

SCOPED CERTIFIED APPLICATION DESIGN

IBM Resilient

Table of Contents

1	Overview	3
1.1	References.....	Error! Bookmark not defined.
2	Application overview and intended use	3
2.1	Third-party dependencies	3
3	Software architecture overview	4
4	Software design description	5
5	Performance considerations and concerns	Error! Bookmark not defined.
6	Security considerations and concerns.....	7
7	Application components	7
7.1	Components created by the application:	7
7.2	Integration components created by the application:.....	8
8	Future plans.....	9
9	Service Level Agreement Definition.....	9
10	Debugging and troubleshooting	9
10.1	Debugging tips:	9
10.2	Failure modes:.....	9

1 Overview

Your Design Document describes the design of your scoped application and will be used throughout the certification process. The document is used internally by ServiceNow only and is not shared with your customers.

To prepare your Design Document, enter a thorough response in each section of this template document. If a section is not applicable, state the reasons clearly.

This document is intended to enable ServiceNow Certification Engineers and Technical Support members to:

- Understand your integration completely
- Identify areas of risk to platform security or platform stability
- Troubleshoot any potential third-party issues

2 Application overview and intended use

The Application:

- Integrates the Incident Table in ServiceNow with IBM Resilient
- Records from the ServiceNow Incident Table can be used to create Incidents/Tasks in IBM Resilient
- Business Rules allow Work Notes and Additional Comments to be sent to a related IBM Resilient Incident/Task
- The application includes custom REST API Endpoints that allows IBM Resilient to create/update Records in the Incident Table

The Business Need:

- Allows a bi-directional integration between ServiceNow Incidents and IBM Resilient Incident/Tasks

Use Cases:

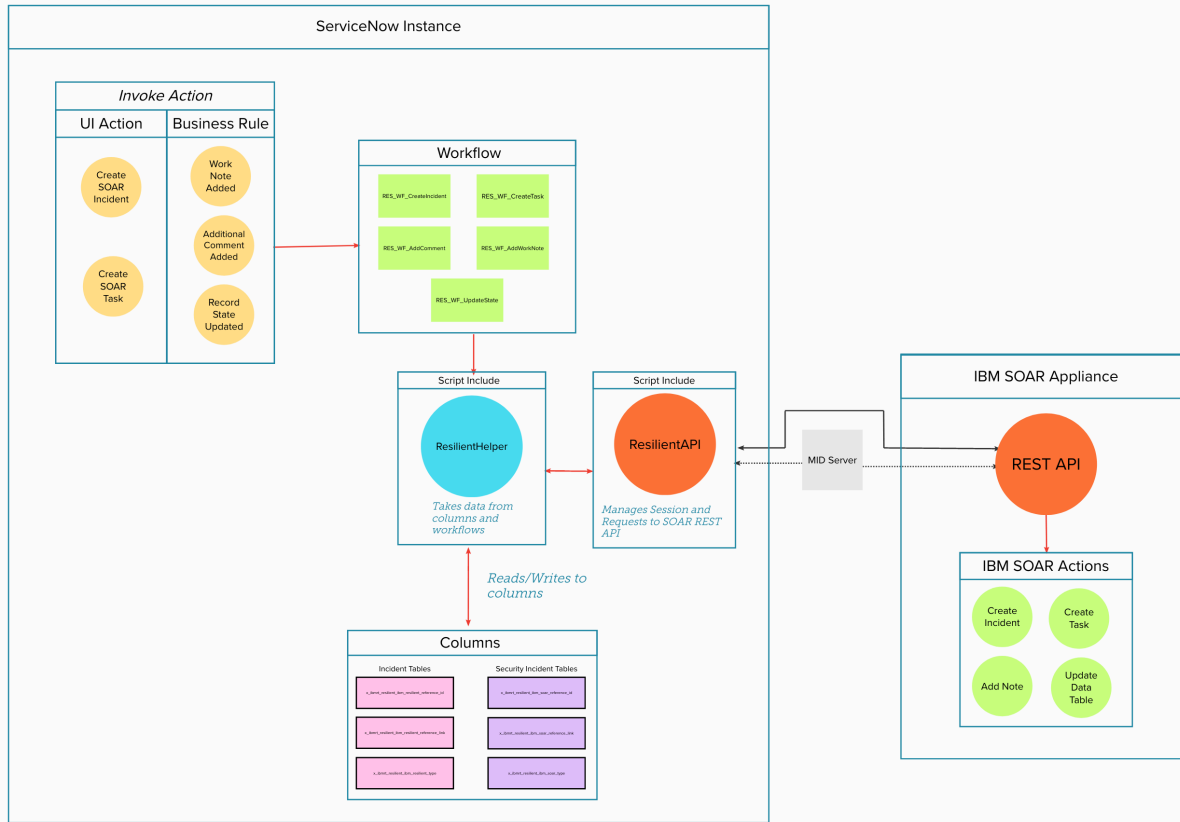
- A SOC Analyst needs assistance from another team that uses ServiceNow for their workflow. They can now easily send, synchronize and communicate an Incident/Task from IBM Resilient to a team using ServiceNow
- An Incident is raised in ServiceNow and the customer needs their SOC's assistance to respond to the Incident. They can now easily send, synchronize and communicate a ServiceNow Record in the Incident table to their IBM Resilient team

