

CONF-971034--1

**MULTILEVEL ARCHITECTURES FOR  
ELECTRONIC DOCUMENT RETRIEVAL**

James A. Rome  
Mathematical Sciences Section  
Computer Science and Mathematics Division  
Oak Ridge National Laboratory  
Oak Ridge, Tennessee 37831

**RECEIVED**  
**APR 10 1997**  
**OSTI**

and

Johnny S. Tolliver  
Computational Physics and Engineering Division  
Oak Ridge National Laboratory  
Oak Ridge, Tennessee 37831

**MASTER**

"This submitted manuscript has been authored by a contractor of the U.S. Government under Contract No. DE-AC05-96OR22464. Accordingly, the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes."

---

\*Research supported by the Applied Mathematical Sciences Research Program of the Office of Energy Research, U.S. Department of Energy, under contract DE-AC05-96OR22464 with Lockheed Martin Energy Research Corporation.

**Multilevel Architectures for  
Electronic Document Retrieval**

James A. Rome  
jar@ornl.gov  
Johnny S. Tolliver  
jxt@ornl.gov

Oak Ridge National Laboratory  
P.O. Box 2008, Oak Ridge, TN 37831

**DISCLAIMER**

**Portions of this document may be illegible  
in electronic image products. Images are  
produced from the best available original  
document.**

# Multilevel Architectures for Electronic Document Retrieval

## Abstract

Traditionally, most classified computer systems run at the highest level of any of the data on the system, and all users must be cleared to this security level. This architecture precludes the use of low-level (pay and clearance) personnel for such tasks as data entry, and makes sharing data with other entities difficult. The government is trying to solve this problem by the introduction of multilevel-secure (MLS) computer systems. In addition, wherever possible, there is pressure to use commercial off-the-shelf software (COTS) to improve reliability, and to reduce purchase and maintenance costs.

This paper presents two architectures for an MLS electronic document retrieval system using COTS products. Although we believe that the resulting systems represent a real advance in usability, scalability, and scope, the disconnect between existing security rules and regulations and the rapidly-changing state of technology will make accreditation of such systems a challenge.

## Introduction

There exists a need for the electronic storage and retrieval of multilevel classified documents. In most existing systems, all documents are stored on a system that runs at syshi (the highest classification level of any document in the system), and all users must also be cleared to this high level. The U.S. government is under great pressure to reduce costs by reducing the number of high-level security clearances. As a result, there is a recognized need to move to an MLS computer architecture that would allow users with different security clearances to access the same system, but that would also protect all data against unauthorized disclosure.

One way to achieve MLS systems is to use compartmented mode workstations (CMW) to enforce security. Such systems have been evaluated at a *B1* level of trust in the "Orange Book," but also support parts of higher levels of trust. Where they are deficient is in the areas of covert channels and code assurance and testing. Moreover, according to the regulations, only two adjacent classifications and two compartments are allowed on a multilevel system; such a requirement makes it impossible to actualize a realistic security environment where many more need-to-know categories are required, and where four classification levels are desirable (Unclassified, Confidential, Secret, Top Secret).

The government is following the lead of other industries in the decision to use COTS products where ever possible. The use of COTS greatly reduces the cost of long-term maintenance associated with custom software products, and shifts the burden of user support to the software manufacturer. In most cases, this results in a better, more up-to-date product, greater user acceptance, and lower costs. However, the use of COTS in a MLS system poses numerous challenges.

The entire code may be a "black box," which probably needs to raise several system privileges in order to run. In such cases, these privileges need to remain in effect for the entire time that the program is running. How can we decide whether or not the program abuses this trust? License demons are especially bad in this respect because they might require read/write access to license files at any security level. Allowing this action requires an override of mandatory access control (MAC), the primary mechanism that exists to enforce security policies. Usually such programs (e.g., WordPerfect) can only be run at a single level without greatly compromising security.

