Improving System Reliability against Rational Attacks under Given Resources

Li Wang, *Student Member, IEEE,* Shangping Ren, *Senior Member, IEEE,* Bogdan Korel, *Senior Member, IEEE,* Kevin Kwiat, and Eric Salerno

***Abstract*—System reliability has always been a challenging issue for many systems. In order to achieve high reliability, redundancy and voting schemes are often used to tolerate unintentional component failures. For unintentional failures caused by, for instance, normal wear-outs, hardware failures, or software bugs, etc., adding more redundancies often improves a system's reliability. However, when attack-caused failures exist, the number of redundant components and the number of participating voting entities may not be positively proportional to system reliability. In this paper, we study system reliability and system defense strategies when the system is under rational attacks. In particular, we analyze how defense and attack strategies may impact system reliability when both the defender and attacker are given a fixed amount of resources that can only be used for adding camouflaging components or enhancing existing components' cyber protection by defenders, or selecting a subset of components to attack by attackers, respectively. We also present an algorithm to decide the optimal defense strategy in fighting against rational attacks.**

***Index Terms*—System Reliability, Attacker-defender Problem, Voting Strategy, Resource Allocation.**

I. Introduction

RELIABILITY represents a system's ability to work correctly and continuously [1]. This property is critical for many systems [2]. Unfortunately, the reliability of individual components (a component can be hardware, software, or a composition of hardware and software that work together to perform a task) of a given system is rarely one [3].

To improve system reliability in the presence of possible component failures, redundancy and voting schemes are often used to tolerate natural-caused component failures. Generally speaking, when the reliability of a single component (replica) is high, the more redundancies that are added to the system, the more reliable the system behaves [4], [5]. However, when the system is under intentional cyber attacks, voting components may be compromised [1] by the attacks, causing the system to produce incorrect results. In fact, when both the reliability of components and the possibility of components being compromised are taken into considerations, the number of components that participate in a voting process has significant impact on the performance of the voting algorithms, i.e., the reliability of the system [3]. We use an example to explain the point.

Example 1: Assume a system consists of nine replicas that provide the same functionality, but with different implementations. The reliability of each replica is 0.90. If only five out of the nine replicas are used in deciding the final result through a majority voting, the system reliability is

$$P = \sum_{j=\lceil \frac{N_v+1}{2} \rceil}^{N_v} \binom{N_v}{j} p^j (1-p)^{N_v-j} \quad (1)$$
$$= \sum_{j=3}^{5} \binom{5}{j} 0.9^j (1-0.9)^{5-j} = 0.9914$$

where N_v is the number of voting components and p is the component's reliability.

Fig. 1 depicts the relationship between the system reliability and the number of voting components. As shown in Fig. 1, the more voters used, the higher the system reliability. The maximum reliability that the system can achieve is when all nine available components participate in the voting process (i.e., when $N_v = 9$), which is 0.9991.

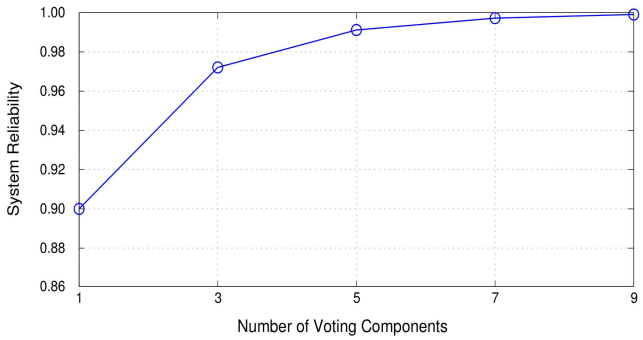

Fig. 1. The relationship between system reliability and the number of voting components ($p = 0.90$)

However, if the system is under attack, its components may be compromised. Under such a scenario, it is no longer the case that the more voters the system has, the more reliable the system is. As an example, assume five out of the nine components are compromised. Under this case, if we still choose all nine components to vote, the reliability of the system is 0 because the majority of the voters are compromised. However, if we choose a single component as a voting component, the reliability of the system is $\frac{4}{9} \times 0.90 = 0.40$, which is better than the case where all nine components are used. □

Li Wang, Shangping Ren, and Bogan Korel are with the Department of Computer Science, Illinois Institute of Technology, 10 West 31st Street, Chicago, IL 60616 USA. Email: {lwang64, ren, korel}@iit.edu.

Kevin Kwiat is with the Cyber Science Branch, Air Force Research Laboratory, Rome, NY 13441 USA. Email: kwiatk@rl.af.mil.

Eric Salerno is with the Department of Mechanical and Aerospace Engineering, University at Buffalo, 12 Capen Hall, Buffalo, New York 14260 USA. Email: esalerno@buffalo.edu.

[1]By a component being compromised, we mean that the component's reliability is significantly decreased. For analysis simplicity, we assume it decreases to 0.

The example shows that in the presence of cyber attacks, system reliability not only depends on the level of redundancy, but it is also affected by the number of components in the voting process and whether these components are compromised.

In order to reduce the probability of a system being compromised, we have to increase the difficulty for an attacker to launch successful attacks if such attacks cannot be fully prevented. One way to increase the difficulty of a successful attack is to enhance the cyber protection of existing system components. Another solution is to decrease the probability that an attacker would strike the system's voting components.

Our earlier work [6] presented an information hiding technique to avert attackers from quickly identifying the difference between critical and non-critical components. In addition, a defender can dynamically select different sets of voting components, and hence make it more difficult for attackers to quickly determine the roles of different components in the system. In this paper, we assume attackers cannot distinguish the differences among components, hence adding camouflaging components, such as installing honeypots [7], decreases the probability of a voting component being attacked. For easy discussion, we assume that enhancing component protection and adding camouflaging components are the two approaches a defender is to use with the available resources. However, the technique presented in the paper is not limited to two defending approaches.

Based on Levitin et al. [8], [9], [10], [11], it is known that between a defender and an attacker, the one who invests more resources on a component wins that component (i.e., the component survives the attack, or is compromised by the attack). We revisit Example 1 below.

Example 2 (Example 1 Revisited): Assume the number of components and their reliability are the same as given in Example 1, both the attacker and defender are given 18 units of resources, and the cost for creating a camouflaging component is 2 units of resources. In addition, we assume the attack is random, but the attacker can make rational decisions in selecting the number of components to attack. By rational, we mean the attacker can always take the most favorable strategy.

Assume the attacker chooses six components to attack and evenly distributes his/her resources on the selected components; while the defender allocates his/her resources to protect all nine components which all participate in the voting process. In this case, the defender allocates $18/9 = 2$ units of resources to protect each component, while the attacker allocates $18/6 = 3$ units of resources on each of the selected components which are more than what the defender has put. Therefore, based on [8], [9], [10], [11], all six components being attacked are compromised. The majority voting from the nine components results in 0 reliability.

However, suppose the attacker's strategy remains the same, but the defender changes his/her strategy to create three camouflaging components, and the remaining resources are allocated to protect three components that are chosen as voters. In this case, as all three protected components have $(18 - 2 \times 3)/3 = 4$ units of defense resources, they survive the attacks. As these protected components are the only voting components, according to (1), the system reliability is 0.9720.

