Introduction [1]

This manual describes the UNICOS multilevel security (MLS) feature (also referred to as *security enhancements*) and how you, as a nonadministrative user, can use it. This chapter describes the following:

- Basic security concepts
- The U.S. Department of Defense (DoD) criteria used to define security
- The divisions used by the DoD to classify trusted systems
- How the UNICOS MLS feature is implemented on Cray Research mainframes

Warning: In previous releases of the UNICOS operating system, the term *Trusted UNICOS* was used to refer to the configuration that most closely approximated the B1 evaluated configuration of UNICOS release 8.0.2. In the UNICOS 10.0 release, this configuration is referred to as the *Cray ML-Safe configuration* of the UNICOS operating system. Although the Cray ML-Safe configuration of the UNICOS operating system is not an evaluated product, this configuration fully supports all functionality described in the B1 evaluation criteria.

1.1 Introduction to security concepts

The following sections introduce concepts of a trusted (Cray ML-Safe) computer environment and the UNICOS MLS feature.

1.1.1 Concepts of computer security

On traditional UNIX systems, the integrity of both system and user data is preserved by using both simple passwords and permission mode bits. Security is usually left up to the individual user without much intervention on the part of a system administrator. Such a system does not provide many mechanisms for preventing deliberate destruction or corruption of data.

In some computing environments, such a system may not meet the need to protect data, and a trusted environment is used instead. A trusted computer environment should enforce policies that prevent the following (and are summarized in Figure 1, page 3):

- Unauthorized system access - A trusted system should have mechanisms that prevent users from bypassing the authentication process or guessing other user's passwords, plus provide the means necessary to detect such attempts.

- Unauthorized data object access - A trusted system should prevent a user without the correct permissions from gaining access to protected parts of the system (for example, a file, a terminal, or memory).

- Excessive use of system resources - A trusted system should deny a user the ability to monopolize system resources so that other users are denied the resources necessary to perform their work.

- Loss of system integrity - A trusted system should prevent a user from inadvertently or deliberately destroying a data object that is not his or hers to destroy.

The UNICOS MLS feature provides mechanisms not found on a non-MLS UNICOS system to protect both system integrity and sensitive information. The following sections describe the DoD specifications that shaped the UNICOS MLS feature and explain how the feature is implemented on a Cray Research computer system running the UNICOS system.

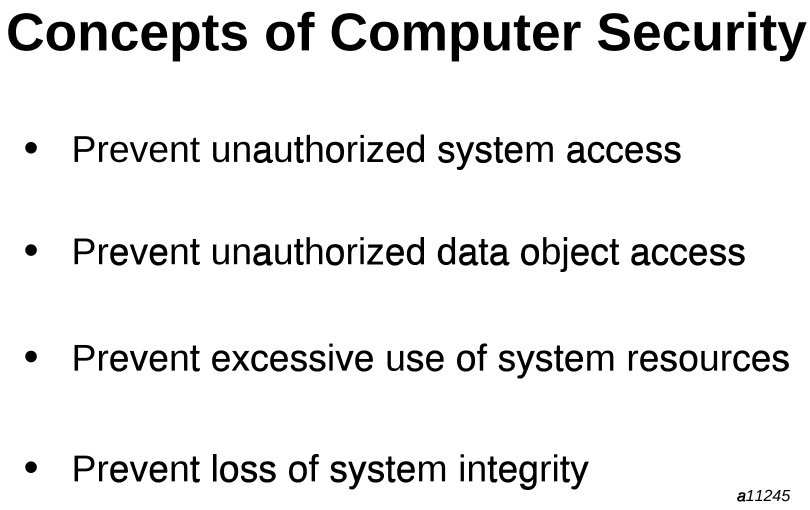
1.2 DoD criteria for trusted systems

Design specifications for the UNICOS MLS feature were derived from the *Department of Defense Trusted Computer System Evaluation Criteria* (also known as the *Orange book*). These criteria outline the system software capabilities needed to satisfy government security requirements and the divisions that classify computer systems according to how well the security criteria are met.

The evaluation criteria are divided into three categories: security policy, accountability, and assurance.

Computer systems are classified in one of four divisions, D through A, with A representing the most trusted system.

The following sections discuss the criteria in more detail. For more information on the four divisions, see Appendix A, page 97.



Concepts of Computer Security

- Prevent unauthorized system access
- Prevent unauthorized data object access
- Prevent excessive use of system resources
- Prevent loss of system integrity

a11245

Figure 1. Concepts of computer security

1.2.1 Security policy

The DoD criteria state that a trusted system must enforce an explicit and well-defined security policy. A security policy is defined as the set of rules and practices by which a system regulates the processing of sensitive information. The criteria also state that each subject and object in the system be marked with a clearance or classification label, respectively, for controlling access.