Other programs come as a combination of proprietary code in black boxes, together with developer's hooks, and perhaps source code to allow customization of the high-level interfaces. In these cases, it may be possible to turn the program into a *trusted program*. This process invokes the security mechanisms of the trusted platform to raise privileges only when necessary (privilege bracketing). However, the granularity of the hooks into the black box code might not be fine enough to assure that privileges are not raised for some operations that take place within the black box code. In addition, such code modifications must be maintained whenever the system or the COTS product are updated, thus obviating some of the advantages of using COTS.

Finally, the regulations and procedures for evaluating the security of COTS or modified COTS are murky, and outdated.

## System environment

In spite of the above drawbacks, we have designed and implemented a MLS CMW-based electronic document retrieval system that uses modified COTS products. However, the environment for this system is rather special, and requires explanation. Presently, the system will be attached to a classified LAN on which all users are cleared to the highest clearance (Top Secret), but who may not have "need-to-know" requirements for all categories (or compartments in the CMW parlance). In addition, the users are all connected to the LAN by PCs

running Windows 3.1. The PCs all run at syshi (Top Secret plus all categories), but because Windows has no knowledge of the labeled packets used by CMW systems, the PCs are all regarded as "untrusted" by the CMW and come in with a single, fixed security label.

Because users share offices and PCs, and because the PCs are assigned dynamic ip addresses, it would be infeasible to assign a different label to each PC. Therefore, it makes sense to label all PCs either syshi or syslo. Because all the PCs are in fact at syshi, one could argue that they should all be assigned to syshi. However, in that case, if users were able to access a shell on the CMW through some security flaw, they would dominate any mandatory access control (MAC) label and have free run of the system. Accordingly, we assign all PCs a syslo label recognizing that it is rather arbitrary.

Thus, our system is truly multilevel, but its security assurances need only protect against access to unclassified compartments. However, in the future it is felt that such systems can be made strong enough so that they could provide access to users not cleared to the highest level. Future enhancements to these systems might include such things as Fortezza cards with strong encryption and authentication. By fielding this system now, all of the data will be properly labeled so that it can be the basis of a later (improved) system. Also, the development team have been careful to make a design that does not preclude the possibility of connecting the system to an Unclassified LAN.

Taking all of this into account, we feel that CMW systems have reached a stage of development where their advantages for such applications outweigh the bureaucratic hurdles that must be faced to gain accreditation.

### Functional Description

The Department of Energy Office of Safeguards and Security (OSS) Information Management Center (IMC) currently has about 700 linear feet of documents in storage, with an additional 25% in long-term storage. These documents range from Unclassified to Top Secret, and are subject to handling caveats and need-to-know restrictions on access. The Department is legally required to preserve these documents for varying periods of time.

There is a great need to be able to search these documents in an efficient and timely manner which implies that they need to be converted into an electronic format. Legally, this format must preserve the "look" of the original document. In addition to searches over the document text, it is also necessary to search over

keywords such as author, subject, date, security label, etc.

Initially the Electronic Document Center (EDC) shall be connected to a classified LAN on which all users have appropriate clearances, but not necessarily all need-to-know categories. Later, users cleared to a lower level will be able to access the LAN. Eventually, it would also be desirable to allow access to the unclassified portions of this database from the unclassified LAN.

Therefore, the challenge is to provide a secure but user-friendly method of accessing, searching, and retrieving documents from the EDC. For implementing these requirements, the architecture chosen employs the ubiquitous and familiar World Wide Web technology in an "intranet" environment coupled with a Compartmented Mode Workstation to enforce the multilevel security. Commercial off-the-shelf software is used whenever possible to reduce the amount of custom programming required.

