Formal Modeling and Verification of Opportunity-enabled Risk Management

Aldini, Alessandro; Seigneur, Jean-Marc; Ballester Lafuente, Carlos; Titi, Xavier; Guislain, Jonathan

Formal Modeling and Verification of Opportunity-enabled Risk Management

A. Aldini
Dipartimento di Scienze di Base e Fondamenti
University of Urbino
Urbino, Italy

J.-M. Seigneur, C. Ballester Lafuente, X. Titi, J. Guislain
Centre Universitaire d'Informatique
University of Geneva
Carouge, Switzerland

***Abstract*—With the advent of the Bring-Your-Own-Device (BYOD) trend, mobile work is achieving a widespread diffusion that challenges the traditional view of security standard and risk management. A recently proposed model, called opportunity-enabled risk management (OPPRIM), aims at balancing the analysis of the major threats that arise in the BYOD setting with the analysis of the potential increased opportunities emerging in such an environment, by combining mechanisms of risk estimation with trust and threat metrics. Firstly, this paper provides a logic-based formalization of the policy and metric specification paradigm of OPPRIM. Secondly, we verify the OPPRIM model with respect to the socio-economic perspective. More precisely, this is validated formally by employing tool-supported quantitative model checking techniques.**

***Keywords*-BYOD; opportunity analysis; risk management; model checking;**

I. Introduction

Nowadays, the Bring-Your-Own-Device (BYOD) and mobile work trend are spreading in corporate environments. On one hand, remote access to sensitive information and company services represents a fundamental resource to increase the opportunities that the flexible work model of the BYOD paradigm brings. On the other hand, such a paradigm introduces new and major risks of security incidents, because the predefined security perimeter of the company is weakened and may be circumvented. In fact, the classical closed corporate environment is characterized by situations in which every agent involved (either human being or software/hardware device) is implicitly trustworthy and no real-time adjustments are actually needed. However, mobile corporate users are increasing the use of alternative environments, like, e.g., private homes, airports, trains, customer's offices, and conference centers. At the same time, they use a wide range of different devices, like, e.g., personal laptops, smartphones, tablets, public and customer's desktops. Sometimes, they also need to interact with external collaborators and contractors. Hence, this extremely dynamic context contributes to diminish the level of control that the company has on the environment. New means to dynamically assess in real time access control involving the use of corporate assets are required that consider trust (towards the user and the environment in which the user issues access requests), potential threats, and opportunities.

Securing the emerging communication and working models based on the BYOD paradigm requires the definition of ad-hoc policies [1]. While the complexity of the threats emerging in such a scenario is increasing and requires a reinforcement of the risk policies, the major opportunities that arise in the BYOD model cannot be underestimated. Such a tradeoff can be evaluated correctly only if the costs of incidents caused by threats and the benefits deriving from opportunities are both treated as first-class citizens in the analysis of risk.

Historically, risk is defined as the combination of the probability that a given threat successfully exploits vulnerabilities of an asset with the cost of the negative consequences to the owner balanced with the benefit of the positive consequences of an available opportunity. Unfortunately, in standard threat methodologies [2], while typically the variables involved are mainly concerned with the consequences of threats, the potential opportunity benefits are not taken into account as a major aspect. In this respect, an asset is anything that has value to the owner, either tangible (hardware, software, infrastructures) or intangible (knowledge, trust relations, reputation), while a vulnerability is a weakness of an asset that can be exploited by a threat. Threats and opportunities are actions or events with the potential to cause negative and positive outcomes, respectively. Hence, risk relies on several – very often complex and hard to evaluate – aspects that require careful analysis intended to establish risk context and criteria, identification and estimation of risk, communication and treatment of risk. As emphasized above, in spite of the fact that there may be positive consequences of opportunities that are taken, risk management tends to focus on the negative outcomes and on the negative events. Among these aspects, a prominent one is given by the threat modeling, which can be asset-driven, attacker-driven, and design-driven [3].

In this paper, the assets are related to corporate assets that are requested by users in order to carry out their work. In this setting, we concentrate on the definition of a decision making process that, based on the threat and opportunity

modeling, is carried out to decide whether a requesting user shall be granted access or not to the assets needed to complete a given work. In particular, recently a new model called opportunity-enabled risk management (OPPRIM), has been proposed that offers a real-time framework in which analysis of the threats arising in the BYOD setting and analysis of the opportunities emerging in such an environment are balanced [4]. This is done by combining mechanisms of risk estimation with trust and threat metrics.

The contribution of this paper is to extend previous work as follows. Firstly, we provide a logic-based formal definition of the OPPRIM modeling framework. Secondly, we perform a socio-economic analysis related to the use of the OPPRIM system. More precisely, the verification is conducted on a formal representation of the system by employing automated quantitative model checking techniques.

The rest of the paper is organized as follows. In Section II, we discuss related work on traditional risk management methodologies and to which extent they deal with both risk and opportunities. Section III presents the design and architecture of the opportunity-enabled risk management system. Then, a formal specification framework is defined for the description of risk policies as well as threat and trust based metrics. In Section IV, we show the results of the formal verification of the OPPRIM model, which is described and analyzed through the model checker PRISM [5]–[7]. Section V concludes the paper and discusses future work.

II. Related Work

ISO 27005 [8] (information security risk management) and ISO 31000 [9] (risk management standard) define the peculiarities of risk management. The ISO/IEC Guide 73 [10] specifies the risk management vocabulary and the risk estimation process, which assigns values to the probability and consequences of risk, based on, e.g., costs, benefits, the concerns of stakeholders, and other variables.

