CyberVis: Visualizing the Potential Impact of Cyber Attacks on the Wider Enterprise

Sadie Creese, Michael Goldsmith, Nick Moffat, Jassim Happa and Ioannis Agrafiotis
Cyber Security Group, Department of Computer Science, University of Oxford
Wolfson Building, Parks Road, Oxford OX1 3QD, UK
Email: firstname.lastname@cs.ox.ac.uk, Telephone: +44 1865 610805

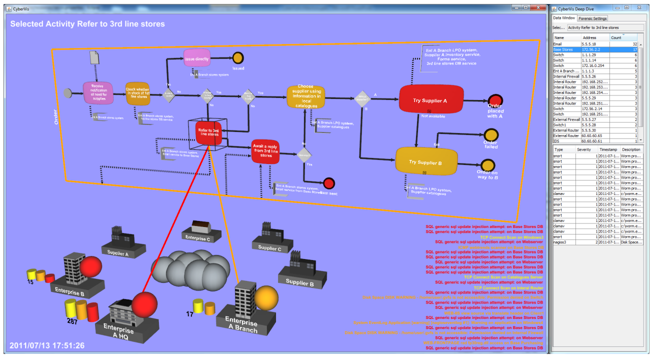
***Abstract*—A variety of data-mining tools and filtering techniques exist to detect and analyze cyber-attacks by monitoring network traffic. In recent years many of these tools use visualization designed to make traffic patterns and impact of an attack tangible to a security analyst. The visualizations attempt to facilitate understanding elements of an attack, including the location of malicious activity on a network and the consequences for the wider system. The human observer is able to detect patterns from useful visualizations, and so discover new knowledge about existing data sets. Because of human reasoning, such approaches still have an advantage over automated detection, data-mining and analysis. The core challenge still lies in using the appropriate visualization at the right time. It is this lack of situational awareness that our *CyberVis* framework is designed to address. In this paper we present a novel approach to the visualization of enterprise network attacks and their subsequent potential consequences. We achieve this by combining traditional network diagram icons with *Business Process Modeling and Notation* (BPMN), a risk-propagation logic that connects the network and business-process and task layer, and a flexible alert input schema able to support intrusion alerts from any third-party sensor. Rather than overwhelming a user with excessive amounts of information, CyberVis abstracts the visuals to show only noteworthy information about attack data and indicates potential impact both across the network and on enterprise tasks. CyberVis is designed with the *Human Visual System* (HVS) in mind, so severe attacks (or many smaller attacks that make up a large risk) appear more salient than other components in the scene. A *Deep-Dive* window allows for investigation of data, similar to a database interface. Finally, a *Forensic Mode* allows movie-style playback of past alerts under user-defined conditions for closer examination.**

I. Introduction

In recent years, a wide variety of complementary visualization techniques have emerged designed to make traffic or intrusion alert patterns and the potential impact for the wider system tangible to security analysts. However, available tools do not provide understanding of potential impact on enterprise critical tasks. It is difficult to determine the consequences of cyber attacks when investigating network- and technology-centric alert data such as that typically handled by *Intrusion Detection Systems* (IDSs), anti-malware solutions and network activity logs. This is especially the case for networks with large volumes of data. *Security Operations Centers* (SOCs) may have knowledge about an organizations network activities, but they generally do not possess enough information about how the network supports enterprise operations and, more importantly, how network alerts relate to enterprise-critical tasks.

Those charged with making operational decisions regarding attack response and recovery actions do so typically without the intuition or understanding of which enterprise tasks are currently critical, nor knowledge of any interdependence between tasks and the information infrastructure. It is this lack of situational awareness that our *CyberVis* framework is designed to address.

CyberVis builds on existing intrusion detection practices and interfaces sensors such as Snort [22], Nagios [16] and ClamAV [4]) alerts, as well as existing standards in traditional computer network-diagram icons and *Business Process Modeling and Notation* (BPMN) [2], to help form a useful and easily understood representation of the current situation and raise awareness of what is currently happening on the network. A screenshot of CyberVis can be found in Figure 1.


