

Reporting

Fundamentals of IdentityIQ Implementation

IdentityIQ 7.0

Overview

Reporting

- Understand the reporting architecture of SailPoint IdentityIQ
- Learn about creating custom reports including
 - Defining Data Sources
 - Report Column Configuration
 - Query Configuration

Reporting Overview

- **Reports Tab**

- Report Templates
 - Used for creation of reports

- **My Reports Tab**

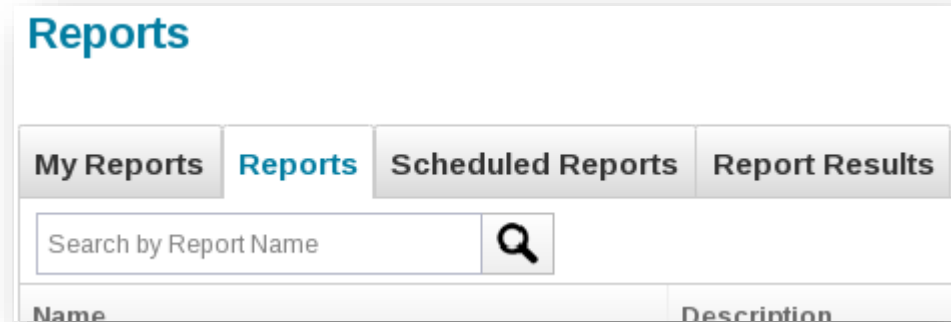
- Saved Reports configured from Report Templates
 - Represent saved configurations of report templates

- **Scheduled Reports Tab**

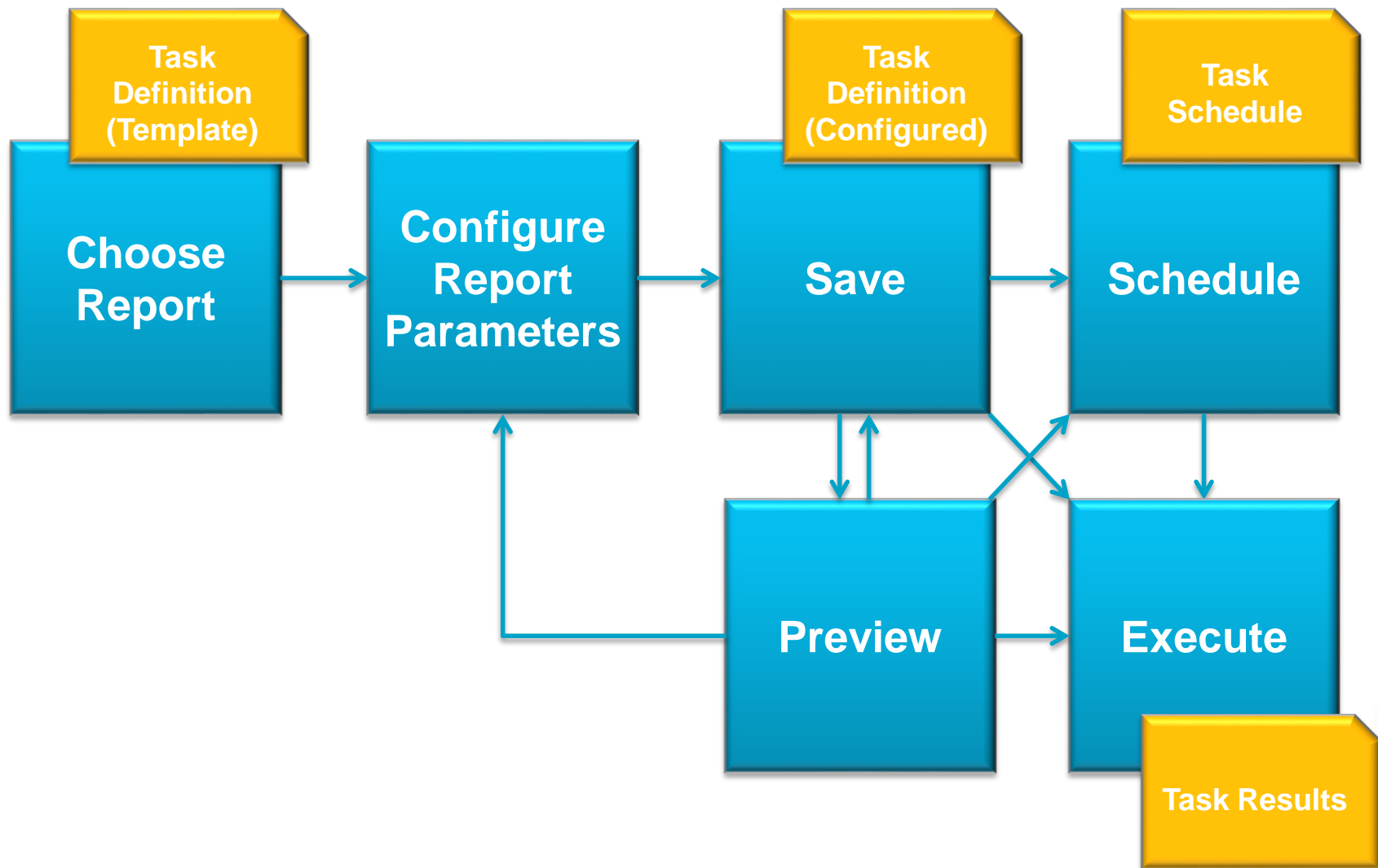
- Details information about report schedules

