greatly challenged by large and high-dimensional data. Large data sets often cause cluttered visualizations when the complete raw data is being represented, which makes it difficult for a user to understand an overview of the data and then determine which parts of the data can be filtered or zoomed in for more detail. Moreover, large data sets slow down the user interaction, making the data exploration process laborious.

In this paper, we develop an interactive graph-based visualization method designed to be used for security analysis, which aims to “connect all dots” between security events by providing summarized representations of multi-dimensional clusters of security events which are likely to be due to the same phenomenon. Our visualization method leverages an *abstract graph* representation designed to address two shortcomings of existing graph visualization techniques: *scalability* of the visualization and *comprehensibility* of the results. Our approach is based on a transformation technique for k-partite graphs which exploits the nodes' *degree* to provide a simplified, aggregated and more abstract representation that focuses on the most important clustering elements of the data set and their inter-relationships. As a result, the

[*]This work has been partially supported by the European Commission through project FP7-ICT-257495-VIS-SENSE funded by the 7th Framework Programme (FP7). The opinions expressed in this paper are those of the authors and do not necessarily reflect the views of the European Commission.

VizSec '12, October 15 2012, Seattle, WA, USA

presented tool enables the analysis of large network data structures, revealing the complex relationships between cluster entities, as well as any relevant attributes, statistical patterns or groupings associated with them. The proposed visualization reduces significantly the number of items to be displayed and presents an abstraction of the data set. Yet the visual analysis remains accurate, as the graphical aggregation preserves the significant features present in the original data, and the incorporation of an *importance metric* on graph edges enables the user to generate abstract graphs at varying levels of details.

To demonstrate the use of our tool, we provide a case study on a comprehensive spam data set from 2011. It is commonly agreed today, that, over the last decade, unsolicited bulk email spam has evolved dramatically in its volume, its delivery infrastructure and its content [24]. Multiple reports indicate that approximately 70-80% of all email traffic traversing the Internet today is considered to be spam [19]. Today spammers have turned their illegal activities into a very lucrative business and most of the spam campaigns are being distributed through very large spam botnets, *i.e.* large-scale networks of compromised computers that are controlled by cyber criminals and used for various profit-oriented activities such as spam [24].

Therefore, we provide various visualizations of spam campaigns observed in March 2011, by which we analyze the impact of the `Rustock` takedown [20] on the spam botnet ecosystem. With this application on a real-world email data set, we aim to show how our visualization tool helps get insight into the way those attack campaigns are being organized by well-resourced and motivated spammers. While we have primarily focused in previous works on the *analytical* aspects of those spam campaigns and how to identify them [25, 24], in this paper we focus instead on the security visualization aspects, and we introduce a new visual analytics approach based on *abstract graphs* to represent them in condensed, yet effective manner.

The rest of the paper is structured as follows. In Section 2 we review some related works in visualization methods designed for the representation of multi-dimensional clusters and high-dimensional data. Section 3 presents our abstract graph-based visualization technique and its associated user visual interaction. Then, in Section 4 we present our case study on spam email analysis to illustrate the use of our visualization tool. Section 5 concludes this paper.

2. RELATED WORK

The two most prevalent algorithms for clustering of high-dimensional data, which can also be visually represented, are Multidimensional Scaling (MDS) [14] and Self Organizing Maps (SOM) [13], [26]. Both algorithms actually aim at producing low dimensional views of high dimensional data. What makes them appropriate for visual cluster analysis is the fact that they approximately preserve the topological properties of the high-dimensional input. However, the produced visualizations are, by themselves, not sufficient to explain the outcome of the clustering process. This is due to the fact that the operation of the algorithms is based on a dimensionality reduction process and the new dimensions can be difficult to interpret, making it hard to understand clusters in relation to the original data space. Moreover, these techniques are not effective in identifying clusters that may exist in different subspaces of the original data space [1].

Parallel coordinates is a widely used visualization technique for exploring large, multidimensional data sets [10]. It is powerful in revealing a wide range of data characteristics such as different data distributions and functional dependencies [11]. However, one of the limitations with parallel coordinates is the cluttered visualization problem caused by rendering more polylines than available pixels. Overlapped lines often obscure the underlying patterns of the data, especially in areas with high data density.

Angular histograms and attribute curves, which are frequency based approaches to large, high-dimensional data visualization, are presented in [8]. Angular histogram and attribute curves offer an intuitive way for the user to explore the clustering, linear correlations and outliers in large data sets without the over-plotting and clutter problems associated with traditional parallel coordinates. These techniques consider each polyline-axis intersection as a vector. Both the magnitude and direction of these vectors are visualized to demonstrate the principle trends of the data. Users can dynamically interact with the plot to investigate and explore additional patterns.

To facilitate cluster interpretation, evaluation, and comparison, DICON is proposed in [3], which is an icon-based cluster visualization that embeds statistical information into a multi-attribute display. A treemap-like icon is designed to represent a multidimensional cluster, and the quality of the cluster can be conveniently evaluated with the embedded statistical information. A novel layout algorithm is further developed which can generate similar icons for similar clusters, making comparisons of clusters easier. User interaction and clutter reduction are integrated into the system to help users analyze clustering results for large datasets.