CyberVis
Selected Activity Refer to 3rd line stores
Order
Receive notification of need for supplies
Check whether in stock at 1st line stores
Issue directly
Issued
Ent A Branch stores system
Yes
No
Refer to 3rd line stores
Await a reply from 3rd line stores
Choose supplier using information in local catalogues
Ent A Branch LPO system, Supplier catalogues
Ent A Branch LPO system, Supplier A inventory service, Forms service, 3rd line stores DB service
A
B
Try Supplier A
Not available
Order placed with A
fail
Order failed
Try Supplier B
Order on way to B
Success?
Item sent
Enterprise C
Supplier A
Supplier C
Supplier B
Enterprise B
Enterprise A HQ
Enterprise A Branch
15
287
17
2011/07/13 17:51:26
CyberVis Deep Dive
Data Window
Forensic Settings
Selec...

```
SQL generic sql update injection attempt: on Base Stores DB
SQL generic sql update injection attempt: on Base Stores DB
TCP Connect Scan on Windows3
SQL generic sql update injection attempt: on Webserver
ICMP webtrends scanner on Base Stores DB
SQL generic sql update injection attempt: on Base Stores DB
SQL generic sql update injection attempt: on Base Stores DB
SQL generic sql update injection attempt: on Webserver
SQL generic sql update injection attempt: on Base Stores DB
SQL generic sql update injection attempt: on Base Stores DB
TCP Connect Scan on Catalogues Server
SQL generic sql update injection attempt: on Webserver
TCP Connect Scan on Interal Router
SQL generic sql update injection attempt: on Base Stores DB
Disk Space DISK WARNING - /home/user/.gvfs is not accessible: Permission denied on Switch
SQL generic sql update injection attempt: on Base Stores DB
SQL generic sql update injection attempt: on Webserver
WEB-IIS view source via translate header on Client1
SQL generic sql update injection attempt: on Base Stores DB
System EventLog Application [warning] [nagioseventlog #1]: Warning on application on Windows4
SQL generic sql update injection attempt: on Base Stores DB
Disk Space DISK WARNING - /home/user/.gvfs is not accessible: Permission denied on Internal Firewall
SQL generic sql update injection attempt: on Webserver
WEB-FRONTPAGE rad fp30reg.dll access on Base Purchasing
SQL generic sql update injection attempt: on Base Stores DB
```

Activity Refer to 3rd line stores

Name	Address	Count
Email	5.5.5.18	32
Base Stores	172.56.2.2	17
Switch	1.1.1.29	6
Switch	1.1.1.14	6
Switch	172.16.0.254	6
Ent A Branch ...	1.1.1.3	5
Internal Firewall	5.5.5.26	3
Interal Router	192.168.252....	3
Interal Router	192.168.253....	3
Interal Router	192.168.254....	3
Interal Router	5.5.5.29	3
Interal Router	192.168.251....	3
Switch	172.56.2.14	3
Switch	192.168.251....	3
External Firewall	5.5.5.27	2
Switch1	5.5.5.28	2
External Router	5.5.5.30	1
External Router	60.60.60.65	1
IDS	60.60.60.61	1

Type	Severity	Timestamp	Description
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...
clamav	1	2011-07-1...	c:\worm.e...
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...
clamav	1	2011-07-1...	c:\worm.e...
clamav	1	2011-07-1...	c:\worm.e...
clamav	1	2011-07-1...	c:\worm.e...
snort	1	2011-07-1...	Worm pro...
nagios3	2	2011-07-1...	Disk Space...

Fig. 1. A screenshot of CyberVis.

The remainder of this paper is divided into the following sections. Section II summarizes related work and the state of the art in network-security visualization. Section III provides the design overview of CyberVis, also describing its implementation. The Concept of Operations is outlined in Section IV. Finally, a discussion on usability, can be found in Section V. The paper concludes in Section VI, where future areas of research are also identified.

II. Related Work

To this day, most cyber-attack visualizations provide a segmented view of the world. Some of these favor graph-based representations, port-scanning activity characteristics, network-traffic patterns, payload characteristics or event-log forensics. Conti [7] and Marty [15] provide a detailed discussion on this topic. Some example tools include the open source Rumint [6]

978-1-4799-1535-4/13/$31.00 ©2013 IEEE

and Wireshark [5] for traffic forensics. Many tools for analyzing network usage patterns exist [1, 12–14]. Other approaches include geographical representations of malware activity [7], and city-level by Yu et al [27]. The potential wider impact of attacks on network assets was visualized using tree map-like visuals by Chu et al [3]. Intrusion-detection event correlations have also been visualized by Rasmussen et al [20]. The commercial SecureScope tool [24] addresses business impact of attacks by mapping clusters of potentially malicious network activity to business roles or organizational units (such as Human Resources) and geographical locations. Similarly, the commercial Arcsight tool [8] also provides mapping between event alerts, source IP addresses and business roles. Another application, Tenable 3D Tool [25] can visualize topology based on vulnerability scans and change the visible features of machines accordingly. The Data Exchange for Visualizing Security Events (DEViSE) [21] is an open-source architecture designed to enable data sharing between security visualization tools, acting as a middleware layer that manages these interactions so one visualization tool can transfer security-related information to another application.