On the other hand, if the defender keeps the winning strategy, but the attacker changes his/her strategy to attack only four components, the system reliability reduces to 0.6598. The detailed procedure of calculating the system reliability for this scenario is discussed in Section IV. □

From this example, we can see that the best strategy for the defender depends on how the attacker allocates his/her resources; similarly, the best strategy for the attacker also depends on how the defender allocates his/her resources and how many components are chosen to vote.

The main contributions of the paper are: (1) formal analysis of the relationship between attack and defense strategy, and how they affect system reliability, (2) development of an algorithm to determine the optimal defense strategy against rational attacks, and (3) an experimental study of how the defense and attack resources impact the defender's strategy.

The rest of the paper is organized as follows. Section II discusses related work. In Section III, we first present the system model and list the assumptions this paper is built upon. Based on the model and the assumptions, we formulate the problem of improving system reliability against rational attacks under given resources. In Section IV, we analyze system reliability under given defense and attack strategies. Section V provides an algorithm to determine the optimal defense strategy against rational attacks. The experimental results are shown and discussed in Section VI. Finally, we point out future work in Section VII.

II. Related Work

How to improve system reliability in the presence of attacks has been intensely studied from different perspectives. For instance, Bier et al. have studied *series and parallel* systems and showed that the optimal resource allocation for defenders not only depends on the structure of the system and the cost-effectiveness of component protection investments, but also on the adversary's goals and constraints [12], [13]. Yalaoui et. al. have considered redundant components in *series* systems and proposed a dynamic programming method to calculate the minimum cost required for a system to satisfy the minimum reliability requirement [14].

Levitin and Hausken have done substantial work in the area of system reliability and resource allocations. In particular, they consider *series and parallel systems*, *series systems of parallel subsystems*, and *parallel systems of series subsystems*, and analyze system reliability when systems have or do not have budget constraints in [15], [16], [17], [18]. They further analyze how to allocate resources between deploying camouflaging components and enhancing component protection when only one component needs to be protected against attacks [8]. In [10], the approaches of protection and redundancy are provided to reduce the expected damages caused by attacks. The vulnerability of each system element is determined by an attacker-defender contest success function, and the expected damage caused by the attack is evaluated as the system's unsupplied demand. While in [9], they propose three approaches to minimize system damage when both the defender and attacker have limited resources, and illustrate

how both the defender and attacker choose their strategies when the contest intensity changes.

The main difference between the work presented in the paper and those discussed is that, in our work, the reliability of the system does not depend on the system's structure nor on the number of uncompromised system components, but instead it is decided by a nonempty subset of the components in a system that form the voting components. This is in contrast to the work discussed above where the system reliability either depends on the system structure or the amount of uncompromised components.

In [19], Hardekopf et al. propose a decentralized voting algorithm that improves system dependability and protects the system from faults and hostile attacks. Tong et al. [20] show how to choose optimal weight assignments for the majority voting strategy in the system and also propose new effective vote assignment algorithms which aim to maximize the system reliability. Davcev proposes a dynamic weighted voting scheme for consistency and recovery control of replicated files in distributed systems [21]. Additional weighted voting schemes are discussed in [22], [23], [24].

There are two differences between our work and those discussed above. First, in our system model, a voting scheme as outlined in [3] is employed. However, different from [3], we extend Random Troika to encompass, if needed, up to n components to form the voters. Second, the work in [19], [20], [21] considers the performance of the algorithm when the number of system components is fixed; while in this paper, the number of composing components can be changed under different resource allocation strategies, which may lead to a change in the voting strategy to achieve a higher reliability.

In [25], Vanderbei presented a Linear Programming approach to solving the two-person zero-sum game problem [26]. However, this approach cannot be applied to solve our attacker-defender problem. This is because his approach assumes the game will be played multiple times, and it allows the probability of strategy selection to be any real number between 0 and 1. For example, suppose a player has 3 strategies, the probability of selecting these three strategies is 0.3, 0.3, and 0.4, respectively, and game is played n times. In these n times, strategy 1, strategy 2, and strategy 3 are chosen $0.3n$, $0.3n$, and $0.4n$ times, respectively. The order for selecting the strategies does not make a difference. For the attacker-defender problem this paper addresses, the game can only be played once, and we must determine which strategy should be chosen.

Our earlier work focused on using an information hiding approach to prevent the attackers from quickly identifying the location of critical components in the system [6], deciding optimal resource allocation for improving system reliability under random attack [27], [28], and determining a voting strategy for a set of clusters when they are under rational attacks [29]. The current work differs from the previous work in two aspects. First, the system models are different. In this paper, the system reliability depends on the reliability of a set of selected voting components and whether they are compromised or not, while in [27] and [28], the system reliability is the probability that *all* critical components survive the attacks, and [29] aims to maximize the overall reliability of a set of clusters rather than an individual system. Second, the protection approaches are different. The previous work either considers the voting mechanism or protection approaches (i.e., creating camouflaging components, or enhancing component protection, etc.), while the work presented in this paper considers a more comprehensive approach that integrates the voting mechanism with the protection approaches.

III. Assumptions and Problem Definition

Before presenting the formal description of the system model and its assumptions, we first introduce the notations to be used throughout the paper.

N_s	number of system components
N_c	number of camouflaging components
N_p	number of protected components
N_v	number of voting components
N_a	number of attacked components
N_a^v	number of attacked voting components
N_v^f	number of compromised voting components
N_a^{pv}	number of attacked protected voting components
R_d	total defense resources
R_a	total attack resources
r_d	amount of defense resources on each protected component
r_a	amount of attack resources on each attacked component
C	cost for creating a camouflaging component
p	component's reliability
P	system reliability
M	reliability matrix
$\overrightarrow{X}$	defense strategy selection vector
$\overrightarrow{Y}$	attack strategy selection vector
S_d	number of defense strategies
S_a	number of attack strategies

A. System Model and Assumptions

We assume a system consists of N_s diverse replicas, and the reliability of each replicated component is p. The system's result is decided through a majority voting among a subset of these components, called voting components or voters. We use N_v to denote the number of voting components, where $N_v \leq N_s$. The system reliability P is defined as the probability that a correct result is obtained through a majority voting among voters.

As we assume replicas have the same reliability, therefore, if the sizes of randomly selected voting groups are the same, the reliability of voting results is the same. For instance, the two different selections of five voting components shown in Fig. 2 produce the same system reliability.

Assume a system defender is given a fixed amount of defense resources, R_d, which can be used to create camouflaging components or to enhance component cyber protection, and the cost for creating a camouflaging component is C, and N_c camouflaging components are created, where $N_c \times C \leq R_d$.

