Managing Effective Collaboration in Cybersecurity Alliances Using Social Computational Trust

Ameneh Deljoo
System and Network Engineering Lab
University of Amsterdam
Amsterdam, The Netherlands
a.deljoo@uva.nl

Ralph Koning
System and Network Engineering Lab
University of Amsterdam
Amsterdam, The Netherlands
r.koning@uva.nl

Tom van Engers
Leibniz Institute
University of Amsterdam
vanengers@uva.nl

Leon Gommans
System and Network Engineering Lab
University of Amsterdam
Amsterdam, The Netherlands

Cees de Laat
System and Network Engineering Lab
University of Amsterdam
Amsterdam, The Netherlands
delaat@uva.nl

***Abstract*—To enable effective collaboration, trust in the ability of an alliance member to adequately perform joined tasks is an essential element. Such trust needs to be organized, evaluated and maintained amongst all alliance members. In this paper, we present a social computational trust model (SCTM) to evaluate trust as featured by alliance members. Specifically, we consider three different distinctive trustworthiness elements: competence, benevolence ,and integrity. To evaluate the trust of a particular member, we take into account two sources of evidence as well as the context of interactions between the parties. Based on our SCTM model, we have developed an algorithm that ranks the members based on their capabilities, behavior, and integrity in the context of a task that is expected to be performed. A cybersecurity alliance case study is presented to demonstrate the applicability of the SCTM model. Experimental results from a real-world testbed are used to validate the presented SCTM model in selecting the right partner to defend against cyber-attacks.**

***Index Terms*—Computational Trust, cybersecurity Alliance, Competence, Integrity, Partner Selection**

I. Introduction

Trust in a network partners' abilities to adequately perform joined tasks is an essential pre-condition for effective collaboration. This trust needs to be organized in such a way that each member knows what the expected adequate behavior means. Inadequate behavior will affect the quality and effectiveness of joined tasks. Nowadays, alliance members provide their services in automated, software-defined ways [1]. To minimize the response time to jointly limit the impact of cyber-attacks, in particular considering the highly prevalent DDOS type of attacks, requires automated decision making. Such a response should result in a fast and optimal joined set of defense actions where each party is trusted to deliver expected service quality. Within a cybersecurity alliance, finding a reliable partner to collaborate with is a challenge due to the inherent risks of collaboration with competitors or unknown parties. To create an alliance, we identify below requirements [2] for mechanisms that:

- organizes, maintains and evaluates trust amongst members, estimate interaction risk, and
- define common rules, policies, and standards for its members.

To model trust within a cybersecurity alliance, the social computational trust model (SCTM) is based on the three distinctive components:

- Benevolence denotes as the kinship of alliance members even if unexpected contingencies arise (act toward the alliance goal),
- Competence refers to the alliance members' ability to perform tasks.
- Integrity means the alliance members adhere to a set of rules agreed upon and acts accordingly to fulfill the commitments.

In our previous works, we addressed the competence and benevolence functions and their roles in the alliances in detail [3], [4].

In this paper, we extend the SCTM model by assigning weight factors to the SCTM functions and rank the members based on their shown behaviors, capabilities, and integrity in providing the requested services to the alliance members. We will present a partner selection algorithm based on the context of an interaction between two parties. This paper will explain:

- the context-based trust evaluation algorithm,
- how such an algorithm can be used to rank the members based on their benevolent behavior, capabilities, and integrity
- the evidence gathering approach to collect and aggregate the indirect evidence to evaluate a members trustworthiness
- our evaluation of this context-based SCTM with the Secure Autonomous Research NETworks (SARNET) research environment

978-1-7281-3949-4/19/$31.00 ©2019 IEEE

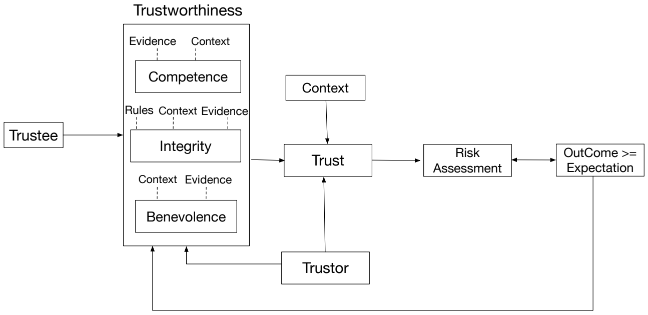


Figure 1: Trust Framework.

- show its practical applicability for partner selection in a distributed cyber defense mechanism context

The paper is organized as follows: Trust in the alliances and its factors is presented in Section II. In Section III, we will present the notation, and the SCTM model with its components described in Section III-B. The SARNET Research environment that was used to evaluate the SCTM model is presented in SectionIV. We describe the simulation setup and the results in Sections Vand VI. In Section VII we review some of the computational trust models. The conclusion will summarize our findings.

II. Trust in the Alliance Context

Trust amongst the members of alliances has been empirically demonstrated to be important for alliance formation [5]. Organizing trust amongst the members comes with benefits such as being a substitute for formal control mechanisms, reducing transactions costs, facilitating dispute resolution, and increasing the flexibility.