The aforementioned visualizations do not attempt to model the impact of cyber attacks on business processes, except for where they connect the information infrastructure components to specific departments or business units. The coarse granularity of business roles in existing visualization applications represents a fundamental limitation; they do not enable the user to understand the specific tasks or processes within the enterprise that might be at risk (or the level of risk or potential for propagation of risk across an enterprise's activity). In most environments people tasked with monitoring systems will not possess the knowledge required to formulate such reasoning. This means that the potential ramifications of attacks on an enterprise activity will not be understood until a monitoring officer has escalated an attack alert (or set of alerts) to somebody who can form a judgment on the enterprise-level impact. This leads to needless time spent on prioritizing response options and could result in lost opportunities for formulating optimal risk-mitigation strategies and recovery actions.

III. CyberVis Overview

A. General Design Goals

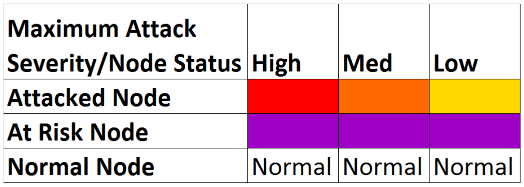
The overall aim of CyberVis is to significantly improve situational awareness and decision making capability in the face of cyber-attacks by introducing explicit consideration of business activities and priorities into the detect-and-response process. The design goals of this project is a system that:

- Produces **real-time visuals** that draws attention to areas being affected within an enterprise, both on a network level and business-task level, based on already processed network alerts.
- **Uses alert feeds** from heterogeneous third-party best-of-breed monitoring tools. Our prototype supports for Snort, Nagios and ClamAV alerts. We have also implemented our own XML schema to take in alerts, and plan to extend this in the future.
- Ensures that important elements in the scene are made more **visually salient** (objects that should be more noticeable are made very clearly noticeable) and has been optimized with the HVS in mind.
- **Supports users with both high and low technical proficiency**. A security analyst as well as high-level management must be able to gain insight into what is happening on the network and how it affects the enterprise.
- **Reflects uncertainty** in environments where threat intelligence comes from sources of varying provenance or reliability. Unknown assets on the network are represented as a question mark cube.
- Supports a **clear methodology** for initialization and an unambiguous data format to ensure the integrity of past events when revisiting previously seen attacks.

B. Visual Approach

Three factors have influenced the visual approach to CyberVis: 1) **Simplicity and user-friendliness**: making the application both easy to understand, and useful for the technical security analyst as well as the high-level business management. 2) **Scalability**: abstracting data to deliver a comprehensive overview, yet still have fast access to detailed data when necessary. This is where both the drill-down feature, and the deep-dive window become important. 3) Remaining **faithful to industry standards** to ensure a short learning curve. CyberVis provides a 3D diagrammatic representation of both the network and business processes being monitored. Maintaining both in 2D space can lead to significant amounts of visually overlapping data, which occludes elements of the scene. By mapping each 2D space to a wall in 3D allows for connectivity between layers to be mapped straightforwardly. The analyst can also change their position at any time.

The HVS is important to consider as part of the design and vision allows us to find and focus on relevant information quickly and efficiently. Rendering the scene with all available machines in the network at the same time (even in 3D space) and their respected alerts would probably overwhelm the user for larger scale networks. This is especially the case if a considerable number of devices exist within a network or subnet. To alleviate this, the network view is abstracted into several subnets. Each enterprise has its own set of subnets, and compromised machines are clustered together within those subnets (e.g. all yellow alerts are bundled to form one machine). Factors such as color, movement, size, brightness and orientation of an object are known to affect saliency [10]. We have designed CyberVis to exploit these factors, and guide users to view what is most important first.



Maximum Attack Severity/Node Status	High	Med	Low
Attacked Node	🟥	🟧	🟨
At Risk Node	🟪	🟪	🟪
Normal Node	Normal	Normal	Normal

Fig. 2. Visual Output of Risk Associated with Incoming Alerts.

Our breakdown of colors that affect the scene is as follows: Default icon color or black means that an asset or task is not affected. Purple indicates that the tool asserts an asset or task is at risk, but no actual attacks have been detected. This is the prediction functionality of CyberVis and allows us to convey uncertainty. Yellow is low severity of alert, orange is a medium severity alert, and finally, red is the highest severity of alert; see Figure 2.