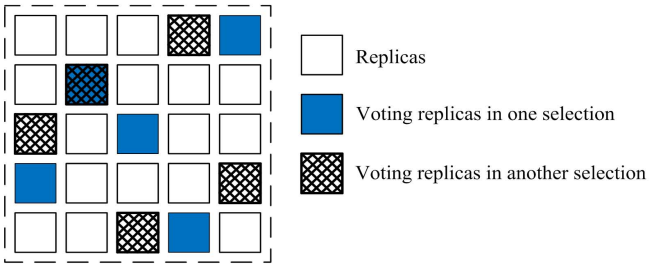

Fig. 2. Different selections of voting components

The remaining defense resources ($R_d - N_c \times C$) are evenly distributed to enhance the cyber protection of a subset of system components, called protected components N_p, $0 \leq N_p \leq N_s$. Hence, the resources r_d used on each protected component is given by (2).

$$r_d = \begin{cases} \frac{R_d - N_c \times C}{N_p} & \text{if } N_p > 0 \\ 0 & \text{if } N_p = 0 \end{cases} \tag{2}$$

As the attacker cannot distinguish the differences between unprotected, protected, camouflaging, or voting components, the selection of components to attack is random. However, the attacker can intelligently decide the number of components to attack. Assume the total amount of resources that an attacker has is R_a and they are evenly distributed to attack a subset of components N_a ($1 \leq N_a \leq N_s + N_c$), the amount of resources allocated to each attacked component is r_a

$$r_a = \frac{R_a}{N_a} \tag{3}$$

Based on Levitin et al. [8], [9], [10], [11], the attack success probability on a single component can be modeled by a contest success function given in (4), where r_a and r_d are the amount of resources invested by contesters, i.e., an attacker and a defender, respectively, and P_a is the attack success probability.

$$P_a = \frac{(r_a)^m}{(r_d)^m + (r_a)^m} \tag{4}$$

When the contest intensity indicator $m = +\infty$, the attack success probability function (4) reduces to "winner-takes-all", i.e., whoever invests more effort wins the game. For reference purpose, we restate the conclusion from [8], [9], [10], [11] as an axiom.

Axiom 1: Assume the amount of resources invested by an attacker and a defender on a component is r_a and r_d, respectively. If $r_a > r_d$, the attacker compromises the component; otherwise, the component survives the attack. □

When fighting against rational attacks, the system defender cannot guarantee that the attacker does not know his/her resource information and defense strategies. We consider the defender assumes the worst-case scenario [30], i.e., the attacker knows the defender's resource information and defense strategies.

For the defender's knowledge of the attacker's resources, we again assume a worst-case where the attacker, with complete knowledge of the defender's resources and strategies, can fully exploit any vulnerability that exists. However, the attacker is not all-powerful because the defender is assumed to be a knowledgeable practitioner of information assurance principles. An upper bound on attack resources (i.e., R_a) would be a combination of favorable-for-the-attacker values of the following: requisite attacker skill-level; time budget for the attack; and the computation and communication expenditures for an attack. This upper bound, while optimistic for the attacker, is pessimistic for the defender.

In summary, we make the following assumptions regarding the defender and attacker's knowledge, which are similar to those made in [12], [13]:

Public information shared by the defender and the attacker:

1) the amount of resources that the defender and the attacker have, i.e., the value of R_d and R_a.
2) the cost for creating a camouflaging component, i.e., the value of C.
3) the number of system components and camouflaging components, i.e., N_s and N_c.
4) the reliability of a system component without protection hardening, i.e., the value of p.

Private information:

1) the defender does not know which components are currently being attacked.
2) the attacker cannot differentiate unprotected, protected, camouflaging, and voting components.
3) the attacker does not know which components are voting components.

Since the attacker randomly selects a subset of components to attack, for each component, the probability of being attacked is the same. Therefore, if the number of voting components is no larger than the number of protected components, it is obvious, from the defender's perspective, that the voting components should only be chosen from the protected set.

Based on the above discussion, we formulate the problem of improving system reliability against rational attacks as below:

Problem 1: Given a system with N_s redundancy and R_d defense resources, where the cost for creating a camouflaging component is C, determine the number of camouflaging, protected, and voting components, i.e., N_c, N_p, and N_v, respectively, so that the system reliability is maximized in the presence of a rational attacker who has R_a resources and makes rational decisions in choosing the number of components, N_a, to attack so that the system reliability is minimized. More precisely, given N_s, R_d, C, and R_a, decide N_c, N_p, N_v, and N_a, such that

$$\max_{\{N_c, N_p, N_v\}} \min_{N_a} P(N_c, N_p, N_v, N_a) \tag{5}$$

where $P(N_c, N_p, N_v, N_a)$ is the system reliability under the given values. □

The next two sections will discuss how the problem is addressed.

IV. System Reliability under Given Defense and Attack Strategies

Assume the defender has made the decisions on the value of N_c, N_p ($0 \leq N_p \leq N_s$), N_v ($1 \leq N_v \leq N_s$) and the selections

of protected and voting components; also the attacker has made his/her decision on the value of N_a ($1 \leq N_a \leq N_s+N_c$) and the selection of N_a components.

Clearly, if an attack strikes on a camouflaging component or a system component that does not participate in the voting process, the attack has no impact on the reliability of the system. Hence, we only need to consider the attacks that target the voting components. Assume N_a^v out of N_a attacked components are voting components, the range of N_a^v is

$$\max\{0, N_a - (N_s + N_c - N_v)\} \leq N_a^v \leq \min\{N_a, N_v\} \quad (6)$$

The probability, $\delta(N_a^v, N_a)$, that the attacker targets exactly N_a^v voting components is

$$\delta(N_a^v, N_a) = \frac{\binom{N_v}{N_a^v}\binom{N_s+N_c-N_v}{N_a-N_a^v}}{\binom{N_s+N_c}{N_a}} \quad (7)$$

In order to obtain the system reliability, we need to know the number of compromised voting components (N_v^f) in a voting process. From Section III, we know the amount of resources a defender and an attacker spend on a voting component, i.e., formula (2) and (3), respectively. Based on Axiom 1, if $r_a > r_d$, each voting component being attacked is compromised, i.e., $N_v^f = N_a^v$; or none of the protected voting components are compromised, but unprotected voting components are, that is $N_v^f = N_a^v - N_a^{pv}$, where N_a^v is the number of attacked voting components, and N_a^{pv} is the number of attacked protected voting components. In other words, we have

$$N_v^f = \begin{cases} N_a^v & \text{if } r_a > r_d \\ N_a^v - N_a^{pv} & \text{if } N_v > N_p \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

Let $\varphi(N_v^f, N_v)$ denote the probability that a correct result is obtained through a majority voting when N_v^f components are compromised, and we have

$$\varphi(N_v^f, N_v) = \sum_{j=\lceil \frac{N_v+1}{2} \rceil}^{N_v-N_v^f} \binom{N_v - N_v^f}{j} p^j (1-p)^{N_v-N_v^f-j} \quad (9)$$

According to the system model, for a defender, if $N_v \leq N_p$, i.e., the number of voting components is smaller than the number of protected components, there will be no unprotected components participating in the voting, i.e., $N_a^{pv} = N_a^v$. However, if $N_v > N_p$, not all voting components are protected, and the range of N_a^{pv} is

$$\max\{0, N_p + N_a^v - N_v\} \leq N_a^{pv} \leq \min\{N_p, N_a^v\} \quad (10)$$

The probability, $\theta(N_a^{pv}, N_a^v)$, that N_a^{pv} out of N_a^v attacked voting components are protected components is given by (11).

