Australian Government
Department of Defence
Defence Science and
Technology Organisation

Review of Data Integrity Models in Multi-Level Security Environments

Patrick Garnaut and Jonathan Thompson

Command, Control, Communications and Intelligence Division

Defence Science and Technology Organisation

DSTO–TN–0971

ABSTRACT

As there is an increased reliance upon information in defence operations and in network centric warfare, ensuring the security of the information systems involved is becoming an increasingly important objective. Within the realm of national security, research and development has predominantly focused on ensuring the confidentiality of data within these systems. There has, however, been recent recognition of the role that integrity has in the overall security of a system. To facilitate a better understanding of data integrity as it relates to information security, this paper provides a review of the associated literature including the prevalent data integrity models, evaluation mechanisms and integrity centric implementations.

Published by

DSTO Defence Science and Technology Organisation
PO Box 1500
Edinburgh, South Australia 5111, Australia

Telephone: (08) 7389 5555
Facsimile: (08) 7389 6567

February, 2011

APPROVED FOR PUBLIC RELEASE

Review of Data Integrity Models in Multi-Level Security Environments

Executive Summary

As there is an increased reliance upon information in defence operations and in network centric warfare, ensuring the security of the information systems involved is becoming an increasingly important objective. Within the realm of national security, research and development has predominantly focused on ensuring the confidentiality of data within these systems. There has, however, been recent recognition of the role that integrity has in the overall security of a system. To facilitate a better understanding of data integrity as it relates to information security, this paper provides a review of the associated literature including the prevalent data integrity models, evaluation mechanisms and integrity centric implementations. Specifically, the Biba and Clark-Wilson data integrity models are discussed along with their variations. Evaluation mechanisms such as the Trusted Computer System Evaluation Criteria, Common Criteria and Director of Central Intelligence Directives are presented, as are the Minos, TrustedBSD, XTS-400, RS-Linux and Trusted IRIX CMW integrity centric implementations.

THIS PAGE IS INTENTIONALLY BLANK

Contents

1 Introduction

As there is an increased reliance upon information in defence operations and in network centric warfare, ensuring the security of the information systems involved is becoming an increasingly important objective. Data security is composed of three major areas: confidentiality, integrity and availability. Within the realm of national security, research and development has predominantly focused on ensuring the confidentiality of data within these systems. The reason for this is that the leaking of classified information can damage the security of the nation. There has, however, been recognition of the role that integrity has in the overall security of a system. This is evident from the addition of integrity verification criteria to evaluation schemes such as the Common Criteria (CC) and Director of Central Intelligence Directives (DCIDs).

The concept of data integrity is critical to a multitude of information systems and processes, particularly in the commercial world with regard to underpinning the credibility of transactions, and also in regard to the integrity of audit trails in both financial and government scenarios, as well as the integrity of the underlying software. This paper, however, focuses on the integrity of data within Multi-Level Security (MLS) systems, an area in which the traditional focus has been primarily on data confidentiality, but in which the integrity of the data is also critical to mission success. *"A Multi-Level Security (MLS) system is one where a single device is used to communicate or process data at different security classifications"* [Zellmer 2003]. In this context, the definition of data integrity adopted in this paper is the property of the system adhering to a well-defined code of behaviour, that is, the system will perform as was intended by its creator. No a priori statement as to the properties of this behaviour are relevant to the definition.

The integrity of data within information systems is becoming an equally important factor of security within defence as the data becomes more integrated and relied upon in operations and decision making. In some cases, the protection from malicious modification of crucial information can be of greater importance than preventing unauthorized observation [Biba 1977]. This paper reviews the theory and implementation of data integrity models that have been proposed in the literature. Although a brief qualitative evaluation of the models is presented, a discussion of the critical area of measuring data integrity, that is, quantitative data integrity metrics, is deferred

There have been two prominent models of data integrity: the Biba integrity model and the Clark-Wilson integrity model. These are outlined in sections 2 and 3 of this paper respectively. Extensions to the Biba and Clark-Wilson integrity models, as proposed in the literature, are also discussed. Section 4 provides an overview of evaluation schemes that have been developed previously for determining the level of security provided by information systems. Section 5 discusses implementations of data integrity models and information systems that have been developed to provide data integrity assurance. Section 6 summarises the results that became evident through the review.

2 Biba Data Integrity Model

2.1 Overview

With the intention of developing a data integrity model to complement data confidentiality, the Biba integrity model [Biba 1977] evolved out of an evaluation of various approaches to Mandatory Access Control (MAC) and Discretionary Access Control (DAC). The approaches to MAC evaluated were the Low-Water Mark, Ring and Strict Integrity policies. The approaches to DAC were Access Control Lists (ACLs) and the Ring Integrity policy.