System security can be defined by three security properties: simple security property, *-property (star property), and discretionary security property.

The simple security property states that no subject can read an object unless the subject's security level is greater than or equal to the object's security level and the subject's compartments are a superset of the object's compartments; this is referred to as *no read up*. The UNICOS MLS feature supports this property by using the mandatory access controls.

The *-property states that to write to an object, the object's current security level must be greater than or equal to the subject's security level and the object's compartments must be a superset of the subject's compartments. This property prevents a subject from writing information from one object into another object with a lower classification and is known as *no write down*. The UNICOS MLS feature supports the *-property by using the mandatory access controls.

The discretionary security policy allows a subject with the authority (for example, the owner of an object) to define both the subjects that can access an object and the mode of access. The UNICOS MLS feature supports the discretionary security policy by using discretionary access controls.

1.2.2 Accountability

Individual accountability is the key to securing and controlling any system that processes information on behalf of users or groups of users. The DoD criteria identify the following accountability requirements:

- Individual subjects must be identified and authenticated. The trusted system maintains the identification and authentication information; this information must then be associated with every security-relevant user activity in the trusted system.
- Security audit information must be selectively kept and protected so that events related to security can be traced to the responsible party. This requirement implies a need for a security logging device that permits regular surveillance of system security. It is also essential that an authorized agent (in most cases, the security administrator) be able to selectively access and evaluate security audit information. Of course, audit data must be protected from alteration and unauthorized destruction.

1.2.3 Assurance

The DoD assurance objective states that the computer system must contain hardware and software mechanisms that can be independently evaluated to provide sufficient assurance that the system enforces the security policy and accountability objectives. These mechanisms must be continuously protected against unauthorized changes. To satisfy the DoD evaluation criteria, the UNICOS system must guarantee that the security policy is effectively enforced and must be subjected to analysis and tests, the completeness of which can be assured.

1.3 The UNICOS implementation of MLS

The following topics are described in this section:

- An overview of a Cray ML-Safe configuration of the UNICOS system
- An overview of discretionary access control implementation

- An overview of mandatory access control implementation

1.3.1 Cray ML-Safe configuration of the UNICOS system

The Cray ML-Safe configuration of the UNICOS operating system supports processing at multiple security labels and system administration using only nonsuper-user administrative roles. The Cray ML-Safe configuration consists of the subset of UNICOS software that offers these capabilities. The Cray ML-Safe name does not imply maintenance of the UNICOS 8.0.2 security evaluation.

> **Warning:** For the UNICOS 10.0 release, the functionality of the Trusted UNICOS system has been retained, but the `CONFIG_TRUSTED` option, which enforces the conformance to the strict B1 configuration, is no longer available. All references to the Trusted UNICOS system have been replaced by the *Cray ML-Safe configuration* in UNICOS 10.0 documentation.

A Cray ML-Safe configuration provides a fully functional, practical, and usable trusted operating system when integrated into a heterogeneous network set of Cray ML-Safe components. Contact your site security administrator to see if your site is using the Cray ML-Safe configuration.

If your site is using a Cray ML-Safe configuration, this means your system configuration is using many specific security mechanisms that enforce the requirements in the *Trusted Computer System Evaluation Criteria* (TCSEC). These mechanisms include many of mechanisms, such as security labels and access control lists (ACLs), plus the use of the privilege mechanism, multilevel directories (instead of wildcard directories), CrayML-Safe mail, and many other configuration options that are used to define a Cray ML-Safe environment.

In addition, the Cray ML-Safe configuration uses the basic security option (BSO) and common IP security option (CIPSO) at the IP layer to provide packet labeling across a network. This allows connections to other trusted systems in a heterogeneous set of Cray ML-Safe components. The following network services and protocols rely and build on BSO and CIPSO to maintain object labels across a network:

- TCP/IP
- RPC
- NFS
- NQE/NQS
- User sockets

- `telnet`(1B), `rlogin`(1B), `rexecd`(8), `rsh`(1), `rcp`(1), `ftp`(1B), `lpd`(8), and `finger`(1B)

Other major Cray Research products included in the set of Cray ML-Safe components are the GigaRing mechanism, the system workstation (SWS), the I/O subsystem model E (IOS-E), the SSD solid-state storage device model E (SSD-E), the operator workstation (OWS) model E, the maintenance workstation (MWS) model E, the tape subsystem, the Cray Data Migration Facility (DMF), and Cray/REELlibrarian (CRL).

