

Release Notes for SAS® Fraud Management 4.1_M0, Hot Fix 6

Description	Component	Summary	Test Scenario
<p>Database connections close prematurely in an implementation where socket connections are used for interfacing with the Java OnDemand Scoring Engine (J-OSE).</p>	<p>UUSC</p>	<p>When you use a socket to interface with J-OSE, the database connection thread exits when the socket is closed. This problem occurs in the following scenarios:</p> <ul style="list-style-type: none"> • after an input interface socket is closed • after a score engine restart • after the retry of a transaction fails 	<p>After you apply the hot fix, the following actions occur when you socket connections:</p> <ul style="list-style-type: none"> • System initialization opens database connections before transactions are accepted. • When a socket is closed, the database connection is returned to the cache. • Performing a restart does not close database connections.
<p>When you delete a user variable, you might see the error <code>ArrayIndexOutOfBoundsException</code> in the J-OSE log.</p>	<p>RULES</p>	<p>If a user variable is deleted, it causes the variable segment to shrink in size, resulting in a malformed message. This behavior can cause for the following error during rule deployment or restart of the scoring engine:</p> <pre style="margin-left: 40px;">ERROR: 5306 transaction processor error during first wait for completion...java.util.concurrent.ExecutionException: java.lang.ArrayIndexOutOfBoundsException.</pre>	<p>After you apply the hot fix, deletion of user variables is correctly adjusted in the segment.</p>
<p>SAS sessions need to synchronize during the restart of J-OSE when the implementation has multiple J-OSE servers.</p>	<p>UUSC</p>	<p>In implementations with multiple J-OSE servers, the build number check does not handle transactions correctly. This problem happens when a restart occurs and one of the servers is slower to complete the restart.</p>	<p>After you apply the hot fix, the build number check is performed within the J-OSE servers, which are independent of one another so as to handle the transactions effectively.</p>