

CU*ANSWERS ITEM PROCESSING DISASTER RECOVERY TEST REVIEW

Event Dates: 1/11/2016 – 1/14/2016

Report Revision Date: 2/03/2016

SUMMARY

As part of a robust business continuity program, CU*Answers actively tests recovery plans to ensure validation of procedures for recovering critical processes and to identify opportunities to improve recovery efforts and minimize the impact of a disruption to the organization and its stakeholders.

During the week of January 11, 2016, select team members from the CU*Answers Item Processing department completed a disaster recovery test of the Electronic Check Processing environment by restoring CheckLogic Manager application and database servers and performing critical business functions from the secondary datacenter located in Muskegon, MI.

For the purpose of this test, data replication was suspended at the secondary datacenter to allow Item Processing staff to “replay” the events of a typical day to ensure the processes on the servers in the test environment performed as those in the production environment. This test was performed in parallel with the production environment with zero impact to clients. This was the first recovery test performed since migrating to the CheckLogic Manager platform in 2015. The test was performed by members of the Item Processing, Network Services, and eDOC support teams.

This report identifies the details of the test, challenges observed, lessons learned, and recommendations for consideration based on the results of this exercise.

EVENT REVIEW

The existing production CheckLogic environment includes data volumes hosted on a SAN (Storage Area Network) located at the primary facility with data replicated to a redundant SAN at the secondary facility. As mentioned above, data replication was suspended prior to the test window. Production application servers were cloned (virtualized) in a sandboxed environment at the secondary facility with drives mapped to data volumes on the backup SAN. Test servers were configured to communicate with other servers in the test environment by manually changing host lookup tables and application INI files.

On the morning of 1/11, Network Services team members stopped data replication for the CheckLogic environment and began preparing servers for the test environment. On 1/12, recovery teams participated in testing individual components of the CheckLogic application in the test environment in preparation for performing critical IP functions listed below. On 1/13 and 1/14, Item Processing team members performed critical functions

identified above for the purpose of this test. All identified functions were completed with the exception of generating daily reports.

The Item Processing business critical functions identified for this test included:

- Download Electronic Check Processing (ECP) files from FRB
- Import ECP files using Fed Admin
- Perform repairs on the rejected images
- Compare individual client totals and reports with FRB totals
- Generate and submit transmission files for online clients and each off-line client representing all delivery channels (CUAPROD, GoAnywhere/SFTP, etc.)
- Download chargeback files from FRB
- Process pay/no-pay decisions on chargebacks
- Generate redeposit files for chargebacks
- Print Image Replacement Documents (IRD) created for chargebacks
- Receive and balance online return file
- Create and submit stacked return file
- Generate daily reports

Item Processing Team participants operated from the primary production datacenter while accessing workstations in the recovery test environment at the secondary datacenter using remote access tools.

CHALLENGES

Many of the documented challenges below are the result of efforts to perform a recovery test parallel with the production environment (no downtime for clients). In an actual disaster recovery effort (recovering the production environment), most of these challenges would not exist.

1. When preparing to suspend SQL replication prior to preparing the test environment on Monday (1/11), it was learned that some objects had not replicated fully to the secondary server. The replication service was restarted and given time to complete, correcting the issue. With this delay, the test was postponed until the next day.
 - a. As a result, a monitor check has been added to issue an alert if these conditions should reoccur.
2. The initial attempt to import ECP files failed due to the application's use of both hostnames and network addresses at multiple points within the software used to locate and communicate with other servers for processing data.
 - a. Recovery test hosts are created by cloning production servers in a sandbox environment. The configuration files on the cloned hosts are then modified to allow communications with other test hosts while preventing communication with production hosts. Identifying all of the required changes throughout the application process for this initial test proved to be somewhat time consuming, with occasional moments of trial and error.
3. The initial Reject/Repair process failed due to the method the process is called within the application.
 - a. To resolve this, the processes were reference by <http://localhost/> instead of the production path. This allowed the Reject/Repair process to be performed successfully.

4. An iSweep service running on the “cloned” test servers was configured to start automatically on boot up to process queued redeposit files. During our initial test, these files were unintentionally processed and sent to production servers creating duplicate files in the production queues. This created the potential for double-posting on the production servers. A monitor application on the production server detected the duplicate files and alerted IP team members who promptly removed the duplicate files before they were processed.
 - a. To prevent this from occurring in future tests, procedures have been updated to reconfigure the iSweep service during the server setup process, setting the service to “manual”.
5. Several folder and system permissions on the servers at the secondary site (test environment) required modification to allow not only production servers but also test servers to read, create, write, and delete image files. This created a delay in this initial recovery test.
 - a. All system and application modifications for testing have been documented. We anticipate that tests will be completed more quickly and efficiently.
 - b. To validate the documentation, the next recovery test will be scheduled within the next six months and progress reported.

CONTINUING EFFORTS AND RECOMMENDATIONS

1. As expected, this initial recovery test of the CheckLogic Manager environment resulted in significant documentation changes. As a result, recovery teams gained new insight into the process flow and system interdependencies of the application. This recovery test experience will add value to not only in the software development process but also application support efforts.
 - a. For the next recovery test scheduled in June, 2016, recovery teams will build on the results on the January test and seek to expand the scope incrementally to further increase the benefit of the event and the confidence in the capability to recover in a timely manner should an actual disruption occur.
2. For the purpose of this recovery test, identified functions were limited to internal activities performed by IP staff.
 - a. Future recovery tests will include processes that are normally performed by external clients who access the application through the CheckLogic web server pool. While the processes will be tested, at this time access will still be limited to internal staff (to prevent unintentional crossover with the production environment).
 - b. Future recovery tests will also include the generation of daily reports.
3. One of the positive outcomes of this recovery exercise is the identification of all configuration settings that require alteration for the purpose of the test (or in the event of an actual recovery at the secondary datacenter). With all of the intricate moving parts of the CheckLogic application that work together to process check images, one incorrect configuration can cause the application test to fail. The knowledge and experience gained from this recovery test will help us enhance the product to further improve performance and reliability.
 - a. Recovery teams who participated in this test will compare the notes generated with existing product documentation to ensure support staff have all of the relevant information readily available in an effort to improve customer service.