Automated Insider Threat Detection System using User and Role-based Profile Assessment

Philip A. Legg, Oliver Buckley, Michael Goldsmith and Sadie Creese

***Abstract*—Organisations are experiencing an ever-growing concern of how to identify and defend against insider threats. Those who have authorised access to sensitive organisational data are placed in a position of power that could well be abused and could cause significant damage to an organisation. This could range from financial theft and intellectual property theft, through to the destruction of property and business reputation. Traditional intrusion detection systems are not designed, nor are capable, of identifying those who act maliciously within an organisation. In this paper, we describe an automated system that is capable of detecting insider threats within an organisation. We define a tree-structure profiling approach that incorporates the details of activities conducted by each user and each job role, and then use this to obtain a consistent representation of features that provide a rich description of the user's behaviour. Deviation can be assessed based on the amount of variance that each user exhibits across multiple attributes, compared against their peers. We have performed experimentation using 10 synthetic data-driven scenarios and found that the system can identify anomalous behaviour that may be indicative of a potential threat. We also show how our detection system can be combined with visual analytics tools to support further investigation by an analyst.**

***Keywords*—Insider threat, anomaly detection, cyber security.**

I. Introduction

The insider-threat problem is one that is constantly growing in magnitude, resulting in significant damage to organisations and businesses alike. Those who operate within an organisation are often trusted with highly confidential information such as intellectual property, financial records and customer accounts, in order to perform their job. If an individual should choose to abuse this trust and act maliciously towards the organisation, then their position within the organisation, their knowledge of the organisational systems, and their ability to access such materials, means that they can pose a serious threat to the operation of the business. The range of possible activities could be anything from taking money from a cash register, to exfiltrating intellectual property from the organisation to sell on to rivals which could effectively destroy the successful operation of the organisation. Capelli *et al.* from the Carnegie Mellon University Computer Emergency Response Team (CMU-CERT) group identify three main groups of insider threat: IT sabotage, theft of IP, and data fraud [1]. There are also a growing number of cases that media attention has highlighted in recent years that reveal both businesses and governments alike have suffered similar experiences, whereby top secret information has been exfiltrated and passed on to oppositions. The threat posed by the insider is very real, and requires serious attention from both employees and organisations alike.

Over the years, technological advancements have meant that the way organisations conduct business is constantly evolving. It is now common practice for employees to have access to large repositories of organisation documents stored electronically on distributed file servers. Many organisations provide their employees with company laptops for working whilst on the move, and use e-mail to organise and schedule appointments. Services such as video conferencing are frequently used for hosting meetings across the globe, and employees are constantly connected to the Internet where they can obtain information on practically anything that they require for conducting their workload. Given the electronic nature of organisational records, these technological advancements could potentially make it easier for insiders to attack. From the organisational view, one advantage to this is the capability of capturing activity logs that may provide insight into the actions of employees. However, actually analysing such activity logs would be infeasible for any analyst due to the sheer volume of activity being conducted by employees every day. What is required is a capability to analyse individual users who conduct business on organisational systems, to assess when users are behaving normally, and when users are posing a threat.

In this work, we present a systematic approach for insider threat detection and analysis based on the concept of anomaly detection. Given a large collection of activity log data, the system constructs tree-structured profiles that describe individual user activity and combined role activity. Using these profiles, comparisons can be clearly made to assess how the current daily observations vary from previously-observed activities. In this fashion, we construct a feature set representation that describes the observations made for each day, and the variations that are exhibited between the current day and the previously-observed days. This large feature set is reduced into multiple anomaly assessment scores using Principal Component Analysis (PCA) [2] decompositions on subsets of features, to identify the degree of deviation for each grouping. The anomaly assessment scores can be used either with classification schemes to produce a list of suspicious users, or can be visualized using parallel co-ordinates plots to provide a more in-depth view. To test the performance of the approach, a red team developed 10 simulated insider

P. A. Legg was with the Cyber Security Centre, University of Oxford, Oxford OX1 3QD, U.K. He is now with the Department of Computer Science, University of the West of England, Bristol BS16 1QY, U.K. (e-mail: phil.legg@uwe.ac.uk).

O. Buckley was with the Cyber Security Centre, University of Oxford, Oxford OX1 3QD, U.K.. He is now with the Centre for Cyber Security and Information Systems, Cranfield University, Cranfield MK43 0AL, U.K., and also with Defence Academy of the United Kingdom, Wiltshire SN6 8LA, U.K.

M. Goldsmith and S. Creese are with the Cyber Security Centre, University of Oxford, Oxford OX1 3QD, U.K.

threat scenarios for experimentation that are designed to cover a variety of different types of insider attacks that are often observed. It was found that the system performed significantly well for detecting the attacks using the classification alerts, and the visualization enabled analysts to identify what particular attributes caused the insider to be detected. The remainder of the paper is as follows: Section II discusses the related works. Section III describes the requirements of an insider threat detection system. Section IV presents the proposed system, describing in detail the different components. Section V presents process of constructing effective simulation data and the experimentation of the detection system, and Section VI concludes the paper.

II. Related Works

The topic of insider threat has recently received much attention in the literature. Researchers have proposed a variety of different models that are designed to prevent or detect the presence of attacks (e.g., [3], [4]). Similarly, there is much work that considers the psychological and behavioural characteristics of insiders who may pose a threat as means for detection (e.g., [5], [6], [7]). Kammüller and Probst [8] consider how organisations can identify attack vectors based on policy violations, to minimise the potential of insider attacks. Likewise, Ogiela and Ogiela [9] study how to prevent insider threats using hierarchical and threshold secret sharing. For the remainder of this section, we choose to focus particularly on works that address the practicalities of designing and developing systems that can predict or detect the presence of insider threat.

Early work by Spitzner [10] discusses the use of honeypots (decoy machines that may lure an attack) for detecting insider attacks. However, as security awareness increases, those choosing to commit insider attacks are finding more subtle methods to cause harm or defraud their organisations, and so there is a need for more sophisticated prevention and detection. Early work by Magklaras and Furnell [11] considers how to estimate the level of threat that is likely to originate from a particular insider based on certain profiles of user behaviour. As they acknowledge, substantial work is still required to validate the proposed solutions. Myers *et al.* [12] consider how web server log data can be used to identify malicious insiders who look to exploit internal systems. Maloof and Stephens [13] propose a detection tool for when insiders violate need-to-know restrictions that are in place within the organisation. Okolica *et al.* [14] use Probabilistic Latent Semantic Indexing with Users to determine employee interests, which are used to form social graphs that can highlight insiders. Liu *et al.* [15] propose a multilevel framework called SIDD (Sensitive Information Disseination Detection) that incorporates network-level application identification, content signature generation and detection, and covert communication detection.