$$\theta(N_a^{pv}, N_a^v) = \frac{\binom{N_p}{N_a^{pv}}\binom{N_v-N_p}{N_a^v-N_a^{pv}}}{\binom{N_v}{N_a^v}} \quad (11)$$

As the number of attacked voting components N_a^v varies from $\max\{0, N_a - (N_s + N_c - N_v)\}$ to $\min\{N_a, N_v\}$, and if $N_v > N_p$, the number of protected components N_a^{pv} in the N_a^v attacked voting components ranges from $\max\{0, N_p + N_a^v - N_v\}$ to $\min\{N_p, N_a^v\}$, otherwise it becomes N_a^v, otherwise. Therefore, the system reliability is

$$P(N_c, N_p, N_v, N_a) =$$

$$\begin{cases} \sum_{N_a^v=\max\{0,lb\}}^{\min\{N_a,N_v\}} \delta(N_a^v, N_a) \times \varphi(N_a^v, N_v) & \text{if } r_a > r_d \\ \sum_{N_a^v=\max\{0,lb\}}^{\min\{N_a,N_v\}} \delta(N_a^v, N_a) \times \theta'(N_a^{pv}, N_a^v) \times \varphi(N_v^f, N_v) & \text{otherwise} \end{cases} \quad (12)$$

where $lb = N_a - N_s - N_c + N_v$, and

$$\theta'(N_a^{pv}, N_a^v) =$$

$$\begin{cases} \sum_{N_a^{pv}=\max\{0,N_p+N_a^v-N_v\}}^{\min\{N_p,N_a^v\}} \theta(N_a^{pv}, N_a^v) & \text{if } N_v > N_p \\ 1 & \text{otherwise} \end{cases} \quad (13)$$

We use an example to illustrate the steps in deriving system reliability.

Example 3: Consider Example 2 presented in Section I, where the defender creates $N_c = 3$ camouflaging components, allocates the remaining resources to protect $N_p = 3$ components, and chooses these three protected components as voters, that is $N_v = 3$; whereas the attacker randomly selects $N_a = 4$ components to attack. Therefore, the amount of attack resources r_a on each attacked component is $r_a = R_a/N_a = 18/4 = 4.5$, and the amount of defense resources r_d on each protected component is $r_d = (R_d - N_c \times C)/N_p = (18 - 3 \times 2)/3 = 4$.

Based on the information and (6), the number of the attacked voting components N_a^v ranges from 0 to 3. The probability that the attacker attacks exactly N_a^v voting components, $\delta(N_a^v, 4)$, is

$$\delta(N_a^v, 4) = \frac{\binom{3}{N_a^v}\binom{9}{4-N_a^v}}{\binom{12}{4}} \quad (14)$$

In addition, as $r_a > r_d$, from (8), we know $N_v^f = N_a^v$. The probability that a correct result is obtained when N_v^f components are compromised, $\varphi(N_v^f, 3)$, is

$$\varphi(N_v^f, 3) = \sum_{j=2}^{3-N_v^f} \binom{3 - N_v^f}{j} (0.9)^j (0.1)^{3-N_v^f-j} \quad (15)$$

Therefore, based on (12), the system reliability is

$$P(3,3,3,4) = \sum_{N_a^v=0}^{3} \frac{\binom{3}{N_a^v}\binom{9}{4-N_a^v}}{\binom{12}{4}} \sum_{j=2}^{3-N_a^v} \binom{3-N_a^v}{j} (0.9)^j (0.1)^{3-N_a^v-j}$$
$$= 0.6598$$

□

We have shown how system reliability is derived once R_d, R_a, C, N_c, N_p, N_v, and N_a are given. In the next section, we will discuss how to make the choices with respect to N_c, N_p, and N_v so that the system reliability is maximized under the worst-case scenario, i.e., under the scenario where it is more favorable to an attacker.

V. Algorithms to Determine Defense Strategy against Rational Attacks

For a fixed number of camouflaging components N_c ($0 \leq N_c \leq \lfloor R_d/C \rfloor$), a defender can vary the value of N_p and N_v. As N_p ranges from 0 to N_s, and N_v from 1 to N_s, therefore, the total number of strategies the defender can choose is $S_d = (N_s + 1)N_s$. We arrange the possible defense strategies in a lexicographical order. In particular, for a given N_c, the ith ($1 \leq i \leq S_d$) defense strategy corresponds to when $N_p = \lfloor (i-1)/N_s \rfloor$ and $N_v = i - N_s \times \lceil i/N_s \rceil + N_s$.

On the other hand, for the attacker, the number of attacked components ranges from 1 to $(N_s + N_c)$. Therefore, the total number of possible attack strategies is $S_a = N_s + N_c$, and the jth ($1 \leq j \leq S_a$) attack strategy is $N_a = j$.

When both the defense strategy (N_c, N_p, N_v) and the attack strategy (N_a) are determined, the system reliability can be calculated by using (12). Therefore, we define a matrix $M = (r_{i,j})_{S_d \times S_a}$ to record the system reliability under each possible defense and attack strategy, where $r_{i,j}$ refers to the system reliability when the defender chooses the ith defense strategy and the attacker chooses the jth attack strategy. In other words, $r_{i,j} = P(N_c, \lfloor (i-1)/N_s \rfloor, i - N_s \times \lceil i/N_s \rceil + N_s, j)$.

Clearly, for each defense strategy, depending on which attack strategy is taken by an attacker, the system reliability can vary. We introduce two vectors, i.e., $\overrightarrow{X} = [x_1, \ldots, x_{S_d}]^T$ and $\overrightarrow{Y} = [y_1, \ldots, y_{S_a}]^T$, where $x_i \in \{0, 1\}$ and $y_j \in \{0, 1\}$ denote whether the ith defense strategy is chosen by a defender and the jth attack strategy is chosen by an attacker, respectively. Since a defender and an attacker can choose only one strategy at a time, we have $\sum_{i=1}^{S_d} x_i = 1$, and $\sum_{j=1}^{S_a} y_j = 1$.

Based on the defense strategy selection vector $\overrightarrow{X}$ and the attack strategy selection vector $\overrightarrow{Y}$, the system reliability is given by (16).

$$P = \overrightarrow{X}^T M \overrightarrow{Y} = \sum_{i=1}^{S_d} \sum_{j=1}^{S_a} x_i r_{i,j} y_j \tag{16}$$

The objective of the defender is to maximize the system reliability under the worst-case scenario, i.e., if the defender's strategy is determined, an attacker chooses a strategy that minimizes the system reliability. In other words, the defender's objective is to maximize P_{min}, where

$$P_{min} = \min_{1 \leq j \leq S_a} \sum_{i=1}^{S_d} x_i r_{i,j} \tag{17}$$

Algorithm 1 gives the procedure of finding the system's maximized minimum reliability when the number of camouflaging components N_c is fixed.

A brief explanation of Algorithm 1: from Line 4 to Line 8, we obtain the system's minimum reliability under all of the possible attack strategies when the defender chooses the ith ($1 \leq i \leq S_d$) defense strategy. From Line 9 to Line 12, the defender chooses the strategy under which the system's minimum reliability is maximized. Finally, we output the system's maximized minimum reliability P_{maxmin} and the corresponding defender's strategy vector $\overrightarrow{X}$ (Line 14).