The *Low-Water Mark* policy is a dynamic policy in the sense that the integrity labels of subjects (processes and threads) change according to their behaviour (floating integrity labels). These integrity labels reflect the level/measure of integrity and reliability of the subject in a similar way that security clearance levels represent how trustworthy the subject is. While a subject's integrity level starts at the highest level, it will ultimately become the least integrity level (low-water mark) of all objects (data items) observed by the subject (a subject can read any object regardless of integrity level). Consequently, subject integrity levels are dynamic monotonically decreasing values. Subjects are only allowed to modify objects with integrity levels lower than or equal to their own (no write up). While assurances on data integrity are guaranteed under this scheme, it is susceptible to improper observe access, whereby a system is reduced to a state of low integrity through the system-wide reading of low integrity data.

The *Ring policy* differs to the Low-Water Mark policy in that it maintains fixed integrity levels for both subjects and objects. Subjects are only allowed to modify objects where their integrity level is greater or equal to that of the target object (no write up) and observation is unrestricted, that is, a subject may observe any object regardless of integrity levels. While the absence of decreasing integrity levels (low-water marks) results in the policy's increased flexibility, assurances on integrity are not guaranteed and programmatic subject-based checking is necessary. This is because a subject may read low integrity data and remain able to write it to (contaminate) data of higher integrity.

The *Strict Integrity policy* utilises fixed integrity labels for both subjects and objects. To provide assurances on data integrity however (unlike the Ring policy), the observation of objects with lower integrity levels is strictly forbidden. This approach lends itself to two axioms: the "simple integrity condition" (no read down) and the * (star) property (no write up). The "simple integrity condition" ensures that a higher integrity subject can not be tainted through the observation of lower integrity objects. Similarly, the * (star) property prevents the flow of low integrity data to high integrity objects through direct modification. A direct consequence of tightening observe access is that the policy becomes largely inflexible and less compatible with applications.

An *Access Control List* is directly associated with an object and it specifies a list of users whose subjects can access the object by outlining observe or modify permissions. ACLs are maintained for each object in the system and, given the discretionary nature of the mechanism, can be dynamically updated by any appropriately privileged subject.

The *Discretionary Ring policy* derives from the aforementioned MAC Ring policy with the primary difference pertaining to the introduced capacity for discretionary modification of access privileges.

With a strong focus on military/defence applications, the Biba model employs the Strict Integrity policy to enforce data integrity. The Biba model specifically addresses direct external threats (sabotage) to integrity that require access control enforcement mechanisms to be applied repeatedly, that is, not only at system startup. It has been architected to accommodate varying granularities of protected system components and is sympathetic to the pairing needs of an appropriate granularity of enforcement with the granularity of policy (e.g., a policy that protects access to parts of a file is paired with enforcement mechanisms that govern access to parts of the file, not the whole file only). Under the Biba model, the integrity problem space encompasses three classes of modification; national, user and application specific security.

The selection of the Strict Integrity policy is primarily based on its provision of the greatest protection against direct malicious modification (access is strictly prevented), the fact that its afforded protection is relatively easy to understand and its ability to prevent improper observe access (subject and object integrity levels are static and therefore cannot be exploited to reduce the system to a state of low integrity). A drawback to this scheme however, is that many objects are rendered inaccessible and that some environments may find its constraints too cumbersome for practical use.

2.2 Extensions

2.2.1 BIFI Biba-Invoke

A significant disadvantage of the Biba Strict policy is that trusted subjects become isolated from untrusted inputs. The problem is not satisfactorily overcome through the adoption of the Low-Water Mark policy as trusted subjects will have their integrity levels lowered, eventually resulting in all subjects reaching a common low integrity level (improper observe access). A mechanism introduced by Hu & Feng [2008] aims to manage the risk involved in allowing low-to-high integrity information flows associated with the Biba Ring policy.

An Integrity Agent (IA) mechanism is added to the model and all low-to-high information flows pass through the IA. The IA is a trusted process and all subjects requiring untrusted input will make requests through the IA. This provides a centralised mechanism for integrity upgrading and can be further used for audit or recovery purposes.

Hu & Feng implemented this model with extensions to SELinux and the IBM Integrity Measurement Architecture (IMA). The IA mechanism is provided through the addition of an IA_{type} to the typing system. By default, all processes are only allowed input from the Trusted Computing Base (TCB) or other trusted subjects. Any process requiring input from an untrusted subject can only do so through an explicit invoke-agent() system call. With regard to system performance, it was claimed that that there was an average 7.5% additional data access latency penalty incurred.

