By Dawn May

Tweet

RSS

Recent Posts

Meet the PowerHA Family
11/01/2017

PowerVM LPM/SRR Automation Tool
10/23/2017

STACK_INFO IBM i Service
10/17/2017

PowerVM Live Partition Mobility and Simplified Remote Restart
10/03/2017

System Monitoring for HTTP
09/26/2017

Previous Post | See All Posts

IBM i Embraces Syslog

November 14, 2017

This blog post has been guest-authored by Robert Andrews. Robert is a Managing Consultant in the IBM Systems Lab Services team specializing in security and high-performance computing. He can be reached at robert.andrews@us.ibm.com.

Syslog is a very popular reporting system that runs on many devices and OSes. It uses various parts and programs to encode, transmit, consolidate, and analyze messages from a wide range of devices. Everything from Windows and UNIX to firewalls and IoT devices participate in sending and centralizing messages from across the enterprise. That is, except for IBM i. Of course, we always must be different (usually for good reasons). However, when a security team wants to have an enterprise-wide view of events with a tool such as a SIEM (Security Information and Event Management), we sometimes feel left out. Until now!

With the release of IBM i 7.3 Db2 for i PTF Group level 7 and IBM i 7.2 Db2 for i PTF Group level 19 on October 27, 2017, IBM i now has methods to generate Syslog formatted messages for both the Audit Journal (QAUDJRN) and the History Log (QHST). With the assistance of Db2 for i Table Functions, these sources of information can be translated from their native IBM i format to that of either RFC3164 (older) or RFC5424 (newer, preferred) formats. You can find out more about the exact syntax of the functions at Db2 for i generated syslog history and audit journal.

Each Syslog message contains a header and an event message. The header contains information such as the priority (the lower the number, the more urgent or severe), date, time, and originating system. Keep in mind most Syslog messages are sent from all end points to one, centralized repository (SIEM) for analysis and notification. The event message itself uses another format called CEF, or Common Event Format. This message contains the OS level, source (Audit Journal or History Log), message name (Journal Code and Type or Message ID), severity, and detailed key-value pairs. These key-value pairs are items such as “msg=User BADUSR name not valid” or “sproc=123456/QTCP/QTFTP00001” (where sproc means source process—aka IBM i three-part job name).

If you want to try generating these messages yourself, first make sure you have the Db2 PTF Group listed above applied to your system. Then from any SQL Interface (have you tried the latest Run SQL Scripts inside IBM i Access Client Solutions yet?), run a statement such as:

```
SELECT syslog_facility, syslog_severity, syslog_event
FROM TABLE (QSYS2.DISPLAY_JOURNAL('QSYS', 'QAUDJRN',
    GENERATE_SYSLOG =>'RFC5424') ) AS X
    WHERE syslog_event IS NOT NULL;
```

Now, not every Audit Journal entry translates to a Syslog message. Hence, we added the WHERE clause to only get those entries that generate a Syslog event. You should see one row per message in your Audit Journal. Use the information above to try to understand the message. Keep in mind it is the centralized Syslog repositories job to read these messages, not you, so don’t feel bad if you don’t understand it.

Getting information from the History Log is just as simple, but uses a different Db2 Table Function. For the History Log, all messages are translated to Syslog events. Again, using the WHERE clause,

Links

IBM i Knowledge Center
IBM i developerWorks
IBM i
IBM i Resources
Performance Management on IBM i
Upgrade Planning and Future
Software Support
COMMON
Blog of Blogs

we can limit what entries from the History Log are converted.

```
SELECT syslog_facility, syslog_severity, cast(syslog_event as varchar(2048) CCSID 37)
  FROM TABLE (QSYS2.HISTORY_LOG_INFO(GENERATE_SYSLOG =>'RFC3164') ) AS X
    WHERE cast(syslog_event as varchar(2048) CCSID 37) LIKE '%QSECOFR%'
```

Here we used a LIKE statement to only get messages that refer to QSECOFR. In addition, we cast the syslog_event field as a character with CCSID of 37. By default, the message is generated and sent as UTF-16 Unicode. Some native IBM i programs may have trouble processing or displaying Unicode. This is an easy way to translate the data to EBCDIC.

Finally, now that we have the information from IBM i in the proper Syslog format, we need to send those messages to our centralized Syslog repository. Your security team should be able to give you the information about what system to send the messages to. You can use the built-in PASE command “logger” to send a message to your Syslog repository. To configure the logger command on IBM i, see http://www-01.ibm.com/support/docview.wss?uid=nas8N1013082. Note that this document was written prior to this new support so the information about generating Syslog messages is old, but the method of transmitting them is still valid.

Posted November 14, 2017 | Permalink

Post a Comment

Note: Comments are moderated and will not appear until approved