Algorithm 1 Finding the system's maximized minimum reliability when N_c is fixed

Input: A reliability matrix $M = (r_{i,j})_{S_a \times S_d}$.
Output: System's maximized minimum reliability P_{maxmin}, and defense strategy vector $\overrightarrow{X}$.

1: $P_{maxmin} \leftarrow 0$; $\overrightarrow{X} \leftarrow 0$;
2: **for** $i \leftarrow 1$ to S_d **do**
3: $\quad P_{min} \leftarrow 1$
4: $\quad$ **for** $j \leftarrow 1$ to S_a **do**
5: $\quad\quad$ **if** $P_{min} > r_{i,j}$ **then**
6: $\quad\quad\quad P_{min} \leftarrow r_{i,j}$
7: $\quad\quad$ **end if**
8: $\quad$ **end for**
9: $\quad$ **if** $P_{maxmin} < P_{min}$ **then**
10: $\quad\quad P_{maxmin} \leftarrow P_{min}$
11: $\quad\quad$ **Set** x_i to 1, and the rest to 0.
12: $\quad$ **end if**
13: **end for**
14: **return** $P_{maxmin}, \overrightarrow{X}$

Theorem 1: Algorithm 1 obtains the system's maximized minimum reliability under all of the possible attack strategies when the number of camouflaging components is fixed. □

Proof: We prove the theorem by contradiction. If there exists a higher value of the system's maximized minimum reliability under all possible attack strategies, the corresponding defense strategy must be one of the S_d strategies. In Algorithm 1, we compare the system's minimum reliability under each defense strategy and choose the largest one. Therefore, the assumption does not hold. □

We use an example to illustrate how to use Algorithm 1 to determine the system's maximized minimum reliability when the number of camouflaging components is fixed.

Example 4: Assume a system consists of $N_s = 5$ functional components, and the reliability of the components is $p = 0.95$. The defender and the attacker have $R_d = 35$ and $R_a = 20$ units of resources, respectively. The cost for creating a camouflaging component is $C = 3$ units of resources.

Suppose the defender creates two camouflaging components, i.e., $N_c = 2$. Under this case, the total number of defense strategies is $S_d = (N_s + 1)N_s = 6 \times 5 = 30$, and the total number of attack strategies is $S_a = N_s + N_c = 7$. Matrix $M = (r_{i,j})_{30 \times 7}$ stores the system reliability under each possible defense and attack strategy, where $r_{i,j} = P(2, \lfloor (i-1)/5 \rfloor, i - 5 \times \lceil i/5 \rceil + 5, j)$.

Based on the given values, i.e., N_s, R_a, p, R_d, N_c, and C, we follow the procedure shown in Algorithm 1 and obtain $x_{18} = 1$ and $P_{maxmin} = 0.9541$. Therefore, the optimal defense strategy is the 18th strategy, i.e., $i = 18$, that is $N_p = \lfloor (i-1)/N_s \rfloor = \lfloor 17/5 \rfloor = 3$, and $N_v = i - N_s \times \lceil i/N_s \rceil + N_s = 18 - 20 + 5 = 3$, and the system reliability is $P = 0.9541$. □

The discussion we have so far is under the assumption that the number of camouflaging components is fixed. Now, we relax the constraint and let N_c vary from 0 to $\lfloor R_d/C \rfloor$. Algorithm 2 shows the procedure of choosing N_c and the (N_p, N_v) pair that maximize system reliability.

Algorithm 2 DECIDE THE FINAL DEFENSE STRATEGY UNDER GIVEN RESOURCES

Input: Number of system components N_s, defense resources R_d, attack resources R_a, cost for creating a camouflaging component C, and components' reliability p.

Output: System's maximum reliability $P_{maximum}$, defense strategy (N_c, N_p, N_v).

1: $P_{maximum} \leftarrow -1$; $\overrightarrow{X}_{final} \leftarrow 0$
2: $N_c \leftarrow 0$; $N_p \leftarrow 0$; $N_v \leftarrow 0$;
3: **for** $N'_c \leftarrow 0$ to $\lfloor R_d/C \rfloor$ **do**
4: $S_d \leftarrow (N_s+1)N_s$; $S_a = N_s + N'_c$
5: **Obtain** reliability matrix M by using (12)
6: **Apply** Algorithm 1 to get P_{maxmin} and $\overrightarrow{X}$
7: **if** $P_{maximum} < P_{maxmin}$ **then**
8: $P_{maximum} \leftarrow P_{maxmin}$
9: $\overrightarrow{X}_{final} \leftarrow \overrightarrow{X}$; $N_c \leftarrow N'_c$
10: **end if**
11: **end for**
12: $i \leftarrow Index\_Value\_Is\_One(\overrightarrow{X}_{final})$
13: $N_p \leftarrow \lfloor (i-1)/N_s \rfloor$
14: $N_v \leftarrow i - N_s \times \lceil i/N_s \rceil + N_s$
15: **return** $P_{maximum}, N_c, N_p, N_v$

A brief explanation of Algorithm 2: under each resource allocation of camouflaging component creation (Line 3), we first obtain the system's maximized minimum reliability under different resource allocations (from Line 4 to Line 6), and then choose the largest among those as the final system reliability (from Line 7 to Line 10). In Line 12, we obtain the index of the element in $\overrightarrow{X}_{final}$ whose value is equal to 1, and then based on the index, we get the defender's strategy N_p and N_v (Line 13 and Line 14). Finally, we return the system's maximum reliability $P_{maximum}$ and the corresponding defense strategy N_c, N_p, and N_v (Line 15).

Theorem 2: Algorithm 2 obtains the system's *maximum* reliability under all possible attack strategies. □

Proof: We prove the theorem by contradiction. If we can find a higher system reliability, it must exist in one of the resource allocations of the camouflaging component creation. Algorithm 2 compares the maximal reliability under each resource allocation and chooses the largest among them. Therefore, we cannot obtain a higher system reliability than the one produced by Algorithm 2. □

Example 5 (Example 4 Revisited): In Example 4, the maximum number of camouflaging components which can be created is $\lfloor R_d/C \rfloor + 1 = \lfloor 40/3 \rfloor + 1 = 14$. Under each resource allocation of camouflaging component creation, the system's maximal reliability and the corresponding number of protected and voting components are shown in Table I. After comparing the system's maximal reliability under different resource allocations, the maximum system reliability is $P = 0.9860$, and the corresponding defense strategy is $N_c = 0, N_p = 4$, and N_v = 5. □

VI. SIMULATION RESULTS

We have implemented a simulator which follows the steps in Algorithm 2 to investigate how the defense and attack resources impact the defender's strategy. In addition, we compare the system reliabilities when the number of voting components is fixed versus when it is optimally determined.

For the first set of experiments, we study how the defense resources impact the defender's strategy. In this experiment setting, we assume that a system consists of seven replicas, and the reliability of each replicated component is 0.95. The amount of attack resources is 40 units, and the cost for creating a camouflaging component is 3 units of resources. In other words, we have $R_a = 40$, $C = 3$, $p = 0.95$, and $N_s = 7$. The amount of defense resources R_d is increased from 10 to 110 units. Under each increase of defense resources, the defense strategy is obtained by using the simulator which implements Algorithm 2.

