Towards a User and Role-based Sequential Behavioural Analysis Tool for Insider Threat Detection

Ioannis Agrafiotis*, Philip Legg, Michael Goldsmith, and Sadie Creese
Cyber Security Centre, Department of Computer Science, University of Oxford.

Abstract

Insider threat is recognised to be a significant problem and of great concern to both corporations and governments alike. Traditional intrusion detection systems are known to be ineffective due to the extensive knowledge and capability that insiders typically have regarding the organisational setup. Instead, more sophisticated measures are required to analyse the actions performed by those within the organisation, to assess whether their actions suggest that they pose a threat. In this paper, we propose a proof-of-concept that focuses on the use of activity trees to establish sequential-based analysis of employee behaviour. This concept combines the notions of previously-proposed techniques such as attack trees and behaviour trees. For a given employee, we define a tree that can represent all sequences of their observed behaviours. Over time, branches are either appended or created to reflect the new observations that are made on how the employee acts. We also incorporate a similarity measure to establish how different branches compare against each other. Attacks can be defined as where the similarity measure between a newly-observed branch and all existing branches is below a given acceptance criteria. The approach would allow an analyst to observe chains of events that result in low probability activities that could be deemed as unusual and therefore may be malicious. We demonstrate our proof-of-concept using third-party synthetic employee activity logs, to illustrate the practicalities of delivering this form of protective monitoring.

Keywords: Insider threat, Anomaly detection, Attack trees

1 Introduction

Insider threat remains to be a serious and persistent problem within many organisations and governments. The in-depth knowledge that insiders have of the inner workings of their organisation, and the elevated privileges for accessing sensitive records, places insiders as an organisation's greatest threat, should they ever choose to act as such. Insiders may attempt to: steal information, such as company secrets, customer details, or financial accounts, and subsequently sell it on to rival organisations to gain competitive advantages; sabotage information systems and business processes of organisations as an act of vengeance; disclose classified information committing national security crimes; or unintentionally, due to carelessness or manipulation, misuse organisational resources [4].

Designing a system to prevent, deter, monitor and detect insider threats poses great challenges, which many anomaly detection problems do not suffer from. This is due to the fact that insiders possess knowledge of the organisations' policies, security procedures and computer systems. In addition, most of the times they have authorisation to access core assets of an organisation, establishing them in a unique position to mask their malicious activities. These are not the only challenges, however, since usually the malicious activity is a very small portion of insiders' daily workload, whereas their attack methods constantly evolve creating a dynamic environment. Addressing these challenges requires not

Journal of Internet Services and Information Security (JISIS), volume: 4, number: 4 (November 2014), pp. 127-137
*Corresponding author: Cyber Security Centre, Department of Computer Science, University of Oxford, Parks Road, Oxford, OX1 3QD, UK, Tel: +44-(0)1865-273838, Email: ioannis.agrafiotis@cs.ox.ac.uk

only monitoring for anomalous activities but a deeper understanding of the insiders' motivations and behaviours.

Understanding the activities which insiders conduct may provide a significant benefit towards monitoring, detecting and preventing the actions of a potential attack. In particular, how an insider chooses to conduct activities, including the sequence of the activities, may be indicative of a change in behaviour that could imply threatening activity or reveal their motives. In this paper, we propose the use of activity trees for assessing the behaviours of insiders. Building on the concept of attack trees [18], activity trees show the range of possible activities that an insider may conduct in order to achieve their expected daily workload, rather than being restricted to only the activities that are deemed as an attack. By doing so, we acquire a richer picture of an attack which may reveal insiders' motives and facilitate our understanding of their behaviours.

In what follows, Section 2 provides a description of related work in the area of attack trees, Section 3 details our approach to constructing the activity tree and Section 4 elaborates on how we define the threshold of what is considered an attack and what as a normal activity. Section 5 describes the case study where we apply our approach and we conclude with Section 6 where we provide opportunities for future work.

2 Related Work

The work presented in this paper draws on the concept of attack trees from the security literature and on the concept of behaviour trees from the Artificial Intelligence (AI) literature. Trees are widely used within many areas of Computer Science. They provide a well-defined hierarchy that is well suited for describing sequential activity. Behaviour trees in particular are popular within Artificial Intelligence works, and are used primarily to describe the range of possible actions that a character may take on [10] [12]. For example, if developing a game that involves a sword fight, the character would first need to find the sword, then pick up the sword, and then use the sword, in order to attack. The behaviour tree allows the sequence of activities to be defined, clearly indicating what activity may precede and follow.