The majority of the screen is occupied by only what the user currently needs to see. By rapidly obtaining an overview at an enterprise level the user gets coarse information fast, but also has the possibility to investigate information in detail. The same applies to the business process boxes hovering above the network; they display the universe of processes being monitored. This includes the current level of risk exposure as deduced from the various attack alerts received by the tool. The content of visuals are drawn in two main windows on screen: a Viewport Window to render the scene in virtual 3D environment and the Deep Dive window to view and query text data in greater detail.

C. Viewport Window

The Viewport Window renders a 3D scene populated by 3D modeled objects. Different views can be rendered in the Viewport, dependent on what the user is interested in viewing. So far a *Network View* and *Business Processes View*. Objects within each view have clickable icons, boxes and connections that display the content from the network and BPMN definitions (XML) and their relationships. Each physical piece of hardware is assigned an icon. Icons are based on existing network diagrams for straightforward communication.

*1) **Viewport Window - Rendering Alerts**:* Alerts are used to capture the user's attention when changes in system occur. Alerts are designed with five principles in mind, to show:

- When new alerts arrive (Spheres).
- How many alerts have arrived in total (Cylinders).
- Network assets and business tasks dependencies (Lines).
- Severity of alerts on network assets (Colors).
- Consequential severity of impact (Colors).

Spheres grow and shrink to highlight that new alerts have occurred within the system. Motion is used to communicate ongoing activity in an otherwise static scene. The cylinders communicate the relative volume of alerts on network nodes, and the relative severities of alerts. Straight lines between network assets and the business process lanes show the dependencies within them. Each time the status of a dependency changes, so does its color. The default color is black, which means the associated network assets are not under any immediate threat. Machines that have associated alerts have the highest severity alert color added to the surface in order to stand out more clearly to the user. Business processes and their tasks also change color when they are affected by alerts.

There are two main views the user can navigate in: the Network View (see Figure 3) and the Business Process View (see Figure 4). Users start at the enterprise/organization unit level of the Network view, and can swap to the business processes view or investigate a business process lane or the subnet of an enterprise.

*2) **Viewport Window - Network View Layout**:* Completely accurate models of network architecture are difficult to work with for many reasons including:

- They are highly complex.
- They require large scale simulations (in storage, manipulation and real-time access).
- Manual recording is error prone and almost never matches the actual system (and is made less accurate by ad-hoc changes to the topology).
- Automated network capture tools may be unable to acquire all the necessary information without becoming a security risk themselves.

CyberVis draws the network architecture as a star topology in the network view. The number of star lines/arms depends on how many enterprise units there are in the enterprise unit level. Each unit consists of at least one subnet. If only one subnet exists, the center of the star (in the subnet level) is drawn as the subnet. The user is able to zoom into an organization and view its subnet units, the user is presented with a subnet star topology, see Figure 3. Each network level is drawn on the XZ plane in a Cartesian coordinate system.

A set of business process lanes are abstracted as boxes above and behind the current network view. These lanes are drawn on the XY plane, and allow for connections between the Network view and the Business Processes to be drawn. In the center of the star topology is the Internet (if at the organization level) or a switch to enter each subnet. This simplification through abstraction approach is detailed enough to communicate the relevant information while being able to remove unnecessary complexity.

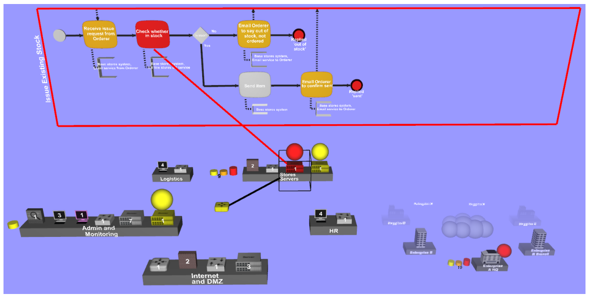

Issue Existing Stock
Receive issue request from Orderer
Check whether in stock
No
Yes
Email Orderer to say out of stock, not ordered
Replied 'out of stock'
Base stores system, Email service from Orderer
Base stores system, Email service to Orderer
Send item
Base stores system
Email Orderer to confirm sent
Replied 'sent'
Base stores system, Email service to Orderer
4
1
Logistics
9
2
1
Stores Servers
1
3
1
1
7
1
Admin and Monitoring
4
1
HR
2
1
3
Internet and DMZ
Enterprise B
10
Enterprise A HQ
Enterprise A Branch

Fig. 3. Representation of subnets in CyberVis. Subnets represented here are: DMZ, HR, Admin, Logistics and Servers and Stores.

