Auditing

Version 2024.1
2024-05-02

InterSystems Corporation   1 Memorial Drive   Cambridge MA 02142   www.intersystems.com

Auditing
InterSystems IRIS Data Platform Version 2024.1 2024-05-02

For Support questions about any InterSystems products, contact:

InterSystems Worldwide Response Center (WRC)
Tel: +1-617-621-0700
Tel: +44 (0) 844 854 2917
Email: support@InterSystems.com

Table of Contents

List of Tables

Auditing

Logging certain key events in a secure audit database is a major aspect of InterSystems security. InterSystems IRIS® allows you to monitor events and add entries to the audit database when these events occur. These events can occur within InterSystems IRIS itself or part of an application. The knowledge that all activities are being monitored and that all logs can be reviewed is often an effective deterrent to malicious behavior.

Note: This document describes how to manage audit events with the Management Portal. To manage audit events programmatically, use the Security.Events class.

You can also enable *structured logging*, which will write the same messages seen in the audit database to a machine-readable file that can be ingested by your choice of monitoring tool. See Setting Up Structured Logging.

1 Basic Auditing Concepts

InterSystems IRIS allows you to enable or disable auditing for the entire InterSystems IRIS instance. When auditing is enabled, InterSystems IRIS logs all requested events. Auditable events fall into two categories:

- System audit events — InterSystems IRIS system events that are only logged if they are explicitly enabled.
- User—defined audit events — Application events, which are only logged if they are explicitly enabled.

InterSystems IRIS system events are built-in events that monitor actions within InterSystems IRIS, such as start-up, shutdown, logins, and so on; security-related events, such as changes to security or audit settings; and interoperability-related events, such as changes to a production configuration or schema.

InterSystems IRIS does not automatically audit database activity, such as inserts, updates, or deletes for a table, because this kind of activity typically generates so many audit entries as to be useless — or even counterproductive — due to the performance impact on the system. For example, if a medical records application were to log all access to patient medical information, then one such access event might result in hundreds or thousands of database accesses. It is much more efficient to have the application create a single audit entry, rather than have the database manager generate thousands.

2 Enable or Disable Auditing

In the **Auditing** menu (**System Administration** > **Security** > **Auditing**), there are selections to enable and disable auditing. If the **Enable Auditing** choice is available, this means that auditing is disabled; if the **Disable Auditing** choice is available, this means that auditing is enabled. InterSystems IRIS auditing is disabled by default for minimal-security installations; it is enabled by default for normal and locked-down installations.

If you enable (turn on) auditing, then InterSystems IRIS audits:

- All system events that are enabled
- All user-defined events that are enabled

2.1 Enable Auditing

To turn on auditing, on the **Auditing** menu (**System Administration** > **Security** > **Auditing**), select **Enable Auditing**.

2.2 Disable Auditing

To turn off auditing, on the **Auditing** menu (**System Administration** > **Security** > **Auditing**), select **Disable Auditing**.

3 Elements of an Audit Event

Audit information is available in the IRISAUDIT database. New entries are added to the end of the log. When you view the audit log, you see the following elements for each entry:

Time (also called UTCTimestamp)

UTC date/time when the event was logged.

Event Source*

The component of the InterSystems IRIS instance that is the source of the event. For InterSystems IRIS events, this is “%System” or “%Ensemble”. For user-defined events, the name can be any string that includes alphanumeric characters or punctuation, except for colons and commas; it can begin with any of these characters except for the percent sign. This can be up to 64 bytes.

Event Type*

Categorizing information for the event. This string can include any alphanumeric characters or punctuation, except for colons and commas; it can begin with any of these characters except for the percent sign. This can be up to 64 bytes.

Event* (also called Event Name)

Identifier of the event being logged. This string can include any alphanumeric characters or punctuation, except for colons and commas; it can begin with any of these characters except for the percent sign. This can be up to 64 bytes.

PID (also known as a Process ID)

Operating system ID of the InterSystems IRIS process that logged the event. InterSystems IRIS uses the OS PID in its native form.

Web Session (search results only)

The session ID, if there is one, of the web session that caused the event.

User (also called Username)

Value of *$USERNAME* for the process that logged the event.

Description

A field of up to 128 characters that applications can use to summarize the audit event. This field is intended for a user-readable explanation or display (as compared to the combination of EventSource, EventType, and Event, which uniquely define the audit event).

*Each different kind of event is uniquely identified by the combination of its EventSource, its EventType, and the Event itself.

When you click **Details**, you see some of the same elements and the following additional elements:

Timestamp

Date/time when the event was logged, in local time.

JobId

ID of the job.

