Defence Research and Development Canada
Recherche et développement pour la défense Canada

DRDC | RDDC
technologysciencetechnologie

A cyber security course of action selection approach for automated computer network defence

Maxwell Dondo
DRDC – Ottawa Research Centre

Workshop on Applications and Techniques in Cyber Security (ATCS 2017)

Date of Publication from Ext Publisher: October 2017

Defence Research and Development Canada
External Literature (P)
DRDC-RDDC-2017-P093
October 2017

Canada

IMPORTANT INFORMATIVE STATEMENTS

This document was reviewed for Controlled Goods by Defence Research and Development Canada (DRDC) using the Schedule to the *Defence Production Act*.

Endorsement statement: This publication has been peer-reviewed and published by the Editorial Office of Defence Research and Development Canada, an agency of the Department of National Defence of Canada. Inquiries can be sent to: Publications.DRDC-RDDC@drdc-rddc.gc.ca.

Template in use: Normal.dotm

1

Cyber security decision support for remediation in automated computer network defence

Maxwell Dondo

Defence Research and Development Canada
Ottawa, ON K1A 0Z4

Summary. In making important cyber security course of action (COA) decisions, experts mostly use their knowledge and experience to collate and synthesise information from multiple and sometimes conflicting sources such as the continually evolving cyber security tools. Such a decision making process is resource intensive and could result in inconsistencies from experts' subjective interpretations of how to address the network's security risks. The push towards automated computer network defence (CND) systems requires autonomous decision making and recommendation approaches for network security remediation. In this work, we present such a novel approach through a TOPSIS-based multi-attribute decision making COA selection technique. Our model uses a survey of experts to show that human experts' decisions are indeed inconsistent, even when they are provided with the same information. We then present our decision making approach that is based on considering multiple COA selection factors in an operational environment and implementing a multi-objective selection method that provides network defenders with the best actionable COAs for an automated CND system. Our results show consistency that is unmatched by human experts.

Key words: course of action, vulnerability, patching, attack graph, remediation, decision-making

1.1 Introduction

Computer networks supporting modern business processes or missions are becoming increasingly complex. Unfortunately, that complexity means more effort is required to determine and address its vulnerabilities to maintain network security. As a result, defenders have increasingly relied on automation and recommendation tools to assist them in providing the information necessary to implement effective network defence [1,2]. As explained in Section 1.2, some of the automation tools such as MulVAL simplify the network defence task by presenting to the defender all the possible ways that attackers could use to reach certain goals on the defended network [3]. The defender must

use this information to determine the set of defensive activities to prevent or make it difficult for the attacker to reach those goals. These defensive activities, such as patching vulnerable software or reconfigurations through firewall rule changes are the cyber COAs that are the subject of this study [4,5]. From the options presented to them, defenders must select the COAs that maximally improve the security of the network given finite resources, plausibility of remediation methods and the need to maintain business continuity.

Although making expedient COA selections is a hard problem, it is required. Network security tools and the defenders' expertise provide a holistic understanding of the security posture of the defended network. But, explicit information about the best COAs that maximally improve the network's security and maintain business continuity is not readily available to defenders. Considering the multiple factors that need to be taken into account to select such COAs, the reliance on human expertise can be resource-intensive and can lead to inconsistent results due to the difficulty in making multi-factor decisions inherent in human operators [6,7]. Automation and stand-alone selection tools have been touted as obvious solutions for such limitations [2]. But the context-aware methodologies they need to support consistent and repeatable COA selections are unavailable. In this work, we provide such a methodology. Our approach selects the best actionable network security COAs for implementation given finite remediation resources while minimising disruptions to business processes. We also show the inconsistency of human operators even when they are presented with the same information.

As explained in Section 1.2, existing tools such as Altiris [8] or Redseal [9] currently support COA selection. But, they are not designed to incorporate operational data in their decision making process. Human operators must apply their operational knowledge and experience to information from such tools, as well as other contextual data, to complete the remediation picture and make the necessary COA selection decisions. Multiple experts often make such security decisions, which they have been shown to be mostly incapable of delivering with the consistency expected in CND [6,7]. Thus, we argue that operators need a consistent autonomous methodology such as ours to select the best actionable COAs in an operational environment.

In our approach, which we present in Section 1.3, we first determine factors (attributes) that affect COA selection in the operational environment [10]. Then we formulate the COA selection problem as a multi-attribute decision-making (MADM) problem (a problem that depends on multiple attributes) based on these factors. To solve it, we chose the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) [2, 11]. Among many other possible MADM techniques, TOPSIS is well suited for our problem definition since it has been widely and successfully used in solving MADM problems similar to ours [1, 2, 11].

We applied our model to COA data that we generated using an arbitrarily simulated operational environment on an in-house tool, the inteGrated ENd to End deciSIon Support (GENESIS) [12]. Separately, we used a survey of

experts to analyse the effectiveness of using human network security operators in COA selections. Our experts, drawn from knowledgeable colleagues with a minimum of ten years cyber security mitigations experience each, were elicited for their COA selection recommendations in the same simulated environment. We then compared the experts' selections with those from our methodology.

As described in Section 1.4, the survey results validated the difficulty of getting consistent selections among the different experts given the multiple factors they need to consider. This is a known human limitation that was also reported by Miller *et al.* [7] and Kim *et al.* [2]. Although we did not have a way to validate the selections of our model, which we leave for possible future work, it produced self-consistent results that agree with the original multiple objectives of our approach. We present our conclusions in Section 1.5.

1.2 Courses of action

Computer network security COAs are the remediation activities required to improve the security of a network. In typical networks, which are usually made up of many interconnected assets, these COAs can be numerous and difficult to determine. Fortunately, attack graph-based algorithms, such as MulVAL [3], simplify that task by determining all the possible paths that an attacker can take to achieve certain goals on a vulnerable network.

An attack graph shows how an attacker could link together network configuration and vulnerability information to achieve their goals [3]. A typical attack graph, showing how an attacker could reach their goal, is illustrated in Figure 1.1.

