# CRAC: Confidentiality Risk Assessment and IT-Infrastructure Comparison

Ayşe Moralı*, Emmanuele Zambon*, Sandro Etalle* † and Roel Wieringa*
*University of Twente
Email: {ayse.morali, emmanuele.zambon, sandro.etalle, roel.wieringa} (at) utwente.nl
†Eindhoven Technical University
Email: s.etalle (at) tue.nl

***Abstract*—Confidentiality is a critical aspect in todays Risk Assessment (RA) practices for many industrial organizations. Assessing confidentiality risks is challenging and the result of a confidentiality RA is still largely based on the subjective opinion of the risk assessor(s). The presence of cross-organization cooperations (e.g. outsourcing), makes a confidentiality RA even more challenging because there are additional threat agents to take into account (e.g. an outsourcers employee). In this paper we present CRAC, an IT infrastructure-based method for assessing and comparing confidentiality risks of IT based collaborations. The method determines confidentiality risks by taking into account the effects of the leakage of confidential information (e.g. industrial secrets and user credentials), and the paths that may be followed by different attackers (e.g. insider, outsider and outsourcer). We also show how the CRAC-method can be applied in practice and we evaluate its effectiveness by applying it to a real-world outsourcing case. *Index Terms*— onfidentiality, Risk Assessment, IT-Infrastructureonfidentiality, Risk Assessment, IT-InfrastructureC**

## I. Introduction

Nowadays, most data exchange within and across organizations boundaries takes place electronically. Exchanged data often contains confidential information, e.g. employee records, client information, and financial data. Collection, storage and use of information assets are usually issue to privacy and legislations (e.g. BASEL [3], Directive 95/46/EC [1], HIPAA [7], PIPEDA [30] and SOX [35]). Accordingly, loss of confidential information often results in economical and personal damage, both for the business and for the data owner (see e.g. [13], [22], [25]).

Good security (on the other hand) is also costly and even the best and most expensive countermeasures cannot mitigate all possible confidentiality incidents. This is mainly because of the impossibility of monitoring *all* confidentiality breaches. Therefore, the goal of security officers is to strike the right balance among security, budget, and system usability. To achieve this, they typically refer to well-established standards and best practices such as COBIT [10], ISO 27002 [17] and NIST 800-30 [4].

Assessing IT (confidentiality) risks becomes particularly challenging in the presence of cross-organizational cooperations, e.g. IT outsourcing or managed cloud computing. As part of the cooperation, organizations typically connect together their IT-infrastructures and they grant access rights to each other's (confidential) information to each other's employees. This process establishes a so-called *IT-enabled network of organizations* cooperating with each other, each with different roles and with different (and often conflicting) goals. IT-enabled networks of organizations increase the complexity of an IT-confidentiality risk assessment because one has to deal with a more complex IT-infrastructure and with an extended set of potential threats. For example, in a standard scenario threats can originate from within the organization (insider) or from outside the organization (outsider). However, in an outsourcing scenario further threat agents can originate from the outsourcing-provider. For example, an employee of the outsourcer have some privilege on the information of the outsourcing-client but the security policies of the organization does not apply to him.

To make informed decisions on the (security) design of its IT-infrastructure, an organization needs to fully understand the confidentiality risks that arise when there is a collaboration with other organizations. How and where confidential data is stored has a big impact on the security of the IT-infrastructure. Decision makers need to be able to assess and compare different solutions about the design of IT-infrastructures based on the confidentiality risks. This can only be achieved if risks are consistently assessed for each considered solution. However, the result of typical risk analysis methods cannot be consistently compared with each other if they were carried out by different people. This happens because they are mostly based either on subjective opinion of the different risk assessor(s) or on event histories. Confidentiality risks are even harder to assess in an inter-subjective (independent of personal judgement) way, because of their non-functional nature and the lack of logs about past incidents.

To solve these problems, in this paper we present the Confidentiality Risk Assessment and Comparison (CRAC) method. With the CRAC-method one can assess confidentiality risks by taking into account both how information assets flow in the underlying IT architecture and the different paths attackers can use to find their way in the architecture. We use *information flow* [28] and a customized version of *attack paths* [34] to elicit necessary information on confidentiality

This research is supported by the research program Sentinels (http://www.sentinels.nl). Sentinels is being financed by Technology Foundation STW, the Netherlands Organization for Scientific Research (NWO), and the Dutch Ministry of Economic Affairs.

aspects. We use information flow to analyze where and "how much" critical information is located in the system, and attack paths to model how attackers with different profiles may be able to reach this information.

The main contribution of CRAC to confidentiality risk assessment is that it support decision providers by allowing them to *compare* the confidentiality risks of alternative IT-infrastructure design solutions. CRAC is meant to be employed for infrastructures used by IT-enabled network of organizations, in such a way that it reduces the subjectivity of the risk assessment results.

We show the feasibility of the CRAC method by applying it to a real-world case and by evaluating its subjectivity, practicality and precision based on the success criteria that we derive from the case-study stakeholders.

CRAC improves and extends our earlier work, DCRA [27], for confidentiality risk assessments. Please refer to the related works for a description of the extensions and improvements over DCRA.

The structure of the paper is as follows: in Section II we describe the industrial case, in Section III we present the CRAC method and we show how to use it by referring to the industrial case, in Section IV we evaluate the CRAC method with respect to the criteria of the industrial case stakeholders. Finally, in Section V we present the related work and in Section VI we draw our conclusions and future work.

## II. Industrial Context

In this section we describe the industrial case we will use both to present the CRAC-method and to evaluate it. We present the organizations involved, the IT-infrastructure of the system to be risk assessment, the stakeholders of these organizations and their goals.

### A. Case Description

A large multinational electronics manufacturing company is outsourcing the management of its authentication and authorization system to a multinational IT service provider. From here on we will refer to the electronics company as *the Company*, the outsource supplier as *the Outsourcer* and the authentication and authorization service to be delivered by the Outsourcer as *the System*. The System is used by the Company's employees to access the Company's data and services, and by the employees of the Outsourcer for configuring, monitoring and maintaining the system (from now on we refer to this as the Managed Services).

The Outsourcer proposed to replace the IT-infrastructure that the System is currently built on with an alternative IT-infrastructure. The company needs to know if the new IT-infrastructure is at least as secure as the IT-infrastructure that is currently in use. However, there are several confidentiality related, architectural and organizational trade-offs between the two infrastructures. Therefore, the Company can not compare their relative confidentiality risks intuitively or with a subjective risk assessment method. Our goal here is to analyze and compare the two infrastructures w.r.t. the confidentiality risks.

The above mentioned two alternative solutions are illustrated (in a simplified version) in Fig. 1, which also shows the access paths that can be followed by the Outsourcer's employees to get access to the Company's systems.