IP Address

IP address of client associated with the process that logged the event.

Executable

The client application associated with the process that logged the event, if there is one.

System ID

The machine and InterSystems IRIS instance that logged the event. For example, for the machine *MyMachine* and the instance *MyInstance*, the system ID is `MyMachine:MyInstance`.

Index

The index entry in the data structure containing the audit log.

Roles

For all events except LoginFailure, the value of *$ROLES* for the process that logged the event. For LoginFailure, a value of “ ”, as the user is not logged in.

Namespace

The namespace that was current when the event was logged.

Routine

The routine or subroutine that was running when the event was logged.

User Info

User-defined information about the process, added programmatically via the %SYS.ProcessQuery interface.

O/S Username

Username given to the process by the operating system. When displayed, this is truncated to 16 characters.

This is the actual operating system username only for UNIX® systems.

For Windows:

- For a console process, this is the operating system username.
- For Telnet, this is the *$USERNAME* of the process.
- For client connections, this is the operating system username of the client.

Status

The value of any %Status object that was audited.

Event Data

A memo field where applications can store up to 3632952 bytes of data associated with the audit event. For example, it can contain a set of application values at the time of the event or can summarize the old and new states of a record or field.

4 About System Audit Events

System audit events are predefined events that are available for auditing by default. General information about them appears in the table on the **System Audit Events** page (**System Administration** > **Security** > **Auditing** > **Configure System Events**), where the columns are:

- Event Name — The Event Source (which is %System or %Ensemble), Event Type, and Event proper, all together and concatenated with slashes (“/”). The %Ensemble Event Source is used for events related to the interoperability features in InterSystems IRIS.
- Enabled — Whether or not the event is enabled (turned on) for auditing.
- Total — The number of events of this type that have occurred since the last startup of InterSystems IRIS.
- Written — The number of events of this type that have been written to the audit log since the last startup of InterSystems IRIS. This number may differ from the total occurrences.
- Reset — Allows you to clear the audit log for this event and reset its counter to zero. For more information on counters, see Audit Event Counters.
- Change Status — Allows you to enable or disable the event. For more information on these actions, see Enable or Disable an Audit Event section.

They monitor events within the InterSystems IRIS system, including changes to InterSystems IRIS productions. System events are distinguishable by their Event Source value of %System or %Ensemble.

Table 1: System Audit Events

Event Source	Event Type and Event	Occurs When	Event Data Contents	Default Status
%Ensemble	%Message/ ViewContents	A user views the contents of a message in the Message Viewer.	Metadata about the message.	On
%Ensemble	%Production/ ModifyConfiguration	A user modifies the configuration of a production.	A summary of the change.	On
%Ensemble	%Production/ StartStop	A user starts or stops a production.	Action (start or stop) and the username for the initiator of the action.	On
%Ensemble	%Schema/ Modify	A user creates, modifies, or deletes a schema structure.	A summary of the change.	On

Event Source	Event Type and Event	Occurs When	Event Data Contents	Default Status
%System	%DirectMode/ DirectMode	Any command is executed in direct mode.	The text of command.	Off
%System	%Login/ JobEnd	The JOB command ends a background job.	The routine where the **Job** command was executed and the database where the routine is stored. If the values for these fields are null, then the **Job** command was executed from the shell.	Off
%System	%Login/ JobStart	The JOB command starts a background job.	The routine where the **Job** command was executed and the database where the routine is stored. If the values for these fields are null, then the **Job** command was executed from the shell.	Off
%System	%Login/ Login	A user successfully logs in.	The protocol, port number, process ID, and application associated with the login. The user's login roles.	Off
%System	%Login/ LoginFailure	A login attempt fails.	Username.	Varies*
%System	%Login/ Logout	A user logs out.	The application (and, if relevant, the class) associated with the logout.	Off
%System	%Login/ TaskEnd	The Task Manager ends a process.	None. See the Description for the name of the task.	Off
%System	%Login/ TaskStart	The Task Manager starts a process.	None. See the Description for the name of the task.	Off
%System	%Login/ Terminate	A process terminates abnormally.	Varies, as does the Description field's content; see below.	Off
%System	%SMPExplorer/ Change	Data is altered using the Portal, such as by creating, editing, deleting, compiling, dropping, replacing, or purging classes or tables.	Varies, as does the Description field, depending on the action taken. Includes relevant content such as the compile flags or the schema and table being dropped.	Off

