RicherPicture: Semi-automated cyber defence using context-aware data analytics

Arnau Erola, Ioannis Agrafiotis, Jassim Happa,
Michael Goldsmith, Sadie Creese
Department of Computer Science,
University of Oxford, UK
Email: *first.last*@cs.ox.ac.uk

Philip A. Legg
Department of Computer Science and Creative Technologies,
University of the West of England, UK
Email: phil.legg@uwe.ac.uk

Abstract—**In a continually evolving cyber-threat landscape, the detection and prevention of cyber attacks has become a complex task. Technological developments have led organisations to digitise the majority of their operations. This practice, however, has its perils, since cybespace offers a new attack-surface. Institutions which are tasked to protect organisations from these threats utilise mainly network data and their incident response strategy remains oblivious to the needs of the organisation when it comes to protecting operational aspects. This paper presents a system able to combine threat intelligence data, attack-trend data and organisational data (along with other data sources available) in order to achieve automated network-defence actions. Our approach combines machine learning, visual analytics and information from business processes to guide through a decision-making process for a Security Operation Centre environment. We test our system on two synthetic scenarios and show that correlating network data with non-network data for automated network defences is possible and worth investigating further.**

I. Introduction

Over the last decade technological advances have become prevalent in our daily lives. Organisations have digitised the majority of their operations, aiming for new capabilities and efficiency. This progress, however, has its perils, since cyberspace offers a new attack-surface [13]. Detecting, deterring and predicting cyber-threats remains a challenging task, with a continually-evolving threat landscape [5]. At the same time, new variants of data sources become available every year, but our ability to correlate these in order to extract information and asses cyber threats is still in its infancy.

As our society becomes more connected and the impact of threats more detrimental, we cannot rely on network-traffic monitoring alone. We need to enhance the abilities of Security Operation Centres (SOCs) to respond to threats in a more automated fashion, while considering multiple data sources to identify an attack. Additionally, every incidence-response activity should be aimed to restore the functionality of organisations [13]. At the moment, however, SOCs lack information regarding the context within which organisations operate. Another significantly limiting factor of existing threat and attack detection tools is that they fail to capture real-world contextual information.

Little progress has been made on exploring the intrinsic relationships between cybersecurity related data and data describing business process and operational estate of an organisation, in order to make better-informed decisions when responding to cyber-attacks. This poses the research question: How can we effectively use heterogeneous operational data sources to better identify, understand, and mitigate cyber attacks?

In this work, we aim to advance the state of the art by proposing a novel way of characterising behaviours inside and outside an organisation we wish to protect. We demonstrate how our approach can be used in a real SOC in a proof-of-concept. Our tool processes multiple non-network traffic data sources for threat detection, identifies attacks and recommends or executes automatically appropriate courses of actions. We refer to the implementation as ***RicherPicture***, with the intention that this system provides a richer and more complete picture of the related organisational landscape. We validate our tool on two synthetic scenarios and show how the courses of actions on these scenarios maximise the functionality of organisations when these are under attack. Finally, we reflect on the results and propose future directions for this research.

The paper is structured as follows. Section II presents related work. Section III details our design and implementation of our approach. Section IV presents our proof-of-concept demonstration scenario and outlines our results. Section V discusses the results in-depth and outlines potential avenues for future research. Section VI concludes the paper.

II. Background

Several state-of-the-art reports have been published covering how machine learning can benefit those within the cyber-security domain [4], [10], [8]. Detection methods can be classified into groups based on statistical techniques only (e.g. looking at deviations, variance, without any prior knowledge about the system in question), knowledge-based systems (having existing knowledge about the organisation and comparing this to newly observed data) or methods that learn gradually about an organisation's network traffic behaviours. Commercial and open source Intrusion Detection Systems (IDS), Intrusion Prevention Systems (IPS) and Security Information and Event Management (SIEM) systems today suffer from an overwhelming false-positive rate and possible false negatives, and are often incapable of understanding whether those alerts fall into the context of risks to an organisation. Existing solutions:

- typically consider network events only
- often generate many false positives, leading to a “needle in a haystack” problem
- do not predict likely future scenarios
- cannot easily react to new types of attacks
- often block legitimate services causing hinderance and frustration
- may be laborious to configure and maintain

Our solution aims to tackle these limitations by proposing a system that can consider non-network data in the interest to raise situational awareness. The approach is inspired from our previous work, which addresses techniques for insider threat detection [12], [11], [1], by considering heterogeneous data sources that organisations hold and correlating them to detect possible attacks. We also investigated how to compute risk-propagation from formalised business processes and IDS events [6], and investigated methods to deliver physical situational awareness based on different data sources by computing how much trust should be assigned to known data sources [15].

III. Design and Implementation

Our tool focuses on taking account of threat intelligence [9], attack trends and likelihood of consequence for the enterprise/mission. More specifically, our model provides three core capabilities:

- **Predictive analytics** to establish the likelihood of threat manifesting based on past and currently available data.
- **Recommendation of what course of action** the organisation should take to mitigate disruption.
- **Automatic reconfiguration** of network defences.

We have designed and developed a system that is capable of gathering and analysing multiple data sources, to provide predictions on future events and to provide suitable recommendations for appropriate action. Figure 1 provides an overview of the system architecture. As a modular system design, RicherPicture is made up of five core modules:

- **Parser module.** This module parses, cleans and normalises the input data.
- **Machine learning module.** This module extracts insights from the data and raise alerts and predictions from machine learning algorithms.
- **Decision system module.** This module acts on alerts and predictions by giving automated actions or recommendations to the analysts.
- **Visual analytics module.** This module presents results to analysts and options for undoing automated actions.
- **Network defence module.** This module acts upon decisions made by either the decision system or human analyst.