2.1 Third-party dependencies

None

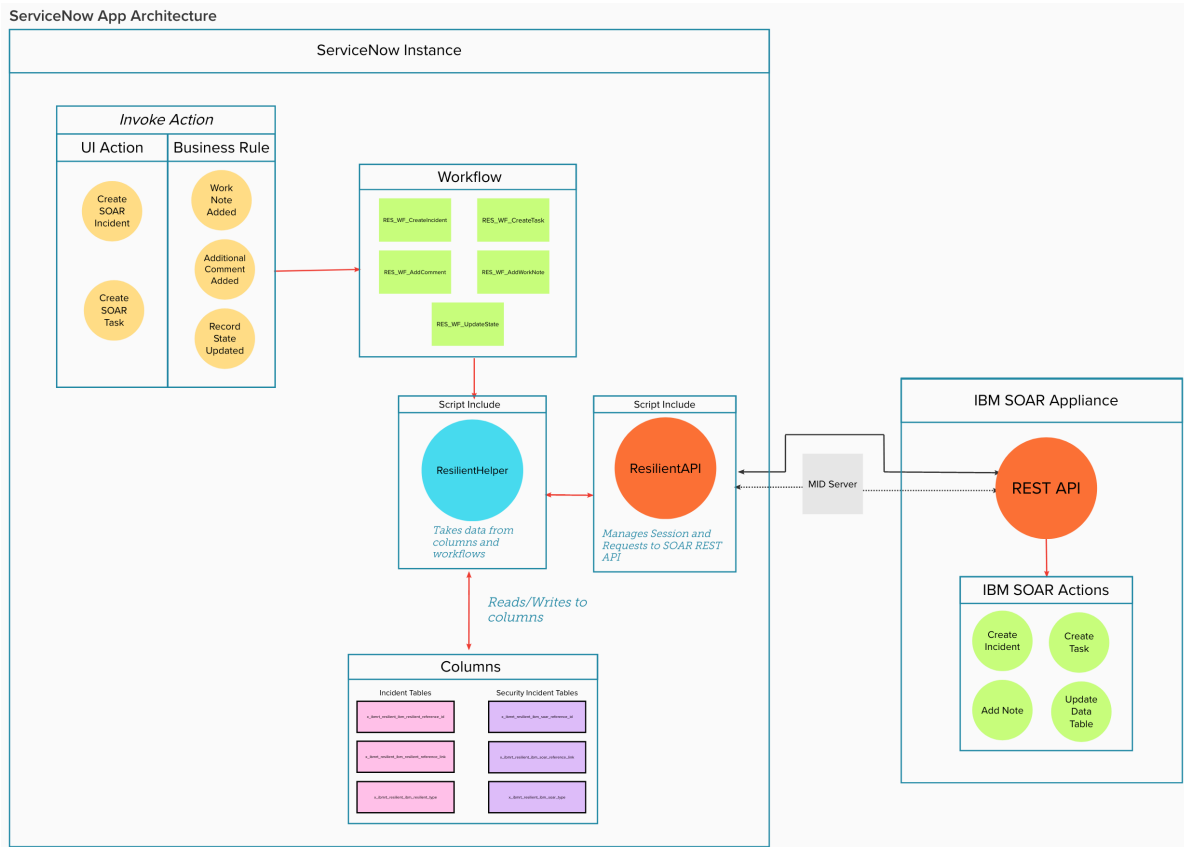
3 Software architecture overview

ServiceNow App Architecture



- A user can create an Incident/Task in IBM Resilient by clicking a ServiceNow UI Action or update an Incident/Task in IBM Resilient or send a Work Note/Additional Comment to IBM Resilient using a ServiceNow Business Rule
- The UI Action/Business Rule starts a ServiceNow Workflow
- The Workflow contains a Run Script that instantiates the ResilientHelper Class
- The ResilientHelper Class reads data from the ServiceNow Record manipulates and maps it, then uses the ResilientAPI Class to make requests to the Resilient Appliance REST API
- If the Resilient Appliance is not publicly accessible (not behind a firewall) a ServiceNow MID Server is needed

4 Software design description



ResilientHelper API

create(GlideRecord record, String snRecordId, String caseName, Object options)

- Creates an Incident/Task in IBM Resilient
- Sets the the following Columns in ServiceNow:
 - x_ibmrt_resilient_ibm_resilient_reference_id
 - x_ibmrt_resilient_ibm_resilient_type
 - x_ibmrt_resilient_ibm_resilient_reference_link
- Adds a new row to the Data Table in IBM Resilient

Parameters:

Name	Type	Description
record	GlideRecord	The current ServiceNow record
snRecordId	String	Is the ID of the record, unique to its table. E.g. 'INC123456'
caseName	String	The name of the Incident/Task to create in IBM Resilient
options	Object	There are 3 valid options that are all optional: <code>incidentId</code> : a String, if defined will create a Task in Resilient. <code>initSnNote</code> : a String, if defined will add it as a Work Note to the ServiceNow record. <code>optionalFields</code> : an Object, if defined will take each key/value pair and will set those fields in Resilient when creating the Incident/Task

Return:

Returns an object with the following keys:

Key	Type	Description
res_reference_id	String	Unique ID relative to the Incident/Task created in IBM Resilient. E.g: RES-1234-100001
res_reference_link	String	A link to the Incident/Task in IBM Resilient
res_reference_type	String	The type of case created in IBM Resilient. Incident/Task
snLink	String	A link to the ServiceNow Record

