2. Vulnerabilities in Critical Information Infrastructure

A vulnerability is a flaw in a system or protection mechanism that exposes a system to cyberattacks [5]. Attackers can use cyberthreats to exploit vulnerabilities in order to steal confidential information, damage information or take down websites, thus making information unavailable to authorized users.

Fig. 1 shows the Common Criteria Model which depicts security concepts and relationships [7]. These security concepts and relationships will be examined before discussing the various types of vulnerabilities which are possessed by critical information infrastructure. By applying this model to this paper, owners refer to organisations who value their assets. These assets represent information which is stored on systems and delivered via networks such as the internet.

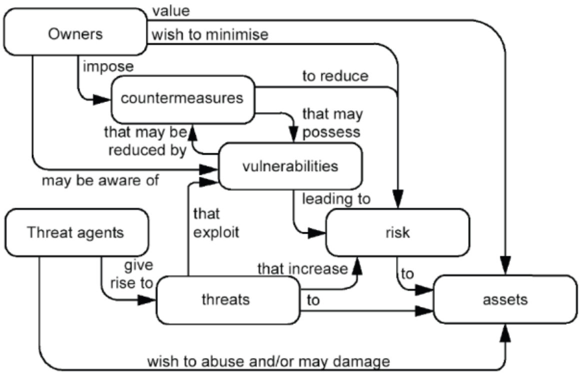

Figure 1. Common Criteria Model [7]

On the other hand, threat agents may wish to abuse or damage these assets by stealing confidential information, sabotaging or modifying information and preventing access to information. Thus, the CIA principles will not be preserved. Examples of threat agents include hackers, disgruntled employees and other entities. These threat agents give rise to threats which target assets. Examples of threats include malware, cybersabotage and Distributed Denial of Service attacks (discussed in section 3). Threat agents are often successful in damaging or abusing assets as they exploit vulnerabilities which are present in critical information infrastructure.

However, owners may be aware of vulnerabilities in critical information infrastructure and thus impose countermeasures (i.e. security controls: discussed in section 4) to reduce them. As a result, risks to assets will be reduced. In contrast, countermeasures such as antivirus software may possess vulnerabilities, which will prevent them from identifying and rectifying the latest software vulnerabilities. Software vulnerabilities will be discussed next.

2.1. Software Vulnerabilities

A software vulnerability is a flaw in the design of a computer program [8]. It is these flaws which malware exploits in order to gain unauthorized access to a system. Once access has been gained, the system is at the disposal of the attacker who launches the cyberattack. Although computer programs such as database software can be beneficial for organisations working with large amounts of information, a vulnerability in this software can potentially cause the information stored in the database to be accessible to the attacker. Two categories of software vulnerabilities: unpatched systems and lack of input validation will be examined next.

2.1.1. Unpatched Systems. Unpatched systems create an opportunity for malware to exploit, which leaves a system open to attack [9]. Despite this, many software programs notify users when new patches are available, which can be installed automatically and secure any vulnerabilities.

Organisations such as Microsoft release a number of patches every year [10]. Keeping up with all these patches can overwhelm a user. As a result of this, users may not be consistent when updating their systems, which is crucial since new vulnerabilities arise frequently. If these vulnerabilities are not patched, critical files will be accessible to an attacker and can then be stolen or corrupted. Lack of input validation in software is another vulnerability which is discussed below.

2.1.2. Lack of Input Validation. Input validation is a process which ensures that input data follows certain rules [8]. Examples of input data are usernames and passwords. Input data which is submitted on websites should be verified to make sure that it meets certain rules. No or incorrect validation will allow attackers to steal confidential information by using SQL injections which exploit the lack of validation. Attackers, such as cybercriminals, can enter SQL commands into fields in order to retrieve confidential information from the databases of online banking websites. Thus, it is important that username and password fields are validated while any commands entered are rejected. Although validation is implemented, password vulnerabilities will still be exploited. Password vulnerabilities will be discussed next.

2.2. Password vulnerabilities

Password vulnerabilities consist of weak passwords and are one of the most common vulnerabilities exploited by attackers [11]. Weak passwords such as an employee's name and date of birth can get exploited by an attacker. Due to this, an

[22] Y. Lin, X. Duan, C. Zhao, and L. D. Xu, *Systems Science: Methodological Approaches*. Boca Raton: CRC Press, 2012.

[23] M. S. Olivier, Information Technology Research: A practical guide for Computer Science and Informatics, 3 ed., Pretoria: Van Schaik Publishers, 2009.