2.2.2 Strict Integrity Policy (SIP) with dynamic characteristics

The policy presented by Zhang [2009] represents a dynamic variant of the Strict Integrity Policy (SIP) under the Biba data integrity model. The presented policy claims to maintain

the strictness of the original Biba SIP while facilitating greater flexibility and compatibility with subjects/software.

To do so, the policy employs the use of separate reading and writing subject integrity level ranges in place of a single subject integrity level. Further, the lowest object integrity level read and the highest object integrity level written to by the subject to date is maintained. Whereas access to objects is governed by the "simple integrity condition" (no read down) and * (star) property (no write up), relative to a subject's integrity level under SIP, access control under the proposed scheme is arbitrated according to a subject's read/write ranges and its access history.

While the increased flexibility and compatibility of the proposed policy presents an advantage, its fundamental workings and correctness are confusing at worst and unclear as best. In conjunction with the overhead required to maintain subject histories and integrity levels, it is unclear as to whether the gain in dynamicity is worth the added complexity.

2.2.3 Bell-LaPadula and Biba dynamic union

Various models have been put forward over the years to address the concepts of confidentiality and integrity; the Bell LaPadula (BLP) and Biba models being the two most common models for addressing confidentiality and integrity respectively. Enforcing confidentiality and integrity requires opposing techniques. Consequently, the development of a system that addresses this duality issue is difficult.

As an attempt at combining BLP with Biba, Zhang, Yun & Zhou [2008] have put forward a dynamic union model with check domain that facilitates the monitoring and updating of subject/object security labels. Normally, incompatibilities between the BLP and Biba models prohibit direct access to objects by subjects (combining BLP and Biba both under strict policies would render the system largely unusable, that is, no read up/down no write up/down only read/write at the same levels or read/no write, write/no read). To overcome this, the authors make use of a check domain combined with the current security label of the accessing subject to facilitate the dynamic adjustment of the subject and checked object's security labels. Further, objects that have been read and written are signed by their accessing subjects to enable accident tracking. This approach realises indirect access that continues to satisfy appropriate conditions while assuring confidentiality and integrity.

Access arbitration and relevant dynamic security adjustments are governed by nine rules. The first three rules are basic security rules to accommodate simultaneous BLP and Biba while the remaining six are complementary security rules to accommodate when the security label of a subject and object cannot fit the BLP or Biba models. Specifically, when security labels do not fit, a new object is created in the check domain by a privileged subject. The confidentiality and integrity of the new object is set to that of the accessing subject which facilitates the subject's access to the newly created copy. Once modified by the accessing subject, the privileged subject then evaluates the confidentiality and integrity of the object. If there are no violations, the privileged subject changes its confidentiality and integrity levels to that of the original object and performs the modify operation

permanently. As mentioned above, a copy of the object is encrypted and signed by the accessing subject for accident tracking.

While the use of a check domain facilitates the union of BLP and Biba to provide both data security and integrity, the accuracy of the system is largely dependent on the correctness of the privileged subjects and check domain. This remains an ongoing research challenge.

3 Clark-Wilson Data Integrity Model

3.1 Overview

Clark & Wilson [1987] claimed that two policies must be enforced to maintain data integrity within a system. Firstly, it requires that data can only be manipulated in constrained ways using verified transformation procedures (TPs). Secondly, critical operations must be separated into sub-parts and executed by different users, thus enforcing the separation of duty principle. This principle preserves the external consistency of the data — the correspondence between the external world and the data in the system.

In contrast to the Biba model, the Clark-Wilson (CW) model has only two logical integrity levels. Data whose integrity has not yet, or cannot be, verified is defined as an Unconstrained Data Item (UDI). A Constrained Data Item (CDI), in contrast, is one whose integrity has been verified and can continue to be verified at any later stage using Integrity Verification Procedures (IVPs). These procedures must be certified as being able to correctly determine the integrity of the data they accept. This certification was claimed by Clark & Wilson to require formal verification, for example, through the analysis of the procedures' code.

