From Reactionary to Proactive Security: Context-Aware Security Policy Management and Optimization Under Uncertainty

Sivadon Chaisiri
Cyber Security Lab, Department of Computer Science
The University of Waikato, New Zealand
chaisiri@waikato.ac.nz

Ryan K. L. Ko
Cyber Security Lab, Department of Computer Science
The University of Waikato, New Zealand
ryan@waikato.ac.nz

***Abstract*—At the core of its nature, security is a highly contextual and dynamic challenge. However, current security policy approaches are usually static, and slow to adapt to ever-changing requirements, let alone catching up with reality. In a 2012 Sophos survey, it was stated that a unique malware is created every half a second. This gives a glimpse of the unsustainable nature of a global problem; any improvement in terms of closing the "time window to adapt" would be a significant step forward. To exacerbate the situation, a simple change in threat and attack vector or even an implementation of the so-called "bring-your-own-device" paradigm will greatly change the frequency of changed security requirements and necessary solutions required for each new context. Current security policies also typically overlook the direct and indirect costs of implementation of policies. As a result, technical teams often fail to have the ability to justify the budget to the management, from a business risk viewpoint. This paper considers both the adaptive and cost-benefit aspects of security, and introduces a novel context-aware technique for designing and implementing adaptive, optimized security policies. Our approach leverages the capabilities of stochastic programming models to optimize security policy planning, and our preliminary results demonstrate a promising step towards proactive, context-aware security policies.**

I. Introduction

Traditional information security policies, such as access control and firewall policies, usually rely on static information about subjects (e.g. users and processes) [1]. Such policies are slow to adapt to dynamic conditions of surrounding environments. Hence, granting a subject static access to a certain resource may no longer be sufficient to deal with security threats since the subject can be dynamically involved in different context such as time, location, environmental conditions, activities, and behaviors. Changes in such context can have impact on security postures such that security requirements and policies have to be refined [2]. For example, security policies for Bring-Your-Own-Device (BYOD) paradigm [3], Internet-of-Things (IoT) [4], and 5G networks [5] have to be adaptable to context relevant to security and other factors such as computing performance and monetary costs [6].

Context-aware computing is a promising solution for addressing dynamic natures of context [6]. In this paper, we propose a context-aware security policy management framework inspired by a context-aware role-based access control (RBAC) model [7]. Different from traditional RBAC models, this context-aware model records and observes current context status for granting and revoking access permissions. A permission given to a subject can be changed after the subject is involved in new context. For example, access to a file is granted to a subject only when the subject remains in a certain location, and the access is revoked when the subject is away from the location.

In addition to the RBAC-based security control, this paper also considers how other security controls such as lock screen, virtual private network (VPN), and intrusion detection system (IDS), can be appropriately managed in different context. For example, a 4-digit-PIN lock screen can be assigned to protect a mobile device from unauthorized access when the device is located in secure context (e.g. home), whereas the lock screen is switched to a complex-password lock screen in unsafe context (e.g. walking in crowded places) to secure data in the device being stolen or lost. A trade-off between security and other factors (e.g. productivity, computing performance, and monetary costs) needs to be well balanced. For example, although the complex-password lock screen for mobile devices is more secure than the 4-digit-PIN one, it is not efficient to be used if the devices have to be frequently accessed by owners.

This paper considers the adaptive and cost-benefit aspects of security, and also proposes a novel context-aware technique for designing and implementing adaptive, optimized security policies. Cost-benefit analysis [8] is applied to appraise values of access and security controls in different context. Due to uncertainty of threats and resource accesses, our approach leverages the capabilities of stochastic programming models [9] to maximize the total benefit value of security policy planning, and our preliminary results demonstrate a promising step towards proactive, context-aware security policies.

The rest of this paper is organized as follows: Section II presents a review of related work. Section III describes the system model of our proposed framework. The stochastic programming model and algorithm designed for the framework are discussed in Section IV. Section V presents the performance evaluation results. Conclusions are given in Section VI.

II. Related Work

Context-aware RBAC models have been studied for over a decade [1], [7], [10]–[14]. A generic context-aware RBAC model and a policy specification language for defining context-aware RBAC security polices were proposed in [7], [10]. Similar to the RBAC model in [7], context-aware RBAC approaches were implemented in [1], [11] that can detect changes in context and adjust access control according to the changes. An RBAC approach for enforcing a security policy to security controls or devices that lack in functionalities to be aware of context was proposed in [12] by leveraging the policy decision and enforcement points. A programming framework proposed in [13] presents how a context-aware access control policy can be defined and enforced through programmable control. In [14], the concept of *context constraints* was proposed to implement a context-aware RBAC system where certain conditions specified in a context constraint must be first met before processes (e.g. business workflow processes) can be executed.

Not based on an RBAC model, a context-aware security framework in [2] primarily designed for mobile ad-hoc networks monitors behaviors of neighbors (e.g. mobile devices located in the same ad-hoc network) to identify malicious neighbors based on context information (e.g. mobile resource status and other environmental conditions).

Although context-aware security was widely studied, none of the above literature addressed uncertainty of security threats and resource accesses. Optimization of context-aware security policy planning was not well studied in the literature. To the best of our knowledge, mathematical optimization primarily designed for context-aware security policy management under the uncertainty has never been exclusively studied.

III. System Model and Assumption

A. Framework Overview

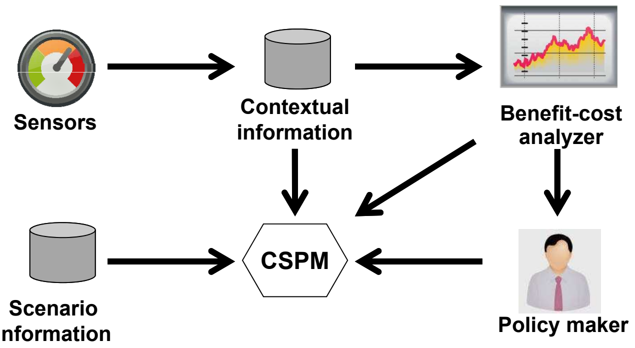

Fig. 1. Key components of the proposed system model.

We propose the context-aware security policy management (CSPM) framework for security policy planning under uncertainty. In its system model, users use devices (e.g. laptops and smartphones) to access local resources (e.g. files and applications) in the devices or remote resources (e.g. clouds). The security policy defines how the access control is configured and other security controls are allocated for securing both local and remote resources in different context. Fig. 1 presents major entities of the framework including CSPM, contextual information, sensors, scenario information, policy maker, and benefit-cost analyzer.