To handle large and inhomogeneous data spaces, the VisBricks concept is proposed in [15]. The main advantage of VisBricks is their ability to handle all types of inhomogeneities within data, both in the dimensions and in the records. At the very core of the VisBricks strategy are two concepts: the creation of homogeneous sub-parts of the data and the establishment of multiform visualization for those parts. This is achieved by treating each homogeneous subpart of the data with the best available computational and visual tools. By using abstractions in the bricks, VisBricks are very scalable in terms of the magnitude of records. At the same time, the division into bricks and the rich set of interaction patterns allow users to employ multi-level approaches, in which each brick contains an abstraction suitable to show the data at the desired level of detail.

In [6] a technique called motif simplification is introduced. Motif simplification leverages the repeating motifs in networks to reduce visualization complexity and increase readability. Two frequently occurring and high-payoff motifs: a fan motif consisting of a fan of nodes with only a single neighbor connecting them to the network, and a parallel motif of functionally equivalent nodes that span two or more other nodes together. The authors contribute the design of representative glyphs for these motifs, algorithms for detecting them, a publicly available reference implementation, and initial case studies and user feedback that support the motif simplification approach.

Multi-dimensional data structures that can be represented by undirected graphs can be visualized by calculating simple layouts that belong to a class known as force-directed algorithms [7]. Also known as spring embedders, such algorithms calculate the layout of a graph using only information contained within the structure of the graph itself, rather than relying on domain-specific knowledge. Graphs drawn with these algorithms tend to be aesthetically pleasing, exhibit symmetries, and tend to produce crossing-free layouts for planar graphs. Their purpose is to position the nodes of a graph in two-dimensional or three-dimensional space so that all the edges are of more or less equal length and there are as few crossing edges as possible.

Forced-directed algorithms can be incredibly time consuming to do well. Therefore, a lot of effort has been put into developing automated network layout algorithms that have lower complexity than $\mathcal{O}(N^2)$ [2, 5]. Even in this case, the results of applying a layout algorithm can vary greatly depend on the size and topology of the network, and the layout generated is highly dependent on the algorithm used. Moreover, forced-directed algorithms usually converge to poor local minima which can be, in many cases, considerably worse than a global minimum, resulting in a low-quality drawing and reduced comprehensibility of the results. Additionally to force-directed algorithms, *edge-bundling* was introduced in [9] as a method for visualizing complex node-link graphs while reducing at the same time visual clutter. In contrast to these previous methods, our visualization technique introduces an abstract graph representation that aims at *simplifying* node-link graphs by automatically aggregating information to highlight only the most important elements.

3. ABSTRACT GRAPH VISUALIZATION

To enable the interactive exploratory analysis of a wide range of security datasets, we aim at building a visual analytics tool that facilitates the visual analysis of multidimensional interconnections among security events that are attributed to the same *attack campaign*. Security events visualized by our tool can refer to a large variety of phenomena, such as spam botnet operations [24], targeted attacks [23], malware propagation, rogue AV campaign, etc. The ultimate goal is to get a better comprehension of the root causes behind the structure and the organization of those attacks.

As introduced before, the developed tool is based on abstract graph representation. It aims to address two shortcomings of existing graph visualization techniques: scalability of visualization and comprehensibility of results. The produced visualization can be easily combined with analytical techniques for clustering and attack attribution [22].

The analysis of security datasets involves data objects of multiple types that might be related to each other, which can be naturally formulated as a k-partite graph. For example, spam emails are related to geolocation and IP address of the spammers, subject of the email, etc. However, the research on mining the hidden structures from a k-partite graph is still limited and preliminary [17].

Therefore, our research work aims at proposing a general visual analytics model, to find the hidden structures in datasets that can be represented by a k-partite graph. The model aims at providing a principal framework for unsupervised learning on k-partite graphs of various structures. Under this model, our objective is to identify the hidden structures of the graph by identifying strongly connected nodes using neighbourhood information. The novelty and strength of our approach resides in its ability to incorporate multiple features and searching for clusters in the multidimensional space.

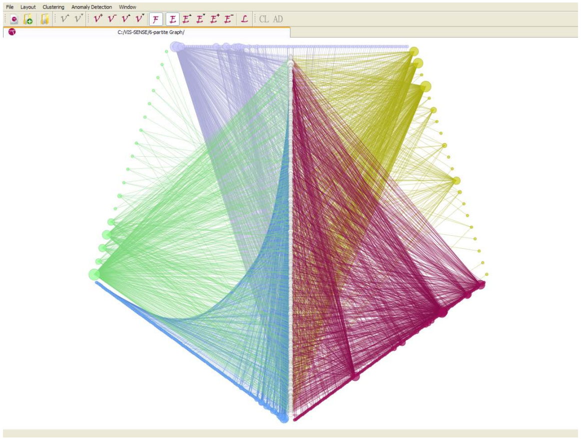


Figure 1: An example of 6-partite graph, which is formed by considering all relationships among security events (*white* nodes in the middle) with respect to 5 different features (places on the sides of a pentagon using a different colour for each).