The policies of the Clark-Wilson (CW) model are enforced by applying the following 9 rules:

Certification Rules:

- C-1: IVPs must be certified as being able to correctly determine the integrity of a particular set of CDIs
- C-2: TPs must be certified as able to transform a particular set of CDIs from one valid state to another
- C-3: The list of $(User, TP, (CDI_1, CDI_2, ...))$ relations (from E-2) must be certified as maintaining separation of duty for critical processes
- C-4: All TPs must be certified to write to a log for auditing purposes
- C-5: Any TP taking a UDI as input must be certified to perform only valid transformations for any accepted input, or do nothing for input not accepted

Enforcement Rules:

- E-1: Only TPs certified for a given CDI may be executed
- E-2: (E-1 extension) Only executions described in a (User, TP, (CDIs)) relation are allowed
- E-3: Users must be authenticated before allowing TP executions
- E-4: Agents able to certify an entity may not execute that entity and are the only agents able to modify the list of associated users.

Rules C1, C2 and E1 together, ensure internal consistency and rules E2 and C3 enforce separation of duty, that is, external consistency. E4 ensures the preceding rules remain mandatory rather than discretionary.

Clark & Wilson claimed that their model improves on the Biba model in that a security officer or administrator is not needed for upgrading the integrity of an entity. Instead, the administration body certifies the TPs that are able to transform input UDIs to CDIs. With data input being the main function of many systems, this automation can result in a significant improvement in system performance.

A disadvantage of this model is that no mention was made of how the certification of IVPs and TPs should be performed beyond the possibility of it including formal code analysis that can be complex and costly. The model also places greater responsibility on the administration body for determining adequate separation of duty throughout the system, however this can be error prone. A further disadvantage is that rule C-2 does not require the integrity of a CDI to be maintained during the execution of a TP and therefore, a system can require sequential execution of its TPs which may degrade system performance.

3.2 Extensions

3.2.1 Integrating with existing MLS systems

Originally, the CW model was designed for commercial environments where integrity takes precedence over confidentiality. An extension of the CW model was developed by Hanigk [2009] in response to his claim that data confidentiality alone is not sufficient in a national security environment. Hanigk's objective was to extend the CW model and integrate it into an existing MLS environment, specifically within a management architecture for military mobile ad-hoc networks.

Hanigk claimed integrating the CW model directly into an MLS system would require complex and inefficient extended use of the $(User, TP, CDI)$ access triples. The alternative approach proposed was to replace the User and CDI entities with equivalence classes with respect to some given predicates. An example predicate that may define a User Entity Class is $(Clearance(User) >= SECRET)$. The original CW model can then be seen as a special case of this extended model where predicates select out single entities.

The rules of the CW model are rewritten to operate on these equivalence classes rather than groups of entities. For example, rule C-2* is that all TPs must transform a CDI *class* from one valid state to another.

To integrate the model into an existing MLS system, user entities are given clearance and integrity level attributes. Maintaining data confidentiality is possible because the TPs have a user entity class, defined by a security classification predicate, associated with them. Furthermore, the MLS requirements impose an additional invariant upon the CW model by requiring that the security classification of a CDI-class is never lowered by a TP.

Hanigk acknowledges that the complexity of the lookup algorithms for access triples grows with the generality of the entity attributes and transformation invariants. This is because total ordering of the structures is not always possible. The model has, however, been successfully implemented in a management system for a military mobile ad-hoc network.

3.2.2 Integrating with existing database systems

One of the major disadvantages to the CW model is that it can be difficult to implement. Ge, Polack & Laleau [2004] however, claim this implementation difficulty is less relevant when applying the CW model to a Database Management System (DBMS). The existing security frameworks in conventional DBMSs provide sufficient support for the implementation of several aspects of the CW model. For example, built-in DBMS entity and referential integrity constraints as well as other static constraints and triggers can be used to enforce the integrity of CDIs that are accessed by TPs.

The required CW access-control triples cannot, however, be fully implemented for user transactions using only SQL mechanisms — support from the operating system is also required. The approach taken uses stored procedures as the CW TPs and grants users EXECUTE access to them. In this way the access triples defined by CW can be enforced and CDIs will only be accessible through the TPs. Ensuring the separation of duty principle is upheld is still an issue that requires future investigation. The integrity constraints on the data items are also context specific and thus might not suit more generalised systems where the semantic integrity of the data items cannot easily be deduced.

4 Evaluation

4.1 Trusted Computer System Evaluation Criteria

The Trusted Computer System Evaluation Criteria (TCSEC) [*Trusted Computer System Evaluation Criteria* 1985] — also referred to as "the Orange Book" — was one of the original criteria developed by the U.S. Department of Defense to address computer system security, and in particular, MLS system security. The main focus of this criteria was, however, data confidentiality assurance and not of data integrity. Furthermore, it was very expensive to evaluate a product against this criteria which led to few commercial products pursuing accreditation. It also placed strong emphasis on the functional requirements of MLS systems and did not generalise to many other types of security products.

