Rocket

WHITE PAPER

Audit Logging for Rocket® UniVerse®

Improve Security | Assist with Compliance and Audits

Audit Logging for Rocket® UniVerse®

Introduction

Every day brings a new and complex set of interactions, events, and activities within your business applications and data. What happens when something unexpected occurs related to one or more of them? Whether as a best security practice for regulation compliance, or to pass an audit organizations today need to know what's happening to their data and applications.

Compliance is the most obvious driver for introducing Audit Logging to a Rocket® UniVerse® implementation. When it comes to compliance, you have to be right, and you have to move quickly. In addition to compliance, organizations can also use Audit Logging for:

- Accountability: Identify the accounts associated with certain events and use the information to make decisions about training or disciplinary action
- Reconstruction: Track data to the "tick" (microsecond) level and sequentially within each unique user process, to see and understand what happened before and during an event
- Intrusion Detection: Review unusual and unauthorized events—such as failed login attempts or logins outside of a specified schedule—that might indicate attempts to breach security
- Problem Detection: Analyze log data to identify problems that you need to address, such as reviewing resource utilization or failed jobs

Audit Logging

With Audit Logging, organizations running Rocket® UniVerse® can lower their risks by gaining insight into activities in their data and application environment, so they can address underlying issues. For example, with Audit Logging, if a user is viewing clients not assigned to him or her, or collecting sensitive information such as social security numbers, proper logging can help auditors identify improper access privileges or detect and put a stop to patterns of abuse.

The basic principles of Audit Logging in UniVerse involve the monitoring of system, data, and user resources that trigger events. An audit event is an action on a system resource. It might be a user running a maintenance program that uses resources like files, system resources, programs, and utilities (for details on types of events, please see Appendix A).

Audit Logging always logs some events, such as system-level and security-related operations. For example:

- Any creation of a hashed file
- An execution of a UniVerse BASIC READ statement
- A logon failure by a client through a UniVerse server

Logging database resource usage and related authentication and authorization operations can help you significantly reduce preparation time for compliance audits—and increase your rate of passing these audits. Whether for HIPAA, HITECH, PCI-DSS, SOX, GDPR, the Fair Credit Reporting Act, or others, Audit Logging gives you the data you need. It also includes reporting and archiving features that help you demonstrate compliance.

Audit Logging Change Data Capture (CDC)

CDC captures changes at the individual field level (instead of at the record level), providing specifics of what data was changed by who (and when) while also resulting in improved performance. CDC also helps with compliance reporting since it captures exactly what was changed, which many mandates require. You'll see a reduction in time, resources, and money spent complying with audit data requirements for regulations including PCI and HIPAA.

> *"Audit Logging allows users to check off numerous compliance questions when undergoing security audits."*
>
> Steve McConnell,
> Columbia Ultimate Technical Support Supervisor

Improved Performance

To boost performance, we've made Audit Logging multi-threaded, and streamlined the audit architecture. This new architecture improves performance by minimizing the number of processes that deal directly with the Audit Logging product. For example, a single process reads the configuration file, changes the Audit Logging environment, and handles the events and user auditing processes, rather than using separate processes. You can configure up to eight audit threads to run simultaneously, each handling a large volume of process logging.

> *"UniVerse has all the answers, if you can think of an audit question, it can be configured to provide the answer."*
>
> Russell Patterson,
> IT Specialist, Rural Finance

Log Only What You Need to Log

Audit Logging now also gives you the ability to log only what you need for simplified reporting.

- Flexible: you can configure it through policies to selectively or collectively log events
- Easy to configure: authorized users can make changes to Audit Logging on the fly without having to stop or restart the application
- Customizable: if we don't currently log it, you can add a custom event to any custom code that is then logged like any other audit event

One Rocket customer that must adhere to the Fair Credit Reporting Act uses Audit Logging to record user access to sensitive customer data and other events that might indicate a security breach. The company creates an exception report so it has a record of what's important (primarily unauthorized access and possible security breaches). The result? Simplified reporting and lower demand for system resources.