Research on trust shows that separate trust factors derive from a partner's competence (i.e., ability, technical skills, experience, and reliability), benevolence (i.e., intention of goodwill, kindness) and integrity (i.e., motives, honesty, and character), and that these factors have potentially unique effects [6], [7], [8], [3]. The trust framework is depicted in Fig. 1. Essentially, the framework says that a member is trustworthy if he has an ability to perform a task in a given situation, his integrity, and has a positive relationship with the trustor. Once trust is established, the trustor is willing to take the risk and the outcome of the risk estimation block will serve as feedback to update the perception of the trustee's factors. Therefore, it is important to estimate the trustee's trustworthiness by considering each of these three factors individually, and dynamically combine them. However, most of the computational trust approaches evaluate the trustee's trustworthiness as a black box and do not consider different trustworthiness's dimensions.

III. Our Social Computational Trust Model

In the following, we present our social computational trust model (SCTM) and its functions to evaluate trust based on benevolence, competence and integrity factors. These functions provide the basis for the decision-making process that each member has to perform when deciding on collaborating or not with the given members (trustees).

Algorithm 1: Calculate the outcome of task based on a task deadline.

```
Require: $\Delta t_w$: time window.
Require: $t_{request}$: request time.
Require: $t_{report}$: report time.
Require: $t_{delay}$: acceptable delay.
1: $d_7 = t_{report} \quad t_{request}$
2: if $d_7 = 0 \vee t_{delay} > 5s$ then
3:     $d_8 = V$
4: else if $d_7 > \Delta t_w \wedge t_{delay} < 5s$ then
5:     $d_8 = Fdd$
6: else if $d_7 =< \Delta t_w \wedge t_{delay} = 0s$ then
7:     $d_8 = Fd$
8: end if
9: return $d_8$
```

A. Notation

The alliance network denotes A, and it includes trustor x and trustee $y \in A$. In this research, each member can be represented as a trustee or trustor. $Tr(x, y, s_i)$ represents the amount of trust x has in y in a situation $s_i \in S$, where $S = \{s_1, s_2, ..., s_n\}$ is the set of all the possible situations in the society.

1) Context Definition: In order to define a situation that lead to an agreement between a trustee and a trustor, in this work, we adapt and extend the context definition by [9], [10]. We recognize eight dimensions for a context $\{d_1, d_2, ..., d_8\}$, where dimensions d_1 and d_2 indicate the parties: a trustor and a trustee, respectively; d_3 and d_4 represent time and the location of agreement; and d_5, d_6, d_7 and d_8 characterize the task type, its complexity, deadline, and the outcome of the task, respectively. We distinguish three different type of outcomes for d_8 dimension, $d_8 \in \{Fd, Fdd, V\}$, where Fd (fulfill duty) denotes that the trustor believes that the trustee performed the given task on time, Fdd (fulfill duty with delay) means that the trustee performed the task (or duty) with an (un)expected delay and V (violation) means that the trustee did not perform the requested and agreed task.

In this paper, we consider different deadlines (d_7) and different outcomes (d_8) for each type of task τ. We define the task deadline as The trustee needs to answer the trustor's request within a certain time window Δt_w, therefore, we calculate the deadline of the task (d_7) for each (sub)-task which can be varied for different requests. And, we employ Algorithm 1 to calculate the task outcome. The Δt_w window is defined by the trustor, and the trustor sends a request to the trustee at time $t_{request}$. The trustee will answer the request, this time called report time, t_{report}. In line 1, we calculate the deadline of the task and following the Algorithm reports the outcome of the task by comparing the deadline to the time window that has been set by the trustor. The trustor waits for an acceptable delay $t_{delay} =< 5s$ to receive the answer from the trustee. In line 2, if the trustor did not receive an answer from the trustee $t_{request} = t_{report}$ or $t_{delay} > 5s$, then, $d_7 = 0$ and the outcome will be V. Otherwise, if d_7 is bigger than Δt_w

and $t_{delay} < 5s$ then the outcome will be Fdd (line 4 and 5). Finally, the outcome will be Fd if the value of d_7 is smaller than Δt_w and $t_{delay} = 0s$ (line 6 and 7).

In Table I, we have summarized the notations that we use for the rest of the paper. The outcome of interactions between trustor x and trustee y is called evidence (E). In the current SCTM framework, we consider all the available evidence on a trustee. And, each trustor has a knowledge based Kb that contains all the interactions with its neighbors. The trustor stores the following information from its interactions in its Kbs, originator's Id, destination's Id, $t_{request}$, t_{report}, task's type and outcome of tasks (quality of task performance) (see Fig. 2). We consequently define evidence (E) as the outcome of the interaction between trustor x and trustee y for a situation $s_i \in S$. According to Algorithm 1 we denote this evidence by $d_8(x, y, s_i)$. Next, we define function $val(.) : d_8 \to [0, 1]$ that assigns a value in the interval $[0, 1]$ to d_8:

$$val(d_8) = \begin{cases} 1 & \text{if } d_8 = Fd \\ 0.5 & \text{if } d_8 = Fdd \\ 0 & \text{if } d_8 = V \end{cases}$$

Evidence Gathering For all the trust components, we assume that the trustor and the trustee has not been collaborated before. Tto gather the evidence and evaluate the trustee's trust, the trustor needs to ask the trustee's direct neighbors opinion on the given trustee. Then, the trustor will aggregate the received evidence and evaluate the trustee's trust. In the following, we explain the evidence gathering mechanism from the trustee's direct neighbors.