- **Report Results Tab**

- Results of ad-hoc or scheduled reports (Results and links to PDF/CSV)



General Reporting Flow



Configuration of Report Parameters

- Configuration specifies
 - Standard properties
 - Name
 - Description
 - Email recipients
 - Data filtering
 - By date range
 - By application
 - By manager
 - etc...
 - Column layout (adjustable in preview as well)
 - Which columns are visible
 - Sort by
 - Group by

Reporting Configuration

Sections

- All reports have three main sections

Edit Report

Summary

Sections:

1. Standard Properties
2. Policy Violation Properties
3. Report Layout

Report Properties

Standard Properties

Name*

Previous Result Action

Rename Old



Scope



Description

Displays information about all current policy violations in detailed format.

Email Recipients



Previous

Next

Save

Save and Preview

Save and Execute

Cancel

This section is common to all reports. It configures name, description, general functionality and email recipients.

Reporting Configuration

Report Specific

- Middle section is report specific

This middle section varies for each type of report. For custom report templates, this section is created using Form objects

Edit Report

Summary <<

Sections:

1. Standard Properties
2. Policy Violation Properties
3. Report Layout

Report Properties

Policy Violation Properties

Identities ?

Violation Activity ?
☐ All Violations ☐ Active Violations
☐ Inactive Violations

Edit Report

Summary <<

Sections:

1. Standard Properties
2. Identity Attributes
3. Identity Extended Attributes
4. Additional Identity Properties
5. Report Layout

Reporting Configuration

Layout

- Report Layout (Grid Control)

Sort By, Group By, Columns, Summary and Detail display options

Edit Report

Summary <<

Report Properties

Report Layout

Sort By
 ▼ ☒ Sort Ascending

Group By

Columns ?

↑

↑

→

←

↓

↕

First Name
Last Name
Identity
Policy
Violation Owner
Rule
Status
Summary

Disable Report Summary Display ?
☐

Disable Report Detail Display ?
☐

Developing Custom Reports

- Report definition is a specialized task
- Reports inherit IdentityIQ Task characteristics
 - Configuration (saving configured Task Definition objects)
 - Schedules (TaskSchedule objects)
 - Results (TaskResult objects)
- Report definitions define
 - Summary and Charting options
 - What object type to report on (Identity, Policy Violation, etc.)
 - Report Layout (Columns and ordering to display)
 - Query to retrieve supporting data
 - SailPoint Filter, Java Datasource, HQL

Developing Custom Reports

Implementation Details

- Report Executor

- Java class: `sailpoint.reporting.LiveReportExecutor`
- `LiveReportExecutor` uses the XML report definition encapsulated within the `TaskDefinition` to generate the report

- Creating New Custom Reports

- Existing Live Reports can be used as examples for custom reports
- Generate new Task Definition including
 - Configuration Options
 - Query to execute
 - Columns to retrieve

Report XML Overview

Policy Violation Report

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE TaskDefinition PUBLIC "sailpoint.dtd" "sailpoint.dtd">
<TaskDefinition created="1346692200540" executor="sailpoint.reporting.LiveReportExecu
  <Attributes>
    <Map>
      <entry key="report">
        <value>
          <LiveReport title="Policy Violation Report">
            <Chart category="status" groupBy="status,policyName" series="policyName"
            <DataSource objectType="PolicyViolation" type="Filter"> ☐ ☐ ☐ </DataSource>
            <ReportForm> ☐ ☐ </ReportForm>
            <Columns> ☐ ☐ </Columns>
            <ReportSummary title="rept_cert_summary_title"> ☐ ☐ </ReportSummary>
          </LiveReport>
        </value>
      </entry>
    </Map>
  </Attributes>
  <Description> ☐ ☐ </Description>
  <RequiredRights> ☐ ☐ </RequiredRights>
  <Signature> ☐ ☐ </Signature>
</TaskDefinition>
```

LiveReport

Chart Definition

DataSource Definition

Form Reference

Column Configuration

Report Summary

Data Query

- Start by thinking of the construct of a basic SQL statement
select <columns> from <table> where <criteria>;

Example:

select id, name from spt_identity where name = 'Adam.Kennedy';
columns *table* *criteria*

- In our reporting engine, the mappings for each item are:

<columns>	Column Configuration
<table>	SailPoint Object
<criteria>	Query (Defined as Filter, HQL or Java DataSource)

Example – Policy Violation Report