A. Parser module

The parser module receives different CSV files and converts them to our own JSON schema. Our decision to assume a generic CSV format was informed by the fact that most technical analysts should be able to write a parser for CSV straightforwardly – irrespective of their existing cyber and non-cyber sensor formats. In addition, some sensors have built in CSV output support, including a number of SIEM tools and network monitoring tools [14], [3], [7], [2]. Nevertheless, the parser module could be easily extended to support XML exports used by some threat intelligence tools.

Once CSV files are parsed, we populate these using our JSON schema. An advantage in converting CSV to our own intermediate format is that it allows us to swap out raw input formats later, without affecting the whole RicherPicture framework; thus providing flexibility in parsing different data sources and linking information from these to critical parts of the Machine Learning Module. All we would need to change is the parser module. We deem the intermediate format necessary to ensure the system is able to work with standards from different real-world domains (where repeating names are likely to happen).

Both the CSV format and JSON schema are written in an easy to understand manner should an analyst wish to use a different visualization or database tool to investigate the data further – allowing for our tool to be adopted more straightforwardly in most defence environments.

B. Machine learning module

Machine Learning is becoming increasingly common in many different applications, such as anomaly detection. The RicherPicture framework incorporates a predictive model for timeseries forecasting based on previously observed activities that are logged in the system. Given a set of observations $x_1 \dots x_n$ at time t, the objective is to be able to predict what the subsequent observation at time $t+1$ will be, for any x. It may then be desirable for analysts to use such predictions to further investigate future timesteps (e.g., $t+10$, or $t+50$).

For the purposes of our prototype, we develop a regression-based predictive model using the popular Python library *scikit-learn*. Since our data is a timeseries, we consider the first m data entries as the training data. For each instance of our training, we use the observations x_i at time i, and the outputs are the observations x_{i+1} at time $i+1$. After that, each instance is used as observation input, and the output results are logged. At the next time instance, a prediction error can then be calculated based on the difference between the predicted output at time $t+1$ and the newly-observed input at time $t+1$. When the deviation of the prediction error at time $i+1$ exceeds a certain threshold, it is indicating an abnormal behaviour. The threshold is calculated based on the standard deviation of the cost function (mean squared error) for the previous predictions. To reduce error, the system can also be configured to only alert after a given number (e.g., three) of consecutive abnormal observations.

We should note that one of the advantages of our modular design is that we can replace the machine learning algorithms used in this module with those which are deemed more effective for the datasets analysts are interested in investigating. As long as anomalous behaviour is flagged up, the system should be able to parse these alerts and act upon these.