Event Source	Event Type and Event	Occurs When	Event Data Contents	Default Status
%System	%SMPExplorer/ ExecuteQuery	A query is executed using on the Portal’s SQL page.	The syntax of the executed query.	Off
%System	%SMPExplorer/ Export	Data is exported through the Portal.	The options selected for data export.	Off
%System	%SMPExplorer/ Import	Data is imported through the Portal.	The options selected for data import.	Off
%System	%SMPExplorer/ ViewContents	Data is viewed through the Portal.	The filters that determined what data was viewed. The Description field specifies what was viewed, such as a list of classes, an individual global, or process information.	Off
%System	%SQL/ DynamicStatement	A dynamic SQL call is executed.	The statement text and the values of any host-variable arguments passed to it. If the total length of the statement and its parameters exceeds 3,632,952 characters, the event data is truncated.	Off
%System	%SQL/ EmbeddedStatement	An embedded SQL call is executed. See below for usage details.	The statement text and the values of any host-variable arguments passed to it. If the total length of the statement and its parameters exceeds 3,632,952 characters, the event data is truncated.	Off

Event Source	Event Type and Event	Occurs When	Event Data Contents	Default Status
%System	%SQL/ PrivilegeFailure	There is an SQLCODE=-99 error, which occurs because a user attempts to execute an SQL statement without the required privilege.	• The SQL error message • The required privilege that the user doesn't have • The entity type where the privilege is missing, such as table, view, stored procedure • The table, view, or other entity on which the user lacks privileges • If there are column-level privileges, the relevant fields	Off
%System	%SQL/ XDBCStatement	A remote SQL call is executed using ODBC or JDBC.	The statement text and the values of any host-variable arguments passed to it. If the total length of the statement and its parameters exceeds 3,632,952 characters, the event data is truncated.	Off
%System	%Security/ Application-Change	An application definition is created, changed, or deleted.	Action (create new, modify, or delete), old and new application data.	On
%System	%Security/ AuditChange	Auditing is stopped or started, entries are erased or deleted, or the list of events being audited is changed.	Action (stop, start, erase, delete, or specify), old and new audit settings.	On
%System	%Security/ AuditReport	Any standard audit report is run.	Identification of audit report.	On
%System	%Security/ DBEncChange	There is a change related to database or data-element encryption.	Varies, as does the Description field's content. See below.	On
%System	%Security/ DocDBChange	A document database application definition is created, changed, or deleted.	A summary of the change and a list of the current values, if applicable.	On

Event Source	Event Type and Event	Occurs When	Event Data Contents	Default Status
%System	%Security/ DomainChange	A domain definition is created, changed, or deleted.	Action (new, modify, delete), old and new domain data.	On
%System	%Security/ KMIPServerChange	A KMIP server definition is created, changed, or deleted, or KMIP servers are exported or imported.	A summary of the action and a list of the current values, if applicable. See the Description for additional details.	On
%System	%Security/ LDAPConfigChange	An LDAP configuration is created, changed, or deleted.	A summary of the change and a list of the current values, if applicable.	On
%System	%Security/ OpenAMIdentityServicesChange	OpenAM Identity Services records are exported or imported.	File name and the number of records exported to or imported from the file.	On
%System	%Security/ PhoneProvidersChange	A mobile phone service provider is created, updated, or deleted.	For creating a provider, its name and the value of its SMS gateway For updating a provider, its name, and the old and new values of its SMS gateway. For deleting a provider, there is no event data; the name of the deleted provider is in the event description	On
%System	%Security/ Protect	A process generates a security protection error.	The error.	Off
%System	%Security/ ResourceChange	A resource definition is created, changed, or deleted.	Action (new, modify, or delete), old and new resource data.	On
%System	%Security/ RoleChange	A role definition is created, changed, or deleted.	Action (create new, modify, or delete), old and new role data.	On
%System	%Security/ SSLConfigChange	A TLS configuration's settings are changed.	The changed fields with old and new values.	On
%System	%Security/ ServiceChange	A service's security settings are changed.	Old and new service security settings.	On

Event Source	Event Type and Event	Occurs When	Event Data Contents	Default Status
%System	%Security/ SystemChange	System security settings are changed.	Old and new security settings.	On
%System	%Security/ UserChange	A user definition is created, changed, or deleted.	Action (create new, modify, or delete), old and new user data.	On
%System	%Security/ X509CredentialsChange	A user creates, updates, or deletes a set of X.509 credentials.	Varies by event. See below	On
%System	%Security/ X509UserChange	Event defined, but not available for auditing until a future release.	N/A	On
%System	%System/ AuditRecordLost	An audit entry has not been added to the audit database due to resource limitations that constrain the audit system (such as disk or database full).	None.	On
%System	%System/ ConfigurationChange	InterSystems IRIS successfully starts with a configuration different than the previous start, a new configuration is activated while InterSystems IRIS is running, or a lock is deleted through the Portal or through the ^LOCKTAB utility.	Username for the user who made the change; previous and new values of the changed element. For deleted locks, information about which lock was deleted.	On
%System	%System/ DatabaseChange	There are changes to database properties. See below.	Details about the particular change. See below.	On
%System	%System/ JournalChange	Journaling is started or stopped for a database or process.	When journaling is started, the name of the database and its maximum size; when journaling is stopped, none.	On
%System	%System/ OSCommand	An operating-system command is issued from within the system, such as through a call to the $ZF(-100) function.	The operating system command that was invoked; the directory in which it was invoked; and any flags associated with the command.	On

