

MLS Guidance for Infrastructure Programs

Multi-Level Definition

Multi-level end user applications can provide defense and intelligence users with a single interface that spans multiple networks or enclaves. Examples of this would include a single inbox for email across many security levels, a single C2 system across coalition systems, or a single search application that can query indexes on different networks.

The unique property of MLS applications is that they enable this fusion of data from multiple networks while preserving mandatory access controls, as defined in NSA's CNSSI 4009 glossary definition:

“Concept of processing information with different classifications and categories that simultaneously permits access by users with different security clearances and denies access to users who lack authorization.”

This is significantly different from cross-domain solutions, which either allow access (typically separate instances of Microsoft Windows in virtual machines or remote desktop sessions) or transfer (e.g. via data guards or data diodes). NSA's CNSSI 4009 definition for 'cross domain solution' is as follows:

“Information assurance solution that provides the ability to access or transfer information between two or more security domains.”

Note that cross domain access (UCDMO's baseline terminology) is essentially synonymous with the term 'multiple independent levels of security' (MILS). Also UCDMO treats multi-level security as a subset of 'cross domain' in the third category of its baseline – 'cross domain multi-level'.

Multi-Level Ecosystem

Within the multi-level ecosystem, there are three primary areas:

1. **Multi-level capable operating systems:** To be a multi-level capable or trusted operating system, an OS must be capable of processing data at multiple security levels while enforcing mandatory access controls. For example, it should be possible to run two processes of different classification levels on the same physical computer / CPU 'without' the risk of these processes interacting with each other in unauthorized ways. In order to be multi-level capable (as opposed to cross domain access or MILS), however, it must be possible for processes at different security levels to interact with each other in appropriately authorized ways to orchestrate a single system.
2. **Multi-level infrastructure applications:** These are backend applications that typically implement labeling in their application code to enforce mandatory access controls, e.g. a database that can store data at different classification levels. Note that the system must be capable of orchestrating requirements (e.g. queries) across multiple security levels in an

integrated fashion – having two completely separate databases in virtual machines on a single computer would not be classified as MLS.

3. **Multi-level end user applications:** These are client-side applications that present information at different security levels to the end user in a single interface while preserving mandatory access controls, e.g. an MLS email client that shows emails from different levels in a single inbox.

Note that both multi-level infrastructure applications and multi-level end user applications require a multi-level capable or trusted operating system underneath them in the stack on a computer device in order to function – the applications integrate with and leverage the trusted capabilities of the underlying operating system.

There are three key statements that can help qualify out systems from being MLS under the CNSSI 4009 definition:

1. **Windows Based:** Microsoft Windows does not have any implementation of labeling in its code base, and digital rights management controls in Windows are generally judged to be of insufficient security to qualify as being MLS capable. While Windows may form an important part of an MLS system (e.g. a guest OS in a hypervisor for a user interface), it cannot be the underlying operating system on the computer device for a system to be considered to be MLS.
2. **Tagging:** Tagging content, especially using XML, has become increasingly appropriate and popular in parts of the defense and intelligence communities, e.g. for attribute-based access controls. However, when used to prevent unauthorized document access, such application tagging provides discretionary access controls as opposed to mandatory, as the controls are not robust enough to prevent unauthorized user alteration, e.g. a user ‘copy and pasting’ from an email with one classification level into a new email with a lower classification level. A system that relies on tagging at the end user level to take ‘allow / deny’ decisions about transferring content through a cross-domain transfer data guard cannot be considered to be MLS.
3. **Pure Infrastructure Applications:** A labeled database can enforce mandatory access controls in the backend content repository and itself be considered as MLS capable. However, if users are interacting with the content through a single level (not multi-level) end user interface on an operating system that is not trusted (e.g. Windows), the system cannot enforce mandatory access controls, so the whole system cannot be considered to be MLS. In reality, the database is itself acting as a cross-domain transfer device, moving data from a lower security level to a higher security level in order to fuse it together with content in a unified view for the user.

In order for a system to be considered as being multi-level, then, the end user interface must be multi-level, so that users with different clearances can process information with different classifications and categories while access to users who lack authorization is denied (mandatory access controls). Note that, it is possible for a multi-level end user application to draw on content from different single level backend applications and services while enforcing mandatory access controls, so while a multi-level capable or trusted operating system is always required for an MLS system, multi-level infrastructure applications are not a prerequisite.

Multi-Level Requirements

Often, in defense and intelligence acquisition cycles, the infrastructure acquisition becomes divorced from the applications acquisition. While this decoupling may make sense programmatically and contractually, it can sometimes lead to infrastructure being acquired that does not support the mission requirement set, which is itself more application and capability focused.

The following set of sanitized requirements provides examples of capabilities that can be mandated in an infrastructure acquisition process to ensure support of MLS end user applications:

- **MLS Interface:** The system should be capable of providing unified access to a specified range of security enclaves (domains) by means of an MLS end user interface, providing a common operating picture that spans multiple security enclaves (domains).
- **MAC:** The system should be capable of enforcing mandatory access controls at the user interface level for MLS applications, so that actions such as unauthorized 'copy and paste' by users can be prevented.
- **Independently Labeled Application Windows:** The system should support application windows in the user interface that are appropriately marked for classification and are decoupled from the underlying operating system in which the application is hosted. Specifically, application windows with different classification levels should be able to overlap and intermingle as a single set in the user interface, without being constrained to being grouped into separate sets at each classification level (e.g. locked to an operating system desktop). In the context of hypervisors, this is often termed 'seamless windows'.
- **Cross-Domain Communication:** The system should provide mechanisms to allow appropriately authorized communications between security enclaves (domains).
- **Shared Administration Services:** The system should provide administrative interfaces to control cross-domain policies and audit the use of cross-domain functions.
- **Others:** Additional, trusted infrastructure services (e.g. directory services) are required to define users, their clearance range, access to multi-level services, and any other access control policies that are used to enforce multi-level security policies.

Multi-level security offers compelling capabilities to warfighters and intelligence analysts supporting them – especially in scenarios that involve joint or coalition tasks. These new capabilities can be delivered in addition to the infrastructure savings that can be achieved through consolidating multiple single level PCs using trusted desktop devices, so long as the trusted desktop devices selected are capable of supporting multi-level end user applications.