```
<LiveReport tit
  <Chart catego
    <DataSource d
    <ReportForm>
    <Columns> ...
    <ReportSummar
  </LiveReport>
```

- What to search for?

- Defined as objectType and type

`<DataSource objectType="PolicyViolation" type="Filter">`

- We will search for Policy Violation objects using a Filter

Example – Policy Violation Report

- What Query to Run?

- Query is configurable
 - Data Source
 - Query Parameters
 - Custom Form
- Parameters can be hardcoded, calculated or based on report arguments
- Default: return all objects of the type specified

```
<LiveReport tit
  <Chart catego
    <DataSource d
    <ReportForm>
    <Columns> ...
    <ReportSummar
  </LiveReport>
```

```
<DataSource objectType="PolicyViolation" type="Filter">
  <QueryParameters>
    <Parameter argument="identities" property="identity.id"/>
    <Parameter argument="status" property="active">
      <ValueScript>
        ...
      </ValueScript>
    </Parameter>
  </QueryParameters>
</DataSource>
```

Example – Policy Violation Report

Signature

The screenshot shows the 'Edit Report' interface. On the left, the 'Summary' tab lists sections: 1. Standard Properties, 2. Policy Violation Properties, and 3. Report Layout. On the right, the 'Report Properties' tab is active, showing 'Policy Violation Properties'. The 'Identities' field is highlighted with a red box. An inset shows XML code with the <Signature> tag highlighted.

- Signature defines filters which can be specified by report user

<Signature>

<Inputs>

<Argument multi="true" name="identities" type="Identity">

<Description>rept_input_violation_report_identities</Description>

</Argument> ...

</Inputs>

</Signature>

- Signature arguments provide values for queries

<QueryParameters>

<Parameter argument="identities" property="identity.id"/>

Example – Policy Violation Report

- What columns to retrieve?
- Defined as Columns and ReportColumnConfig Items in XML
- Lists the items that we need to display defined by properties

```
<LiveReport tit
  <Chart catego
  <DataSource o
  <ReportForm>
  <Columns> ...
  <ReportSummar
</LiveReport>
```

<Columns>

<ReportColumnConfig field="firstname" header="FirstName" property="identity.firstname" sortable="true" width="110"/>

<ReportColumnConfig field="lastname" header="LastName" property="identity.lastname" sortable="true" width="110"/>

<ReportColumnConfig field="name" header="Identity" property="identity.name" sortable="true" width="110"/>

...

Report Data				
First Name	Last Name	Identity	Policy	Violat
James	Smith	James.Smith	TRAKK SOD Policy	The Ac

Example – Policy Violation Report

ReportForm (Optional)

- Defines form to use for presentation of report filtering options

<ReportForm>

```
<Reference class="sailpoint.object.Form" id="ff8080814528eeb90854981201b4"  
  name="Policy Violation Report Form"/>
```

</ReportForm>

```
<LiveReport tit  
<Chart catego  
<DataSource o  
<ReportForm>  
<Columns> ⚙ <  
<ReportSummar  
</LiveReport>
```

Edit Report

Summary

Sections:

1. Standard Properties
2. Policy Violation Properties
3. Report Layout

Report Properties

Policy Violation Properties

Identities ?

Violation Activity ?

- ☐ All Violations ☐ Active Violations
☐ Inactive Violations

Violation Status ?

Policy ?

Violation Date ?

Reporting Option

External Business Intelligence (BI) Tools

- Process
 - Replicate the database (or a portion of the database)
 - Use off-the-shelf BI tools to query the data
- IIQ Console command
 - *iiq exportviews*
 - Creates views for exported data (supported mappings for extended attributes)
 - Example: “extended1” becomes “division” in view

Note: Data Export task will remain for backwards compatibility.

Questions?

Exercise Preview

Section 3, Exercises 7

- Exercise 7: Creating and Extending a Custom Report
 - Load custom report from XML file
 - Investigate the report and modify its layout
 - Extend the report to display additional data
- Optional Extension Exercises
 - Extend the report using signature and forms
 - Extend the report to limit returned data

Report Exercise Preview

- </home/spadmin/ImplementerTraining/config>
 - Report-CustomCapabilities.xml
 - **Initial custom report – no parameters**
 - **Provides all identities and their capabilities**
 - Report-CustomCapabilities2.xml
 - Extended report
 - Added “Region” and “Location”
 - Report-CustomCapabilities3.xml
 - Signature example
 - Filter by “Identity” and “Manager”
 - Report-CustomCapabilities4.xml
 - Form example
 - Format the data input section: title and format
 - Uses Report-CapabilitiesForm.xml
 - Report-CustomCapabilities5.xml
 - Added query option to show only capabilities with values