It is worth mentioning that there is no particular preference when deciding the amount of attack resources. We set the amount of attack resources to 40 units and the range of defense resources from 10 to 110 units, so that we are able to see how the defense strategy changes under different cases, i.e., the amount of defense resources is less than, equal to, and greater than the amount of attack resources. In the second set of experiments, we fix the defense resources and vary the amount of attack resources from 10 to 110 units to further investigate how the amount of attack resources impacts the defense strategy.

Fig. 3 shows how the defense strategy (N_c, N_p, and N_v) and unreliability of the system (i.e., $1 - P$) change as the amount of available defense resources increases. The primary (left) y-axis shows the optimal number of protected, camouflaging, and voting components under a specific amount of defense resources. The secondary (right) y-axis shows the corresponding minimized maximum unreliability of the system. From Fig. 3, we observe the following:

1) When the amount of defense resources is small, the defender should allocate resources to protect a small set of functional components (Line L2 in Fig. 3) and choose these protected components to vote (Line L3 in Fig. 3).
2) As the amount of defense resources increases, the defender should protect more components (Line L2 in Fig. 3) and choose these as voting components as well (Line L3 in Fig. 3).
3) All protected components should participate in the voting (Line L2 and L3 in Fig. 3).
4) The unreliability of the system decreases, i.e., the system's reliability increases, as the amount of defense resources increases. (Line L4 in Fig. 3).
5) When the amount of defense resources increases, the defender should create more camouflaging components (Line L1 in Fig. 3).

The reason for the first two observations lies in the fact that when the amount of defense resources is small, allocating all of the resources to protect a single voting component maximizes the probability that the voting component survives the attack. However, if the amount of defense resources is large enough, i.e., the amount of protection resources on the voting components can guarantee that the majority of the voting components are not compromised by the attacker, it is better to protect more components and also choose these

TABLE I
THE MAXIMAL SYSTEM RELIABILITY UNDER EACH RESOURCE ALLOCATION

$R_d = 40, R_a = 20, N_s = 5, C = 3$														
N_c	0	1	2	3	4	5	6	7	8	9	10	11	12	13
P	0.9860	0.9500	0.9541	0.9589	0.9524	0.9500	0.9500	0.9025	0.9025	0.8821	0.8867	0.8313	0.7265	0.0000
N_p	4	1	3	3	4	1	1	2	2	1	1	1	1	1
N_v	5	1	3	3	5	1	1	3	3	1	1	1	1	1

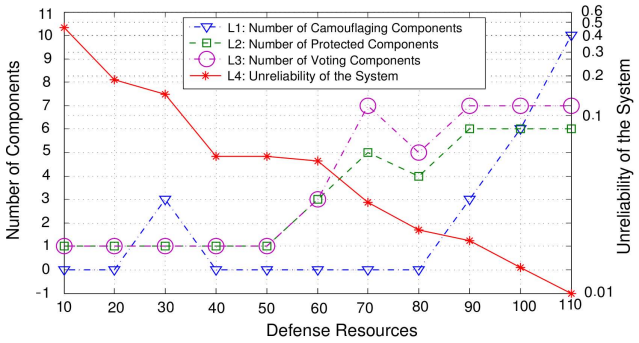

Fig. 3. The relationship between defense resources, defender's strategy, and unreliability of the system when the amount of attack resources is 40 units

protected components as voters for lower unreliability of the system.

As the system reliability not only depends on the number of voting components but also on the probability of components being compromised. Therefore, protecting non-voting components does not contribute to the improvement of system reliability. On the contrary, it reduces the amount of resources that could be applied to the protection of voting components and henceforth reduces the voting components' probability to survive an attack. Therefore, all protected components should participate in the voting (observation three).

The explanation for the fourth observation is that when the amount of defense resources increases, the probability that a voting component is compromised decreases because more camouflaging components can be created or more protection resources are added to the voting components.

Although adding camouflaging components can lower the probability that voting components are attacked, the benefit to unreliability of the system decrease is not obvious because of the existence of non-voting components in the system. Therefore, when the amount of defense resources is small, it is more appropriate to allocate the resources to component protection. If the amount of defense resources is large enough, we can create camouflaging components to further lower the probability that voting components are attacked, therefore, decreasing unreliability of the system (observation five). That is why when the amount of defense resources is below 80 units (except the case where $R_d = 30$), no camouflaging component is created.

It is worth pointing out that when the amount of defense resources is 30 units, the defender allocates resources to create three camouflaging components, which is against our judgment. However, the reason to create camouflaging components is not because the benefit of creating camouflaging components outweighs protecting voting components, but because under the cases in which $R_d = 30$ and $R_a = 40$, we only need 20 units resources to protect the single voting component, therefore, the remaining defense resources are used to create camouflaging components.

To be more specific, if the amount of attack resources is 40 units and the amount of protection resources on the single voting component is 20 units, the attacker must use all available resources to attack a single component, because if he/she attacks more than one component, according to Axiom 1, none of the attacked voting components will be compromised. As the attacker allocates 40 units of resources to attack only one component, allocating 20 units or 30 units to protect the voting component does not make a difference, i.e., once the voting component is attacked, it will be destroyed. Therefore, the defender should allocate the remaining 30 - 20 = 10 units of resources to create $\lfloor \frac{10}{3} \rfloor = 3$ camouflaging components to lower the attack probability of the voting component.

In this experiment, the system reliability is evaluated under the worst-case scenario. In other words, no matter what strategy the attacker takes, the system reliability is no less than the system's maximized minimum reliability. Therefore, when the attacker randomly selects his/her strategy, i.e., the attack strategies have uniform probability, the system's expected reliability will be greater than or equal to the maximized minimum reliability, and their difference is shown in Fig. 4 where the values of N_v, N_p, and N_c are the same as in Fig. 3.

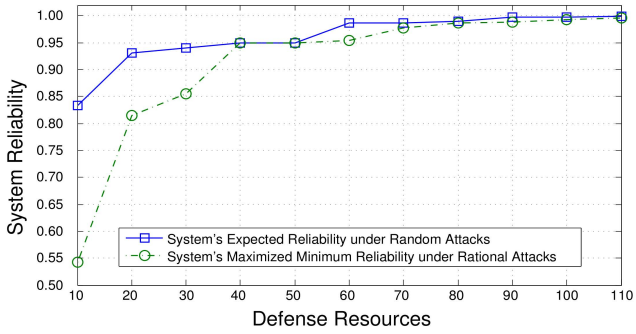

Fig. 4. Difference between system's maximized minimum reliability and expected reliability

From Fig. 4, we can see that the more defense resources the defender has, the smaller the difference between the system's maximized minimum reliability and expected reliability. This is because when the amount of defense resources is small, the probability that the majority of the voting components survive the random attacks is much larger than the probability under the worst-case scenario. When the amount of defense resources

increases, more resources are available to protect the voting components, and the system's maximized minimum reliability increases accordingly. Therefore, the difference between the system's maximized minimum reliability and expected reliability decreases.