CSPM responsible for making and enforcing security policies is the centralized broker of the framework. To make a security policy, CSPM solves the optimization model derived in Section IV with parameters taken from the policy maker.

The contextual information (in Fig. 1) provides parameters about *contextual states*. A contextual state is a set of information describing context in which users can be involved such as location, time, environmental conditions (e.g. illuminance and ambient noise level of the location), users' actions (e.g. walking and driving), and other information (e.g. TO-DO lists and appointments recorded in cloud services, accessible networks, and remaining battery power of users' devices). For example, {*location: Bob's car, action: driving*}, {*location: customer-A, networks: 3G, WiFi*}, and {*location: Bob's home, action: sleeping, time: 10PM-7AM*} can be three different contextual states associated with Bob i.e. Bob driving his car, Bob positioned at a customer's office where 3G and WiFi networks are available, and Bob sleeping at home during the specific time duration, respectively.

Scenario information provides stochastic information about uncertain parameters that have impact on benefits and costs of security policy enforcement. This scenario information is discussed in Subsections III-E.

With complete contextual and scenario information, CSPM can implement appropriate security policies. Sensors (in Fig. 1) such as closed-circuit televisions, network monitoring systems, positioning systems, haptic sensors, and activity-tracking devices (e.g. Fitbit and Apple Watch) can be leveraged in the framework to provide contextual information. Although machine learning (e.g. *ConXsense* [15]) can be applied to define contextual states, contextual states in this paper are assumed to be predefined by the policy maker. Let $\mathcal{Z}$ denote the set of contextual states.

In Fig. 1, the policy maker, who could be an information security manager or a team of cyber security specialists, harnesses CSPM to make and enforce the security policy. The policy maker provides CSPM with parameters about costs and benefits of different access control configurations and security control allocations. The costs and benefits can be obtained from the benefit-cost analyzer which is discussed in Subsection III-F.

B. Access Control and Other Security Controls

In this paper, CSPM makes a security policy of access control and other security controls. Access control defines users' permissions to access resources. This access control can be defined as an access control matrix that users and resources are called subjects and objects, respectively. Let $\mathcal{S}$ denote the set of subjects. Based on RBAC, a group of subjects can be assigned to the same role such that $\mathcal{S}$ can be the set of roles and subjects.

Let $\mathcal{O}$ and $\mathcal{P}$ denote the sets of objects and permissions (i.e. access modes), respectively. The set of permissions available to object $o \in \mathcal{O}$ is denoted by $\mathcal{P}_o$ ($\mathcal{P}_o \subseteq \mathcal{P}$). For example, {*read, write, read&write*} and {*select, select&update, select&update&insert&delete*} are sets of permissions available to a file (i.e. an object) and a database, respectively. At most one permission for an object can be granted to a subject in the same contextual state.

Let $\alpha_{s,o} \in [0, 1]$ denote the access index of subject $s \in \mathcal{S}$ associated with object o. An access index refers to how necessary or how frequent an object is required by a subject. The higher access index of an object means the object being more necessary. Access indices can be strategically defined by the policy maker. For example, access indices 0.9, 0.8, 0.5, and 0 associated with a database are assigned to database administrator, database analyst, programmer, and accountant roles, respectively. In this example, the database access is not needed by the accountant role to which the access should not be granted. The access indices can be also derived from historical data of accesses which is discussed in Subsection III-E.

In addition to the access control, CSPM can allocate other security controls for securing objects. Basically, a security control refers to a combination of software, hardware, and procedures designed for obtaining confidentiality, integrity, or availability of an information system and providing protection, detection, or recovery control. For example, allocating a VPN to make a connection between a mobile device and a private cloud can achieve confidential communications and allocating a cloud backup service to mobile devices can maintain availability and integrity of data stored in the devices.

Let $\mathcal{C}$ denote the set of security controls. For example, {*VPN, hash-function, lock-screen*} is a set of security controls. Let $\mathcal{V}$ and $\mathcal{V}_c$ ($\mathcal{V}_c \subseteq \mathcal{V}$) denote the set of security control settings and the set of settings of security control $c \in \mathcal{C}$. For example, {*MD5, SHA-1*} and {*complex-password, 4-digit-PIN, fingerprint*} are sets of control settings of hash function and lock screen, respectively. More than one security control can be allocated to an object if and only if at least one object is granted to a subject in the same contextual state. In addition, at most one control setting of a security control can be applied to an object in the same contextual state. The security control setting of an allocated security control can be changed by CSPM according to a dynamic change in context. For example, the allocated setting *SHA-1* can change to *MD5* which is a faster hash function [16] to conserve battery power of a mobile device.

Granting accesses and allocating security controls may result in initial monetary costs (e.g. costs of cloud services and software licenses). Let $K_{o,p,z}$ and $\acute{K}_{o,c,z}$ denote the initial costs of granting permission $p \in \mathcal{P}$ of object o in contextual state $z \in \mathcal{Z}$ and allocating security control c to object o in contextual state z, respectively.

C. Threats and Attributes

Based on the threat model in [17], a threat aims attacks at objects and affects different attributes of information systems. Attributes affected by threats, for example, reputation damage, revenue loss, and performance degradation, can be valued as costs (discussed in Subsection III-F). Let $\mathcal{T}$, $\mathcal{A}$, and T_o^t denote the set of threats, the set of attributes, and the number of attacks of threat $t \in \mathcal{T}$ aiming for object o, respectively.

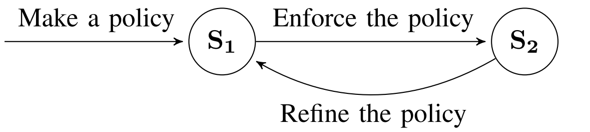

Fig. 2. State diagram of decision stages.

To obtain some security requirements, the policy maker can define $G_o^{a,t}$ denoting the minimum number of attacks of threat t affecting attribute $a \in \mathcal{A}$ of object o that must be mitigated by security controls.