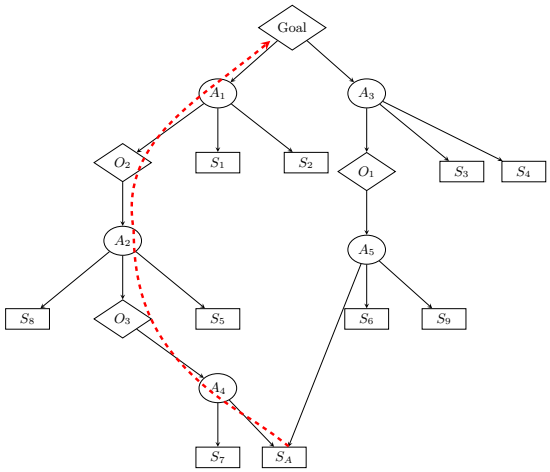

Fig. 1.1. An example of an attack graph. The dashed line illustrates one of the two possible paths from S_A to the goal node (adapted from [10]).

The attack graph in Figure 1.1 is made up of three types of nodes. The rectangular SINKs, oval ANDs and diamond ORs represent facts, logical conjunctions and logical disjunctions respectively. For example, A_2 is only true if A_4, S_5 and S_8 are all true. As illustrated by the dashed line, an attacker located at S_A could reach the *Goal* node through the following logical path: $S_A \rightarrow A_4 \rightarrow O_3 \rightarrow A_2 \rightarrow O_2 \rightarrow A_1 \rightarrow$ Goal. However, that path is not possible if, for example, SINK node S_5 is removed from the attack graph.

Such a removal of a SINK node defines a COA set that we will consider in this work. We represent it as C_1 $[S_5]$ for the first COA set in a set of other COA sets. Other possible COA sets for the dashed path are C_2 $[S_1]$, C_3 $[S_3, S_5]$, C_4 $[S_4, S_5]$ and C_5 $[S_3, S_4, S_5]$. The five COA sets collectively constitute one possible set of COA sets. Our work focuses on determining which one of these five is the best actionable COA set to implement in an operational environment. Further reading on attack graphs can be found in the literature [3–5].

1.2.1 Characterising COAs

In each COA set, there can be SINK nodes of different types, each type representing a weakness on the network that can be exploited by an attacker. Examples of these types are the existence of software vulnerabilities (*vulExists*), the existence of logical connectivity between two network entities (*hacl*) or the existence of some network service such as email (*networkServiceInfo*). Although there are many possible SINK types (ARMOUR [13], an automated CND architecture, for example, defines nine types), we simplify our work by focusing on the above three. These three are the most common types in COA sets [4, 10, 12]. This simplification does not affect the generalisation of the problem at hand, and we therefore defer the inclusion of other SINK types to possible future work.

1.2.2 COA selection factors

In an operational environment, network defenders are presented with many COAs to consider. To select a COA, they must consider the different technical and operational factors that characterise its remediation activities. Examples of such factors are the SINK type (e.g. patching an existing software vulnerability) or the availability of technical resources. Our work focuses on preferentially selecting COAs based on these factors. A summary list of the COA factors (attributes) is shown in Table 1.1. The factors, which were introduced in [10], are listed with their associated ranges of possible numerical scores as used later in our analysis.

From the table, the first three factors represent the SINK type. For example, the factor *Service change* represents the presence of a network service, such as web service, whose mere existence could be exploited by an attacker. The next factor represents the impact on missions or business processes if the

Table 1.1. COA selection factors and numerical scores.

Factor i	**Description**
1. Vulnerability to patch $[0, 20]$	Corresponds to the *vulExists*$(\cdots)$ node in an attack graph, and represents the number of vulnerabilities in a COA set.
2. Firewall change $[0, 10]$	Corresponds to the *hacl*$(\cdots)$ node from an attack graph. Often configured in the firewall, it represents a change in the communication rules between two hosts.
3. Service change $[0, 10]$	Corresponding to an attack graph's *networkServiceInfo*$(\cdots)$ SINK node. It represents changes to the network services.
4. Impact to missions $[0, 10]$	This attribute represents the impact on missions if the COA set is implemented.
5. Patch impossible $[0, 1]$	This attribute represents the feasibility of implementing a patch, even if patch exists.
6. No remediation tool $[0, 1]$	This attribute represents the existence of a remediation tool.
7. Resource limitation $[0, 1]$	This attribute represents the shortage of resources to implement the COA set.
8. COA cost $[0, 50]$	This is a predetermined COA remediation cost. This attribute represents the cost assigned to the COA set $[4, 5]$.
9. Security benefit $[0, 1]$	This attribute represents the percentage of the attack graph that is eliminated by the implementation of the COA set.

COA set is implemented. However, it may be practically infeasible to remove some SINK nodes. We represent this impediment by the fifth factor. Remediation is usually facilitated by using tools such as patching software or scripts. The availability of such tools is represented by the sixth factor.

The next factor represents the shortage of resources to implement the COAs. The COA cost attribute assigns a relative numerical measure representing the total remediation costs. In addition, the effort required in removing some SINK nodes may be higher than others and the network defenders might not have enough resources to ensure the complete removal of all COA set nodes. The final factor represents the security benefit obtained if the COA is implemented. For our security benefit, we use a rank measure developed by Sawilla and Ou [4, 14]. It represents the importance of a graph vertex to an attacker. The security benefit comes from the fraction of these vertices that is removed (through remediation) to prevent an attacker from reaching their goal. An ideal rank elimination is 1 as opposed to an undesirable value of 0.

Based on our research, we found the nine factors presented in Table 1.1 to be vital for COA selection in an operational environment. However, it is possible that there are other factors that we may have missed. Our approach can be extended to include an extended set of factors if necessary.

1.2.3 COA selection challenges

Most network defenders have the knowledge and experience to make remediation decisions based on considering known remediation factors and selecting the best actionable COAs. For each factor, the selection objective is either to maximise or minimise it. For example, from the factors in Table 1.1, the

remediation objective is to minimise mission impacts (Factor 4) and maximise the security benefits (Factor 9).

