LEGIBILITY NOTICE

A major purpose of the Technical Information Center is to provide the broadest dissemination possible of information contained in DOE's Research and Development Reports to business, industry, the academic community, and federal, state and local governments.

Although a small portion of this report is not reproducible, it is being made available to expedite the availability of information on the research discussed herein.

1

LA-UR -89-1511
CONF-890590--7

Received by OSTI

JUN 0 7 1989

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36

TITLE VMS ALAP 1.0: AN AUTOMATED AUDIT TRAIL ANALYSIS TOOL

LA-UR--89-1511

DE89 012593

AUTHOR(S) David P. Martinez

SUBMITTED TO 12th Computer Security Group Conference, Amarillo, May 2-4, 1989

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Los Alamos National Laboratory
Los Alamos, New Mexico 87545

DISTRIBUTION

VMS ALAP 1.0
An Automated Audit Trail Analysis Tool

David P. Martinez
Department of Energy
Center for Computer Security
Los Alamos National Laboratory
Los Alamos, New Mexico 87545

ABSTRACT

Because multiuser computer systems typically record enormous quantities of information about user and system activities into system log files, auditing computer user/system activities is a formidable task. Recognizing that a manual audit of these log files is difficult and usually ineffective, the DOE Center for Computer Security has developed an automated audit trail analysis tool, Audit Log Analysis Package (ALAP). ALAP employs methodology developed at Los Alamos National Laboratory for the detection and analysis of anomalous data in large databases. ALAP is capable of processing vast amounts of audit data for detection and analysis of anomalous computer user and system behavior. The first application tool is VMS ALAP 1.0, targeted for Digital Equipment Corporation (DEC) Virtual Memory Systems (VMS).*

1. INTRODUCTION

As the past year has emphatically reminded us with media coverage of viruses, computer break-ins, thefts, espionage, and other malicious actions, we are vulnerable to many forms of attack. It is frustrating not to be able to administer absolute security over most computing system resources. Ideally, we would like operating systems that, through--and from within--themselves, ensure security against waste, fraud, and abuse of computing resources. Given current technology and the computer security problems we face today, we must develop computer security tools, procedures, policies, and the like to deter and preferably prevent the occurrence of waste, fraud, and abuse of our computing resources.

Department of Energy Orders do require auditing of computer system activities to ensure that waste, fraud, and abuse are detected and prevented.** However, performing effective auditing of computer systems activity is inherently complicated because

1. Most computer systems typically generate enormous amounts of audit data; consequently, it is extremely time consuming to manually audit the data.

*DEC, VAX, and VMS are trademarks of Digital Equipment Corporation.

**"I. RESPONSIBILITIES AND AUTHORITIES: 6. Each COMPUTER SYSTEM SECURITY OFFICER (CSSO) shall: h. Develop, implement and document a continuing audit and review process for the classified ADP system to prevent or detect security infractions and the occurrence of waste, fraud, or abuse." DOE 5637.1, Page I-6]

2. The data are often cryptic in nature; consequently, it is very difficult, and most often impossible, to identify disjoint patterns of misbehavior.
3. Given the large volume and cryptic nature of the data, it is often impossible to perform a complete and effective analysis of audit data.

Our approach to reducing the auditing problem was to develop a tool that

- characterized "normal" computer user and system behavior,
- filtered (i.e., reduced) audit log data down to a manageable level,
- identified suspicious computer system behavior, and
- provided mechanisms for rapid review and analysis of suspicious system behavior.

While it is true that we cannot currently ensure absolute security against all malicious actions and many types of computer security breaches, it is important, and for many systems required, that we be able to identify, analyze, evaluate, and report such actions. With this in mind, the Department of Energy Center for Computer Security (DOE/CCS) developed ALAP.

2. THE VMS ALAP 1.0 TOOL

The first version of ALAP is targeted for standard VAX/VMS (VMS 4.6 or higher) operating systems. The software is written in the VAX C programming language and utilizes the standard VAX C Curses library for its screen interface. The screen interface is designed to run on a DEC VT100 or fully compatible terminal. ALAP utilizes VMS Image Accounting termination records as the data used for audit/analysis. Using this data, ALAP generates a profile (i.e., rulebase) of normal and valid computer system activity and utilizes this profile for analysis of data from subsequent VMS accounting files.