D. Decision Stages

Decision stages are time epochs that CSPM is executed by the policy maker and necessary actions are conducted to make and enforce a security policy. As depicted in Fig. 2, two decision stages ($\mathbf{S_1}$ and $\mathbf{S_2}$) are involved in this framework as follows:

- *Policy making stage* ($\mathbf{S_1}$) – A security policy is made by the CSPM i.e. the access control is configured and security controls are allocated in different contextual states.
- *Policy enforcing stage* ($\mathbf{S_2}$) – CSPM observes benefits of granted permissions and security controls and changes in scenario and contextual information. According to the observed information, CSPM chooses appropriate security control settings to objects in each contextual state. Great fluctuation of benefits, scenario and contextual information requires the policy maker to refine the enforced security policy, and hence the policy making stage is repeated to make a new security policy, and so on.

E. Uncertainty of Threat and Access

Generally, the number of attacks of a threat (i.e. $T_{o,\omega}^t$) is not perfectly known by the policy maker but it can be described by scenarios. A scenario defines a possible number of attacks. Let $\Omega_{t,o}$ denote the set of scenarios of threat t aiming for object o. It is assumed that probability distribution of $\Omega_{t,o}$ for every threat and object is available and has *finite support* i.e. it is comprised of a finite number of scenarios with respective probabilities. Each threat is assumed to be independent of other threats.

Without any historical data of accesses, the policy maker strategically assigns appropriate access indices associated with objects (i.e. $\alpha_{s,o}$) to their subjects (or roles) on a scale of 0 to 1. When CSPM executes until an amount of historical data of accesses can be obtained, access indices can be then derived from the historical data. Let $\grave{\Omega}_{s,o}$ of subject s denote the set of scenarios of access to object o that each scenario refers to the number of accesses. It is assumed that probability distribution of $\grave{\Omega}_{s,o}$ with respective probabilities for every subject and object is available and has *finite support*, and each subject's accesses are independent of other subjects'. For scenario $\omega \in \grave{\Omega}_{s,o}$, the access index of subject s associated with object o can be defined as $\alpha_{s,o,\omega} = \Upsilon[\alpha'_{s,o,\omega}]$ where $\alpha'_{s,o,\omega}$ denotes the possible number of accesses under scenario ω. Function $\Upsilon[x] \in [0,1]$ used for a normalization purpose (discussed more in Subsection III-F) is fully defined as follows:

$$\Upsilon[x] = \frac{x}{U_x} \tag{1}$$

where U_x denotes the upper bound of x. For parameter $\alpha'_{s,o,\omega}$, its upper bound is the maximum number of accesses to object o.

It is assumed that all subject's accesses are independent of every threat. Hence, all scenarios denoted by Ω can be obtained through the Cartesian product as follows:

$$\Omega \;=\; \prod_{t\in\mathcal{T}}\prod_{o\in\mathcal{O}}\Omega_{t,o} \times \prod_{s\in\mathcal{S}}\prod_{o\in\mathcal{O}}\grave{\Omega}_{s,o}. \tag{2}$$

Let π_ω denote the probability of scenario $\omega \in \Omega$. If the set of objects is $\{o_1, o_2, \ldots, o_n\}$ (n is the cardinality of set $\mathcal{O}$ or $\mid \mathcal{O} \mid= n$), the scenarios of threats and accesses for every object under scenario ω are represented by vectors $\xi^t_\omega = \left(T^t_{o_1\omega}, T^t_{o_2\omega}, \ldots, T^t_{o_n\omega}\right)$ and $\grave{\xi}^s_\omega = \left(\alpha_{s,o_1,\omega}, \alpha_{s,o_2,\omega}, \ldots, \alpha_{s,o_n,\omega}\right)$, respectively.

In Fig. 1, the scenario information provides CSPM with the stochastic information i.e. values of π_ω, ξ^t_ω and $\grave{\xi}^s_\omega$. This scenario information can be updated according to historical data related to threats and accesses.

F. Cost-Benefit Analysis

Each permission can yield different benefits and costs e.g. full access to a database may increase productivity gain but it may incur costs (e.g. reputation damage) if sensitive data from the database is leaked due to an uncertain threat. Similarly, each allocated security control can result in different benefits and costs e.g. *MD5* is a faster hash function but less secure than *SHA-1* [16]. CSPM must increase the total benefit while the total cost must be reduced by deploying appropriate access control and other security controls in every contextual state. It is assumed that costs and benefits are obtained by the benefit-cost analyzer (in Fig. 1) which could be tools (e.g. business intelligence tools) and professionals (e.g. information security manager and risk analysts).

Cost-benefit analysis (CBA) can be used to estimate benefits and costs of information security controls [8]. Since a discussion about CBA is beyond the scope of this paper, we adopt the multi-attribute risk assessment from [17] to the framework that multiple attributes (i.e. $\mathcal{A}$) can be valued as benefits and costs of access control or security control. For example, a 7-point Likert-type scale values reputation impact [17], and security and performance levels with a scale of 0 to 1 value security and performance attributes [18]. An attribute can be classified as either benefit or cost attribute. Benefit attribute values (e.g. security level and revenue gain) and cost attribute values (e.g. reputation damage and revenue loss) have to be maximized and minimized, respectively. Let $\mathcal{A}^+ \subseteq \mathcal{A}$ and $\mathcal{A}^- \subseteq \mathcal{A}$ ($\mathcal{A}^+ \cap \mathcal{A}^- = \emptyset$) denote the sets of benefit attributes and cost attributes, respectively. In this framework, a multi-attribute value equals the total value of all benefit attributes subtract the total value of all cost attributes.

Let $\beta_{s,o,p,z,\omega}$ denote the multi-attribute value (i.e. the sum of benefits and costs) of permission p of object o granted to subject s in contextual state z under scenario ω as defined as follows:

$$\beta_{s,o,p,z,\omega} \;=\; \alpha_{s,o,\omega}\sum_{a\in\mathcal{A}^+} W_a \Upsilon[B^a_{s,o,p,z}] \tag{3}$$

where $B^a_{s,o,p,z}$ of attribute a denotes the benefit of granted permission.

$\Upsilon[x]$ defined in Subsection III-E normalizes attribute values to have the same lower and upper bounds, although attributes use different units of measurement (e.g. hours, dollars, and bytes). Hence, the normalized attributes can be summed together. W_a denotes weight of attribute a where $\sum_{a\in\mathcal{A}} W_a = 1$. The policy maker prioritizes the weighted attributes i.e. the larger weighted attribute increases its priority.