3.1 Building the k-partite Graph

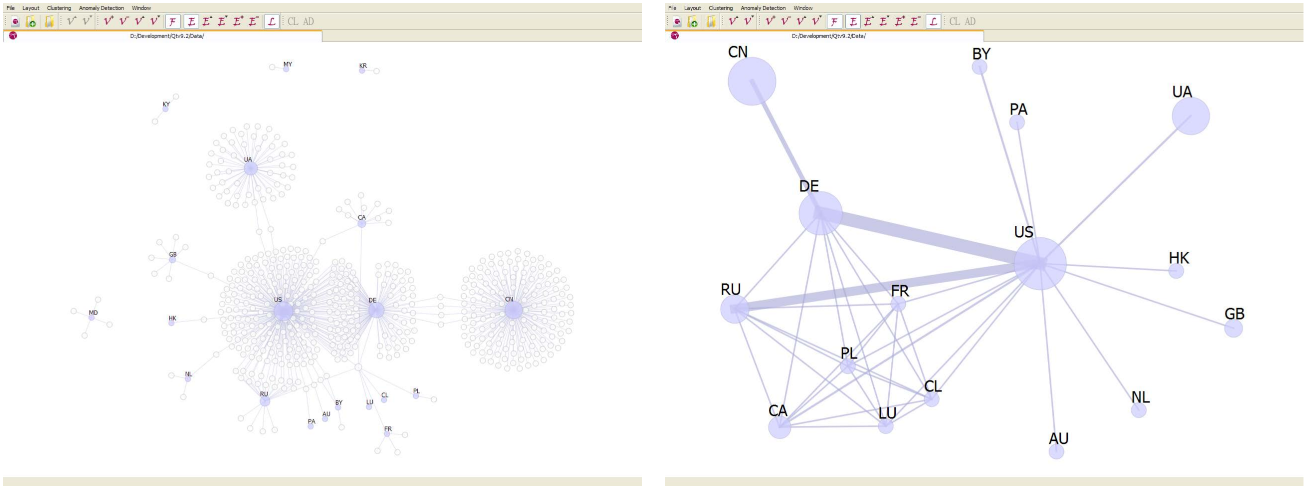
A k-partite graph can be built by considering the relationship of security events (e.g. spam e-mails) with the feature values from all considered attack features.

A k-partite graph is a graph where nodes can be divided in k disjoint groups $(V_0, \ldots, V_{k-1})$, such that no edge connects the vertices in the same group. More formally, an edge weighted k-partite graph is a pair (G, w), where $G = \langle V_0 \cup \ldots \cup V_{k-1}, E \rangle$, where $V_l = \{n_i | 1 \leq i \leq N_l\}$, $\forall l \in [0, k-1]$, $E \subset \bigcup_{l=1}^{k-1} \{V_0 \times V_l\}$, and $w : E \rightarrow \mathbb{R}$ is a weight function. The weight of an edge shows the number of occurrences of a connection of a security event with a specific feature value in the raw data. Nodes in V_0 correspond to security events, while nodes in $V_{l \neq 0}$ correspond to feature values. Therefore, a k-partite graph shows the connections of security events to feature values of $k-1$ different attack features.

An edge-weighted 6-partite graph is presented in Fig. 1. Security events (nodes in V_0) are presented with white circles, while nodes in $V_{l \neq 0}$ which correspond to feature values of different features are represented using a different colour for each feature. A simple geometric layout is used, where the nodes corresponding to the feature values of the 5 different attack features are placed on the sides of a pentagon and security events are placed in the middle of the pentagon. Nodes in $V_{l \neq 0}$ can have edges only with nodes in V_0. While such a geometric layout of security events and feature values can provide an overview of the most important nodes in the graph, it is inadequate for describing the interrelationships among different security events and feature values.

3.2 Abstracting Visual Information

Since force-directed algorithm suffer from high complexity and are insufficient to consistently produce understandable network visualizations [16], we aim at developing an abstract graph visualization tool that will result in fast layouts and


File Layout Clustering Anomaly Detection Window
D:/Development/Qt/v9.2/Data/
MY
KR
KY
UA
CA
GB
US
DE
CN
MD
HK
NL
RU
BY
AU
PA
LU
CL
PL
FR
File Layout Clustering Anomaly Detection Window
D:/Development/Qt/v9.2/Data/
CN
BY
PA
UA
DE
US
HK
RU
FR
GB
PL
CL
CA
LU
NL
AU

(a) Visualization of the bipartite graph corresponding to the connections between spam emails sharing the same subject (white nodes) and the origin country (purple nodes)

(b) An abstract graph visualization is created by considering only nodes corresponding to feature values (i.e. country of spam email origin) and their interconnections

Figure 2: Abstract graph visualization of the interconnection of 500 security events (i.e. spam emails) with the 20 corresponding countries of origin

increased readability, which will also facilitate the analysis of large security datasets and attack campaigns by revealing the complex relationships between entities, as well as any attributes, statistics, or groupings associated with them.