The first infrastructure (Alternative 1 in Fig. 1) is the one that is currently in use. It comprises a single access path used by all employees of the Outsourcer for reaching the Managed Services. All access attempts to the Managed Services are monitored by the *session directory services*. The *terminal server* makes applications available to the Outsourcer's employees in a terminal session. The *secure gateway (1)*, which is installed on the third party gateway (TPG) of the Company, is responsible of authenticating the Outsourcer employees to the Managed Services. The *presentation server* is used by the (authorized) Outsourcer employees as an interface to manage applications.

The second infrastructure (Alternative 2 in Fig. 1) is the IT-infrastructure that is proposed by the Outsourcer. The first main difference with Alternative 1, is that Alternative 2 contains a second access path with two further IT-components (the *stepping stone portal* and the *stepping stone server*). These IT-components allow a special group of the Outsourcer employees to access the Managed Services in emergency cases. Unlike the first path, which uses two-factor authentication (i.e., a secure ID plus a password), the second path uses IP based authentication. The second main difference in Alternative 2 is that the secure gateway (secure gateway (2)) is located in the intranet of the Company, and not in the demilitarized zone as in Alternative 1.

The information classification scheme adopted by the Company is made of three levels: private, highly confidential and company confidential. For instance, user credentials contain personal information (e.g. social security numbers) and are therefore classified as *private*. Information such as business information and access control lists are considered critical for the business and are therefore classified as *highly confidential*. IT-architectural documents on the other hand may cause a loss only if 3rd parties access them and are therefore classified as *company confidential*. Confidentiality levels determine who is authorized to access the data. In this case, user credentials should be accessed only by the data owner, IT-architectural documents should be accessed only by the employees of the Company and the Outsourcer, and business information and access control lists should be accessed only by those employees of the Company and of the Outsourcer who need to access it to fulfill their duties and should be unaccessible to others.

### B. Stakeholders and Their Goals

The stakeholders involved are the *Global Infrastructure Board* (GIB) and the *Risk, Performance & Compliance Unit* (RMC). They are independent business units of the Company.

GIB is the owner of the System. It requests RMC to assess the risks of all new or updated IT systems. In this particular case GI wants to know which of the two IT-infrastructures is more robust to confidentiality breaches. GIB has determined

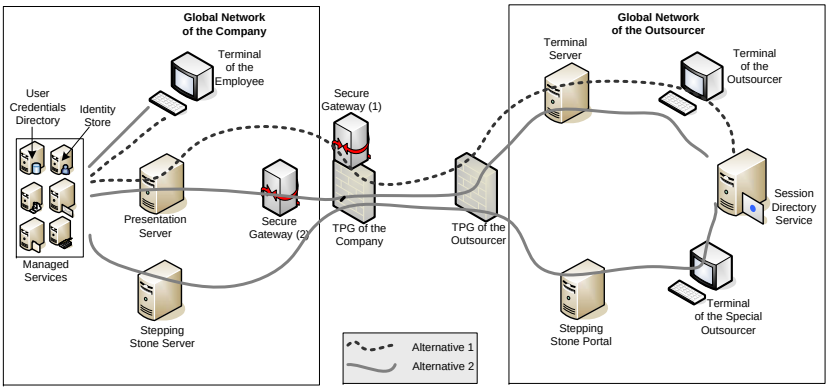


Fig. 1. Alternative access paths to Managed Services using two alternative IT-infrastructures that are under investigation.

the business impact of confidentiality breaches, which in this case depends on (a) the criticality of information asset, (b) the number of instances that get disclosed, and (c) to whom they get disclosed to. Information assets consists of instances. For example, if client data is an information asset, then each client record is an instance of this asset.

RMC is responsible of assessing the risks and compliance requirements of the IT systems. RMC uses a check-lists based risk assessment method that has been developed according to the requirements of the Company. From here on, we will call this method the RA method. The RA method consists of two main parts: (1) Business Impact Analysis; and (2) Threat and Vulnerability Analysis.

According to RMC, although the RA method is practical, the results it delivers rely too much on the subjective opinion of the risk assessors. Furthermore, the RA method assesses risks based on a list of standard threats which are not linked to any component of the IT-infrastructure under assessment. For these reasons, the results RA delivers cannot be used for comparing alternative IT-infrastructures. RMC then asked us to improve its risk assessment method in the following ways: (1) make the risk assessment process less subjective; and (2) change it in such a way that the assessment result allows comparing different IT-infrastructures.

## III. The CRAC-method

The CRAC-method assesses the *ease* of a person accessing critical information intentionally or by accident. For this, it analyzes the components that form the IT-infrastructure (e.g. applications, operating systems, and network segments) and their vulnerabilities that are relevant for an information security risk analysis. To note that, the CRAC-method adopts IT security related constructs from ISO/IEC13335 [18]. Our analytic approach to confidentiality risk assessment is motivated by the impossibility of monitoring *all* unauthorized access to information and thus determining the likelihood of confidentiality breaches using event history.

The CRAC-method is built on two ideas. (1) Information is a logical asset, so it does not stay at one place only but can flow from one component to the other one. For instance, it could flow to an IT-component because a user copies it there. (2) An attacker may penetrate into a system through different components and follow different attack paths. For instance, a hacker may seek ways of revealing the information available on a server over the internet connection, whereas an outsourcer may have physical access to the server and try to read that information directly from the hard disk of the server.

CRAC analysis consists of four steps.

Step 0: Collecting the basic information from available documentation and from interviews with the stakeholders.

Step 1: Analyzing the paths information can flow through an architecture and determine impact.

Step 2: Identifying attack paths that may be followed by threat agents and determine reachability.

Step 3: Combining the results of Step 1 and Step 2 to identify weak spots and evaluate risks.

In what follows, we present these steps closer and illustrate how we apply them to the System.

### A. Step 0: Collecting Basic Information

In this step we collect the following information:

- the list of information assets present on the system, their confidentiality level and homogeneity property;
- the list of IT-components that form the IT-architecture of the system and how components are related to each other;
- the list of vulnerabilities; and
- the list of possible threat agents.

We use the following basic notation. $L$ is the ordered set of all the confidentiality levels (e.g. {top-secret, confidential, public}); $N$ is the ordered set of all the information asset quantity classes (e.g. {all, single, none}); $H$ is the ordered set of all the information asset homogeneity classes (i.e. { homogeneous, non-homogeneous}); $I$ is the ordered set of all the qualitative single impact values (e.g. { very-high, high, medium, low, null }); $TI$ is the ordered set of all the qualitative total impact values (e.g. { very-high, high, medium, low, null }); $P$ is the ordered set of all the qualitative likelihood values (e.g. { very-likely, likely, unlikely }).

We call information assets the "semantic components of an information system that are required for an organization to conduct its mission or business" [21], e.g. customer information and user credentials. $A$ is the set of information assets we consider (e.g. customer data, passwords). To each information asset $a \in A$ we associate a confidentiality level $l : A \rightarrow L$. $C$ is the set of IT components (e.g. computers, applications, rooms) where the information assets may be present. A component $c$ can contain multiple instances of a given information asset $a$ (i.e. multiple data instances) at a time. The mapping $n : A \times C \rightarrow N$ is a qualitative estimate of the number of instances of $a$ that can be retrieved from component $c$ at once. An information asset $a$ is *homogeneous* if the damage due to its disclosure can be considered proportional to the number of its instances that get disclosed. For example "social security numbers" are homogeneous, since the damage due to the loss of one hundred social security numbers is larger than the damage due to the loss of a single social security number. Conversely, an information asset is *non-homogeneous* if the damage due to the disclosure of one instance is as big as the damage of the disclosure of all instances. For example, if credentials of one user get disclosed, the damage to the company is the same as if credentials of 100 users with equal access writes would be disclosed. To model this we use the mapping $h : A \rightarrow H$.