Behaviour trees may also capture dynamic behaviours. In [10] for example, Florez et al focus on retrieving behaviours by considering the super state in which the system is and the underlying goals in order to guide Non-Player Characters (NPC) actions which are dynamically built. To our knowledge there is no work addressing the insider threat problem which uses behaviour trees, unlike the attack trees concept.

Schneier coined the term attack trees defining them as "a way of thinking and describing security of systems and subsystems" [18]. Attack trees can be applied to acquire insights on the objectives of the attacks, obtain information about the attackers, formulate hypothesis about possible occurrences of attacks, draw security assumptions of a system and provide a strategy of how to best spend a security budget. Researchers have widely used Schneier's methodological approach to describe attacks by malicious insiders, detail strategies for countermeasures and provide optimal solutions for allocating finance resources on security policies [2].

Moore et al [13] detail in their paper a high-level methodology for structuring attack trees regarding insider threats. Their intention is to identify frequent occurring attack patterns and to use this information to shape the design of information systems towards more secure solutions.

Based on attack trees, Ray et al [17] propose a framework to identify malicious activities from authorized insiders. Their attack trees concern only network vulnerabilities and they provide a formal definition of these trees. In addition they describe algorithms to generate customised trees for insiders in order to compare their activities with the attack trees. They also provide a model to compute the likelihood of an attack based on the activity of the user. The existence of attack trees which describe network

vulnerabilities is a prerequisite for their system and they do not elaborate on how these trees can be built.

In [9] the authors focus on graph-based approaches to identify anomalous instances of "structural patterns" in data that may indicate insider threat activity. By investigating structural patterns, they highlight activities which resemble normal transactions but due to their differences in structural form are designated as attacks. Results from running their algorithms on different data such as e-mail, and business processes illustrate the success of their method. When their approach is applied, however, to dynamic graphs whose structure changes over time results show "minimal success" [9].

Dewri et al [8] describe a systematic approach to perform a cost-benefit analysis on optimising the selection of a subset of security hardening measures from a wider set. Their methodology is driven by constructing attack-tree models of an organisation's network to facilitate the process of optimisation. Similarly to Dewri et al, Poolsappasit et al [16] suggest a risk management framework which is based on Bayesian networks, enabling organisations to estimate the probability of a network being compromised at different layers. They define a formal language to describe a dynamic Bayesian network and demonstrate how to design a risk mitigation plan.

Bayesian network methodologies are applied in [22] as well. This work considers near real-time security analysis, where important types of uncertainty are identified with the use of Bayesian networks. The Bayesian networks are constructed based on existing security graph models. Wang et al [20] extend attack graphs analysis and apply it to intrusion detection. In their work, attack graphs are acquired, and then compared to received alerts in order to hypothesize for missing alerts which are not captured by the intrusion detection system or to predict future threats.

Another stream of literature on insider threats has focused on applying anomaly detection techniques [3]. In [19], the authors provide a synopsis on methods of a research project intended to develop novel approaches to detect insider threats. They used data from a real corporate database of stored activities on their users' computers to verify their approaches on identifying characteristics of insider threats. They applied seventeen different algorithms for anomaly detection, which were shaped based on patterns of well-established malicious insider behaviour. They concluded their project by providing a visual language for describing features, peer groups, baselines and algorithms to detect anomalies indicative of insider threat behaviour.

Parveen et al in [15] and [14] propose an unsupervised, ensemble learning algorithm to create a set of iterative sequences from dynamic data streams to identify anomalies which could indicate insider threat behaviour. The unsupervised learning techniques provide the basis to define common behaviour. This results in a classifier providing high classification accuracy for data streams containing insider threat uncommon behaviours.

Chen et al [6] also present an anomaly detection system based on unsupervised learning techniques to identify insider threat. The system is using information from recorded access logs and performs statistical analysis to estimate the deviation of users' activities from their communities to predict possible anomalous situations. Their model is applied to six months of access logs from an electronic health record system in a medical centre. Another framework providing support for detecting malicious insider activities is proposed in [11]. In this work, the authors report on a framework which considers information from the cyber-domain but also tries to infer psychological and behavioural factors.