Several different methodologies have been proposed in the literature to implement risk management. For instance, TROPOS provides a goal-risk framework for the analysis of risk and necessary countermeasures [11], [12], while the CORAS approach to model-driven risk analysis encompasses techniques and guidelines for the treatment of risk in changing systems where system and environment evolve dynamically [13]. However, typically opportunities are not taken into account, especially in the field of access control.

Han et al. [14] show that it is necessary to evaluate all the four combinations deriving from the risk/benefit of granting/denying access. In particular, they consider intentional (i.e., explicitly known a-priori) benefits only, while they do not estimate inadvertent benefits, which, instead, are explicitly modeled as opportunity descriptors in our work. The Benefit and Risk Access Control (BARAC) model [15] defines risk and benefit as two vectors used to make the decision by balancing risks of information disclosure with benefits of information sharing. The model we propose is not only focused on information sharing and covers more aspects of risk management. Baracaldo and Joshi [16] extend the role-based access control model with risk and trust management in order to prevent insider threats that are caused by internal, legitimate users. They evaluate all the risk parameters statically, while in our model risk is estimated dynamically in real-time at request time. Shaikh et al. [17] emphasize that a flexible risk-based access control system has to take into account user's trustworthiness, which is calculated on the basis of the history of the outcomes deriving from the access requests granted to the user. More in general, it is well recognized that trust [18], [19] and risk represent, combined together, a critical factor in the decision-making system [20]–[22]. The method proposed in this work encompasses such a philosophy, since trust is considered and used as part of the risk analysis.

III. Opportunity-enabled risk management

In this section, we first present the architecture of the opportunity-enabled risk management system and illustrate its decision-making process. As presented afterwards, this process is based on the application of risk policies that are specified through a formal language and on the real-time computation of threat and trust-based metrics. Two real examples accompany the presentation to exemplify the application field of such a methodology.

A. OPPRIM design and architecture

The OPPRIM model bases the decision-making procedure on two main notions: threats (inducing costs) and opportunities (enabling benefits). An opportunity is an event potentially inducing positive outcomes quantified by a certain benefit. A threat is characterized by the set of pieces of evidence enabling an event that may potentially cause negative outcomes associated with certain costs. Such pieces are called clues and, e.g., could correspond to the version of the antivirus that is running on the device, the specific Wi-Fi connection used by the device, the trustworthiness of the user and of the device security state, and so on. Typically, the cost associated with a threat corresponds to the value of the assets involved, which would be compromised in the case the threat actually materializes as a security incident.

Example 1: Consider a salesman attending a conference, where he meets a potential client for his web services company. To take the chance of a fruitful agreement with a possibly immediate benefit, the salesman, who left his laptop at the hotel, requires remote access to a company server through his smartphone. This opens the door to privacy breaches as the salesman does not properly maintain up-to-date the security configuration of the smartphone. On the other hand, the new client represents another potential negative clue, as the salesman never met him before. In this scenario, the opportunity described leads to a potential

benefit (the value of the contract) and triggers a potential threat whose cost is given by the value of the asset accessed remotely and the value of the web services purchased.

Example 2: Consider a mobile worker waiting at the airport with her personal laptop. An opportunity to work online is given by a wireless connection through which the worker may access sensitive information stored in the company database. In this scenario, the opportunity leads to the benefit represented by the amount of hours the worker can be productive. Moreover, as an additional benefit, the worker would have the possibility of respecting a hard deadline associated with an important project. On the other hand, we have a possibly negative clue related to the security state of the computing environment, which could cause a loss of sensitive information. Another clue is represented by the identity of the worker and, therefore, her trustworthiness as perceived by the company. Indeed, an untrusted worker may require a risky access to the company database and then spend the subsequent hours without taking the opportunity, e.g., by surfing the web.

The OPPRIM architecture is made out of two interacting components, the OPPRIM mobile app running on the user device and the Real-Time Risk & Trust Analysis Engine (RT2AE) running on the company remote server. The operations between these two parties are exemplified in Figure 1. In a preliminary phase, we assume that RT2AE is configured, by specifying, e.g., the risk policy, the domain of potential assets and opportunities, and the corresponding values. Whenever the user issues a new request, set of requested assets and set of current detected clues are communicated to RT2AE, which then infers related opportunity benefit and threat. All these ingredients are then combined together with other quantitative parameters, like, e.g., the user and/or device trustworthiness and the probability associated with opportunity benefit and threat, in order to verify the satisfaction of the risk policy. In general, some of the involved variables are manually configured, while others, as we will see, are managed automatically. Based on the retrieved information and on the policy evaluation, RT2AE computes one of the following decisions:

- *granted*: the access to the asset is enabled.
- *deny*: the access to the asset is denied and no risk treatment is possible to change the decision.
- *maybe*: the access to the asset can be enabled provided that some risk treatment is applied (e.g., by changing the Wi-Fi connection).
- *on your own risk*: the user is invited to observe and understand that a potential risk is concrete, and to choose whether an exception shall be added to access the asset.

Notice that whenever the choice is finally left to the user, as in the last case, then any consequence of the user behavior may have an impact upon the user trustworthiness, which in turn could represent a clue that characterizes a threat impairing future access requests.

Finally, it is worth observing that the feedback resulting from the consequences of the decision implemented, like, e.g., the fact that an opportunity is taken with success or that a threat actually causes a security incident, is used to adjust the configuration parameters.

B. OPPRIM formal model

In this section we formalize the language for expressing risk policies based on the OPPRIM model. We start by introducing the parameters that may occur in the specification of the policy.

Let $\mathcal{O}$ be the domain of opportunities, ranged over by $o, o', \ldots$, $\mathcal{T}$ be the domain of threats, ranged over by $t, t', \ldots$, and $\mathcal{C}$ be the set of all possible clues, ranged over by $c_1, c_2, \ldots$. A clue is any kind of pieces of evidence of the presence of a threat that can be sensed by the mobile application running the OPPRIM protocol. Hence, a threat is actually identified by the set $C \subseteq \mathcal{C}$ of clues revealed in the specific scenario.

