Project Title

Ticketing Integration with ServiceNow

Project Description

The Panaseer UI is a dashboarding platform that allows users to identify cybersecurity issues that must be resolved. Typically, the enterprise security team will identify these security gaps and raise work to the IT team to resolve. This is often orchestrated, as part of an ITSM process, through a ticketing system. ServiceNow is the market leader in ITSM ticketing, and the Panaseer UI is to be integrated, via the Panaseer REST API, to ServiceNow in order to create new tickets and read the state of existing tickets. High-Level Design section has further details.

Engagement Model

Outsourced Project – capability delivered to agreed specifications & budget by fully-managed, outsourced team. The capability will be built on the existing Panaseer codebase, rather than a separate, greenfield project.

Ideally, Fixed Price but Time & Materials can be considered. Panaseer has estimated 1-person-month of effort for the initial MVP. If the MVP is delivered successfully, further work will follow to extend the capability.

While process alignment is not required, technical alignment is expected as the developers will be working on the Panaseer codebase where our technical standards & conventions must be followed. All code will be reviewed prior to acceptance and automated testing (unit & integration) is required.

Panaseer will assign a Product Owner and Technical Architect to provide guidance to ensure the final deliverables meets both business and technical requirements.

Required Skills & Experience

For successful delivery, it is expected the team will require the below skills.
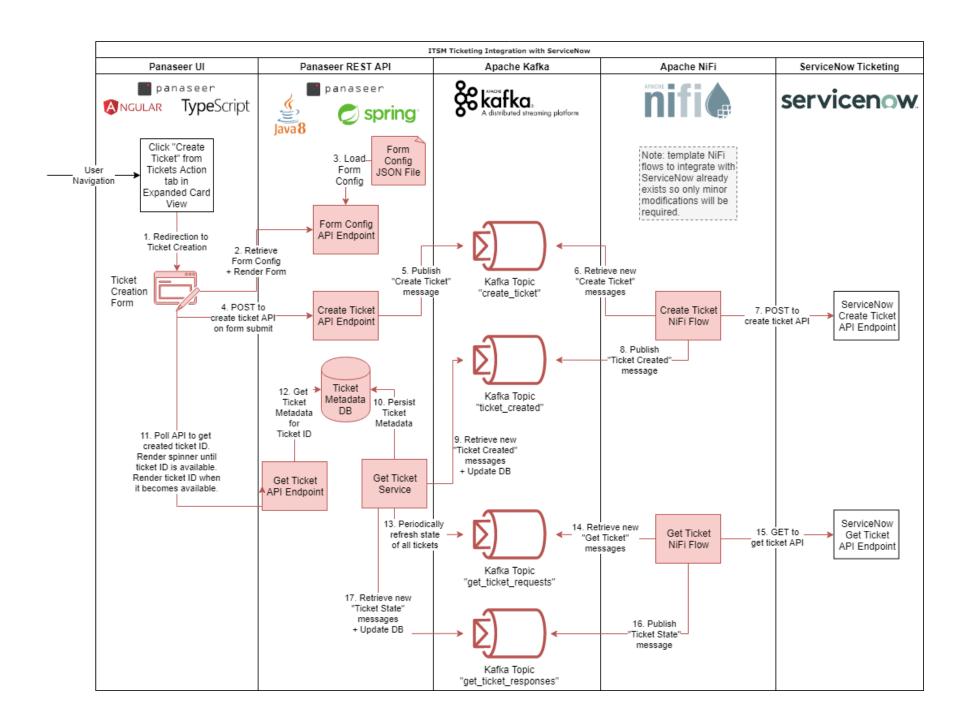
Essential Skills:

- Java 8 + Spring Framework + JPA (e.g. with Hibernate)
- Basic Angular 7 + TypeScript to develop the ticket form using Angular Material & Angular Reactive form components
- Testing
- Git

Desirable Skills:

- Experience with Apache Kafka & Apache NiFi is desirable but not required as the integration is minimal and the requisite skills can be acquired on the project. For NiFi, we have templates that can be adapted, so these will not need to be developed from scratch.

Who are we?

We are a start-up at the intersection of data science and big data technology in the rapidly growing cybersecurity sector. Our platform takes data from security, IT and business solutions to build trusted asset inventories, identify gaps in security controls and arm both security teams and executives with metrics that enable them to make better decisions.

ITSM Ticketing Integration with ServiceNow

Panaseer UI	Panaseer REST API	Apache Kafka	Apache NiFi	ServiceNow Ticketing

panaseer
ANGULAR TypeScript

panaseer
Java8 spring

kafka
A distributed streaming platform

nifi
APACHE

servicenow

Click "Create Ticket" from Tickets Action tab in Expanded Card View

User Navigation

3. Load Form Config

Form Config JSON File

Note: template NiFi flows to integrate with ServiceNow already exists so only minor modifications will be required.

1. Redirection to Ticket Creation

2. Retrieve Form Config + Render Form

Form Config API Endpoint

Ticket Creation Form

5. Publish "Create Ticket" message

Kafka Topic "create_ticket"

6. Retrieve new "Create Ticket" messages

4. POST to create ticket API on form submit

Create Ticket API Endpoint

Create Ticket NiFi Flow

7. POST to create ticket API

ServiceNow Create Ticket API Endpoint

8. Publish "Ticket Created" message

Kafka Topic "ticket_created"

12. Get Ticket Metadata for Ticket ID

Ticket Metadata DB

10. Persist Ticket Metadata

11. Poll API to get created ticket ID. Render spinner until ticket ID is available. Render ticket ID when it becomes available.

9. Retrieve new "Ticket Created" messages + Update DB

Get Ticket API Endpoint

Get Ticket Service

13. Periodically refresh state of all tickets

Kafka Topic "get_ticket_requests"

14. Retrieve new "Get Ticket" messages

Get Ticket NiFi Flow

15. GET to get ticket API

ServiceNow Get Ticket API Endpoint

17. Retrieve new "Ticket State" messages + Update DB

Kafka Topic "get_ticket_responses"

16. Publish "Ticket State" message

High-Level Design

Please refer to the diagram on the previous page for an overview of the design we're considering. The execution flow can be traced by following the enumerated steps. Essentially, the process starts with the write-path (steps 1 to 10) where a new ticket is created, and ends with the read-path (steps 11 to 17) where the state of existing tickets are retrieved.

All items in red (i.e. all red components and interactions) are in scope for delivery in this project.

Further details can be discussed on a scoping call.