**Running example - Part 1.** *In our example, following the information classification scheme that is adopted by the Company,* $L = \{$ high, medium, low$\}$ *and* $N = \{\text{none}, \text{single}, \text{all}\}$. *The information assets that we take into account are User Credentials and Business Information. Employees of the outsourcer use User Credentials to access the managed services.* $l($User Credentials$) =$ high *and* $h($User Credentials$) =$ non-homogeneous*. User Credentials are available (among others) on the User Credentials Directory IT-component of the system. For instance, all of the User Credentials instances in User Credentials Directory can be retrieved at once. However, not all information assets can be retrieved from all components. Business Information is data related to the business of the Company, which is stored and processed in its information systems.* $l(\text{BusinessInformation}) =$ medium *and* $h(\text{BusinessInformation}) =$ homogeneous*.*

**Definition III.1.** ***(architecture graph)*** *An architecture graph* $\text{arch} = \langle C, E \rangle$ *is a directed graph in which* $C$ *is the set of vertices representing IT components and* $E$ *is the set of edges* $E \subseteq C \times C$. $(c_1, c_2) \in E$ *iff there exists a direct connection between* $c_1$ *and* $c_2$ *such that data can flow from* $c_1$ *to* $c_2$.

An architecture graph is a representation of the IT infrastructure under exam which we use to determine how an information asset $a$ can flow from one component to others. We build an architecture graph based on the available documentation about the System and on interview sessions with the stakeholders.

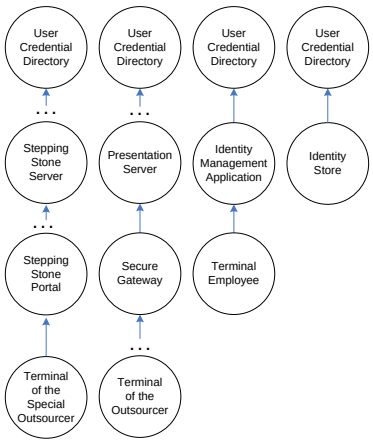


Fig. 2. $FP_{UserCredentials}$ in Infrastructure 2.

### B. Step 1: Analyzing information flow

In this step we first analyze the logical and physical connections among components based on an infrastructure. Then, we determine the impact of each component by considering the information assets that may flow to them. If there is a possibility for an information asset to flow to a component then we assume that that information is present on that component. We furthermore assume that information flows in a predictable way. Thus the information flow analysis can recognize all possible paths according to policies and documented properties of the components.

In more detail, to model information flow we build for each information asset $a$ a set of *flow paths*. A flow path is a path in the architecture graph which starts with a component where $a$ is stored. The nodes of a flow path represent the IT-components in which $a$ can be accessed by an attacker. We represent a flow path by an ordered list $fp = [c_1, \ldots, c_n]$ where $c_1 \ldots c_n \in C$ with no repeated occurrences of $c_i$. We call $FP_a = \{fp_1, \ldots, fp_m\}$ the set of flow paths of $a$. We use the maximum number of instances that may flow to a component $c$ from its connected components to determine the number of instances an attacker can disclose by gaining access to $c$.

**Running example - Part 2.** *Fig. 2 illustrates* $\text{FP}_{\text{UserCredentials}}$ *in Infrastructure 2. User Credential Directory is the component where User Credentials reside. For the sake of presentation we included in the paths only the components which are also listed in Fig. 1. The remaining ones are represented by dots "...". We observe four information flow paths. In the leftmost path credentials flow from the User Credential Directory to the Terminal Of The Special Outsourcer, which is the terminal used by the employees of the Outsourcer with special status. In the second path, credentials flow towards the Terminal Of The Outsourcer. In the third path credentials flow towards the Identity Management Application, which the employees of the Company access remotely. In the last path credentials are synchronized between the User Credential Directory and the Identity Store.*

TABLE I
BEHAVIOR OF THE $\odot$ OPERATOR.

| $\odot$ | all | single | none |
|---|---|---|---|
| high | very-high | high | null |
| medium | high | medium | null |
| low | medium | low | null |

TABLE II
BEHAVIOR OF THE $\oplus$ OPERATOR.

| $\oplus$ | very-high | high | medium | low | null |
|---|---|---|---|---|---|
| high | very-high | very-high | high | high | high |
| medium | very-high | high | high | medium | medium |
| low | very-high | high | medium | medium | low |
| null | very-high | high | medium | low | null |

After constructing the flow paths we determine for each information asset $a$, for each component $c$ and for each flow path *fp* $\in FP_a$, the number of instances of $a$ that are present in $c$ according to *fp* using the function ($n : A \times C \times FP_a \rightarrow N$). If we call *index*($c$,*fp*) the index of $c$ inside *fp*, then $n(a, c,fp) = min_{i \leq index(c)} n(a, c_i)$, where $c_i \in$ *fp*.

Then, we determine for each $a$, $c$ and *fp*, the impact of the disclosure of the instances of $a$ which are present in $c$ according to *fp* using the function (*fp-imp* : $A \times C \times FP_a \rightarrow I$). To this end, we take into consideration the number of instances of an information asset which can be extracted to a component at once, as well as their confidentiality level and homogeneity. Recall that $l(a)$ and $h(a)$ are respectively the confidentiality level and the homogeneity property of information asset $a$ and that $n(a, c)$ is the maximum number of instances of $a$ that can be extracted from $c$.

$$\textit{fp-imp}(a,c,fp) = \begin{cases} l(a) \odot n(a,c,fp) & \text{, if } h(a) = \textit{homogeneous;} \\ l(a) \odot \textit{all} & \text{, if } n(a,c,fp) \neq \textit{none;} \\ \textit{null} & \text{, else.} \end{cases}$$

Where $\odot : L \times N \rightarrow I$ is a monotone composition operator for ordinal scale qualitative values in $L$ and $N$. $\odot$ should be agreed on with the risk assessment stakeholders to guarantee that everybody understands how values are composed.

Now, we are able to compute the *impact* of the disclosure of each information asset $a$ on each component $c$, *imp* : $A \times C \rightarrow I$, as the maximum impact with respect to all the possible flow paths in $FP_a$. We determine impact according to the following equation:

$$imp(c, a) = \textit{max}_{fp \in FP_a} \textit{fp-imp}(a, c, fp) \quad (1)$$

In practice quantitative values are difficult to obtain. Consequently, the CRAC method determines the impact with partially ordered qualitative values, as it is commonly done in many risk assessment methods. These values belong to ordinal scale class. According to the measurement theory the ordinal scale preserves order and is monotonic increasing. Since the $\odot$ operator satisfies these properties we say that it is theoretically valid. However, if quantitative values are available then the $\odot$ operator behaves as a multiplication.