Example 3: On one hand, in the salesman example we have a threat identified by two clues, which are determined by the networking environment and the trustworthiness towards the potential client, respectively.

On the other hand, the threat t associated with the scenario of the mobile worker example is identified by the following clues. The OPPRIM app reveals that the user is running an up-to-date anti-virus on her mobile device (clue c_1) and a browser with strong security settings (clue c_2), while the mobile device is connected to a known public free Wi-Fi hotspot (clue c_3) and uses a virtual private network encrypting traffic to and from the VPN server (clue c_4). The company also retrieves and uses information about the past behavior of the mobile worker to establish her trustworthiness (clue c_5). Therefore, $t = \{c_1, c_2, c_3, c_4, c_5\}$.

Formally, every opportunity $o \in \mathcal{O}$:

- leads with probability $p_o \in [0,1]$ to $m_o \geq 1$ outcomes, each one associated with a benefit $\alpha_i \cdot b_{o_i}$, with $1 \leq i \leq m_o$, $\alpha_i \in [0,1]$, and $b_{o_i} \in \mathbb{R}$.
- opens the door to $x_o \geq 0$ threats, such that if $x_o = 0$ then no threats are possible, otherwise every threat t_j, with $1 \leq j \leq x_o$, leads with probability $p_{t_j} \in [0,1]$ to $n_{t_j} \geq 1$ outcomes, each one associated with a cost $\beta_k \cdot c_{t_{j_k}}$, with $1 \leq k \leq n_{t_j}$, $\beta_k \in [0,1]$, and $c_{t_{j_k}} \in \mathbb{R}$.

Then, the formula expressing the potential balance resulting from the combination of benefits and threats associated with an opportunity $o \in \mathcal{O}$ is as follows:

$$bal_o = \sum_{i=1}^{m_o} \alpha_i \cdot b_{o_i} - \sum_{j=1}^{x_o} \sum_{k=1}^{n_{t_j}} \beta_k \cdot c_{t_{j_k}} \qquad (1)$$

Several variants of (1) can be derived depending on the specific scenario. For instance, if the opportunity $o \in \mathcal{O}$

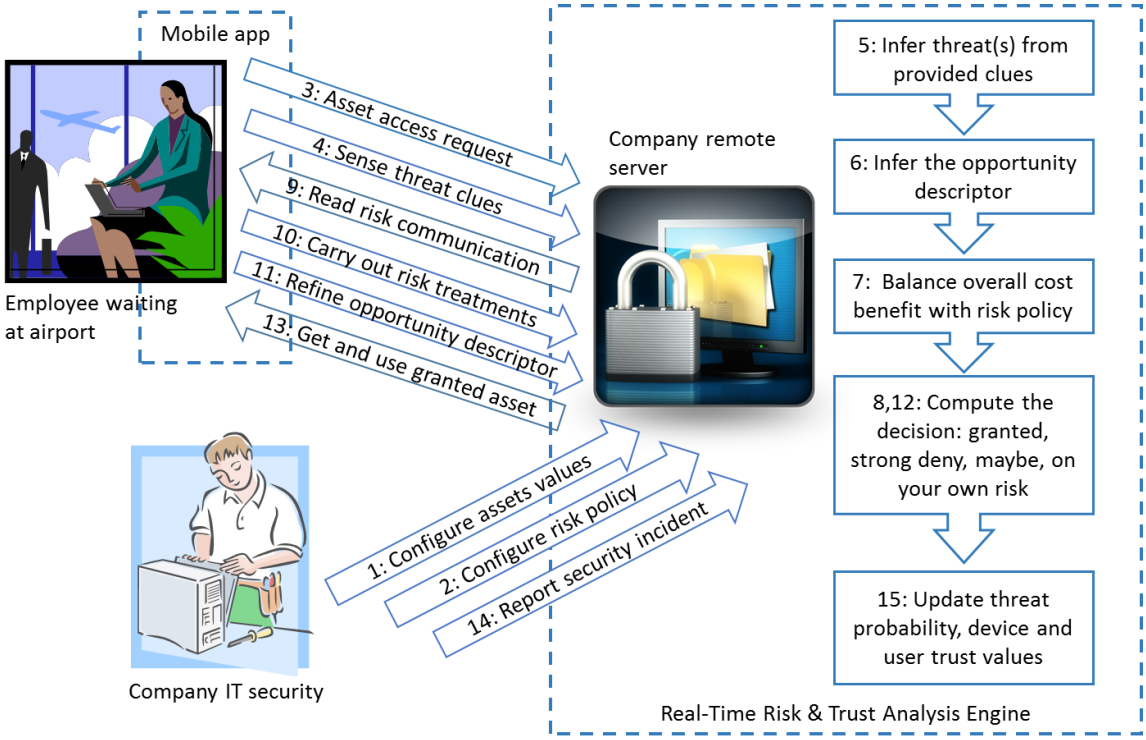


Figure 1: OPPRIM high-level view.

is taken, then we have the two limiting balances obtained in the best case and in the worst case, respectively:

$$\begin{array}{rcl} bcbal_o & = & \sum_{i=1}^{m_o} \alpha_i \cdot b_{o_i} \\ wcbal_o & = & -\sum_{j=1}^{x_o} \sum_{k=1}^{n_{t_j}} \beta_k \cdot c_{t_{j_k}} \end{array} \quad (2)$$