Research work drawing on the attack trees concept prerequisites the existence of attack trees, and mainly focuses on creating risk assessment methodologies for the network. In the anomaly detection stream, there are works that consider unsupervised learning but do not take into consideration the sequence of events as can be described in attack trees. We endeavour to fill this gap by proposing a system which will generate the activity tree automatically, identify the anomalous behaviour and present it in the form of an attack tree.

3 Tree-based analysis

3.1 Attack Trees

Trees have a historical grounding in many applications of Computer Science. From a security viewpoint, Schneier introduced the term *attack trees* that provide a formal approach for defining attack routes. In a traditional attack tree, the attack objective is defined at the root of the tree, and the branches from this define the steps required to achieve this objective. Multiple routes may then exist to achieve the same outcome, and analysts can examine the different routes that could be taken to identify preventative measures for stopping routes from being accessible to attackers. It provides a formal basis for assessing which routes are viable for an attacker to take, and what the full extent of routes to attack are (or at least, as inclusive of all routes as is humanly possible).

Leaf nodes can be attached to root nodes in two different ways, namely the "OR" nodes and the "AND" nodes. Attaching nodes to the root with the OR function, denotes that the goal of this attack (root node) can be achieved in different ways, all of which are independent of each other and only one will suffice to characterise the attack as successful. Linking leaf nodes to the root node with the AND function denotes the different steps which are all required to accomplish the attack. The attack is successful if and only if all the leaf nodes are successful and the sequence that these steps are executed is irrelevant to the goal.

In addition to the dichotomy of the AND and OR functions, attack trees allow attributes to be assigned to leaf nodes. For example a logon node may be assigned the time attribute. The values of the attributes range from a simple boolean value to a continuous variable such as time. Calculating a value for a node requires the calculation of the values of each children. Once these are calculated, the propagation of the values depends on the AND or OR function which connects the children of the node since propagation is calculated differently. For example, if we are interested in calculating the lowest cost with which an attack may be successful, the propagated value for an AND function will be the sum of all the values of the children-leafs, whereas for the value for an OR function will be the lowest value of the children.

The concept of attack trees can easily be extended to describe any sequence of activities that result in an end goal being achieved, and are not limited to only attacks. For example, if an employee wanted to achieve promotion then there may be a number of possible routes that they could take to try and achieve such an end goal. Here, we shall assume that all insiders operate to achieve some end goal that can be defined as to conduct their expected daily workload, achieved by performing their normal sequence of activities during the course of a day (or similarly, from a logon to a logoff).

3.2 Activity Trees

For our system, we use the notion of activity trees for define all behaviour of a user, or in the case of multiple users, a role. This extends on the concept of attack trees, to incorporate not only the sequence of events that would result in an attack, but also the sequence of events that would result in a non-malicious objective. This could be defined as the workload required to complete a particular project, to obtain a promotion, or in a much more general case, it could be the sequence of events that occur in a normal working day.

Figure 1 illustrates the concept of the sequential analysis conducted by the system. Given a user, at the most basic level we can consider two objectives that an employee may aim to achieve: conducting their daily workload as expected (shown in green), or conducting a malicious attack (shown in red). The sequence of events that occur in each branch may either be pre-defined based on what the organisation expects the user to be performing, or what the organisation knows of previously-conducted attacks, or it can be initialised as an empty set which is then populated by the system over time. Typically, an

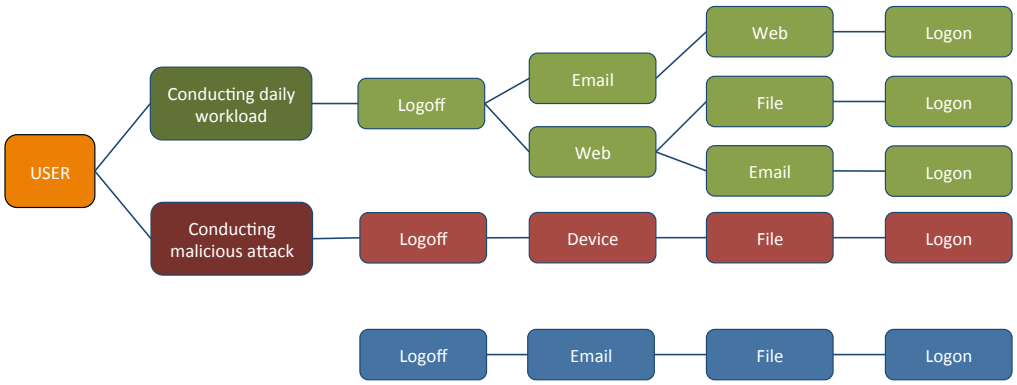


Figure 1: Tree structures for defining expected daily workload and malicious attacks. All users will have a tree that defines their daily workload (green). Some users may also have a tree that defines malicious or suspicious activity that they, or others, have performed (red). The blue tree represents the current observation. The system needs to determine which tree this branch belongs in.

