**Problem Statement[[1]](#footnote-1)**

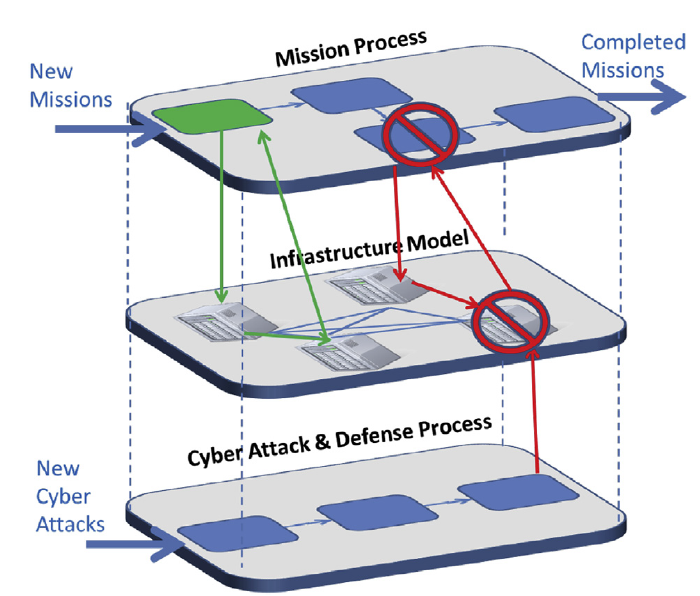
**A. Inference Model**

The use of cyberspace as a platform for military operations has been growing at impressive rates. Yet, it is still a relatively new area with considerable research challenges. Security techniques are not sufficiently effective in protecting IT systems, and most fail to address the correlation between actions and effects across multiple domains. In other words, identifying how actions performed in the cyber domain affect the mission goals is yet an unsolved problem.

It must include the previous approach: an adequate assessment of the correlation between cyber and physical behaviors performed holistically and allowing tasks to be evaluated in real-time. Existing tools and methodologies cannot provide this information level and are unsuitable to support complex cyber threat assessment in real situations. Despite the relatively large body of research on the subject, this significant gap still exists.

The most common approach found in the research literature is to determine how vulnerabilities can be exploited by the enemy (e.g. (Jajodia et al., 2005; Jajodia et al., 2011; Jakobson, 2011a; Buckshaw et al., 2005)) - the threat-centric (Silva and Jacob, 2018). It usually involves the generation of an attack graph (Sheyner et al., 2002), which includes vulnerabilities and exploit strategies. The impact assessment is then calculated using evidence the analyst extracts from the environment, which in most real scenarios is impracticable due to the computational cost of solving the graphs. Another limitation is the requirement of knowing the attack path in advance, which is rarely possible since attacks against critical infrastructure always attempt to explore new attack paths and unknown vulnerabilities (i.e., zero-day attacks).

A more recent approach that avoids this problem involves measuring the impact of cyber-attacks on a mission, mission-centric approach (Silva and Jacob, 2018). The idea is to define how the mission can be impacted through the analysis of the effects produced by the interactions of offensive (enemy) and defensive plans (Musman et al., 2011a) under the premise that it is easier to design one's mission and its restrictions than to know the enemy behavior. This approach is focused on effects and does not require detecting attacks or attackers but only understanding the possible effects of their actions on the mission. The mission is modeled to measure the impact, and all critical components are identified and monitored. Musman studied using this perspective (Musman et al., 2010, 2011a, 2011b; Musman and Temin, 2015). However, that body of research focuses on assessing how to leverage impact estimations to support the designing of security architectures and not on exploring and developing specific and scalable approaches based on the concept.

A new approach is provided by the Cyber-ARGUS Framework, which provides a scalable way of modeling the mission from a holistic perspective, both in its planning and execution phases, establishing the connection of entities within the mission and the cyber domains, and enabling the impact assessment of cyber events to be measured within the mission context (i.e., within the operations domain). An essential contribution is that the framework allows cyber impact assessment of an ongoing mission to be achieved without the need to know the individual actions of the attacker.

