1. Background. Software Unit Testing is a development process which requires that individual processes in software functionality be described, defined, and tested both independently of the containing applications or systems, and as a part of the system. Because Unit Testing promotes strict adherence to requirements as well as code re-use it is well suited for mission critical military applications and systems. Unit Testing is automated and can be applied continuously throughout the development cycle to include “end-to-end” testing. Current “System of Systems” (SoS) testing is a process which operates under the assumption that individual components are functional and does not employ automated unit testing. Because SoS testing can only address what can be observed by humans in real-time – it cannot be used to verify complex interoperability and security requirements that are inherent to modern networked systems.

2. Situation. Current military testing culture assigns responsibility for software functionality on vendors and programs. This involves assumptions that requirements have been clearly defined, implemented and tested before SoS testing occurs. The critical flaw in this approach is in the assumption of that which cannot be known. This culture reflects the reality of a time before automated software testing was invented. It is now possible to know all specified requirements and corresponding compliance with a verifiable level of exactitude. This knowledge can extend to levels of complexity which cannot be directly observed, but can be proven and verified in a consistent and repeatable way.

3. Requirements. The key component in software testing is the development and maintenance of fully defined and specific Unit Tests which can be universally applied. These tests represent operational knowledge which can only be provided by the customer. They are comprehensive requirements statements which are provided as machine readable artifacts which are used to perform automated verification. Current requirements for military software are provided in generalized, human readable formats which are subject to interpretation and cannot be reliably verified. In order to ensure specific, validated, consistent, reliable and inter-operable functionality within, across ,and between systems it is incumbent on the acquisitions community to develop, maintain and provide authoritative unit tests to all software development projects. Once created these tests can be disseminated for use in developmental and end-to-end SoS testing.

4. Security. As the number of computing devices increase in the scope of what are considered to be Systems of Systems, complexity increases. A key requirement for Cyber-Security is the ability to have full awareness of both functional and operational characteristics and status of all systems on a network. It is impossible to achieve this awareness though human observation. Manual security scans which are conducted for devices to be connected will eventually need to be automated. These scans follow the same principles as Unit Testing and require the same levels of detail. It is not unreasonable to assume that Unit Tests for security compliance will soon be required for all systems and will be conducted on live networks as a part of every automatic upgrade event.

5. Conclusion. As with all modern computing technologies the nature and scope of Unit Testing concept continues to evolve. For this reason engagement is imperative for any organization which seeks to verify or validate functionality and security performance on mission critical software systems.