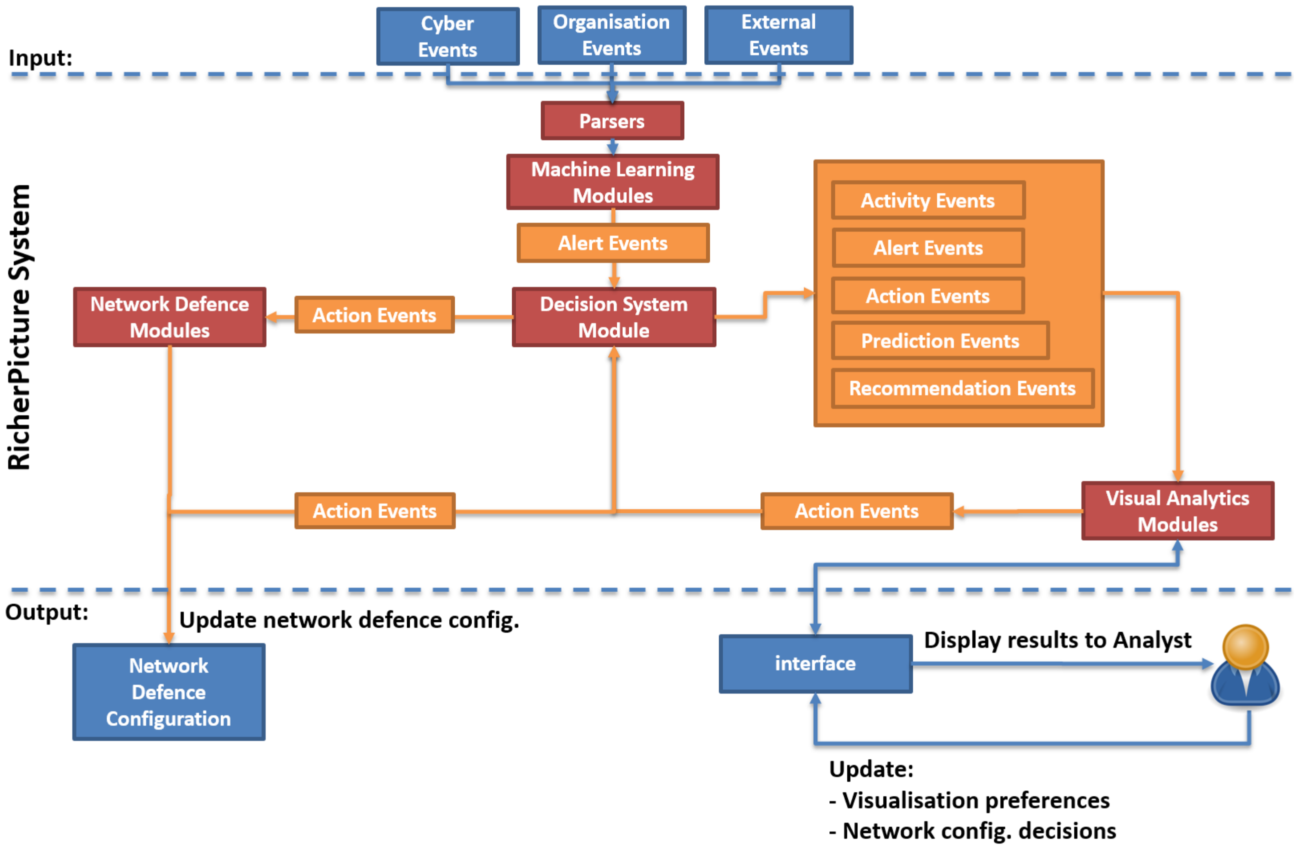


Fig. 1. Architecture diagram of the RicherPicture System. The modules of the system are drawn in red, data I/O in blue and orange boxes represent events in the system. The Analyst can provide feedback to the main Detection System Module to improve the detection accuracy.

C. Decision system module

The decision system acts on the alerts produced by the machine learning module. These alerts are handled either by providing a variety of recommendations for the analyst to consider or by prompting the network defence module to act in an automatic fashion. The decision system is always driven by context to determine whether an automatic action or a recommendation will be provided and which options should be available to the analyst. In order to capture context, we take advantage of the information provided by various business processes and external events, such as Common Vulnerability Enumeration (CVE) reports.

The most prevalent way to describe business processes is the Business Process Modelling Notation (BPMN) [17]. BPMN is a rich, descriptive notation, which allows organisations to capture fully events and relations which form a process. Types of information which can be found in such diagrams are:

- Remaining time of a task
- Person responsible for a task
- Data processed or generated by a task
- Cost of a task
- Type of a task
- Type of an outcome
- Mitigation processes

Although BPMN visually conveys information very effectively, the machine-readable xpdl format would be inefficient to use in RicherPicture. We converted BPMN diagrams into directed acyclic graphs and extended the information available from the BPMN to capture context. The advantage of graphs is that it enables exploration of paths and more information may be inferred. Figure 2 provides an example of a BPMN diagram converted to a graph. The extended information the graphs provide in our system is:

- Remaining time to complete the BP
- Set of machines supporting the task
- Number of redundant machines for each task
- Value of the best possible outcome for the BP
- Value of the worst outcome for the BP
- Number of possible outcomes
- Set of people responsible for all the tasks of the BP
- Overall cost of a BP for the best outcome
- Overall cost of a BP for the worst outcome
- Financial loss in case the BP is cancelled
- Value of a Business Process
- Value of the specific instantiation of a Business Process

The aforementioned information is available at each node of the graph and is summarised in a BP log. Every time a task is completed it triggers a different BP event, denoting the

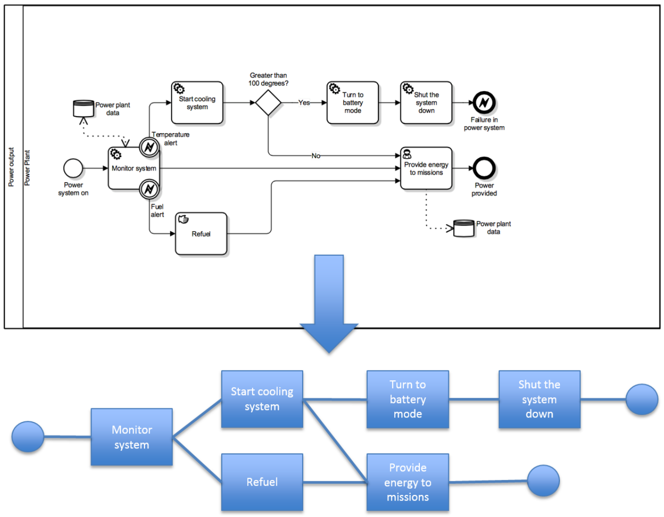


Fig. 2. BPMN graph conversion to expand support for additional data sources.

progress degree of the BP (which is the current state of the graph). Once an alert is generated, the decision system will check the type of the alert, the machines which the alert concerns and will identify which BPs should be considered. It will then recall the relevant BP data and will make a decision based on the information provided. Figure 3 illustrates the decision tree used in the two Scenarios presented in Section IV. Once the affected machines and the type of alerts are established, the decision system will request information about the importance of the BP running on the affected machines and the time required for it to run to completion. The importance of the BP is described by a numerical value representing the criticality of the mission. Thresholds to determine when an action can be applied during a mission were manually chosen. The decision system will then provide a set of recommendations or an automatic action to reconfigure the network.

In predetermined time intervals (often but without overloading the system), the module stores the current state and the last checked events. Thus, the tool will be able to restart the execution from a past recent point.

D. Visual analytics module

The visual analytics module presents a graphical user-interface in which raw data, outputs from the machine learning module and recommendations are shown (see Figure 6). It was implemented in Python *TkInter*. The module receives the events generated by the machine learning module and the output of the decision system and displays this information. Specifically it shows a historical list of alerts and how they relate to data sources, and different timeseries plots corresponding to the machine learning module. It is designed in such a way that it should be straightforward to add or remove other data sources. We should note that the visualisations presented in Figure 6 are designed to convey information for the two synthetic scenarios presented in the Section IV.