The framework has a mission-centric approach, which requires the cyber and mission concepts to be defined and the connection between the two domains to be explicitly expressed. Figure 1 shows a pictorial representation of how Cyber-Argus understands the mission and its concepts. There, you can see different perspectives represented in layers. The first layer represents the mission and the required tasks to perform it successfully. The second layer represents the required services for each layer, where OT and IT systems could provide the services. Finally, the last layer is the cyber domain layer.

Figure 1 - The overall model includes a mission process model, a cyber adversary process model, a cyber defender process model from (Kott et al., 1710).

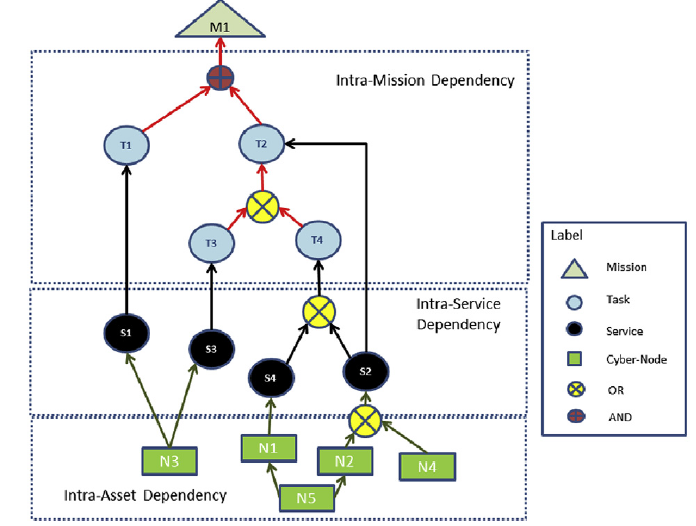
However, this representation is relatively abstract. To be usable with a computer system, the concept of the impact graph must be adapted, as presented by Jakobson (2011b). The adapted graph (impact graph) includes all relationships between tasks, services, and nodes, resulting in a structure that makes it easier to assess the consequences generated when a node is compromised (see Figure 2).

The impact graph is a type of dependency network[[2]](#footnote-2). A dependency network approach provides a system-level analysis of the activity and topology of directed networks. The approach extracts causal topological relations between the network's nodes (when the network structure is analyzed) and provides an essential step towards inference of causal activity relations between the network nodes (when analyzing the network activity).

Figure 2 presents an adaptation of Jakobson's representation of the impact graph. At the top is the Intra-Mission Layer, which is compounded by a mission goal and a set of tasks required to accomplish it. In the example, you have tasks T1 and T2 that both are required to accomplish (AND gateway) successfully to accomplish the mission M1 has accomplished in a thriving state. Also, you can see in the Intra-Mission Dependency layer that tasks T3 and T4, which are required (at least one with successful performance – OR gateway) to task T2, can be performed successfully.

You have the services required for each task at the Intra-Service Dependency layer. There, you can see individual services required to accomplish the tasks, complex one compound by two or more services (using OR/AND gateways); it is not represented in the Figure, but you can have a service that depends on other services.

Finally, you have the last layer, the Intra-Asset Dependency or the Cyber Layer, which is compounded by the cyber-nodes and has the same topology design properties as the concepts in the other layers.



*Figure 2 - Impact Graph.*

Note that the Impact Graph represents the mission, and this step aims to assess the influence between the nodes in the graph. In other words, assess the interdependence between the various components of this mission representation. This assessment is performed using a Reasoning Model.

The Reasoning Model can be implemented via four main approaches. The first involves using algebraic expressions to model the relations, a typical example being a linear regression model. There are two main issues with this approach. One is that such models require generating statistical historical data, which in most cases does not exist. Even in the few cases where it exists, its success in predicting cyber-attacks is fragile as these tend not to follow predefined patterns. The second issue is related to the need to interpret evidence by an analyst, which makes it more complicated when an arid mathematical model is used (Daniels et al., 2008).