**Running example - Part 3.** *We agreed with the Company on the binary merge operator $\odot$ on* L *and* N *as shown in Tab. I.*

*Let us now determine the impact of the disclosure of User Credentials and Business Information on the IT components of the architecture. Because User Credentials is a* non-homogenous *asset, its confidentiality level is* high *and in all the four flow paths* $n(\text{UserCredentials}, c, \text{fp}) \neq \text{null}$*, then the impact on all components in the architecture on which User Credentials flow is* high*. Business Information is* homogeneous *and its confidentiality level is* medium*. The amount of instances of it flowing to Secure Gateway is* all*. Accordingly, the impact of Secure Gateway is* high*. However, the number of instances flowing from Secure Gateway to its children is* single*. Consequently, the impact on the children of Secure Gateway is* medium*.*

Summarizing, in this step we have built a set of information flow paths: one for each architecture graph, information asset, component the assets resides on and graph path. Then, we determined the impact of the leakage of an information asset on each component in the architecture. By merging the impact values relative to a specific component with respect to the different information assets we obtain the *total impact* of the component.

The total impact for component $c$ is the impact of the disclosure of all confidential information assets available on $c$. If $c$ contains only one information asset $a$, then $imp(c) = imp(c, a)$. On the other hand, if $c$ contains two or more assets (say $a_1$ and $a_2$) then we "add" $imp(c, a_1)$ and $imp(c, a_2)$. To this end we use the monotone operator $\oplus : TI \times I \rightarrow TI$ defined in Tab. II (as for $\odot$, $\oplus$ shall be agreed on with the stakeholders). More formally, the total impact of $c$ is given in the following definition.

**Definition III.2.** ***(Total impact)***

*Given a component c and a set of information assets A, we call* total impact of *c expressed by the function* imp : $C \rightarrow TI$ *the cumulative loss caused by the disclosure of all confidential information available on c.* $imp(c)$ *is given by the following equation:*

$$imp(c) = \oplus_{a \in A} imp(c, a) \quad (2)$$

**Running example - Part 4.** *We assume that on the component Terminal Of The Special Outsourcer only the information assets User Credentials and Business Information are available. Now, recall form Running example 3 that* imp(Terminal Of The Special Outsourcer, User Credentials) = high *and* imp(Terminal Of The Special Outsourcer, Business Information) = low. *By applying the $\oplus$ operator we obtain the total impact:* high*.*

### C. Step 2: Constructing APGs

In the second step of the CRAC-method we build the Attack Propagation Paths (APPs) which describe how different threat agents might penetrate into the IT infrastructure. Then,

we determine the likelihood of a threat agent to access the information available on each IT-component.

In CRAC, a threat agent $t$ is someone who, intentionally or by mistake, causes a confidentiality breach that may result in the disclosure of the information assets available in a component. We call $T$ the set of all threat agents in the System. For assessing the risks of a system the threat agents need to be enumerated in agreement with the RA stakeholders.

**Running example - Part 5.** *For assessing the risks of the System we distinguish three threat agents:* $T = \{Employee, Outsider, Outsourcer\}$.

Vulnerabilities are weaknesses of the components which allow attacks propagation. We call $V$ the set of all the vulnerabilities in the System. We represent the fact that $v$ is a weakness of $c$ by means of the mapping $w : V \times C \rightarrow \{true, false\}$.

Furthermore, we represent the likelihood that a threat agent $t$ exploits a vulnerability $v$ to compromise an IT component $c$ by the mapping $p : T \times V \times C \rightarrow P$.

To model confidentiality breaches we build for each threat agent $t$ a set of APPs. APPs are based on the concept of attack trees [34]. Unlike in classic attack trees, the nodes of an APP do not represent possible actions constituting an attack, but they are the IT-components that an attacker compromises during an attack. We say an attack can propagate if two components are physically or logically connected to each other (i.e., if they are connected in the architecture graph). We build each APP in two steps. We first add a node ($c_1$) to the APG for each IT-component that can be directly reached by a threat agent (for external threat agents we can add a special fictitious IT-component "the internet"). Second, we iteratively add new nodes and edges as follows: if node $c_1$ is in the APP and $c_1$ is connected to the IT-component $c_2$ in the architecture graph, then we add $c_2$ to the APP. We repeat this operation until the component we just inserted is not connected to any other component. Similarly to information flow paths, we represent an APP by an ordered list $app = [c_1, \ldots, c_n]$ where $c_1 \ldots c_n \in C$ with no repeated occurrences of $c_i$. We call $APP_t = \{app_1, \ldots, app_n\}$ the set of APPs a threat agent $t$ can follow.

**Running example - Part 6.** *Fig. 3 illustrates the APPs that an employee of the Outsourcer may follow to access User Credentials on the User Credential Directory, in the scenario of IT-infrastructure 2. The Outsourcer employee may access User Credentials Directory via the Terminal Of The Special Outsourcer or via the Terminal Of The Outsourcer. We iteratively included components that are physically or logically connected to Terminal Of The Special Outsourcer and Terminal Of The Outsourcer until all connected components of the System are present in the APG.*

After constructing APPs we determine the likelihood of each threat agent $t$ compromising each IT component $c$ by following each attack propagation path in $APP_t$. In doing so we need to take into account two properties. (1) Each IT component may have more than one vulnerability that $t$ can exploit, in this case we assume the threat agent will exploit the vulnerability with the highest associated likelihood. (2) The threat agent needs to compromise other components in order to compromise $c$, in this case we assume the likelihood of compromising $c$ is the lowest likelihood of the list (i.e. the hardest step).

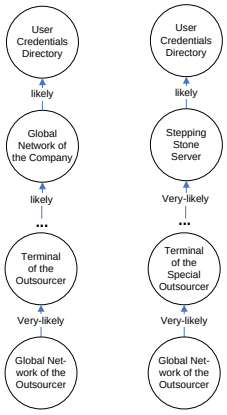


Fig. 3. APPs followed by the Outsourcer to access User Credentials according to IT-infrastructure 2

**Definition III.3.** ***(Attack Propagation Likelihood)*** *Given a component $c$, a threat agent $t$, a set of vulnerabilities $V$, an attack path* $\text{app} \in \text{APG}_t$*, and* $\text{index}(c, \text{app})$ *the index of $c$ in the ordered list* app*, we call $p : T \times C \times \text{APP}_t \rightarrow P$ the likelihood of $t$ compromising $c$ by following* app *where*

$$p(t, c, \text{app}) = \min_{i<\text{index}(c,\text{app})} \max_{v \in \{v | v \in V, w(v,c)=\text{true}\}} p(t, v, c_i) \quad (3)$$

