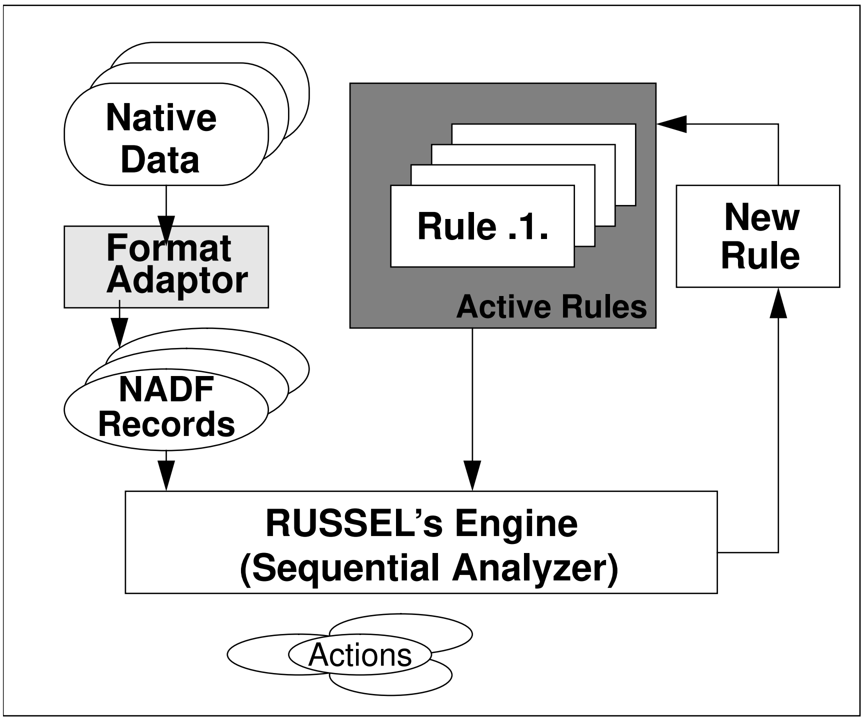
ASAX - The Advanced Security audit trail Analyzer on uniX

ASAX CONCEPTUAL OVERVIEW

THE POWER OF RUSSEL

ASAX's language, RUSSEL, can be viewed as a procedural language using classical programming structures to express conditional, repetitive and sequential actions.
But RUSSEL disposes furthermore of a particular predefined control structure, tuned to analyze sequences of records as described hereunder:

The data is analyzed sequentially (record by record) by means of a collection of rules. At any given time, there is a current record being analyzed and a collection of rules active.
Active rules encapsulate all relevant knowledge about the past analysis. This knowledge is applied on the current record along with the active rules for that record. The process, in turn, generates new rules to be applied in the future.
A rule can be conceived as a parametrized procedure, however with one particular distinction : every rule can trigger off new rules. This action implies a.o to specify when the rule should be fired : for the current record, for the next record or at the end of the analysis.



For more information, contact:

Institut d'Informatique

Abdelaziz MOUNJI
rue Grandgagnage, 21
B-5000 NAMUR
BELGIUM

Tel : +32.81.72 49 87
Fax: +32.81.72 49 67
email: amo@info.fundp.ac.be

INTRODUCTION

Analyzing substantial amounts of data and extracting relevant information out of huge sequential files has always been a nightmare. And ... it will probably remain so, unless you use ASAX, FUNDP' Advanced Security audit trail Analyzer on uniX.

Using highly sophisticated and powerful algorithms, ASAX tremendously simplifies the intelligent analysis of sequential files.

Of course, the data should fit the analyzer. Therefore, ASAX has defined a normalized audit file format (NADF) with built-in flexibility to guarantee a simple and straightforward translation of any stream of native data into the normalized sequential files ASAX understands.

But ASAX's real power is unleashed by deploying its embedded, easy to use rule based language RUSSEL; this tailor-made analysis tool solves very intricate queries on any sequential data.

ASAX KEY FEATURES

UNIVERSALITY

Heterogeneous computing burdens the tool with the necessity to analyze data originating from various sources. ASAX fulfils this requirement by procuring a broad range of format adaptors; they convert existing data into ASAX's universal NADF records. Imparting a firm "open" character to ASAX, these simple-to-write format adaptors eliminate the need to tune the product for every possible source of data.

POWERFUL RULE-BASED LANGUAGE

It is common knowledge that rule-based languages are extremely powerful to process streams of data incrementally. For example, they can be tuned to recognize particular patterns in sequential files and trigger off appropriate actions. ASAX's rule-based language, RUSSEL, is a conceived accordingly as a state-of-the-art language amalgamating procedural and declarative paradigms.

RUSSEL provides classical control structures as well as a particular rule-based triggering mechanism to express the most complicated queries.

EFFICIENCY

In most cases, efficiency is a prerequisite given the large amount of data to be analyzed. Two key principles of the design cope with this need: ASAX scans the NADF records one and only one pass and optimizes repetitive steps as much as possible. Despite very complex queries, it is unnecessary to read previous records again: the required information is automatically memorized and released when no longer needed.

The above features make ASAX the ultimate tool to examine security audit trails, automate time-consuming tasks in computer centres ... or to simplify any job involving the analysis of large amounts of sequential data.