A second approach for building a Reasoning Model is the use of a black box approach, such as Neural Networks (e.g., Deep Learning). However, in addition to the pitfalls of the previous approach, there is the problem of the intrinsic nature of the mapping between the input and output of such models. That is, there is no clear way of building a narrative that explains the model’s results and, thus, no way of making the interrelationships in the graph explicit or explainable.

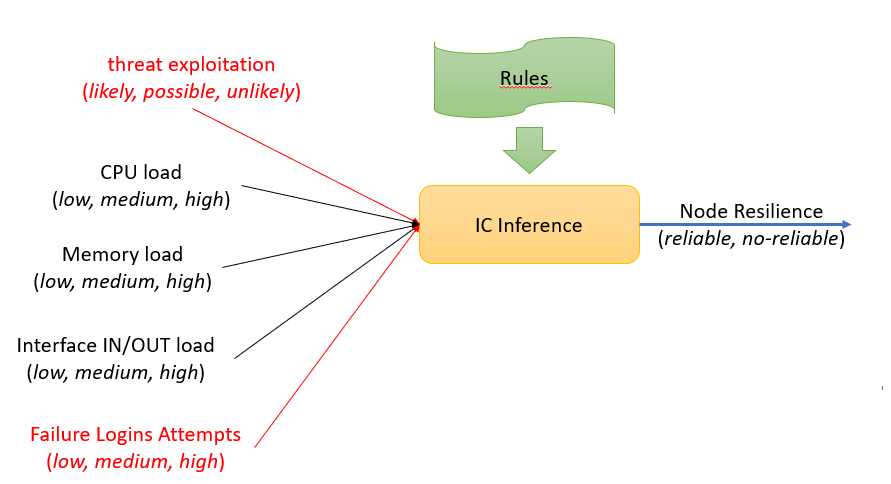
Markov models for security risk are a third alternative, such as the one developed by Kim et al. (2007). However, using Markov processes to propagate Impact Assessment highlights the technique's weakness of its inability to represent non-monotonic dependencies. For instance, in this technique, two independent variables must be directly connected by an edge merely because some other variable depends on both (Pearl, 1986).

The last alternative, used by Cyber-Argus, is Bayesian Networks (BNs). This probabilistic graphical technique represents a set of random variables and their joint probability distribution via a Directed Acyclic Graph (DAG) (Russell and Norvig, 2009). BNs are cognitively meaningful and directly interpretable. Unlike traditional rule-based systems, BNs employ a coherent calculus to manage the uncertainty and absorb evidence as it accrues (cf. (Daniels et al., 2008)). In contrast to other classical statistic approaches, BNs provide backward inference (i.e., allowing for what-if analyses). Fenton lists some of the advantages of BNs, such as a) explicitly model causal factors; b) provide reasoning from effect to cause and vice-versa; c) reduce the burden of parameter acquisition; d) overturn previous belief in light of new evidence; e) make predictions with incomplete data; f) combine diverse types of evidence including both subjective beliefs and objective data and arrive at decisions based on visible, auditable reasoning (Fenton and Neil, 2013).

Given that Cyber-Argus's main idea is to interpret how events in the cybernetic layer influence the performance of elements at the mission level and vice versa, it is required to represent the cyber node's health. In other words, it is the ability to provide the required resources and services within a certain level of quantity, quality, effectiveness, and cost. At Cyber-Argus, the node health is measured by the Infrastructure Capacity (IC).