More recently, Eldardiry *et al.* [16] also propose a system for insider threat detection based on feature extraction from user activities. However, they do not factor in role-based assessment. In addition, the profiling stage that we perform allows us to extract many more features beyond the activity counts that they suggest. Brdiczka *et al.* [17] combine psychological profiling with structural anomaly detection to develop an architecture for insider-threat detection. They use data collected from the multi-player online game, World of Warcraft, to predict whether a player will quit their guild. In contrast to real-world insider threat detection, they acknowledge that the game contains obvious malicious behaviours, however they aim to apply these techniques to real-world enterprises. Eberle *et al.* [18] consider Graph-Based Anomaly Detection as a tool for detecting insiders, based on modifications, insertions and deletions of activities from the graph. They use the Enron e-mail dataset [19] and cell-phone traffic as two preliminary cases, within the intention of extending to the CERT insider threat datasets. Senator *et al.* [20] propose to combine structural and semantic information on user behaviour to develop a real-world detection system. They use a real corporate database, gather as part of the Anomaly Detection at Multiple Scales (ADAMS) program, however due to confidentiality they can not disclose the full details and so it is difficult to compare against the work. Parveen *et al.* [21] use stream mining and graph mining to detect insider activity in large volumes of streaming data, based on ensemble-based methods, unsupervised learning and graph-based anomaly detection. Parveen and Thuraisingham [22] extend the work with an incremental learning algorithm for insider threat detection that is based on maintaining repetitive sequences of events. They use trace files collected from real users of the Unix C shell [23], however this public dataset is relatively dated now.

One clear observation from these related works is that access to real-world data is extremely difficult, and so researchers synthesise data that is similar to that of a real-world enterprise, or use a subset of data points, or apply insider threat detection techniques to other problem domains (e.g., online games). In our work, we particularly wanted to represent the variety and volume of data that would be observed in a modern real-world organisation, and show how this could be combined to form an overall assessment for each user and for each role. We also wanted to clearly demonstrate a wide variety of insider threat scenarios as represented by our synthetic data generation, and show how our detection system would be capable of detecting the different attacks.

III. Requirements Analysis

The work described in this paper was carried out as part of a wider inter-disciplinary project that includes computer scientists, security researchers, and cyber-psychology experts. As the problem of insider threat continues to be of growing concern to businesses and governments alike, there becomes a critical need for practical tools to help alleviate the threat that is posed. Our understanding of what we believe to constitute as insider threat is the result of close inter-disciplinary collaboration between industry, government and academia. The system that is proposed here aims to address the majority of scenarios that are understood from the knowledge that has been shared by organisations experiencing such attacks, and case studies that have been documented in research reports and the media.

Our initial work on insider threat detection was to develop a conceptual model of how a detection system could connect the

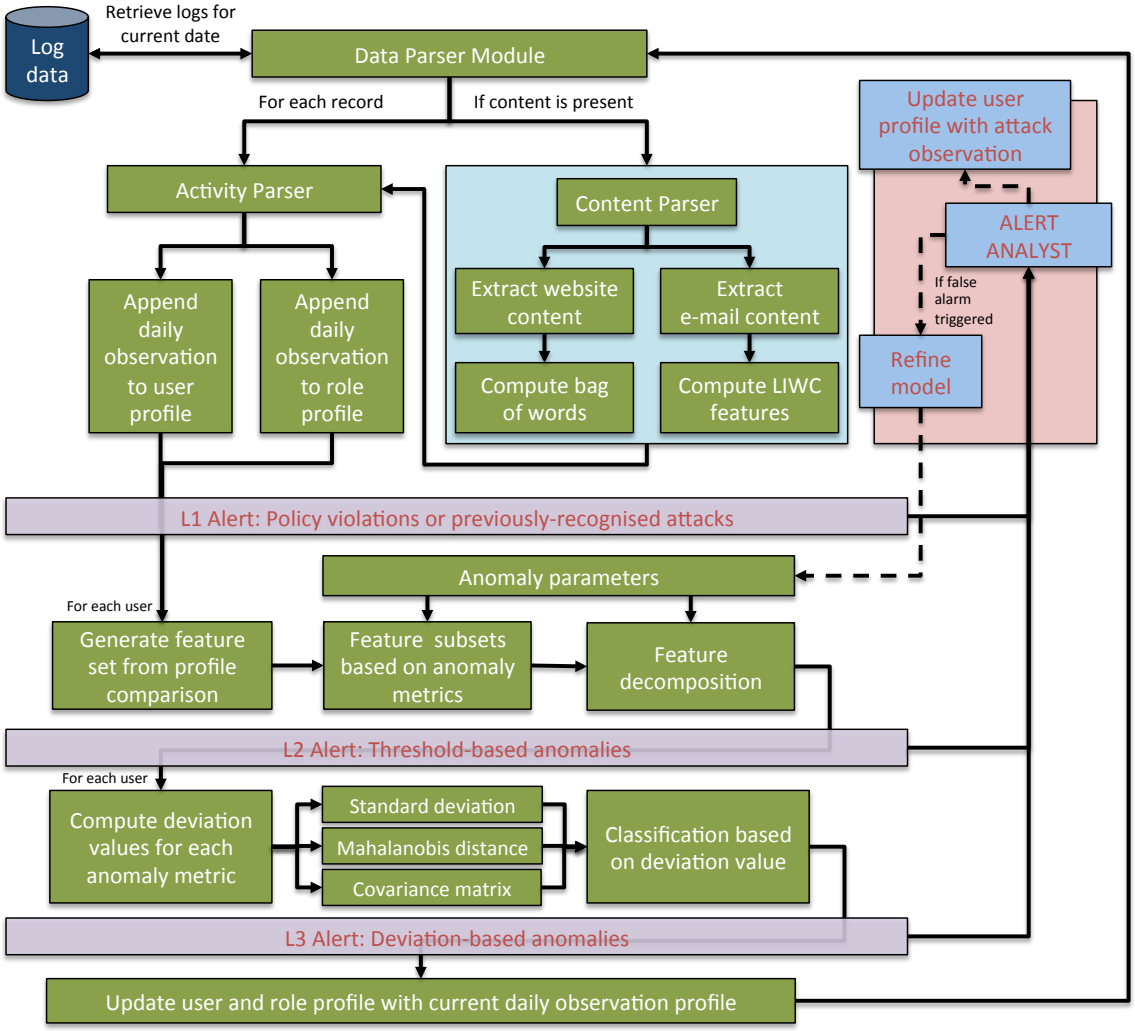


Fig. 1. Architecture of the insider threat detection system. The system comprises of a number of key components that process incoming log data records and construct a profile of user and role behaviour for the current day, and assess the level of threat posed by the invididual. Alerts can be automatically triggered at three levels: policy violations and previously-recognised attacks, threshold-based anomalies or deviation-based anomalies. Alerts are dealt with by an analyst who can then determine whether the individual does actually pose a threat or not. If deemed not to be a threat, the analyst can refine the detection model to minimise the false positive rate for future observations.