In addition to this manual, you should also read the Cray ML-Safe and MLS (security enhancements) information in the following manuals:

- *TCP/IP Network User's Guide*, Cray Research publication SG–2009
- *Tape Subsystem User's Guide*, Cray Research publication SG–2051
- *NQE Administration*, Cray Research publication SG–2150
- *Cray/REELlibrarian (CRL) User's Guide*, Cray Research publication SG–2126

For the most part, the use of these Cray ML-Safe features should be transparent to you as a nonadministrative user within the restrictions of a Cray ML-Safe configuration of the UNICOS system. If you encounter difficulties, contact your security administrator. Throughout this manual, assume that the explanations apply to both the basic UNICOS configuration and the Cray ML-Safe configuration, unless otherwise indicated.

1.3.2 Discretionary access controls

Discretionary access controls are rules that define a subject's access to an object, based on the subject's identity and groups to which it belongs, and the object's ownership, permissions modes, and access control list (ACL). The mode permission bits (that is, read (r), write (w), and execute (x) bits) are masked against the ACL entries to allow the owner of a file to control both the subjects who can access the file and the type of access (rwx) granted. For a finer granularity of control than that offered by the mode permission bits, the file's owner can also explicitly deny a subject access to the file by making a null/none (n) entry in the ACL for that subject. See Section 3.3, page 55, for more information on this masking operation.

File owners usually establish the discretionary access rules for files they own; however file access is also governed by the mandatory access controls established by the security administrator.

Refer to the `spacl`(1) man page and chapter Chapter 3, page 37, for more information on how ACLs are used and examples that show how to create and maintain ACLs.

1.3.3 Mandatory access controls

Mandatory access controls are rules that control how users access a system in order to prohibit the unauthorized disclosure of any system or user data. The mandatory part of the definition comes from the fact that the enforcement of the controls is done by administrators and the system, and is not left up to the discretion of users as is done with discretionary access controls (described in the previous section).

The UNICOS system uses mandatory access rules to form the UNICOS security policy. This policy controls access based directly on a comparison of the subject's clearance and the object's classification.

Broadly stated, the UNICOS security policy enforces the following rules for nonadministrative users:

- A subject cannot read an object unless the subject's clearance is greater than or equal to the object's classification.
- A subject cannot write or append to an object unless the subject's clearance is equal to the object's classification.
- Any object created by a subject inherits a security label equal to that of the subject.
- A nonadministrative subject cannot lower its own clearance. On a Cray ML-Safe configuration, a nonadministrative subject cannot raise or lower its own clearance.
- A subject cannot change an object's classification.

A subject's clearance and an object's classification consist of security levels and compartments, which form security labels. The security label is the focus of the UNICOS security policy and is defined for you by your security administrator in the user database (UDB).

A security label should be thought of as a single entity and is treated as such when the UNICOS system uses it for access control decisions. However, many of the UNICOS MLS interfaces (that is, user commands) address the component parts (that is, the security level and security compartment) of the label. Because of this situation, it is easier to understand the UNICOS security policy if you first understand the concepts of a security level and compartment and how they are used to form a security label.

1.3.3.1 Security levels

As stated earlier, a security label consists of a security level and a set of security compartments. For nonadministrative users, a security level can be a hierarchical value from 0 to 16, that indicates the classification of an object or the clearance of a user. System high and system low levels are also used. See Section 1.3.3.6, page 17, for more information on these levels.

Depending on how your UNICOS system is installed, default names are given to each level (`level0`, `level1`, `level2`, and so on). Your security administrator can change these names to accommodate the needs of your site. For example, `level0` could be named `unclassified`, `level1` could be named `classified`, `level2` could be named `secret`, and so on.

In the following example, assume that a user has been assigned a security level of 13 and the system’s range is 0 to 16. According to the broadly stated rules in Section 1.3.3, page 7 (and ignoring the use of compartments), this user would be constrained by the following rules:

- The user can read objects with security levels of 13 or less, but cannot read an object with a security level of 14, 15, or 16.
- The user cannot write to an object unless the object’s security level is also 13.
- All objects created by the user would be assigned security level 13.

Because of the hierarchical nature of security levels, only one security level can be active for a user at any time; an object can be labeled with only one security level.

1.3.3.2 Security compartments

A security compartment is a nonhierarchical value that indicates the type or topic of information contained in an object for which the subject is cleared. Unlike security levels, you can have more than one compartment active at a time and objects can have more than one compartment assigned to them.

The UNICOS system supports 63 compartments for site definition and use. Compartments can be used to separate groups working on different projects on a UNICOS system. For example, if group one is working on a propeller project, the security administrator could assign the `prop` compartment to all users in that group, while all of group two, working on weather forecasting, could be assigned the winds compartment. A director of both projects could be assigned both compartments.

