## [AsyncDHGatewayService] State based Filter in SecureReturnEventHandler

### Problem Statement

Related ticket: [https://t.corp.amazon.com/V1050577692/](https://t.corp.amazon.com/V1050577692/communication) (recurring Sev3)

### Reason

This happens in cases of DELIVERED/CANCELLED statuses, Container TR is Updated which should not happen.

### Solution

Strategy # 1

Creating a state-filter for the identified statuses such that they do not undergo UpdateTR and TRMService ProcessPackageException call in SecureReturnHandlerManager

New FilterByState class

public class FilterByState {  
    final static Set<String> containerStatus = Sets.newHashSet(DELIVERED, CANCELLED);  
   
    public boolean shouldNotProcess(ContainerState tsmsContainerState) {  
   
        if(isNull(tsmsContainerState)){  
            log.info("TSMS ContainerState is Null");  
            return false;  
        }  
   
        String status = CommonUtil.chainWalk(() -> tsmsContainerState.getStateInformation().getStatus(), null);  
            if(containerStatus.contains(status)) {  
                LambdaMetrics.putMetric("FILTERED\_CONTAINER\_STATUS\_" + status, 1, Unit.COUNT, Aggregate.SUM);  
                log.info("Event filtered out had Container ID: {} and status: {}", tsmsContainerState.getContainerId(), status);  
                return true;  
            }  
        return false;  
    }  
}

Updated SecureReturnHandlerManager

ContainerIdentifiers containerIdentifiers = ContainerIdentifiers.builder()  
                .withContainerId(secureReturnModel.getContainerId())  
                .withTrackingId(secureReturnModel.getTrackingId())  
                .build();  
   
        ContainerState tsmsContainerState = tsmServiceAdapter.getState(containerIdentifiers);  
   
        if (filterByState.shouldNotProcess(tsmsContainerState)) {  
            return;  
        }

Related CRs: <https://code.amazon.com/reviews/CR-103547245>

<https://code.amazon.com/reviews/CR-104161537>

**UPDATE**

Strategy # 2  **[PREFERRED]**

Instead of filtering out DELIVERED/CANCELLED statuses we can instead process only WAITING\_FOR\_PICKUP status

Updated FilterByState class

public class FilterByState {  
  
    public boolean shouldProcess(ContainerState tsmsContainerState) {  
  
        if(isNull(tsmsContainerState)){  
            log.info("TSMS ContainerState is Null");  
            return true;  
        }  
  
        String status = CommonUtil.chainWalk(() -> tsmsContainerState.getStateInformation().getStatus(), null);  
            if(WAITING\_FOR\_PICKUP.equals(status)) {  
                LambdaMetrics.putMetric("PROCESSED\_CONTAINER\_STATUS\_WAITING\_FOR\_PICKUP", 1, Unit.COUNT, Aggregate.SUM);  
                log.info("Event processed had Container ID: {} and status: WAITING\_FOR\_PICKUP", tsmsContainerState.getContainerId());  
                return true;  
            }  
        LambdaMetrics.putMetric("FILTERED\_CONTAINER\_STATUS\_" + status, 1, Unit.COUNT, Aggregate.SUM);  
        return false;  
    }  
}  
  
  
  
/\* Flow will be reversed in SecureReturnHandler \*/  
  
if (**!**filterByState.shouldProcess(tsmsContainerState)) {  
            return;  
    }

Related CRs: [https://code.amazon.com/reviews/CR-105091080](https://code.amazon.com/reviews/CR-105091080/revisions/2#/reviewers)

<https://code.amazon.com/reviews/CR-105371734> (Minor UT’s fix)

### Deployment Pre-Requisite

* “Pending Acceptance” endpoints need to be manually accepted for TSMS Prod AWS Accounts in EU and NA regions to allow receiving requests from AsyncDHGatewayService.

Related MCM: <https://mcm.amazon.com/cms/MCM-89067870>

### Monitoring (post deployment)

* Validating lamba metrics for all statuses

### Challenges Faced

1. **Problem**: AsyncDHGatewayService was not sending requests to TSMService during gamma testing.                   **Reason**: Endpoints were not configured correctly in TSMS Gamma AWS Account for receiving requests from Async.           **Solution**: Manually configured endpoints to available status.