organisation would populate the normal daily workload branches based on known-normal, such as a period of one month where no security incidents were observed.

Each node within the tree represents an activity that has been conducted by the user and observed by a monitoring tool (either automatically or manually). In the example, it can be seen that there are three branches of normal activity, and one branch of malicious activity. In addition to the activity name, each node can also maintain additional information, including the time of day that this activity was conducted, and the attributes associated with this (e.g., filename, e-mail contact, web address, machine used). Once the initial learning period has taken place, the system then treats all new observations as a comparison between the given observation and either the normal working behaviour or malicious behaviour. In the example, a new observation is made (shown in blue). The system needs to be able to consider which branch of activity this chain of events should belong to.

4 Branch similarity

In order for the proposed approach to be effective, we need to consider how a similarity measure can be defined. Given a branch, the system needs to be able to determine the similarity between this branch and the existing tree, so as to classify the observation as being either normal or malicious. In the case where the branch matches exactly with an existing branch, then the tree remains unmodified. Otherwise, it may be that there is a partial match between the observed branch and a particular branch in the existing tree, in which case the system may be able to append the observation and extend that particular branch. It may also be the case that the observed branch does not occur in the tree, however there are some similarities such as time or attribute that are shared. For each branch in the current tree, the system should perform a branch similarity calculation against the newly-observed branch.

As a simple approach, assuming that the current branch being compared and the newly-observed branch are of an equal length, we could compute the number of nodes where there is a mismatch between the two corresponding activities. Of course, sequences may not necessarily be of equal length, so we may wish to treat this as a sliding window against the current branch if the current branch is longer than the newly-observed activity. Extending this further, it would also be beneficial to be able to incorporate whether the values associated with the activity, such as time of day or attributes are equal or not.

Likewise, we may also choose to provide a weighting term to the assessments made on activity, time, and attribute, to further enhance the comparison.

It is clear that different similarity measures will provide different results for the construction of the attack branches. If our similarity measure is too strict, then the tree may result in an overly-complex structure due to the fact that each observation is represented by a unique branch. Similarly, if our similarity measure is too relaxed, then it may become the case that attacks are accidentally classified as normal behaviour. As more complex data sets become available that represent insider-threat activity, we aim to develop the range of similarity measures that could be defined further, based on the more varied range of activities that could be conducted by real human users.

Determining similarity between trees is a fertile area for research that has attracted the interest of many researchers [23] [5] [21]. Cohen *et al.* [7] try to tackle this problem by introducing a class of tree distance functions. They illustrate their algorithm, which considers the distance functions, and present evidence of its performance on experimental cases. White *et al.* [21], present a dynamic structure for similarity indexing, focusing on improving the response time of a system comparing branches. As a similarity measure they use the weighted Euclidean distance metric of nodes. Yang *et al.* [23] focus on computing the tree edit distance which is used to determine the similarity of the branches and describe a novel algorithm which integrates the distance into a filter-and-refine framework. Determining the similarity measure is an open problem and for the needs of this paper we decided to use distance difference of nodes.

5 Experimentation

To demonstrate sequential tree-based profiling, we have developed a proof-of-concept software tool for constructing the tree profile, assessing the similarity of newly-observed branches, and for visualizing the results. The system is written using the Python programming language for the back-end processing, with a web-based front-end based on Javascript and D3 for visualizing the results.

We experiment using the publicly-available test datasets provided by Carnegie Mellon University's Insider Threat programme [1]. The datasets represent a synthetic organisation consisting of 1000 employees, each with a defined job role. The datasets consist of observed activities that employees perform within the organisation, including logging in and out of machines, sending e-mails, accessing files, accessing web sites, and using USB storage devices. In each of their example datasets, a scenario has been crafted and one or more employees exist within the data who perform malicious behaviours at some point during the observed time period (typically a year and a half).