2.1. System Data Requirements

By utilizing standard VMS audit and accounting data for ALAP processing, we did not have to modify VMS operating system software. We designed the software to run and operate on standard VMS operating systems and related data and worked to maintain portability of ALAP across VMS operating systems. We reviewed and evaluated available standard VMS audit/accounting data and gave special consideration to

- memory requirements for the raw VMS audit data, ALAP system files, and ancillary data files;
- processing and maintenance requirements; and
- the value of the audit data information to the overall audit function.

After review and evaluation of the available standard VMS audit/accounting data, we selected the VMS Image Accounting data record type because

1. The raw data was collected and stored in a condensed binary format (in an already existing accounting data file--VMS ACCOUNTNG.DAT). The condensed Image Accounting data are relatively small compared to other available VMS audit data.

2. Standard VMS utilities provide invaluable resources for selecting and maintaining the raw audit data sets.

3. The Image Accounting data can encompass all system and user VMS image activations and provide valuable information about those actions.

Because we use VMS raw data, the ALAP rulebase (or profile of a system's activity) consists of rules derived from the historical VMS Image Accounting data (see Table I).

TABLE I

Field	Definition
CPU time	The amount of central processor unit time used by the image executed.
Terminal	The name of the terminal port used to invoke the given image.
Node_name	The name of the alternate node if a user has signed into the system from another node on the system's network.
Node_ID	The node identification for the given image transaction.
Day	The day the given image was executed.
Hr	The time of day the image was executed.
Image	The name of the image executed.
Username	The VMS user account name under which the given image transaction was executed.
Privilege	The VMS privilege code representing account privilege(s) active during the execution of the image.
Status	The final status code returned upon completion of the executed image.
Dir IO	The number of direct data input/output operations performed by the executed image.
Buf IO	The number of buffered data input/output operations performed by the executed image.

Other data fields contained within the VMS Image termination record type are not used by ALAP. In addition, there are other VMS audit data elements that could conceivably enhance the ALAP audit data information; however, overhead requirements to collect and maintain such data is simply too high.

2.2. The VMS ALAP Methodology

The methodology incorporated into ALAP is one developed at Los Alamos National Laboratory, by Hank Vaccaro, for the detection and analysis of anomalous transactions in large databases. It is based on the concept of anomalous data being detected using rules derived from a profile of acceptable behavior for a given system. ALAP uses historical data that encompass normal activity for a given system to develop a rulebase of acceptable behavior for that system. Then, subsequent transactions for the same system can be compared against the rulebase to determine if anomalies exist or not.

2.3. The VMS ALAP Design

ALAP is designed to significantly reduce computer auditing tasks by processing audit log data to detect computer user and system abnormalities and to display anomalies for analysis. Although ALAP's processing speed permits real-time processing, the VMS application of ALAP was not designed to have this capability. Instead, the design permits the user to establish an audit environment tailored to the user's own needs and computing environment.

Application of the ALAP audit tool can vary significantly from system to system. Some systems have a set of well-defined tasks that are performed on a routine basis, with a well-defined set of active user accounts. These systems typically yield audit data that is well behaved. That is, the activities performed on the system do not vary greatly. Systems of this type are generally easier to audit. In contrast, systems with a fluctuating set of active user accounts and on which a variety of tasks is performed are generally more difficult to audit. For this type of system (say, a research and development environment), the audit data must be carefully scrutinized to determine if activities are acceptable or not.

VMS ALAP derives the "normal" patterns of behavior from raw VMS Image Accounting data and builds profiles of expected behavior for computer users, terminals, image executions, CPU time utilized by the image executed, and all other fields in the audit record (see Table I). It also develops rules for user/terminal-related activities, referred to as "sessions." A session is defined as a set of transactions for a given username-terminal combination, from the time the first transaction is encountered (possibly a login) to the last transaction encountered (possibly a log out), in the audit data. Following are some examples of the type of rules VMS ALAP can generate:

- Given Terminal=OPA0 and Image=AUTHORIZE.EXE implies that User=SYSTEM.
- Given User=OPERATOR, Time=08:00, and Day=Friday implies that Image=(BACKUP.EXE, COPY.EXE, MOUNT.EXE, etc.)

