

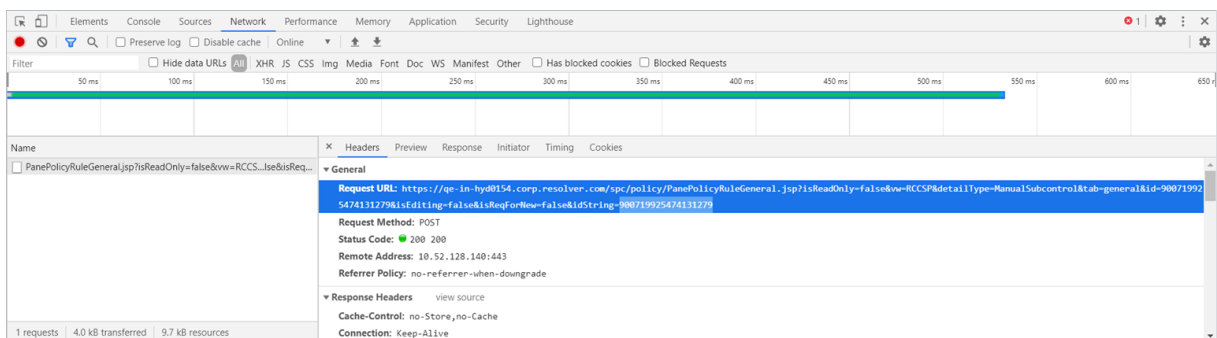
Fix Editing Errors for Imported Controls or Subcontrols

There have been reported instances in which users who attempt to edit the description of an imported control or subcontrol receive the following error message: **Error This [control or subcontrol] cannot be edited because it is a duplicate record. Please contact RiskVision Support for help in resolving this issue.**

In order to resolve this issue, follow the below steps:

To resolve this issue:

1. Navigate to the C:\Server\config folder.
2. Open the **agilience.properties** file and place the following properties:
 - `com.agilience.policy.ignoreDiffCopied=false`
 - `com.agilience.common.utils.idencryption.keylength=256`
 - `com.agilience.common.utils.idencryption.skipencryption=true`
 - `com.agilience.common.utils.idencryption.acceptPlainIdString=true`
3. Save and close the file.
4. Restart the RiskVision Tomcat service.
5. Log into RiskVision.
6. Go to **Content > Controls and Questionnaires**.
7. Navigate to the control or subcontrol that is returning the error message.
8. Press the F12 button on your keyboard to open the browser console and click on the **Network** tab.
9. In the RiskVision UI, navigate to the **General** tab of the control or subcontrol.
10. In the browser console, click on **PanePolicyControlDetail.jsp** for controls and **PanePolicyRuleGeneral.jsp** for subcontrols.
11. Copy the **IdString** value from the **Request URL**.



The IdString in the browser console.

12. Stop the RiskVision Tomcat service and connect to the database
13. Paste the IDString values for each control or subcontrol in the appropriate sections of the below queries,
 - For Controls:

```
----- POLICYSETS -----
```

```
-- Step 1 : Create Temporary table with list of identified policysset ids
```

```
DROP TABLE tmp_policysset_1;
```

```
CREATE TABLE tmp_policysset_1 AS
```

```

SELECT  policyset_id, permanent_id, title
FROM    agl_policyset
WHERE   policyset_id in
(
[inset comma separated list of the IdString values for all affected subcontrols]
);

SELECT * FROM tmp_policyset_1;

-- Step 2 : Create Temporary table with list of updateable policyset ids

DROP TABLE tmp_policyset_2;

CREATE TABLE tmp_policyset_2 AS
SELECT  p.policyset_id, p.name, p.title, p.user_label, p.permanent_id, p.last_update_time,
        case when p.policyset_id = t.policyset_id then 'N' else 'Y' end to_be_updated
FROM    agl_policyset p
        INNER JOIN tmp_policyset_1 t
        ON t.permanent_id = p.permanent_id
WHERE   (p.user_label is null or p.user_label = '');

COMMIT;

SELECT * FROM tmp_policyset_2;

-- Step 3 : Update agl_policyset table from Temp table #2

UPDATE  agl_policyset p
        INNER JOIN tmp_policyset_2 t
        ON t.policyset_id = p.policyset_id
SET     p.user_label = concat('0_user_label_', p.policyset_id)
WHERE   t.to_be_updated = 'Y';

COMMIT;

-- Step 4 : (MUST BE EXECUTED ONLY AFTER EVERYTHING IS VERIFIED)

-- DROP TABLE tmp_policyset_2;

-- DROP TABLE tmp_policyset_1;

```

- o For Subcontrols:

```

-- Step 1 : Create Temporary table with list of identified policy ids

DROP TABLE tmp_policy_1;

CREATE TABLE tmp_policy_1 AS
SELECT  policy_id, permanent_id, title
FROM    agl_policy
WHERE   policy_id in
(
[inset comma separated list of the IdString values for all affected subcontrols]
);

SELECT * FROM tmp_policy_1;

-- Step 2 : Create Temporary table with list of updateable policy ids

DROP TABLE tmp_policy_2;

CREATE TABLE tmp_policy_2 AS
SELECT  p.policy_id, p.name, p.title, p.user_label, p.permanent_id, p.ct_last_update_time,
        case when p.policy_id = t.policy_id then 'N' else 'Y' end to_be_updated
FROM    agl_policy p
        INNER JOIN tmp_policy_1 t
        ON t.permanent_id = p.permanent_id
WHERE   (p.user_label is null or p.user_label = '');

COMMIT;

SELECT * FROM tmp_policy_2;

-- Step 3 : Update agl_policy table from Temp table #2

UPDATE  agl_policy p
        INNER JOIN tmp_policy_2 t
        ON t.policy_id = p.policy_id
SET     p.user_label = concat('0_user_label_', p.policy_id)
WHERE   t.to_be_updated = 'Y';

```

```
COMMIT;
```

```
-- Step 4 : (MUST BE EXECUTED ONLY AFTER EVERYTHING IS VERIFIED)
```

```
-- DROP TABLE tmp_policy_2;  
-- DROP TABLE tmp_policy_1;
```

14. Copy and paste the above queries into the database and execute them.
15. Verify the details of the controls or subcontrols in RiskVision once the queries have been executed.