The TCSEC was later partly superseded by the Information Technology Security Evaluation Criteria (ITSEC) [*Information Technology Security Evaluation Criteria* 1991],

which was designed to address some of the shortcomings of the TCSEC. In particular, the ITSEC allowed specifying the "security target" of the evaluation. Thus evaluation and certification of only certain security aspects of the product became an option for companies.

4.2 Common Criteria

Further improvements to the ITSEC and TCSEC were made with the introduction of the Common Criteria (CC) in 2000 [*Common Criteria* 2000]. Importantly, data integrity and availability became possible targets for evaluation with the CC, which was one aspect of the TCSEC that was determined to be lacking. Evaluation against the CC provides recognised evaluation in Australia and many other countries. The CC defines 7 levels of assurance, from EAL-1 to EAL-7 with EAL-7 being of the highest assurance. One criticism of this criteria is that the assurance level alone does not provide any indication of what security mechanisms or functions the product was evaluated for. In contrast to this, the TCSEC was clear on the functional assurance provided by its various evaluation levels.

4.3 Director of Central Intelligence Directives

The U.S. Director of Central Intelligence Directives (DCIDs) [*Director of Central Intelligence Directives* 1999] are a series of documents pertaining to the U.S. intelligence community-wide policies. Since 2005, many of these documents have been superseded by the newer U.S. Intelligence Community Directives (ICDs) and in particular, DCID 6/3 will be replaced by ICD 503 [*Intelligence Community Directives* 2008]. The DCID 6/3 document outlines policy guidlines for ensuring the data integrity and availability as well as confidentiality within information systems. This is in contrast with the TCSEC, for example, which concerned itself primarily with data confidentiality in MLS systems.

5 Implementations

5.1 Minos

An example of a specialised use of the Biba Low-Water Mark policy is the work done by Crandall & Chong [2004]. The Biba Low-Water Mark policy was enforced for the prevention of control flow attacks such as buffer overflow and code injection attacks. The motivation was that processes should not allow their control flow to be altered by untrusted input data. Any situation where this does occur can be indicative of an attack. Such control flow modifications can occur through manipulation of function return addresses and pointers, addresses of libraries and any data that will be loaded into the program counter. The model thus provides an approach to managing the risk of introducing data into a system that can corrupt running processes.

The enforcement of the Low-Water Mark policy requires modifications to the hardware and the operating system kernel. Each addressable word in memory and each process is

augmented with an integrity bit (0 being low and 1 being high). The calculation of a subject's new integrity level (low-water mark) follows from a single bitwise AND operation on the integrity bits of the interacting subject and object to select their minimum.

The definition of a trusted object used by Crandall & Chong was based on the time the object was introduced into the system. An establishment-time can be defined at a particular moment after the system is fully configured and any object that is introduced after this point in time will not be considered trusted. The authors note that this approach to defining trust was used primarily for the testing of the prototype and that more practical approaches can be devised. An example would be the provision of authentication and verification procedures for upgrading the integrity of certain objects.

The mechanism used to manage access to objects is primarily through the modified system calls such as read() which handle disk, network socket access and inter-process communication. Data read through these system calls will be forced to the low integrity level and hence if a process's control flow depends on this data, then its integrity level is also lowered.

The results of their work showed success in preventing many control flow attacks. It is likely to be only a partial solution to ensuring general data integrity throughout a system. One reason for this is that only two integrity levels are defined which may not be adequate for national security or military operational environments. Defining finer granularity would require more than double the hardware extensions. Furthermore, it requires that all newly introduced data be interpreted as having low integrity. There are situations where this is not practical and to address these situations integrity upgrading procedures would still need to be implemented.

5.2 TrustedBSD

The TrustedBSD project [Watson et al. 2003] provides a MAC Framework to extend Linux based operating systems, specifically open source FreeBSD, with modular and flexible access control modules. In addition to a Multi-Level Security (MLS) module, the MAC Framework provides Biba and Low-Water Mark Access Control (LOMAC) integrity Loadable Kernel Modules (LKMs) that can be loaded into the target operating system's kernel at compile-time, boot-time or run-time. Registration flags are maintained to indicate that they must be loaded prior to kernel object allocation and that they must not be removed once attached. In utilising LKMs, different confidentiality/integrity policies can be loaded without requiring modification to the kernel itself.

The Biba LKM [Watson et al. 2003] is an implementation of the fixed-label Biba Strict Integrity policy; "simple integrity condition" (no read down) and the * (star) property (no write up). The policy is both hierarchical and compartmental and employs centralised logic for access control arbitration. Ubiquitous labelling of all subjects and objects within the system facilitates access evaluation using a dominance operator (dominate function compares elements with respect to their types, grades and compartments) over pairs of labels.