Direct Evidence The set defines the direct evidence of the interaction between trustor x and trustee y in situation s_i:

$$Ed(x, y, s_i; Kb_x) = \{d_8(x, y, s_i) \in Kb_x\}, \tag{1}$$

that is the set of d_8 values from all entries in the knowledge-base Kb_x of trustor x that deal with the interaction between x and y in situation s_i. To extract the evidence of the other dimension of context, we can replace d_8 by other dimensions such as d_5 or d_6 to extract the evidence of that specific dimension.

We define the function $val_d(.) : Ed \to [0, 1]$ that assigns a value in the interval $[0, 1]$ to the set Ed as:

$$val_d(Ed(x, y, s_i; Kb_x)) = \frac{1}{N_x} \sum_{d_8(x,y,s_i) \in Ed(x,y,s_i;Kb_x)} val(d_8(x, y, s_i)), \tag{2}$$

where N_x [1] is the number of entries in Kb_x that deal with x and y in situation s_i.

InDirect Evidence Likewise, we define the available evidence of the interaction between the neighbors of y as a trustor and y as a trustee in situation s_i as the set:

$$Ec(nbr_y, y, s_i) = \{Ed(u, y, s_i; Kb_u) | u \in nbr_y\}, \tag{3}$$

where nbr_y is the set of neighbors of y.

For this set we define function $val_c(.) : Ec \to [0, 1]$ that assigns a value in the interval $[0, 1]$ to the set as:

$$val_c(Ec(nbr_y, y, s_i)) = \frac{1}{N_{nbr}} \sum_{Ed(u,y,s_i;Kb_u) \in Ec(nbr_y,y,s_i)} val_d(Ed(u, y, s_i; Kb_u)), \tag{4}$$

where N_{nbr} is the number of neighbors that contribute to val_c. In section III-A, we present our evidence gathering approach to extract the all/available evidence on trustee y in situation s_i. However, to extract the task related evidence, we need to restrict the evidence gathering search in the Kbs to a specific task. This search is given by:

$$d_8(x, y, s_i) \wedge d_5, \tag{5}$$

where d_5 represents the type of task τ that performs by the trustee in a request of the trustor.

B. Social Computational Trust Model (SCTM)

The social computational trust model (SCTM) that we explain in this paper combines three distinct functions, namely benevolence, competence, and integrity functions. In this section, we present these functions. We illustrate the SCTM model in Fig. 2.

C. The benevolence evaluation function

Several scholars have considered the benevolence as one of the key elements of trust and the trustworthiness's antecedent (e.g., [11], [12]). The value of the benevolence, $Ben(x, y, s_i)$, of trustee y toward trustor x is computed from their mutual interactions in situation s_i that τ needs to be perform. To calculate the benevolence, trustor x extracts the direct evidence of its interactions with trustee y from its knowledge-base Kb_x (see Fig. 3a), $val_d(Ed(x, y, s_i; Kb_x))$ represents the function that extracts the direct evidence between x and y in situation s_i. The estimated value for the benevolence function, which is in the interval of $[0, 1]$, is

$$Ben(x, y, s_i) = val_d(Ed(x, y, s_i; Kb_x)) \tag{6}$$

where N is the number of entries in Kb_x with the associated value, in which x has interacted with y.

D. The competence evaluation function

The competence function $Com(nbr_y, y, s_i)$ evaluates the given trustee's ability in performing a given task τ in the specific situation s_i. The competence function takes all the evidence available on the trustee under evaluation as inputs. A trustor will request the evidence from the direct a trustee's direct neighbors about the ability of the trustee on performing the task τ. The $Ec(nbr_y, y, s_i)$ represents the complementary evidence on trustee y from its direct neighbors. Fig. 3b shows trustor x sends a request to trustee y's neighbors to gather the evidence about the performance of trustee y. The value for the

[1] We assume that $N_x >= 2$.

Table I: Notations and values

Description	Representation	Value Range
Agent	x,y	
The alliance network (trustor, trustee)	$x, y \in A$	
Knowledge based of trustor x	Kb_x	
Set of Situations	$S = \{s_1, s_2, ..s_n\}$	
Tasks	τ	
Sub-tasks	$\tau_{s1}, ...\tau_{sn}$	
Context	$D = \{d_1, d_2, ...d_8\}$	
d_8	$\{Fd, Fdd, V\}$	1, 0.5, 0
All the direct evidence on y in situation s_i	$Ed(x, y, s_i; Kb_x)$	
All the available evidence on y from y's neighbors in situation s_i	$Ec(nbr_y, y, s_i)$	
Trustee's trustworthiness toward trustor x in situation s_i	$TW(x, y; s_i)$	[0,1]
Trust x on y in situation s_i	$Tr(x, y; s_i)$	[0,1]