actions of the real world with the hypothesis that a particular individual is an insider [3]. It is crucial that organisations looking to deploy insider threat detection tools have a clear understanding of the valuable assets of the organisation, and the monitored activities that relate to these assets, to therefore understand the type of attacks that could potentially arise. In developing our conceptual model, we identified the different elements that exist within organisations to understand what elements could be affected as a result of an insider attack. As a result, we can define the requirements of the detection system as given below:

- The system should be able to determine a score for each user that relates to the threat that they currently pose.
- The system should be able to deal with various forms of insider threat, including sabotage, intellectual property theft, and data fraud.
- The system should also be able to deal with unknown cases of insider threat, whereby the threat is deemed to be an anomaly for that user and for that role.
- The system should assess the threat that an individual poses based on how this behaviour deviates from both their own previous behaviour, and the behaviour exhibited by those in a similar job role.

Whilst we aim for a well-defined detection system that can alleviate the presence of insider threat, to promise a system that can eradicate the problem is a bold claim that we do not try to state here. By the very nature of an insider attack, a sophisticated attacker would be conscious of covering their

tracks to avoid being detected. For example, they could attempt to falsify or delete the activity logs that are reported to the detection system, or they could attempt to circumvent standard monitoring practices. In theory, the very nature of modifying or deleting log files should be detected and so should raise an alert, given that this behaviour should not be deemed as normal. Such attacks would therefore most likely be detected through a combination of both online and offline behaviours, such as acting suspiciously in the workplace.

IV. System Overview

The architecture of the detection system is detailed in Figure 1. Here, the detection system connects with a database that contains all available log records that exist for the organisation. Such examples may be computer-based access logs, e-mail and web records, and physical building access (e.g., swipe card logs). All records for the current date are retrieved and parsed by the system. For each record, the user ID is used to append the activity to their daily observed profile. Likewise, the activity is also appended to the daily observed profile of their associated role, if applicable. Once the daily observation profiles are constructed, the system proceeds to assess each user based on three levels of alerts: policy violations and previously-recognised attacks, threshold-based anomalies, and deviation-based anomalies. At each stage in the assessment, the system can trigger an alert to the analyst to notify of a supposed threat being observed. The analyst can investigate the alert, and then decide whether this alert is correct. Should the analyst decide that the alert is not correct, then they have the capability to reject a detection result which then refines the parameters within the system, to minimise the false positive rate for future observations.

In the following sections, we will detail how each of the key components of the system are performed to identify at-risk individuals. We consider the key components of the system to be the retrieval of records from the organisational database, user and role-based profiling, profile feature extraction, anomaly assessment from features, and classification of threat from anomaly scores. For this work, a pilot detection system was developed using the Python programming language. In addition, visualization components have also been developed that allow the analyst to explore different components of the detection process, such as user profiles and multiple anomaly scores. Our visualization components are developed using a Python back-end and the popular D3 javascript library for the front-end display [24].

A. Data Input

At the first stage of the pipeline is the Data Parser Module that interfaces with the organisation. For each day, the system requests the set of records from the log data that correspond with the current date. In theory, this could consist of many different captures of data from different sensors within the organisation. Our initial work was based on the datasets provided by CMU-CERT. In these datasets, the organisation activity logs consist of five different files that correspond to the different activities that can be performed: login, usb device, e-mail, web, and file access. Each record is parsed to obtain a timestamp, a user ID, a device ID (i.e., what device logged the action) and an activity name (e.g., login, e-mail). Some activities (i.e., e-mails, files, websites) may also contain further information that we assign as the attribute, such as the e-mail recipients, the filename accessed, or the website accessed. Where an attribute is provided, the system is also capable of retrieving and analysing content that can be assigned as the final property of the record, which is handled by the Content Parser.

The Content Parser consists of two main techniques of analysing textual data: bag of words, and Linguistic Inquiry Word Count (LIWC) [25]. For analysing website and file content, Content Parser will scrape the given URL and retrieve all text that is recognised to exist within the English dictionary. Using a bag of words approach to construct a feature set, this feature vector is assigned to the given record. Similarly, for e-mail content we construct a feature vector, however rather than using the raw text content, we use features defined by LIWC. The justification of this is three-fold. Firstly, given the sensitivity of e-mail content, many organisations are concerned with directly monitoring the content of e-mails. Secondly, the LIWC categories have well-defined meaning with regards to psychological context, and so could provide more meaningful information regarding the e-mail content than the raw message would do in any case. Finally, there are 80 features defined by the LIWC tool, so it means that the size of the feature vector can easily be reduced. It would be possible to use either technique for assessment of each activity, however we make this distinction due to e-mails being user-generated, rather than websites or files that are only being read by a user. Each content-based feature vector is combined with the user and role-based daily observation profiles, which we will describe further in Section IV-B.

The Content Parser serves as an optional module within our architecture. It is understood that many organisations currently do not maintain records of all content from e-mails being sent, due to privacy concerns. However, organisations may well change their position on this, especially if it is believed that such content would help in combatting against the threat of insider attacks. For the development of our system, we have worked with a number of synthetic data sets including CMU-CERT insider threat scenarios, the published Enron e-mail dataset, sample data provided by CPNI (Centre for the Protection of National Infrastructure), and in-house generated data. One challenge with using synthetic datasets such as CMU-CERT and our own, is that whilst the data may show that e-mails were sent or files were accessed, since these are purely synthetic there is no substantial content within the files or e-mails. E-mail content may be a collection of randomly-chosen words that define a topic, rather than a meaningful communication sent by a human user. Whilst we have been able to trial such methods on e-mail and web analysis in isolation, without these pairing up with corresponding insider threat scenarios it is difficult to truly validate the approach. However, it is incorporated into the overall architecture since it serves as an optional complimentary anomaly metric that analysts can choose whether to utilise, based on the availability

of data.