Next, let $\acute{\beta}_{o,c,v,z,\omega}$ denote the multi-attribute value of setting v of security control c allocated to object o in contextual state z under scenario ω as defined as follows:

$$\begin{aligned}\acute{\beta}_{o,c,v,z,\omega} =& \mid \mathcal{T} \mid \left(\sum_{s\in\mathcal{S}}\alpha_{s,o,\omega}\right)\sum_{a\in\mathcal{A}^+} W_a \Upsilon[\acute{B}^a_{c,v,z}] \\ &-\sum_{t\in\mathcal{T}}(1-E^t_{c,v})\ \Upsilon[T^t_{o,\omega}]\sum_{a\in\mathcal{A}^-} W_a\Upsilon[\acute{D}^{a,t}_o]\end{aligned} \tag{4}$$

where $\acute{B}^a_{c,v,z}$ and $\acute{D}^{a,t}_o$ of attribute a denote the benefit of the allocated security control and the damage cost per attack of threat t affecting object o that cannot be blocked by the security control, respectively. $E^t_{c,v}$ denotes percentage of threat t that can be blocked by the security control i.e. effectiveness of the security control. According to strength of a security control, a confidence level [19] can be applied to $E^t_{c,v}$ as well.

G. Decision Variables

A solution for CSPM is represented by decision variables describing access control configurations and security control allocations. The framework is comprised of three groups of binary decision variables as follows:

- $X_{s,o,p,z} \in \{0,1\}$ indicates if permission p for object o is granted to subject s in contextual state z (i.e. $X_{s,o,p,z}$ equals 1). If subject s is refused to access object o in contextual state z, $X_{s,o,p,z}$ equals 0 for every permission $p \in \mathcal{P}_o$.
- $Y_{o,c,z} \in \{0,1\}$ indicates if security control c is allocated to object o in contextual state z (i.e. $Y_{o,c,z}$ equals 1).
- $R_{o,c,v,z,\omega} \in \{0,1\}$ indicates if setting $v \in \mathcal{V}_c$ of security control c is applied to object o in contextual state z under

scenario ω (i.e. $R_{o,c,v,z,\omega}$ equals 1). If control c allocated to object o is disabled in contextual state z, $R_{o,c,v,z,\omega}$ equals zero for every $v \in \mathcal{V}_c$.

IV. Security Policy Optimization

A two-stage stochastic programming (SP) model [9] can be formulated to obtain an optimal solution for CSPM as follows:

$$\text{Max}\left[-\mathscr{F}(X,Y) + \mathbb{E}_{\Omega}\left[\mathscr{Q}(X,Y,\omega)\right]\right] \tag{5}$$

$$\text{s.t.} \quad X_{s,o,p,z} \in \{0,1\} \ ; \ \forall s \in \mathcal{S}, o \in \mathcal{O}, p \in \mathcal{P}_o, z \in \mathcal{Z} \tag{6}$$

$$Y_{o,c,z} \in \{0,1\} \ ; \ \forall o \in \mathcal{O}, c \in \mathcal{C}, z \in \mathcal{Z} \tag{7}$$

$$\sum_{p \in \mathcal{P}_o} X_{s,o,p,z} \leq 1 \ ; \ \forall s \in \mathcal{S}, o \in \mathcal{O}, z \in \mathcal{Z} \tag{8}$$

$$Y_{o,c,z} \leq \sum_{s \in \mathcal{S}} \sum_{p \in \mathcal{P}_o} X_{s,o,p,z} \ ; \ \forall o \in \mathcal{O}, c \in \mathcal{C}, z \in \mathcal{Z}. \tag{9}$$

The objective function in (5) maximizes the total benefit of both access control configuration and security control allocation under the uncertainty. Constraints (6) and (7) control possible values of binary variables $X_{s,o,p,z}$ and $Y_{o,c,z}$, respectively. Constraint (8) ensures that at most one permission of object o is granted to subject s in contextual state z. Constraint (9) ensures that a security control can be allocated to an object if and only if the object is granted to at least one subject in the same contextual state.

The objective function (5) contains two cost functions i.e. $\mathscr{F}(\cdot)$ and $\mathscr{Q}(\cdot)$. X and Y denote composite variables of all decision variables $X_{s,o,p,z}$ and $Y_{o,c,z}$, respectively. Function $\mathscr{F}(\cdot)$ denoting the total initial cost incurred in the policy making stage can be defined as follows:

$$\begin{aligned}\mathscr{F}(X,Y) \ = \ & \sum_{s \in \mathcal{S}} \sum_{o \in \mathcal{O}} \sum_{p \in P_o} \sum_{z \in \mathcal{Z}} \Upsilon[K_{o,p,z}] \ X_{s,o,p,z} \\ & + \sum_{o \in \mathcal{O}} \sum_{c \in \mathcal{C}} \sum_{z \in \mathcal{Z}} \Upsilon[\acute{K}_{o,c,z}] \ Y_{o,c,z}.\end{aligned} \tag{10}$$

In (10), the total cost from $\mathscr{F}(\cdot)$ includes costs of access control configuration and security control allocation. As discussed in Subsection III-F, $\Upsilon[x]$ is used for the normalization purpose. $\mathscr{F}(\cdot)$ is expressed as a negative function value as shown in (5) because it yields the cost that needs to be minimized in the maximization problem.

$\mathbb{E}_{\Omega}[\cdot]$ in (5) denotes the expectation (i.e. expected value) of costs incurred by function $\mathscr{Q}(\cdot)$ given every scenario $\omega \in \Omega$. Function $\mathscr{Q}(\cdot)$ in (5) denotes the total benefit value incurred in the policy enforcing stage under scenario $\omega \in \Omega$ and can be fully expressed as a maximization problem as follows:

$$\mathscr{Q}(X,Y,\omega) = \text{Max} \ \left[\mathscr{G}(X,Y,\omega)\right] \tag{11}$$

$$\begin{aligned}\text{where } \mathscr{G}(X,Y,\omega) = & \\ & \sum_{s \in \mathcal{S}} \sum_{o \in \mathcal{O}} \sum_{p \in P_o} \sum_{z \in \mathcal{Z}} \beta_{s,o,p,z,\omega} \ X_{s,o,p,z} \\ & + \sum_{o \in \mathcal{O}} \sum_{c \in \mathcal{C}} \sum_{v \in \mathcal{V}_c} \sum_{z \in \mathcal{Z}} \acute{\beta}_{o,c,v,z,\omega} \ R_{o,c,v,z,\omega}\end{aligned} \tag{12}$$

$$\text{s.t.} \ R_{o,c,v,z,\omega} \in \{0,1\} \ ; \ \forall o \in \mathcal{O}, c \in \mathcal{C}, v \in \mathcal{V}_c, z \in \mathcal{Z} \tag{13}$$