In particular, $bcbal_o$ expresses the maximum benefit obtained in the absence of any security incident, while $wcbal_o$ expresses the maximum damage, which is obtained whenever all the potential threats cause security incidents, while the opportunity does not actually lead to the promised benefits. On the other hand, if the opportunity is not taken, in the best case the balance is null, but in the worst case the opportunity benefit is lost forever and the threats occur in any case for other reasons:

$$\overline{wcbal}_o = -\sum_{i=1}^{m_o} \alpha_i \cdot b_{o_i} - \sum_{j=1}^{x_o} \sum_{k=1}^{n_{t_j}} \beta_k \cdot c_{t_{j_k}} \quad (3)$$

Then, the variables that may occur in the specification of a policy are of two types. Firstly, we have probabilistic parameters, which are associated to opportunities and threats. Secondly, we have balance parameters, which derive from (1) and related variants. Informally, an atomic predicate occurring in the specification of a policy is a boolean comparison expression including a variable and a constant as operands and a classical comparison operator. Hence, the policy specification language derives from a logical combination of atomic predicates, the satisfaction of which guides the decision making process.

From the syntax standpoint, the set $\mathcal{P}$ of policies is generated through the following grammar:

$$\begin{array}{rcl} P & ::= & \alpha \mid \text{if } \pi \text{ } \alpha \text{ else } P \\ \alpha & ::= & \text{grant} \mid \text{deny} \mid \text{maybe} \mid \text{onyourown} \\ \pi & ::= & \text{true} \mid (v \ op \ k) \mid (w \ op \ k') \mid \neg\pi \mid \pi \vee \pi \mid (\pi) \end{array}$$

where α represents the four possible decisions, π is a comparison expression, $op \in \{=, \neq, <, >, \leq, \geq\}$, $v \in \mathcal{V}$ ranges over the set of probabilistic parameters, $k \in [0, 1]$, $w \in \mathcal{W}$ ranges over the set of balance parameters, and $k' \in \mathbb{R}$. We omit from the grammar further logical connectives as $\{\neg, \vee\}$ is functionally complete.

The interpretation function $\mathcal{I} : \mathcal{P} \to \mathcal{D}$, with $\mathcal{D}$ the domain of possible decisions, is defined as follows:

$$\begin{array}{rcl} \mathcal{I}(\alpha) & = & \alpha \\ \mathcal{I}(\text{if } \pi \text{ } \alpha \text{ else } P) & = & \begin{cases} \alpha & \textit{if } [\![\pi]\!] = 1 \\ \mathcal{I}(P) & \textit{otherwise} \end{cases} \end{array}$$

where $[\![\pi]\!]$ is the classical semantic function taken from first order logic.

Example 4: Consider the salesman example. The described opportunity o is characterized by:

- $p_o = 0.4$, which is the company estimation of the percentage of contracts actually completed for the specific service negotiated with potential new clients.

- one positive outcome such that $b_o = 1000$, expressing the maximum value of the service negotiated, and $\alpha \in [0.8, 1]$, representing the estimated weight of the service configuration chosen by the new client with respect to the premium service with maximum benefit.
- one threat t that, based on its constituting clues, can lead to two negative outcomes, i.e., the loss of confidential data (with cost equal to $1 \cdot 5000$, which is the value of the information involved in the access request) and the loss due to possibly dishonest behaviors of the client (with cost equal to $1{\cdot}1000$, which is the maximum value of the service purchased).
- $p_t = 0.2$, as resulting from the analysis of the history of analogous threats occurred in the past.

Hence, we have, e.g., $bal_o = -5000$ and $bcbal_o = 1000$ in the case $\alpha = 1$, and $wcbal_o = -6000$. If the policy is:

$$\text{if } \pi \text{ grant else if } \rho \text{ maybe else deny}$$

with $\pi = (bal_o \geq 0) \wedge ((p_t \leq 0.1) \vee (p_o \geq 0.5))$ and $\rho = bcbal_o \geq 900$, then, in the case $\alpha = 1$ the access request could be granted, assumed that, e.g., in order to manage the risk treatment the salesman provides additional references concerning the potential client. However, if α is estimated to be at most 0.8, then the access request is denied.

C. Threat and trust metric

In the previous section, we have included the probabilities associated with beneficial opportunities and threats among the parameters specifying the policy. As illustrated in the example, such probabilities require estimations possibly taking into account knowledge concerning the history of events. In the following, we show how to make such an evaluation automatic through the definition of specific metrics. For the sake of presentation, we assume that different risk events are mutually independent and we illustrate the initialization and management of the probabilistic parameter associated to a given threat t.

By assuming that initially no threat history is available, a threat t starts with a probability $p_t = 0.5$. More specific initialization rules can be applied by considering, e.g., the ratio between the amount of negative clues and the overall number of clues occurring in C.[1] Then, the system takes into account the number of times t may potentially occur, as well as the number of times t actually causes a security incident. Formally, the threat metric expressing the expected probability p_t of a threat t is computed as follows:

$$p_t = \begin{cases} 0.5 & \textit{if } tot_t = 1 \wedge bad_t = 0 \\ bad_t/tot_t & \textit{otherwise} \end{cases} \quad (4)$$

[1] A clue c is negative if the number of security incidents caused by threats in which c occurs is greater than the number of times a threat including c is enabled without inducing any security incident.

where bad_t is the number of times t actually materializes with a bad outcome out of the total amount of times tot_t the same threat is potentially enabled.

Whenever user trustworthiness represents one of the involved clues and the system adopts computational trust management techniques (see, e.g., [19]), it is worth using explicitly the user trust metric (denoted by $T_u \in [0, 1]$, where 1 is maximum trust) in the estimation of the expected probability p_t. To take advantage of such an integration of risk management and trust system, we formally combine the threat metric of (4) with the user trust metric T_u computed by the trust system to estimate the user trustworthiness, thus obtaining:

$$p_t = \begin{cases} \frac{0.5+(1-T_u)}{2} & \textit{if } tot_t = 1 \wedge bad_{ut} = 0 \\ \frac{\frac{bad_t}{tot_t}+(1-T_u)}{2} & \textit{otherwise} \end{cases} \quad (5)$$

We emphasize that the user trust metric depends on the specific trust system, which is supported by the risk management whenever taking into account the history of (potentially risky) decisions that are left to the user and the history of security incidents (even partially) caused by the user behavior.