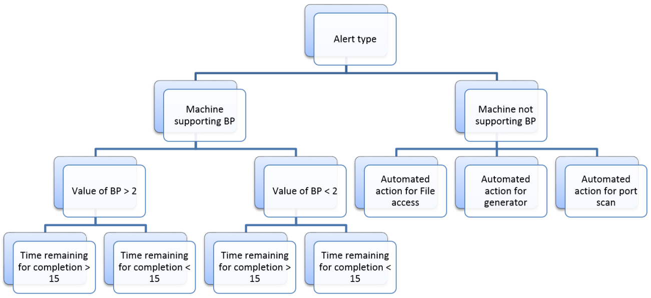

Alert type
Machine supporting BP
Machine not supporting BP
Value of BP > 2
Value of BP < 2
Automated action for File access
Automated action for generator
Automated action for port scan
Time remaining for completion > 15
Time remaining for completion < 15
Time remaining for completion > 15
Time remaining for completion < 15

Fig. 3. Decision System Tree

E. Network defence module

The purpose of network defence module is to act upon events from the decision system module. The actions proposed focus on reconfiguring the network defence parameters. The network defence module has access to a library of scripts listing the commands determined by the decision system. The actions depend on the controls placed on the network. Full control over machines may provide more options in the future. Similar to the two other modules in our RicherPicture framework, the network defence module is a middleware module between our system and the underlying defence technologies, enabling our tool to be easily integrated in new environments or swapped should we wish to interface with other network defence systems. Automated actions capability demonstrated in our concept demonstrator includes:

1) Block/Enable IP or port on a machine to isolate it
2) Block/Enable physical access to a CD/DVD drive on a machine
3) Force shutdown of a specific machine

All scripts in our library currently assume a Linux environment. In the future, we envisage a *defence script manager* that could be responsible for running appropriate scripts for the specific OS of the machines. Naturally, additional network reconfigurations can be added to the vocabulary of the network defence module. For the purpose of demonstration, we implemented a system focusing on developing enhanced decision-making capability (rather than adding overly-complex actions).

IV. Simulations and Results

To demonstrate the capabilities of our proposed situation-awareness tool, we created a fictional military scenario with our project funder for experimentation purposes. This scenario involves a large fictional island that comprises several military bases scattered across it. The island has four nations, two of which are not on friendly terms.

First, we determined the types of data sources we considered in our fictional world, including network related and non-network related data sources. Our aim was to demonstrate how non-conventional data sources may be used in SOC environments. Therefore we created two different cases: in the first case, there is a data exfiltration incident, whereas in the second case there is a cyber-physical system behaviour manipulation.

We synthesised data for both scenarios that consists of file activity, port scan activity, data about electrical generators, and business processes. More specifically, organisation events comprised of BPMN-events (tokens that describe how far along we are in a business process), the definition of the business processes themselves and the gateways as part of the BPs. Figure 4 shows an example of a BPMN used in our demonstrator, describing the communication processes. We had a list of people affiliated with the nation, these people went about their daily tasks as part of their jobs. A scheduler had them visit their jobs and do their daily tasks with a pseudo-random offset to introduce noise in their behaviours.

In our scenario we assumed that a power generator would support the military campuses and a set of basic rules would apply in our simulation to acquire data for the generator. For instance, fuel levels of a power generator would depend on how much power is required to keep the Forward Operating Base (FOB) running. The power is determined by workload on computers and personnel. The power generator reports it needs refuelling to engineers when it is low. The engineer refuels and performs maintenance at regular and non-regular intervals. The BPs were that of the power plant, a patrol, communication processes and processes related to UAVs.

At pre-determined intervals a website named *CVE-watch* (a fictional equivalent to the US National Vulnerability Database (NVD) [16]) would publish new vulnerabilities. We would link new CVEs and CVSS scores with alerts raised by the machine learning module, if the machine for which an alert is generated is mentioned in the CVE data. One of the simulation vulnerabilities was related to affecting a cyber-physical system that our fictional military organisation had access to. The fictional nation has a world temperature that several entities would take into account (another external measurement relevant for the cyber-physical system).

Finally, we also included cybersecurity related data, including: login logs, network logs, IDS logs, file access logs, and power generator logs.

Two attack scenarios were created in which we demonstrated:

1) **Detection of exfiltration of sensitive data by an insider threat** who scanned the local network and attempted to copy files across the network. Our network defence isolated the machine after having determined that the user was likely an insider threat conducting an attack.
2) **Detection of an attack adulterated firmware** (and thus not picked up by AV software) affecting a power generator. The attacker changed the temperature threshold for the cooling system. In this case, the exploit used by the attacker relates to a published vulnerability on CVE-Watch.

In these scenarios, noise data (data about what other entities are doing) was also generated to make the attack less obvious for our detection system. We should note that the emphasis of our project was not to extend the knowledge in the field of machine learning, we were rather interested in incorporating information from business processes to decision-making processes in SOC environments. Therefore, in the cases presented below, we demonstrate how RicherPicture considers information from business processes and acts based on maximising the functionality and resilience of the organisation under attack.

