Risk Modelling of Access Control Policies with Human-Behavioural Factors

Simon Parkin
School of Computing Science
Newcastle University
Newcastle-upon-Tyne, UK
s.e.parkin@ncl.ac.uk

Aad van Moorsel
School of Computing Science
Newcastle University
Newcastle-upon-Tyne, UK
aad.vanmoorsel@ncl.ac.uk

I. Introduction

Organisations can gain competitive advantage by taking risks within their market. An organisation may promote a particular approach to business opportunities within its employees.

Increasingly organisations within a “knowledge-based economy” trade in information assets. A simple example may be an employee travelling to a potential client’s premises to present details of their organisation’s work. Here the asset is the work being presented, which has value to the presenting party. A possible benefit is that the presented work influences the potential client to enter into a business partnership.

There are also risks in the previous example that may equally result in losses for the presenting party. The details of the presented work may be lost or stolen in transit, or retained by the potential client against the wishes of the presenting party. It may even be that the individual(s) presenting the work have malicious intentions of their own which are then satisfied once they have the organisation’s information assets in their possession.

An organisation will seek to permit some activities – and forbid others – as part of its risk approach. Senior management will often have a sense of what should and should not be done with the organisation’s information assets. These commands may then be communicated to the information security manager e.g. the Chief Information Security Officer (CISO), or whoever is responsible for managing the security of the organisation’s information assets. The information security manager (or their staff) must then translate the risk approach into security controls within the organisation’s information security infrastructure.

Permitted and prohibited actions are most readily communicated across an organisation through policies. These may be defined in natural language. Different parts of the information security software/hardware infrastructure may however be governed by machine-readable security policies or configurations.

There is a need to provide tool support for security policy managers to allow them to configure machine-readable information security policies, and at once be provided with a quantifiable measure of the risks and benefits that arise from specific policy properties. Although such tools already exist (e.g. [11]), risks are often driven by employees, and so provision of a perspective on human behaviour in the workplace would also be beneficial. This would then allow for fine-tuned, quantifiable assessment of the risks that employee behaviour presents to an organisation’s information assets within the context of the information security infrastructure and its configuration. As far as we are aware no such tool support currently exists.

Here we provide a software tool that analyses access-control policies, and provides quantifiable feedback of potential behaviour-oriented benefits and risks that policy properties create for an organisation and its information assets. The intention is to demonstrate that changes to an organisation’s software-level information security policies can be directly reflected in a behaviour-oriented risk assessment model. This then provides the capacity to consider human behaviour within an established and widely-understood information security process.

We focus specifically on the use of removable USB storage devices (e.g. USB memory sticks) by employees and how access permissions for these devices can be expressed in the eXtensible Access Control Markup Language (XACML) [5] access-control language. Using this example we illustrate how a specialised, executable risk model can be used to provide a risk assessment of the permitted and prohibited employee activities defined within a machine-readable access-control policy.

II. IMPLEMENTATION

To achieve our goal we chose a software-based USB risk model and an information security policy definition tool, and created logic to bring these two elements together. This logic analyses policies to identify those employee activities that may influence the security of information assets stored on a removable USB storage device. These qualities are then encoded as configuration properties for the USB risk model.

The authors are supported in part by EPSRC grant EP/G011389/1 (“AMPS: Analysis of Massively Parallel Stochastic Systems”) and UK Technology Strategy Board (TSB), grant nr. P0007E (“Trust Economics”).

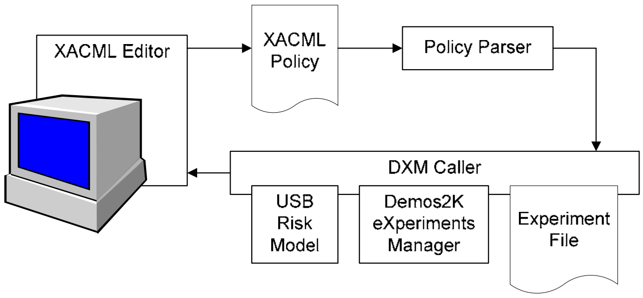


Figure 1. Overview of policy risk modelling system

An overview of the system that was produced is illustrated in Figure 1. The components of the system are as follows:

- *XACML Editor:* allows creation and validation of XACML access control policies.
- *XACML Policy:* the machine-readable access-control policy created by the XACML Editor. This may be used by the information security infrastructure to manage machine access to information assets.
- *Policy Parser:* examines the XACML Policy for a pre-configured set of access control characteristics, to be used as parameters in the risk modelling process.
- *Experiment File:* the parameters identified by the Policy Parser are written to a configuration file, which is then used to calibrate the modelling process.
- *Demos2K eXperiments Manager (DXM):* The Demos2K eXperiments Manager (DXM) [6] is a tool for managing instances of Demos2K [7] models (in this case a USB risk model). The DXM takes its configuration from the Experiment File.
- *USB Risk Model:* the USB Risk Model [2, 3] is a Demos2K model that models USB device usage by organisation employees. The model properties are based in part upon elements of the XACML Policy examined by the Policy Parser.
- *DXM Caller:* manages Experiment Files and instances of the DXM and USB Risk Model. The DXM Caller takes the output parameters of the USB Risk Model from the DXM and returns them to the XACML Editor.

A. USB Risk Model

There may be benefits to allowing employees to use removable storage devices but there are also inherent risks. Ideally organisations will have the capability to assess these benefits and risks, and use the results to inform their risk management strategy.

The USB Risk Model [2, 3] models the actions that an employee or those in close proximity to the employee may enact upon a USB storage device projected over a pre-configured time period.

The registered owner of a device is modelled as moving between a fixed set of location types: their workstation; their home (perhaps as part of a ‘teleworking’ program); a conference; a client’s premises, or; in transit between any of these locations.

The device owner can perform any of a set of actions upon the device during the duration of the model, these being: write data; read data; delete data, or; wipe device contents.

The behaviour of individuals other than the device owner is also modeled. The USB Risk Model considers that work colleagues or malicious parties internal or external to the organisation (referred to as ‘traitor’ and ‘foe’ respectively) may also gain access to the device and its contents (either accidentally or deliberately).

In this work the actions of an employee are assumed to be dictated by a company policy. There is also an assumption here that the permitted and prohibited actions in any policy that is analysed apply to everyone in the organisation, and are enforced at all times.

Here we consider how to make the USB Risk Model directly applicable and accessible within a real organisational setting. Here we exploit the fact that the model can be configured to represent qualities of an organisation’s risk strategy. The model has an execution time that makes it responsive enough for environments where policy changes must be promptly communicated to various stakeholders.

By feeding qualities of an organisation’s access control policies into the USB Risk Model, we are able to calculate the projected risks and benefits of organisation-specific USB access control policies and provide direct feedback as to their suitability.

B. XACML

XACML is an access control policy specification language that is widely used to define access control schemes for resources distributed across a managed network. In this work we use XACML policies to specify access control rules for local access to removable media devices (as also seen in e.g., Nextlabs Enterprise DLP [1]).

By associating specific properties of an XACML policy with an instance of the USB Risk Model, we show that changes in an organisation’s security configuration can be analysed to produce a measure of the inherent risks and benefits that result from these changes.

C. XACML Editor

The University of Murcia (UMU) created the UMU XACML Editor [4] to provide support for the creation and validation of XACML access control policy files.

We altered the XACML Editor to include a ‘Modelling’ menu to allow execution of the USB Risk Model for whichever policy is open within the editor.

D. Policy Parser

The Policy Parser uses the Enterprise Java XACML API [8] to create Java objects that represent the various components of an XACML policy (e.g. Rules, Obligations, etc.). Logic within the Policy Parser examines these components for specific properties that relate to the configuration of the USB Risk Model. Note that no specialised elements or annotations need to be added to an XACML policy to enable this process.

The XACML properties that are examined are described as follows:

- *Action Permissions:* the actions modeled in the USB Risk Model are assumed to be explicitly represented within two distinct Targets in an XACML Policy – the set of actions that the policy will ‘Permit’, and the set of actions that it will ‘Deny’.
- *Duration of Usage:* the USB Risk Model is configured to model the projected risks and benefits of USB device usage over a specified length of time (1 year by default). Furthermore, the amount of time within a working day when an individual can use their USB device can also be configured, based upon the time restrictions stipulated in an XACML Policy for ‘Permitted’ activities.
- *Encryption Policy:* an organisation’s encryption policy for USB devices can also be modelled. An XACML Policy is examined for an Obligation identifier which indicates whether USB device encryption is mandated.