According to the broadly stated rules in Section 1.3.3, page 7 (and ignoring the use of security levels), users with these compartments would be allowed the following when attempting read access:

- A user with the winds compartment has read access to all objects whose compartment sets are a subset of wind. This means that a user with the winds compartment has read access to an object with the winds compartment or an object with an empty compartment set.
- A user with the prop compartment has read access to all objects whose compartment sets are a subset of prop. This means that a user with the prop compartment has read access to an object with the prop compartment or an object with an empty compartment set.
- A user with the compartments winds and prop has read access to an object with an empty compartment set, an object with the winds compartment, an object with the prop compartment, or an object with a compartment set that includes both winds and prop.

1.3.3.3 Definition of dominance

With security levels, the concept of equal or not equal is easy to apply, as levels are hierarchical. If a subject's security level is the same as an object's level, they are equal. If not, then one is less than or greater than the other.

Determining this type of relationship between compartments is not easy to do as compartments are not hierarchical. If you use the concept of sets, subsets, and supersets when comparing compartments, the job of comparing compartment sets becomes easier.

To avoid the comparison issues introduced by the hierarchical nature of levels versus the nonhierarchical nature of compartments, it is necessary to understand the concept of dominance between security labels.

A security label is said to *dominate* another security label if the first security label meets the following criteria:

- The first security label has a security level that is greater than or equal to the security level of the second security label.
- The first security label has a compartment set that is equal to or a superset of the compartment set of the second security label.

A security label is equal to another security label if both security labels have exactly the same security level and compartment set.

By using the concepts of a security label and dominance, the broadly stated rules in Section 1.3.3, page 7, translate as follows:

- A subject cannot read an object unless the security label of the subject dominates the security label of the object.
- A subject cannot write to an object unless the security label of the subject is equal to the security label of the object.
- When a subject creates an object, the object inherits the security label of the subject.
- A subject cannot change its own security label to a security label that does not dominate its current security label.
- A subject cannot change the security label of an object.

1.3.3.4 User's security label ranges

You can be assigned a range of security labels that you can use on a UNICOS system. This range is defined in the user database (UDB) by your security administrator in terms of a minimum security level, a maximum security level, a minimum compartment set, and an authorized set of compartments.

In the simplest configuration, the minimum security level and an empty compartment set form your minimum security label; the maximum security level and an empty compartment set form your maximum security label.

You are allowed to log into the system at any label that dominates your minimum label and is dominated by your maximum label. Once you have logged in, however, the subject created on your behalf can only have a single label at any given time.

In addition to your assigned label range, you are assigned a default login label by your security administrator. On a UNICOS system **not** running a Cray ML-Safe configuration, the default label defines the label at which you log in, assuming no other consideration is made. See Section 2.1, page 21, and Section 2.2, page 24, for more information on how your security label is determined.

It is possible to configure the UNICOS system to define a minimum compartment set or to use your default label as the minimum label. This allows your security administrator to define minimum user security labels with nonempty compartment sets. See your security administrator for information on how your system is configured.

On a Cray ML-Safe configuration, the default label is not very useful, as your login label is based on the label of the socket connection itself and not any label requested by you. See Section 2.2, page 24, for more information on how your security label is determined on a Cray ML-Safe configuration.

1.3.3.5 Examples of security labels

In the previous sections, the concept of a security label was explained. The following sections provide some examples of how labels can be used to protect data on a UNICOS system. These examples show the following:

- The use of a label with an empty compartment set
- A system that uses labels that all contain the same level with different compartment sets
- A system that uses labels with different levels and compartment sets

1.3.3.5.1 Example of using hierarchical security labels

In Figure 2, data is stored in levels 0 through 5 with no compartments. For the following examples, security labels are represented as a level number and a set of compartments separated by a colon. For example, level 3 and a compartment set of AB would be represented as 3:AB.

In Figure 2, a user called Mary has a security label range that allows her to log in at any label from 0:NONE to 4:NONE. Mary’s default security label is 3:NONE. Mary’s default security label becomes her active label when she logs in.