The LOMAC LKM [Fraser 2001] is an implementation of the Biba Low-Water Mark policy. As mentioned above, the module can be loaded into Linux kernels during bootstrap

to enforce system-wide security/integrity protection against malicious users and code. To facilitate access arbitration, LOMAC utilises interposition to intercept security-relevant system calls and maintains its own data structures/security attributes to assist in the process.

In consequence of maintaining data structures and other security attributes, LKMs are larger and more complex than comparable reference monitor approaches and are therefore more difficult to verify with formal methods. Other drawbacks to LKMs are their vulnerability to tampering from the kernel itself and other LKMs, and their limited capacity for enforcing the principle of least privilege. The LOMAC LKM is also susceptible to the phenomenon of improper observe access associated with the Low-Water Mark policy. Advantages of LKMs are their applicability to standard Linux kernels, compatibility with existing applications, they require no site-specific configuration and are largely invisible to users.

5.3 XTS-400

Another example of a general purpose operating system incorporating both the Bell-LaPadula confidentiality and Biba integrity models is the XTS-400 Trusted Computer System by BAE-Systems [2005]. It provides a Linux-like interface for the developer such that untrusted applications designed for other Linux systems can easily be run on the XTS-400 without compromising the security and integrity of the rest of the system. The operating system was successfully evaluated to the level of EAL5 according to the Common Critera (CC) [*NIAP CCEVS - Validated Product: XTS-400* 2005]. Furthermore, its predecessor, the XTS-300, was one of the first general-purpose operating systems to be successfully evaluated to the Trusted Computer System Evaluation Criteria (TCSEC) B3 class. Criticisms of the system include its binding to specific hardware and consequently unsuitability for embedded systems, its interfaces, such as the commandline tools, being less usable than other systems, and the performance loss due to additional security checks.

5.4 RS-Linux

RS-Linux is a research experiment aimed at implementing a secure operating system that enforces Mandatory Integrity Protection (MIP) [Liang & Sun 2001]. Enforcement in RS-Linux is carried out through a kernel component called RS-MIP. To achieve integrity protection, RS-MIP adopts the Biba Low-Water Mark (LWM) model; primarily because of the increased usability associated with dynamic adjustment of subject privileges.

Central to the implementation of LWM in the RS-Linux kernel is its Security Architecture (RSSA). The RSSA extends the ISO Access Control Framework to facilitate support for multiple security policies. A Security Policy Enforcement Function (SPEF) operates as a reference monitor, forwarding invocations of extended security-relevant system calls to a Security Policy Decision Function (SPDF). Subject access is determined by the SPDF through the application of three decision rules against potentially different security policies. The three decision rules are as follows:

- Integrity read decision rule ($IRDR$) => verified by LWM invoke/observe axioms

- Integrity write decision rule ($IWDR$) => verified by LWM modify axiom
- Integrity read & write decision rule ($IRWDR$) => verified by IRDR & IWDR

Under the implementation, LWM subjects are mapped to RS-Linux users and processes while objects are mapped to files/directories/devices/basic system data/IPC. Only binary integrity levels of 0 (lowest) and 1 (highest) are employed. To assist in evaluating access, integrity levels are assigned to every subject and object in the system.

Liang & Sun claim that RS-MIP prevents unauthorised users from making malicious modifications. They also claim that it partially maintains internal and external consistency, that is, the self-consistency of interdependent data and the consistency of real-world environment data. The prevention of authorised users from making improper modifications is not addressed under the implementation. Liang & Sun claim the disadvantages of RS-MIP are twofold. First is that given its self-focus on access control, it needs to combine the roles mechanism to satisfy the principle of least privilege. Second, overall integrity results are still dependent on trusted subjects not exploiting covert channels to bypass access protection mechanisms. Furthermore, the use of binary integrity levels may not be sufficient in national security environments.

5.5 Trusted IRIX CMW

Trusted IRIX CMW (Compartmented Mode Workstation) [SGI 2002] is a security enhanced feature set designed to integrate with the commercial SGI IRIX operating system to create a trusted computing environment. Central to Trusted IRIX is an MLS framework for the enforcement of fully configurable MLS policies and the assertion of accountability and assurance. Of particular interest is the incorporated capacity for both data confidentiality and data integrity. It should be noted, however, that there is no mention of which model or policies were implemented to provide the data integrity assurances.