### Single CMW Architecture

The major components of the system were determined by comparing the various storage media, formats and COTS products that were available. The major components and design decisions of the system are:

- ➡ Documents converted to Adobe Portable Document Format (PDF) to satisfy the legal requirement of "looking like the original," and to enable full text searches.
- ➡ Documents are stored on a read/write magneto-optical disk jukebox using a security-labeled file system.
- ➡ Indexes are stored on hard disks.
- ➡ The Excalibur Technologies RetrievalWare (RW) text retrieval database is used for search and retrieval.
- ➡ The Apache Web server would interface with the Web front end of RW.
- ➡ Adobe Capture would be used to convert scanned TIFF images to PDF files. Capture's optical character recognition capabilities (OCR) provide searchable text.

The challenge is to determine how to utilize these COTS solutions in a CMW venue to produce a solution that satisfies user needs while enforcing security. The overall architecture of the system is shown in Fig. 1.

The heart of The EDC is a Hewlett-Packard (HP) J-200 workstation running HPUX 10.16, a SecureWare-developed version of the CMW operating system.

### Security issues

#### Client level

Although two CMWs can exchange labeled packets across an Ethernet connection, the clients in our case are all PCs running Windows 3.1 and the Netscape Web browser. They can only communicate using unlabeled packets, and hence must be enrolled into the CMW's network security database (*M6RHDB* file) at a single, fixed security level. But which level should be used? It only makes sense to choose either the highest (*syshi*) or lowest (*syslo*) security level available in the system.

Because running at *syshi* would bypass most of the MAC security features of the system, the PCs are all assumed to run at *syslo* *even though this is not the case*.

#### Port access

It is possible to raise and lower the security levels of the ports of the CMW in accord with the clearance of the user, or to override the MAC of the port. We felt that it provided more resistance to attack from the network to keep the port MAC intact. Because of the stateless nature of the Web interactions, the multiple Web server demons that can be launched, and the complicated client-server database engine, it proved difficult to ensure that each Web request was served at the correct (user's) level so we decided to leave the ports at *syslo* at all times. To avoid the server problems, the user is