Figure 2 illustrates the activity tree for a single user, during the process of training, consisting of 27 distinct paths that can be taken through the tree. The system constructs a tree as shown here for each user and for each role. All the nodes record the time of the event, along with additional attributes as described below. For a logon, logoff, or USB device node, the attribute is the computer that this event was performed from. For an email node, the attributes are the size of the email, the device from which the email was sent and the recipient of the email. For a web node, the attributes are the website visited and the device from which the website was visited. For a file node, the attributes denote the filename, the device that it was accessed from.

After the training period of one month, the system then begins performing branch similarity on all newly-observed sequences of activities. For the purposes of this case study, we define abnormal behaviour to be where the newly-observed behaviour consists of more than two nodes that differ in terms of activity, or where there is more than one nodes difference and a difference of 8 hours in the observed time period. The testing of the branch similarity is then performed on all data observed after the one month learning period. Should a sequence not reach the required criteria, then the system appends this

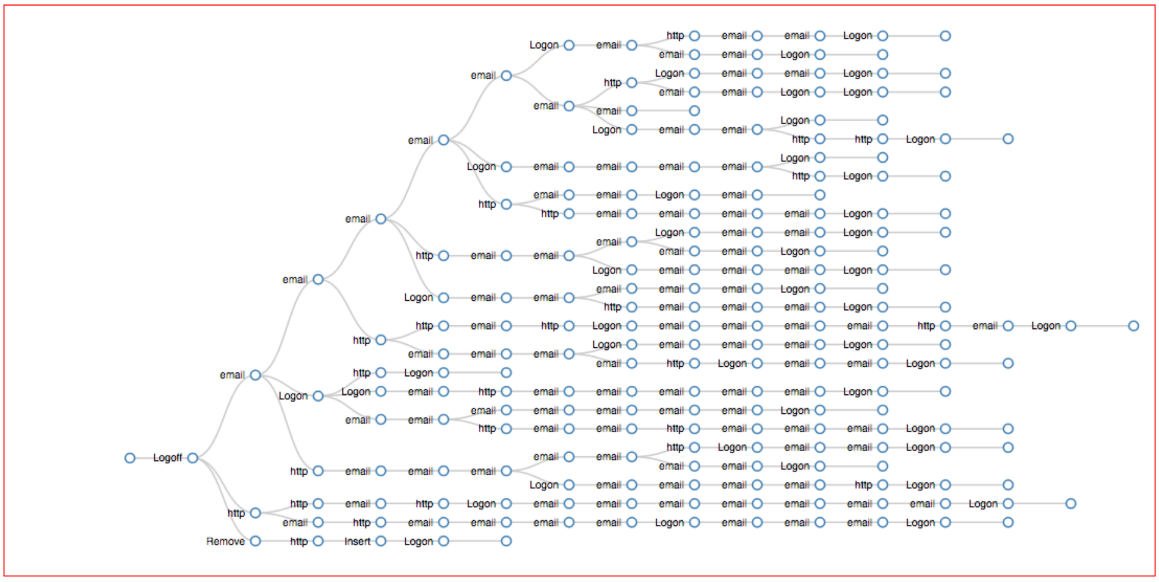


Figure 2: Activity tree for a single user over the period of one month. There exist 27 distinct paths through the tree, where each represents an observed sequence of normal activity. Using the tree-structure, paths can be constructed hierarchically to show commonality in behaviours.

observation to the attack branch.

Figure 3 shows the attacks sequences that were obtained for the malicious user based on our experimentation. Our system is able to identify one insider threat and construct the different ways in which the insider committed malicious activities. Of the 8 paths shown, 7 of these are defined to be malicious (the one that was not is the single Logoff sequence that was performed at an unusual time). In 6 of the branches, the employee has been observed using a USB storage device. Of these branches, there exist similar chains of activities, however these have been observed at different times during the night, hence the distinct branches. When compared against their normal activity, they have never been observed using a USB device before. The results obtained from this study certainly are encouraging, and the sequence of events that an employee is observed making certainly should be deemed as an important assessment towards insider threat detection.