*3) **Viewport Window - Business Process Layout**:* During program initialization, CyberVis requires an XML format for representing business process diagrams and dependencies on the information infrastructure high-level services. Since we represent high level service dependencies using standard BPMN 2.0 elements (annotations and associations), it suffices to use any XML format capable of representing BPMN 2.0 documents. We have considered two such formats: BPMN Diagram Interchange (BPMN DI); and XPDL 2.2, which is the latest version of XPDL (XML process definition language). We have chosen the second of these because it includes geometry information enabling us to support directly 3D visuals.

There are two ways to see the details of the business process lanes: either within the network view and selecting the individual process lane, or by opening up the business process view. This view displays all business process lanes at once. The output is nearly identical to that of a traditional BPMN diagram, except that the objects in the diagram are drawn on the XY plane in 3D space. This allows the user to obtain a clear overview of all business processes and how each lane in the pool relates to the others, as seen in Figure 4.

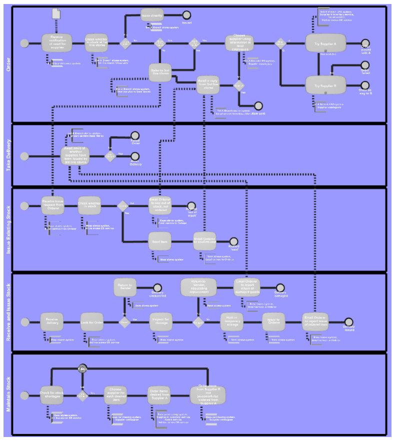


Fig. 4. Example of a set of BPMN swim lanes expanded to provide an overview of all business tasks, in the Business Process View

Business processes are configured by outlining which work activities are required to operate a business. This is turned into a flow chart that follows the BPMN standard. By capturing their own knowledge about their work activities and how that related to their technological assets, a business analyst or manager can straightforwardly provide input for CyberVis. Generic business processes can also be created allowing (for instance) finance, technology or military sectors to use these as approximations of how their enterprise operates.

D. Viewport Window - Dependency Semantics

Underpinning CyberVis is a semantic model which determines how the tool interprets network-level alerts in terms of potential impact on business processes. To achieve this we consider how to express the dependence of business processes on the underlying network, and how to determine processes and nodes on the network as potentially at-risk. This information is loaded in from a Service Support Group Configuration File, and maps business processes to their machine dependency.

We calculate the ability of the business process to terminate successfully and use the same color scheme to communicate the exposure of the various routes through the process. CyberVis depicts all nodes in a subnet with a node currently under attack as vulnerable. Such vulnerabilities are considered propagated up to the business process layer. Network and business process elements that are vulnerable in this sense are indicated using the color purple (unless colored yellow, orange or red due to some more direct risk).

E. Third-Party Implementation Technologies

CyberVis was implemented using Java [17] for cross-platform development and integration with MySQL. MySQL [18] is used to cache database content locally that is either queried from an external machine/node or imported. CyberVis is able to take in a variety of network and business process configurations and draw the scene accordingly. The layout of the scene is generated during startup from XML [26] schemas to enables straightforward definitions of the network architectures and their available business processes. Attack data can be polled from servers or fed to the system via our own attack XML schema. JOGL [11] (Java OpenGL) is also used. CyberVis also supports Stereoscopic 3D.

IV. Concept of Operations and Workflow

A. Concept of Operations

Our concept of operations is as follows: CyberVis is initialized using two configuration files: a network topology and identification of information infrastructure functions supported by it, such as SQL servers, mail servers and the like (created by someone with appropriate knowledge), a capture of the business processes and identification of the information infrastructure services required (created by someone with appropriate knowledge likely to be different to the person handling the network layer capture). Once these have been established then one or more analysts would utilize CyberVis in real-time to maintain situational awareness of the risk exposure of the systems. We expect that during periods where specific processes are crucial to the enterprise (such as merger and acquisition, floatation, military operations etc.) that an analyst might focus monitoring onto a specific subset of the processes, whilst another might maintain a whole-system view. As the system becomes exposed to attacks, the analysts is able to monitor the exposure to risk of the business processes and their respective tasks. At any point in time the analyst would utilize CyberVis to prioritize the response activities (raising of service tickets etc.) towards the technology which is crucial to keeping business processes able to function (i.e. successfully terminate).

As the situation changes, the manager of the SOC and/or the analysts would alert appropriate risk owners of escalating volumes of alert and estimated severity, and the prioritization for response and associated rationale (relating to processes and tasks) according to an agreed policy. Should the risk owners wish then they can prioritize activities for response and remedial action (as in current practice). If necessary, the risk owners can then go on to identify business workarounds where appropriate and possible. This is formulated into the policy. Relevant staff are informed and adopt the new policy. Meanwhile the security operations staff continue to identify the network assets that are supporting the high-priority activities identified, and raise appropriate service tickets/requests to initiate investigation into the compromised assets. The network assets are then either fixed, or the alerts are resolved as false positives (i.e. there is no attack).