$$\sum_{v \in \mathcal{V}_c} R_{o,c,v,z,\omega} \leq 1 \ ; \ \forall o \in \mathcal{O}, c \in \mathcal{C}, z \in \mathcal{Z} \tag{14}$$

$$R_{o,c,v,z,\omega} \leq Y_{o,c,z} \ ; \ \forall o \in \mathcal{O}, c \in \mathcal{C}, v \in \mathcal{V}_c, z \in \mathcal{Z} \tag{15}$$

$$\begin{aligned}G_o^{a,t} \leq \sum_{c \in \mathcal{C}} \sum_{v \in \mathcal{V}_c} \sum_{z \in \mathcal{Z}} E_{c,v}^t \ T_{o,\omega}^t \ R_{o,c,v,z,\omega} \ ; \ \forall a \in \mathcal{A}, & \\ t \in \mathcal{T}, o \in \mathcal{O}.&\end{aligned} \tag{16}$$

Function $\mathscr{Q}(\cdot)$ in (11) maximizes the total benefit incurred by function $\mathscr{G}(\cdot)$ while the decision variable values of X and Y are fixed values of the optimization model with only decision variables $R_{o,c,v,z,\omega}$. As defined in (12), function $\mathscr{G}(\cdot)$ yields the benefit value of granted permissions and executed security controls under scenario ω. Constraint (13) controls possible values of binary variable $R_{o,c,v,z,\omega}$. Constraint (14) ensures that at most one setting of an allocated security control can be applied to an object in each contextual state. Constraint (15) ensures that a security control must be first allocated to an object in the policy making stage before its control setting can be applied to the object in the policy enforcing stage. Constraint (16) governs the number of blocked threats that must be mitigated by allocated security controls.

Based on the assumption of the probability distribution of Ω having *finite support*, the SP model in (5)$-$(9) can be transformed into the deterministic equivalent model as follows:

$$\text{Max}\left[-\mathscr{F}(X,Y) + \sum_{\omega \in \Omega} \pi_{\omega} \ \mathscr{G}(X,Y,\omega)\right] \tag{17}$$

$$\text{s.t.} \quad (6),(7),(8),(9)$$

$$\begin{aligned}R_{o,c,v,z,\omega} \in \{0,1\} \ ; \ \forall o \in \mathcal{O}, c \in \mathcal{C}, v \in \mathcal{V}_c, z \in \mathcal{Z}, & \\ \omega \in \Omega &\end{aligned} \tag{18}$$

$$\sum_{v \in \mathcal{V}_c} R_{o,c,v,z,\omega} \leq 1 \ ; \ \forall o \in \mathcal{O}, c \in \mathcal{C}, z \in \mathcal{Z}, \omega \in \Omega \tag{19}$$

$$\begin{aligned}R_{o,c,v,z,\omega} \leq Y_{o,c,z} \ ; \ \forall o \in \mathcal{O}, c \in \mathcal{C}, v \in \mathcal{V}_c, z \in \mathcal{Z}, & \\ \omega \in \Omega &\end{aligned} \tag{20}$$

$$\begin{aligned}G_o^{a,t} \leq \sum_{c \in \mathcal{C}} \sum_{v \in \mathcal{V}_c} \sum_{z \in \mathcal{Z}} E_{c,v}^t \ T_{o,\omega}^t \ R_{o,c,v,z,\omega} \ ; \ \forall a \in \mathcal{A}, & \\ t \in \mathcal{T}, o \in \mathcal{O}, \omega \in \Omega.&\end{aligned} \tag{21}$$

The optimization model in (17)$-$(21) can be solved by a branch-and-bound method [20] or optimization solver software supporting integer programming problems such as GLPK [21], NEOS Solvers [22], and GAMS/CPLEX [23].

Algorithm 1 Security Policy Optimization Algorithm

```
1: while CSPM is running do
2:     Begin the policy making stage.
3:     Solve the optimization model in (17)−(21) to obtain a new security
       policy i.e. $X^*$, $Y^*$, and $R^*$.
4:     Configure access control according to $X^*$.
5:     Allocate security controls according to $Y^*$.
6:     Enforce the security policy.
7:     End the policy making stage.
8:     Begin the policy enforcing stage.
9:     repeat
10:        Observe the contextual information, the scenario information, and
           benefits and costs of permissions and security controls.
11:        Choose control settings according to $R^*$ and observed contextual
           states.
12:        if CSPM requires the policy maker's attention then
13:            Inform the policy maker of observed information.
14:        end if
15:    until the policy maker refines the security policy.
16:    End the policy enforcing stage.
17: end while
```

CSPM is executed by Algorithm 1 where X^*, Y^*, and R^* denote decision variable values for access control configuration (i.e. $X_{s,o,p,z}$) and security control allocation (i.e. $Y_{o,c,z}$ and $R_{o,c,v,z,\omega}$). In the policy enforcing stage (lines 8−16), the enforced security policy can be refined by the policy maker when observed key information (i.e. line 10) significantly fluctuates e.g. there are great changes in the probability distribution of scenarios and new contextual states to be added.

V. Performance Evaluation

To evaluate the proposed framework, the optimization model in (17)−(21) is implemented and solved by GAMS/CPLEX [23]. A numerical study with test parameters is conducted through a teleworking case [24] to which the optimization model can be applied. Then, a simulation program is implemented by GAMS/CPLEX to evaluate Algorithm 1 and compare the solution achieved from the algorithm with other competitive solutions.

A. Numerical Study

In this numerical study, two roles or groups of subjects i.e. programmer role (s_1) and sales role (s_2) are considered. Both roles are allowed to work in a co-working space named *CW* and their houses. Only a file server (o_1) and voice over IP (VoIP) (o_2) with two permissions each are enabled to their teleworking i.e. *read-only* (p_1), *read&write* (p_2) for o_1 and *receive-only* (p_1), *receive&dial* (p_2) for o_2. That is, p_2 provides a higher privilege than p_1. The initial costs for granting every permission ($K_{o,p,z}$) equal zero.

TABLE I
Effectiveness of security controls.

$E^t_{c,v}$	c_1		c_2		c_3	
	v_1	v_2	v_1	v_2	v_1	v_2
t_1	0.00	0.00	0.40	0.60	0.00	0.00
t_2	0.40	0.75	0.00	0.00	0.00	0.00
t_3	0.00	0.00	0.50	0.20	0.00	0.00
t_4	0.00	0.00	0.00	0.00	0.95	0.65

TABLE II
Scenario parameters.