Once the Policy Parser has examined an XACML Policy, it constructs parameter-name/value pairs which are then sent to an instance of the DXM Caller, which manages configuration and execution of the USB Risk Model.

A number of assumptions are made to simplify use of the Policy Parser and configuration of the USB Risk Model:

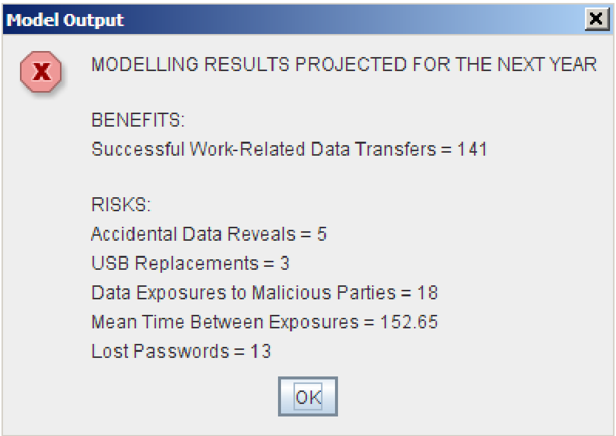
- The USB Risk Model assumes that each individual uses only one USB storage device. However for simplicity we also assume that only one device drive will ever be used to connect a USB device to a computer, and that for all computers used by an individual the drive has the same, generic label;
- The XACML policy refers to the access permissions of 'everybody', since the USB Risk Model models device usage for a single, unclassified individual;
- If encryption is stated as an Obligation, that this refers to all data that is written to or read from a USB device;
- If encryption is required that it cannot be circumvented, and that it applies in all modeled locations.

E. DXM Caller

The DXM Caller creates the experiment file that the DXM uses to manage a USB Risk Model instance. The DXM Caller populates the experiment file with the configuration properties obtained by the Policy Parser.

With the DXM configuration file built, the DXM Caller creates an instance of the DXM. The DXM then runs the USB Risk Model and produces output files documenting intermediate and final values of state variables as obtained from the model.

The DXM Caller examines these output files for the final values of selected risk measurement metrics (note that this is only a subset of the results produced by the USB Risk Model). These results are then extracted and associated with meaningful identifiers (for readability), and presented within the modified XACML Editor.



MODELLING RESULTS PROJECTED FOR THE NEXT YEAR

BENEFITS:
Successful Work-Related Data Transfers = 141

RISKS:
Accidental Data Reveals = 5
USB Replacements = 3
Data Exposures to Malicious Parties = 18
Mean Time Between Exposures = 152.65
Lost Passwords = 13

Figure 2. output of the policy risk modelling process

As shown in Figure 2, the following parameters are retrieved from each instance of the USB Risk Model:

- *Successful Work-Related Data Transfers:* each permitted transfer of data (e.g. a file) from the USB device is seen as benefiting the organisation;
- *Accidental Data Reveals:* the number of occasions when the contents of the storage device are accidentally revealed to or left open to access by co-workers. Whether co-workers have malicious intentions or not, they are nonetheless not meant to see the contents of another individual’s USB device;
- *USB Replacements:* number of occasions when the USB storage device has to be replaced (at a cost), based upon modelled employee behaviour;
- *Data Exposures:* an estimate of the number of times that a malicious party either within or outside the organisation will have an opportunity to read the contents of a USB storage device over the simulated lifetime of the model;
- *Mean Time Between Exposures:* the average time between data exposures, measured in hours;
- *Lost Passwords:* the number of times when an individual forgets the password used to authenticate access to encrypted device contents. ‘Losing’ the password in this way will result in the individual having no access to files stored on the device.

The metrics modelled within the USB Risk Model were derived through consultation with a senior information security officer in an industrial organisation. With this it is perceived that a security officer using the modified XACML Editor will be able to analyse the results of the USB Risk Model and relate them to measures of risk that they are familiar with. It is assumed that the user is able to compare the results to a pre-defined, quantified risk management strategy, or otherwise use the results as evidence for information security management decisions.