Unlike the reference paper, in this Exam, we will simplify the reasoning model, making it more straightforward than the one used in the paper. In the approach used in this activity, the IC is calculated by a fuzzy inference box. Using fuzzy sets in cybersecurity has several advantages, particularly in dealing with uncertainty, imprecision, and complexity in risk assessment and decision-making processes. In the particular case of cyber nodes, making some inferences useful for the other layers is challenging. Fuzzy sets are excellent for modeling uncertain and imprecise data, often expressed in natural language terms like "high risk," "moderate impact," or "low likelihood." Also, Fuzzy logic can handle incomplete and noisy data more gracefully than other methods. It does not rely heavily on complete datasets, which are often difficult to obtain in cybersecurity.

We use the Fuzzy Inference engine presented in Figure 3 to calculate the IC. There, you can see that you calculate IC based on five inputs collected from cyber sensors and, based on a set of rules, infer the node resilience, which will be used to infer the reliability of the following layers.



*Figure 3 - IC Calculation.*

Each input variable has its membership representation in Table 1, where it is presented the functions that transform the crips value from the sensor measurements to a fuzzy value.

|  |  |  |
| --- | --- | --- |
| **Variable** | **Membership Function** | |
| Threat Exploitation  (likelihood) | Likely | Triangular (a=10, b=8, c=6) |
| Possible | Gaussian (c = 5, s=2, m=2) |
| Unlikely | Triangular (a=5, b=7, c=0) |
| Failure Logins Attempt  (#number) | Low | Triangular (a=0, b=30, c=50) |
| Medium | Trapezoidal (a=20, b=30, c=50, d=300) |
| High | Triangular (a=200, b=400, c=1000) |
| CPU Load  (%) | Low | Triangular (a=0, b=20, c=30) |
| Medium | Trapezoidal (a=20, b=30, c=40, d=60) |
| High | Triangular (a=50, b=70, c=100) |
| Memory Load  (%) | Low | Triangular (a=0, b=20, c=30) |
| Medium | Trapezoidal (a=20, b=30, c=40, d=60) |
| High | Triangular (a=50, b=70, c=100) |

|  |  |  |
| --- | --- | --- |
| Interface IN/OUT  (%) | Low | Triangular (a=0, b=20, c=30) |
| Medium | Trapezoidal (a=20, b=30, c=40, d=60) |
| High | Triangular (a=50, b=70, c=100) |

Finally, you have the target variable described in Table 2.

|  |  |  |
| --- | --- | --- |
| Node Resilience  (Boolean) | Reliable | Gaussian (c = 5, s=2, m=2) |
| No-reliable | Gaussian (c = 7, s=2, m=2) |

The model is based on these rules to support the inference of node resilience. The first one is that node resilience is no-reliable when all of its inputs are high, and the threat of exploitation is likely. The second rule is that when the threat exploitation is possible, or the other variables are high, the node resilience is no-reliable. The node resilience is reliable when the threat exploitation is unlikely and the Failure Logins Attempt is low. When one of the CPU, memory, and IO interfaces load is low, the node resilience is reliable.

Finally, the inference model uses a center-of-gravity approach, and the aggregation is a MAX function.