According to security labels (sensitivity/confidentiality and integrity), the framework enforces policy-governed access control through the segregation of user interactions. A sensitivity label (composed of a level and category) outlines the classification of data or the clearance level of a user whereas an integrity label (composed of a grade and division) stipulates data reliability. In conjunction with a loaded MLS policy, security labels are used to arbitrate user access to resources. A system audit trail provides system administrators with detailed activity logs which are essential for overseeing events, tracking changes to sensitive data and for identifying malicious use [SGI 2002].

Trusted IRIX has been developed to meet the following evaluation schemes/assurance levels:

- Labelled Security Protection Profile (LSPP) System Assurance Level
- Controlled Access Protection Profile (CAPP) System Assurance Level
- Common Criteria for IT Security Evaluation (ISO Standard 15408)
- Trusted Computer Security Evaluation Criteria (TCSEC - Orange Book) B1 Level

Through the labelling of imported/exported data, Trusted IRIX also supports trusted networking. Data labelling takes place at the network level (IP) and also at the session manager level. To facilitate controlled data exchange between trusted operating systems, the Trusted Security Information Exchange (TSIX) was derived. Further, the Security Attribute Modulation Protocol (SAMP) and Security Attribute Token Mapping Protocol (SATMP) are used to share security attributes between nodes. To prevent the compromise of data, all transactions on remote nodes are restricted to execute under the label of the local node.

6 Summary

This review has identified that there are several compromises that must be made during the selection or development of data integrity models for information systems. One of the earliest integrity models proposed in the literature, the Biba Strict Integrity model, for example, provides a high level of protection and simplicity in its design. The disadvantage, however, is that the usability of the system is restricted, even more so when integrated into a system with confidentiality policies in place. Extensions to the original Biba models were developed to improve their usability, such as the BIFI Biba-Invoke and the Biba-Bell-LaPadula Dynamic Union models. These were shown to improve the usability, however they were also more complex and could therefore be more difficult to understand and costly to formally verify their correctness.

The Clark-Wilson model was proposed as an alternative to the Biba model and its constituent policies with the objective of improving the usability of the system without reducing the level of integrity protection afforded. This model and its extensions shared some characteristics with the Biba model extensions; in particular, while the usability was improved, the simplicity of the system was significantly reduced. The model was found to be less applicable to general purpose systems and the complexity of developing an implementation highlights this. The model does, however, share similarities with database management system (DBMS) models and this was shown to simplify the process of developing a DBMS implementation.

Another compromise that is difficult to quantify is the associated performance cost with some of the models. It was noted, for example, that the Clark-Wilson model imposed certain restrictions on the system, such as requiring processes to be executed sequentially, which can greatly reduce the performance. This is in contrast to the Biba models where there is generally negligible performance overhead incurred with its introduction into a system. In general, all models reviewed require the maintenance of additional data structures and security-attributes. Consequently, this also impacts on overall system performance.

The integrity of data maintained by a system can be no greater than that of the system's lowest-integrity component, including the source of input [Irvine & Levin 2001]. Some models, such as the Clark-Wilson model attempt to address this by providing verification procedures to verify the integrity of data at any instance. Further, it has been identified that assurances on integrity rely on the propriety of externally assigned integrity levels to both subjects and objects. Ensuring the correctness of these assigned labels is a difficult task that the models reviewed are unable to address.

With regard to integrity metrics, Table 1 provides a qualitative assessment of the models and related extensions reviewed in this paper according to integrity protection afforded, the overall usability of the system, how simple the model is to understand and implement, and whether data confidentiality is explicitly addressed. It should be noted that no quantitative metrics for the assessment of data integrity were identified in the review, consequently this remains an area for future research.

Model	**Protection**	**Usability**	**Simplicity**	**Confidentiality**
Biba Strict	High	Low	High	No
Biba Low-Water Mark	High	Low	High	No
Biba Ring	Low	High	High	No
BIFI Biba-Invoke	High	Medium	Medium	No
Strict Integrity Policy	High	Medium	Low	No
BLP and Biba Union	High	Medium	Low	Yes
Clark-Wilson	High	Medium	Low	No
Clark-Wilson - MLS	High	Medium	Low	Yes
Clark-Wilson - DBMS	High	Medium	Medium	No

Table 1: *Comparison of data integrity models*

Acknowledgements

The authors would like to thank Paul Montague for his input and the supervision provided. They would also like to thank Duncan Grove and Damian Marriott for their feedback.

References

BAE-Systems (2005) Xts-400 trusted computer system. URL – http://www.baesystems.com/ProductsServices/bae_prod_csit_xts400.html.