ω	α_{s_1,o_1}	α_{s_1,o_2}	α_{s_2,o_1}	α_{s_2,o_2}	$\Upsilon[T^t_o]$	π
ω_1	0.95	0.75	0.55	0.85	1.00	0.02
ω_2	0.80	0.60	0.40	0.70	0.10	0.20
ω_3	0.45	0.20	0.15	0.30	1.00	0.08
ω_4	0.20	0.10	0.15	0.10	0.10	0.70

The company provides both roles with three security controls, i.e. VPN (c_1), host-based IDS (c_2), and lock screen (c_3), with two control settings each (v_1 and v_2). The normalized initial costs to allocate the security controls ($\Upsilon[\acute{K}_{o,c,z}]$) are 0.67, 1, and 0, respectively. To deal with four threats, namely intrusion (t_1), eavesdropping (t_2), malware (t_3), and device loss/theft (t_4), the policy maker of the company leverages CSPM for enforcing a security policy in four contextual states i.e. *CW_8:00AM-6:59PM* (z_1), *CW_7:00PM-7:59AM* (z_2), *CW_TeamMeeting* (z_3), and *Houses* (z_4). The first three contextual states are relevant to the co-working space, while the last contextual state is relevant to the subjects' houses. These four threats aim for each object with the same number of attacks. Effectiveness to block threats of each security control ($E^t_{c,v}$) is presented in Table I. The minimum number of threats that must be blocked by each security control ($G^{a,t}_o$) is zero. Two positive attributes and one negative attribute are used to evaluate benefit and cost values of access and security controls i.e. productivity gain, security-level, and reputation damage, respectively. Weights assigned to the attributes (W_a) are 0.30, 0.50, and 0.20, respectively. That is, the security-level is the top priority attribute.

This numerical study evaluates four scenarios ($\omega_1 - \omega_4$) whose values are presented in Table II including access indices of objects o_1 and o_2 associated with s_1 (α_{s_1,o_1} and α_{s_1,o_2}) and s_2 (α_{s_2,o_1} and α_{s_2,o_2}), the normalized number of attacks of each threat ($\Upsilon[T^t_o]$) and the probability of each scenario (π). Benefit values of granting permissions ($\beta_{s,o,p,z,\omega}$) and allocating security controls ($\acute{\beta}_{o,c,v,z,\omega}$) are shown in Table III (at the top of the next page) which are synthesized from data in [16]–[18], [25] and parameters in Table II.

TABLE III

Benefit values of access permissions ($\beta_{\mathbf{s,o,p,z},\omega}$) and security controls ($\acute{\beta}_{\mathbf{o,c,v,z},\omega}$).

$\beta_{s,o,p,z,\omega}$			z_1				z_2				z_3				z_4			
			ω_1	ω_2	ω_3	ω_4	ω_1	ω_2	ω_3	ω_4	ω_1	ω_2	ω_3	ω_4	ω_1	ω_2	ω_3	ω_4
s_1	o_1	p_1	0.29	0.25	0.14	0.06	0.41	0.35	0.20	0.09	0.61	0.51	0.29	0.13	0.43	0.36	0.20	0.09
		p_2	0.35	0.29	0.16	0.07	0.49	0.42	0.23	0.10	0.51	0.43	0.24	0.11	0.34	0.29	0.16	0.07
	o_2	p_1	0.27	0.22	0.07	0.04	0.33	0.27	0.09	0.04	0.34	0.27	0.09	0.05	0.53	0.43	0.14	0.07
		p_2	0.24	0.20	0.07	0.03	0.30	0.24	0.08	0.04	0.30	0.24	0.08	0.04	0.38	0.30	0.10	0.05
s_2	o_1	p_1	0.21	0.15	0.06	0.06	0.23	0.16	0.06	0.06	0.31	0.23	0.09	0.09	0.14	0.10	0.04	0.04
		p_2	0.10	0.07	0.03	0.03	0.12	0.09	0.03	0.03	0.20	0.14	0.05	0.05	0.02	0.02	0.01	0.01
	o_2	p_1	0.44	0.36	0.16	0.05	0.43	0.35	0.15	0.05	0.55	0.45	0.19	0.06	0.60	0.50	0.21	0.07
		p_2	0.45	0.37	0.16	0.05	0.44	0.36	0.16	0.05	0.56	0.46	0.20	0.07	0.53	0.43	0.19	0.06
$\acute{\beta}_{o,c,v,z,\omega}$			z_1				z_2				z_3				z_4			
			ω_1	ω_2	ω_3	ω_4	ω_1	ω_2	ω_3	ω_4	ω_1	ω_2	ω_3	ω_4	ω_1	ω_2	ω_3	ω_4
o_1	c_1	v_1	5.53	4.75	1.93	1.35	5.53	4.75	1.93	1.35	6.73	5.71	2.41	1.63	7.33	6.19	2.65	1.77
		v_2	9.13	7.63	3.37	2.19	9.13	7.63	3.37	2.19	9.73	8.11	3.61	2.33	10.93	9.07	4.09	2.61
	c_2	v_1	4.49	3.81	1.61	1.09	4.49	3.81	1.61	1.09	4.49	3.81	1.61	1.09	0.89	0.93	0.17	0.25
		v_2	2.65	2.37	0.85	0.67	2.65	2.37	0.85	0.67	2.65	2.37	0.85	0.67	2.65	2.37	0.85	0.67
	c_3	v_1	0.38	0.54	-0.05	0.13	0.38	0.54	-0.05	0.13	0.38	0.54	-0.05	0.13	0.38	0.54	-0.05	0.13
		v_2	5.38	4.57	1.92	1.31	5.38	4.57	1.92	1.31	5.38	4.57	1.92	1.31	5.38	4.57	1.92	1.31
o_2	c_1	v_1	5.91	5.15	1.51	0.75	5.91	5.15	1.51	0.75	7.19	6.19	1.91	0.91	7.83	6.71	2.11	0.99
		v_2	9.82	8.28	2.78	1.24	9.82	8.28	2.78	1.24	10.46	8.80	2.98	1.32	11.74	9.84	3.38	1.48
	c_2	v_1	4.63	4.11	1.11	0.59	4.63	4.11	1.11	0.59	4.63	4.11	1.11	0.59	0.79	0.99	-0.09	0.11
		v_2	2.69	2.55	0.49	0.35	2.69	2.55	0.49	0.35	2.69	2.55	0.49	0.35	2.69	2.55	0.49	0.35
	c_3	v_1	0.39	0.59	-0.14	0.06	0.39	0.59	-0.14	0.06	0.39	0.59	-0.14	0.06	0.39	0.59	-0.14	0.06
		v_2	5.71	4.95	1.48	0.72	5.71	4.95	1.48	0.72	5.71	4.95	1.48	0.72	5.71	4.95	1.48	0.72

TABLE IV

Access control configuration.