For the second set of experiments, we analyze the effect of the amount of attack resources on defender's strategies and system reliability. The experiment settings for C, p, and N_s are the same as in the previous experiment, and the amount of defense resources is 40 units. In other words, we have $R_d = 40$, $C = 3$, $p = 0.95$, and $N_s = 7$. The amount of attack resources increases from 10 to 110 units.

For each R_a, we also use the simulator which implements Algorithm 2 to decide the defender's strategy. Similar to Fig. 3, the primary y-axis of Fig. 5 shows the optimal number of camouflaging, protected, and voting components under a specific amount of attack resources, and the secondary y-axis is the corresponding system reliability. From Fig. 5, it is easily seen that when the amount of attack resources is small, i.e., $R_a = 10$, all of the system components are protected and chosen to vote (Line L2 and L3 in Fig. 5). However, when the amount of attack resources increases, i.e., $R_a = 30$, there is only one component being protected in the hopes that the voting component survives the attack (Line L2 and L3 in Fig. 5).

The main reason for the strategy change is that when the amount of attack resources is small, choosing more voting components increases the system reliability, and at the same time, we can guarantee that the majority of the voting components are not compromised by the attacker. However, when the amount of attack resources increases, the majority of the voting components will be compromised by the attacker if we evenly distribute the same defense resources to a large set of voting components. Therefore, a better defense strategy would be to invest the resources into a smaller group, improving their probability of surviving the attack.

In addition, from Fig. 5, we can see that when the amount of attack resources increases, the system reliability decreases (Line L4 in Fig. 5). The reason for the system reliability change is that when more attack resources are used, the probability that the voting components becoming compromised increases accordingly.

Another observation from Fig. 5 is that when the amount of attack resources increases from 50 units to 110 units, the number of camouflaging components changes drastically (Line L1 in Fig. 5). The reason for the change is similar to the one illustrated in the first set of experiments, i.e., the defender creates camouflaging components not because the effect of creating camouflaging components is better than component protection, but because remaining resources exist after component protection.

For example, when the amount of attack resources is 50 units, the protection resources on the only voting component should be greater than or equal to 25 units, because in this case, the attacker can only attack one component. If the attacker attacks more than one component, the attack resources on the attacked components is less than 25 units, and the voting component will not be compromised. As the defender has 40 units of defense resources in total, and the minimum amount of desired protection resources is 25 units, the defender will use the remaining (40 - 25) = 15 units of resources to create $\frac{15}{3} = 5$ camouflaging components to lower the attack probability of the voting component. Similar analysis applies in other cases where the number of camouflaging components is nonzero.

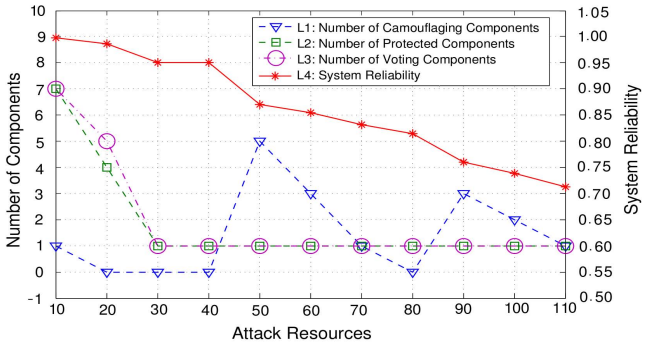

Fig. 5. The relationship between attack resources, the defender's strategy, and system reliability when the amount of defense resources is 40 units

When the system is under random attacks, as shown in Fig. 6 (where the values of N_v, N_p, and N_c are the same as in Fig. 5), the difference between the system's maximized minimum reliability and expected reliability grows when the amount of attack resources increases, this is because when the amount of attack resources increases, the probability that the voting components are compromised under worst-case scenario increases faster than the probability under random attacks.

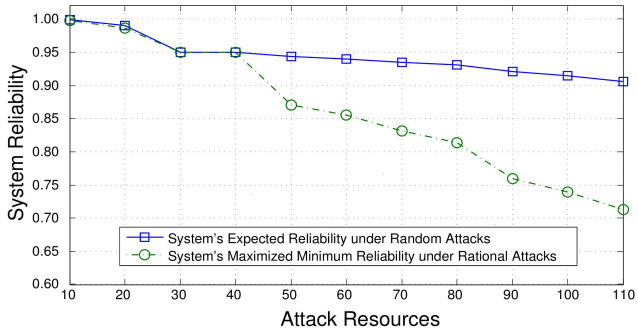

Fig. 6. Difference between the system's maximized minimum reliability and expected reliability

For the third set of experiments, we compare the system reliability when the number of voting components is fixed versus when it is optimally decided by the algorithm. More precisely, we set $R_a = 40$, $C = 3$, $p = 0.95$, $N_s = 7$, and let defense resources vary from 10 to 70 units. The value of N_v is set to constant 1, 3, 5, and 7, or chosen by the Algorithm 2, respectively.

Fig. 7 shows the system reliability under different numbers of voting components. From Fig. 7 we can see that the system reliability with a fixed number of voting components is never greater than the case in which the number of voting components is optimally decided.

To be more specific, when the amount of defense resources is below 55 units, the optimal number of voting components is 1, when the amount of defense resources increases to 60, 65, and 70 units, the optimal number of voting components is 3, 5, and 7, respectively. This result validates our conjecture that when the system is under attack, the number of components that participate in a voting processing may not be positively proportional to the reliability of the system.

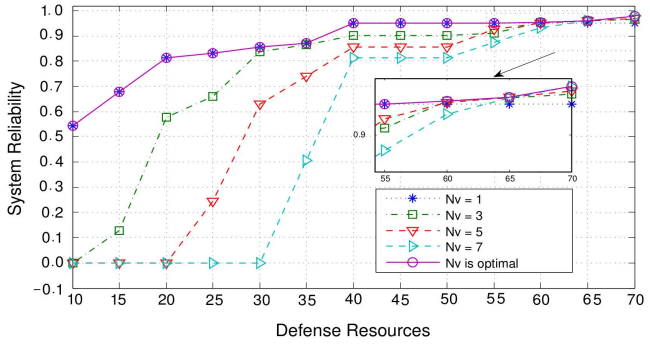

Fig. 7. The system reliability under different numbers of voting components when the amount of attack resources is 40 units

VII. Conclusion

Redundancy and voting schemes are often used to tolerate natural-caused component failures. In general, the more reliable components that participate in a voting process, the higher reliability the system can achieve. However, when a system is under intentional cyber attacks, the system reliability is not necessarily proportional to the number of redundancies of voting components.

This paper has analyzed system reliability when both a defender and an attacker are given a fixed amount of resources and studied how their resource allocation strategies impact system reliability. Based on the analysis, we have developed an algorithm for system defenders to optimally allocate their resources and decide the number of voting components so that the system reliability is maximized even under the scenario that is most favorable to attackers. The experimental results show that when a defender has sparse resources compared to what the attacker has, the defender should invest the resources into protecting a fewer number of components and select only the protected components for voting; in contrast, if the resources are abundant, the defender should increase the number of protected components and allow more components to vote.