addNote(String res_reference_id, String noteText, String noteFormat)

- Creates a note in IBM Resilient

Parameters:

Name	Type	Description
res_reference_id	String	Unique ID relative to the Incident/Task created in IBM Resilient. E.g: RES-1234-100001
noteText	String	The text of the note
noteFormat	String	Format of the note to create. Can be <code>text</code> / <code>html</code> . Use <code>html</code> if you want to create a RichText note in IBM Resilient

Return:

Does not return anything

updateStateInResilient(String res_reference_id, String snTicketState, String snTicketStateColor)

- Updates the `servicenow_status` column in the Data Table in IBM Resilient with the `snTicketState`

Parameters:

Name	Type	Description
res_reference_id	String	Unique ID relative to the Incident/Task created in IBM Resilient. E.g: RES-1234-100001
snTicketState	String	The current state of the record. E.g.: 'In Progress'
snTicketStateColor	String	"green", "orange", "yellow" or "red"

Return:

Does not return anything

5 Performance considerations and concerns

- **High Volume Transactions:**
 - In testing, sending multiple IBM Resilient Incidents/Tasks to ServiceNow did not seem to effect the Platform
- **Inbound Web Services:**
 - One ServiceNow Record is created in the Incident Table when a user chooses to send an IBM Resilient Incident/Task to ServiceNow
- **Outbound Web Services:**
 - One ServiceNow Record is exported each time the user clicks the UI Action to create an Incident/Task in IBM Resilient
- **Error Handling:** all API request exceptions are caught and should be printed in the logs and also a Work Note with the error will be added to the ServiceNow Record
- **Languages:** Currently only supports English
- **Mobile:** not designed to support mobile
- **Browsers:** all that are supported by ServiceNow

6 Security considerations and concerns

- **Insecure Protocols:** None
- **Roles:**
 - x_ibmrt_resilient.admin
 - x_ibmrt_resilient.user
 - x_ibmrt_resilient.integrator
- **Users:**
 - User ID: ibmresilient
 - Email: ibm_resilient_integration@example.com
 - First Name: IBM
 - Last Name: Resilient
 - Role: x_ibmrt_resilient.integrator
- **Sensitive Information Handling:**
 - Passwords stored as type password2