Event Source	Event Type and Event	Occurs When	Event Data Contents	Default Status
%System	%System/ RoutineChange	A method or routine is compiled or deleted on the local instance. For more details, see below.	No content, though the Description field depends on the change itself; see below.	Off
%System	%System/ Start	The system starts.	Indication of whether recovery was performed.	On
%System	%System/ Stop	InterSystems IRIS is shut down.	None.	On
%System	%System/ SuspendResume	A process is suspended or resumed.	The process ID of the process.	Off
%System	%System/ UserEventOverflow	An application attempts to log an undefined event.	The name of the event that the application attempted to log.	On

*The LoginFailure event is off by default for minimal-security installations; it is on by default for normal and locked-down installations.

Important: If auditing is enabled, then all enabled events are audited.

4.1 %System/%Login/Logout and %System/%Login/Terminate

A process generates a %System/%Login/Logout event if the process ends because of:

- A HALT command
- Exiting application mode because of a QUIT command
- Executing the **Terminate** method of the SYS.Process class to terminate itself (which is the same as executing **HALT**).

A process generates a %System/%Login/Terminate event if the process exits for any other reason, including:

- The user closes the Terminal window, resulting in a Terminal disconnect. If the process is in application mode, the Description field of the audit record includes the statement "^routinename client disconnect" (where *routinename* is the first routine that the process ran); if the process is in programmer mode, the Description field includes the statement "Programmer mode disconnect."
- A Terminal session is ended by an action in another process, including ^RESJOB, ^JOBEXAM, or the Management Portal. If the process is in application mode, the Description field of the audit record includes the statement "^routinename client disconnect" (where *routinename* is the first routine that the process ran) ; if the process is in programmer mode, the Description field includes the statement "Programmer mode disconnect." Note that the event data will contain the pid of the process which terminated them.
- A core dump or process exception. When a process gets a core dump or exception, it is too late for it to write to the audit file. Therefore, when the clean daemon runs to clean up the state of the process, it writes an audit record to the log with a description "Pid <process nunber> Cleaned".

- A TCP Client disconnect. When a process detects that a client has disconnected, this results in an audit record with a Description field which contains the name of the executable that disconnected, such as “<client application> client disconnect”.

4.2 %System/%SQL/EmbeddedStatement

To use the %System/%SQL/EmbeddedStatement event, you must both enable the event *and* the `#sqlcompile audit` macro preprocessor directive:

```
#sqlcompile audit = ON
```

For reference information, see #sqlcompile audit.

If %System/%SQL/EmbeddedStatement is enabled, then executing any embedded SQL after a `#sqlcompile audit = ON` directive generates an EmbeddedStatement audit event. For example:

```
  ...
#sqlcompile audit = ON
  ...
 &sql(delete from MyTable where %ID = :id)
 // This statement is audited at runtime if %System/%SQL/EmbeddedStatement is enabled.

  ...

#sqlcompile audit = OFF
  ...
 &sql(delete from MyOtherTable where %ID = :id)
 // This statement is not audited at runtime even if %System/%SQL/EmbeddedStatement is enabled.
  ...
```

Because an application may have hundreds or thousands of SQL statements (such as those generated as part of compiled class code and those included in system code), the combination of the audit event and the preprocessor directive allows you to be selective in defining which embedded SQL statements to audit.

Additional notes:

- The `#sqlcompile audit = ON` directive on an INSERT, UPDATE, or DELETE statement does not cause the embedded SQL code in any trigger to be audited. To audit a nested SQL statement, you must include an additional `#sqlcompile audit = ON` directive in the nested code. For example, if trigger code contains embedded SQL, there must be a `#sqlcompile audit = ON` directive in that trigger code.
- The results of the audited statement are not recorded.