Biba, K. J. (1977) *Integrity Considerations for Secure Computer Systems*, Technical report, MITRE Corp.

Clark, D. D. & Wilson, D. R. (1987) A comparison of commercial and military computer security policies, *in IEEE Symposium of Security and Privacy*, pp. 184–194.

Common Criteria (2000) URL – http://www.commoncriteriaportal.org.

Crandall, J. & Chong, F. (2004) Minos: Control data attack prevention orthogonal to memory model, pp. 221 – 232.

Director of Central Intelligence Directives (1999) URL – http://www.fas.org/irp/offdocs/dcid.htm.

Fraser, T. (2001) Lomac: Mac you can live with, *in USENIX Annual Technical Conference, FREENIX Track*, pp. 1–13.

Ge, X., Polack, F. & Laleau, R. (2004) Secure databases: an analysis of clark-wilson model in a database environment, *in Advanced Information Systems Engineering - 16th International Conference, CAiSE 2004*, pp. 7–11.

Hanigk, S. (2009) Confidentiality is not enough: A multi-level clark-wilson model for network management, *in MilCIS 2009*.

Hu, H. & Feng, D. (2008) Bifi: Architectural support for information flow integrity measurement, Vol. 3, pp. 605 –609.

Information Technology Security Evaluation Criteria (1991) URL – http://www.ssi.gouv.fr/site_documents/ITSEC/ITSEC-uk.pdf.

Intelligence Community Directives (2008) URL – www.dni.gov/electronic_reading_room/ICD_503.pdf.

Irvine, C. E. & Levin, T. E. (2001) Data integrity limitations in highly secure systems.

Liang, H. & Sun, Y. (2001) Enforcing mandatory integrity protection in operating system, pp. 435 –440.

NIAP CCEVS - Validated Product: XTS-400 (2005) URL – http://www.niap-ccevs.org/st/vid3012/.

SGI (2002) Multilevel security (mls) by trusted irix. URL – www.sgi.com/pdfs/3241.pdf.

Trusted Computer System Evaluation Criteria (1985) URL – http://www.boran.com/security/tcsec.html.

Watson, R., Feldman, B., Migus, A. & Vance, C. (2003) Design and implementation of the trustedbsd mac framework, *DARPA Information Survivability Conference and Exposition,* **1**, 38.

Zellmer, D. (2003) Multi-level security: Reality or myth.

Zhang, J., Yun, L.-J. & Zhou, Z. (2008) Research of blp and biba dynamic union model based on check domain, Vol. 7, pp. 3679 –3683.

Zhang, M. (2009) Strict integrity policy of biba model with dynamic characteristics and its correctness, Vol. 1, pp. 521 –525.

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION DOCUMENT CONTROL DATA	1. CAVEAT/PRIVACY MARKING

2. TITLE	3. SECURITY CLASSIFICATION
Review of Data Integrity Models in Multi-Level Security Environments	Document (U) Title (U) Abstract (U)
4. AUTHORS	5. CORPORATE AUTHOR
Patrick Garnaut and Jonathan Thompson	Defence Science and Technology Organisation PO Box 1500 Edinburgh, South Australia 5111, Australia

6a. DSTO NUMBER	6b. AR NUMBER	6c. TYPE OF REPORT	7. DOCUMENT DATE
DSTO–TN–0971		Technical Note	February, 2011

8. FILE NUMBER	9. TASK NUMBER	10. TASK SPONSOR	11. No. OF PAGES	12. No. OF REFS
2010/1112431/1	07/015	ISG	15	21

13. URL OF ELECTRONIC VERSION	14. RELEASE AUTHORITY
`http://www.dsto.defence.gov.au/publications/scientific.php`	Chief, Command, Control, Communications and Intelligence Division

15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT

Approved for Public Release

OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE, PO BOX 1500, EDINBURGH, SOUTH AUSTRALIA 5111

16. DELIBERATE ANNOUNCEMENT

No Limitations

17. CITATION IN OTHER DOCUMENTS

No Limitations

18. DSTO RESEARCH LIBRARY THESAURUS

Data Integrity

19. ABSTRACT

As there is an increased reliance upon information in defence operations and in network centric warfare, ensuring the security of the information systems involved is becoming an increasingly important objective. Within the realm of national security, research and development has predominantly focused on ensuring the confidentiality of data within these systems. There has, however, been recent recognition of the role that integrity has in the overall security of a system. To facilitate a better understanding of data integrity as it relates to information security, this paper provides a review of the associated literature including the prevalent data integrity models, evaluation mechanisms and integrity centric implementations.