Once assets are either cleaned or alerts identified as false, the alerts are marked as “dealt with”. Alternatively, the business management team may decide that the solution is to change a business process in response to the situation, rather than conducting technical investigations and response. Should this happen, then new policy on process is formed and communicated to appropriate staff, and normal operations are resumed. A workflow diagram is illustrated in Figure 5 (this workflow assumes an initial configuration has been set up).

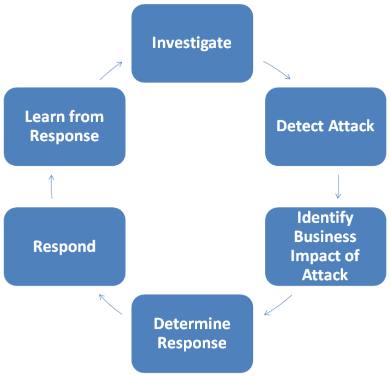


Fig. 5. Workflow of CyberVis

At any point in time an analyst may decide that they need to investigate the circumstances surrounding a particular set of events being observed, in which case they can move to a *Forensic Mode* and examine historical views in CyberVis. These historical views can be de-scoped according to a range of variables (explained in detail below) and CyberVis visuals played back similar to a "movie" at variable speeds.

B. Workflow: Navigation

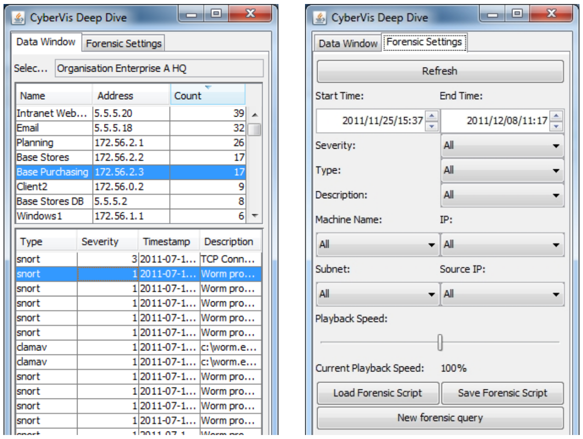
Exploratory navigation of attack-data through interaction is an important part of CyberVis. Instead of simply showing visuals, we decided to engage the user by allowing efficient navigation through the data. This serves two purposes: 1) keeping the user engaged in the system in order not to miss an attack; and 2) ensuring the user does not lose interest in what is currently happening.

The basic control scheme is the same for all views: Mouse movements change the viewing direction if the left mouse button is held down. A single left mouse click selects a scene object and displays its relevant information in the Deep Dive window. Only one icon can be selected at a time. A double-click allows for items in the enterprise level to be zoomed into (zoom selection). The *WASD* keys enable the user to "fly" in any user-given direction. *Arrow keys* allow users to strafe side-to-side (left/right) and move up or down on the vertical axis. The *B* key swaps to Business Process view, and *R* resets the view to the default position.

From a usability perspective, the zoom-selection/drill-down feature is made context-dependent. For instance, if an organization icon is double-clicked, the user zooms into the subnet level of that organization. Single-clicking makes the organization "selected" and all information about that organization is displayed in a side panel window (Deep Dive). The selected icon name as well as its alerts will also appear in the Deep Dive Window. In order to reset the camera, the Internet cloud was chosen as a neutral restarting point.

C. Workflow: Deep Dive

The Deep Dive is a separate window to the right of the screen occupying 20% of the whole screenspace. Its purpose is to allow the analyst to access detailed information on the alerts in the manner they are used to seeing. This should build confidence in novice users and allow for thorough investigation of attacks. Currently, this window consists of two tabs: *Data Window* and Forensic Mode. Figure 6 shows what parameters the users can alter in the two tabs.



Selec... Organisation Enterprise A HQ

Name	Address	Count
Intranet Web...	5.5.5.20	39
Email	5.5.5.18	32
Planning	172.56.2.1	26
Base Stores	172.56.2.2	17
Base Purchasing	172.56.2.3	17
Client2	172.56.0.2	9
Base Stores DB	5.5.5.2	8
Windows1	172.56.1.1	6

Type	Severity	Timestamp	Description
snort	3	2011-07-1...	TCP Conn...
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...
clamav	1	2011-07-1...	c:\worm.e...
clamav	1	2011-07-1...	c:\worm.e...
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...
snort	1	2011-07-1...	Worm pro...