The general format of a rule is

> Given a specific value or values for a field or set of audit fields on the left-hand side implies the value(s) for an audit field on the right-hand side as X(i), where X(i) is a value, or set of values, derived by ALAP from the historical data as acceptable for the field on the right-hand side.

In essence, ALAP builds rules on all the data fields (Ref. Table I), and all probable combinations thereof. Some rules are conditional on values for several fields, while others are derived from data values for a single data field. ALAP can generate tens to hundreds of thousands of rules, depending on the characteristics and attributes of the data

ALAP uses the rules (i.e., the rulebase) derived from the raw VMS Image Accounting data as its premise for detecting anomalies. ALAP's design permits the user to tune the ALAP anomaly detection algorithm to site-specific needs. This is done via the ALAP screen interface through the adjustment of an anomaly detection threshold, sometimes referred to as a "Figure of Merit"[1] value. During its monitoring phase, ALAP compares the values of each field in the audit record to the rulebase. It then generates scores for the values of each of the audit record's fields. The score for each field is a function of the rule grades of the rules violated for each field, where a grade is defined as a measure of the accuracy of the rule.[1] Theoretically, a field value is considered "perfectly normal" if the score for it is equal to zero.

As ALAP compares each transaction against the rulebase, it sums the individual field scores to obtain a total score for the transaction and the related session. ALAP is designed to keep scores for up to 32 of the most recent session transactions. A user-tunable decay factor is provided to allow the user to decay past session transactions as desired. The algorithm used to decay the session score is defined as follows:

$$FOM\_S_n = FOM\_T_n + FOM\_S_{n-1} * \text{Decay Factor}$$

where,

FOM S = the ALAP session score.
FOM T = score for the current transaction.
Decay Factor = the decay factor for the cumulative session score (greater than zero, and less than or equal to one).

ALAP uses the above algorithm each time it processes a new transaction and detects a transaction as an anomaly when the session score exceeds the user-defined anomaly detection threshold.

2.3.1. The ALAP User Interface

ALAP is designed with a user-friendly software interface that provides numerous options and features to permit easy and effective auditing and analysis of computer system/users activities. The user interface features include

- an on-line Help facility that enables the user to acquire help for all menu options,

- an ability to skip forwards or backwards in time within the bounds of the audit data time interval,

- a logging facility that records, to a standard ASCII file, information about anomalous transactions, and

- an ability to review and analyze failed rules for anomalous transactions to determine the reason(s) the transaction is anomalous.

Some options are user tunable to meet the user's site-specific needs and characteristics of the computing environment, such as

- the Select option, used to select specific audit records for auditing;
- "Watch" windows to allow the user to actively watch selected username/terminal session activity; and
- the anomaly detection threshold.

3.0. THE VMS ALAP PROCESSES

Before VMS ALAP can be used for auditing/analysis of computer system/user activities, raw VMS Image Accounting data must be subjected to data filtering, formatting, and condensing phases of processing.

3.1. Preprocessing VMS Accounting Data

The preprocessing step filters VMS Image Accounting data from the raw VMS Accounting data file (i.e., ACCOUNTNG.DAT). During this phase, data are extracted, reformatted, and written to a binary file that is subsequently used as an input file to ALAP. The filtering process is independent of ALAP and is used whenever the user wants to filter new raw VMS Accounting data to serve as the data to be audited or to filter historical data subsequently used by ALAP to develop a rulebase.

A VMS command procedure is provided with the software distribution to assist the ALAP user with the preprocessing function.[2] The command procedure steps the user through the preprocessing phase by prompting for information relative to the raw data being processed.