By following an orthogonal approach, analogous definitions of the opportunity metric enable the automatic estimation of the probabilistic parameter p_o.

Example 5: Consider again the mobile worker example. The opportunity enabled by the remote access would allow the worker to be productive for two hours – with a benefit b_1 equal to $2 \cdot 200$, where 200 represents the hourly cost of the worker – and to respect a project deadline, with a benefit b_2 equal to 5000, which is the estimated benefit in the case the deadline is met. Moreover, we assume $p_o = \alpha_1 = \alpha_2 = 1$. As far as the threat t is concerned, we have two negative outcomes: the potential loss of the project confidential information required by the worker to complete the work by the deadline ($\beta_{t_1} = 1$ and c_{t_1} estimated equal to 1500), and the potential untrusted behavior of the worker missing the deadline ($\beta_{t_2} = 1$ and $c_{t_2} = b_1$). Moreover, we use (5) to estimate p_t, by assuming $T_u = 0.8$ (as estimated by the company trust system) and a history revealing that in two out of five analogous situations the worker spent several hours to surf his favorite social network without completing the expected work. Hence, we obtain $p_t = 0.3$. Then, if the policy is:

$$\text{if } \pi \text{ grant else if } \rho \text{ maybe else deny}$$

with $\pi = (bal_o \geq 1000) \wedge (p_t \leq 0.05) \wedge (p_o \geq 0.8)$ and $\rho = (bal_o \geq 2500) \wedge (p_t \leq 0.3)$, it turns out that the access request could be granted, provided that, e.g., in order to manage the risk treatment the worker leaves the free Wi-Fi and joins the nearby business lounge, which, on the basis of past experience, is known to offer a WPA2 protected Internet connection.

IV. Formal verification

In this section, we analyze formally the OPPRIM system by comparing several alternative models differing for the way in which opportunity and/or risk are taken into account. The analysis is conducted by using the model checker PRISM [5]–[7], through which the different alternative models are described in terms of state-based formal specifications given in a mathematical formalism inspired by the Reactive Modules of [23]. From such formal descriptions it is possible to derive automatically probabilistic models, in the form of discrete-time Markov chains, which can be analyzed through model checking techniques. The properties of interest, which include probabilistic and temporal information, are given in a reward-based probabilistic extension of the Computation Tree Logic (CTL).

A. Scenario

In the case study we consider, the user asks for up to 20 remote access requests that may involve four different assets. The value of the assets is 100, 200, 300, and 1500, respectively. Similarly, we distinguish two potential benefits, with values equal to 150 and 300, respectively, and two different clue sets defining one threat each. In the model, the choice of the asset and of the clue set is probabilistic with uniform distribution, while the choice of the benefit is probabilistic and depends on the given asset requested (the higher the value of the asset is, the higher the probability of the most valued benefit). The combination of assets, benefits, and threats gives rise to several different opportunities characterized as follows. Each opportunity o leads with probability $p_o = 1$ to one benefit with $\alpha = 1$, and to one threat with $\beta = 1$ and associated cost equal to the value of the asset. Four different policies are evaluated, each one defined formally as:

$$\text{if } \pi_i \text{ grant else deny}$$

meaning that the satisfaction of the formula π_i implies that the access request is granted, while it is denied in the opposite case. Each π_i, with $1 \leq i \leq 4$, is expressed as follows:

1) opportunity-enabled risk policy (called *opprim*), represented by the formula:

$$((c_{ot} < tct) \wedge (p_{ot} < tpt)) \vee (p_{ot} < tpt') \vee (bal_o \geq 0) \quad (6)$$

where $c_{ot} = |wcbal_o|$ represents the cost paid because of a security incident caused by the threat associated with the opportunity. Such a cost is equal to the value of the related asset. Moreover, parameter tct is the threat cost threshold, p_{ot} is the probability associated with the threat at hand and is estimated by using (4), while tpt and tpt' represent threat probability thresholds. Notice that this policy combines requirements related to both opportunity benefits and threat costs.

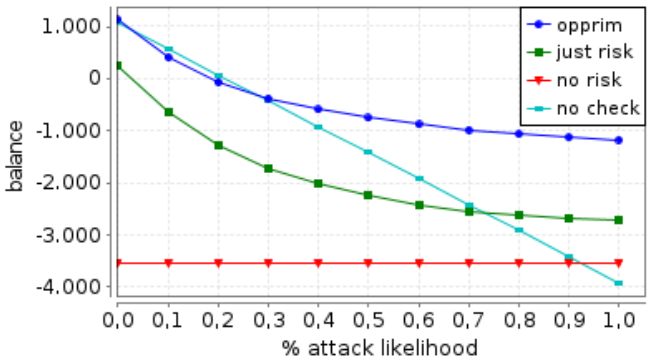


Figure 2: Comparison among the four policies.

2) risk policy (called *just risk*), which ignores any opportunity and is given by a variant of (6) in which the subformula $(bal_o \geq 0)$ is omitted.
3) no risk policy (called *no risk*), which does not admit any risk and is described by the formula $c_{ot} = 0$.
4) no policy (called *no check*), which ignores any risk and is represented by the formula true.

