

Customizing Incident Management

RiskVision version 4.1 and later supports a hook that allows custom-scripted asset manipulation during the classification process.

The general interface for this hook is:

```
package com..survey.asset;

import com..common.ALException;

import com..dal.model.Asset;

public interface EntityScriptUpdater {

    /**
     * update the asset attributes
     * @param asset
     * @throws ALException
     */
    public void updateEntity(Asset asset) throws ALException;

}
```

The hook is controlled by these properties in the .properties file:

```
classification.post.script.enabled=true
```

```
classification.entity.update.groovy.source=file:D://main/config/CustomerAssetUpdater.groovy
```

The first property enables or disables the hook and the second property specifies the custom groovy script. The default implementation without specifying the groovy source is

```
classification.entity.update.groovy.source=classpath:com//risk/
```

DefaultRiskUpdater.groovy

A good use case for this functionality is to customize the incident risk score calculation after the incident is done with classification, as the implementation of incident classification is performed through a hidden asset and the asset is associated with the incident by `Asset#getIncident()` or `Incident#getAsset()`.

Here are the steps (these are all one-time steps):

1. Save the custom groovy file (based on the interface above) to a folder such as `server/config`.
2. Set the following properties in the `.properties` file, update based on your deployment:

3. `classification.post.script.enabled=true`

```
classification.entity.update.groovy.source=file:D://main/config /CustomerAssetUpdater.groovy
```

4. Restart the RiskVision Tomcat service, or click Reload in the Configuration section of **Administration > Server Administration** menu.