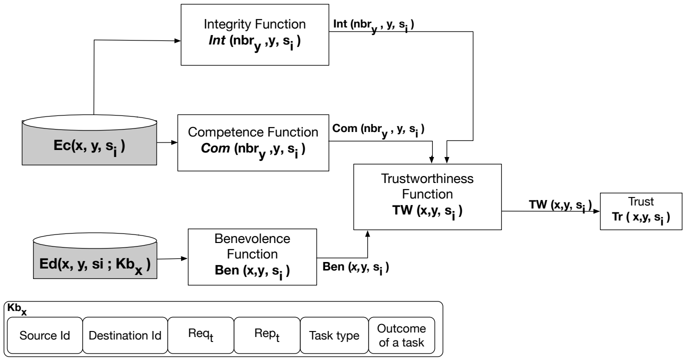
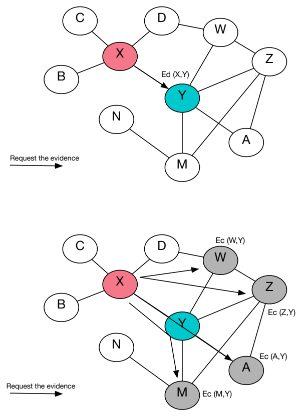


Figure 2: **S**ocial **C**omputational **T**rust **M**odel (SCTM).



(b)

Figure 3: Gathering all the available evidence on trustee y. (a) Gathering the direct evidence on trustee y. (b) Gathering the evidence on trustee y's from its direct neighbors.

competence of the trustee, $Com(nbr_y, y, s_i)$, is in the interval of $[0, 1]$.

$$Com(nbr_y, y, s_i) = \\ val_c(Ec(nbr'_y, y, s_i)), nbr'_y = nbr_y \setminus \{x\}, \tag{7}$$

Based on the assumption, x did not collaborate with y before; therefore, we exclude x from the evidence gathering of y's neighbors.

More broadly, by using the Eq. 5 we can determine the benevolence and competence of a trustee based on a specific task.

E. Integrity function

Integrity refers to the consistency of trustees' behaviors to adhere to a set of norms (agreed contract) [6]. We define a trustee's integrity, as his consistency in his past action. That means that the trustee is consistent in fulfilling his promises and perform the given task successfully (promises are regarded as the agreed contract). A trustee's integrity is computed by:

$$Int(nbr_y, y, s_i) = \frac{1}{N_{Ec}} \sum_{Kb_u \in nbr_y} N_{Fd}(Kb_u, y) \tag{8}$$

where $N_{Fd}(Kb_u, y) = |Ed(u, y, s_i; Kb_u)|u \in nbr_y \wedge val(d_8(u, y, s_i)) = Fd|$ is the number of evidence in the Kb_u that trustee y completed the agreed task successfully ($val(KB_u d_8) = Fd$) and $N_{Ec} = |Ec(nbr_y, y, s_i)|$ represents the total number of evidence on trustee y from all its direct neighbors in situation s_i.

In the SCTM model, integrity and benevolence are behavioral properties, while competence depends on the ability of the trustee to perform the given task τ.

F. The Trustworthiness evaluation function

The trustworthiness evaluation function $TW(x, y, s_i)$ estimates the trustworthiness of trustees toward trustors from the combination of the competence and the benevolence function, such that:

$$TW(x, y, s_i) = \frac{1}{3}(Ben(x, y, s_i) + Com(nbr_y, y, s_i) \\ +Int(nbr_y, y, s_i)) \tag{9}$$

. The $TW(x, y, s_i)$ is normalized to be between 0 and 1.

It should be mentioned that, unlike our trustworthiness evaluation function, in the work of [13] and [14] the multiplication operations used instead of the summation operation. The reason is that they tried to rank the factors instead of computing the total value of the trustees' trustworthiness. Nevertheless, they claimed that the value could be small due to the multiplication operations.

G. The trust evaluation function

The trust evaluation function $Tr(x, y, s_i)$ estimates the trust that trustor x has on trustee y in situation s_i. As illustrated in Fig. 2, this function takes the estimated value of the trustee's trustworthiness, which is given by Eq. 9. This means that, so far it concerns the work of this paper $Tr(x, y, s_i) = TW(x, y, s_i)$. We conclude that we can evaluate trust between any trustees and trustors, no matter whether they have direct interactions or not.

As we mentioned in [3], the trust factors have potentially unique effects on evaluating the members. Moreover, Mayer et al. [6] stated that trust computes from different factors in different situations. Therefore, in this paper to determine a proper set of values for the three components of the SCTM model. We employ three different weights for the benevolence, competence and integrity functions to determine the impact of each factor to select the right partners. Therefore, $Tr(x, y, s_i)$ in Eq. 10 returns the trust value that trustor x has in trustee y in situation s_i and a tau needs to be performed by y, and is presented by:

$$\begin{aligned} Tr(x, y, s_i; \tau) = \frac{1}{3}(\alpha * Ben(x, y, s_i; \tau)+ \\ \beta * Com(nbr_y, y, s_i; \tau) + \gamma * Int(nbr_y, y, s_i; \tau)) \end{aligned} \tag{10}$$