B. User and Role-based Profiling

The second stage of the system is user and role-based profiling. Each user and each role that exists within the organisation is defined by a tree-structured profile that describes the different devices, activities, and attributes that they have been observed on. This notion of a tree-structured profile provides a consistent representation for all users, and for all roles that can be used for comparative assessment, either between multiple employees, or between multiple time steps for a particular employee. Figure 2 illustrates the profile of a typical user using our tree visualization tool. At the root of the tree is the user ID (or in the case of a role tree, the role title), which can consist of three child nodes: observations made for the current date (daily), observations that have previously been made and exist within the normal profile (normal), and if applicable, observations that have been deemed to be suspicious (attack). For each of these branches, we define the same hierarchical structure to facilitate comparison. At the first level down, all devices that the user has been observed on are given. In the case of a role tree, this would be all devices observed by all users who act within this particular role. These typically would be computers, however this could well be extended to other electronic devices such as printers or door locks. The next level of the tree shows all the activities the user has performed on each of these devices. The level below this then shows the attributes, if applicable, such as the files or web sites accessed, or the e-mail addresses that the user has contacted. Each node in the profile maintains a 24-bin histogram that denotes the hourly usage for that particular state, based on the observed records. In addition, attributes can also maintain the results of the Content Parser as a cumulative histogram.

For each record, the system first compares this record against the state of the user's current daily profile. If the device-activity-attribute tuple does not exist within the tree-structured profile, then a new node is created at the appropriate location within the daily profile tree. The associated histogram for the node is then updated based on the timestamp of the observed record. Similarly, the tuple is also compared against the corresponding role profile that the user belongs to. This provides a profile that describes the currently daily activity for all users and for all roles. If the user belongs to multiple roles, then the system can be configured to either populate all roles that the user belongs to, or to create a specific role type that is then populated (e.g., *technician-engineer* could define the set of users who act in both roles). Once all daily records have been observed, the next stage of the system is to derive a feature set that provides a comprehensive and comparable description of each user's profile. To do this, the system compares the current daily profile against the existing previous profile.

Once the daily observation profile is constructed, the system can perform a comparison against organisational policies and previously-recognised attack patterns. A rule-based approach can be specified using a policy language, that can be used to state how particular observations should be treated (e.g., all logins out of hours should be flagged to the analyst with a medium severity level). If there are no violations flagged up at this stage then the system proceeds to the next level.

C. Feature Extraction

Once we have computed the current daily profile for each user and for each role, we perform our feature extraction. Since the profile structure is well-defined, it means that a wide variety of comparisons between users, roles, or time steps can be easily made. We define a series of features that consider new observations across devices, activities, and attributes, for the user compared against their previous behaviours, and for the user compared against the previous behaviour of all users within the same role (e.g., *New device for user*, *New activity for device for role*, *New activity for any device for user*). We also define a series of features that assess the hourly and daily usage counts for each particular device, activity, and attribute (e.g., *Hourly usage count for device*, *Hourly usage count for activity*, *Hourly usage count for attribute*, *Daily usage count for activity*). Finally, we define time-based features for each particular device, activity, and attribute (e.g., *Latest logon time for user*, *Earliest USB time for user*, *USB duration for user*). The full feature matrix that we currently consider consists of 168 columns (the full list of extracted features is available in [26]). The complete set of features allow for assessment of three key areas: the user's daily observations, comparisons between the user's daily activity and their previous activity, and comparisons between the user's daily activity and the previous activity of their role.

D. Threat Assessment

Once the feature set for the current daily observation has been computed, the next stage of the system is to determine whether these features show significant deviation in behaviour compared with all previously-accepted observations. To do this, an $n \times m$ matrix is constructed for each user, where n is the total number of sessions (or days) being considered, and m is the number of features that have been obtained from the profile. The bottom row of the matrix represents the current daily observation, with the remainder of the matrix being all previous observation features. To derive the amount of variation that is exhibited in the multivariate feature space, we perform PCA to obtain a projection of the features into lower dimensional space based on the amount of variance exhibited by each feature. What this means is that features that have a higher variance can be projected into a lower-dimensional space whilst preserving separability between similar and dissimilar features. It is often used to enable visualization and understanding of large datasets using only 2 or 3-dimensions, to observe the clustering of similar data records. For our application, we also allow a weight to be associated with each feature so that features of greater importance can be emphasised, as dictated by an analyst. In this way, the analyst can generate different models for analysis based on different configurations of weighted combinations. If no weights are specified then the weight is taken to be $1/f$ where f is the