**Running example - Part 7.** *To enumerate vulnerabilities we refer to the threat and vulnerability list the Company uses in their RA-method. NIST SP 800-30 [36] indicates that the likelihood of a threat agent exercising a system vulnerability depends on his competencies and conditions. Competencies are the problem-solving capabilities of the threat agent, e.g. hacking skills and system knowledge; whereas conditions are its owned environmental rights, e.g. physical access. Several researchers in the security field (see for instance [14], [15], [26], [37]) follow this indication and refer to competencies and conditions by determining the likelihood of an incident. In agreement with the stakeholders, here we adopt three competencies and conditions: Physical Access, System Knowledge and Hacking Skills. We determine $p(t, v, c)$ by cross-checking if the competencies and conditions of a threat agent $t$ ($x_t$) to exploit a vulnerability $v$ given the required competencies and conditions ($x_v$), and if $v$ is a weakness of $c$. Accordingly, $p(t, v, c)$ is given by the following equation:*

$$p(t, v, c) = \begin{cases} \text{unlikely}, & \text{if } x_t \bigcap x_v = \emptyset \text{ OR } w(v, c) = \text{False} \\ \text{very-likely}, & \text{if } x_t \supseteq x_v, \\ \text{likely}, & \text{if } x_t \subset x_v \text{ and } x_t \bigcap x_v \neq \emptyset, \end{cases} \quad (4)$$

*If the set of available competencies and conditions of a threat agent is equal to (or exceeds) the set of competencies and conditions required to exploit a vulnerability, then the threat agent is* very-likely *to exploit the vulnerability. If the threat agent has only a subset of the required competencies and conditions, then we say that she is* likely *to exploit the vulnerability. In case there are no common competencies and conditions between the required and available sets of competencies and conditions, then she is* unlikely *to exploit the vulnerability.*

*In the System, the employee of the Outsourcer has both Physical Access and System Knowledge. The vulnerabilities associated with the component Terminal Of The Special Outsourcer are Security Unawareness and Weak Authentication. The attack propagation likelihood of Terminal Of The Special Outsourcer in the APP built for the Outsourcer is* very-likely*, e.g. the likelihood of the Outsourcer exploiting Security Unawareness (*very-likely*) is higher than the likelihood of the Outsourcer exploiting Weak Authentication (*un-likely*). Furthermore, the likelihood of the Outsourcer employee penetrating into Global Network Of The Company is* likely*. We determine this likelihood as follows. The vulnerabilities of Global Network Of The Company are Virtual Security Zones, Lack Of Monitoring and Weak Authentication Mechanisms. These vulnerabilities require System Knowledge and Hacking Skills or only Hacking Skills. This leads to* likely *likelihood for this attack step. Furthermore, each attack path that leads to Global Network Of The Company contains more than 1 attack propagation and the lowest likelihood in each attack path is* likely.

Summarizing, in this step we have build APGs (one for each threat agent and alternative IT-infrastructure) and determine the attack propagation likelihood of each component in the APGs. Now, by merging the attack propagation likelihoods for a component with respect to the different APPs, we determine its *reachability level*.

**Definition III.4.** ***(Reachability Level)*** *Given a component c, the reachability level of c* reach : $C \rightarrow P$ *is given by the highest attack propagation likelihood with respect to all the possible attach propagation paths, as follows:*

$$\text{reach}(c) = max_{t \in T}(max_{\text{app} \in \text{APP}_t}(p(t, c, \text{app}))) \quad (5)$$

**Running example - Part 8.** *Terminal Of The Special Outsourcer is on the APG of the Outsourcer and Outsider. Its attack propagation likelihood, according to the APG of the Outsourcer is* very-likely*, whereas it is* likely *according to the APG of the Outsider. According to III.4 we say that the reachability level of* $reach(TerminalOfTheSpecialOutsourcer)$ *is* very-likely.

### D. Step 3: Risk Calculation and Comparison

In this step we combine the output of steps 1 and 2 to identify the weak spots in the system and compare the security of alternative IT-infrastructures. We identify the weak spots, which are confidentiality-critical IT-components, based on their confidentiality *risk*.

**Definition III.5.** ***(Risk)*** *Given an IT-component c with total impact* $imp(c)$ *and reachability level* $reach(c)$*, the risk of c is the pair* $risk(c) = \langle imp(c), reach(c) \rangle$.

**Running example - Part 9.** *In step 1 we computed the total impact value of TerminalOfTheSpecialOutsourcer (*high*), in step 2 we computed its reachability level (*high*). Therefore, the risk of TerminalOfTheSpecialOutsourcer for IT-infrastructure 2 is* $\langle high, high \rangle$.

After determining the risk of all components for alternative IT-infrastructures we sort them. For sorting we first group together all IT-components with the same total impact in one of the architectures and then sort the components in each group according to their reachability level in a descending manner. The components with the highest total impact and reachability level are the most critical ones. Then we determine which infrastructure is more robust w.r.t. confidentiality risks by counting the number of assets on the different architectures with the same risk level.

**Running example - Part 10.** *The risk of the components we discussed in earlier are presented in a sorted manner in Tab. III. In infrastructure 2 the risk of the User Credential Directory is* $\langle high, medium \rangle$*. Although the total impact of the Terminal Of The Special Outsourcer and the User Credential Directory are the same, the risk of the Terminal Of The Special Outsourcer is higher than the risk of the User Credential Directory. In other words, Terminal Of The Special Outsourcer is a more critical component than User Credential Directory. This is due to the fact that it is less likely that an attacker targets information available on the User Credential Directory.*

*Comparing the risk of the IT-components of IT-infrastructure 1 and 2 we see that 1 is more robust than 2. In particular, the risk of Stepping Stone Server and Stepping Stone Portal are* $\langle very\text{-}high, high \rangle$ *and they are present only in IT-infrastructure 2. Furthermore, in IT-infrastructure 2 three components have a higher reachability level than in IT-infrastructure 1.*

For more complex systems presenting risk in a table may be unsuitable. For that cases one can calculate the percentage of components with the same total impact and reachability level, and present the results in a (smaller) matrix.

## IV. Evaluation

In this section we discuss how effective the CRAC method has been in our case study on bringing the stakeholders closer to their goals. Here, we follow the evaluation approach proposed by Wieringa [38] for technical solutions.

### A. Solution Criteria

According to the stakeholders a successful confidentiality risk method should satisfy the following criteria (explained below):

TABLE III
PART OF THE RISK PRESENTATION TABLE.

| Components | Infrastructure 1 | | Infrastructure 2 | |
|---|---|---|---|---|
| | Total impact | Reachability level | Total impact | Reachability level |
| Terminal Of The Outsourcer | very-high | high | very-high | high |
| TPG Of The Outsourcer | very-high | medium | very-high | high |
| Secure Gateway | very-high | medium | very-high | medium |
| Managed Services | very-high | medium | very-high | medium |
| Identity Store | very-high | medium | very-high | medium |
| TPG Of The Company | very-high | medium | very-high | medium |
| Terminal Server | very-high | medium | very-high | high |
| Identity Management Application | very-high | low | very-high | low |
| Presentation Server | very-high | low | very-high | low |
| User Credentials Directory | high | medium | very-high | high |
| Terminal Of Employee | high | low | high | low |
| Global Network Of The Company | null | medium | null | medium |
| Terminal Of The Special Outsourcer | null | medium | very-high | high |

(C1) The method should allow a detailed representation of risk;
(C2) The method should be practical to implement; and
(C3) The method should deliver less subjective results than the currently employed check-list based risk assessment method.