You can audit all embedded SQL statements *except*:

- %BEGTRANS
- %CHECKPRIV
- %INTRANS
- %INTRANSACTION
- COMMIT
- GET
- ROLLBACK
- SAVEPOINT
- SET OPTION
- STATISTICS

4.3 %System/%Security/DBEncChange

A process generates a %System/%Security/DBEncChange event because of:

- Encryption key activation
- Encryption key deactivation
- Encryption key and key file creation
- Encryption key file modification
- Encryption settings modification, such as enabling interactive database encryption activation at startup.

The EventData includes data relevant to the event, such as the encryption key ID and key file or a key file administrator name.

4.4 %System/%Security/X509CredentialsChange

For create or update operations, the event data lists the changed properties, subject to security considerations. For *Subject Key Identifier* and *Thumbprint*, the event data is a hexadecimal string of space-separated one-byte words; for *Certificate*, *PrivateKey*, *PrivateKeyPassword*, and *PrivateKeyType*, there is no event data.

For delete operations, there is no event data.

4.5 %System/%System/DatabaseChange

A process generates a %System/%System/DatabaseChange because of any of the following changes to a database:

- Creation
- Modification
- Mounting
- Dismounting
- Compaction
- Truncation
- Global compaction
- Defragmentation

For creation and modification, changes to the following properties cause auditing events (which are included in the event data):

- BlockSize (Create only)
- ClusterMountMode (Cluster systems only)
- ExpansionSize
- GlobalJournalState
- MaxSize
- NewGlobalCollation
- NewGlobalGrowthBlock
- NewGlobalIsKeep

- NewGlobalPointerBlock
- ReadOnly
- ResourceName
- Size

For mounting and dismounting, the event data records the database that was mounted or dismounted. For compaction, truncation, global compaction, and defragmentation, the event data includes include the parameters that the user selected.

4.6 %System/%System/RoutineChange

A process generates a %System/%System/RoutineChange event because a routine has been compiled or deleted. When enabled, this event causes a record to be written to the audit log whenever a routine or class is compiled. The Description field of the audit record includes the database directory where the modification took place, what routine or class was modified, and the word “Deleted” if the routine was deleted.

InterSystems IRIS audits events on the local server but not for associated instances. For example, if one instance of InterSystems IRIS is an application server that is associated with another instance that is a database server, creating and compiling a new routine on the application server is not audited on the database server, even if the RoutineChange audit event is enabled on the database server. To create a comprehensive list of all changes on all associated instances, enable the relevant events on all the instances and combine their audit logs.

5 Manage User-Defined Audit Events

This section includes the following topics:

- About User-Defined Audit Events
- Create a User-Defined Audit Event
- Add an Entry to the Audit Log
- Delete a User-Defined Audit Event

For information on enabling or disabling a user-defined audit event, see Enable or Disable an Audit Event.

5.1 About User-Defined Audit Events

In addition to system events, InterSystems IRIS allows you to create custom events that your application can add to the audit database. These are known as *user-defined audit events* or *user audit events*.

All currently defined events are listed on the **User-Defined Audit Events** page (**System Administration** > **Security** > **Auditing** > **Configure User Events**).

5.2 Create a User-Defined Audit Event

For InterSystems IRIS to audit a user-defined event, it must be added to the list of events and then enabled. The procedure is:

1. In the Management Portal, go to the **User-Defined Audit Events** page (**System Administration** > **Security** > **Auditing** > **Configure User Events**).
2. Click **Create New Event**. This displays the **Edit Audit Event** page.

3. On this page, enter values in the **Event Source**, **Event Type**, **Event Name**, and **Description** fields where these components have the purposes described in Elements of an Audit Log Entry.
4. By default, the **Enabled** check box on this page is selected. Click it to disable the event.
5. Click the page’s **Save** button to create the event.
6. Make sure that auditing is enabled.
7. Once the event is defined and auditing is enabled, you can add the event to the audit log by executing the following command:

   ```
   Do $SYSTEM.Security.Audit(EventSource,EventType,Event,EventData,Description)
   ```

   using the *EventSource*, *EventType*, *Event*, and *EventData* values that you defined in the Portal. For more details, see Add an Entry to the Audit Log.

5.3 Add an Entry to the Audit Log

Applications can add their own entries to the audit log with the **$SYSTEM.Security.Audit** function:

```
Do $SYSTEM.Security.Audit(EventSource,EventType,Event,EventData,Description)
```

where *EventSource*, *EventType*, *Event*, *EventData*, and *Description* are as described in Elements of an Audit Log Entry. Both the *EventData* and *Description* arguments can hold variables or literal values (where strings must appear in quotation marks). InterSystems IRIS provides all other elements of the log item automatically.