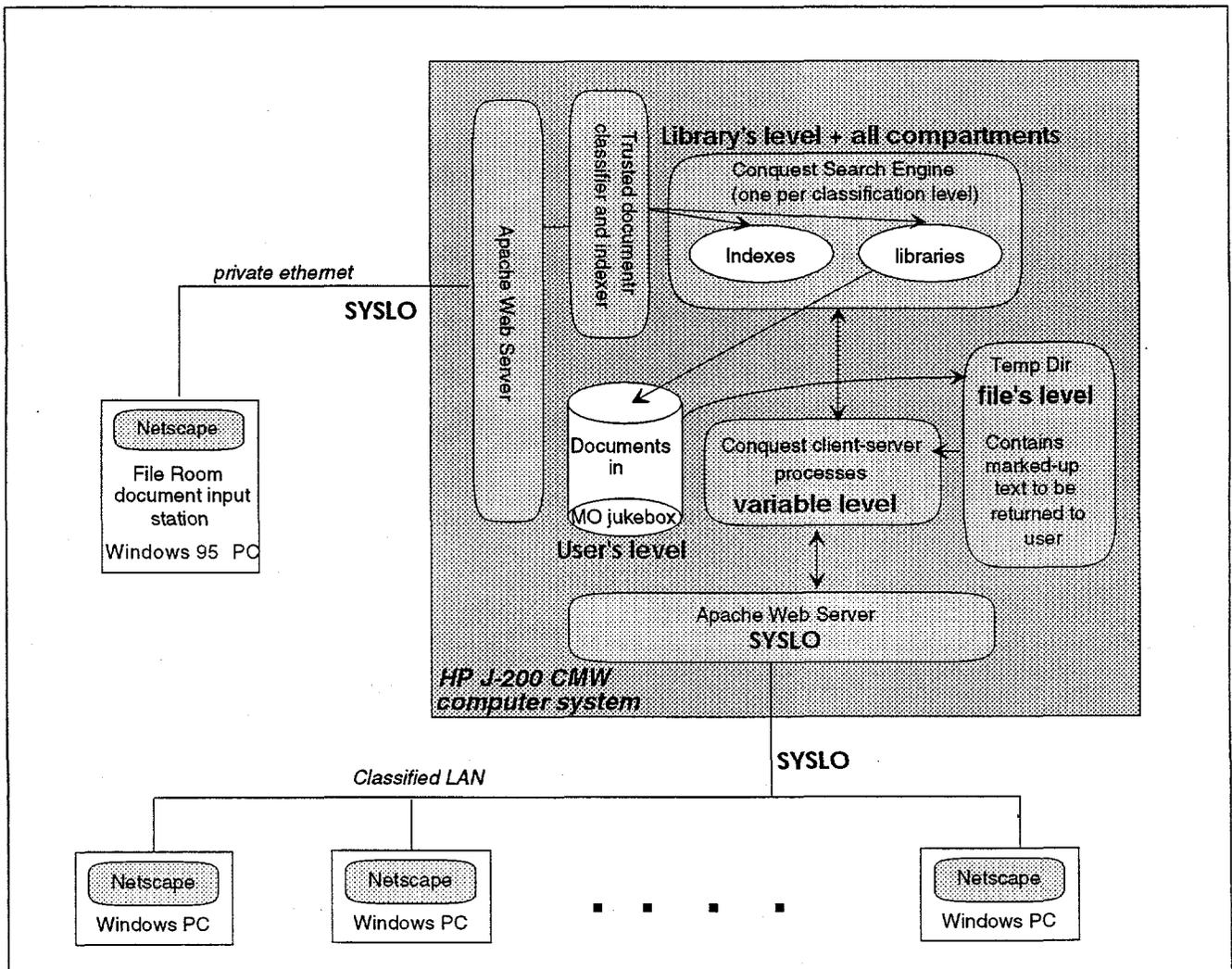


Fig. 1. Overall architecture of the CMW-based Electronic Document Center

reauthenticated for each Web request, and a new server is launched by the *inet* demon to service the request.

Setting all the ports and Windows clients to *syslo* solves the above problems but raises another. All documents are stored in labeled files with their correct security label (classification + compartments), and if they are to be served to the users at *syslo*, somewhere in the system, they will have to be "declassified" in order to send them through the *syslo* port. To perform this function, we utilized the fact that the CMW's trusted computing base (TCB) only checks MAC privileges when a file is opened. Therefore, we can obtain a file handle at one level and use it at another level. The file itself is not declassified. As is indicated in Fig. 1, the RetrievalWare client-server processes float their security levels up and down to be able to access the documents at their correct level and then to deliver them to the Apache server at *syslo*. Because the transaction only contains information dominated by the user's clearance, this level change does not cause a security risk provided that we make sure that the correct information is sent to the correct user. This bookkeeping is performed by the RW engine using a session key for each user. In addition, we reauthenticate the user for every transaction.

#### *Web server*

The classified data are organized into four separate libraries, one for each classification level. When the user logs into Apache, we use his user id (*uid*) and *ip* address to identify the user and to determine his security level. The Apache server's authentication routines were modified to use the CMW's TCB for authentication (rather than the *.htpasswd* file), and also to reauthenticate the user for each Web request.

#### *RetrievalWare security flow*

Knowing the user's clearance, we present an html page that only allows the user to select from those libraries dominated by his clearance. This could be overridden by a clever user, so we check the user's selections against his clearance in the RetrievalWare CGI driving routine. Even if this step is bypassed, the user will be unable to retrieve any document from a library not dominated by his clearance because MAC will prevent it when the document is opened with the user's clearance. The RW Web front end keeps track of the user, using the *uid*, *ip* address and the time of day to form a hashed session key. Because the *uid* is used to identify the user to the CMW, and hence determine his privileges, it is vital that at every stage of the query process, we keep track of the *uid*. Unfortunately, the early versions of RW did not transmit the *uid* to the back-end client/server (c/s)

processes. To overcome this deficiency, we tack the *uid* onto the end of the query string, and then recover it in the back-end query engine.