Anomaly detection systems such as those described in Section 2 would have indicated that a user inserting a portable device or uploading data to wikileaks.org webpage renders users' activities malicious. Our approach to tackle the problem of insider threat is complementary to the anomaly detection methodology, since constructing the attack tree may offer useful insight to determine the motivation of the attacker. By having the sequence of events we obtain a richer picture than just recognising the anomalous behaviour. In our case study for example, since the user first stored the data in a portable device and then used this device to upload the data on wikileaks.org webpage, an analyst may be able to draw assumptions on the employee's motives, which could help address how the insider should be dealt with after the detection stage.

6 Conclusion

Insider threats remain a significant problem and a serious concern to the organisations. The insightful knowledge that insiders possess, provides them with opportunities to mask their behaviour, thus raising

Figure 3: Activity tree for the malicious user over the period of a year and a half, showing only the 'attack' branch. The user has been observed logging in during the night and using a USB drive. In one instance, they also access the webpage wikileaks.org which they would not normally do.

the difficulty for any anomalous detection system. In this paper we have proposed a sequential analysis approach for insider threat detection. Based on the notion of attack trees, we extend this to define insider behavioural trees. As new observations of activity sequences are made, the system determines whether the new branch should be appended to the tree that defines normal behaviour, or whether it should be appended to the tree that defines threatening behaviour. Our testing on the commonly-used CMU-CERT datasets appears encouraging at this early stage, and is capable of detecting the malicious insider from the normal users.

The work is currently in its early stages, and described as a proof-of-concept. We aim to extend upon this concept as future work. In particular, we would want to experiment with this approach using a wide variety of insider-threat case scenarios, and to identify the computational complexity of extending the tree-profiling concept when dealing with a wider range of activities and attributes than are available from synthetic data. Finally, we intend to explore the notion of branch similarity in the context of insider-threat detection further. As has been described in this paper, to establish a similarity measure is a challenge in itself, to ensure that the balance is maintained between a concept representation of user and role-based activity, whilst also not losing sight of valuable information. Also, the weighted contribution that should be made by additional factors such as time of day and associated attributes needs to be explored further. It is possible that a number of different models may provide a greater detection capability, where different weighted combinations are deployed. There is much work that can extend on the notion described here, however it is expected that sequential-based techniques such as this could well prove complementary to other anomaly detection systems. By providing a richer picture of insider attack based on the sequence of the events, organisational analysts can have more tools at their disposal for detecting and preventing insider threats.

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] CERT Insider Threat Tools. `http://www.cert.org/insider-threat/tools/`.

[2] Modeling network attacks: Extending the attack tree paradigm.

[3] Matt Bishop, Borislava Simidchieva, Heather Conboy, Huong Phan, Leon Osterwell, Lori Clarke, George Avrunin, and Sean Peisert. Insider threat detection by process analysis. In *Proc. of the 35th IEEE Symposium on Security and Privacy Workshops (SPW), San Jose, USA*, 2014.

[4] Dawn M Cappelli, Andrew P Moore, and Randall F Trzeciak. *The CERT Guide to Insider Threats: How to Prevent, Detect, and Respond to Information Technology Crimes (Theft, Sabotage, Fraud)*. Addison-Wesley, 2012.

[5] Moses S. Charikar. Similarity estimation techniques from rounding algorithms. In *Proc. of the 34th ACM Symposium on Theory of computing (STOC '02), San Diego, CA, USA*, pages 380–388. ACM, 2002.

[6] You Chen and Bradley Malin. Detection of anomalous insiders in collaborative environments via relational analysis of access logs. In *Proc. of the 1st ACM Conference on Data and Application Security and Privacy (CODASPY '11), San Antonio, TX, USA*, pages 63–74. ACM, 2011.

[7] S. Cohen and N. Or. A general algorithm for subtree similarity-search. In *Proc. of the 30th IEEE International Conference on Data Engineering (ICDE '14), Chicago, IL, USA*, pages 928–939, March 2014.

[8] Rinku Dewri, Indrajit Ray, Nayot Poolsappasit, and Darrell Whitley. Optimal security hardening on attack tree models of networks: a cost-benefit analysis. *International Journal of Information Security*, 11(3):167–188, 2012.

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

[10] Gonzalo Flórez-Puga, Marco Gomez-Martin, Belen Diaz-Agudo, and Pedro Gonzalez-Calero. Dynamic expansion of behaviour trees. In *Proc. of the 4th AIII conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE '08)*, pages 36–41, 2008.

[11] L Frank and Ryan E Hohimer. Modeling human behavior to anticipate insider attacks. *Journal of Strategic Security*, 4(2):3, 2011.

