The BESSPIN Scale

1 Introduction

The BESSPIN scale is the security figure of merit that is used to evaluate the SSITH hardware. As any security metric, it is more subjective than empirical, but more structured than qualitative. Since the SSITH program fundamentally relies on the MITRE CWEs enumeration to determine the various weaknesses against which it works on protecting, the BESSPIN scale is heavily inspired by the structure of CWSS. Nevertheless, to make it more intuitive and more relevant to SSITH, we designed our own metrics and rules.

The scale is represented by a percentage, where 0% means there is no protection against any SSITH CWEs, while an ideal SSITH processor is 100%-SSITH on the BESSPIN scale.

2 Formulation

To compute the BESSPIN scale, $\mathcal{B}$, for a given processor, we need the individual CWEs tests scores, which we group then into weakness categories. We use a structurally similar method to CWESS in combining these categories scores to obtain a separate score per vulnerability class, and the overall BESSPIN scale.

The SSITH CWEs, $\mathcal{S}$, is the set of all CWEs in the SSITH program picked from the MITRE CWEs. $\mathcal{S}$ is divided into a set of vulnerability classes (see Section 3). Each class, $\mathcal{V} \subset \mathcal{S}$, is also a set of CWEs. Each class is decomposed into some weakness categories or concepts. A category of weaknesses, $\mathcal{C} \subset \mathcal{V}$, is the set of CWEs whose description intersects with its definition; A CWE can belong to more than a single category. The class and SSITH CWEs can thus be written as:

$$\begin{aligned} \mathcal{V} &= \{cwe : cwe \in \mathcal{C} \ \wedge \ \mathcal{C} \in \mathcal{V}\} \\ \mathcal{S} &= \{cwe : cwe \in \mathcal{V} \ \wedge \ \mathcal{V} \in \mathcal{S}\} \end{aligned} \tag{1}$$

The BESSPIN security evaluation tool assigns each CWE a score value, $score_{cwe}$, from the SCORES enum object in scoreTests.py. Any value conveying a failure of any kind would disqualify the computation of the scale. This leaves us with the following acceptable scores: NONE, DETECTED, LOW, MED, and HIGH. As explained in the security evaluation document, HIGH means the weakness is critical and leads to the lowest score on the BESSPIN scale, and either NONE or DETECTED denotes the best score. For the sake of the scale, the CWE score can thus be normalized as follows:

$$\begin{aligned} \mathbf{S}(cwe) = & \\ 0, \quad & score_{cwe} = \text{HIGH} \\ 0.33, \quad & score_{cwe} = \text{MED} \\ 0.67, \quad & score_{cwe} = \text{LOW} \\ 1, \quad & score_{cwe} = \text{NONE or DETECTED} \end{aligned} \tag{2}$$

The score of a category of weaknesses is the arithmetic average of the scores of its CWEs:

$$\mathbf{S}(\mathcal{C}) = \overline{\mathbf{S}(cwe)}, \quad cwe \in \mathcal{C} \tag{3}$$

The BESSPIN coefficient, $\beta(\mathcal{C})$, is a measure that reflects the importance of a category of CWEs, $\mathcal{C}$, and its relevance to the SSITH program. It is formalized as:

$$\beta(\mathcal{C}) = \mathbf{TI}(\mathcal{C}) \times \mathbf{AV}(\mathcal{C}) \times \mathbf{ENV}(\mathcal{C}) \times \mathbf{SSITH}(\mathcal{C}) \tag{4}$$

where each of the factors can take the values 0.33, 0.67, and 1:

- **TI**($\mathcal{C}$): As defined by CWSS, the technical impact (TI) is the potential result that can be produced by the weakness, assuming that the weakness can be successfully reached and exploited. This is also related to the possible acquired privilege (AP). It can take the values: 1 for critical, 0.67 for moderate, and 0.33 for limited.
- **AV**($\mathcal{C}$): The access vector (AV) identifies the channel through which an attacker must communicate to reach the functionality that contains the weakness. It can take the values: 1 for user mode, network, or application, 0.67 for supervisor mode, i.e. operating system, and 0.33 for machine mode, i.e. hardware access.
- **ENV**($\mathcal{C}$): The environmental metric group for the BESSPIN scale constitutes of two concepts: 1. Business impact, which is the potential impact to the business or mission if the weakness can be successfully exploited. 2. The likelihood of the weakness to be discovered and exploited. **ENV**($\mathcal{C}$) can take the same values as **TI**($\mathcal{C}$). To reduce the subjectivity of the evaluation, we limit each concept to two scores only; either 0.5 for high or 0.17 for low. The summation of the two scores will thus be limited to the same three values as all other factors.
- **SSITH**($\mathcal{C}$): The SSITH factor is a measure of how relevant the weakness category $\mathcal{C}$ is to the SSITH program. It can take the values: 1 for strongly relevant, 0.67 for relevant, and 0.33 for somewhat relevant.

We can thus compute the BESSPIN scale for a particular vulnerability class, $\mathbf{S}(\mathcal{V})$:

$$\mathbf{S}(\mathcal{V}) = 100\% \times \frac{\sum_{\mathcal{C}_j \in \mathcal{V}} \Big(\beta(\mathcal{C}_j) \cdot \mathbf{S}(\mathcal{C}_j) \Big)}{\sum_{\mathcal{C}_j \in \mathcal{V}} \beta(\mathcal{C}_j)} \tag{5}$$

Similarly, we can compute the BESSPIN scale for a given processor:

$$\mathcal{B} = 100 \times \frac{\sum_{\mathcal{V}_i \in \mathcal{S}} \sum_{\mathcal{C}_j \in \mathcal{V}_i} \Big(\beta(\mathcal{C}_j) \cdot \mathbf{S}(\mathcal{C}_j) \Big)}{\sum_{\mathcal{V}_i \in \mathcal{S}} \sum_{\mathcal{C}_j \in \mathcal{V}_i} \beta(\mathcal{C}_j)} \ \% \text{ SSITH} \tag{6}$$

3 Vulnerability Classes and Categories

3.1 Buffer Errors

3.2 Permission, Privileges, and Access Control

3.3 Resource Management

3.4 Information Leakage

Information leakage is the exposure of parties to information which they are not intended to see. Beyond leaking sensitive data, this can include revealing information that enables subsequent attacks. Information can be leaked by directly sending data to unauthorized parties, or through side or covert channels that indirectly allow parties to learn something about otherwise secret data. It is divided to the following weakness categories:

1. **Information Exposure:** Data was *intentionally* disclosed to actors, but the data contained information that should not have been made available to those actors.
    - Covered by CWEs: 200, 201, and 202.

2. **Observable Discrepancy:** The product behaves differently in a way that is observable to an unauthorized user.
    - Covered by CWEs: 203, 205, and 206.
3. **Improper Sanitization:** Resources were made accessible to unauthorized users, but those resources were previously containing sensitive information, which *unintentionally* discloses them to those users.
    - Covered by CWEs: 200, 212, 226, 244, and 524.

3.5 Numeric Errors

3.6 Hardware/SoC

3.7 Injection

4 BESSPIN Coefficient Values

The BESSPIN coefficients document has the factors values used to compute the BESSPIN coefficients for each weakness category. They were evaluated by the BESSPIN team and DARPA.