In order to demonstrate this, we ran both scenarios twice with the exact same data. The only condition changing in the system is the operational aspect for which the machines under attack are responsible for (i.e. change in context). Specifically, we focus on the issue of *availability* of systems, and how predictions and alerts can help identify whether a system is or will be compromised and should be taken offline. In both runs of the system, the machine learning module raises the same alert, and the decision module links this alert to the information derived from business processes and either recommend a course of action to the analyst or execute an action automatically. Our results demonstrate that it is possible to prioritise actions based on the mission of interest and be more specific about which recommendations should be presented to analysts.

A. Scenario 1 – Data exfiltration

In the first scenario a registry officer is using a writable CD to take critical data about a forthcoming operation. In addition, the CD includes malware which generates a series of internal network port scans and attempts to write this information in the CD as well. The registry officer then takes the CD with the valuable information, resulting in a major delay in the military operation.

Our efforts in data generation for this vignette focused on generating file-access logs and internal network scan logs (reconnaissance work). We combine this information with the sensitivity of the files stored on the mission network.

1) BPs present for the machine we generate an alert for: A port scan and abnormal file-access were detected and alerts raised. These are two separate events, each of them driving to different actions or recommendations. In this case, the port scan may derive to network isolation and the file-access to shut down the machine. However, the machine which the insider used was responsible for sending commands to a UAV that was currently on a mission, using the internal-systems network communications. Therefore it is to the interest of the military to not isolate the machine until the UAV has successfully completed its mission. Recommendations are suggested to the analyst along with information related to the BP. The analyst may contact the person in charge of the UAV mission or decide to shut down the machine after 30 minutes, which is the time left to complete the mission. The DVD drive is blocked however, as this action does not affect the mission. The recommendations, as well as the visualisations of the network data, are presented in Figure 5.

2) No BPs present for the machine we generate an alert for: Using the same dataset, a port scan and abnormal file-access were detected and alerts raised. However, since the machine which the insider used to access the files is not participating in any of the BPs, the decision system is able

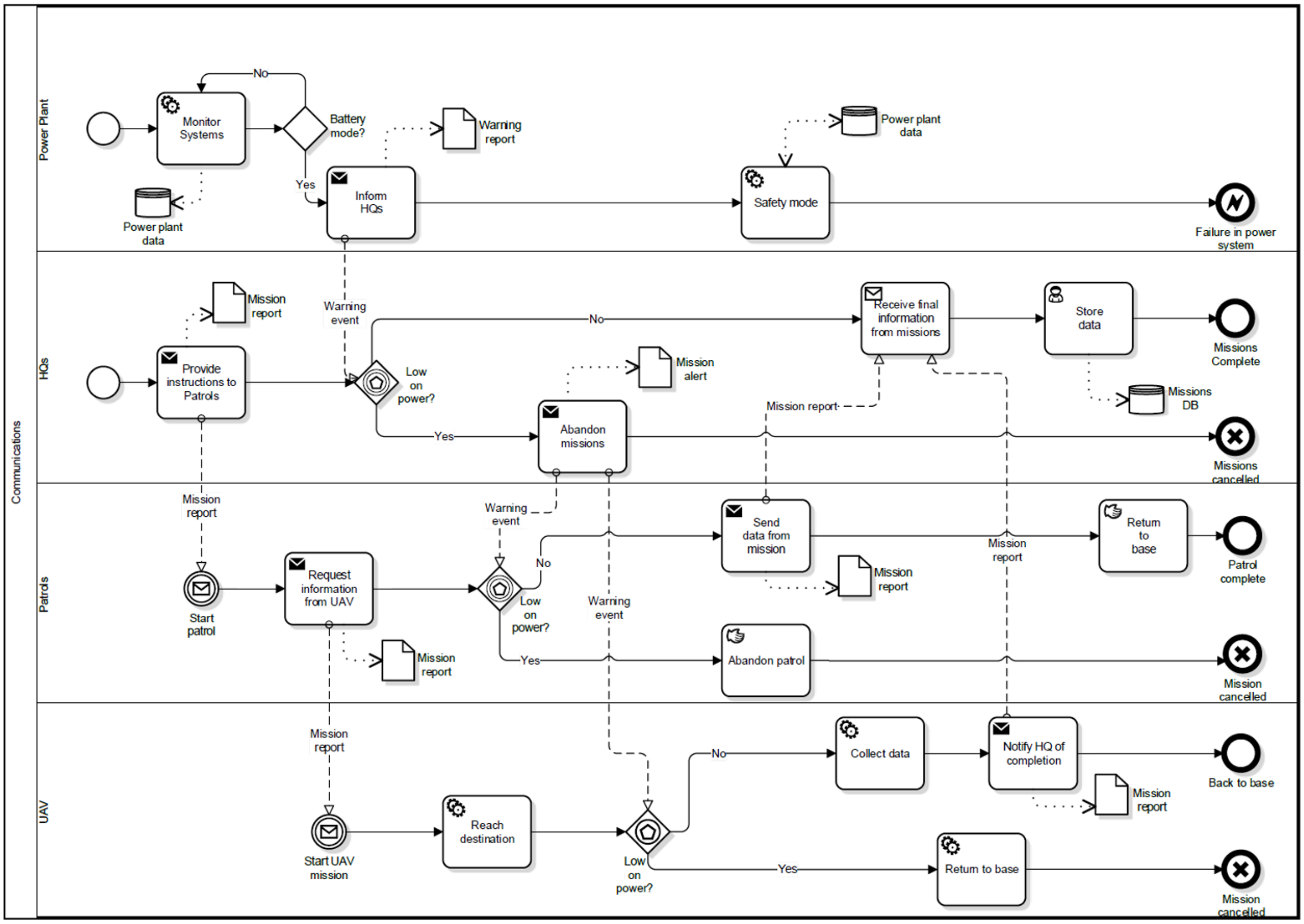


Fig. 4. Communications processes.