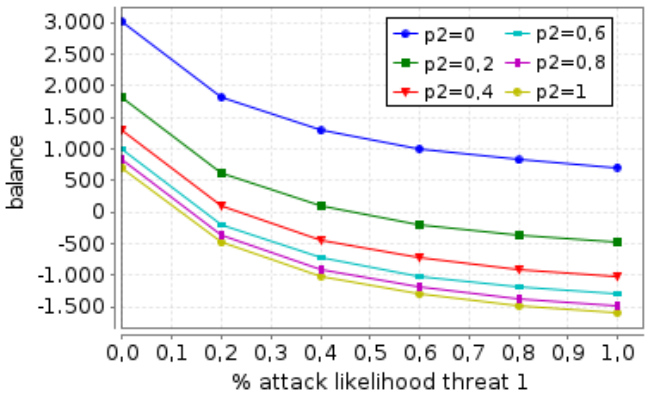
For the experiment of Figure 2, we set $tct = 250$, $tpt = 0.55$, and $tpt' = 0.15$. Moreover, we assume that the actual likelihood of a security incident is 0.5 for the first threat, while for the second threat such a probability varies in the interval $[0, 1]$, as represented in the horizontal axis of the figure. The result reported in the vertical axis expresses the total balance obtained after 20 access requests, which is defined formally by a reward-based CTL formula expressing the following assumptions:

- every time the request is granted a positive reward equal to the related opportunity benefit b_o is added.
- every time the request is granted and a security incident occurs, a negative reward $-c_{ot}$ corresponding to the related threat cost is added.
- every time the request is denied a negative reward $-b_o$ corresponding to the related opportunity benefit is added.

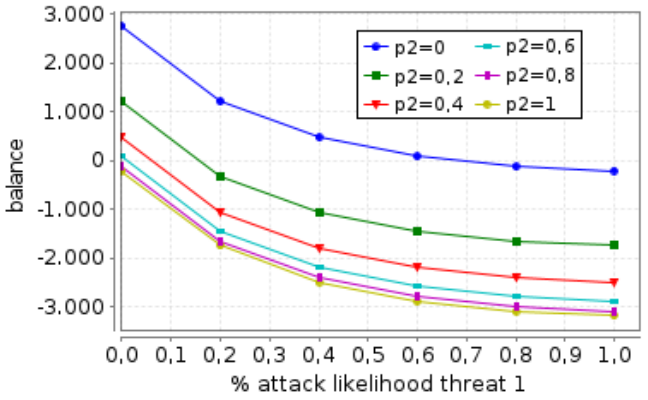
Figure 3 shows the results of sensitive analysis, where for both threats the attack likelihood varies in the interval $[0, 1]$. In particular, probability p_1, reported in the horizontal axis, is associated with the first threat, while each curve refers to a different value of probability p_2, which is related to the second threat.

Figure 4 illustrates the impact of parameter tct, which varies in the interval $[0, 1000]$ in the setting $p_1 = p_2 = 0.5$.

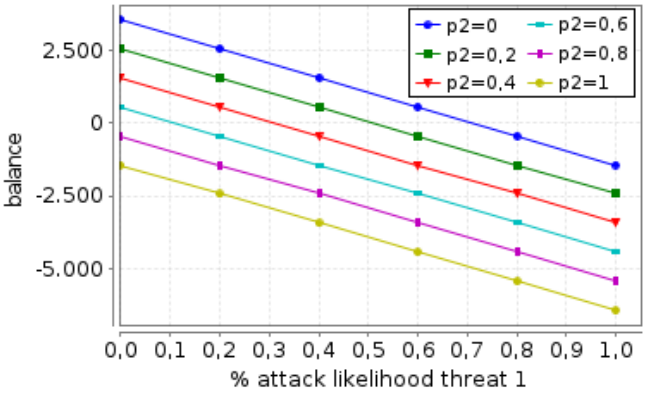
In another set of experiments, we make one of the two abstract threats of the basic model surveyed above more concrete, by assuming that it involves also the trust towards the user. Therefore, for such a threat we use (5), by assuming that initially $T_u = 0.5$ and then it is increased (resp., decreased) by 0.1 every time the threat is potentially enabled



(a) Policy *opprim*.



(b) Policy *just risk*.



(c) Policy *no check*.

Figure 3: Sensitivity analysis by varying threat probabilities.

and a user related security incident does not occur (resp., actually occurs).

In this setting, the *opprim* policy is modeled by means of the following formula:

$$(c_{ot} < tct) \vee (p_{ot} < tpt) \vee (bal_o \geq 0) \qquad (7)$$

The experiment of Figure 5 refers to such a scenario, where $tct = 250$, $tpt = 0.2$, and, similarly as before, the *just risk* policy is a variant of (7) without the condition $(bal_o \geq$

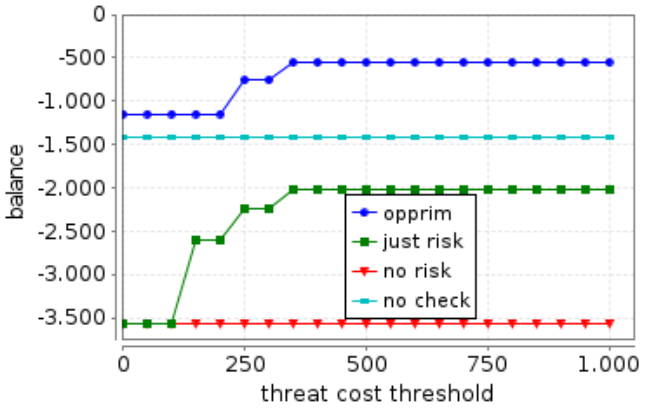


Figure 4: Sensitivity analysis by varying parameter *tct*.