Humans can effectively handle one objective at a time. But, when multiple objectives are to be simultaneously considered, research has shown that the consistency and reliability of such selection decisions become questionable [6, 7]. This limitation has raised the need for selection automation and recommendation methodologies to assist the human operator, a capability gap that our approach aims to fill.

1.2.4 Related work on COA selection

There are some commercial COA selection tools such as patch management systems (e.g. Altiris [8,15]) or reconfiguration management systems (e.g. Redseal [9]). However, they are not designed to incorporate operational context information that is important for selecting the best COAs while maintaining business continuity. The limited information they provide leads to inconsistent subjective decisions by human operators [2, 7], a limitation that we aim to address in our work.

COA selection approaches by researchers such as Sawilla *et al.* use the attack graph context [4,5,14]. Their approaches include assumed cost measures representing the limitation of resources as well as a measure of the security benefit obtained by making a particular selection. However, their approaches do not explicitly include operational context or mission-related factors. Their selections do not address cases where missions or business continuity could be impacted by the implementation of the COAs on operational networks. In addition to utilising the attack graph and resource limitation concepts used by Sawilla *et al.*, our work includes other operational factors in deciding the COAs to select for implementation.

Other researchers have focused their selection methodologies on COA remediation costs [16] or network risk [17]. The former uses lowest cost COAs to recommend graph cuts. The latter selects COAs based on the risks and costs determined from the vulnerability and host importance in the attack paths. However, both approaches do not address operational impacts which are important in defence operational networks. But, we find the use of host importance measures by Hong *et al.* [17] relevant in providing operational context to defended networks. So, we borrowed that concept and applied it to mission impacts in our work.

The work by Kim *et al.* [2] focuses on security event prioritisation, the remediation of which is the same as the COAs we are focusing on in this work. What is important about their work is a prioritisation approach that includes host importance measures as one of their deciding factors. They also take into consideration the asset criticality based on the mission that the asset is part of. We consider these factors to be important in COA selection and include them in our work. Another important aspect of their approach is the use of a modified TOPSIS technique to determine their final prioritisation. The

modified approach allowed them to avoid changes in prioritisation based on different input scores. It also allowed them to compare results across different calculation runs. In addition to TOPSIS's wide acceptance and success in solving MADM problems [1,18–21], its application by Kim *et al.* shows that it is well suited to solve our selection problem. We therefore adopt that approach and similarly incorporate missions and impact data.

1.3 Our TOPSIS approach

1.3.1 TOPSIS

TOPSIS, which was proposed by Hwang and Yoon, is a MADM methodology that selects the best alternative in a multi-attribute problem [11]. The idea is centered on the premise that the best alternative should have the shortest geometric distance from a hypothetical positive ideal solution (the zenith) and longest geometric distance from a hypothetical negative ideal solution (the nadir).

Consider a problem to make prioritised selections from m alternatives $\mathrm{C_i}: \mathrm{i} = 1, \cdots, \mathrm{m}$. Each alternative $\mathrm{C_i}$ is characterised by n factors such that the score for the ith factor of the jth alternative is x_{ij}. These alternative scores are represented by the decision making matrix shown in Table 1.2. The

Table 1.2. The decision matrix.

	Factor 1	**Factor** 2	$\cdots$	**Factor** n
C1	x_{11}	x_{12}	$\cdots$	x_{1n}
C2	x_{21}	x_{22}	$\cdots$	x_{2n}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
Cm	x_{m1}	x_{m2}	$\cdots$	x_{mn}
Weights	w_1	w_2	$\cdots$	w_n

weights w_i represent the overriding preferences of one factor over others.

To determine the relative closeness of the alternatives from the zenith, TOPSIS's first step is to normalise the decision matrix shown in Table 1.2 as follows:

$$z_{ij} = \frac{x_{ij}}{\sqrt{\sum_{j=1}^{m} x_{ij}^2}} \tag{1.1}$$

The normalised decision matrix is then multiplied by the weights for each factor to give the weighted normalised decision matrix such that $v_{ij} = w_i z_{ij}\ \forall i, j : i = 1, \cdots, m,\ j = 1, \cdots, n$.

The zenith $A^+ = \{v_1^+, \cdots, v_m^+\}$ is made up of the best values for each criterion and the nadir $A^- = \{v_1^-, \cdots, v_m^-\}$ is made up of the worst values of each criterion. For example, v_1^+ is the highest value for a maximisation

objective on Factor 1, and v_1^- is the lowest value. Similarly v_2^+ is the lowest value for a minimisation objective on Factor 2, and v_2^- is the highest value.

These concepts are illustrated in Figure 1.2. Given the Euclidean distances

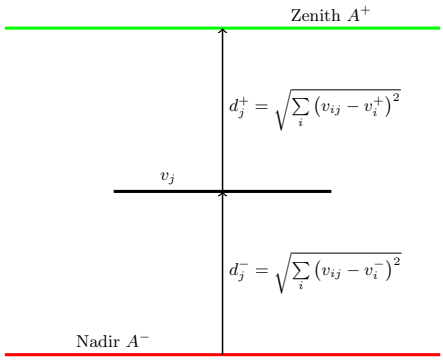

Fig. 1.2. A simplified illustration of the TOPSIS approach.

shown, the relative closeness score t^+ for each alternative to the zenith is calculated from

$$t_j^+ = \frac{\left[\sum_i \left(v_{ij} - v_i^-\right)^2\right]^{\frac{1}{2}}}{\left[\sum_i \left(v_{ij} - v_i^+\right)^2\right]^{\frac{1}{2}} + \left[\sum_i \left(v_{ij} - v_i^-\right)^2\right]^{\frac{1}{2}}} \tag{1.2}$$

This means that selection alternative a is better than b if and only if $t_a^+ > t_b^+$, and indistinguishable if $t_a^+ = t_b^+$.

1.3.2 Our approach

Our model uses TOPSIS to analyse the multiple factors and their corresponding objectives so as to select the best actionable COAs for the given factors. We first populate the decision matrix shown in Table 1.2 with scores from the COA sets that we need to choose from by assigning values to each factor for all the COA alternatives C_i under consideration.