IV. Secure Autonomous Research NETworks implementation

In [15], [16] Koning et al. use the SARNET research environment to create a multi-domain overlay network using virtual machines and virtual network functions. Each domain acts autonomously using its own agent that uses the domains' resources to defend against attacks. Since defending against distributed attacks benefits from cooperation, we decided to facilitate this by applying the model in this paper. Algorithm 2 presents a method to the defense against a Distributed Denial of Service (DDoS) attack by asking help from its most trusted nodes. The SARNETs domain-agents in the overlay cooperate by requesting certain tasks to be executed by other members in response to an emulated attack (see [16] Section 6). We identify two distinct types of tasks for the alliance members during the attack period, informative and executive tasks. The informative task concerns the behavior of the given member, while the executive task represents the ability of the given member to perform the task. The Informative task is based on the information flow and requesting threat information locally or delegated the responsibility to a member to act on your behalf. On the other hand, executive tasks are the actions and tasks that need to be performed in order to have an effective collaboration in defense and mitigation of the attack.

- *Informative task* to provide and respond to the requested information.
  - *Informative tasks perform locally*: Trust a member to identify a traffic source or to request information.
  - *Informative tasks delegate to a member*: Trust a member to continuously send threat information.
- *Executive task* Trust a member to implement a countermeasure to reduce the impact of an attack.
  - *Executive tasks perform locally*: Trust direct neighbors to perform actions.
  - *Executive tasks delegate to a member*: Trust a member to act on your behalf.

The SARNET domain-agents' behavior can be changed according to some pre-defined parameters. For example, we can give the agents a pre-filled database of evidence, or, set the probability that an agent executes the task. Table II shows the pre-filled values that we used for the SARNET agents. Therefore, we assign the α, β, and γ as the weights to the trust factors based on the task type.

Table II: Pre-filled evidence and the success probability of executing the task for the members participating in the alliance.

member	FD	FDD	V	success probability
M	3	1	1	0.0000001
Y	1	1	1	1.0
Z	1	1	1	0.6
A	1	1	1	0.1
W	1	1	1	0.0000001

Algorithm 2: Evaluate the trustworthiness and rank the alliance members based on the task type (τ).

```
Require: $tau$.
Require: $Com$: competence of the given trustee.
Require: $Int$: Integrity of the given trustee.
Require: $Ben$: benevolence of the given trustee..
Employ $Ben(x, y; s_i)$, $Com(nbr_y, y, s_i; \tau)$, and integrity
  $Int(nbr_y, y, s_i; \tau)$ functions to calculate the competence, integrity and
  benevolence for the given members.
Consider the $\alpha$, $\beta$, and $\gamma$ weight to evaluate trust of a trustee based on a $\tau$ (see
  Eq. 10).
$Lst_r$ = Rank the members based on their trustworthiness values and task type $\tau$.
if two members have the same Rank in $Lst_r$ then
    Select a member with the maximum benevolence value.
end if
return $Lst_r$
```

V. Simulation Setup

To validate and test the SCTM model, we implement the SCTM model in two distinctive environments. First, we use an Agent-Based Model (ABM) to setup the SARNET alliance network [2], shown in Fig. 4 in the Jadex platform [17], where the nodes represent the alliance member. Second, the SARNET research environment as described in Section IV is used to validate the SCTM model.

Basically, in the case of attacks such as DDoS, collaboration and coordination amongst the organizations is an essential [18]. In the case of DDoS attack, the victim can start to mitigate and defend against the DDoS attack locally within its own domain or delegate responsibility to a member [19], [18]. In both cases, the victim needs to trust a member to perform the given action or provide the requested information. In our simulation, we present the evaluation and ranking algorithms that aim to evaluate and rank the members based upon their shown behaviors, capabilities, and integrity. The ranking algorithm will help the victim to select the right partner to resolve the situation.

Let us consider two following scenarios that present two distinct attack situations:

- *single attacker close to the victim ("N")* and the attacker in position "12";

[2] The technical details about the SARNET project can be found in http://delaat.net/sarnet/index.html

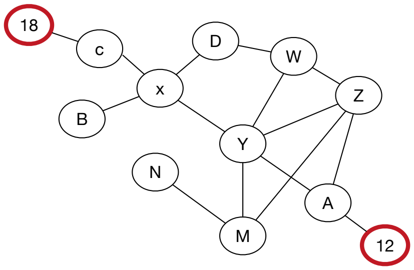


Figure 4: The SARNET Alliance Network Schema.

Table III: Simulation settings and their illustrations.

Parameters	Values	Illustrations
A	Fixed	Nodes Number
τ	3	Informative locally Executive locally and Executive delegated
N_x	6	Number of entries in the Kbs
$t_{request}$	-	Request time
t_{report}	-	Report time
Δt_w	10 s	Time window
t_{delay}	5 s	Acceptable delay
α, β, γ	0.4, 0.7, 0.3[4]	Weight factors
S	2	number of situations

- *single attacker far from the victim*, with the attacker in position “18";

We implement these scenarios in the Jadex platform. And, we rank the members based upon their trustworthiness which computes from the trustee’s capabilities, behaviors, and integrity to perform the given context of task.