A visually appealing result of force-directed algorithms is that they let visual clusters of security events to be created, allowing the security analyst to easily inspect the relationship and shared connections among a group of events. A visual cluster is created either when different security events share many (low weight) connections to the same set of nodes representing feature values or have edges with a large weight to a small number of nodes. Therefore, visual graph clustering results in partitioning the vertices of a graph into disjoint sets such that each partition is a well-connected and coherent group (i.e. cluster).

However, the simplicity and beauty of force-directed layouts turns into clutter and confusion when the number of nodes and links gets too high. This is due to the fact that many nodes (and their corresponding outgoing and incoming edges) present little meaningful information, yet dominate much of the display space and can obscure the more interesting parts of the network. Hence, we believe that replacing these nodes and their edges with a more abstract graph visualization will create network visualizations that require less screen real estate, and will make the role of each node easier to understand in the global context of the network.

In examining the k-partite attack graph that is formed by considering the connections of security events (nodes in V_0) with different feature values, as in Fig. 2a, we observe that the nodes representing security events dominate a large part of the display, yet the information they bear can easily be represented in a more abstract form. The visual representation of security events nodes actually conveys two pieces of information to the security analyst: (i) the volume of security events associated with a single feature value and (ii) the volume of security events that are linked to two or more feature values and are, therefore, responsible for the creation of interconnections and interdependencies among different nodes in the graph.

In Fig. 2 the bipartite graph corresponding to the origin country of spam emails with the same subject is shown. Fig. 2a makes evident the fact that there are certain countries which exhibit higher activity than others and that there is a high number of spam emails with the same subject originating from different (highly active) countries. Moreover, we can observe that many spam emails originate both from a specific small country and the USA, probably due to the fact that English is a widely spoken language. The security analyst can still easily distinguish clusters of spam emails that originate from a specific country and other clusters that are linked to two or more countries, thanks to the rather low number of spam represented here.

Yet, the same graph can be visualized using abstract graph visualization, as shown in Fig. 2b. The information concerning the volume of spams originating form each country is conveyed by the size of graph nodes, while the interconnections between country nodes, which reflect the number of spams having the same subject and originating from the same set of countries, has become even more obvious. Moreover, Fig. 2b makes now evident an observation that was not easily conceivable with the visualization of the whole graph: there is a number of smaller countries that are closely related to each other with regard to spam emails having the same subject, while some of them share connections with the USA as well.

3.3 Defining Different Levels of Abstraction

While the abstract graph visualization already increases the readability of the graph by significantly reducing the number of nodes and the associated edges, visualizing an excessive number of feature value nodes and their interconnections is not so helpful for a security analyst, as it can distract him in the overall analysis of the network representing attack campaigns.

In fact, we observe that certain feature value nodes can be less informative, in that they do not reveal important information on how a specific attack campaign is designed and launched, since few security events are related to these feature values. While they might still contain important information that could be useful with regard to a specific attack campaign, they are also distracting to a human analyst and often hide the root causes of the security problem under study.

We refer to nodes and their corresponding edges that are not immediately useful for a human analyst to understand the underlying root causes of a security phenomenon as "superfluous" security events and edges. This classification of "informative" and "superfluous" edges is meant to reflect a *prioritization* of the data at a given level of abstraction defined by the user. Obviously, these superfluous nodes are still valid and may be important to consider when closely studying all aspects of a specific attack campaign and determining upon appropriate countermeasures. However, when the analyst is first presented with the graph representing the attack campaigns under study, understanding all interconnections (also for those less important nodes) is probably not crucial for shedding light on the ongoing attack phenomena. Hence, we believe that it is beneficial to first present the user with an even more abstract graph, so that she can quickly gain insights without being overwhelmed with low-level information.

We now provide further simplification of abstract graphs by enabling the definition of several levels of abstraction of k-partite graphs. To do this, we assign an *importance metric* $I_e^{s \to t}$ to each edge e linking source node s with target node t, as follows:

$$I_e^{s \to t} = E^s \frac{S^{s \to t}}{N^s},$$

where E^s is the number of edges of node s in the abstract graph, $S^{s \to t}$ is the number of nodes connecting node s with node t in the k-partite graph and N^s is the number of security events associated with node s in the k-partite graph. It is important to note that, such a definition of the importance metric ensures that, at a higher level of abstraction, the security analyst will be also presented with feature values that are related to a lower volume of security events but are well connected with each other and possibly explain a single attack campaign.

Once the importance score of each edge is defined, we calculate the z-score of the importance metric of each edge e.

$$z_e = \frac{I_e^{s \to t} - \mu}{\sigma},$$

where μ is the mean of the importance metric of all edges and σ the standard deviation. Then, the level of abstraction of the graph can easily be increased by removing all edges that have a z-score below a given threshold. Nodes that are not connected to any other node are omitted from being displayed as well. The analyst can increase the threshold in small steps, until he can have a clear overview of the presented information.

3.4 User Visual Exploration

Visual exploration often follows the *information seeking mantra*: "analyze first and show the important; zoom, filter and analyze further; provide details on demand" [18, 12]. Any visual analytics application should support these basic tasks. In order to assist security analysts in correlating findings and insights, our developed tool should go beyond individual graphical representations fulfilling a number of non-functional requirements: smooth user interaction, easy data exploration through filtering and zooming, iterative refinement and readable presentation of the results.

