Observability Demo: E2E

Demo

RCA

Setup

1. Open the button label="Elasticsearch" Instruqt tab
2. Navigate to `Alerts`
3. Refresh until there are no active alerts (leftover from startup)
4. Navigate to `Workflows`
5. Enable workflow `alert_queue`

Generate Alerts

1. Open the button label="Trader" Instruqt tab
2. Navigate to `ERROR`
3. Open `DB`, select `Generate errors`, and click `SUBMIT`

Monitoring

1. Open the button label="Elasticsearch" Instruqt tab
2. Navigate to `Workflows`
3. Note that `alert_queue` will trigger, then `alert_dequeue`, then `alert_process`, then `case_dequeue`, then `case_process`. You can `alert_process` and `case_process` to follow along.

Outcome

1. Open the button label="Elasticsearch" Instruqt tab
2. Navigate to `Cases`
3. Eventually you should have a new Case created with all 3 alerts correlated and attached, followed by a Root Cause analysis

Remediation

1. Open the button label="Elasticsearch" Instruqt tab
2. Navigate to `Cases`
3. Open the created Case
4. Follow the link at the bottom to `continue the investigation`
5. In the AI Assistant, enter the following `can you restart the proxy service?`
6. Wait for confirmation that the process has been restarted
7. Navigate to `Workflows`
8. Open the `remediation_restart_service` workflow, click `Executions`, and follow along with the last execution
9. Open the button label="host-1" Instruqt tab
10. Enter the following on the command line

```
kubectl -n trading-1 get pods
```

11. Note that the `proxy` pod has been recently restarted!