We measure how well our solution scores w.r.t. these criteria based on the following measures (explained below):

(M1) the number of risk-related aspects the method is able to represent;
(M2) the percentage of optional risk-related aspects;
(M3) the percentage of risk-related aspects that may be used at different granularities; and
(M4) the percentage of inter-subjective aspects.

(C1) indicates that a good risk assessment method should allow the risk assessor to represent the complexity of the target of assessment in a detailed manner and is justified by the goal of RMC. We measure (C1) with (M1) and (M3). (M1) expresses the number of confidentiality-related aspects a method is able to model (e.g. attacker profiles, attack propagation and the amount of instances that may get disclosed). From here on we call them *aspects*. [1] For this comparison we assume that all the aspects are equal weighted. We implicitly assume that the more aspects the method considers the more precisely it can assess risks. (M3) indicates the possibility of using the method with information at different detail levels. For instance, when considering threat agents, if there is only one type of threat agent (e.g. attacker) or many types of threat agents (e.g. insider, outsider and outsourcer). The accuracy of a risk assessment method (C1) often has a negative impact on the ease of its implementation (C2). The implementation effort of a method should ideally be adjustable to the criticality of the system to be assessed (RMC needs to assess the risks of low critical systems with lower effort than for high critical systems). We measure (C2) with (M2) and (M3), which express the flexibility of the method. This is a desirable feature in the case in which acquiring complete and detailed information is not possible because of limited resources. Here, flexibility can be described as (1) how well a risk assessment method can be adjusted to work at different detail levels without compromising accuracy and (2) how easy it is to refine or abstract the method in technical level. The goal of the Company is to compare the confidentiality risks of two alternative IT-infrastructures. This requires assessing the risks of these two IT-infrastructures separately and then comparing the assessment results. Different risk assessors must be able to work on the two assessments. Therefore, the method they use must be inter-subjective (C3). Since risk is defined as the combination of the likelihood of an incident and its consequences (impact) [20], and the subjectivity of assessment results depend on the subjectivity of the aspects that are used for determining the incident likelihood and impact, we measure the subjectivity of the risk assessment results with the percentage of non-subjective aspects (M4). The aspects that we consider as inter-subjective are: (1) documented facts (e.g. the IT-components determined based on IT-architectural drawings), (2) the knowledge shared among all the stakeholders (e.g. the list of vulnerabilities determined based on a publicly available vulnerability data base) and (3) any combination of the first two (e.g. the reachability value for each IT-component determined based on the available and required capabilities and conditions of threat agents and components).

[1] A complete list of aspect we identify and use in our comparison can be found in the appendix.

### B. Comparison

To complete the evaluation we now compare three risk assessment methods with respect to the success criteria we presented. The methods we consider are: (1) the CRAC method; (2) the Company's own RA method (which is a customized check-list based approach that follows ISO 27001 [16] and ISO 27005 [19]) and (3) the well known CRAMM method [6]. For this comparison we disregard the governance related concepts of CRAMM and check-list based approach, which are outside the scope of this paper. Tab. IV reports a summary of this comparison.

*M1::* Using the CRAMM method one is able to take into account almost twice as many aspects than with the check-list based approach and our CRAC-method. Some of

TABLE IV
COMPARISON OF THE CRAC-METHOD WITH THE CHECK-LIST BASES RISK ASSESSMENT METHODS AND THE CRAMM-METHOD.

| Measures | CRAC | Check-list | CRAMM |
|---|---|---|---|
| M1: number of aspects | 18 | 16 | 25 |
| M2: percentage of optional aspects | 16% | 44% | 4% |
| M3: percentage of aspects with different granularity levels | 44% | 25% | 36% |
| M4: percentage of non-subjective aspects | 78% | 56% | 72% |

the aspects that the CRAMM method takes into account (and CRAC does not) are the number of persons using the assets, threat level and potential impact scenarios. However, there are also aspects that CRAC considers and CRAMM doesn't. They are information homogeneity and volume. We conclude that, although the CRAMM method represents in total a higher number of aspects than the other two, it does not cover all the relevant risk-related aspects.

*M2::* The check-list method allows one to ignore almost half of the aspects it introduces, whereas the CRAC-method only allows one to ignore 16% of the aspects. However, analyzing the optional concepts in the check-list method we see that they are all used to determine the impact of an incident, e.g. type of data, type of users, sensitivity period. In CRAC the optional aspects affect both the impact and the likelihood, e.g. homogeneity, information flow, and competency and conditions. Furthermore, the CRAMM-method allows ignoring only "threat and vulnerability questions". These questions aim to analyze extend of vulnerabilities and threats. Therefore these questions allow an assessor to analyze the likelihood of incidents but not impact. Accordingly, we argue that the optional aspects in CRAC are more evenly distributed than in the check-list method and the CRAMM-method.

*M3::* The CRAC-method considers 19% more aspects than the check-list based approach and 8% more aspects than the CRAMM-method with adjustable granularity. For instance, if the target of assessment is critical then (differently from the check-list based approach and the CRAMM-method) the CRAC-method enables the assessor to determine the impact by differentiating among different volumes of instances flowing from one component to another. Consequently, with the CRAC-method the risk assessor can adjust the granularity of the impact determination depending on the criticality of the target of assessment. To note that the criticality of a system depends on the confidentiality levels of the information assets that the system contains. Since the CRAC-method has more such aspects with adjustable granularity than the other two, we conclude that it allows adjusting the precision of a confidentiality risk assessment to the criticality of the target of assessment better than the other two.

*M4::* Among the three methods, CRAC uses the highest percentage of non-subjective aspects. It is followed by the CRAMM-method, which uses only 6% less non-subjective aspects than CRAC. This happens because most of the information it uses is either generally well-documented or it must previously be agreed on by all stakeholders. Although, the CRAC-method considers almost the same number of aspects as the check-list method, there are almost twice as many aspects that can be adjusted to the desired granularity at which to carry out the risk assessment. Accordingly both the CRAC-method and the CRAMM-method represent confidentiality risks better than the companies check-list method.

With respect to the success criteria defined by the stakeholders, the CRAC-method satisfies (C1). Furthermore, we assume that ignoring some aspects can save a risk assessor more time than assessing all aspects in a coarse grained way. Therefore, we conclude that both the CRAC-Method is less practical than the check-list based approach. However, their detail levels of the CRAC-method can be adjusted more evenly according to the criticality of the target of assessment. Consequently, although the CRAC-method does not fully satisfy (C2) in the sense of practicality, it can be more gracefully adjusted for assessing confidentiality risk on systems with different granularity requirements. Finally, the CRAC-method delivers 22% less subjective results than the check-list based approach and therefore it satisfies (C3).