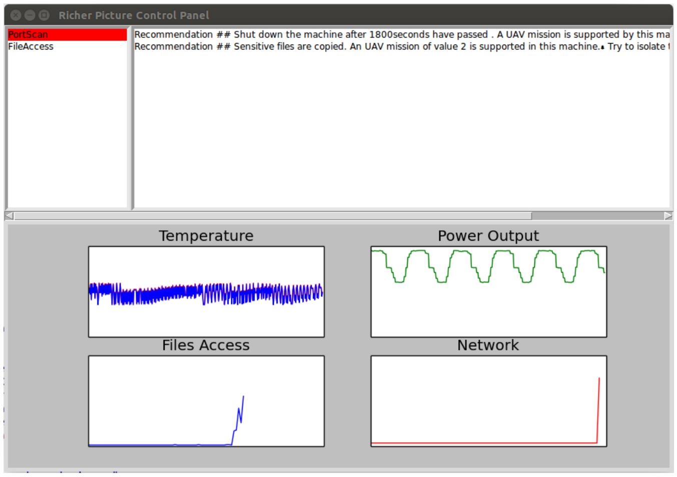
to suggest an action to defend against data exfiltration in an automatic fashion. Therefore, the system isolates the machine from the network and blocks the removable media access (in our case, it locks the DVD drive). In this instance, the tool did not know about any dependency to other organisation assets, and therefore it isolated the machine. Conceivably, this sort of approach could be expanded to erasing data in the interest of mission protection – however, we believe further thinking would be necessary in this regard to do so responsibly. We should note that all the automated actions were implemented and shown in a demo to the sponsor.

B. Scenario 2 – Cyber-physical system behaviour manipulation

The second case study focuses on an insider attack of a cyber-physical system, a power generator which was compromised through a monthly firmware update. There is a cooling system to control the increase in the temperature of the generator, which starts if a certain threshold in the temperature is exceeded. There is also a safety mechanism in the generator, shutting it down once the temperature exceeds another threshold. Normally, the threshold for the cooling system must be lower than the threshold for the shutdown mechanism. The firmware vulnerability, that is exploited during an update of the system, takes advantage of the fact that there is no control in place to check these two thresholds. The attacker, sets a threshold for the shutdown of the system in such a degree that the cooling system never has the chance to start. For this scenario, there is also a vulnerability report in the fictional CVE website.

After the vulnerability has been exploited, the cooling system of the power generator does not get turned on when it is supposed to. This effect is not noticed until a few days after the installation as the ambient temperature increases in the country. RicherPicture should be able to detect potential causes earlier as it is able to correlate the data to CVE publication, the cooling system (not turning on), the performance of the power generator and the world temperature.

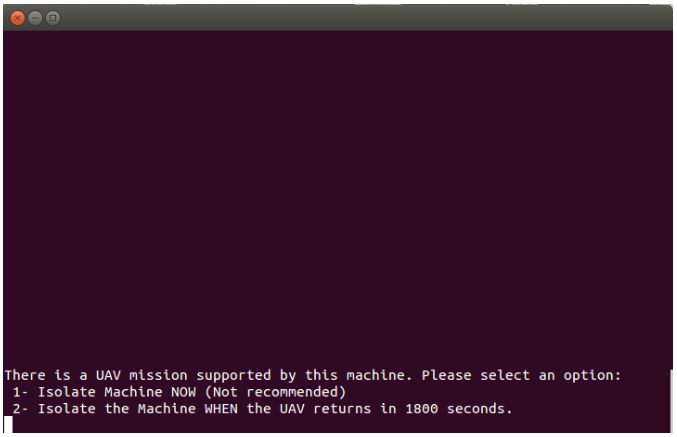
1) BPs present for the machine we generate an alert for: Figure 6 shows how RicherPicture utilises linear regression analysis, using temperature as the dependent variable, and cooling system, exterior temperature, fuel consumption and power output as the independent variables of the regression. The system detects deviations of real values from predicted ones and raises a recommendation to the analyst, as seen in Figure 7. Notice that in the case where the ambient



```
PortScan
FileAccess
```

```
Recommendation ## Shut down the machine after 1800seconds have passed . A UAV mission is supported by this ma
Recommendation ## Sensitive files are copied. An UAV mission of value 2 is supported in this machine.• Try to isolate t
```

Fig. 5. The moment the system has identified the attack in *Scenario 1*.

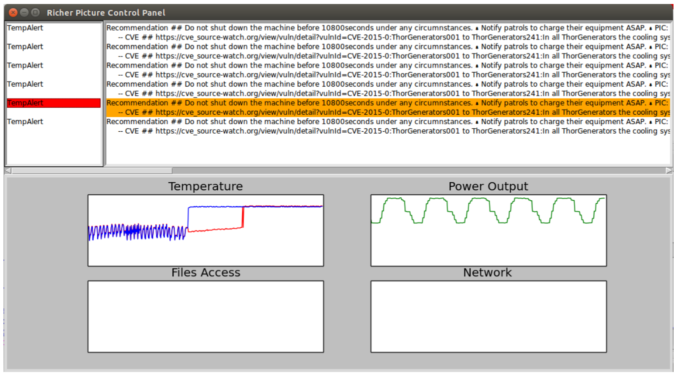


```
There is a UAV mission supported by this machine. Please select an option:
 1- Isolate Machine NOW (Not recommended)
 2- Isolate the Machine WHEN the UAV returns in 1800 seconds.
```

Fig. 7. Console recommendation prompt, showing how a UAV mission is supported by a particular machine. Analysts can decide the appropriate action.

temperature is so high that may cause the generator to shut down legitimately, our system will not trigger an alert.