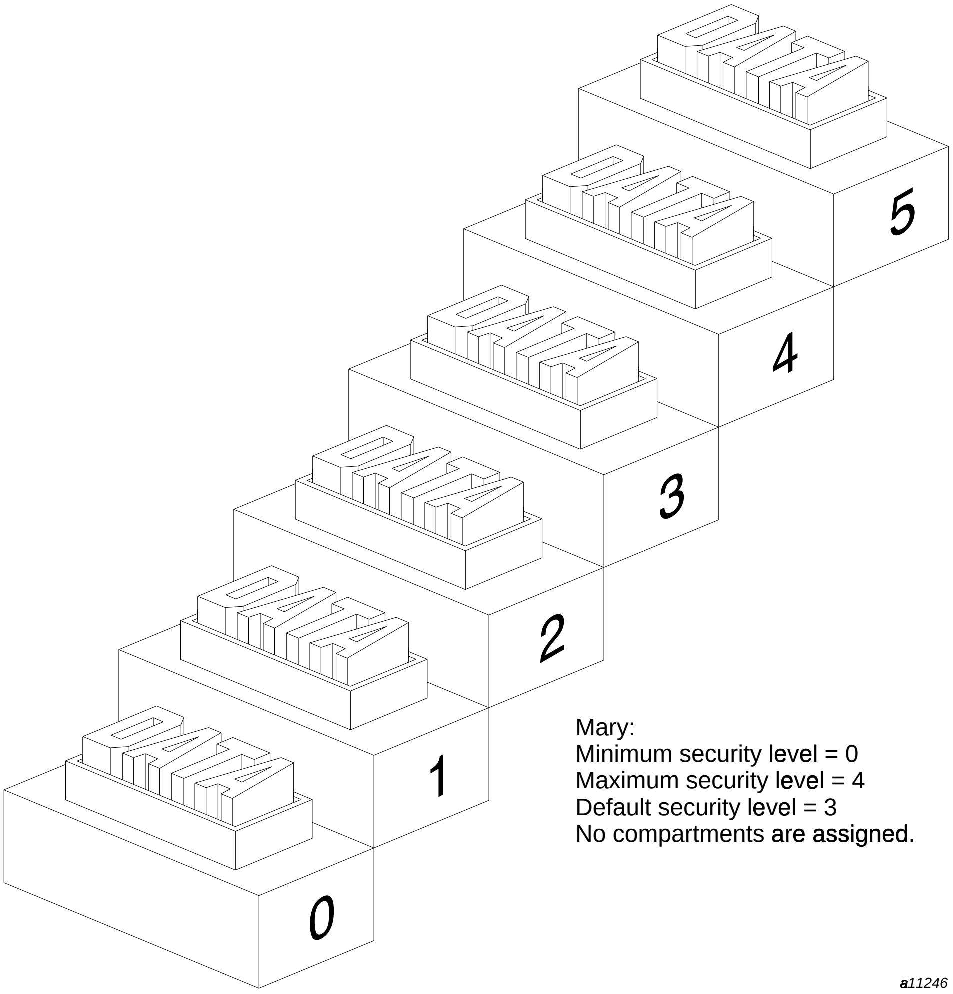
> **Note:** On a Cray ML-Safe configuration, Mary may not be assigned her default security label when she logs in. Her label depends on the socket connection’s security label through which she connects to the Cray ML-Safe configuration. See Section 2.2, page 24, for more information on how your security label is determined. For this example, assume that Mary logged in with a label of 3:NONE.

When Mary logs in at her default label, she can read and change data that has a security label of 3:NONE. She can also read data that has a label of levels 0:NONE, 1:NONE, 2:NONE, or 3:NONE. On a UNICOS MLS system, Mary can also raise her label to 4:NONE.

> **Note:** On a Cray ML-Safe configuration, Mary cannot raise her security label after she logs in. Her process label range, which is derived from the label range of her connection and her UDB label range, allows her to work only at the label at which she logged in. For this example, assume that Mary logs out and logs in to change her security label.

If she raises her label, Mary can change data that has a security label of 4:NONE (although she cannot change data with a label of 3:NONE). She can also read data with a label that has a level of 0:NONE, 1:NONE, 2:NONE, 3:NONE, or 4:NONE.

Mary cannot raise her security label to 5:NONE, as it is out of her defined range. Also, once she raises her label from 3:NONE to 4:NONE, she cannot lower it back; she must log off and log back on to have a label of 3:NONE.



Mary:
Minimum security level = 0
Maximum security level = 4
Default security level = 3
No compartments are assigned.

Figure 2. Example of using hierarchical security labels

1.3.3.5.2 Example of using nonhierarchical security labels

In Figure 3, data is labeled with compartments A (Data 1); B (Data 2); C (Data 3); A and B (Data 4); A and C (Data 5); B and C (Data 6); A, B, and C (Data 7); and with no compartments (Data 8); all the labels contain a security level of 0.

In Figure 3, page 14, a user named Mary has been assigned a security label range that includes 0:NONE to 0:AB. Her default security label is 0:B.

Mary can log in at a label of 0:NONE, 0:A, 0:B, or 0:AB. Mary can never log in with a label containing compartment C, because compartment C is not part of her authorized compartment set.

> **Note:** Your security administrator can configure your system so that your default security label is treated as the minimum security label. If this is true for your site, then in this example Mary would not be allowed to log in with a label containing the empty compartment set or a compartment set containing only compartment A. Her only choices would be label of 0:B or 0:AB.