We believe that one of the reasons why we achieved such good results is that the CRAC method is specifically designed for assessing "confidentiality" risks, whereas the other methods aim to assess confidentiality, integrity and availability risk as a hole. Furthermore, we developed the CRAC-method with the success criteria defined by the stakeholders in our minds, whereas the CRAMM-method is not developed to serve the goals of the stakeholders in this case. Sofar, we could apply the CRAC method only to one case. Therefore, we don't know yet if these results are generalizable to other case.

## V. RELATED WORK

In this section we present other works related to (1) model-based risk assessments, (2) modeling information flow (for risk assessment) and (3) modeling attack paths.

Systematic risk assessment methods are carried out based on a model of the target of assessment target of assessment. Models differ according to the aspects they focus on (e.g. attack propagation) and to the techniques they use to deal with these aspects (e.g. constructing attack graphs).

Some well known risk assessment methodologies, such as CORAS [12], CRAMM [6] and OCTAVE [2], give detailed recommendations about which modeling techniques are more suitable for which step of a risk assessment. CRAMM is the UK Government's preferred Risk Analysis and Management Method. It analyzes the risks introduced by different threat agents and presents the results in relation to the supporting IT-architecture. It is furthermore supported by a commercial

available tool that assists among others for analysis of technical options, where the technical security and contingency issues associated with each option may need to be investigated or refined. CORAS is mainly concerned with eliciting and communicating risk related information from and to stakeholders at different domains, e.g. system architects and business owners. OCTAVE on the other hand presents a technology-neutral risk evaluation approach to bridge the gap between an organization's operational and IT requirements. The CRAC-method extends these risk assessment methodologies by modeling confidentiality risk at IT-infrastructure level: it links vulnerabilities to IT-components and determines reachability of these IT-components according to the profile of a threat agent. Differentiating between threat agents and considering the effects of IT-infrastructure on the risk is essential for assessing risk of systems on which networks of organizations are built.

In the literature we found a number of risk modeling frameworks (e.g. [11], [14], [29], [33]) in which likelihood is determined by taking into account different threat agents and properties of the system. Although this way of determining the likelihood instinctively leads to more detailed results, it also increases the complexity of the model.

In [27] we introduced the DCRA model. CRAC improves and extends DCRA for outsourced IT-system at industrial organizations. In such scenarios detailed information on confidentiality aspects (such as volume of information stored on each IT component) is not explicitly available. Therefore, the CRAC-method presents a more practical approach that systematically elicits information on confidentiality aspects of not very confidentiality critical systems. Here we consider the volume of information flowing and information flow paths. Furthermore, DCRA method does not consider attacker profiles and to whom the information gets disclosed. These concepts become especially critical at cross-organizational cooperations. CRAC addresses these concepts by extending DCRA method with the concept of threat agents at identifying attack paths and determining impact.

For confidentiality it is essential to model how information flows [3]. In the literature we find a number of approaches for modeling security with information flow graphs, e.g. [8], [28], [23]. These approaches are distinguished according to what the nodes of the information flow graph model and to the information flow criteria. To the best of our knowledge, only Chivers [8] uses information flow trees for analyzing risk. Nodes in these graphs represent information carriers (e.g. data, messages and events) whereas the edges represent system behavior (e.g. system functions and services). In a following work [9], Chivers et al. extend these information flow trees with security propagation and construct attack paths based on the information flow. However, such diagrams can neither be used for determining the criticality of system components nor for comparing risks of two IT infrastructures.

Attack paths and attack trees are introduced by Schneier [34] and are widely used in the security literature (e.g. [5], [15], [24], [31], [32]) to model different ways of compromising a system. In most of the cases the nodes of an attack graph represent threats or vulnerabilities, as threat trees do. Our approach resembles attack threes because we model how an attack propagates. However we carry out the propagation analysis at the IT-infrastructure level.

## VI. Conclusion and Future Work

In this paper we presented the CRAC-method and how it can be used (1) as supplement to the existing risk management approaches for practically assessing confidentiality risks of a IT-system that an industrial organization outsources to an organization that is expert in IT-systems, and (2) as a stand alone tool for comparing the security of IT-infrastructures w.r.t. confidentiality. The CRAC-method extends the concept of architecture-based confidentiality risk assessment in the absence of explicit information on confidentiality aspects by (a) eliciting impact related information by modeling the information flow and (b) eliciting the reachability information on critical information assets by modeling attack paths.

We validated the CRAC-Method by applying it to a real-world case, in which confidentiality risks are used to choose among two alternative infrastructure design options.

The CRAC-method makes two assumptions. First, some IT-architectural documentation on the system to be risk assessed is available. Second, for risk assessment purposes staff with good security understanding can be interviewed. Large outsourcing providers are subject to deliver a high level IT-architectural document describing the system to be outsourced. Furthermore, large outsourcing clients employ security staff and chief security officers. Therefore, if applied to a case where the outsourcing provider and client are big organization then both assumptions are satisfying. These indicate reusability of CRAC to any context that satisfies the two assumptions.

We furthermore evaluated the method based on the success criteria defined by the stakeholders of the company that provided the case. According to the evaluation results, the CRAC-Method represents the confidentiality risks in a more detailed manner than the currently employed check-list method. However, CRAC is less detailed than the CRAMM-method. We also show that CRAC can be gracefully adjusted to work at different detail levels, according to the criticality of the target of assessment target of assessment. Finally, we argue that out approach is a significant step towards less subjective risk assessment.

In the future we intend to extend CRAC with tool support.

## References


[1] Directive 95/46/EC on the protection of individuals with regard to the processing of personal data and on the free movement of such data, 1995.

[2] C. Alberts and A. Durofee. An Introduction to the OCTAVESM Method.

[3] Basel II: Revised international capital framework, 2005. http://www.bis.org/publ/bcbsca.htm.

[4] P. Bowen, J. Hash, and M. Wilson. *Information Security Handbook: A Guide for Managers*. NIST Special Publication 800-100, 2006.

[5] R. Breu, F. Innerhofer-Oberperfler, and A. Yautsiukhin. Quantitative assessment of enterprise security system. In *Int. Workshop on Privacy and Assurance*. IEEE Computer Society, 2008.