$X_{s,o,p,z}$			z_1	z_2	z_3	z_4
s_1	o_1	p_1			✓	✓
s_1	o_1	p_2	✓	✓		
s_1	o_2	p_1	✓	✓	✓	✓
s_2	o_1	p_1	✓	✓	✓	✓
s_2	o_2	p_1				✓
s_2	o_2	p_2	✓	✓	✓	

TABLE V

Security control allocation.

$R_{o,c,v,z,\omega}$			z_1	z_2	z_3	z_4
o_1	c_1	v_2	✓	✓	✓	✓
o_1	c_2	v_1	✓	✓	✓	
o_1	c_2	v_2				✓
o_1	c_3	v_2	✓	✓	✓	✓
o_2	c_1	v_2	✓	✓	✓	✓
o_2	c_2	v_1	✓	✓	✓	
o_2	c_3	v_2	✓	✓	✓	✓

Next, the optimization model in (17)−(21) is solved with the described parameters to obtain a security policy. Table IV presents how the access control is configured according to decision variable values of $X_{s,o,p,z}$ i.e. $X_{s,o,p,z}$ equals 1 (✓ in the table) for a granted permission. The *read&write* permission (p_2) of the file server (o_1) is granted to the programmers (s_1) in contextual states z_1 and z_2 since they can generate more benefits when working at the co-working space. When the programmers have a meeting (z_3) or work at home (z_4), the *read-only* permission (p_1) is granted to the programmers due to the lower benefits of *read&write* permission e.g. working at home does not require a high access right which potentially compromises data security. The *receive-only* permission (p_1) of VoIP (o_2) is granted to the programmers in every contextual state since they cannot produce higher benefits from the *receive&dial* permission (p_2). For the sales role (s_2), the permission *read-only* of the file server is granted to them in every contextual state since they do not need the higher access right of the *read&write* permission due to low access to o_1 as presented in Table II and they cannot produce greater benefits. When working at the co-working space, the sales team has the *receive&dial* permission of VoIP due to the higher benefits of the permission e.g. the sales team can regularly contact their customers. However, the *receive-only* permission is granted to the sales team when they work at home e.g. due to a privacy reason, the sales team working at home may receive calls from customers but may not pick up the calls.

Table V (at the bottom of the previous page) shows how the security controls and their control settings are allocated to and selected for objects in different contextual states, respectively, according to decision variable values of $R_{o,c,v,z,\omega}$ i.e. $R_{o,c,v,z,\omega}$ equals 1 (✓ in the table) for a selected control setting. Based on the test parameters, an allocated security control will be applied to every scenario. Security setting v_2 of the VPN control (c_1) is allocated to both objects in every contextual state due to its higher effectiveness than v_1 as shown in Table I e.g. a TLS-based VPN server using 256-bit AES keys (v_2) is more secure than a PPTP-based VPN server (v_1) [25]. In contrast, although the effectiveness of v_2 of the lock screen control (c_3) is lower than that of v_1, v_2 is applied to both objects in every contextual state e.g. a 4-digit-PIN lock screen which is less secure than a complex-password one provides a higher benefit value (i.e. higher productivity gain in this evaluation). For the host-based IDS (c_2), control setting v_1 is applied to both objects accessed from only the co-working space e.g. data security in the co-working space can be compromised by intrusion (t_1) and malware (t_3) more than the subjects' houses (z_4).

B. Simulation Results

The two decision stages discussed in Subsection III-D are simulated to compare the security policy obtained from the optimization model with other solutions. The subjects and objects from the previous numerical study are reused, while benefits and costs of allocated security controls and the four threats are ignored. One hundred contextual states are considered in this simulation where the benefit of permission ($\beta_{s,o,p,z,\omega}$) in each contextual state is generated by the uniform distribution function, provided by GAMS [23], with a fixed seed value (i.e. $SEED = 3,141$). A discrete probability distribution based on the normal distribution with mean 50.50 and variance 36 is applied to one hundred scenarios.

The simulation compares the security policy obtained from Algorithm 1 (i.e. CSPM solution) with security policies obtained from competitive solutions as follows:

- *Perfect solution (PS)* – This solution is based on an assumption that a scenario that will occur in the (future) policy enforcing stage is perfectly known since the (present) policy making stage, and hence this solution always yields the perfect solution. Note that this perfect solution is an ideal solution which cannot be practically implemented.
- *Best-benefit-permission solution (BBPS)* – This solution chooses access control permissions that yield the best benefits without knowing the probabilities of their occurrences.
- *Random solutions (RS)* – This solution randomly configures access control permissions. For the evaluation, the uniform distributions with seeds $1,000$, $2,000$, and $3,000$ are used for generating three different random solutions.

For each compared solution, the simulation repeats $1,000$ iterations of the policy making and enforcing stages to obtain $1,000$ security policies. In each iteration, a scenario is sampled based on a Monte Carlo method [26] such that the access indices of all objects are observed and the final security policy can be obtained. Then, the average value of the total benefits of all security policies is calculated and compared with the other solutions.

Obviously, PS obtains the best (but ideal) security policy. Therefore, we use the average benefit value of PS as the baseline. Percentage differences of average benefit values between the baseline and the CSPM, BBPS, RND with $SEED = 2,000$, RND with $SEED = 3,000$, and RND with $SEED = 1,000$ solutions are 25.42%, 28.56%, 53.58%, 56.59%, and 61.13%, respectively. The results show that the CSPM solution outperforms the other solutions when the uncertainty is taken into account.

VI. Conclusions and Future Work

We proposed the CSPM framework for planning a context-aware security policy by formulating and solving the stochastic programming model. The security policy defines access control and security control settings for different contextual states. That is, access and security controls are adaptively adjusted according to context (e.g. location, time, and environmental conditions). The framework can maximize benefits of access and security controls, while the uncertainty of threats and accesses is taken into account. The numerical study and simulation were conducted to evaluate the framework. The evaluation results show that the framework can outperform other competitive solutions when the uncertainty is taken into account. The framework with ambient intelligence will be useful for information security managers to make adaptive security policies in context-aware computing environments e.g. BYOD [3], IoT [4], and 5G wireless networks [5].