The content of *EventData* can span multiple lines. Its content is processed in a manner similar to the argument of the ObjectScript Write command, so it uses the following form:

```
"Line 1"_$Char(13,10)_"Line 2"
```

In this case, the content listed in the Audit Detail is displayed as “Line 1”, then `$Char(13,10)` is a carriage return and line feed, then there is “Line 2”.

For example, a medical records application from XYZ Software Company might use values such as:

```
 $SYSTEM.Security.Audit(
     "XYZ Software",
     "Medical Record",
     "Patient Record Access",
     765432,
     "Access to medical record for patient 765432"
     )
```

Note that the application uses the EventData element to record the ID of the patient whose record was accessed.

Further, if there is an “XYZ Software/Record Update/Modify Assignment” event defined and enabled, then the following code changes the value of a user-selected element of a list and notes the change in the audit database:

ObjectScript

```
For i=1:1:10 {
    Kill fVal(i)
    Set fVal(i) = i * i
}

Read "Which field to change? ",fNum,!
Read "What is the new value? ",newVal,!
Set oldVal = fVal(fNum)
Set fVal(fNum) = newVal
Set Data = "Changed field " _ fNum _ " from " _ oldVal _ " to "_ newVal _ "."
Set Description = "Record changed by user with an application manager role"
Do $SYSTEM.Security.Audit(
    "XYZ Software",
    "Record Update",
    "Modify Assignment",
    Data,
    Description
)
Write "Field changed; change noted in audit database."
```

Audit returns 1 or 0 to indicate that the addition succeeded or failed.

No privilege is required to add an entry to the audit log.

5.4 Delete a User-Defined Audit Event

To delete a user event:

1. From the Management Portal home page, go to the **User-Defined Audit Events** page (**System Administration** > **Security** > **Auditing** > **Configure User Events**).
2. On this page, locate the event that you wish to enable or disable and select **Delete** from the column near the right-hand part of the table.
3. When prompted, confirm that you wish to delete the event.

Note: If you delete a user-defined audit event, it is no longer available as part of the InterSystems IRIS instance for auditing.

6 Enable or Disable an Audit Event

To enable or disable an audit event:

1. From the Management Portal home page, go to either:
    - The **System Audit Events** page (**System Administration** > **Security** > **Auditing** > **Configure System Events**).
    - The **User-Defined Audit Events** page (**System Administration** > **Security** > **Auditing** > **Configure User Events**).
2. On the **System Audit Events** or the **User-Defined Audit Events** page, locate the event that you wish to enable or disable and select **Change Status** from the right-most column of the table. This changes the **Enabled** status from **No** to **Yes**, or vice versa.

7 Manage Auditing and the Audit Database

When events are logged, they are visible in the audit database, IRISAUDIT. The audit database also contains general information, including the name of the server, the name of the InterSystems IRIS configuration, when the log was started, and when the log was closed.

The following actions are available for managing the audit log:

- View the audit database
- Copy, export, and purge the audit database
- Encrypt the audit database
- General management functions

7.1 View the Audit Database

To view the audit database:

1. Select **View Audit Database** from the **Auditing** menu, which displays the **View Audit Database** page (**System Administration** > **Security** > **Auditing** > **View Audit Database**).
2. To refine the search, use the fields in this page's left pane and select the **Search** button at the bottom of the pane. (Select **Reset Values** at the bottom of the pane to restore the default values.) See below for a list of fields for refining your search.
3. To see more detailed information about a particular audit event, click the **Details** link in its row.

To refine the search, the fields are:

- **Event Source** — The component of the instance that is the source of the event.
- **Event Type** — Any categorizing information for the event.
- **Event Name** — The identifier of the event being logged (also known simply as the Event).
- **System IDs** — An identifier for the instance that appears in each audit log entry of the form *machine_name*:*instance_name*. For example, an instance called *MyInstance* running on a machine called *MyMachine*, then its system ID is *MyMachine:MyInstance*.
- **PIDs** — The operating-system ID of the process that logged the event.
- **Users** — The user who performed the activity that triggered the event.
- **Authentications** — How the user who triggered the audit event was authenticated to the instance.
- **Begin Date/Time** — The date and time for the first event to be displayed (midnight at the beginning of the current day, by default). To choose a starting date from a calendar, click the calendar icon to the right of the field.
- **End Date/Time** — The date and time for the last (most recent) event to be displayed (the current time, by default). To choose an end date from a calendar, click the calendar icon to the right of the field.
- **Maximum Rows** — The maximum number of rows to display in a listing of the audit log (up to 10,000).

Note: For fields with an arrow to the right, click the arrow to display a list of all the values in use. For fields that display an initial asterisk ("*"), choose or enter the asterisk to display all possible values for the field.