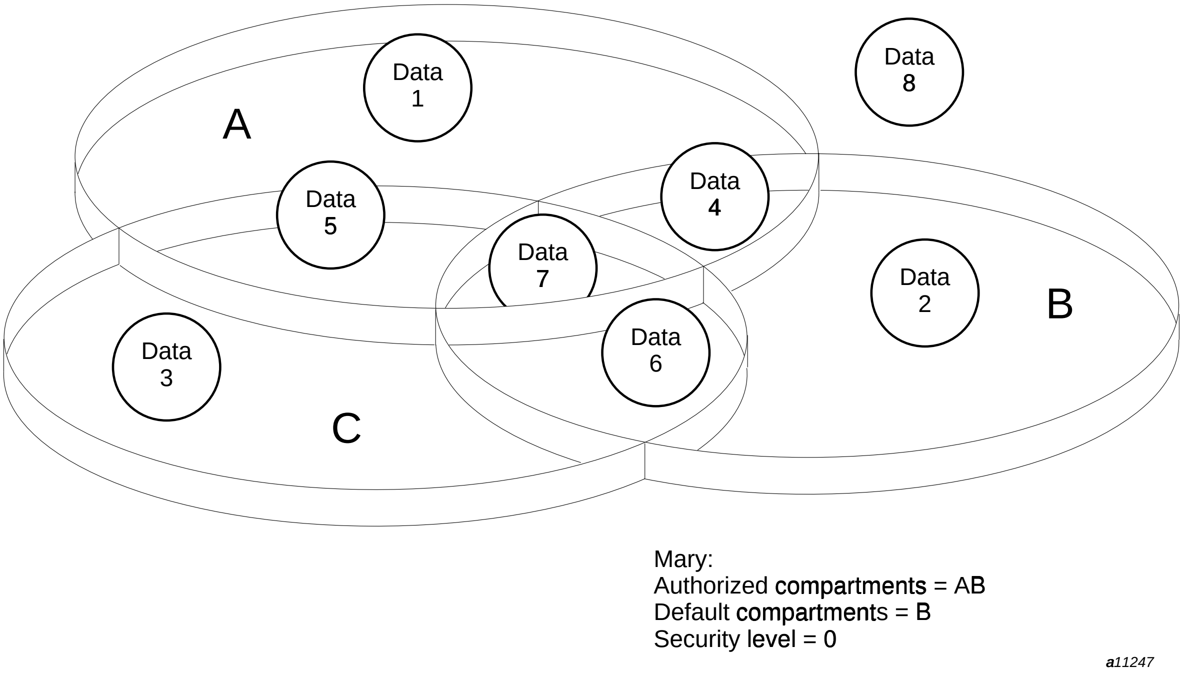


Figure 3. Example of using nonhierarchical labels

Table 1 shows all possible combinations of data that can be read or written (in Figure 3) by users having labels containing all possible combinations of compartments and a level of 0.

Table 1. Accessing data with nonhierarchical labels

Active label	Data that can be read	Data that can be written
0:A	Data 1,8	Data 1
0:B	Data 2,8	Data 2
0:C	Data 3,8	Data 3
0:AB	Data 1,2,4,8	Data 4
0:AC	Data 1,3,5,8	Data 5
0:BC	Data 2,3,6,8	Data 6
0:ABC	Data 1 - 8	Data 7
0:NONE	Data 8	Data 8