(a) Data Window Tab

Refresh

Start Time: 2011/11/25/15:37
End Time: 2011/12/08/11:17
Severity: All
Type: All
Description: All
Machine Name: All
IP: All
Subnet: All
Source IP: All
Playback Speed:
Current Playback Speed: 100%

Load Forensic Script | Save Forensic Script | New forensic query

(b) Forensic Mode Tab

Fig. 6. Our Deep Dive Window currently consists of two tabs: Data Window and Forensic Mode

*1) **Deep Dive: Data Window**:* The default tab (Data Window) enables viewing of text information about the currently selected object in the Viewport Window. When an icon is selected, all machines that are represented by the icon will be listed in a table in the Data Window. For instance: when an enterprise, subnet or group of (infected or not-infected) machines are selected, all machines represented by the selected icon are listed, along with relevant information about them. Such information includes machine name, IP address and active alert count.

More details on the alerts (such as sensor type, severity and cause of the alert) can be investigated in this window by selecting (left-clicking) the machine name of interest. All this information can also be provided by selecting specific BPMN tasks, and shows which machines and alerts are relevant for the selected task. Right-clicking a row in the table brings up a context menu that gives the option to mark the selected machine as "cleaned" (exploits or malware removed). This command can be applied to a single device or the entire network. The menu also enables unknown machines to be renamed, so machines which are dynamically added to the network to be identified textually for ease-of-use in the Viewport Window.

*2) **Deep Dive: Forensic Mode**:* Forensic Mode allows users to do a more detailed investigation of the impact of previously seen attacks on the network. This playback mode consists of a series of filters which allow only the attack events corresponding to the filter settings to be viewed. Forensic Mode also allows users to see the impact of these filtered events in both the network view and the business process view. It does so by temporarily being in command of the Viewport

Window to render the alerts in the scene. Once Forensic Mode is switched off, CyberVis rebuilds the scene and continues in real-time mode again.

Currently, the system contains the following filters; *time period, severity, type, description, subnet, IP address, machine name* and *source IP address* (where an attack is thought to originate). The speed of playback is controlled by a time slider, which enables a user to fast-forward over uninteresting events, or slow down time to focus on time periods with heavy activity. When forensic mode is active, a red border is drawn around the screen to remind the user that they are not viewing real-time events, but are seeing events from the past. Alerts can be filtered and replayed for a user-specified timeframe. During this playback, time can be slowed down to allow the analyst to focus on attacks event-by-event, even with hundreds of attacks per second. The added benefits of Forensic Mode include: 1) **A segmented view** of the whole data set. Attacks can also be filtered by preferred parameters to investigate how the system is affected by the individual components of an attack. 2) **Playback of alerts** at user-defined specifications. The ability to replay events is a major help for new IDS analysts, but can also be used as a lessons learned tool to help analysts understand attack vectors better.

V. Usability Discussion

The aim of usability has been to make the process of capturing the necessary information about the business tasks to be monitored as usable as possible; in line with the interface design guidelines of Shneiderman [19], as well principles of on visual saliency [10]. CyberVis also reflects other usability criteria of Ibrahim et al [9] for end-user security tools in the following ways:

Visibility of the alert detector name: With our drill-down/zooming ability, users of CyberVis can quickly access details about the attack alerts being visualized including the names and additional assessment of attack type and severity.

Establish standard colors to attract user attention: We adopted the common conventional use of red, orange and yellow to reflect severity warnings contained in alerts (red being the most severe and yellow the least). However, a new color is also introduced; purple. Purple communicates risk-propagation potential (where no alerts are currently raised).

Use icons as visual indicators: The main CyberVis interface uses iconography throughout, with limited textual information (restricted to component names). The additional Deep Dive Window provides a text based interface, designed to provide detailed textual data at the point where it is needed.

CyberVis deviates from the advice of Ibrahim et al in one important way; it does not use explicit words to communicate security risk levels (severity) instead it adopts the said color scheme, indicating its priority.

VI. Future Work and Conclusion

A. Future Work

There are research challenges to be addressed. Some of these are incremental improvements to the system as a whole. CyberVis has been informally exposed to a number of different research organizations, as well as people from SOC environments. We plan to conduct a visualization usability-assessment to improve our visual and risk-propagation further using security expert (SOC analyst) participants. Furthermore, this allows us to assess the information model and add additional features required within the Deep Dive window specifically tailored for intrusion detection systems analysts.