For background information on the fields displayed, see Elements of an Audit Event.

7.2 Copy, Export, and Purge the Audit Database

The audit log is stored in the %SYS.Audit table in the %SYS namespace; all audit data is mapped to the IRISAUDIT database and protected by the **`%DB_IRISAUDIT`** resource. By default, the %Manager role holds the **`Read`** permission on this resource and no role holds the **`Write`** permission.

The audit log database is managed with the same tools as other InterSystems IRIS databases. For example, you can use the Management Portal to specify its initial size, growth increment, and location. To help avoid losing audit events, we intentionally disallow the audit log database from having a specified maximum size. However the database is still constrained by disk space and other such factors. Please note, if you try to set the maximum size of the audit log database while auditing is disabled, it will appear to allow you to do so, but when you subsequently enable auditing, the maximum size will revert to a setting of 0, indicating no maximum size.

The Management Portal allows you to perform special management operations on the audit database:

- Copying — You can copy entries for one or more days to a specified namespace.
- Exporting — You can export entries for one or more days from the log to a file.
- Purging — You can remove entries for one or more days from the log.

Note: All these operations act on all entries for one or more days. There are no operations for particular entries.

7.2.1 Copy the Audit Database

InterSystems IRIS allows you to copy all or part of an audit database to a namespace other than IRISAUDIT. To do this:

1. From the Management Portal home page, go to the **Copy Audit Log** page (**System Administration** > **Security** > **Auditing** > **Copy Audit Log**).
2. On the **Copy Audit Log** page, first select either:
    - **Copy all items from the audit log**
    - **Copy items that are older than this many days from audit log** In the field here, enter a number of days; any item older than this is copied to the new namespace.
3. Next, use the drop-down menu to choose the namespace where you wish to copy the audit entries.
4. If you wish to delete the audit items after they are copied, select the check box with that choice.
5. Click **OK** to copy the entries.

InterSystems IRIS places the selected audit log entries in the *^IRIS.AuditD* global in the selected namespace. To view this data:

1. From the Management Portal home page, go to the **Globals** page (**System Explorer** > **Globals**).
2. From the **Globals** page, select the following items in the following order:
    a. The **Databases** radio button from the upper left area of the page.
    b. The name of the database holding the copied audit log entries.
    c. The **System** check box that appears above the list of globals.

    This displays a list of globals in the database, including *^IRIS.AuditD*. Globals are listed without the preceding “^” character that is needed to manipulate them programmatically or in the Terminal.

Note: Clicking **View Globals** on this page refreshes the page but unchecks in the **System** check box, thereby making *^IRIS.AuditD* unavailable.

3. Click **Data** from the IRIS.AuditD line to display detailed information on the audit log entries.

Once you have copied audit data to another namespace, you can use the queries of the %SYS.Audit class to look at that data.

7.2.2 Export the Audit Database

InterSystems IRIS allows you to export all or part of an audit database. To do this:

1. From the Management Portal home page, go to the **Export Audit Log** page (**System Administration** > **Security** > **Auditing** > **Export Audit Log**).
2. On the **Export Audit Log** page, first select either:
   - **Export all items from the audit log**
   - **Export items that are older than this many days from audit log** In the field here, enter a number of days; any item older than this is exported to the new namespace.
3. Next, in the **Export to file** field, enter the path of the file where you wish to export the audit entries. If you do not enter a full path, the root for the path provided is *install-dir*/Mgr/.
4. If you wish to delete the audit items after they are exported, select the check box with that choice.
5. Click **OK** to export the entries.

7.2.3 Purge the Audit Database

InterSystems IRIS allows you to purge all or part of a database.

Important: Purging the database is not a reversible action — purged items are permanently removed. You cannot restore items to the audit database once you have purged them.

To do this:

1. From the Management Portal home page, go to the **Purge Audit Log** page (**System Administration** > **Security** > **Auditing** > **Purge Audit Log**).
2. On the **Purge Audit Log** page, first select either:
   - **Purge all items from the audit log**
   - **Purge items that are older than this many days from audit log** In the field here, enter a number of days; any item older than this is purged.
3. Click **OK** to purge the entries.

7.3 Encrypt the Audit Database

InterSystems IRIS allows you to encrypt the database that holds the audit log. This is described in Configure Encryption Startup Settings.