Because there are only four separate libraries, they must contain documents with all of the different need-to-know categories. Therefore, the indexes for each libraries must have a security tag which is the classification level plus all of the compartments available at that level. To search the mounted indexes, the RW query process must be raised to the user's classification level plus all of the compartments. Note that this requirement implies that although compartments can have a lower bound, they cannot have an upper bound. If they did, a label could not be formed that would dominate the compartment at a classification higher than the compartment's highest level.

The user is not allowed to see all of the documents in the mounted libraries unless he is a member of all the compartments. To prevent the user from seeing hits on any documents in unallowed compartments, the trusted code performs a Boolean hidden query over the compartments of the documents (stored as a searchable field). If the compartments of the document are not all contained in the user's compartment list, no hit is returned. Thus, the RW search engine is relied upon to enforce the inadvertent disclosure of the title of a document in an unallowed compartment.

However, MAC is used to ensure that the contents of a document in an unallowed compartment are not disclosed. When the user selects a document to be returned, the retrieval engine's security level is set equal to the user's level and the document is opened. This process fails unless the user's clearance dominates the document's clearance. Once the document handle is obtained, the process level is lowered to *syslo*, and the information can be sent out the *syslo* port to the user on the Classified LAN.

#### *Connecting to an Unclassified LAN*

Figure 2 shows how the scheme of Fig. 1 can be replicated to allow users on the Classified (C) LAN to query over documents contained on the Unclassified (U) LAN. Because RW supports remote libraries, the Unclassified Library can be moved to the U LAN. The only connection between the two LANs is via a private Ethernet link. The two trusted port monitors make sure that only RW queries can flow from the C LAN to the U LAN, and that only hit lists and retrieved documents can flow in the opposite direction. If the port monitors do their job properly (the main task for this project), the only chance for release of information to the U LAN is if a

user on the C LAN makes a "classified query." The bandwidth for this is quite low (the query string is limited in length), and in any event, the RW engine on the U LAN will dispose of the query string when the query is completed.

### Multiple Computer Architecture

To make the scheme of Fig. 1 work, both the Web server and the RW processes must be converted into trusted processes. The Web server, the authentication routine is localized and can be swapped in easily if the server is upgraded. However, more extensive modifications have to be performed on the RW code. Three of the RW servers undergo significant code changes. Many of these changes broke when the version of RW was upgraded from 5.0 to 5.2. These just are the problems that the use of COTS was supposed to avoid. Accordingly, we have devised a scheme that requires much less extensive code modification, but at the expense of using multiple computers. The system, shown in Fig. 3, is fronted by a CMW-based system for several reasons. In the first place, the CMW offers enhanced resistance to attack and functions as a very secure firewall. The outside world has access to the web server on one Ethernet port, and the inside (the database computers) is attached to a second Ethernet port via a smart hub (not shown).

Second, the software in the CMW system is modified to allow enforcement of the user's clearance level and his need-to-know categories. The user's classification level is used to determine which database servers are mounted to fulfill his search request. This adjudication is conceptually shown by the dashed red line in the data flow diagram. For example, if the user is cleared to SECRET, the TOP SECRET server is not mounted.

Need-to-know categories are enforced by the database system as in the previous scheme. The code on the CMW system is modified to create a hidden query over the category keyword field that goes along with each of the user's queries. The hidden query prevents a hit on any document that is labeled with categories not possessed by the user.