3.2. Processing Historical Data

After filtering a system's raw historical data, the user is ready to process the data using ALAP. Upon executing ALAP, the user is prompted with some startup screens. After the startup screens, ALAP displays the main menu. There are three processing options at the main menu level of ALAP:

1. Process Historical Data,
2. Generate a Rulebase, and
3. Monitor Activity.

To initiate the "Process Historical Data" option, the user merely enters the single character P. ALAP then reads in the filtered historical data (i.e., the ALAP.HST file generated during the preprocessing phase) and processes it. During processing, ALAP identifies unique patterns of behavior exhibited within the historical database. To obtain high performance in rulebase generation, ALAP processes the data through a condensing process. The condensing process is performed with two passes through the data. The first builds a dictionary of unique values encountered in audit record data file. The second creates a condensed file of indexes pointing to the unique values stored in the dictionary during the first pass. Most fields have fewer than 254 unique values so the index can usually be represented by 1 byte. Remaining fields are represented by 2-byte indexes, allowing at most 65534 distinct values. This approach permits ALAP to process the data in random access memory, resulting in high performance during the rulebase generation phase.

3.3. Generating a Rulebase

After processing the filtered system data by means of the "Process Historical Data" option, the user can generate a rulebase. To initiate the "Generate a Rulebase" option, the user enters the single character G. During this process, ALAP develops a rulebase by utilizing all the unique values identified during the "Process Historical Data" option. The rules constitute a measure of typical and expected behavior for a given computing system environment. The more consistency that exists within the daily operations performed on a given system, the more effective you can expect the rulebase to be in identifying anomalous behavior.

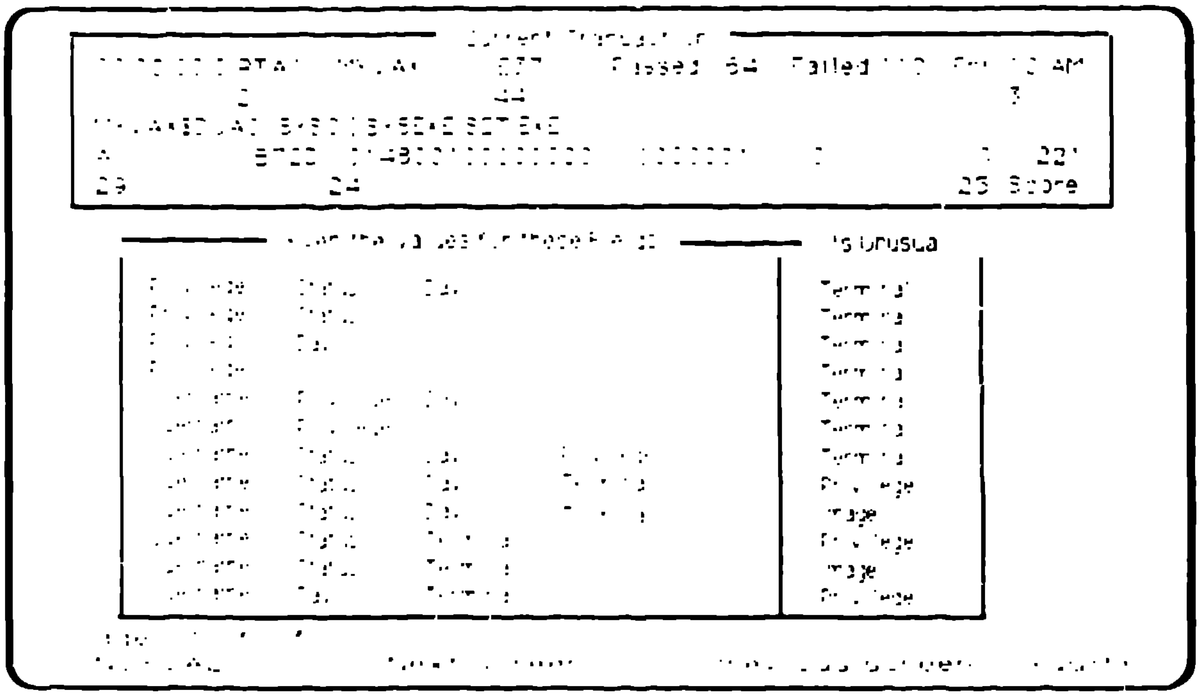
The ALAP user must keep abreast of the system's environment characteristics to ensure that the historical data used to generate the rulebase appropriately reflects the normal or acceptable behavior for the system. If the rulebase becomes outdated (by introduction of new user-accounts, software tools, hardware upgrades, and so forth), the user should generate a new rulebase to incorporate new "acceptable" system activity.