Supporting visual analytics of multiple large-scale multi-dimensional datasets requires a high degree of interactivity and user control beyond the conventional challenges of visualizing such datasets. Therefore, we enable the security analyst to interact with the produced visualization by repositioning the nodes of the graph, parametrizing on the fly the force-directed algorithm to enhance the visualization and making use of standard tools for visual exploration like zoom in and out, fish-eye effect, etc. The purpose of the abstract graph visualization is to allow users to create advanced visual queries by iteratively selecting and filtering into the multidimensional data. This results in an enhanced visual clustering of graph nodes, where the produced visualization reflects better the relationships among security events across all dimensions.

4. VISUALIZING SPAM CAMPAIGNS

We make use of the proposed abstract graph visualization tool for the analysis of spam email campaigns observed in 2011. Specifically, we pay attention to the spam activity in March 2011, when the takedown of the "Rustock" botnet took place. Specifically, between March 16th and March 17th, 2011, many Rustock C&C servers located in the U.S. were seized and shut down by federal law enforcement agents, as a result of a coordinated anti-botnet action led by Microsoft, FireEye and U.S. federal authorities [24].

Some observations and hypotheses regarding the impact of this phenomenon were made by Thonnard et al in [24]. By using analytical methods, the authors were in the position of doing a strategic analysis of spam botnet operations and have drawn several conclusions about the impact of the Rustock takedown on the spam botnet ecosystem. It is important to notice that their observations generally contradicted the reports from other analysts. Consequently, we aim at using the developed tool to test and validate or reject the following hypotheses:

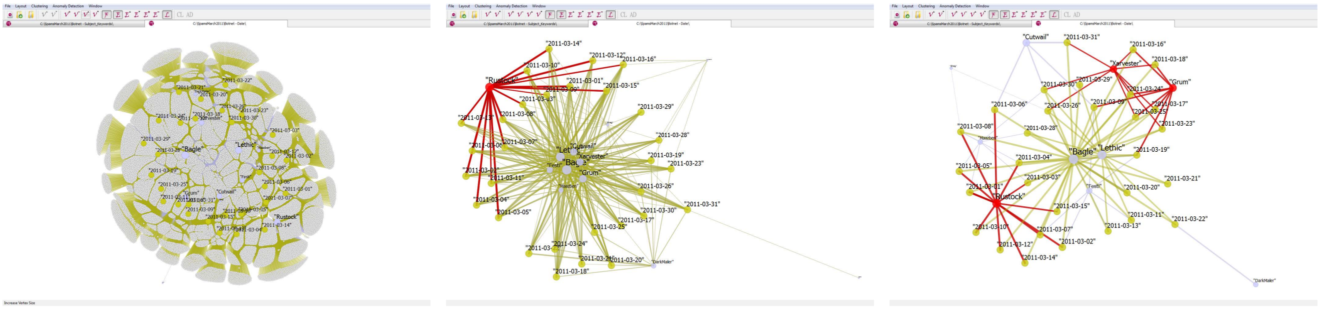
1. Rustock takedown: even though Bagle took over the role in terms of spam volume, Grum had more likely taken over the Rustock spam campaigns just after its shutdown (due to similarities between their respective campaigns at that time).

2. Botnets inter-relationships: some family of botnets are performing very similar spam campaigns, probably to load-balance their activities.

3. Campaign dynamics were different for botnet families: Rustock/Grum/Cutwail had similar behavior, whereas Lethic/Bagle had a more dynamic (polymorphic) behavior.

4.1 Spam Emails Dataset

The spam data set was provided by Symantec.cloud and has been collected through worldwide distributed spamtraps. All email traffic sent to these spamtrap domains is analyzed



(a) The 3-partite graph corresponding to the connection of spam emails with the sending botnets and the day that they were sent

(b) The corresponding abstract graph including all edges (Rustock and its edges are highlighted in red)

(c) The abstract graph at a higher level of abstraction (Rustock, Grum and Xarvester and their edges highlighted in red)

Figure 3: Abstract graph visualization of March 2011 spam campaigns

by honeypots that extract many different features from the emails, including email headers, message content, sender's IP address, name of the bot (if available from CBL [4] rules), the embedded URIs, etc. Because of the overwhelming volume of messages, the spamtrap traffic is sampled on a daily basis, with about 10,000 random samples stored each day in a separate database, which serves as a baseline for the statistical analysis of spam and botnet traffic [21]. During the whole year 2011, a total of 3,111,140 spam messages were collected, parsed and stored in the database, whereas 336,921 messages were collected only for March 2011.

Every spam record of the data set is characterized by a number of features extracted from the email header and body:

- *email features*: "From" and "To" addresses (*i.e.*, sender and recipient), subject line, timestamp, set of embedded URIs (if any), character set and language used, attachment names and message size.
- *sending host features*: source IP address, hostname of the sending machine, country and geolocation data (IP geo-mapping).
- *bot-related features*: bot name (*e.g.* "Rustock"), OS details (*e.g.*, "XP SP1+") as obtained from passive OS fingerprinting.