In this architecture, all back-end systems are single-level hosts running at syslo. The database and Web server running on the CMW system also run at syslo. Therefore, as compared to the pure CMW architecture, the security level of all processes running on the front-end machine are never changed (*principle of tranquillity*). MAC of classification is provided by mounting the correct databases, and MAC over categories is provided by the database engine via the hidden query. However, the extra step of assuming the user's security level to actually open the document is skipped. The RW code

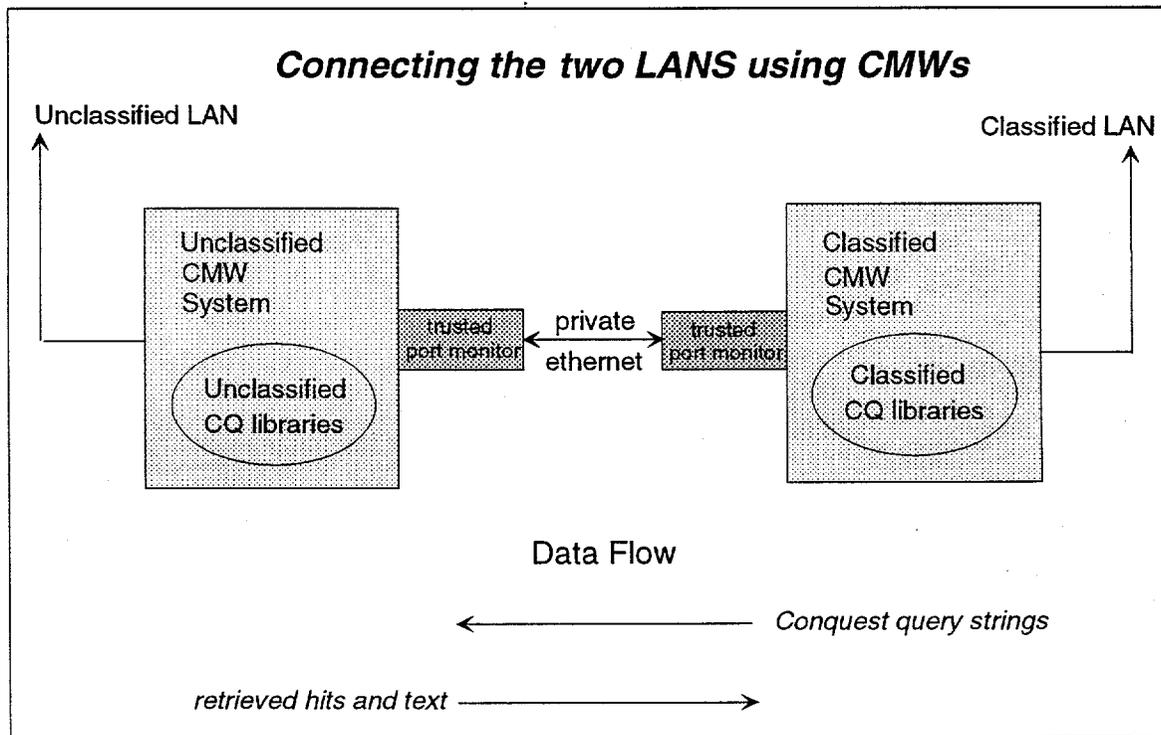


Fig. 2. Connecting the Classified and Unclassified LANs using an information monitor.