7 Application components

7.1 Components created by the application:

- **Script Includes:**
 - ResilientAPI (Server)
 - ResilientHelper (Server)
 - CreateTask (Client Callable)
- **Newly Defined Tables:**
 - None
- **Business Rules:**
 - Add Comment to Resilient
 - Add Work Note to Resilient
 - Update State in Resilient
- **Client Scripts (UI Actions):**
 - Create Resilient Incident
 - Create Resilient Task
- **Modules:**
 - Properties (Set settings to connect to IBM Resilient Appliance)
 - Test Connection (Test connection with IBM Resilient Appliance)
 - Support (Gives link to our Community Forum for Support)

- **Properties:**
 - ResilientHost
 - ResilientOrgName
 - ResilientUserEmail
 - ResilientUserPassword
 - ServiceNowUsername
 - UseMidServer
- **Scope Used:**
 - x_ibmrt_resilient
- **User Needed:**
 - User ID: ibmresilient
 - Email: ibm_resilient_integration@example.com
 - First Name: IBM
 - Last Name: Resilient
 - Role: x_ibmrt_resilient.integrator
- **Roles:**
 - x_ibmrt_resilient.admin (Can see UI Actions, Properties Module and Test Connection Module)
 - x_ibmrt_resilient.user (Can see UI Actions)
 - x_ibmrt_resilient.integrator (Has ACL for all custom endpoints)
- **Components modified by the integration:**
 - Incident Table: A new section added with 3 columns

7.2 Integration components created by the application:

- **Inbound web services:**
 - api/add [POST]
 - Add a work note, additional comment or attachment to a Record in the Incident Table
 - api/close_record [POST]
 - Change the state of a Record in the Incident Table to closed/resolved etc.
 - api/create [POST]
 - Create a new Record in the Incident Table
 - api/get_sys_id [GET]
 - Search any table for a records sys_id
 - api/test_connection [GET]
 - Just simply returns a response if authenticated. Allows us to test connection from IBM Resilient

- api/update [PATCH]
 - Update fields of a ServiceNow Record in the Incident Table

8 Future plans

- To have a custom table with all related IBM Resilient Incident/Tasks
- 'Bindings/references' between Records in the Incident Table and our custom table will be supported using Reference Fields
- Be able to integrate IBM Resilient with any table in ServiceNow and not just the Incident Table

9 Service Level Agreement Definition

In this section, you define the Service Level Agreement (SLA) for your customers.

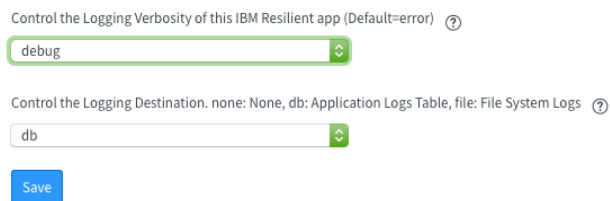
Customers will be instructed to contact the integration provider (your organization) for technical support. If a customer first contacts ServiceNow Customer Support, then ServiceNow Customer Support will isolate the problem and instruct the customer to resolve the issue with your organization.

- **Contact Method:** Website
- **Contact Details:** ibm.biz/resilientcommunity

10 Debugging and troubleshooting

Debug the Integration:

- Logging can be changed to DEBUG mode
 - Go to IBM Resilient > Properties
 - And change the Logging Verbosity to debug:



The screenshot shows the configuration interface for the IBM Resilient app. It has two dropdown menus. The first is labeled 'Control the Logging Verbosity of this IBM Resilient app (Default=error)' and is set to 'debug'. The second is labeled 'Control the Logging Destination. none: None, db: Application Logs Table, file: File System Logs' and is set to 'db'. There is a blue 'Save' button at the bottom.

- To debug, change the logging verbosity and try create a Resilient Incident again

Anticipated issues:

- If using a MID Server, ensure it is up, running and validated

10.1 Debugging tips:

- Change debugging verbosity and check the logs

10.2 Failure modes:

- Cannot reach IBM Resilient Appliance
- User not authenticated

- When creating a Task, ensure the Incident Number is correct

END OF DOCUMENT