The first three factors are simple counts of the number of SINKs of each type in the COA set. For example, if there are 2 vulnerabilities in the COA set, then the score for the first factor would be 2. Since a network can only be as secure as its least secure assets, the missions impact values are determined by the highest mission impact score of the asset, or set of assets, whose SINK nodes are associated with a COA set [7]. That means, the mission impact for C_i would be the highest mission impact on hosts identified by the vulnerable nodes identified in the COA set. For example, if the COA set points to SINKs on hypothetical nodes 7, 8, and 9 (see Figure 1.1), then the mission score for

that COA set is the maximum score for the missions supported by those three nodes. Although the mission impact score ranges in $[0, 10]$, we used discrete values of 0, 1, 5, 8 and 10 representing *None* (N), *Low* (L), *Medium* (M), *High* (H) and *Very High* (VH) respectively. Such assignments correspond to those used in operational networks [2, 10].

Operators assign scores for the next three factors from known remediation impediments. The cost and rank scores are assigned by GENESIS based on Sawilla's algorithms [4, 14]. To simplify our problem, we assumed that all the scores would be automatically assigned by an autonomous remediation module that aggregates network security data. For military networks, we further assumed that missions data would be readily associated with each asset on the network. In keeping with organisational policy or prevailing security risks, operators can assign relative weights to the factors so that selection preference can be given to some factors over others. For our work, we assume that all factors have equal weights although our model can handle varying them to represent operational preferences.

Before applying TOPSIS, we slightly modified it by changing the zenith and nadir vectors. In their work, Kim *et al.* [2] noted that new input values can change the zenith and/or nadir. Such changes require the recalculation of t^+, which could result in changes to the selection alternatives. They avoided this problem by fixing the values of the zenith and nadir to the maximum (for maximisation) or minimum (for minimisation) possible scores for the zenith and the opposite for the nadir. We use this technique in our work to allow for selection comparisons across multiple sets of COA sets.

We then apply TOPSIS to our decision matrix. Our model, which simultaneously combines all selection objectives, calculates the values of t^+ for each COA set alternative. We then rank the COA sets based on the their relative scores t^+. The COA set with the highest score becomes the first choice for implementation. We tested our model on a simulated experimental network.

1.4 Experimental results

1.4.1 Test data

For test data, we used COA sets that we generated using an existing virtualised lab network prototype, GENESIS, shown in Figure 1.3. The network consists of fully configured virtual hosts running real operating systems and servers with real vulnerabilities. To emulate an operational network, it is made up of three zones, the demilitarised zone (DMZ), the security and corporate zones. Despite its small size, the COAs generated from it are the same as those generated in a large enterprise network, except that the latter would have a significantly larger number of COAs.

We simulate an attacker, located on the Internet, targeting any of the assets on the network. We also assumed that an attacker could target any

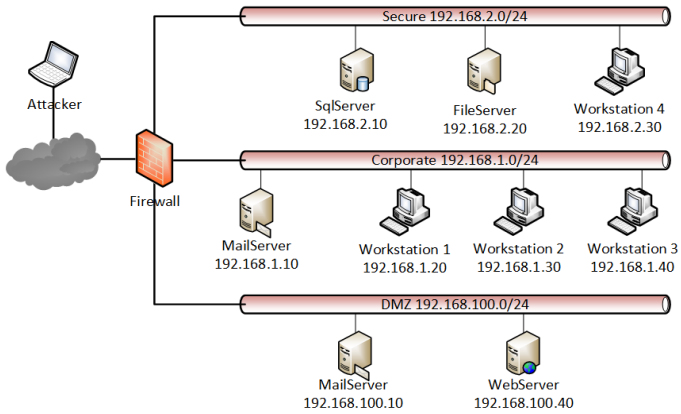

Fig. 1.3. The test network in a virtual environment (adapted from [4]).

asset on the network from any other network host, a reasonable assumption given that attackers can launch multi-step attacks from any other node. This setup allowed us to generate more data for our testing than we could have achieved otherwise. We then arbitrarily assigned attackers and targets on the network and used MulVAL to generate attack graphs for each attacker-goal combination [3–5]. For each combination, we generated COA sets by repeatedly relaxing the remediation budget limits using Sawilla's algorithm [4]. This approach enabled us to generate 120 unique sets of COA sets for our experiments.

A typical set of COA sets generated this way is shown in Table 1.3. The table shows 6 COA sets C_i for $i = 1, \cdots, 6$. When MulVAL [3] generates the

Table 1.3. A set of COA sets generated using MulVAL.

COA set	Nodes in COA set
C_1	[99]
C_2	[99,114]
C_3	[99,114,163]
C_4	[99,114,21,163]
C_5	[99,114,21,29,163]
C_6	[99,114,121]

attack graph, it assigns identification numbers to each SINK node in the COA set (see Section 1.2). These node numbers are represented in square brackets in Table 1.3. For example, the COA set C_1 requires the removal of SINK node 99, which may be patching a software vulnerability. We will use the above set of COA sets in the examples of our results later on.

1.4.2 Survey of experts

In order to analyse the selections of human experts under given scenarios represented by our selection attributes, we carried out a survey of cyber security experts drawn from experienced colleagues. We presented them with the network shown in Figure 1.3, whose assets had been arbitrarily assigned to missions. We assumed that this network is able to support different missions that could be impacted differently by remediation, a reasonable assumption for a typical operational network.

To limit the time spent on the survey, we arbitrarily selected 50 sets of COA sets for the survey. Through a custom survey application, we tasked the experts with completing the survey three independent times. During the survey, they were repeatedly presented with a set of COA sets similar to the one in Table 1.3 together with selection rationales as represented by the values of selection attributes. The experts were then expected to use their knowledge and experience on those rationales to select the best and second best COA sets to implement to improve network security. For example, from Table 1.3, an expert could select C_5 and C_2 as the best and second best COA sets. We recorded the survey results for further analysis.