[12] Chong-U Lim, Robin Baumgarten, and Simon Colton. Evolving behaviour trees for the commercial game defcon. In *Applications of Evolutionary Computation*, pages 100–110. Springer, 2010.

[13] Andrew P Moore, Robert J Ellison, and Richard C Linger. Attack modeling for information security and survivability. Technical report, DTIC Document, 2001.

[14] P. Parveen and Bhavani Thuraisingham. Unsupervised incremental sequence learning for insider threat detection. In *Proc. of the 9th IEEE International Conference on Intelligence and Security Informatics (ISI '12), Beijing, China*, pages 141–143, June 2012.

[15] Pallabi Parveen, Nate McDaniel, Varun S Hariharan, Bhavani Thuraisingham, and Latifur Khan. Unsupervised ensemble based learning for insider threat detection. In *Proc. of the 4th IEEE Privacy, Security, Risk and Trust (PASSAT '12), Amsterdam, Netherlands*, pages 718–727. IEEE, 2012.

[16] Nayot Poolsappasit, Rinku Dewri, and Indrajit Ray. Dynamic security risk management using bayesian attack graphs. *IEEE Transactions on Dependable and Secure Computing*, 9(1):61–74, 2012.

[17] Indrajit Ray and Nayot Poolsapassit. Using attack trees to identify malicious attacks from authorized insiders. In *Proc. of the 10th European conference on Research in Computer Security (ESORICS'05), Milan, Italy, LNCS*, volume 3679, pages 231–246. Springer Berlin Heidelberg, 2005.

[18] Bruce Schneier. Attack trees: Modeling security threats. *Dr. Dobb's Journal*, December 1999.

[19] E Ted, Henry G Goldberg, Alex Memory, William T Young, Brad Rees, Robert Pierce, Daniel Huang, Matthew Reardon, David A Bader, Edmond Chow, et al. Detecting insider threats in a real corporate database of computer usage activity. In *Proc. of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '13)*, pages 1393–1401. ACM, 2013.

[20] Lingyu Wang, Anyi Liu, and Sushil Jajodia. Using attack graphs for correlating, hypothesizing, and predicting intrusion alerts. *Computer communications*, 29(15):2917–2933, 2006.

[21] David A White and Ramesh Jain. Similarity indexing with the ss-tree. In *Proc. of the 12th IEEE International Conference on Data Engineering, New Orleans, LA, USA*, pages 516–523. IEEE, 1996.

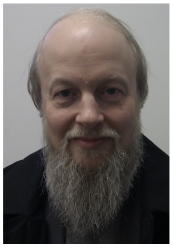
[22] Peng Xie, Jason H Li, Xinming Ou, Peng Liu, and Renato Levy. Using bayesian networks for cyber security analysis. In *Proc. of the 40th IEEE / IFIP International Conference on Dependable Systems and Networks (DSN '10)*, pages 211–220. IEEE, 2010.

[23] Rui Yang, Panos Kalnis, and Anthony K. H. Tung. Similarity evaluation on tree-structured data. In *Proc. of the 31st ACM International Conference on Management of Data and Symposium on Principles Database and Systems (SIGMOD '05), Baltimore, MD, USA*, pages 754–765. ACM, 2005.

Author Biography

Ioannis Agrafiotis is a post-doctoral researcher at the Cyber Security Centre, in the Department of Computer Science at the University of Oxford. Currently, Ioannis is working on the Corporate Insider Threat Detection project to analyse behavioural indicators and common attack steps taken in insider threat scenarios. Previous, he worked on CyberVis, a Dstl funded project, aiming to create a tool to increase situational awareness in CERT environments. He has also worked on the TSB and EPSRC funded TEASE project, researching how people understand information quality and provenance, and looking to develop mechanisms that effectively communicate trustworthiness. Prior to his post-doctoral experience, Ioannis completed a PhD in Engineering at the Warwick Manufacturing Centre (WMG), University of Warwick, focusing on formal methods for information privacy. During his doctoral studies Ioannis worked on the EnCoRe project through which he received an EPSRC studentship. He also holds an MSc in Analysis, Design and Management of Information Systems from the London School of Economics and Political Science in the UK and a BSc in Applied Informatics from the University of Macedonia in Greece.

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



Professor 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 Oxford's 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 Warwick's 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.