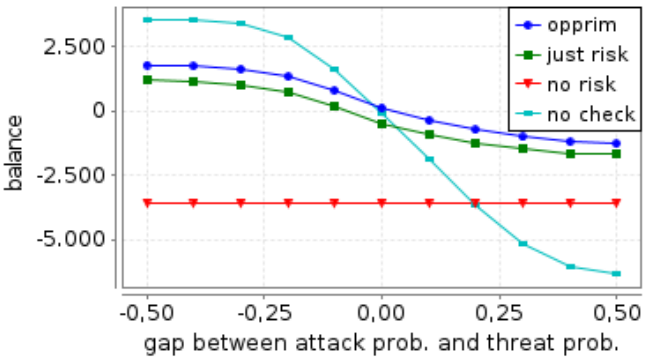


Figure 5: Comparison among the four policies with trust metric.

$0)$. The likelihood of a security incident varies according to the function $p_t + gap$, where the variable gap, which is reported in the horizontal axis of the figure, represents the accuracy with which the threat probability p_t estimated by the policy approximates the actual incident probability. The lower the absolute value of gap is, the higher the accuracy. More precisely, a negative gap stands for an overestimation of the potential security problem induced by the threat, while a positive gap means underestimation.

As an extension of this last scenario, we permit that, once a resource access is granted, the user may decide to miss the opportunity, thus losing the related benefit. We model such a behavior as a function of the user trust, by assuming that the probability of taking the opportunity benefit is $p_o = T_u$. Every time the user does not take the opportunity, then the related trust is decreased by 0.1 if no security incident actually occurs and by 0.2 otherwise. The result of the same analysis of Figure 5 in this enriched setting is reported in Figure 6.

B. Discussion

From the analysis of Figure 2 we observe what follows. Firstly, taking no risk is not a viable option, even in hostile

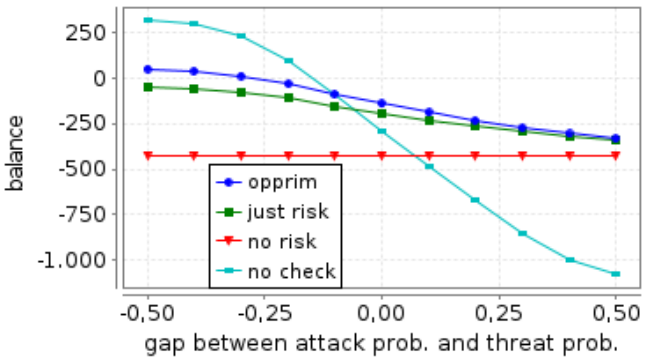


Figure 6: Comparison among the four policies with trust metric and possibly untrusted behavior of the user.

scenarios. In practice, running no risks causes the loss of any opportunity benefit. Secondly, mixing opportunity and risk in the policy allows the decision system to take into account more complete information, thus justifying the performance gain with respect to the case in which opportunities are ignored. Obviously, the gap between these two policies strictly depends on the way in which opportunity conditions are combined with risk constraints. Thirdly, ignoring any risk policy is an option that can provide good results in an ideal scenario in which threats and related security incidents are minimized. In any case, we point out that too rigid constraints in the risk policy represent a severe obstacle for real opportunities and impact dramatically the size of the area in which the *just risk* curve outperforms the *no check* curve.

The sensitivity analysis reported in Figures 3 and 4 confirms the observations above. In particular, Figure 3 reveals that the relation among the policies is invariant with respect to parameters p_1 and p_2. Moreover, Figure 4 shows how tuning one of the parameters affecting the formula, i.e., the threat cost threshold tct, can help to optimize the balance. However, by comparing Figure 4 with Figure 2 in the case $p_1 = p_2 = 0.5$, it is worth observing that the relation among the policies is not sensitive to parameter variations.

By analyzing Figure 5, we first observe that a negative gap means that the threat probability, and therefore the perceived risk, is overestimated by the first two policies, thus motivating the better performance of the *no check* policy. As the gap increases, the scenario becomes more and more hostile and, even if the first two policies tend to underestimate the risk, they turn out to perform better than the *no check* policy. In any case, the *opprim* policy outperforms the *just risk* policy, while both ensure less variability of the balance with respect to the *no check* policy. Finally, Figure 6 shows that introducing more threat factors, like the undesirable behavior of a user asking for exposing resources to risk without any good motivation, emphasizes the advantages of the first two policies. The reason is that these policies take into account the trust metric and are able to adapt the decision making process to the fluctuations of the user (or environment) behavior.

In general, it can be viewed that in the absence of serious threats the application of opportunity and risk policies leads to more conservative behaviors with respect to the case in which a decision system is not used. However, as soon as threats and related consequences become more invasive and of utmost importance, then employing accurate policies balancing in real time opportunity and risk turns out to be fundamental to achieve an optimal tradeoff between benefits and costs.

V. Conclusion

The current BYOD-oriented mobile work trend imposes a redefinition of classical risk management systems, as both threat costs and opportunity benefits shall be considered as first-class citizens in the risk analysis process. The opportunity-enabled risk management system formalized in this paper focuses on both the negative outcomes caused by threats that compromise the assets and the positive outcomes bringing the benefits induced by a successful and secure use of the assets. All the ingredients behind the functioning of OPPRIM have been formalized in a policy specification language. Validation has been provided through model checking based analysis with the aim of emphasizing the role of the various OPPRIM risk, trust, and threat metrics.

With respect to the formal verification, additional experiments in which number (and value) of assets and threats are scaled confirm the results of Section IV. This further analysis is still in progress and is also aimed at providing more details about the impact of each configuration parameter. Analogous results have been obtained through a Java simulator, which has more than 30,000 lines of Java code including a Swing graphical user-interface, publicly available as open source on Github (https://github.com/jmseigneur/opprim-sim).

The OPPRIM model has been integrated in the architecture of the EU-funded MUSES project open source software [24], whose objective is to prevent security incidents by mobile workers accessing corporate data with BYOD and from remote locations such as airports and coffee shops.