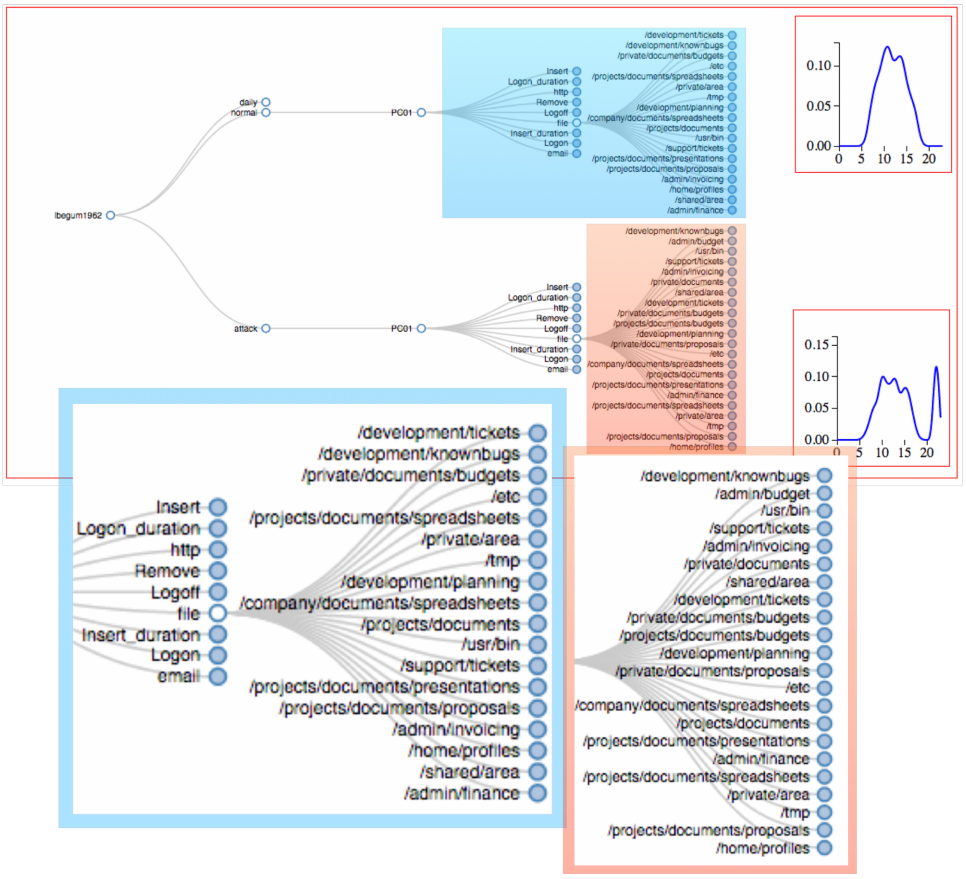


Fig. 2. Tree-structured profiles of user and role behaviours. The profile shows all the devices, activities, and attributes that the user has been observed performing. The probability distribution for normal hourly usage is given in the top-right, and the distribution for the detected attack is given in the bottom-right. Here it can be seen that the user has accessed resources late at night.

total number of features. All feature columns are normalized before the PCA decomposition is performed. By default, we consider a decomposition of the features to a 2-dimensional space. If all feature observations were identical, then all points in the new space would be clustered at the origin. However, given the deviation that is expected of human behaviour, points are likely to be clustered near to, but not directly at, the centre. For the new matrix, we consider only the current observation, which is the bottom-most record in the matrix. We compute the distance of this point from the origin in the new space, and take this to be the anomaly score of this metric at this observation. This process is performed for each of the anomaly metrics, where each metric consists of a subset of the overall feature set, and if specified, a corresponding weighting function for each feature. Each anomaly metric can be configured to alert if the score obtained for that particular metric is above a particular threshold.

The anomaly metrics that are currently considered include: *Login_anomaly*, *Login_duration_anomaly* , *Logoff_anomaly* , *USB_inserstion_anomaly* , *USB_duration_anomaly* , *Email_anomaly* , *Web_anomaly* , *File_anomaly* , *This_anomaly* , *Any_anomaly* , *New_anomaly* , *Hourly_anomaly* , *Number_anomaly* , *User_anomaly* , *Role_anomaly* , and, *Total_anomaly*. The system could easily support the addition of further anomaly metrics, based on the observation of different activity types. From our researching into case studies of insider threat, most cases could be associated with either performing a new activity, performing an existing activity at a new time of day, or performing an existing activity more or less often than previously. These define our ‘new’, ‘hourly’, and ‘number’ metrics. The combination of multiple metrics also provide support for greater confidence in the result obtained regarding an individual. For example, we may observe that a particular individual scores higher than other users not only on ‘hourly_anomaly’ or ‘total_anomaly’, but also on ‘file_anomaly’ and ‘email_anomaly’. By considering how the different subsets of features score, rather than a single overall score, it allows an assessment to be made on

#	# users	# records	# days modified	Malicious activities
1	100	2486663	4	(20 file, 4 email)
2	305	8452267	5	(34 file, 5 http)
3	12	115745	10	(34 file, 20 usb, 10 login)
4	50	905053	5	(10 login, 9 http)
5	200	2692373	1	(200 email)
6	100	2514792	2	(2 file, 1 login, 2 usb, 4 email)
7	305	8458402	2	(3 file, 6 email)
8	12	117195	10	(10 login, 24 file)
9	50	893700	4	(6 login, 1 file, 2 usb)
10	200	2697772	1	(2 http, 1 file)

TABLE I. CHARACTERISTICS OF THE 10 INSIDER THREAT SCENARIOS, INCLUDING THE VOLUME AND TYPE OF INSERTED MALICIOUS ACTIVITY. EACH SCENARIO CONSISTS OF 365 DAYS OF ACTIVITY LOGS.

not only that an individual is posing as a threat, but also on what attack vectors they are acting on. Here, we observe that the user is logging in at an unusual time to access new files and e-mail new contacts.

E. Classification of threat

The final stage of the system is to provide assessment of the threat that is posed by an individual, given the observation of their activity, and the collection of anomaly scores that have been assigned to their daily observation profile. One approach to this as a relatively effective measure is to simply normalise each column of the anomaly score matrix, and then take the maximum standard deviation as a integer classification of importance. We would expect most data to exist within 2 standard deviations of the norm, so anything above this should certainly be investigated. Likewise, we can also compute the mahalanobis distance to assess how far away an individual's observation are from the rest of the distribution. As a third approach that can be deployed, we compute the covariance matrix of a user's anomaly scores, and on each daily observation assess the signed differences between the covariances. The system could well be extended to support other classification schemes in the future as desired by the analyst. The classification can be used flag up users to the analyst, and also to determine whether a user's daily observation profile should be included within their previously-observed normal profile. If the observation is deemed to be too much of an anomaly, then the observation is recorded as an attack rather than their normal. This is a vital stage so as to not contaminate a user's previously-observed profile with malicious behaviour, whilst also providing the capability for each daily observation to contribute towards the previously-observed profile.

V. EXPERIMENTATION

To be able to assess the performance of the detection system, we conduct a series of experimentation scenarios using the prototype system. As part of the wider project on insider threat, 10 scenarios have been developed that cover the broad range of possible attacks that an insider could perform against their organisation. For each scenario, a narrative has been devised that explains what has happened, including why the individual has chosen to act against the organisation, and what they have done. Each scenario is modelled within a unique synthetically-generated dataset that represents the normal activity of the organisation. The data contains all employee activity within the organisation for the period of 365 days, including that of the insider. We consider the first 15 days as training data, where no attacks are initiated, so that an initial normal baseline can be obtained. The remaining 350 days are then used as testing, whereby each newly-observed day that is deemed to be normal then contributes towards the normal baseline. The scenarios were developed in isolation of the detection system, so not to have been bias by this, and have been designed to test a variety of different scenarios that could occur over different attack vectors. In addition to our own synthetic data, we have also used third-party datasets generated by CMU-CERT to further validate the performance of the approach described.