A more detailed description of those spam features and their relative importances or usefulness for clustering spam messages is provided in [24].

4.2 Spam Campaign Analysis

We begin our spam campaign analysis with the visual validation of the Rustock takedown. Fig. 3 presents the 3-partite graph formed by considering the interconnections of spam emails with the sending botnet and the day when the spam email was sent. In Fig. 3a, the whole 3-partite graph is visualized. Although we can certainly observe that certain areas of the graph are dominated by nodes corresponding to specific feature values, the interconnections of spam emails to specific botnets and dates are far from obvious.

The distillation of this information using the abstract graph visualization in Fig. 3b allows the security analyst to obtain an overview of the operation of different botnets on specific dates without being overwhelmed with superfluous information. The structure of the abstract graph makes evident the fact that botnets operating the whole course of March 2011 are located in the center of the graph, while botnets operating on specific dates are positioned outside the ring formed by nodes representing dates (yellow nodes). Fig. 3b makes apparent the fact that Rustock is highly active only until the 15th of March.

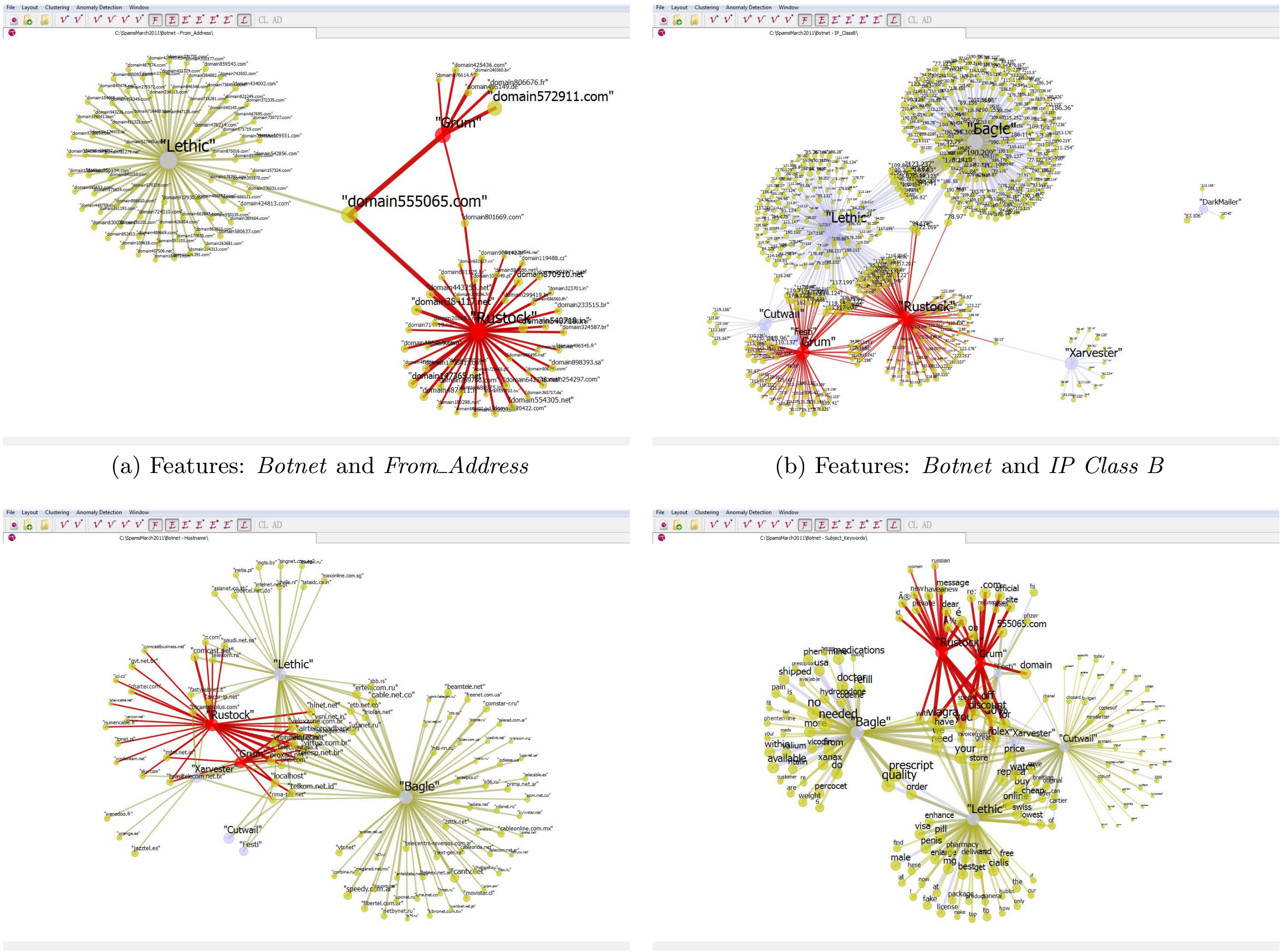
By increasing the level of abstraction as shown in Fig. 3c, we can focus on the botnets with the largest activity and the dates that this activity was observed. While Bagle exhibits a high activity during the whole course of March 2011, there are two botnets (Grum and Xarvester) that exhibit their highest activity just after Rustock was taken down. This observation forces us to examine more features that are related to the carried spam campaigns in order to validate the observation that Rustock campaigns were run by Grum (and not Bagle) after its takedown.