1.4.3 Decision making with TOPSIS

TOPSIS consistency

We used attribute scores to populate the TOPSIS decision matrix (see Table 1.2). Then we determined the relative closeness score t^+ for each COA set, and selected the set with the highest value of t^+ as the best alternative. We also ranked the rest of the alternatives based on the COA sets' scores. For example, in one scenario, the rankings of the set of COA sets in Table 1.3 were, from best to worst, C_3, C_4, C_2, C_6, C_5 and C_1. However, before we discuss the selections in detail, we analyse the validity and consistency of our TOPSIS approach for the given attributes.

To analyse TOPSIS's consistency and repeatability, we determined how well its solutions satisfied the multiple objectives reflected in the decision matrix. For each objective, we determined how changes to other attributes affect the overall TOPSIS score, and therefore the resulting selections. If TOPSIS is consistent, we would expect the variation in scores to show a monotonic increasing curve for maximisation and decreasing for minimisation objectives. We demonstrated these variations using three arbitrarily selected, but representative, scenarios.

Scenario 1

Using the set of COA sets shown in Table 1.3, we first consider the variations of TOPSIS scores on COA set C_6. We varied the missions impact scores on C_6 (equivalent to mission impacts on an asset on node 121 for example) while

keeping all the other attribute scores constant. The variation of the TOPSIS scores on COA set C_6 are shown as blue diamond shapes in Figure 1.4. The

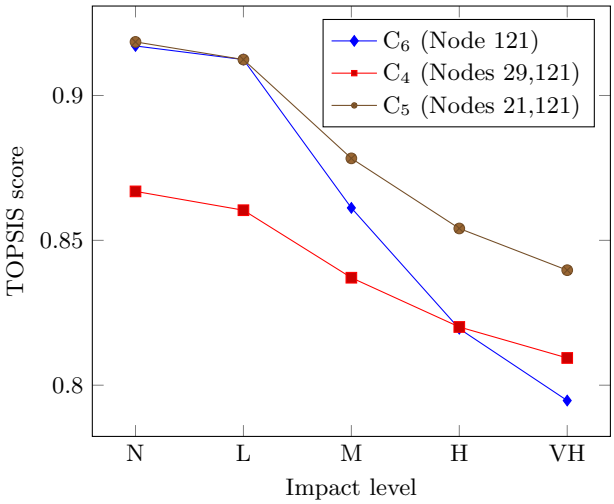

Fig. 1.4. TOPSIS scores for variations in impact levels for COA sets C_5 and C_6 due to changes in impact levels on nodes 21, 29 and 121.

C_6 graph (blue diamonds) shows that as the impact level of the COA set C_6 was increased from N to VH, the TOPSIS score showed a monotonic decreasing trend. This is an expected result, since an increase in the impact should translate into a less favourable alternative, and therefore a lower TOPSIS score.

Scenario 2

In this experiment, we also performed similar analysis on COA sets C_4 and C_5. We simultaneously varied the mission impact scores for C_4, C_5 and C_6 (equivalent to simultaneous mission impacts on assets at nodes 21, 29 and 121). The variations in TOPSIS scores for COAs C_4 and C_5 are respectively shown on red squares and brown circles graphs in Figure 1.4.

Similar to the C_6 (blue diamond) graph, these two graphs are monotonically decreasing with worsening impact scores as would be expected. The curves also show that as the impact scores for COA sets C_4 and C_5 were changed simultaneously by the same value, the resulting TOPSIS scores maintained the superiority of C_5's score over C_4's. Due to the nonlinear nature of t^+, the gap between the curves' scores is not always maintained, although the relative ranking is.

Scenario 3

In the third and final scenario, we analysed the TOPSIS scores on COA set C_6 with variations in impact and rank scores on that COA set. The variations are shown in Figure 1.5. For each graph, the monotonically decreasing shapes are

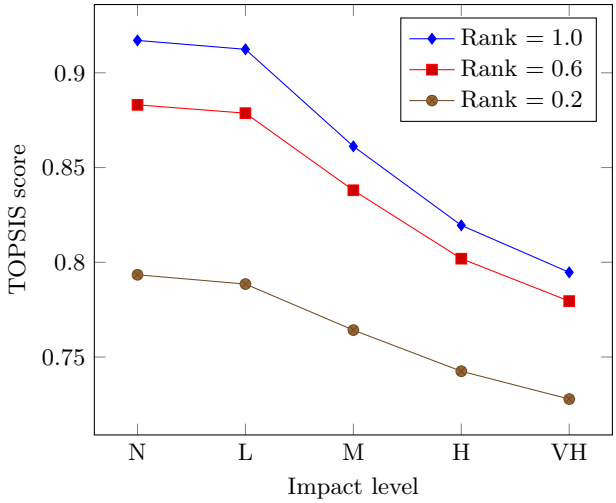

Fig. 1.5. Variations in TOPSIS scores for set C_6 with variations on impact and security benefit scores for the same COA set.

the same as in Figure 1.4–a trend that is expected for the same reasons. For each impact level on C_6, the TOPSIS scores are also monotonically increasing with increasing rank values. These results are also what we expected since the objective on this attribute is to maximise the security benefit score.

The rest of the attributes showed trends pursuant with their decision matrix objectives. This led us to conclude that although we do not have a way to validate these comparisons at this time, they are repeatable and self consistent and, most importantly, meet the objectives for which they were intended.

TOPSIS selection examples

Having determined the consistency of TOPSIS, we performed a number of experiments to show how well it can handle COA selections in an operational environment. We aimed to demonstrate this by showing the different selection options as the values for the COA attributes were changed. The summary of a typical selection ranking using our algorithm is shown in Table 1.4.

The table is divided into three parts represented by circled letters. Each part demonstrates an important aspect of our TOPSIS approach. The first part (*a*) of the table shows ranking variations as the impact values for node

Table 1.4. TOPSIS selections for COAs on nodes with varying impacts. Column VH* also simulates an unavailable patch.

