Article
Anssi Kauppi · Jun 30, 2020 3m read

Replicating Audit Log Near Real Time

Many organisations implement centralised log management systems to separate and centralise the log data in order to e.g. automate threat detection (and response) and to comply with regulatory requirements. The primary systems of interest are the various user facing applications, but increasingly also other kinds of systems including integration platforms.

What comes to the data platform (not the user facing application) most of the the events of interest/required are available as predefined system audit events in audit log - out of the box. They just need to be enabled. The rest are generally easy to implement as (custom) user audit events. The question I am addressing in this article is: how do you implement detecting and replicating new audit events near real time.

The Documented Methods Are Not Optimal For This Purpose ...

The audit log is stored in the %SYS.Audit table accessible in %SYS namespace. The documentation (specifically **Security Administration Guide**) lists two methods for accessing it: 1: exporting all or part of the audit database to file, and 2: copying all or part of the audit database to another namespace (and then querying it using class %SYS.Audit).

Exporting the audit log entries to a file would be obvious choice ***if*** the file would be OK as the integration mechanism ***and*** the target system would be able to read the events in the format exported by IRIS Data Platform (or Ensemble/Caché). Even if so, the requirement of replicating the events near real time makes this approach far from optimal however. I would prefer programmatic access to audit events to be able to use all the interoperability features of IRIS Data Platform (or Ensemble) to integrate to the external log system the optimal way.

In order for your program to get access to audit events, the documentation instructs to copy the audit events of interest to a user namespace and then use the capabilities of class %SYS.Audit to access the audit events. In addition to doing the copy using the management portal, you can invoke method Copy defined in class %SYS.Data programmatically. This would definitely do, if you would not need to do the copying too often.

The audit log is readily available in namespace %SYS, but implementing the user code for replication in namespace %SYS is something that I just wouldn't want to do.

Using SQL Inbound Adapter

This came as a surprise to me. I had been using SQL Inbound Adapter many times in order to integrate to external systems, but just didn't came to think about it as a means to access data available in a different namespace of (the same instance of) IRIS Data Platform until WRC suggested using it!

Implementing a business service using SQL Inbound Adapter for querying new audit events is straightforward. As an additional benefit, filtering the kinds of events of interest may be defined in the SQL Statement which is a configuration setting of SQL Inbound Adapter. What comes to the near real time requirement, replicating the new audit events e.g. every 60 seconds is feasible using this method.

In addition to replicating audit events, SQL Inbound Adapter is one possible means for accessing data in another namespace in general.

Other Possible Methods

You could map map global %SYS.Audit (IRIS) or %SYS.CacheAudit (Ensemble/Caché) to user namespace and read that global directly. If the requirement for real timeliness were seconds rather than minute(s), I would consider this alternative. After mapping the global, implementing a business service class using Ens.InboundAdapter to invoke the service code periodically is straightforward.

Example Code

I did some (prototypes of a) business service classes and an interoperability message class (Ensemble message). However, I am not attaching the code here as it is not finished nor properly tested yet (waiting for our customer to return from summer holiday). But if you want to have a copy of them as they currently are, please just let me know.

#Data Import and Export #Interoperability #JDBC #ODBC #Security #System Administration #Tips & Tricks #Ensemble #InterSystems IRIS #InterSystems IRIS for Health