In the network, each node has a knowledge-base Kb (Kb is presented in Fig. 2) which contains the evidence of its interactions with the direct and indirect neighbors. We extract the related evidence to calculate d_8, Ec and Ed from the node’s Kb. For each task, there are multiple entries in the node’s Kb from all its direct neighbors. The d_3, d_4, d_5 dimensions are selected based the pre-defined values and we calculate the deadline d_7 and task outcome d_8 by employing Algorithm 1. Each simulation was repeated 10 times and we gathered the direct and indirect evidence from the Kbs. Table III gives the details of the simulation settings. Fig. 4 presents the network schema of the SARNET Alliance [3].

VI. Result

The topology shown in Fig. 4 is used to implement the scenarios in Section V. We illustrate the result of our simulation in Fig. 6a and 6b. The horizontal line indicates the number of iteration and the vertical line shows the trustworthiness of the members over time. To calculate the trustworthiness of a member (a trustee), based on the task type (i.e., informative or executive) we extract the evidence from the trustee’Kbs and use Eq. 10 to evaluate the members’ trustworthiness based on the task type. And, Algorithm 2 provides a list (Lst_r) that shows the rank of each member to the trustor. We implement two mentioned scenarios (Section V with the following tasks:

- *Informative tasks perform locally*: Ask your direct neighbors to identify a traffic source.
- *Executive tasks perform locally*: Ask direct neighbors to perform actions.
- *Executive tasks delegate to a member*: Ask a member to act on your behalf.

In both scenarios, we have the combination of the informative and executive tasks in our simulation; therefore, the trustworthiness computes from the available evidence on both local and delegate tasks outcomes. Figs. 6a and 6b illustrate the trustworthiness of each member overtime for two mentioned scenarios. Each line represents the trustworthiness value for a member, we take 10 snapshots of the iteration and rank the members based on the evidence for these 10 iterations, at the moment the member start to perform a task. The different colors or indication symbols presents the different nodes in the alliance. The context-based trust approach is indicated in Algorithm 2 and is compared to the approaches from SARNET research environment. As Fig.6a shows, when the attacker is in position “12", member “Y" recommends to the victim (i.e.,“N") to perform the task and has the rank first. On the other hand, when the attacker is far 6b and locates at position “18", then “A" is selected to help the victim (i.e.,“N").

Fig. 5a shows the ranking of members, when the attacker is close to the victim (i.e.,“N") and locates at “12". In this case, member “Y" recommends to the victim as the right partner. Nevertheless, member “x" has the same ranking number but with the lower benevolence number; therefore, “Y" has been selected from the list.

In Figs. 7a and 7b we compare the SCTM ranking algorithm with the SARNET research environment (i.e., explain in IV for two scenarios. The x axis indicates the iteration numbers and the y axis shows the trustworthiness of each node in the alliance. As we can observe from the result, when the attacker is in position 12, the SCTM simulation and the SARNET research environment results are in a good agreement with each other. For the second part of the evaluation, when the attacker is far, the SCTM and SARNET research environment varies in some cases like member “W". The variations can be explained because of the success probability and predefined evidence that use for each member in the SARNET implementation.

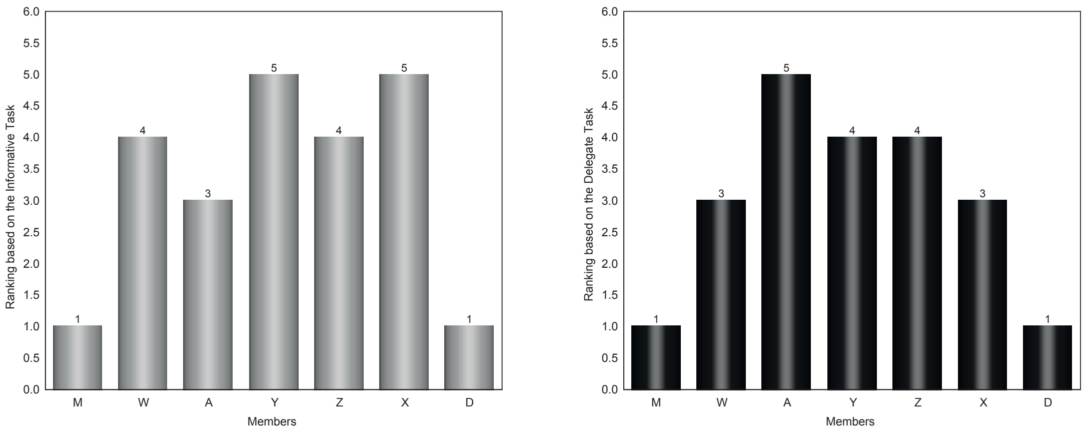
VII. Related work

Different scholars have presented many computational trust models; nevertheless, only a few models are actually social computational models. One of the conceptual models of social trust developed by [20] takes ability, positive intentions, ethics, and predictability as the trustworthiness components. They used a probabilistic approach in their model, however, by realizing the limits of the approach in the treatment of the social concepts their model was not implemented [20].