Improved alert and business task modeling support may allow for more alert types and other business-task models to be incorporated into the system. Scalability testing will be performed to assess the ability of the model to handle a variety of scaled systems (in terms of both network size, and business process size) and high volumes of alerts in order to explore the optimal visual representation. Visualization of raw network traffic data would provide a seamless transition from business process orientated monitoring into network traffic forensics to complement our enterprise-centric approach, similar to previous work [6, 13, 23].

We plan to investigate a range of natural human interfaces to the system, the incorporation of insider-threats analytics, and the development of various learning modules, including machine-learning-based predictive capabilities and quantitative probabilistic analysis.

B. Conclusion

CyberVis demonstrates that it is possible to determine and visualize the potential impact of a cyber attack on business processes; specifically those that might be critical to an enterprise. This represents a potential step-change in situational awareness, since for the first time it should be possible for a business manager/risk owner to quickly understand the possible impact on the enterprise, and offer operational guidance as to where response and recovery efforts should be prioritized. Furthermore, CyberVis still allows the deeper text-based detail to be visible to system operators on demand.

Acknowledgment

We would like to thank the Defence Science and Technology Laboratory (DSTL) who have supported the development of CyberVis, and colleagues at Northrop Grumman Mission Systems Europe (NGMSE) and University of South Wales (USW) for continued consultation and feedback.

References

[1] D. Best, S. Bohn, D. Love, A. Wynne, and W. Pike. Real-time visualization of network behaviors for situational awareness. In *Proceedings of VIZSEC*. ACM, 2010.

[2] BPMN. Business Process Modeling and Notation. http://www.omg.org.

[3] M. Chu, K. Ingols, R. Lippmann, S. Webster, and S. Boyer. Visualizing Attack Graphs, Reachability, and Trust Relationships with Navigator. In *Proceedings of VIZSEC*. ACM, 2010.

[4] ClamAV. Clam Anti-Virus. http://www.clamav.net.

[5] G. Combs. Wireshark. http://www.wireshark.org.

[6] G. Conti. Rumint. http://www.rumint.org.

[7] G. Conti. *Security Data Visualization*. No Starch Press, San Francisco, CA, 2007.

[8] Hewlett Packard. Arcsight Enterprise Security Manager. http://www.arcsight.com.

[9] T. Ibrahim, S. M. Furnell, M. Papadaki, and N. L. Clarke. Assessing the usability of end-user security software. In *Proceedings of TrustBus*. LNCS, 2010.

[10] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(11):1254–1259, 1998.

[11] JogAmp Community. Java OpenGL (JOGL). http://jogamp.org/.

[12] H. Kim, I. Kang, and S. Bahk. Real-time visualizaton of network attacks on high-speed links. *IEEE Network*, 18 Issue 5:30–39, 2004.

[13] S. Lau. The spinning cube of potential doom. *Communications of the ACM*, 47 Issue 6:25–26, 2004.

[14] Q. Liao, A. Striegel, and N. Chawla. Visualizing graph dynamics and similarity for enterprise network security and management. In *Proceedings of VIZSEC*. ACM, 2010.

[15] R. Marty. *Applied Security Visualization*. Addison-Wesley, 2009.

[16] Nagios. Nagios network monitoring software application. http://www.nagios.org/.

[17] Oracle. Java programming language. http://www.java.com.

[18] Oracle. MySQL. http://www.mysql.com/.

[19] B. S. . C. Plaisant. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley, Boston, MA, 2004.

[20] J. Rasmussen, K. Ehrlich, S. Ross, S. Kirk, D. Gruen, and J. Patterson. Nimble cybersecurity incident management through visualization and defensible recommendations. In *Proceedings of VIZSEC*. ACM, 2010.

[21] H. Read, K. Xynos, and A. Blyth. Presenting DEViSE: Data Exchange for Visualizing Security Events. *IEEE Computer Graphics and Applications*, 29(3):6–11, 2009.

[22] M. Roesch. Snort-lightweight intrusion detection for networks. In *Proceedings of the 13th USENIX conference on System Administration*, pages 229–238, 1999.

[23] S. Sandalski. fe3d. http://projects.icapsid.net/fe3d/.

[24] SecureDecisions. Securescope. http://www.securescope.com.

[25] Tenable. Tenable 3D tool. http://www.tenable.com.

[26] World Wide Web Consortium. Extensible markup language (xml). http://www.w3.org/XML/.

[27] T. Yu, R. Lippmann, J. Riordan, and S. Boyer. Ember: A global perspective on extreme malicious behaviour. In *Proceedings of VIZSEC*. ACM, 2010.