As shown in Figure 6, since the decision system knows that the machine is responsible for UAV missions (and how much time is left), this information affects the recommendation to the user. The analyst is also presented with the published CVE information, since the alert is generated for the power generator.



```
TempAlert
TempAlert
TempAlert
TempAlert
TempAlert
TempAlert
```

```
Recommendation ## Do not shut down the machine before 10800seconds under any circumnstances. • Notify patrols to charge their equipment ASAP. • PIC:
    -- CVE ## https://cve_source-watch.org/view/vuln/detail?vulnId=CVE-2015-0:ThorGenerators001 to ThorGenerators241:In all ThorGenerators the cooling sys
Recommendation ## Do not shut down the machine before 10800seconds under any circumnstances. • Notify patrols to charge their equipment ASAP. • PIC:
    -- CVE ## https://cve_source-watch.org/view/vuln/detail?vulnId=CVE-2015-0:ThorGenerators001 to ThorGenerators241:In all ThorGenerators the cooling sys
Recommendation ## Do not shut down the machine before 10800seconds under any circumnstances. • Notify patrols to charge their equipment ASAP. • PIC:
    -- CVE ## https://cve_source-watch.org/view/vuln/detail?vulnId=CVE-2015-0:ThorGenerators001 to ThorGenerators241:In all ThorGenerators the cooling sys
Recommendation ## Do not shut down the machine before 10800seconds under any circumnstances. • Notify patrols to charge their equipment ASAP. • PIC:
    -- CVE ## https://cve_source-watch.org/view/vuln/detail?vulnId=CVE-2015-0:ThorGenerators001 to ThorGenerators241:In all ThorGenerators the cooling sys
Recommendation ## Do not shut down the machine before 10800seconds under any circumnstances. • Notify patrols to charge their equipment ASAP. • PIC:
    -- CVE ## https://cve_source-watch.org/view/vuln/detail?vulnId=CVE-2015-0:ThorGenerators001 to ThorGenerators241:In all ThorGenerators the cooling sys
Recommendation ## Do not shut down the machine before 10800seconds under any circumnstances. • Notify patrols to charge their equipment ASAP. • PIC:
    -- CVE ## https://cve_source-watch.org/view/vuln/detail?vulnId=CVE-2015-0:ThorGenerators001 to ThorGenerators241:In all ThorGenerators the cooling sys
```

Fig. 6. The moment the system has identified the attack. In the temperature graph, the red line is the predicted value and blue is the real value.

2) No BPs present for the machine we generate an alert for: In the case where the machine is not supporting a BP, the system generates the same alert. However, the recommendation of the decision system is different. It prompts the network defence module to shut down the machines powered by this generator, in the interest to prevent damage to the generator from overheating. There is still a link to the appropriate CVE file for the analysts to investigate further.

V. Discussion

Our proof-of-concept demonstrates significant potential in the use of monitoring and analysis of large-scale sensors within organisations to detect and combat security threats. As the volume and types of data continue to increase we need more sophisticated computational tools that can analyse and reason about such data, whilst also allowing the analysts to do so more intuitively. This role of machine learning combined with contextual information and visual analytics to support the analyst in their overall decision-making seems a sensible and viable approach for future security tools to be heading towards.

Converting business processes to graphs provides a great opportunity to convey useful information regarding the operational aspects of an organisation. Data regarding the cost of a task, the time remaining for this task to be completed, the people and machines involved in this tasks provide invaluable insight for decision-making in incident response. Graphs, also enable the system to calculate all possible paths to an end-point (effectively traversing through nodes), therefore possible outcomes can be taken into account (i.e. what is the maximum time required for a process to be completed, how many alternatives exist to accomplish the mission). All this information can be utilised by the decision module, as showed in our synthetic scenarios to guide the decision-making process.

Our results suggest that the decision system is easily reconfigurable and can provide the necessary information or authorise automated actions. However, automated actions should be treated carefully, since they may be irreversible. In our scenarios, we had to identify what is the important information from the business processes and what actions the system should prioritise. This task is not trivial for a real organisation, however, incident response strategies that focus on operational aspects should be used to guide this process.

In many cases, organisations may not want to reveal sensitive information about their business processes to the SOCs. To address this issue, the decision system should prohibit the disclosure of such information and convey the operational aspects in a more abstract way. For example, there is no need for the analyst to know that a UAV mission is controlled by a specific machine. Instead, there could be a message denoting the importance of the mission only. We should note that for both synthetic cases we provided actions and recommendations for two different types of users: those whose credentials allowed access to sensitive information and those who did not have access to this information.

Another issue that must be resolved is that of the actions that are available to the decision system. For every case study we identified possible actions that can be implemented with scripts to achieve automation. Further research should focus on examining what actions analysts in the SOC environments have available and how straightforward is to apply these automatically for network defence.