1.3.3.5.3 Example of security labels

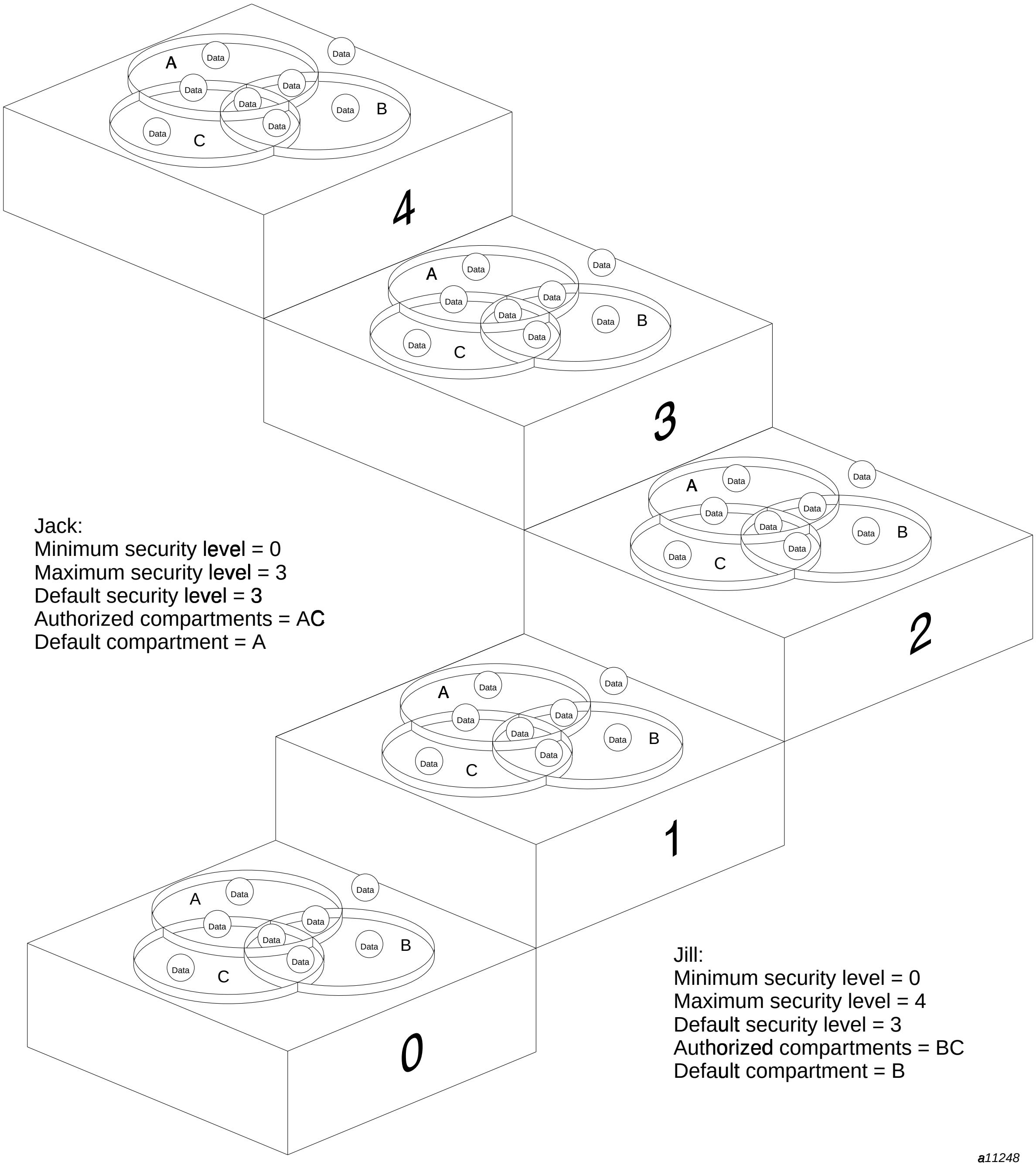
Figure 4 shows a system on which labels can contain 0:ABC through 4:ABC. Two users, Jack and Jill, have the following security label ranges:

- Jack is allowed labels that range from 0:NONE to 3:AC.
- Jill is allowed labels that range from 0:NONE to 4:BC.

Jack’s default label is 3:A and Jill’s default label is 3:B. If Jill creates a data object at her default label (3:B), Jack cannot read the data object (assuming he is using his default label), as his label does not dominate the label of the data object created by Jill. The same restriction would apply to Jill if she tried to read a data object created by Jack. Her label (3:B) would not dominate the label of the security object created by Jack (3:A).

The label range defined for Jack allows him to login at a number of different labels (for example, 3:C, 2:C, 3:a, and 3:AC). Jill’s label range allows her to log in at a number of different labels also (for example, 3:C, 4:C, 3:B, 4:B, 2:C, and 4:BC).

If Jill logged in at 3:C and created a file, Jack could read the file if he had logged in at 3:C or 3:AC. He could not read the file if he logged in at 3:A or 2:C, as neither of these labels dominates the label 3:C.



Jack:
Minimum security level = 0
Maximum security level = 3
Default security level = 3
Authorized compartments = AC
Default compartment = A

Jill:
Minimum security level = 0
Maximum security level = 4
Default security level = 3
Authorized compartments = BC
Default compartment = B

Figure 4. Example of security labels

If Jack logged in at 3:C and created a file, Jill could read the file if she logged in at 3:C, 4:C, or 4:BC, but not if she logged in at a label of 3:B, 4:B, or 2:C, as these files do not dominate the label 3:C.

The important concept to understand from these examples is that both components of the security label must dominate the components of the object's security label in order to read it. So, even if your security level is greater than the object's security level, if your compartment set is not a superset of the object's compartment set, you are not allowed read access to the object.

Although some UNICOS MLS configurations allow you to log in and then change your security label, you may not be able to obtain some labels without logging out and logging in again. For example, if Jill logs in at her default label (3:B), she could not change to a label of 3:C or 2:B, as neither of these labels dominates 3:B. She would have to log off and then log on at 3:C or 2:B.