For our future work, the framework will be developed for real-world applications such as BYOD, parental control, IoT, and 5G network applications. A machine learning approach (e.g. *ConXsense* [15]) will be applied to automatically define contextual states. Effective cost-benefit analysis for context-aware security will be intensively explored. Computational complexity of the stochastic programming model will be studied and solutions for addressing its complexity issues will be proposed.

Acknowledgments

This research was supported by the New Zealand Office of Privacy Commissioner and STRATUS (Security Technologies Returning Accountability, Trust and User-Centric Services in the Cloud) (https://stratus.org.nz), a project funded by the New Zealand Ministry of Business, Innovation and Employment. We would like to thank Dr. Harris Lin and Yangchao Dong from the Cyber Security Lab, the University of Waikato for their comments and suggestions for this research work.

References

[1] S.-H. Park, Y.-J. Han, and T.-M. Chung, "Context-role based access control for context-aware application," in *High Performance Computing and Communications*. Springer, 2006, pp. 572–580.

[2] W. Li, A. Joshi, and T. Finin, "Cast: Context-aware security and trust framework for mobile ad-hoc networks using policies," *Distributed and Parallel Databases*, vol. 31, no. 2, pp. 353–376, 2013.

[3] J. Chase, D. Niyato, and S. Chaisiri, "Bring-your-own-application (byoa): Optimal stochastic application migration in mobile cloud computing," in *2015 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2015, pp. 1–6.

[4] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the internet of things: A survey," *Communications Surveys & Tutorials, IEEE*, vol. 16, no. 1, pp. 414–454, 2014.

[5] B. Bangerter, S. Talwar, R. Arefi, and K. Stewart, "Networks and devices for the 5g era," *Communications Magazine, IEEE*, vol. 52, no. 2, pp. 90–96, 2014.

[6] G. K. Mostefaoui, J. Pasquier-Rocha, and P. Brezillon, "Context-aware computing: a guide for the pervasive computing community," in *Pervasive Services, 2004. ICPS 2004. IEEE/ACS International Conference on*. IEEE, 2004, pp. 39–48.

[7] M. J. Covington, W. Long, S. Srinivasan, A. K. Dev, M. Ahamad, and G. D. Abowd, "Securing context-aware applications using environment roles," in *Proceedings of the sixth ACM symposium on Access control models and technologies*. ACM, 2001, pp. 10–20.

[8] L. A. Gordon and M. P. Loeb, *Managing cybersecurity resources: a cost-benefit analysis*. McGraw-Hill New York, 2006, vol. 1.

[9] A. Shapiro, D. Dentcheva *et al.*, *Lectures on stochastic programming: modeling and theory*. SIAM, 2014, vol. 16.

[10] M. J. Covington, P. Fogla, Z. Zhan, and M. Ahamad, "A context-aware security architecture for emerging applications," in *Computer Security Applications Conference, 2002. Proceedings. 18th Annual*. IEEE, 2002, pp. 249–258.

[11] C. J. Wullems, M. H. Looi, and A. J. Clark, "Towards context-aware security: An authorization architecture for intranet environments," 2004.

[12] S. Preda, F. Cuppens, N. Cuppens-Boulahia, J. G. Alfaro, L. Toutain, and Y. Elrakaiby, "Semantic context aware security policy deployment," in *Proceedings of the 4th International Symposium on Information, Computer, and Communications Security*. ACM, 2009, pp. 251–261.

[13] D. Kulkarni and A. Tripathi, "Context-aware role-based access control in pervasive computing systems," in *Proceedings of the 13th ACM symposium on Access control models and technologies*. ACM, 2008, pp. 113–122.

[14] S. Schefer-Wenzl and M. Strembeck, "Modeling context-aware rbac models for business processes in ubiquitous computing environments," in *Mobile, Ubiquitous, and Intelligent Computing (MUSIC), 2012 Third FTRA International Conference on*. IEEE, 2012, pp. 126–131.

[15] M. Miettinen, S. Heuser, W. Kronz, A.-R. Sadeghi, and N. Asokan, "ConXsense: automated context classification for context-aware access control," in *Proceedings of the 9th ACM symposium on Information, computer and communications security*. ACM, 2014, pp. 293–304.

[16] P. Dhawan, "Performance comparison: Security design choices," *Microsoft Developer Network, Tech. Rep*, 2002.

[17] S. A. Butler, "Security attribute evaluation method: a cost-benefit approach," in *Proceedings of the 24th international conference on Software engineering*. ACM, 2002, pp. 232–240.

[18] W. Zeng and M.-Y. Chow, "A trade-off model for performance and security in secured networked control systems," in *Industrial Electronics (ISIE), 2011 IEEE International Symposium on*. IEEE, 2011, pp. 1997–2002.

[19] J. Al-Muhtadi, A. Ranganathan, R. Campbell, and M. D. Mickunas, "Cerberus: a context-aware security scheme for smart spaces," in *Pervasive Computing and Communications, 2003.(PerCom 2003). Proceedings of the First IEEE International Conference on*. IEEE, 2003, pp. 489–496.

[20] E. L. Lawler and D. E. Wood, "Branch-and-bound methods: A survey," *Operations research*, vol. 14, no. 4, pp. 699–719, 1966.

[21] GLPK (GNU Linear Programming Kit). Accessed: 2016-04-12. [Online]. Available: https://www.gnu.org/software/glpk/

[22] NEOS Solvers). Accessed: 2016-04-12. [Online]. Available: https://neos-server.org/neos/solvers/

[23] GAMS - A User's Guide. Accessed: 2016-04-12. [Online]. Available: https://www.gams.com/help/topic/gams.doc/userguides/GAMSUsersGuide.pdf

[24] Y. Baruch, "Teleworking: benefits and pitfalls as perceived by professionals and managers," *New Technology, Work and Employment*, vol. 15, no. 1, pp. 34–49, 2000.

[25] I. Kotuliak, P. Rybár, and P. Truchly, "Performance comparison of IPsec and TLS based VPN technologies," in *Emerging eLearning Technologies and Applications (ICETA), 2011 9th International Conference on*. IEEE, 2011, pp. 217–221.

[26] M. D. McKay, R. J. Beckman, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 42, no. 1, pp. 55–61, 2000.