A. Constructing experimentation data

The creation of the synthetic datasets was conducted in isolation of the detection system so as to not introduce any bias. The premise of the activity was to craft a synthetic organisation for each scenario, and insert a malicious employee in such a way that their behaviours correspond with those that have been documented by the various case studies of previously-observed attacks. All the while, the intent was to create different scenarios with the objective of beating the detection system, within the confines of the data points available as described in Section IV-A.

The approach used to generate the data sets involved an automated system to generate the normal day-to-day activity (the background noise) and then the attack data was manually injected into the log files. The method used to create the normal activity has focused on the notion of defining a 'virtual organisation'. In our system an organisation is composed of a number of staff roles (e.g. manager, developer), with a number of employees in each of the roles.

The employees role is used as the seed for the data generation process and determines the boundaries of normal behaviour of an employee undertaking that role. An employee's role, within our virtual organisation, defines the normal boundaries of behaviour over a number of data dimensions including: log-in times, USB device insertions, HTTP requests, email contacts, emails sent and file system accesses. The role does not provide entirely uniform behaviour; there are only average values for an employee in that role. For example, an employee in an administrative role may typically log-in to the system between 8am and 10am and log-out between 4pm and 6pm. The data generation system would, typically, assign the employee a log-in time within the specified window but there is also the provision to generate occasional, anomalous values outside of this window.

Once the normal activity has been generated, then the malicious activity is manually inserted into the datasets. For each of the attacks inserted an attack scenario was written, specifying the type of employee (i.e. the employee's role) and describing the nature of the attack. For example, a scenario may specify that a manager, within the organisation, had been arriving at work earlier than they normally would and

#	L2 alerts ($\sigma > 0.1$)	L2 alerts ($\sigma > 0.2$)	L2 alerts ($\sigma > 0.3$)	L3 alerts ($\sigma = 1.0$)	L3 alerts ($\sigma = 2.0$)	L2 anomaly vectors	L3 anomaly vectors
1	935	415	352	276	75	n/a	n/a
2	3033	1481	1259	964	293	n/a	n/a
3	145	92	88	63	24	logon, logon_duration	insert, file, hourly, user, total
4	373	120	83	68	14	logoff, user, role, number, new	logon, logon_duration, total
5	1573	553	474	391	82	user, this	new, email
6	906	394	340	287	52	user, new, this	n/a
7	3068	1462	1254	977	276	n/a	user, file
8	160	94	90	64	25	logon, logon_duration	user, file, total
9	358	125	89	68	20	logon, logon_duration	n/a
10	1645	610	526	452	73	n/a	new, number, user, role, hourly, total

TABLE II. RESULTS FROM 10 INSIDER THREAT SCENARIOS FOR LEVEL 2 AND LEVEL 3 ALERTS.

browsing to new areas of the corporate network that they had not previously visited. Once a scenario is created then an employee in the correct role is selected and the attack data is inserted into the log files. Owing to the random nature of the data generation process very little was known about the behaviour of the 'malicious' employee prior to the insertion of the attack data. The data inserted, about an attack, relates directly to the employee's behaviour, rather than that of the role. If we consider the earlier example of a manager who begins to log-in earlier than before, and accesses new areas of the corporate network then the earlier log-ins would early for that particular employee, not the role as a whole. This makes the attack insertion slightly more subtle and harder to identify. Details of each synthetic dataset are provided in Table I.

B. Results

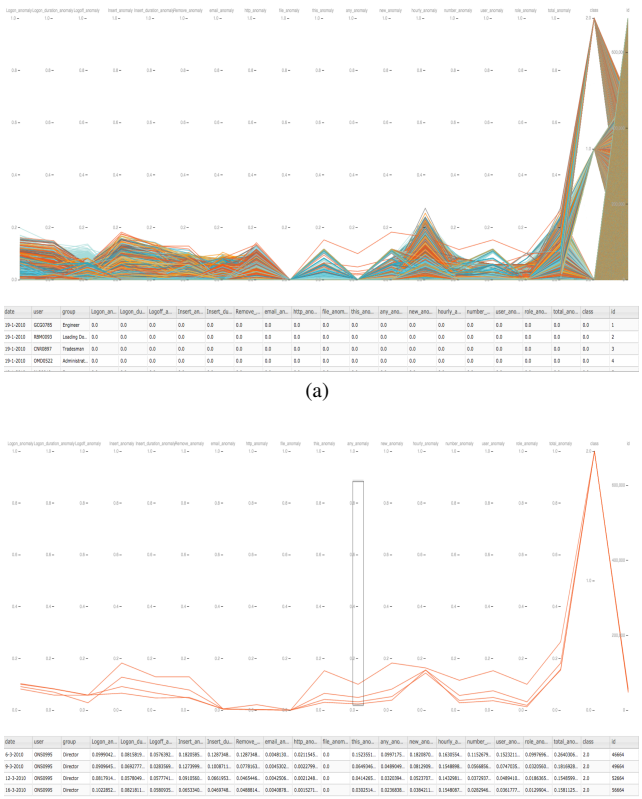
Table II shows the results from the detection system. We show the number of alerts that are generated under different operational schemes for Level 2 and Level 3 alerts. In addition, we show the anomaly metrics that the alerts were triggered for. In this experiment, it is clear that L3 alerts with a deviation of $\sigma = 2.0$ gave the fewest alerts. In real-world operation, it may well be beneficial to preserve alerts generated under different operational schemes, for instance, to observe that an employee is consistently scoring just below a particular threshold. This knowledge, coupled with offline behaviours, could well reveal the employee to be a threat, which would have been missed otherwise. From these results, the best result is obtained from scenario 3. Here, there are 4200 daily assessments made (12 employees for 350 days), of which 24 are flagged as anomalies. From Table I, we see that 10 of the days consisted of malicious activity, of which all 10 days are within the set of 24 detected anomalies. Based on precision and recall, this gives a precision of 42% and a recall of 100%. Whilst it is clear that the system still presents some error, the effort of an analyst to investigate 24 results rather than 4200 is still clearly advantageous. The classification of either being an insider threat or not is somewhat of an ambiguous task, since it is highly dependent on context, and it also involves the analyst or managers to determine what the next course of action should be regarding the individual. What is perhaps most important from any insider threat detection system is that recall is ensured over precision. In this sense, the detection system serves as a means to filter a substantial number of assessments to alleviate the efforts of the analyst.