7.4 General Management Functions

Because the audit log is stored in a table, you can manage it with standard InterSystems IRIS system management tools and techniques:

- Journaling is always turned on for it.
- You can use standard ObjectScript commands to read it. In addition, its contents are accessible via standard SQL and you can use any standard SQL tool to work with it.
- You can back it up using standard InterSystems IRIS database backup facilities.
- If it becomes full, a <FILEFULL> error occurs and is handled in the same way as for any other InterSystems IRIS database. To avoid this situation, see Maintain the Size of the Audit Database

Note: All access is subject to standard security restrictions at the database and namespace levels, or through SQL for table-based activity.

The %SYS.Audit table in the %SYS namespace holds the audit log. All audit data is mapped to the IRISAUDIT database. (You can also copy audit data to any other database using the functionality described in Copy the Audit Database; you can then use the %SYS.Audit class, which is available in every namespace, to query the audit log.)

7.4.1 Maintain the Size of the Audit Database

As InterSystems IRIS runs, it writes to the audit log. Without intervention, this will eventually fill the audit database. If the audit database becomes full, then InterSystems IRIS either continues running without capturing audit entries or halts until it can write to the audit database; the Freeze system on audit database error setting determines this behavior.

To properly store audit information and prevent any issues, you should regularly export and save the contents of the audit database and then purge its contents. To do this:

1. Export the contents of the audit database as described in Export the Audit Database.

   Note: InterSystems recommends that you export all entries from the database.

2. Check that the exported contents of the audit database are valid.

   Important: InterSystems recommends that you confirm that this data is valid, as purging the data is a non-reversible action.

3. Purge old entries from the existing database as described in Purge the Audit Database.

   Important: InterSystems recommends that you purge all entries except those of the last day, which ensures that there is an overlap in the different groups of saved entries.

CAUTION: If the audit database becomes full and InterSystems IRIS continues running, it does not record audit entries for actions that cause audit events. Further, in a forensic context, the existence of only a single AuditRecordLost audit entry indicates that *at least* one record was lost.

8 Other Auditing Issues

This section covers the following topics:

- Freeze the System If It Is Impossible to Write to the Audit Database
- Audit Event Counters

8.1 Freeze the System If It Is Impossible to Write to the Audit Database

During operations of InterSystems IRIS, it may become impossible to write to the audit database. This can happen due to a filled disk, a failed network connection, or some other reason. If this occurs, InterSystems IRIS can then either:

- Generate an error and continue running (the default).
- Freeze the system.

To modify this behavior:

1. Go to the **System-wide Security Parameters** page (**System Administration** > **Security** > **System Security** > **System-wide Security Parameters**). On this page, if auditing is enabled, the **Freeze system on audit database error** check box is available.
2. Select or clear the **Freeze system on audit database error** check box.
3. Click the page's **Save** button.

For example, suppose the audit database fills up. Any attempt to write to the audit log will generate a <FILEFULL> error (disk full). The difference between the behaviors is:

- When generating an error and continuing to run (the default) — The process does not write the audit record to the audit log; the audit record is therefore lost. When the problem is resolved, an entry is written into the audit log that lists how many audit events were lost.
- When freezing the instance if there is an error — The process writes the error message to the messages.log file; the system then freezes.

8.1.1 Tips on Recovering from Audit Log Errors

To recover from a disk full error, force down the system, free up space on the audit disk, then restart the system.

To recover from an error caused by database corruption, delete or move the audit database; then create a new audit database or copy a new one into the old one's place. (To clear the error, you must use a new database rather than simply restarting the system because restarting may write audit records, which will cause the system to freeze again.)

8.2 Audit Event Counters

To facilitate security monitoring, InterSystems IRIS keeps a counter for each audit event type and makes these counters available via the InterSystems IRIS monitoring interface. These counters are maintained even if auditing is not enabled. As an example, a site might monitor the LoginFailure event counter, to help detect break-in attempts.

Note: Audit counters are reset when the instances is restarted.

8.2.1 Reset the Counters for a System Audit Event

To reset the counters for a system event:

1. From the Management Portal home page, go to the **System Audit Events** page (**System Administration** > **Security** > **Auditing** > **Configure System Events**).

2. On this page, locate the event that you wish to enable or disable and select **Reset** from the column near the right-hand part of the table.
3. When prompted, click **OK**. This resets both the **Total** and **Written** counters for the event.

8.2.2 Reset the Counters For a User-Defined Audit Event

To reset the counters for a user event:

1. From the Management Portal home page, go to the **User-Defined Audit Events** page (**System Administration** > **Security** > **Auditing** > **Configure User Events**).
2. On this page, locate the event that you wish to enable or disable and select **Reset** from the column near the right-hand part of the table. This resets both the **Total** and **Written** counters for the event.
3. When prompted, click **OK**. This resets both the **Total** and **Written** counters for the event.