The socio-cognitive trust model developed by Castelfranchi and Falcone [21]. In their view, trust is made by considering

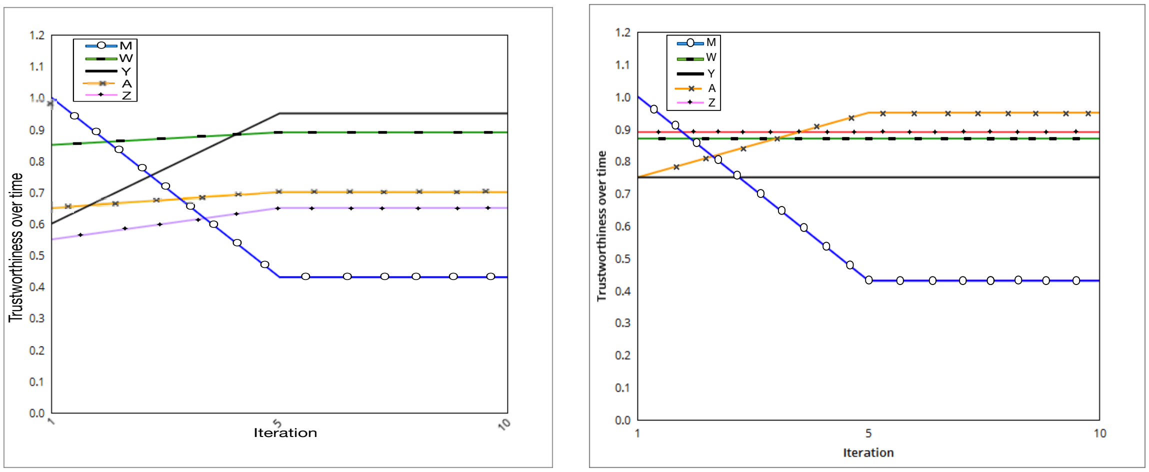
[3] The source code for the SARNET Alliance can be found in $https : //github.com/Adtrust/Collaborative_network$.

[4] The corresponding values for α, β, γ are adopted from [13].



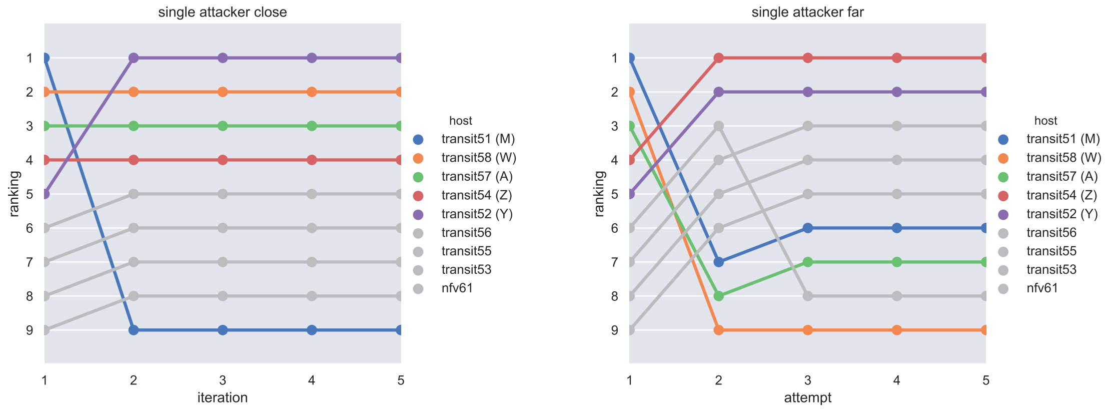
(a) (b)

Figure 5: (a)Rank the Members based on the competence, benevolence and integrity by the SCTM framework when the attacker is in position “12" and (b) when the attacker is in position “18".



(a) (b)

Figure 6: The members trustworthiness over time in performing different tasks (a) when the attacker is in position “12" and (b) when the attacker is in position “18".



(a) (b)

Figure 7: The SCTM implementation result in the SARNET Environment and rank the members (a) Rank the Members when the attacker is in position “12" and (b) when the attacker is in position “18".

the different beliefs that the trustor has about the trustee, both internal (beliefs on competence) and external (opportunities). In practice, it is difficult to implement due to its richness. Situation-aware computational trust Model (SOLUM), developed by author [10]. The model calculates trust based on the trustor's disposition and emotional state. The Urbano's model did not consider different stages of relationships for the competence function. We slightly adapted and modified the Marsh [22] competence formula by considering three different situations for the trustor to decide about the (future) collaboration with the trustee. And, we introduced the risk evaluation approach through the SCTM model [3].

VIII. Conclusion

In this paper, we present a social computational trust model (SCTM) to evaluate trust amongst members of a cybersecurity alliance based on the context of interaction. To evaluate the trustworthiness of a trustee the direct and indirect evidence on the given trustee were taken into account. The trust value is computed by three trust factors: competence, benevolence, and integrity. Benevolence is computed from direct evidence between a trustee and a trustor gained through direct interactions. Competence and integrity are assessed based on the received feedback from the trustee's direct neighbors. We proposed eight dimensions for each context to gather a variety of evidence on a trustee. We assigned a weight to the trust factors and presented the impact of each factor in evaluating the members' trustworthiness. Based on this evaluation, we proposed an algorithm that ranked the members based on their trustworthiness by considering two distinctive tasks: the informative and executive task. To examine the validity of the SCTM model, we compared the results with the SARNET testbed. The SCTM is able to obtain comparable results as obtained from the SARNET testbed.