COA set	Selections									
	(a) Impacts for C_6 nodes					(b) Impacts for C_5 and C_6 nodes				(c)
	N	L	M	H	VH	L	M	H	VH	VH*
C_1	6	6	6	5	5	6	6	5	4	3
C_2	4	4	3	3	3	4	4	3	3	2
C_3	2	2	1	1	1	2	2	1	1	✗
C_4	3	3	2	2	2	3	3	2	2	1
C_5	5	5	5	4	4	5	5	6	6	5
C_6	1	1	4	6	6	1	1	4	5	4

C_6 are changed. The second part (b) shows similar ranking variations for simultaneous impact changes for nodes C_5 and C_6, representing similar impacts on nodes 29 and 121. The third part (c) shows the same ranking selections as in the VH impact of the second part of the table, but with no patch for a vulnerability in COA set C_3.

Impacts on C_6 *nodes*

The first part of Table 1.4 shows that when there was no (N) impact to missions, COA set C_6 was the best choice. The same selection was taken at low (L) impact, although C_3 was selected for higher impacts. These selection variations were a result of the changes in TOPSIS scores. The different selections confirm score changes with variations to mission impacts as illustrated in Figures 1.4 and 1.5. In this case the higher mission impact levels for C_6 gave it a lower TOPSIS score, making it a less favourable alternative and C_3 the best choice. The unchanging selections for some impact levels (e.g. from M to VH for C_3) were due to insufficient relative score variations resulting from changes in impact levels in COA set C_6.

Impacts for C_5 *and* C_6 *nodes*

We obtained similar results when we simultaneously changed the impact scores for C_5 and C_6 (equivalent to changing impact scores on node 121). The ranking trend was the same for the lowest impacts (N and L), and mostly the same for high impacts, but significantly different for medium impact (note that the ranking for no impact is the same as in the previous case). This difference was due to insufficient TOPSIS score changes, resulting from changes in mission impacts from low (L) to medium (M), to allow C_3 (second choice) to be selected instead.

No patch for C_3

Finally, using the same scores for the VH impact scores in the previous part, we simulated an unavailable patch for C_3. The COA set C_3 was eliminated from the ranking and COA set C_4 became the best option.

These selection experiments show the capability of our algorithm to select the best actionable COAs satisfying its multiple objectives. This is the consistency and repeatability we expected. The invariancy of the selections observed in some cases are a result of small changes in the TOPSIS score that were not high enough to trigger selection changes. To further study our algorithm's performance, we analyse surveyed experts' selections and compare them with those from our approach.

1.4.4 Experts' selections

As discussed earlier, we asked three experts, A, B, and C, to make preferential selections from 50 sets of COA sets. We asked each expert to complete the survey three times. Each instance of the experts' survey attempt is represented by the number of that attempt. For example, the first, second and third survey attempts by expert A are represented as A_1, A_2, and A_3 respectively. Our objective was to analyse the degree to which our experts' selections agreed with each other and with our TOPSIS approach.

To analyse the degree of agreement, we define an agreement factor s as the ratio of selection agreements m to the total number of sets of COA sets n under consideration [10]. Thus

$$s = \frac{m}{n} \tag{1.3}$$

Similar to correlation measures, we consider agreement factors close to the perfect agreement $s = 1$ as very strong and those close to no agreement ($s = 0$) as very weak.

The results from the comparisons of the three experts' selections are summarised in Table 1.5. In the table, each expert's three selections are compared

Table 1.5. Comparisons of experts' selection agreements.

		Expert selections								
		A_1	A_2	A_3	B_1	B_2	B_3	C_1	C_2	C_3
Experts' selections	A_1	–	74%	72%	46%	52%	46%	68%	66%	68%
	A_2		–	84%	54%	58%	52%	68%	66%	66%
	A_3			–	54%	58%	52%	70%	72%	72%
	B_1				–	72%	72%	56%	58%	56%
	B_2					–	78%	58%	56%	58%
	B_3						–	62%	60%	62%
	C_1							–	88%	88%
	C_2								–	92%
	C_3									–
TOPSIS		64%	68%	74%	60%	54%	58%	84%	94%	84%
All		44%								
TOPSIS		32%								

against the other experts'. For example, expert A's first survey selections A_1, were compared against their second A_2 and third A_3 selections, as well as those performed by B and C. The third row from the bottom shows the agreements of each expert with our TOPSIS approach. The next row shows the simultaneous agreement levels of all the experts. The simultaneous agreements of all experts and our TOPSIS approach is shown in the last row.

The self-consistency of over 70% among the experts is strong for human experts in a multi-attribute problem. The experts showed their highest self-consistency between the second and third selections. This is an expected result since the experts would have been more familiar with the alternatives during the final two survey attempts than during the first.

The agreements among different experts is not as strong as the experts' self-consistency. The highest agreement was between A's and C's selections, at about 70%. In total, all experts' selections are in simultaneous agreement in 44% of the cases, which is low. Such results underscore the need for a consistent approach to prioritise the COAs in an environment that could be manned by many experts or for an application to autonomous defence modules in automated CND.

With all the inconsistencies in the experts' selections, it is difficult, if not impossible, to determine if the selections are correct. So, we further analysed the selections against the systematic TOPSIS rankings that we have just determined to be consistent in its selections (although we have no way to validate its accuracy at this time). As shown in Table 1.5, expert C has the highest agreement with the TOPSIS ranking. However, all experts' selections are in simultaneous agreement with the consistent TOPSIS selections in only 32% of the cases, implying a collective expert consistency of 32%. This is not a surprising result since the experts could choose COAs that are off the maximal selection (provided by TOPSIS) by varying levels of magnitude.

An example of these selection inconsistencies are shown in Table 1.6. In

Table 1.6. Analyst and TOPSIS selections for COAs on nodes with varying impacts.

COA	Selections													
	Low impact							High impact						
set	A_1	A_2	B_1	B_2	C_1	C_2	T	A_1	A_2	B_1	B_2	C_1	C_2	T
C_1							6							5
C_2							4							3
C_3							2	✓	✓					✓
C_4							3				✓			2
C_5							5					✓	✓	6
C_6	✓	✓	✓	✓	✓	✓	✓			✓				5

the table, we show the first two selections by each expert and compare them to the TOPSIS selection (T). We also show the TOPSIS rankings for selection comparisons. In this example, the experts' selections at low impact levels were

all consistent. However, at a high impact level, the inconsistency is apparent. While A and C were self-consistent, C's selection did not match the TOPSIS selection as A did. B's selections did not match each other nor TOPSIS's, again reinforcing the need for a methodology, such as demonstrated by our approach, that could either be used in automated CND or provide consistent remediation support to network security operators.