3.4. Monitoring/Auditing

For effective monitoring and auditing of system/user activity, it is imperative that the rulebase depict expected system behavior for the given system. Assuming that the rulebase does adequately reflect expected behavior for the system, the user can monitor/audit new VMS Image Accounting activity by, first, extracting the raw VMS Image Accounting data to a file. Secondly, the user should preprocess the data through the ALAP filtering process. Lastly, when the data have been filtered appropriately, the user can audit the data by simply executing ALAP and invoking the the "Monitoring Activity" option from the ALAP main menu.

At this point, ALAP initiates the monitoring/auditing phase by opening the filtered audit data file for auditing and by displaying the monitoring/auditing options menu screen.

The user interface provides several options that permit the user to effectively review, analyze, and evaluate anomalous transactions such as the Unusual option. The Unusual option is available from within the Analyze Menu and displays a summary of the audit record field(s) found anomalous by ALAP.



Other valuable audit analysis options include

SELECT - Displays the Select Menu allowing the user to select specific audit record fields for auditing, such as sensitive, terminal ports, user privileges, and accounts.

SUGGEST - Displays the suggested values for the various audit record fields based on the rules derived from the historical data. Associated with the possible field values are the scores that the field would assume if the particular field value were to appear in the audited transaction.

4. VMS ALAP 1.0 PRODUCT DEVELOPMENT

The Department of Energy, Center for Computer Security (DOE/CCS) has an ongoing task to develop various computer security tools to assist CSSOs and other computer security personnel with their computer security duties and responsibilities. ALAP is the first of the auditing tools. It is our goal to provide the DOE computing community with the best computer security tools possible--tools that perform computer security tasks, or assist in performing computer security tasks, required by DOE orders and the needs of the DOE computing community.

4.1. The ALAP Software Life Cycle

As with many software tools, the concepts and methodology employed within ALAP began in a research environment, namely the Nuclear Materials Control and Accounting (MC&A) environment. The methodology was, and continues to be, investigated as a method for resolving anomalous transactions in large MC&A systems. During the early research stages of the methodology, scientists at Los Alamos National Laboratory considered application of the methodology to other disciplines, one of which was computer security.

A research and development effort was henceforth initiated by DOE/CCS to develop a prototype computer security auditing package for a computer operating system type prevalent in the DOE--the system selected was VMS. The prototype generated useful results during testing; therefore, the Center pursued development, testing, and production of the software auditing tool now known as ALAP.

4.2. ALAP Testing Procedures and Results

Development of a software tool for wide application in different computing environments is difficult, particularly when the software must be integrated with existing software, such as operating system software, computer language compilers, and software libraries. The software must be carefully designed to integrate with such systems in an environment independent and portable manner. That is, the user should not have to modify the computing environment to incorporate such tools. For this reason, ALAP was designed to run and utilize standard VMS operating systems and data, the VAX C compiler, and the standard Curses screen management library.

Issues of primary concern relative to the software development life cycle include software integrity, security, reliability, maintenance, and extensibility. Assurance of these elements in a software product requires application of good software engineering principles. Good software engineering is not only valuable, but absolutely necessary.

The quality and reliability of the tool in different computing environments are also of concern. If the software is developed appropriately, with properly documented requirements specification and design specifications, it can be systematically integrated and tested to ensure that it operates as specified. To do so, the software must be proven to function as designed in appropriate test environments.

The ALAP software, for the most part, was developed in a research environment with strong emphasis on software prototype and proof of concept issues. Consequently, ensuring adequate quality and reliability of the tool for general application in varying VMS computing environments posed a considerable challenge. Initially, the code was restructured and documented to improve the maintenance of the tool. However, most of the burden of quality and reliability assurance was shifted to the test phase. Consequently, strong emphasis was placed upon testing and assurance of software reliability in various computing environments.