III. Discussion

With this work we have demonstrated that an information security policy infrastructure can be augmented to provide real-time behaviour-oriented risk-modelling feedback. This does however raise a number of points, as shall be discussed here.

Firstly, there is an assumption that the security officer responsible for managing a machine-readable security policy understands the organisation's risk approach. The security officer is expected to be able to translate between natural-language policy directives and machine-read security policy content [9]. This task is however time-consuming and error-prone [10].

It is also assumed that the member of staff responsible for maintaining machine-read policies can act autonomously to change policies and in turn contribute to the organisation's risk approach.

With the previous assumptions, our tool essentially requires users to have technical-level skills relating to security management and to be able to interpret the organisation's risk approach. This is perhaps more likely to hold in smaller organisations than in larger enterprises. In the former, individual employees are typically required to 'blend' roles and develop cross-disciplinary abilities. In the latter individuals may specialise and may have to work within more rigorous and formalised procedures, where for instance technical-level IT officers are not expected to contribute to policy decisions. However, smaller and smaller organisations are less likely to deploy deeply-entrenched, organisation-wide and automated (i.e. policy-driven) security controls.

To make the tool more widely applicable it may be necessary to repackage its capabilities. This could involve abstracting technical-level properties to make it applicable to managers (as discussed in e.g. [9]), or feeding in pre-prepared, machine-readable risk properties so that a technical-level security officer can simply modify the policy until the tool deems the results satisfactory, without the officer necessarily having to interpret them.

IV. Conclusion

It would be useful to allow information security managers (and other security policy managers within an organisation) to consider how employee behaviour affects the security of information assets. We have provided a supporting software tool that analyses access-control policies and produces metrics representing the benefits and risks of these policies concerning use of USB devices by individuals in an organisation.

Our tool demonstrates that changes to an organisation's software-level security policies can be represented in a behaviour-oriented risk assessment model positioned within an established information security process.

There is potential for the concept of extracting configuration properties from existing information security mechanisms and applying them to behaviour-oriented risk models to be applied to scenarios beyond USB device usage.

ACKNOWLEDGMENTS

We are grateful to Robert Coles (Merrill Lynch Europe Limited), and Brian Monahan and Jonathan Griffin (HP Labs Bristol) for their contributions to this work.

REFERENCES

[1] Nextlabs, "Enterprise DLP 4.0", http://www.nextlabs.com/html/?q=enterprise-dlp, last viewed 11/05/09

[2] A. Beautement, R. Coles, J. Griffin, B. Monahan, D. Pym, M.A. Sasse, M. Wonham, "Modelling the Human and Technological Costs and Benefits of USB Memory Stick Security", Workshop on Economics in Information Security (WEIS), 2008

[3] R. Coles, J. Griffin, H. Johnson, B. Monahan, S.E. Parkin, D. Pym, M.A. Sasse, A. van Moorsel, "Trust Economics Feasibility Study", In 38th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2008), IEEE Computer Society, pp A45-A50, 2008

[4] University of Murcia (UMU), "UMU-XACML-Editor", http://xacml.dif.um.es/, last viewed 11/05/09

[5] Organization for the Advancement of Structured Information Standards (OASIS), "eXtensible Access Control Markup Language", http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml, last viewed 13/05/09

[6] B. Monahan, "DXM – Demos2k eXperiments Manager", HPL-2008-173, HP Laboratories, 2008

[7] A. Christodolou, R. Taylor, and C. Tofts, "Demos 2000", http://www.demos2k.org, 2000

[8] Z. Wang, "Enterprise Java XACML", http://code.google.com/p/enterprise-java-xacml/, last viewed 13/05/09

[9] V. Tsoumas & T. Tryfonas, "From risk analysis to effective security management: towards an automated approach", Information Management & Computer Security, Vol. 12 No. 1, pp 91-101, 2004

[10] U. Lang & R. Schreiner, "Model Driven Security Management: Making Security Management Manageable in Complex Distributed Systems", Modeling Security Workshop '08, 2008

[11] A. Ekelhart, S. Fenz, T. Neubauer, "AURUM: A Framework for Information Security Risk Management", Proceedings of the 42nd Hawaii International Conference on System Sciences (HICSS), 2009