1.4.5 Discussion and possible future work

Our work showed the inconsistency in the selections by human experts even when they are presented with the same information. In practice, this is not unexpected as it reflects each expert's security preferences, which are based on their knowledge and experience. Unfortunately, such inconsistencies may result in errors when the network can least afford them. In addition, the inconsistencies make it hard to model and incorporate COA selections into automated CND systems such as ARMOUR. This reinforces the need for consistent systematic approaches such as ours, which can be integrated with automated CND systems.

In the absence of a ground truth, it is not possible to validate the solutions from our approach. But, the approach is self-consistent and its selections and rankings are based on systematic measures that represent a combination of multiple objectives that reflect the security requirements in an operational environment. We therefore argue that, given TOPSIS's high success rates in solving MADM problems [1, 2], its solution is a close representation of the multiple objectives we originally identified in our decision matrix. Compared to the inconsistent manual process by human operators, our approach is a good candidate for applications in autonomous network security modules in automated CND.

There are two main possible applications of our work. The first application is for autonomous COA selection decision making in automated CND systems. The approach would get security context data from the defended network environment and aggregate it with operational data to determine selection measures that would help the system to select the best actionable COA under the given conditions. The second possible application is to train cyber security experts in making consistent selection decisions. The system could be used to compare its selections against experts' and the results used to train operators or identify areas needing improvement. It could also be used to identify and correct operators' subjective selection biases in both the selection factors and the operators.

One deficiency from our approach is that it is difficult to quantify a selection miss to determine how far it is from the correct result. The multi-factor score difference could be so minor to be insignificant or so big that it could be a show stopper. All our approach does is to determine a measure of closeness to an ideal solution that meets our objective. We recommend future work to

look into variance measures that reflect how far a selection is from a perceived ideal one.

Although our study is based on a limited number of factors that we determined using our network security expertise, it has produced promising results showing the relative ranking of COAs. While our approach is supposed to work with a broader set of factors than the ones we used in our work, we did not test it as it was not part of our study and we do not know how scalable that expansion would be. We therefore recommend future work to study possible additional factors that could influence COA selection decisions. The study could use consensus-based rating techniques, such as the Delphi method [22], with security experts to determine and prioritise those factors. Factor prioritisation weights can then be assigned accordingly (see Section 1.3).

To improve the accuracy and comparison of our algorithm, we could investigate the aggregation of experts' decisions instead of analysing them as individual decisions. That way, we can compare our algorithm against the consensus decisions of our experts. Methodologies such as the Delphi method [22] could also be used to determine experts' consensus selections. We recommend further studies on whether such consensus selections could be considered as the ground truth against which to compare our algorithm.

Our work selected to use the TOPSIS approach based on its reputation in solving MADM problems similar to ours. However, other techniques such as simple additive weighting (SAW), and analytical hierarchy process (AHP), for example, could be considered as possible solution candidates as well. In addition to consistency and repeatability, the techniques can then be evaluated based on other measures such as simplicity of use, time, and understandability for example. We leave such investigations to possible future work.

Our work was carried out on a simulated lab network with real vulnerabilities. Practical networks are more complicated than the GENESIS network that we used. We did not have data to test our approach with such complex networks, so it is unclear how our approach would perform under such conditions. We expect the number of COA sets to be significantly large, requiring multiple sizeable runs of our algorithm. This could take more time than in the small network used in our study. Such additional time could impact decision making in automated systems where consistent results are needed promptly. We therefore recommend future research to investigate the impact of applying our algorithm on large operational networks.

1.5 Conclusions

In this work we have shown how inconsistent human operators can be when asked to make course of action (COA) selections even if they are provided with the same information. This is undesirable for automated computer network defence (CND), which requires consistent and repeatable COA selections

based on identified contextual and security information. To correct this inherent weakness in human decision making, we have developed a multi-attribute decision-making (MADM) algorithm to select actionable COAs for the effective security of a defended network. For its decisions, our algorithm uses network security, operational and contextual factors that we believe to be the most important for COA selection and prioritisation.

We have shown our approach produces repeatable and consistent selections based on quantifiable security measures from the network and the operational environment. Our solution is based on the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) technique that has found significant successes in solving MADM problems resembling ours. Therefore, in the absence of a ground truth to validate our approach, we argue that TOPSIS's repeatable and consistent solution of our MADM COA selection problem as characterised by the multiple selection objectives, will effectively provide network security that meets those goals. Our results' repeatability and self consistency have been shown to outperform human experts', making our model a good candidate for automated CND applications that require consistent and reliable solutions based on the security environment.

Our approach could also contribute to the efficient utilisation of resources in an operational environment. The low levels of simultaneous agreements among experts show that sole reliance on human expertise could contribute to resource wasting as experts would need to expend more time to resolve their selection disagreements before implementing remediation measures. Such time could be best utilised in performing other security tasks if the technique we report in this work is exploited into an operational tool.

References

1. A. Kim and M. H. Kang, "Determining asset criticality for cyber defense," Naval Research Laboratory., Tech. Rep. NRL/MR/5540–11-9350, 2011.
2. A. Kim, M. H. Kang, J. Z. Luo, and A. Velasquez, "A framework for event prioritization in cyber network defense," Naval Research Laboratory, Tech. Rep. NRL/MR/5540–14-9541, 2014.
3. X. Ou, S. Govindavajhala, and A. W. Appel, "Mulval: A logic-based network security analyzer." in *USENIX security*, 2005.
4. R. Sawilla and C. Burrell, "Course of action recommendations for practical network defence," Defence Research and Development Canada, Tech. Rep. DRDC Ottawa TM 2009-130, 2009.
5. R. Sawilla and D. Skillicorn, "Partial cuts in attack graphs for cost effective network defense," in *HST 12: 2012 IEEE International Conference on Technologies for Homeland Security*, 2012, pp. 291–297.
6. G. A. Miller, "The magical number seven, plus or minus two: Some limits on our capacity for processing information." *Psychological Review*, vol. 101, no. 2, p. 343, 1994.

