

Strong Separation, Virtual Machines, and Private Virtual Networks

by
Edward A. Feustel, Institute for Defense Analyses (IDA)

This position paper describes ongoing work at IDA, concentrating on implementation of key concepts from the Department of Defense (DoD) Goal Security Architecture (DGSA) [DGSA96, SFR97]. This architecture requires strong separation of *information domains*¹ (IDs) and assumes no *a priori* relational structure of the sensitivity of the domains. Transfer of information between two domains is limited to those whose policies permit that transfer and are constrained to occur within platforms that host both IDs. Further, information domains may be hosted simultaneously on multiple platforms that may utilize heterogeneous hardware and software.

From the standpoint of the user, an information domain is like a space in JavaSpacesTM²: it exists as a virtual enclave that protects the information according to the security policy of the ID. Every IO within an ID must have the same set of policy-specified security attributes and the same security attribute values. Users within an ID must have the same set of policy specified security attributes, but may have different security attribute values. Different authorized users may have different privileges for all objects in a domain based on their security attribute values, e.g., specified roles, and a policy of authorization based on role.

In practical systems, two kinds of strong domain separation are required: strong separation of information objects while they are in transit from one platform to another and strong separation while they persist or are in use on a given platform. Our attention has been focused on the use of Private Virtual Networks³ based on the Internet Security Protocol (IPSEC) as a mechanism for separation while an IO is transiting the network and on Virtual Machines (VMs) [PG74, GOLD73] as one of the mechanism to be used for strong separation of IOs on platforms while in-use and in-storage [See also AND99].

One key to successful separation is the elimination of covert channels that convey information from one domain to another in contravention of policy [LAM73]. One way that such infor-

-
- ¹ Informally, an information domain contains an ordered set of information objects used by a set of users whose security attributes determine the evaluation and effect of the security policy governing the use of the objects.
 - ² JavaSpaces and Java Virtual Machine are trademarks of Sun Microsystems Incorporated.
 - ³ A private virtual network features N to N multicast connectivity as a virtual network implemented on one or more physical interconnected networks.

mation transfer could occur is via a shared resource, e.g., a file name in a shared directory. A second way is via an observation of the use of resources of one domain by a user in another, e.g., observation of the times that a specified event happens. We are studying methods of reducing the channel bandwidth by means of reducing the number of shared resources and by reducing the observability of timed events, e.g., by providing virtual time or virtual cycle counts.

Typically we think of security policies whose enforcement mechanism involves access control and reference monitoring. The DGSA takes a broader view of security policy including all aspects of confidentiality, integrity, availability, and accountability. Our investigation must deal appropriately with the separation required for the implementation of these additional policy aspects. For example, integrity constraints may require that implementation of all input-output (I-O) must be fully mediated and this mediation must not be bypassed. One way to achieve this is to separate all I-O mechanisms from the user and require “gated-entry” to all mechanisms performing I-O. VMs such as the Java Virtual Machine™ (JVM) can enforce mediation on all I/O by inclusion of all I/O primitives as part of the basic security classes of the runtime virtual machine.

We are studying mechanisms that might make this non-bypassable mediation possible when the virtual machine permits “assembly language level access” to all instructions and data of the user visible machine, *trapping* as required to assure non-bypassable mediation. If this can be done, users could run popular operating system software in separated “containment VMs” with assurance that they could only receive and transmit material in accordance with the security policy associated with their ID.

- [AND99] James P. Anderson. 1999. *VPNs of VMs for Secure Information Domains Architectures*. Workshop on Domain Architectures, Institute for Defense Analyses, Alexandria, VA. August 1999. (Paper revised July 22, 1999).
- [DGSA96] Defense Information Systems Agency, Center for Standards. 1996. *Department of Defense (DoD) Goal Security Architecture (DGSA), Version 3.0*. Volume 6 of [TAFIM]. http://www-library.itsi.disa.mil/tafim/tafim3.0/pages/tafim_6.htm.
- [GOLD73] R. P. Goldberg. 1973. *Architecture of Virtual Machines*. Proceedings of AFIPS National Computer Conference. AFIPS, New York, NY. June, 1973. pp. 74-112.
- [PG74] Gerald J. Popek and Robert P. Goldberg. 1974. *Formal Requirements for Virtualizable Third Generation Architectures*. Communications of the ACM, Volume 17, No. 7. Association for Computing Machinery, New York, N.Y. pp. 412-421. July 1974.
- [LAM73] Butler Lampson. 1973. *A Note on the Confinement Problem*. Communications of the ACM, Volume 16, No. 10. Association for Computing Machinery, New York, NY. pp. 613-615. October 1973.
- [SFR97] Edward A. Schneider, Edward A. Feustel, and Ron Ross. 1997. *Assessing DoD Goal Security Architecture (DGSA) Support in Commercially Available Operating Systems and Hardware Platforms*. IDA Paper P-3375. Institute for Defense Analyses, Alexandria VA. 22311-1772. November 1997.

Edward Feustel
Adjunct Staff Member
Institute for Defense Analyses

© 2000 Institute for Defense Analysis, Alexandria VA 22311-1772

Abstract

This paper describes the implementation of key concepts from the Department of Defense Goal Security Architecture (DGSA). This architecture requires strong separation of information domains and assumes no a priori relational structure of the sensitivity of the domains. Transfer of information between two domains is limited to those whose policies permit that transfer and are constrained to occur within platforms that host both information domains. Further, information domains may be hosted simultaneously on multiple platforms that may utilize heterogeneous hardware and software.

Virtual Machines (VMs) are used to separate domains on a given platform. Virtual Private Networks (VPNs) are used to separate domains when information is transmitted from platform to platform. Key requirements for the VMs and VPNs to be used will be related to those of the DGSA.

© 2000 Institute for Defense Analysis, Alexandria VA 22311-1772

DGSA Requirements

- Requires Information Domains with common:
 - Security Policy (confidentiality, integrity, availability, accountability)
 - Set of Principals (users) with varying level of privilege
 - Set of uniquely identified Information Objects which for a given principal are all treated identically

© 2000 Institute for Defense Analysis, Alexandria VA 22311-1772

DGSA Characteristics

- DGSA assumes:
 - use of an available common carrier that offers no additional protections
 - Information Domains distributed across heterogeneous platforms possibly using different mechanisms to enforce security
 - No weakest link -- security has a measure no less than Z anywhere the Information Domain is implemented

© 2000 Institute for Defense Analysis, Alexandria VA 22311-1772

DGSA Characteristics (2)

- Information Objects reside on one host at a time (accountability)
- DGSA does not assume that the sensitivity of information in one domain bears any relationship to that of any other: IDs are strictly isolated unless policy permits.
- Information exchange between IDs are governed by inter-ID policy

© 2000 Institute for Defense Analysis, Alexandria VA 22311-1772

DGSA Characteristics (3)

- DGSA is specified in a very abstract way that elides almost all implementation details
- DGSA does not provide any guidance about implementation
- The implementer must show that his implementation preserves the DGSA properties

© 2000 Institute for Defense Analysis, Alexandria VA 22311-1772

Implementation

- Strict separation of Information Domains
 - On each platform
 - When participating in a computation
 - When residing in storage
 - In the network
 - When an Information Object is being transferred from one host to another

© 2000 Institute for Defense Analysis, Alexandria VA 22311-1772