Acknowledgments

SARNET project is funded by the Dutch Science Foundation NWO (grant no: CYBSEC.14.003/618.001.016) and the National project COMMIT (WP20. 11). Special Thanks to our colleague Dr. Arie Taal for his comments that greatly improved the manuscript. We also thank our other research partners KLM and TNO.

References

[1] R. Koning, A. Deljoo, S. Trajanovski, B. de Graaff, P. Grosso, L. Gommans, T. van Engers, F. Fransen, R. Meijer, R. Wilson *et al.*, "Enabling e-science applications with dynamic optical networks: Secure autonomous response networks," in *Optical Fiber Communications Conference and Exhibition (OFC), 2017*. IEEE, 2017, pp. 1–3.

[2] A. Deljoo, T. van Engers, R. Koning, L. Gommans, and C. de Laat, "Towards trustworthy information sharing by creating cyber security alliances," in *IEEE TrustCom-18*. IEEE, 2018, pp. 1506–1510.

[3] A. Deljoo, T. van Engers, L. Gommans, and C. de Laat, "Social computational trust model (sctm): A framework to facilitate selection of partners," in *2018 IEEE/ACM Innovating the Network for Data-Intensive Science (INDIS)*. IEEE, 2018, pp. 45–54.

[4] A. Deljoo, T. van Engers, L. Gommans, and C. de Laat, "The impact of competence and benevolence in a computational model of trust," in *IFIP International Conference on Trust Management*. Springer, 2018, pp. 45–57.

[5] P. S. Ring and A. H. Van de Ven, "Structuring cooperative relationships between organizations," *Strategic management journal*, vol. 13, no. 7, pp. 483–498, 1992.

[6] R. C. Mayer, J. H. Davis, and F. D. Schoorman, "An integrative model of organizational trust," *Academy of management review*, vol. 20, no. 3, pp. 709–734, 1995.

[7] J. B. Barney and M. H. Hansen, "Trustworthiness as a source of competitive advantage," *Strategic management journal*, vol. 15, no. S1, pp. 175–190, 1994.

[8] R. Krishnan, X. Martin, and N. G. Noorderhaven, "When does trust matter to alliance performance?" *Academy of Management journal*, vol. 49, no. 5, pp. 894–917, 2006.

[9] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, "Towards a better understanding of context and context-awareness," in *International Symposium on Handheld and Ubiquitous Computing*. Springer, 1999, pp. 304–307.

[10] J. Urbano, A. P. Rocha, and E. Oliveira, "The impact of benevolence in computational trust," in *Agreement Technologies*. Springer, 2013, pp. 210–224.

[11] D. Z. Levin, R. Cross, L. C. Abrams, and E. L. Lesser, "Trust and knowledge sharing: A critical combination," *IBM Institute for Knowledge-Based Organizations*, vol. 19, 2002.

[12] T. R. Koscik and D. Tranel, "The human amygdala is necessary for developing and expressing normal interpersonal trust," *Neuropsychologia*, vol. 49, no. 4, pp. 602–611, 2011.

[13] G. Guo, J. Zhang, D. Thalmann, and N. Yorke-Smith, "Etaf: An extended trust antecedents framework for trust prediction," in *Proceedings of the 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. IEEE Press, 2014, pp. 540–547.

[14] H. Liu, E.-P. Lim, H. W. Lauw, M.-T. Le, A. Sun, J. Srivastava, and Y. Kim, "Predicting trusts among users of online communities: an epinions case study," in *Proceedings of the 9th ACM conference on Electronic commerce*. ACM, 2008, pp. 310–319.

[15] R. Koning, B. de Graaff, G. Polevoy, R. Meijer, C. de Laat, and P. Grosso, "Measuring the efficiency of sdn mitigations against attacks on computer infrastructures," *Future Generation Computer Systems*, vol. 91, pp. 144–156, 2019.

[16] R. Koning, G. Polevoy, L. Meijer, C. de Laat, and P. Grosso, "Approaches for collaborative security defences in multi network environments," in *The 6th IEEE International Conference on Cyber Security and Cloud Computing (IEEE CSCloud 2019)/(IEEE Edgecom 2019)*, 2019.

[17] L. Braubach, W. Lamersdorf, and A. Pokahr, "Jadex: Implementing a bdi-infrastructure for jade agents," 2003.

[18] Y. Chen, K. Hwang, and W.-S. Ku, "Collaborative detection of ddos attacks over multiple network domains," *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 12, pp. 1649–1662, 2007.

[19] L. Kagal, T. Finin, and A. Joshi, "Trust-based security in pervasive computing environments," *Computer*, vol. 34, no. 12, pp. 154–157, 2001.

[20] S. Adali, W. Wallace, Y. Qian, P. Vijayakumar, and M. Singh, "A unified framework for trust in composite networks," *Proc. 14th AAMAS W. Trust in Agent Societies, Taipei*, pp. 1–12, 2011.

[21] C. Castelfranchi and R. Falcone, *Trust theory: A socio-cognitive and computational model*. John Wiley & Sons, 2010, vol. 18.

[22] S. P. Marsh, "Formalising trust as a computational concept," 1994.