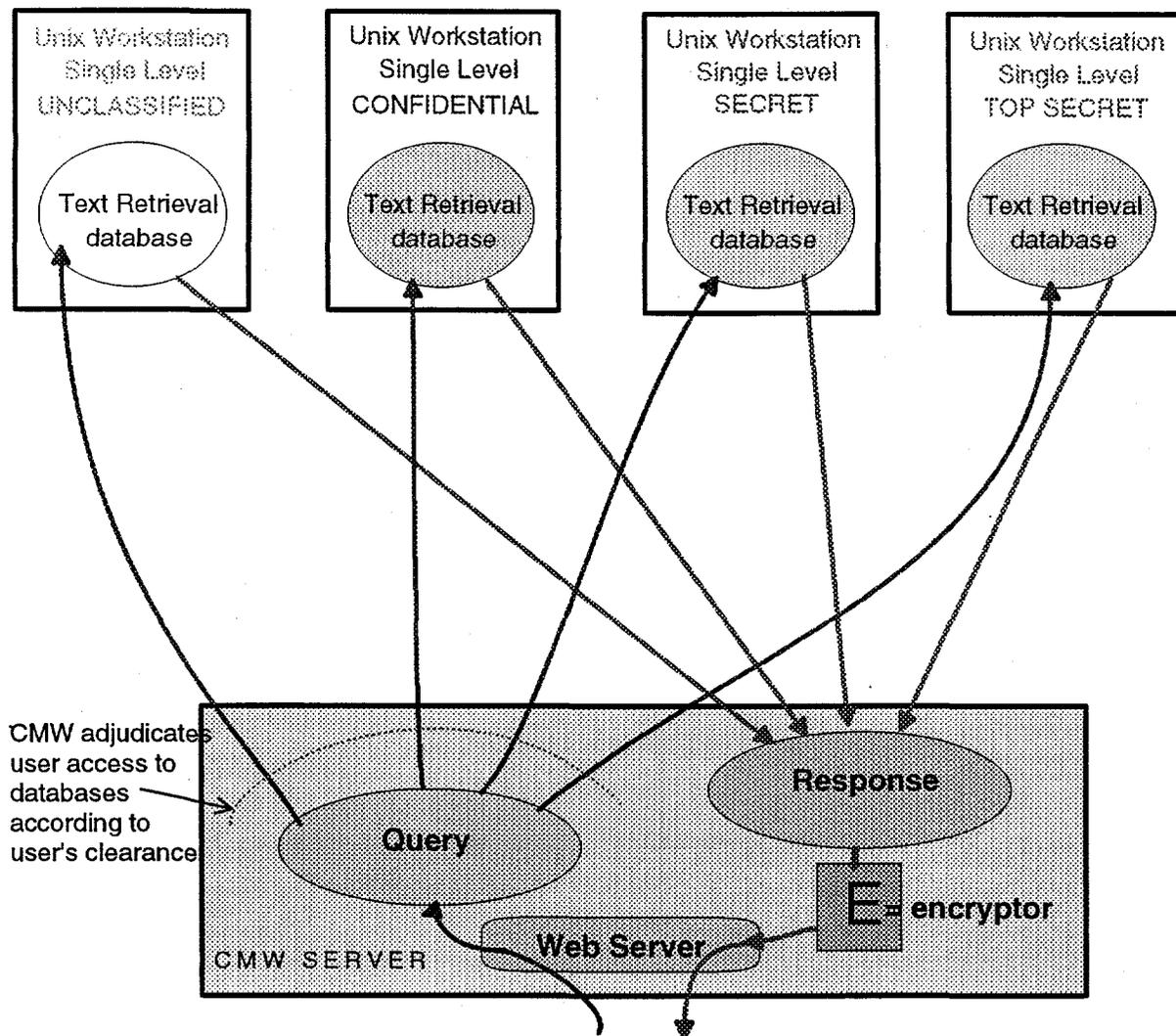


Fig. 3. CMW adjudicates the responses of four single-level workstations.

running on the back-end servers is unmodified COTS. The only RW modification needed on the CMW is the code to mount the correct databases and to perform the hidden query. Both of these changes are made in the RW Web front-end routine using standard developer's hooks which are not subject to upgrade changes.

If the external network has multilevel users, strong encryption can be used to protect the returned documents from the CMW to the user's PC. The encryption would probably be done by a future version of the Fortezza card.

### Summary and Conclusions

We have presented the architecture of an actual CMW-based multilevel document retrieval system to show the tradeoffs that must be made and the problems that must

be solved. Unfortunately, the implementation requires a significant amount (hundreds of lines) of code customization which is quite dependent upon the details of the version of the database engine. To avoid these code modifications, we have proposed a more distributed system that allows the use of unaltered COTS on the back end data storage systems.

The political and procedural hurdles that must be overcome to get these systems approved is the subject of future work.