In Fig. 4a, the abstract graph corresponding to the *Botnet* and the *Sender Address* is shown. Grum and Rustock are strongly connected to two specific domains, while Bagle shares no connections with Rustock and is therefore not shown in the figure. Rustock is also seen to have a strong connection to a common node with Lethic. Moreover, Lethic exhibits a high number of associated sender addresses.

The high number of common characteristics between Rustock and Grum becomes evident by visualizing the abstract graph corresponding to *sending host features* as well. Both the abstract graph corresponding to features *Botnet - IP ClassB* (Fig. 4b) and the graph graph corresponding to features *Botnet - Hostname* (Fig. 4c) make apparent the fact that, besides being strongly connected with each other, Rustock and Grum share also many connections with Lethic and Cutwail respectively.

Last, the most frequent keywords contained in the subjects of the spam messages sent by all botnets are shown in Fig. 4d. Once again, we observe the strong interconnections between Grum and Rustock, but also between Rustock and Festi, meaning that Grum was probably the botnet that carried Rustock's spam campaigns after its take down. Festi seems that may have assisted Grum as well, after the Rustock takedown. However, the volume of spams sent by Festi is much lower than that of Grum.

Due to space limitations, we do not present the results



(a) Features: *Botnet* and *From_Address*

(b) Features: *Botnet* and *IP Class B*

(c) Features: *Botnet* and *Hostname*

(d) Features: *Botnet* and *Subject Keywords*

Figure 4: Abstract graph visualization of the interconnections of spam emails with the sending botnet and 6 different features. Rustock and Grum are highlighted in red.

with regard to features: *Charset*, *Country* and *Recipient Addresses*. However, we notice that similar patterns are observed for all these features, where Rustock and Grum have connections to many common nodes and they also share connections with Lethic and Cutwail respectively. Moreover, Lethic and Bagle exhibit the highest number of linked feature value nodes.

Finally, Fig. 5 reveals the big picture and shows the abstract graph that is created by considering the interconnections of spam messages along 8 dimensions: *Botnet, Subject Keywords, Uri Domain, Country, Recipient Address, Charset* and *Hostname*. We can clearly see that both Rustock and Grum are located in the same dense area of the graph since they share many common connections. This is a clear indication that the spam campaigns launched by both botnets share many commonalities. Evidently, the common characteristics between spam campaigns carried by Rustock and Grum are much more than those between campaigns carried by Rustock and Bagle.

Moreover, we can observe that Cutwail and Festi exhibit many similarities with Grum along many different dimensions. Lethic and Xarvester have, in many occasions, common connections with the group (Rustock / Grum / Cutwail / Festi) but they seem to have a more autonomous operation. During March 2011, Bagle exhibited the most self-standing (fewest commonalities with all other botnets) behavior of all. Furthermore, we can notice that, as stated in [24], Bagle and Lethic seem to exhibit the most dynamic behaviour of all botnets validating the hypothesis that they had the most polymorphic behavior.

5. CONCLUSIONS

In this paper, we presented a graph-based interactive visual analytics tool to assist security analysts in reasoning in an effective way about various attack phenomena observed in the Internet. The developed tool exploits the nodes' degree in the graph to provide a simplified and more abstract, yet accurate, representation of the most important elements of a security data set and their inter-relationships. By applying it to a large corpus of spam emails we were able to visually test and validate several hypotheses following Rustock's take down in March 2011 and we gained insights into the strategic behavior of spam botnets and spammers

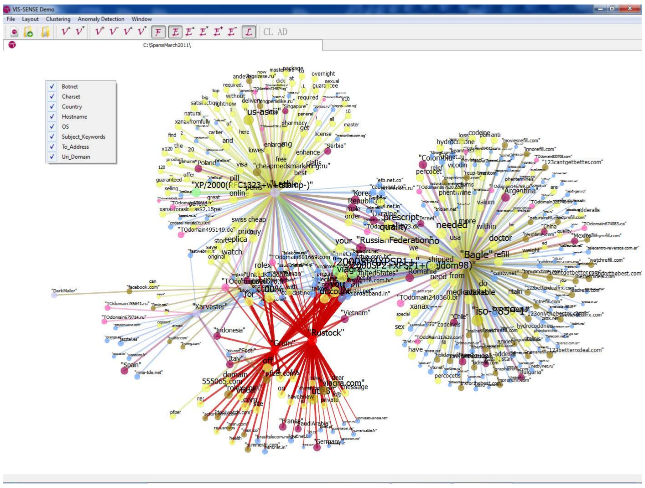


Figure 5: The big picture that is created by taking into account 8 features of spam emails sent during March 2011: *Botnet, Subject Keywords, Uri Domain, Country, Recipient Address, Charset and Hostname.* Rustock and Grum are higlighted in red.

operations. Although the presented results focused mainly on spam emails analysis, our visualization method provides security analysts with a generic visual analytics tool for reasoning interactively about coordinated attack campaigns orchestrated by cyber criminals.