Futhermore, we also present our results using a parallel co-ordinates plot that shows each of the anomaly metrics as an individual axis (as shown in Figure 3). This example is shown for a scenario generated by CMU-CERT, where the dataset consists of 1000 employees, of which 1 is an insider. Figure 3(a) shows 691000 daily observations on the plot, and yet there is a distinct polyline which is seen to be an outlier on multiple axes. By brushing the axis, as shown in Figure 3(b), the analyst can filter the data and reveal information on this particular case. Here, there are now only 4 observations, all of which are the actions of the inserted insider who copied data to a USB drive during unusual work hours. By coupling the detection results with a visual analytics approach, this empowers the analyst much more to be able to identify anomalous behaviour within the daily observation records.

In our experimentation, we found that 7 of the 10 cases were clearly identifiable when using the parallel coordinates plot. Of the cases that did failed, there are a number of factors that could impact on the performance of the system. Firstly, one of the scenarios that failed was dependent on the content of a website, rather than the unusual access of it. Whilst the architecture of the system supports content, we did not include this in for the synthetic experimentation as the website addresses were randomly created and so access to these would not be feasible. Secondly, despite the synthetic data being modelled to reflect human behaviour, it is difficult to truly capture the intentions and motivations of the employees who are supposedly acting normally. Therefore, there is possibility that the normal background data exhibits noise and randomness that real data should not have. Having said this, it is also possible that the opposite could be true for some organisations, and that in fact the synthetic data is too simple and not truly reflecting the dynamic nature of real human behaviour. Nevertheless, we believe that the results presented here, for threshold and deviation-based assessment, and for visual assessment of anomalies, are encouraging for our initial experimentation. We are currently working with a large international corporation to deploy our experimentation system within their environment to test our system against real-world activity data.

VI. CONCLUSIONS AND FUTURE WORK

In this work, we have presented an effective approach for insider threat detection. From the organisational log data, the system generates user and role-based profiles that can describe the full extent of activities that users perform within the



(a)

(b)

Fig. 3. Parallel coordinates plot of the multiple anomaly results that are generated by the detection process. (a) Plot shows 691000 results for a year and a half of monitoring 1000 employees. It can be seen that there exists a record that can be described as an outlier over at least four of the different metrics. (b) Plot shows 4 results that have been highlighted by the *any_anomaly* metric that all correspond to a particular individual. In this example, this invididual was the inserted malicious insider who began using a USB device to copy sensitive records earlier than they would normally act.

organisation. The tree-structured profiles are designed to be easily comparable against other users, role types, and temporal observations. From each daily observation, the system constructs a large set of features that describe the state of the current daily profile, and the previously observed profiles for all users. The system then creates subsets of the features that describe particular anomalies of interest, and computes a PCA decomposition on this to identify features that exhibit high deviation. Alerts are generated when anomaly scores are deemed to be over a particular threshold, measured as the standard deviation from the normalized anomaly scores. From an alert, the analyst can visualize how the user differs from their normal behaviour, or from other users, using a range of visualization techniques. We demonstrate this approach for a variety of synthetically-generated insider threat scenarios, both from our own development and from CMU-CERT, and find that the system performs well for identifying these attacks across the range of anomaly metrics that are considered.

Clearly, by the very nature of an insider threat, the individual in question is purposely attempting to stay below the radar, and so to guarantee 100% detection success is difficult since there could be a number of attacks that are not considered by the designers of the detection system. Our future work is to explore the notion of model evolution, and how multiple detection models could operate in parallel. In our current architecture, we have shown the process of refining the current model, but what if the analyst chose to maintain both models and compare the two? The analyst would then need to be able to assess the performance of each model over time, to decide whether it is worth utilising all models, or whether some models should be discarded. There are also organisational-dependant characteristics that may need to be considered, however the approach described is designed to be flexible to the forms of data that different organisations may collect. We are currently conducting experiments with a large real-world organisation to see how effective the tools can be when studying real users, and in particular, the differences between real normal and real threats. We are also exploring whether decomposition to different levels of dimensionality can improve the precision results for the detection system, to further alleviate analyst efforts. What is very clear however, is that organisations recognise that a real threats exist, and that such systems as this could well detect and alleviate the efforts that are required of organisational security analysts.

ACKNOWLEDGEMENTS

This research was conducted in the context of a collaborative project on Corporate Insider Threat Detection, sponsored by the UK National Cyber Security Programme in conjunction with the Centre for the Protection of National Infrastructure, whose support is gratefully acknowledged. The project brings together three departments of the University of Oxford, the University of Leicester and Cardiff University.

REFERENCES

[1] D. M. Cappelli, A. P. Moore, and R. F. Trzeciak. *The CERT Guide to Insider Threats: How to Prevent, Detect, and Respond to Information Technology Crimes*. Addison-Wesley Professional, 1st edition, 2012.

[2] I. Jolliffe. *Principal component analysis*. Wiley Online Library, 2005.

[3] P. A. Legg, N. Moffat, J. R. C. Nurse, J. Happa, I. Agrafiotis, M. Goldsmith, and S. Creese. Towards a conceptual model and reasoning structure for insider threat detection. *Journal of Wireless Mobile Networks, Ubiquitous Computing and Dependable Applications*, 4(4):20–37, 2013.

[4] M. Bishop, B. Simidchieva, H. Conboy, H. Phan, L. Osterwell, L. Clarke, G. Avrunin, and S. Peisert. Insider threat detection by process analysis. In *IEEE Security and Privacy Workshops (SPW)*. IEEE, 2014.

[5] M. Bishop, S. Engle, S. Peisert, S. Whalen, and C. Gates. We have met the enemy and he is us. In *Proc. of the 2008 workshop on New security paradigms (NSPW'08), Lake Tahoe, California, USA*, pages 1–12. ACM, September 2008.

[6] F. L. Greitzer and R. E. Hohimer. Modeling human behavior to anticipate insider attacks. *Journal of Strategic Security*, 4(2):25–48, 2011.

[7] J. R. C Nurse, O. Buckley, P. A. Legg, M. Goldsmith, S. Creese, G. R. T. Wright, and M. Whitty. Understanding insider threat: A framework for characterising attacks. In *IEEE Security and Privacy Workshops (SPW)*. IEEE, 2014.

[8] F. Kammueller and C. W. Probst. Invalidating policies using structural information. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, 5(2):59–79.

[9] M. R. Ogiela and U. Ogiela. Linguistic protocols for secure information management and sharing. *Computers & Mathematics with Applications*, 63(2):564–572, January 2012.

[10] L. Spitzner. Honeypots: catching the insider threat. In *Proc. of the 19th IEEE Computer Security Applications Conference (ACSAC'03), Las Vegas, Nevada, USA*, pages 170–179. IEEE, December 2003.