Acknowledgment: This work is supported by the EC, under grant 318508, project MUSES: Multiplatform Usable Endpoint Security, FP7-ICT-2011-8, Trustworthy ICT.

References

[1] A. Armando, G. Costa, L. Verderame, and A. Merlo, "Securing the "bring your own device" paradigm," *IEEE Computer*, vol. 47, pp. 48–56, 2014.

[2] J. Clarke, M. Gomez Hidalgo, A. Lioy, M. Petkovic, C. Vishik, and J. Ward, "Consumerization of it: Top risks and opportunities," European Network and Information Security Agency (ENISA), Tech. Rep., 2012.

[3] A. Shostack, "Reinvigorate your threat modeling process," *MSDN Magazine*, 2008.

[4] J.-M. Seigneur, C. Ballester Lafuente, X. Titi, and J. Guislain, "OPPRIM: Opportunity-enabled risk management for trust and risk-aware asset access decision-making," Université de Genève, Tech. Rep., 2015. [Online]. Available: http://archive-ouverte.unige.ch/unige:46443

[5] T. Chen, V. Forejt, M. Kwiatkowska, D. Parker, and A. Simaitis, "Automatic verification of competitive stochastic systems," *Formal Methods in System Design*, vol. 43, no. 1, pp. 61–92, 2013.

[6] M. Kwiatkowska, G. Norman, and D. Parker, "PRISM 4.0: verification of probabilistic real-time systems," in *23rd Int. Conf. on Computer Aided Verification (CAV'11)*, ser. LNCS, vol. 6806. Springer, 2011, pp. 585–591.

[7] V. Forejt, M. Kwiatkowska, G. Norman, and D. Parker, "Automated verification techniques for probabilistic systems," in *Formal Methods for Eternal Networked Software Systems*, ser. LNCS, M. Bernardo and V. Issarny, Eds. Springer, 2011, vol. 6659, pp. 53–113.

[8] *ISO 27005: Information technology-Security techniques - Information security risk management*, International Organization for Standardization Std., 2008.

[9] *ISO 31000: Risk management - Principles and guidelines*, International Organization for Standardization Std., 2009.

[10] *ISO/IEC Guide 73 Risk Management vocabulary*, International Organization for Standardization Std., 2009.

[11] Y. Asnar and P. Giorgini, "Modelling risk and identifying countermeasure in organizations," in *1st Int. Workshop on Critical Information Infrastructures Security (CRITIS'06)*, ser. LNCS, vol. 4347. Springer, 2006, pp. 55–66.

[12] Y. Asnar, R. Moretti, M. Sebastianis, and N. Zannone, "Risk as dependability metrics for the evaluation of business solutions: A model-driven approach," in *3rd Int. Conf. on Availability, Reliability and Security (ARES'08)*. IEEE, 2008, pp. 1240–1247.

[13] M. S. Lund, B. Solhaug, and K. Stølen, "Risk analysis of changing and evolving systems using CORAS," in *Foundations of Security Analysis and Design VI (FOSAD)*, ser. LNCS, A. Aldini and R. Gorrieri, Eds. Springer, 2011, vol. 6858, pp. 231–274.

[14] W. Han, C. Shen, Y. Yin, Y. Gu, and C. Chen, "Using quantified risk and benefit to strengthen the security of information sharing," presented at the 18th ACM Conference on Computer and Communications Security (CCS'11), 2011.

[15] L. Zhang, A. Brodsky, and S. Jajodia, "Toward information sharing: Benefit and risk access control (barac)," in *7th Int. Workshop on Policies for Distributed Systems and Networks*. IEEE, 2006, pp. 45–53.

[16] N. Baracaldo and J. Joshi, "A trust-and-risk aware rbac framework: Tackling insider threat," in *17th Symposium on Access Control Models and Technologies (SACMAT'12)*. ACM, 2012, pp. 167–176.

[17] R. Shaikh, K. Adi, L. Logrippo, and S. Mankovski, "Risk-based decision method for access control systems," in *9th Int. Conf. on Privacy, Security and Trust (PST'11)*. IEEE, 2011, pp. 189–192.

[18] S. Marsh, "Formalising trust as a computational concept," Ph.D. dissertation, Department of Mathematics and Computer Science, University of Stirling, 1994.

[19] A. Jøsang, "Trust and reputation systems," in *Foundations of Security Analysis and Design IV (FOSAD'07)*, ser. LNCS, A. Aldini and R. Gorrieri, Eds. Springer, 2007, vol. 4677, pp. 209–245.

[20] N. Dimmock, "Using trust and risk for access control in global computing," Ph.D. dissertation, University of Cambridge, 2005.

[21] V. Balakrishnan and E. Majd, "A comparative analysis of trust models for multi-agent systems," *Lecture Notes on Software Engineering*, vol. 1, pp. 183–185, 2013.

[22] J.-M. Seigneur, P. Kölndorfer, M. Busch, and C. Hochleitner, "A survey of trust and risk metrics for a byod mobile working world," in *3rd Int. Conf. on Social Eco-Informatics (SOTICS'13)*, 2014, pp. 82–91.

[23] R. Alur and T. Henzinger, "Reactive modules," *Formal Methods in System Design*, vol. 15, pp. 7–48, 1999.

[24] A. M. Mora, P. De las Cuevas, J. J. Merelo, S. Zamarripa, M. Juan, A. I. Esparcia-Alcázar, M. Burvall, H. Arfwedson, and Z. Hodaie, "MUSES: A corporate user-centric system which applies computational intelligence methods," in *29th Annual ACM Symposium on Applied Computing (SAC'14)*. ACM, 2014, pp. 1719–1723.