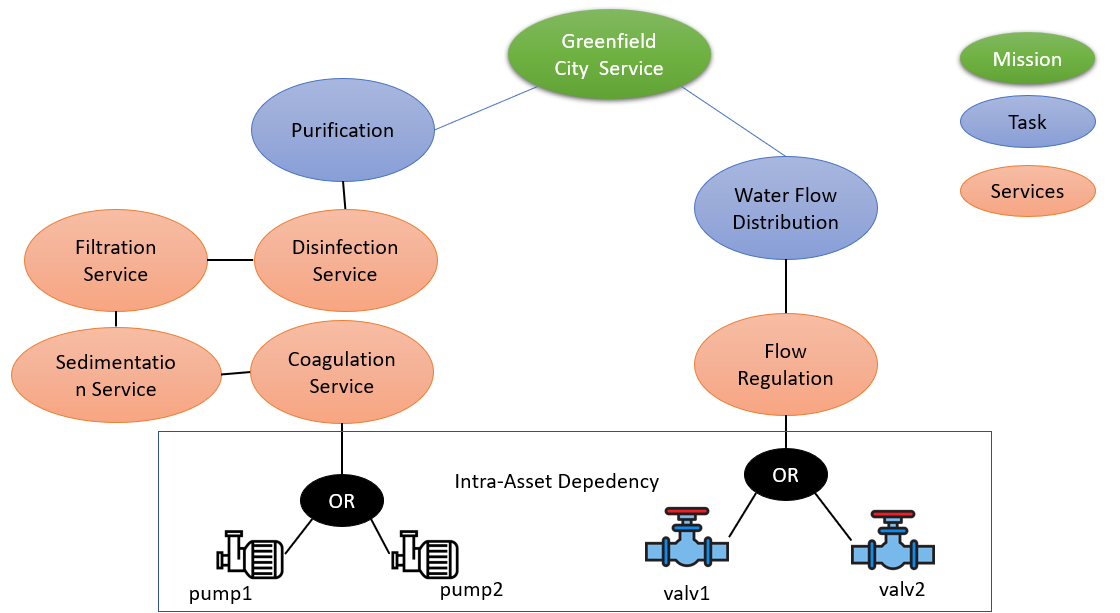
After you calculate the IC and check whether it is reliable, you must propagate this information to the other layers, allowing you to infer whether the mission is reliable (achievable) or not. To perform this task, we use a Bayesian Network that must implement the dependencies described in Jacobson’s impact graph (see Figure 2).

To restrict complexity, all BN nodes are binary, and they can have two possible states: reliable and non-reliable. We will use only simple connections (one parent) or Noisy-OR connections (several parents) to implement the required dependencies. Using Noisy-OR, we assume that each parent influences the target node as a conditional independence assumption.

**B. Study Case**

Greenfield City operates a large municipal water treatment plant (WTP) that provides clean drinking water to approximately 500,000 residents. The WTP uses an automated control system, Supervisory Control and Data Acquisition (SCADA), to monitor and control water purification and distribution processes. This system includes sensors, programmable logic controllers (PLCs), and human-machine interfaces (HMIs).

Figure 4 contains a simplistic version of this service's topology. The cyber layer is provided for two different devices: smart pumps and valves. The intelligent pumps push water from the river and insert it into the water purification system. The valves regulate the flow of purification water to the city. Both components are IoT devices that process capacity and you use this end device to measure the cyber environment. The next layer, the intra-service layer, contains the required services to support the two main tasks of the water system: water purification and water flow distribution.



*Figure 4 - Greenfield City Service.*

You know that the Coagulation service is reliable 80% of the time when pump1 or pump2 is reliable. The Sedimentation Service is reliable at 70%, while the Coagulation Service is reliable. The filtration service is reliable 90% of the time, while the sedimentation service is reliable. The disinfection service is 70% reliable when the filtration service is reliable. Finally, the purification system is reliable 85% of the time when the disinfection service is reliable. When one of these services or the pumps is unreliable, the system moves to an unreliable state.

In the other process, you know that flow regulation is reliable 98% of the time, where one of the valves is reliable. The water flow distribution is reliable 75% of the time when the flow regulation is reliable. They have the same behavior as the previous process: unreliable when one component is unreliable.

The Greenfield City Service is reliable 60% of the time when purification and water flow distribution are reliable, 45% when only one of the processes is reliable, and not reliable when both are not reliable.

1. Based on Alexandre B. Barreto, Paulo C.G. Costa, Cyber-ARGUS - A mission assurance framework, Journal of Network and Computer Applications, Volume 133, 2019, Pages 86-108, ISSN 1084-8045, https://doi.org/10.1016/j.jnca.2019.02.001. [↑](#footnote-ref-1)
2. Kenett, Dror Y., et al. "Dominating clasp of the financial sector revealed by partial correlation analysis of the stock market." *PloS one* 5.12 (2010): e15032. [↑](#footnote-ref-2)