In our work, we only consider that components can be compromised while the communication channel for the voting protocol is reliable. However, in reality, communication channels play an important role when it comes to the system reliability, often times they are the target of attacks [31]. When network reliability is taken into consideration, less communication in reaching a consensus could imply higher reliability of the consensus; on the other hand, fewer voting participants (less communication) could result in lower reliability. It becomes more complicated when intentional attacks exist. Hence, our next step is to include the communication channel into the system model and investigate how the communication channel affects system reliability and defense strategy.

Acknowledgment

The work was supported in part by NSF CAREER Award (CNS0746643) and Air Force Office of Scientific Research (AFOSR). Approved for Public Release; Distribution Unlimited: 88ABW-2012-4239, Aug 3rd, 2012.

References

[1] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *IEEE Trans. Dependable Secur. Comput.*, vol. 1, no. 1, pp. 11–33, Jan. 2004.

[2] Y. Zuo, "A framework of survivability requirement specification for critical information systems," in *System Sciences (HICSS), 2010 43rd Hawaii International Conference on*, Jan. 2010, pp. 1 –10.

[3] K. Kwiat, A. Taylor, W. Zwicker, D. Hill, S. Wetzonis, and S. Ren, "Analysis of binary voting algorithms for use in fault-tolerant and secure computing," in *Computer Engineering and Systems (ICCES), 2010 International Conference on*, 2010, pp. 269 –273.

[4] K. S. Trivedi, *Probability and statistics with reliability, queuing and computer science applications*, 2nd ed. Chichester, UK: John Wiley and Sons Ltd., 2002.

[5] D. P. Siewiorek and R. S. Swarz, *Reliable computer systems (3rd ed.): design and evaluation*. Natick, MA, USA: A. K. Peters, Ltd., 1998.

[6] L. Wang, Y. Leiferman, S. Ren, K. Kwiat, and X. Li, "Improving complex distributed software system availability through information hiding," in *Proceedings of the 2010 ACM Symposium on Applied Computing*, 2010, pp. 452–456.

[7] I. Mokube and M. Adams, "Honeypots: concepts, approaches, and challenges," in *ACM-SE 45: Proceedings of the 45th annual southeast regional conference*. New York, NY, USA: ACM, 2007, pp. 321–326.

[8] G. Levitin and K. Hausken, "False targets efficiency in defense strategy," *European Journal of Operational Research*, vol. 194, no. 1, pp. 155–162, April 2009.

[9] ——, "Redundancy vs. protection vs. false targets for systems under attack," *Reliability, IEEE Transactions on*, vol. 58, no. 1, pp. 58 –68, March 2009.

[10] ——, "Protection vs. redundancy in homogeneous parallel systems," *Reliability Engineering & System Safety*, vol. 93, no. 10, pp. 1444 – 1451, 2008.

[11] K. Hausken, "Production and conflict models versus rent-seeking models," *Public Choice*, vol. 123, no. 1, pp. 59–93, April 2005.

[12] V. A. Vicki M. Bier, "Optimal allocation of resources for defense of simple series and parallel systems from determined adversaries," in *Proceedings of the Engineering Foundation Conference on Risk-Based Decision making in Water Resources X*. American Society of Civil Engineers, 2003, pp. 59–76.

[13] V. M. Bier, A. Nagaraj, and V. Abhichandani, "Protection of simple series and parallel systems with components of different values," *Reliability Engineering and System Safety*, vol. 87, no. 3, pp. 315 – 323, 2005.

[14] A. Yalaoui, E. Chatelet, and C. Chu, "A new dynamic programming method for reliability redundancy allocation in a parallel-series system," *Reliability, IEEE Transactions on*, vol. 54, no. 2, pp. 254 – 261, June 2005.

[15] K. Hausken, "Strategic defense and attack for series and parallel reliability systems," *European Journal of Operational Research*, vol. 186, no. 2, pp. 856–881, April 2008.

[16] G. Levitin and K. Hausken, "False targets vs. redundancy in homogeneous parallel systems," *Reliability Engineering & System Safety*, vol. 94, no. 2, pp. 588 – 595, 2009.

[17] ——, "Parallel systems under two sequential attacks," *Reliability Engineering & System Safety*, vol. 94, no. 3, pp. 763–772, March 2009.

[18] ——, "Meeting a demand vs. enhancing protections in homogeneous parallel systems," *Reliability Engineering & System Safety*, vol. 94, no. 11, pp. 1711 – 1717, 2009.

[19] B. Hardekopf, K. Kwiat, and S. Upadhyaya, "A decentralized voting algorithm for increasing dependability in distributed systems," in *5th World Multiconference on Systemic, Cybernetics and Informatics*, 2001. [Online]. Available: http://goo.gl/ZJsJD

[20] Z. Tong and R. Kain, "Vote assignments in weighted voting mechanisms," *Computers, IEEE Transactions on*, vol. 40, no. 5, pp. 664 –667, May 1991.

[21] D. Davcev, "A dynamic voting scheme in distributed systems," *Software Engineering, IEEE Transactions on*, vol. 15, no. 1, pp. 93 –97, Jan. 1989.

[22] J. J. Bloch, D. S. Daniels, and A. Z. Spector, "A weighted voting algorithm for replicated directories," *J. ACM*, vol. 34, pp. 859–909, Oct. 1987.

[23] D. K. Gifford, "Weighted voting for replicated data," in *Proceedings of the seventh ACM symposium on Operating systems principles*, 1979, pp. 150–162.

[24] L. Nordmann and H. Pham, "Weighted voting systems," *Reliability, IEEE Transactions on*, vol. 48, no. 1, pp. 42 –49, March 1999.

[25] R. J. Vanderbei, *Linear Programming: Foundations and Extensions*, 2nd ed. Springer, 2001.

[26] T. Raghavan, "Chapter 20 zero-sum two-person games," ser. Handbook of Game Theory with Economic Applications, R. Aumann and S. Hart, Eds. Elsevier, 1994, vol. 2, pp. 735 – 768.

[27] L. Wang, S. Ren, K. Yue, and K. Kwiat, "Optimal resource allocation for protecting system availability against random cyber attacks," in *Computer Research and Development (ICCRD), 2011 3rd International Conference on*, vol. 1, march 2011, pp. 477 –482.

[28] ——, "Optimal resource allocation to improve distributed system reliability," in *Workshop on Secure Knowledge Management*, 2010. [Online]. Available: http://goo.gl/NIykx

[29] L. Wang, Z. Li, S. Ren, and K. Kwiat, "Optimal voting strategy against rational attackers," in *Risk and Security of Internet and Systems (CRiSIS), 2011 6th International Conference on*, sept. 2011, pp. 1 –8.

[30] A. Dominguez-Garcia and P. Grainger, "A framework for multi-level reliability evaluation of electrical energy systems," in *IEEE Energy 2030 Conference*, Nov. 2008, pp. 1 –6.

[31] M. J. Freedman, "Design and analysis of an anonymous communication channel for the free haven project," *http://groups.csail.mit.edu/cis/theses/freedman-bachelors.ps*, May 2000.