6. REFERENCES

[1] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. In *Proceedings of SIGMOD '98*, pages 94–105, 1998.

[2] J. Barnes and P. Hut. A hierarchical O(N log N) force-calculation algorithm. *Nature*, 324(6096):446–449, Dec. 1986.

[3] N. Cao, D. Gotz, J. Sun, and H. Qu. DICON: Interactive Visual Analysis of Multidimensional Clusters. *IEEE Trans. Vis. Comput. Graphics*, 17(12):2581–2590, Dec. 2011.

[4] Composite Blocking List. http://cbl.abuseat.org.

[5] J. Dubinski. A parallel tree code. *New Astronomy*, 1(2):133–147, 1996.

[6] C. Dunne and B. Shneiderman. Motif Simplification: Improving Network Visualization Readability with Fan and Parallel Glyphs. Technical Report HCIL-2012-11, University of Maryland, May 2012.

[7] T. M. J. Fruchterman and E. M. Reingold. Graph Drawing by Force-Directed Placement. *Softw. Pract. Exper.*, 21(11):1129–1164, Nov. 1991.

[8] Z. Geng, Z. Peng, R. S.Laramee, J. C. Roberts, and R. Walker. Angular Histograms: Frequency-Based Visualizations for Large, High Dimensional Data. *IEEE Trans. Vis. Comput. Graphics*, 17(12):2572–2580, Dec. 2011.

[9] D. Holten. Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data. *IEEE Trans. Vis. Comput. Graphics*, 12(5):741–748, Sept. 2006.

[10] A. Inselberg and B. Dimsdale. Parallel coordinates: A tool for visualizing multi-dimensional geometry. In *Proceedings of VIS '90*, pages 361–378, 1990.

[11] D. Keim and H.-P. Kriegel. Visualization Techniques for Mining Large Databases: A Comparison. *IEEE Trans. Knowl. Data Eng.*, 8(6):923 –938, Dec. 1996.

[12] D. A. Keim, F. Mansmann, J. Schneidewind, H. Ziegler, and J. Thomas. Visual analytics: Scope and challenges. December 2008. Visual Data Mining: Theory, Techniques and Tools for Visual Analytics, Springer, Lecture Notes In Computer Science (LNCS).

[13] T. Kohonen. The Self-Organizing Map. *Proceedings of the IEEE*, 78(9):1464–1480, Sept. 1990.

[14] J. B. Kruskal and W. M. *Multidimensional Scaling.* Sage Publications, Beverly Hills, CA, 1977.

[15] A. Lex, H.-J. Schulz, M. Streit, C. Partl, and D. Schmalstieg. VisBricks: Multiform Visualization of Large, Inhomogeneous Data. *IEEE Trans. Vis. Comput. Graphics*, 17(12):2291–2300, Dec. 2011.

[16] A. Quigley and P. Eades. FADE: Graph Drawing, Clustering, and Visual Abstraction. In *Proceedings of GD'00*, pages 197–210. Springer-Verlag, 2001.

[17] V. Satuluri and S. Parthasarathy. Scalable graph clustering using stochastic flows: applications to community discovery. In *Proceedings of KDD '09*, pages 737–746, 2009.

[18] B. Shneiderman. The eyes have it: a task by data type taxonomy for information visualizations. In *Proceedings of VL '96*, pages 336–343, Sept. 1996.

[19] Symantec. Internet Security Threat Report: 2011 trends. `http://www.symantec.com/threatreport/`, May 2012.

[20] Symantec Security Response. Rustock takedown's effect on global spam volume. Available online at `http://www.symantec.com/connect/blogs/rustock-takedown-s-effect-global-spam-volume`, March 2011.

[21] Symantec.cloud. Symantec Intelligence Reports. http://www.symanteccloud.com/globalthreats.

[22] O. Thonnard. *A multi-criteria clustering approach to support attack attribution in cyberspace.* PhD thesis, École Doctorale d'Informatique, Télécommunications et Électronique de Paris, March 2010.

[23] O. Thonnard, L. Bilge, G. O'Gorman, S. Kiernan, and M. Lee. Industrial Espionage and Targeted Attacks: Understanding the Characteristics of an Escalating Threat. In *15th International Symposium on Research in Attacks, Intrusions and Defenses*, RAID'12, 2012.

[24] O. Thonnard and M. Dacier. A Strategic Analysis of Spam Botnets Operations. In *Proceedings of CEAS '11*, pages 162–171, 2011.

[25] O. Thonnard, P.-A. Vervier, and M. Dacier. Spammers Operations: A Multifaceted Strategic Analysis. *Special Issue of the Security and Communication Networks*, Spam, Phishing, and Countermeasures for Undesirable Electronic Communications, 2012. to appear.

[26] A. Ultsch. Data Mining and Knowledge Discovery with Emergent Self-Organizing Feature Maps for Multivariate Time Series. In *in Kohonen Maps*, pages 33–46. Elsevier, 1999.