4.2.1. Testing ALAP in the Local Environment

During testing of ALAP, the software was subjected to thorough local testing (i.e., DOE/CCS) to identify all possible problems. The problems were documented and then reviewed and evaluated by the ALAP development team. Revision and refinement plans were developed to correct the known problems in the software. After completion of revisions and refinements, the software was again subjected to local test procedures to validate that the problems had been corrected.

An ALAP User's Guide was also designed and developed. Together, the ALAP software and documentation were subjected to local testing to ensure coherence between the two. Software and documentation refinements were then performed and a new version of ALAP (i.e., Alpha test version) was developed.

4.2.2. Alpha Testing ALAP

After completion of the local test phase, ALAP was subjected to Alpha testing at a few sites at Los Alamos National Laboratory. Test results were accumulated and documented. Several software problems were identified, along with structural problems with the ALAP User's Guide. All problems were again reviewed and evaluated by the ALAP team, and subjected to the Center's local revision and test procedures. The ALAP User's Guide was submitted to a computer software documentation professional for restructuring and editing. After completion of the revisions and refinements of the software and documentation, the package was again subjected to coherence testing. Subsequently, ALAP packages were developed and distributed to volunteer Beta test sites within the Department of Energy.

4.2.3. Beta Testing ALAP

Pre-beta test results have confirmed that VMS ALAP is a valuable mechanism for improved auditing of VMS operating systems. Beta testing was initiated in March 1989 and is ongoing. As with Alpha testing, we anticipate Beta testing to yield invaluable results that will permit the Center to revise and refine ALAP into a general VMS audit tool for public release. The tool is scheduled to be released before the end of fiscal year 1989.

4.2.4. Tests Results

There are some limitations to the current version of the software. Some noted deficiencies include

- lack of real-time processing,
- Image Accounting data inadequacies,
- non-tunable rulebase features,
- lack of software system extensibility, and
- restricted input data flexibility.

On the positive side, ALAP provides mechanisms for improved auditing of VMS operating systems such as

- the capability to detect suspicious system behavior,
- several options for rapid review and analysis of suspicious activity, and
- the filtering down of data to the suspicious subset, thus making the auditing task more manageable.

5. CONCLUSIONS

The supporting technologies for computer auditing have been growing and improving at an incredible pace. The methodology incorporated into ALAP during the development phase has now been extended in a research environment to include improved capabilities such as

- support of multiple rulebases, including human-generated rulebases,
- rulebase editing and tuning, and
- threads.*

As it is true that we have yet to achieve maximum computer processing speeds, so it is true that we have yet to achieve optimum anomaly detection/resolution capabilities. However, as with computers, we must use whatever capabilities current technology permits. ALAP by no means is intended to be the ultimate VMS auditing tool; however, it does provide an improved mechanism for computer auditing of VMS computer operating systems.

For auditing systems of the future, we need to collect VMS ALAP application results and comments from the field. We need to establish a standard Audit Record Definition that encompasses the information needed to audit DOE computer systems activity to meet the DOE requirements effectively and meet the needs of the field as determined through application of VMS ALAP. Lastly, we must redesign and develop new audit trail analysis tool(s) that incorporate new anomaly detection/resolution technology and are designed to provide software integrity, security, reliability, maintenance, and extensibility.

*A "thread" consists of a set of interdependent or related audit data [LA-UR-88-3656, Page 4]. The ALAP Username/Terminal session structure is considered a "thread" type, where the Username/Terminal session values are interdependent or related. Audit field relationships can occur amongst other audit record data fields, yielding valuable audit information.

ACKNOWLEDGMENTS

This work was supported by the U.S. Department of Energy, Office of Safeguards and Security. The methodology and concepts incorporated into ALAP are those developed by Hank Vaccaro (Safeguards Systems Group, Los Alamos National Laboratory). Special thanks to Leon Breault (Branch Chief, Department of Energy Center for Computer Security) and all the management and staff of the DOE/CCS for their contributions to this tool.

REFERENCES

1. H. S. Vaccaro, "Detection of Anomalous Computer Session Activity," Los Alamos National Laboratory document LA-UR -88-3656 (Rev.)

2. Department of Energy, Center for Computer Security, "ALAP for Computer Security/Audit Log Analysis Package."