7. S. Miller, S. Appleby, J. M. Garibaldi, and U. Aickelin, "Towards a more systematic approach to secure systems design and analysis," *International Journal of Secure Software Engineering*, vol. 4, no. 1, pp. 11–30, 2013.
8. Symantec, "It analytics 7.1 for altiris it management suite from symantec," Symantec, Tech. Rep., 2013.
9. RedSeal Networks. (2014, July) Security target. (Access Date : 26 January 2016). [Online]. Available: https://www.redseal.net
10. M. Dondo, "A neural network approach for cyber security course of action selection," Defence Research and Development Canada, Tech. Rep. DRDC-RDDC-2016-R269, 2016.
11. C.-L. Hwang and K. Yoon, *Multiple attribute decision making: methods and applications a state-of-the-art survey*. Springer Science & Business Media, 2012, vol. 186.
12. C. McKenzie, "GENESIS: Integrated end-to-end decision support for computer network defence (proof of concept), design and architecture document," Defence Research and Development Canada, Tech. Rep. DRDC Ottawa CR 2011-009, 2011.
13. R. E. Sawilla and D. J. Wiemer, "Automated computer network defence technology demonstration project (ARMOUR TDP): Concept of operations, architecture, and integration framework," in *2011 IEEE International Conference on Technologies for Homeland Security (HST)*, Nov 2011, pp. 167–172.
14. R. E. Sawilla and X. Ou, "Identifying critical attack assets in dependency attack graphs," in *European Symposium on Research in Computer Security*. Springer, 2008, pp. 18–34.
15. Symantec. (2016) Symantec patch management solution powered by altiris technology. (Access Date : 26 January 2016). [Online]. Available: http://www.symantec.com/products
16. M. Alhomidi and M. Reed, "Finding the minimum cut set in attack graphs using genetic algorithms," in *ICCAT 2013: 2013 International Conference on Computer Applications Technology*, Jan 2013, pp. 1–6.
17. J. Hong, D. S. Kim, and A. Haqiq, "What vulnerability do we need to patch first?" in *DSN 2014: 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, June 2014, pp. 684–689.
18. S. Chakraborty and C.-H. Yeh, "A simulation based comparative study of normalization procedures in multiattribute decision making," in *Proceedings of the 6th Conference on Artificial Intelligence: Knowledge Engineering and Data Bases*, vol. 6, 2007, pp. 102–109.
19. F. E. Boran, S. Gen, M. Kurt, and D. Akay, "A multi-criteria intuitionistic fuzzy group decision making for supplier selection with TOPSIS method," *Expert Systems with Applications*, vol. 36, no. 8, pp. 11 363–11 368, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417409002772
20. H. Safari, E. Khanmohammadi, A. Hafezamini, and S. S. Ahangari, "A new technique for multi criteria decision making based on modified similarity method," *Middle-East Journal of Science and Research*, vol. 14, no. 5, pp. 712–719, 2013.
21. M. Velasquez and P. T. Hester, "An analysis of multi-criteria decision making methods," *International Journal of Operations Research*, vol. 10, no. 2, pp. 56–66, 2013.
22. H. A. Linstone and M. Turoff, "The delphi method," *Techniques and applications*, vol. 53, 2002.

DOCUMENT CONTROL DATA

(Security markings for the title, abstract and indexing annotation must be entered when the document is Classified or Designated)

1. ORIGINATOR (The name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g., Centre sponsoring a contractor's report, or tasking agency, are entered in Section 8.)

DRDC – Ottawa Research Centre
Defence Research and Development Canada
3701 Carling Avenue
Ottawa, Ontario K1A 0Z4
Canada

2a. SECURITY MARKING (Overall security marking of the document including special supplemental markings if applicable.)

CAN UNCLASSIFIED

2b. CONTROLLED GOODS

NON-CONTROLLED GOODS
DMC A

3. TITLE (The complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S, C or U) in parentheses after the title.)

A cyber security course of action selection approach for automated computer network defence

4. AUTHORS (last name, followed by initials – ranks, titles, etc., not to be used)

Dondo, Maxwell; 2017), WorkshoponApplicationsandTechniquesinCyberSecurity(ATCS; 2017, DateofPublicationfromExtPublisher:October

5. DATE OF PUBLICATION (Month and year of publication of document.)	6a. NO. OF PAGES (Total containing information, including Annexes, Appendices, etc.)	6b. NO. OF REFS (Total cited in document.)
October 2017	20	22

7. DESCRIPTIVE NOTES (The category of the document, e.g., technical report, technical note or memorandum. If appropriate, enter the type of report, e.g., interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

External Literature (P)

8. SPONSORING ACTIVITY (The name of the department project office or laboratory sponsoring the research and development – include address.)

DRDC – Ottawa Research Centre
Defence Research and Development Canada
3701 Carling Avenue
Ottawa, Ontario K1A 0Z4
Canada

9a. PROJECT OR GRANT NO. (If appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant.)	9b. CONTRACT NO. (If appropriate, the applicable number under which the document was written.)

10a. ORIGINATOR'S DOCUMENT NUMBER (The official document number by which the document is identified by the originating activity. This number must be unique to this document.)	10b. OTHER DOCUMENT NO(s). (Any other numbers which may be assigned this document either by the originator or by the sponsor.)
DRDC-RDDC-2017-P093	

11a. FUTURE DISTRIBUTION (Any limitations on further dissemination of the document, other than those imposed by security classification.)

Public release

11b. FUTURE DISTRIBUTION OUTSIDE CANADA (Any limitations on further dissemination of the document, other than those imposed by security classification.)

12. ABSTRACT (A brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual.)

13. KEYWORDS, DESCRIPTORS or IDENTIFIERS (Technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus, e.g., Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

[Keywords, Descriptors or Identifiers - if the document is sensitive provide object markers for individual keywords]