Focusing on the machine learning component, in our synthetic scenarios we did not have multiple alerts being triggered at the same time. This is of course an artefact of synthetic data. In real environments multiple alerts may be triggered that may correspond to the same machine being affected. Therefore, prioritisation of response actions and recommendations should consider such scenarios. Similarly, when false positive events are identified by an analyst, the system must provide a feedback mechanism to the machine learning component, which should learn from past experiences and adapt system's responses accordingly.

Finally, we tried to incorporate threat-intelligence information in our recommendations. The way we achieved this was by linking the hardware or software inventory of the organisation to the one present in CVE files. Once alerts were generated for a specific machine, the corresponding CVE files were presented to the analyst. It is not a trivial task, however, to link all the available CVE files to all the hardware and software aspects of an organisation. Further research should focus on how best to take into account such threat-intelligence information.

We believe that with the two vignettes we demonstrated how contextual information can be utilised to support decision-making in SOC environments. One key aspect that was important to the conceptualisation of RicherPicture was the modular design. The advantages of this design are evident in the scenarios, since different actions are implemented for each case. There are further benefits, however, since the library of automated actions is extendible, more detailed business processes can be added and any machine learning algorithm can be applied to derive the alerts without influencing other parts of the system.

VI. Conclusion

We have presented a novel approach to large-scale heterogeneous data analytics: RicherPicture. The objective is to combine multiple sources of data to obtain a richer picture of the current situation. This is considered to be the combination of alerts from anomaly detection systems with information about the operational aspects of an organisation. We used a regression machine learning approach to analyse different data sources in order to indicate suspicious behaviour. We then made informed recommendations for the analysts to avoid undesirable outcomes. Our proof-of-concept shows promise, but further work is necessary to fully validate our method in live cybersecurity environments.

Although in its early stages, we believe our tool is a potential game-changer in semi-automated network defences in that we have been able to demonstrate how both machine learning and visual analytics can be used to investigate how cyber relates to the real world. Our proof of concept has been shown to work well for laboratory conditions, and in the future, we intend to test the scalability of the system (e.g. volume of data), implement other machine learning and visual analytics techniques, integrate support for other tools, enhance the usability of the tool, introduce historic learning capability, use real attack data, add backtracking capability and investigate computational trust in automated defences.

Acknowledgement

The RicherPicture project was sponsored by the UK Defence Science and Technology Laboratory (DSTL).

References

[1] I. Agrafiotis, A. Erola, J. Happa, M. Goldsmith, and S. Creese, "Validating an insider threat detection system: A real scenario perspective," in *Security and Privacy Workshops (SPW), 2016 IEEE*. IEEE, 2016, pp. 286–295.

[2] E. ArcSight, "Arcsight esm white paper," 2010.

[3] J. Beale, R. Deraison, H. Meer, R. Temmingh, and C. V. D. Walt, *Nessus network auditing*. Syngress Publishing, 2004.

[4] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM computing surveys (CSUR)*, vol. 41, no. 3, p. 15, 2009.

[5] M. J. Covington and R. Carskadden, "Threat implications of the internet of things," in *Cyber Conflict (CyCon), 2013 5th International Conference on*. IEEE, 2013, pp. 1–12.

[6] S. Creese, M. Goldsmith, N. Moffat, J. Happa, and I. Agrafiotis, "Cybervis: visualizing the potential impact of cyber attacks on the wider enterprise," in *Technologies for Homeland Security (HST), 2013 IEEE International Conference on*. IEEE, 2013, pp. 73–79.

[7] J. Frahim, O. Santos, and A. Ossipov, *Cisco ASA: all-in-one firewall, IPS, and VPN adaptive security appliance*. Pearson Education, 2014.

[8] P. Garcia-Teodoro, J. Diaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," *computers & security*, vol. 28, no. 1, pp. 18–28, 2009.

[9] M. Glassman and M. J. Kang, "Intelligence in the internet age: The emergence and evolution of open source intelligence (osint)," *Computers in Human Behavior*, vol. 28, no. 2, pp. 673–682, 2012.

[10] A. Lazarevic, L. Ertoz, V. Kumar, A. Ozgur, and J. Srivastava, "A comparative study of anomaly detection schemes in network intrusion detection," in *Proceedings of the 2003 SIAM International Conference on Data Mining*. SIAM, 2003, pp. 25–36.

[11] P. A. Legg, O. Buckley, M. Goldsmith, and S. Creese, "Automated insider threat detection system using user and role-based profile assessment," *IEEE Systems Journal*, 2015.

[12] P. A. Legg, N. Moffat, J. R. Nurse, J. Happa, I. Agrafiotis, M. Goldsmith, and S. Creese, "Towards a conceptual model and reasoning structure for insider threat detection." *JoWUA*, vol. 4, no. 4, pp. 20–37, 2013.

[13] I. Linkov, D. A. Eisenberg, K. Plourde, T. P. Seager, J. Allen, and A. Kott, "Resilience metrics for cyber systems," *Environment Systems and Decisions*, vol. 33, no. 4, pp. 471–476, 2013.

[14] MITRE, "Common vulnerabilities and exposures," https://cve.mitre.org, 2005.

[15] J. R. Nurse, I. Agrafiotis, M. Goldsmith, S. Creese, and K. Lamberts, "Two sides of the coin: measuring and communicating the trustworthiness of online information," *Journal of Trust Management*, vol. 1, no. 1, p. 5, 2014.

[16] N. I. of Standards and Technology, "National vulnerability database," https://nvd.nist.gov/, 2011.

[17] S. A. White, "Introduction to bpmn," *IBM Cooperation*, vol. 2, no. 0, p. 0, 2004.