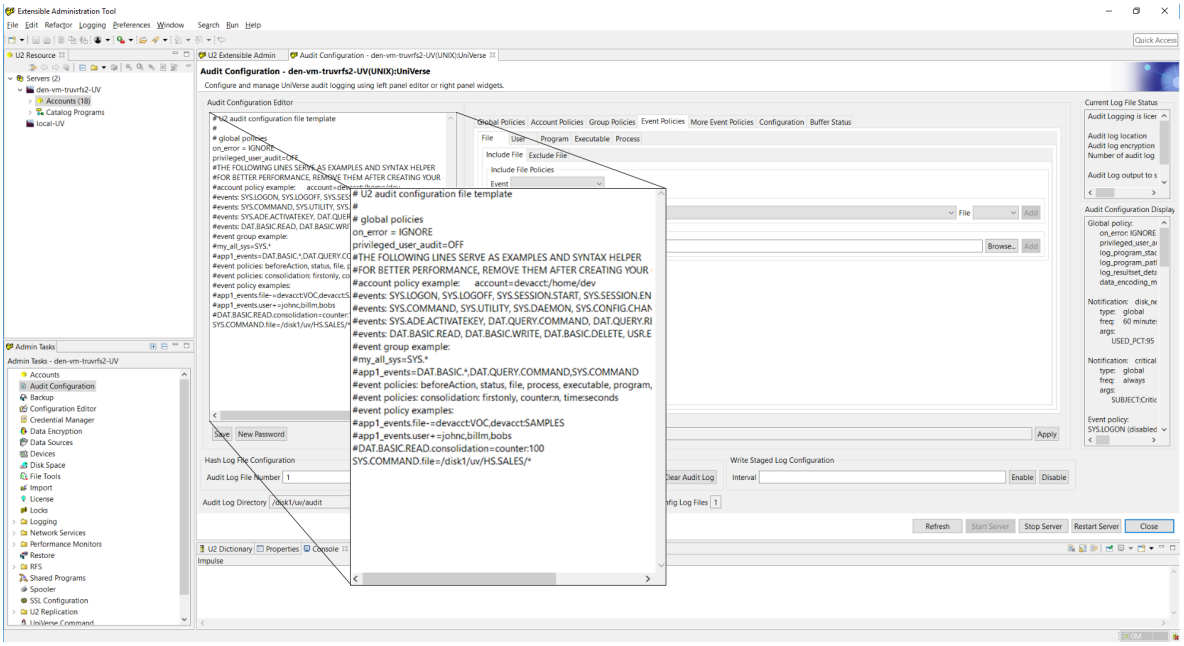

Figure 1: If you're in IT and responsible for supporting a department dealing with compliance and audits, you need to be able to easily manage UniVerse Audit Logging. UniVerse offers a GUI interface using XAdmin or a command line interface to monitor and maintain your audit environment.

Automatic Audit Compression

Automatic Audit Compression helps manage the size of the data and decreases the need for extra storage capacity, as audit log files can become massive. You can also use the non-compression function for a more granular view of what data has changed.

Simplified Reporting

During an audit, you must show that your company is compliant, and Audit Logging can help. By creating reports from the audit log, you can quickly, easily, and accurately answer many of the typical questions that arise during an audit. For example:

- Who updated, deleted, or changed an account?
- When did a specific user log in or out of an account?
- Which users have access to specific data, and when did they access the data?

Audit Logging provides three configurable file types for reporting:

- A hashed type-30 offers the ability to use the native query language of the UniVerse database
- A sequential file offers reporting through popular reporting tools; sequential file logs also significantly improve performance
- The UNIX/Linux syslog can be directed to another system for “offload” reporting

Improved Security

We’ve worked to make Audit Logging more secure than ever. The Config file can only be modified by authorized users. It is encrypted, and you can set it to be password protected. You can also set the Audit Log file to be encrypted.

Upgrade to UniVerse and Harness the Power of Audit Logging

Whether for an audit, compliance, security, or resource-problem detection, Audit Logging provides a comprehensive, flexible, easy-to-configure solution for monitoring UniVerse databases and application activities.

Contact Rocket Software or your application provider today to request a demo.

Appendix A:

UniVerse Audit Logging classifies the following events as:

UniVerse system events

- SYS.LOGON: A user logon request through one of the UniVerse servers
- SYS.LOGOFF: A user logs off from a UniVerse server
- SYS.SESSION.START: A UniVerse session is initiated
- SYS.SESSION.END: A UniVerse session ended
- SYS.ADE.ACTIVATEKEY: Automatic Data Encryption key activation and deactivation
- SYS.COMMAND: A TCL command has been run, other than query commands
- SYS.UTILITY: Running of any UniVerse utilities

NOTE: SYS.COMMAND and SYS.UTILITY can be restricted to a subset of TCL commands or utilities

System configuration events

- SYS.CONFIG.CHANGE A system-level configuration changed, such as a uvconfig, audit configuration, or replication configuration change
- SYS.SECURITY SQL GRANT, REVOKE, future security operations
- SYS.ADE Automatic Data Encryption operations: master key, key store, key creation, key deletion, file encryption, index encryption, password related operations
- SYS.DAEMON Starting or stopping of UniVerse background processes, such as U2Rep services

Data events

- DAT.QUERY.COMMAND LIST, SORT, SELECT, SUM, REFORMAT, COUNT, TLOAD, TDUMP, CVIEW SQL SELECT
- DAT.QUERY.RESULTSET LIST, SORT, SELECT, SUM, REFORMAT, COUNT, TLOAD, TDUMP, CVIEW SQL SELECT
- DAT.BASIC.READ BASIC READ, READV, MATREAD, SELECT, SELECTINDEX, READSEQ, READBLK, BSCAN
- DAT.BASIC.WRITE BASIC WRITE, WRITEV, MATWRITE, WRITEBLK, WRITESEQ, WEOFSEQ
- DAT.BASIC.DELETE BASIC DELETE, CLEARFILE
- DAT.SQL.COMMAND SQL commands, except SELECT, GRANT, and REVOKE

NOTE: DAT.*.WRITE and DAT.*.DELETE events will also capture the before/after data associated with the Change Data Capture capabilities of UniVerse Audit

User event

- USR.EVENT User-defined audit event through the new UniVerse BASIC AuditLog() function

2019-04 RS MV WP UniVerse Audit Logging v3

Rocket®

rocketsoftware.com
info@rocketsoftware.com
US: 1 855 577 4323
EMEA: 0800 520 0439
APAC: 612 9412 5400
twitter.com/rocket
www.linkedin.com/company/rocket-software
www.facebook.com/RocketSoftwareInc
blog.rocketsoftware.com