[6] British Government's Central Computer and Telecommunications Agency. CRAMM: Risk Analysis and Management methodology, 2008.
[7] Centers for Medicare and Medicaid Services. Health Insurance Portability and Accountability Act of 1996 (HIPAA). http://www.cms.hhs.gov/hipaageninfo.
[8] H. Chivers. Information Modeling for Automated Risk Analysis. In *Communications and Multimedia Security*, pages 228–239, 2006.
[9] H. Chivers, J. Clark, and P.-C. Cheng. Risk profiles and distributed risk assessment. *Computers & Security*, 28(7):521 – 535, 2009.
[10] CobiT: Control Objectives for Information and related Technology. http://www.isaca.org.
[11] R. Dantu, K. Loper, and P. Kolan. Risk Management using Behavior based Attack Graphs. In *In Proc. of the Int. Conf. on Information Technology: Coding and Computing (ITCC04)*, 2004.
[12] F. den Braber, T. Dimitrakos, B. A. Gran, M. S. Lund, K. Stolen, and J. O. Aagedal. The CORAS methodology: model-based risk assessment using UML and UP. pages 332–357, 2003.
[13] L. Greenemeier. T.J. Maxx Data Theft Likely Due To Wireless "Wardriving" . *InformationWeek*, 2007.
[14] L. Grunske and D. Joyce. Quantitative risk-based security prediction for component-based systems with explicitly modeled attack profiles. *J. Syst. Softw.*, 81(8):1327–1345, 2008.
[15] T. Ingoldsby. Understanding risk through attack tree analysis. *Computer Security Journal*, 20(2):33–59, 2004.
[16] ISO/IEC 27001:2006 Information Security - Specifications, 2005. http://www.iso.org.
[17] ISO/IEC 27002:2005 Information technology - Security techniques - Code of practice for information security management , 2005. http://www.iso.org.
[18] ISO/IEC 13335 2004: Management of Information and Communication Technology Security - Part 1: Concepts and Models for Information and Communication Technology Security Management., 2004.
[19] BS ISO/IEC 27005 2008: Information technology – Security techniques – Information security risk management, 2008.
[20] ISO/IEC Guide 73 2002: Risk management – Vocabulary – Guidelines for use in standards, 2002.
[21] M. Krause and H. Tipton. *Handbook of Information Security Management*. CRC Press LLC, Auerbach Publishers Inc., 1998.
[22] J. Krim and D. Vise. AOL Employee Charged in Theft Of Screen Names. *The Washington Post*, 2004.
[23] F. Majorczyk, E. Totel, L. Mé, and A. Saïdane. Anomaly Detection with Diagnosis in Diversified Systems using Information Flow Graphs. In *SEC*, pages 301–315, 2008.
[24] S. Mauw and M. Oostdijk. Foundations of attack trees. In *LNCS 3935*, pages 186–198. Springer, 2006.
[25] R. McMillan. SEC, FTC Investigating Heartland After Data Theft. *PCWorld*, 2009.
[26] A. Moore, R. Ellison, and R. C. Linger. Attack Modeling for Information Security and Survivability. Technical report, Carnegie Mellon University, 2001.
[27] A. Morali, E. Zambon, S. Etalle, and P. Overbeek. IT Confidentiality Risk Assessment for an Architecture-Based Approach. In *Proc. of the 3rd IEEE Int. Workshop on Business-Driven IT Management, Salvador, Brazil*. IEEE Computer Society Press, 2008.
[28] S. Osborn. Information flow analysis of an rbac system. In *SACMAT '02: Proc. of the seventh ACM symposium on Access control models and technologies*, pages 163–168. ACM, 2002.
[29] C. Phillips and L. Swiler. A graph-based system for network-vulnerability analysis. In *NSPW '98: Proc. of the 1998 workshop on New security paradigms*, pages 71–79. ACM, 1998.
[30] Personal Information Protection and Electronic Documents Act of Canada, 2000.
[31] I. Ray and N. Poolsapassit. Using attack trees to identify malicious attacks from authorized insiders. In S. D. C. di Vimercati et al., editor, *ESORICS 2005, LNCS 3679*. Springer Verlag, 2005.
[32] R. Sawilla and X. Ou. Identifying Critical Attack Assets in Dependency Attack Graphs. In *ESORICS '08: Proc. of the 13th European Symp. on Research in Computer Security*, pages 18–34. Springer-Verlag, 2008.
[33] S. Schechter. Toward Econometric Models of the Security Risk from Remote Attacks. *Security & Privacy*, pages 40–44, 2005.
[34] B. Schneier. Attack Trees. *Dr. Dobb's Journal*, 12(24):21–29, 1999.
[35] Sarbanes-Oxley Act of 2002, 2002. http://www.sarbanes-oxley.com/.
[36] G. Stoneburner, A. Goguen, and A. Feringa. Risk Management Guide for Information Technology Systems. Technical report, NIST, 2002. SP 800-30.
[37] L. Swiler, C. Phillips, and T. Gaylor. A Grap h-Based Network-Vulnerability Analysis System. Technical report, Sandia National Laboratories, 2001. SAND97-3010/1.
[38] R. Wieringa. Design science as nested problem solving. In *DESRIST '09: Proc. of the 4th Int. Conf. on Design Science Research in Information Systems and Technology*, pages 1–12. ACM, 2009.

## APPENDIX

In the following we plot the tables showing the aspects that the methods in Section IV use by assessing risk and which ones of these aspects are issue to M2, M3 and M4 (indicated by X).

TABLE V
ASPECTS OF THE CRAC-METHOD.

| Aspects | M2 | M3 | M4 |
|---|---|---|---|
| Information asset | | X | X |
| Confidentiality Level | | X | |
| Homogenity | X | | X |
| IT component | | X | X |
| Vulnerability | | X | |
| Threat Agent | | X | X |
| Number of instances that can be retrieved | X | X | X |
| Impact | | | X |
| Total impact | | | X |
| Attack Propagation Graph | | | |
| Attack path | | | X |
| Attack propagation likelihood (1) | | | X |
| Attack propagation likelihood (2) | | | X |
| Attack propagation likelihood (3) | | | X |
| Competencies and Conditions | X | X | |
| risk | | | X |
| Mitigation level | | X | X |

TABLE VI
ASPECTS OF THE CRAMM METHOD.

| Aspects | M2 | M3 | M4 |
|---|---|---|---|
| Assets | | | X |
| Asset value | | X | X |
| Threats | | X | X |
| Extent of vulnerabilities | | X | |
| risk | | X | X |
| Level of threats | | | |
| Countermeasures | | | X |
| Applications | | | X |
| Nr. of persons using that application | | | X |
| Locations | | | X |
| Multi functional assets | | | X |
| Quantity of physical assets | | | X |
| Class of physical assets | | | X |
| Class of software assets | | | X |
| Links between assets | | | X |
| Asset model | | | X |
| Potential impact scenario | | X | |
| Threat source | | | X |
| Financial value | | | X |
| Scale value | | X | |
| Valuation scenario | | X | |
| Likelihood | | | X |
| asset value | | X | X |
| Threat and vulnerability questions | X | X | |
| Measures | | | |

TABLE VII
ASPECTS OF THE CHECK-LIST METHOD.

| **Aspects** | **M2** | **M3** | **M4** |
|---|---|---|---|
| Threat type | | X | X |
| Business impact | X | X | |
| Vulnerabilities | | X | |
| Sensitivity Period | X | X | X |
| Final Business Impact Level | | | |
| Impact | X | | |
| Data type | X | | X |
| Percentage of data | X | | X |
| User type | X | | X |
| Percentage of user | X | | X |
| Interfaces | | | X |
| Threats | | | |
| Final residual risk | | | X |
| Severity | | | |
| Measures | | | X |
| Mitigation level | | | |