Document N0599 **Dated 19 November 2015**

Rules to avoid programming language vulnerabilities

1. Validate input.

2. Check return values from subprograms.

3. Enable compiler static analysis checking and resolve compiler warnings.

4. Run a static analysis tool.

5. Perform range checking.

6. Allocate and free memory at the same level of abstraction.

7. Test constructs that have unspecified behavior for all possible behaviours.

8. Ensure that undefined or deprecated language features are not used.

9. Error detection, reporting, correction, and recovery should be an integral part of a system design.

10. Use only those features of the programming language that enforce a logical structure on the program.

11. Sanitize, erase or encrypt data that will be visible to others (for example, freed memory, transmitted data).

12. Develop and use a coding standard based on this document that is tailored to your risk environment.