[11] G. B. Magklaras and S. M. Furnell. Insider threat prediction tool: Evaluating the probability of IT misuse. *Computers and Security*, 21(1):62–73, 2002.

[12] J. Myers, M. R. Grimaila, and R. F. Mills. Towards insider threat detection using web server logs. In *Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies*, CSIIRW '09, pages 54:1–54:4, New York, NY, USA, 2009. ACM.

[13] M. A. Maloof and G. D. Stephens. Elicit: A system for detecting insiders who violate need-to-know. In Christopher Kruegel, Richard Lippmann, and Andrew Clark, editors, *Recent Advances in Intrusion Detection*, volume 4637 of *Lecture Notes in Computer Science*, pages 146–166. Springer Berlin Heidelberg, 2007.

[14] J. S. Okolica, G. L. Peterson, and R. F. Mills. Using plsi-u to detect insider threats by datamining e-mail. *International Journal of Security and Networks*, 3(2):114–121, 2008.

[15] Liu Y., C. Corbett, K. Chiang, R. Archibald, B. Mukherjee, and D. Ghosal. Sidd: A framework for detecting sensitive data exfiltration by an insider attack. In *System Sciences, 2009. HICSS '09. 42nd Hawaii International Conference on*, pages 1–10, Jan 2009.

[16] H. Eldardiry, E. Bart, Juan Liu, J. Hanley, B. Price, and O. Brdiczka. Multi-domain information fusion for insider threat detection. In *Security and Privacy Workshops (SPW), 2013 IEEE*, pages 45–51, May 2013.

[17] O. Brdiczka, J. Liu, B. Price, J. Shen, A. Patil, R. Chow, E. Bart, and N. Ducheneaut. Proactive insider threat detection through graph learning and psychological context. In *Proc. of the IEEE Symposium on Security and Privacy Workshops (SPW'12), San Francisco, California, USA*, pages 142–149. IEEE, May 2012.

[18] W. Eberle, J. Graves, and L. Holder. Insider threat detection using a graph-based approach. *Journal of Applied Security Research*, 6(1):32–81, 2010.

[19] Bryan Klimt and Yiming Yang. The enron corpus: A new dataset for email classification research. In Jean-Franois Boulicaut, Floriana Esposito, Fosca Giannotti, and Dino Pedreschi, editors, *Machine Learning: ECML 2004*, volume 3201 of *Lecture Notes in Computer Science*, pages 217–226. Springer Berlin Heidelberg, 2004.

[20] T. E. Senator, H. G. Goldberg, A. Memory, W. T. Young, B. Rees, R. Pierce, D. Huang, M. Reardon, D. A. Bader, E. Chow, et al. Detecting insider threats in a real corporate database of computer usage activity. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1393–1401. ACM, 2013.

[21] P. Parveen, J. Evans, Bhavani Thuraisingham, K.W. Hamlen, and L. Khan. Insider threat detection using stream mining and graph mining. In *Privacy, security, risk and trust (PASSAT), 2011 IEEE Third International conference on social computing*, pages 1102–1110, Oct 2011.

[22] P. Parveen and Bhavani Thuraisingham. Unsupervised incremental sequence learning for insider threat detection. In *Intelligence and Security Informatics (ISI), 2012 IEEE International Conference on*, pages 141–143, June 2012.

[23] S. Greenberg. Using unix: collected traces of 168 users. Technical report, 1988.

[24] M. Bostock, V. Ogievetsky, and J. Heer. D3 data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, December 2011.

[25] Y. R. Tausczik and J. W. Pennebaker. The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of Language and Social Psychology*, 29(1):24–54, 2010.

[26] P. A. Legg, O. Buckley, M. Goldsmith, and S. Creese. Caught in the act of an insider attack: Detection and assessment of insider threat. In *IEEE International Symposium on Technologies for Homeland Security (HST 2015)*, (in press).

Philip A. Legg is a Senior Lecturer at the University of the West of England. His interests span across several research areas, including machine learning, data visualization, human-computer interaction, and computer vision. Prior to his current position, he was a post-doctoral research associate in the Cyber Security Centre at the University of Oxford. He worked on the Corporate Insider Threat Detection project to develop techniques for the detection and prevention of insider threat. He has also worked at Swansea University exploring data visualization for sports video analysis. He obtained both his BSc and his PhD in Computer Science from Cardiff University, where his research focused on multi-modal medical image registration.

Oliver Buckley received his BSc in Computer Science (University of Liverpool, 2004) and a PhD in Computer Science from Bangor University specialising in soft tissue deformation simulation using haptics and visualisation. He has experience within industry as a software engineer and also in academia. This has included working on the CITD project as a Cyber Security Researcher at the University of Oxford. Oliver is currently employed as a lecturer at Cranfield University in the Defence Academy College of Management and Technology and is a member of the Cyber and Influence research group.

Michael Goldsmith is a Senior Research Fellow at the Department of Computer Science and Worcester College, Oxford. With a background in Formal Methods and Concurrency Theory, Goldsmith was one of the pioneers of automated cryptoprotocol analysis. He has led research on a range of Technology Strategy Board and industrial or government-funded projects ranging from highly mathematical semantic models to multidisciplinary research at the social-technical interface. He is an Associate Director of the Cyber Security Centre, Co-Director of Oxford's Centre for Doctoral Training in Cybersecurity and one of the leaders of the Global Centre for Cyber Security Capacity-Building hosted at the Oxford Martin School, where he is an Oxford Martin Fellow.

Sadie Creese is Professor of Cybersecurity in the Department of Computer Science at the University of Oxford. She is Director of Oxfords Cyber Security Centre, Director of the Global Centre for Cyber Security Capacity Building at the Oxford Martin School, and a co-Director of the Institute for the Future of Computing at the Oxford Martin School. Her research experience spans time in academia, industry and government. She is engaged in a broad portfolio of cyber security research spanning situational awareness, visual analytics, risk propagation and communication, threat modelling and detection, network defence, dependability and resilience, and formal analysis. She has numerous research collaborations with other disciplines and has been leading inter-disciplinary research projects since 2003. Prior to joining Oxford in October 2011 Creese was Professor and Director of e-Security at the University of Warwicks International Digital Laboratory. Creese joined Warwick in 2007 from QinetiQ where she most recently served as Director of Strategic Programmes for QinetiQ's Trusted Information Management Division.