1.3.3.6 System high and system low security labels

The UNICOS system uses the system high (`syshigh`) and system low (`syslow`) security labels to protect system data and software. These security labels fall outside the range of nonadministrative users.

The `syshigh` label is assigned to system-private databases (for example, `/etc/udb` or the audit log files) and is not dominated by any user security label. This means that system files protected by `syshigh` cannot be read or written to by an unauthorized user.

The `syslow` label is assigned to the majority of Cray ML-Safe binaries, and public databases and directories (for example, `/etc/passwd` or `/bin/cat`). The `syslow` label is dominated by all user labels, but is not equal to any of them. This means that system files protected by `syslow` can be read, but not written to, by unauthorized users, making them available to nonadministrative users, yet protecting the contents of the files.

1.3.4 Special security levels

Your UNICOS system may use a wildcard label (security level 63). This label allows directories to contain files at different security levels. The wildcard security mechanism cannot be used on a Cray ML-Safe configuration; multilevel directories (MLDs) must be used. See Section 5.1.3, page 71, for more information on MLDs.

Security level 51 indicates the system low (`syslow`) label and security level 54 indicates the system high (`syshigh`) label, as explained in the previous section.

1.3.5 Accountability objective

In the UNICOS system, individual accountability is achieved by the following means:

- A user authentication process, activated at the time of system access (for example, login or NQS submission), identifies and authenticates each user; minimum and maximum security levels, an active security level, active compartments, authorized compartments, an active and maximum integrity class, an active category, authorized categories, and a set of permissions granted by the security administrator define the security policy for each authenticated user.
- A security log, maintained by the system, produces an audit trail of user activity. The security log is accessed and processed by the administrative system utility `reduce`(8).

The UNICOS MLS feature provides a set of audit and monitoring utilities that process security log entries and check various other system occurrences to report on individually accountable actions, security policy violations, system integrity, and sensitive information handling. The audit and monitor utilities used are the `reduce`(8), `spfilck`(8), and `spcheck`(8) commands. See the appropriate man page for more information on these commands.

1.3.6 Assurance objective

In the UNICOS system, assurance measures are applied as follows:

- The reference monitor concept is applied within the UNICOS kernel and are always invoked. The reference monitor concept is explained in more detail in the following section.
- The least privilege principle is applied; details of how this principle is applied are found in Section 1.3.6.2, page 19.
- The UNICOS system provides the ability to separate administration functions through the use of categories. See Section 1.3.6.2, page 19.
- Provisions are made to document or audit the use of some covert channels. Information on covert channels can be found in *General UNICOS System Administration*, Cray Research publication SG–2301.

1.3.6.1 Reference monitor concept

The DoD evaluation criteria define a reference monitor concept as an access control concept that refers to an abstract machine that mediates all accesses made by subjects to objects. The reference monitor must be contained entirely in the kernel and must satisfy the following design requirements:

- It must always be invoked.
- It cannot be bypassed.
- It must be safe from tampering.

The UNICOS system supports the reference monitor concept through the entire set of kernel mechanisms that control access to objects, creation of objects and subjects, assignment and use of privilege, as well as the set of kernel mechanisms that maintain subject and object attributes, perform auditing functions, and preserve the isolation of the set of Cray ML-Safe components.

Because the reference monitor resides and executes entirely in the UNICOS kernel, it is always invoked. Because of its location, users cannot tamper with the reference monitor logic or data. It cannot be bypassed.

1.3.6.2 System management

System management is the set of security-related administrative and operational policies and procedures that are needed to maintain system security. One of the procedures needed for system security is the ability to separate operator and administrator functions. The following mechanisms support this need:

Mechanism	Description
The super-user mechanism	This mechanism is the traditional administrative policy that is enforced on UNICOS systems not using the MLS feature. The `root` user ID is used to administer the system. In a UNICOS MLS environment, the super user can override virtually all UNICOS MLS restrictions. This mechanism is often referred to as the `PRIV_SU` system.
The PAL-based privilege mechanism	This mechanism uses privilege assignment lists (PALs) to map active administrative categories to

a set of granular privileges that can be used for the duration of a command execution.

Warning: The UNICOS 10.0 system will support only the following configurations:

- A super-user (`PRIV_SU` enabled) system with PALs
- A nonsuper-user (`PRIV_SU` not enabled) system with PALs

Your site administrator determines which mechanism is used at your site. On a Cray ML-Safe configuration, only a strict PAL-based privilege environment can be used. In general, these mechanisms are used to regulate administrative work, so the impact on nonadministrative users is minimal. For more information on the implementation and use of the mechanisms, see *General UNICOS System Administration*, Cray Research publication SG–2301.