Note:

- Online Help TOC

INCIDENTS

- go to `FortiSIEM` > `INCIDENTS`
- Incidents is the name given by Fortinet to the alerts triggered by configured rule in FortiSIEM.
- Basically, you will have one rule configured with some condition and this rule is listening for the ingested events, if this events match the rule condition, incidents will be triggered
- so, incidents is nothing but the page that displays all the alerts triggered due to condition is fulfilled or satisfied by the incoming events
- The beauty of FortiSIEM is it not only categorize by `security`, it also categorize by `performance`, `availability`, `change` - key feature of FortiSIEM to combine SOC and NOC together
- `change` - is very important for compliance scenarios to check for changes on your different assets like your firewalls or network devices to make sure that these changes are approved and validated
- Incidents
  - Incidents by Category
  - Top Impacted Hosts - By Severity / Risk Score
  - Top Incidents

Tip

- It's always recommended before rushing to start writing your rule, you have to first make sure that the relevant logs or the interesting logs that you are looking to build the rule to check this logs are there in your system.
- To do this go to `ANALYTICS` page, start looking for required logs. For example
  - If you are looking for continuous failed login attempt
  - when we see the logs, we will be able to easily design our rules

- To simulate the failed login attempt
  - go to windows machine you want to monitor the simulation
  - before trying to check this, you have to make sure that auditing itself is enabled on the PC
    - to do this, go to `Local Security Policy` > `Advnaced Audit Policy Configuration > System Audit Policy - Local Group > Logon/Logoff > Audit Logon` and enable it (both success and failure)
  - Try multiple failed and attempts and then one successful attempt
  - this will be logged in FortiSIEM ANALYTICS tab
    - For failed windows logon: Event Type - `Win-Security-4625`
    - Successful windows logon: Event Type - `Win-Security-4624`

- Let's create the rule
  - go to `RESOURCES` > `Rules`
    - we have around 2800+ rules out of the box in FortiSIEM
    - we can turn this one or off by clicking on `Active` check mark
  - before writing your rule from scratch, search for existing rule base
  - you can clone similar rule you are looking for and modify it
    - use some key words before the custom rule, so you can search
  - To modify the cloned rule, click on edit and follow 3 steps
    - Step1: General - Rule name, description
    - Step2: Define Condition - Edit SubPattern - Name, Filters(you can choose `Event Type` {e.g., , `Win-Security-4625` - you can put this `Expression Builder`}, `Reporting IP` etc.), Aggregate (count - you can refer Online Help TOC for the `Attribute` `Expression Builder` (to validate) - just go to `Working with Analytics` section and then `Understanding Search Components`. then you can check Example of Operators in Expression) and Group By.
      - Good feature - you can run `Run as Query` (it will run for old logs) to run for defined condition to validate you are getting the logs. this another layer of validation
      - you also need to give the relationships between for example between Five failed logins and the one successful login in `Given these Subpattern relationships` (here you need to add for each attribute)
    - Step3: Define Action - you need to define four things
      - Action - defining the mapping between the SubPattern or between the Base events that are matched by the SubPattern - we need to define `Event Attribute` and the `Filter Attribute` of the incident
        - For each `Incident Attributes` we need to select one of the SubPatterns - Just match the `Event Attribute` and the `Filter Attribute`

        - e.g., if the `Event Attribute` is Destination IP then the `Filter Attribute` should be Destination IP
    - Watch List - You can add some attributes to watchlist, here we can add another rule to cross reference this watch list following this rule, so if the attribute in there in the watch list, then this second rule will trigger - This allows us to actually make nesting
        - so one rule triggered for some condition and saved some attribute in the watch list. and another rule, cross-reference this watch list and do some other things
        - for example, if you have communication detected with malicious IP, so you can store this malicious IP in some watch list and then have another rule to check for this watch list along with other condition - this allows for designing complex rule
    - Exception
    - Clear - you can define the clear condition so the incidents to be cleared
        - The incidents will be in `INCIDENT` tab, so we can define some condition and activate some rule to automatically clear the incidents within some time frame

Note:

- Make sure all of the fields available in the `ANALYTICS` tab before simply aggregating on empty fields. just make sure all required columns are visible
- Test the rule before blindly following rule, do some test
- Check `INCIDENTS` page, to check if the incident triggered, if the incident is not triggering then there should be some problem in rule creation

Note:

- if there is any problem in the rule creation, it will show `Synk Error` in Rule section of `RESOURCE` section
    - open the error to check what is the issue in `Description` section
- we need to access CLI of FortiSIEM to check what's going on
    - go to `/opt/phoenix/log` to check the useful logs
        - check `phoenix.log` - search for the error which showed in `Description` section of the `Synk Error`

```
tail -f phoenix.log | grep -i ERROR
tail -100 phoenix.log | grep -i ERROR
```

- To check with our rule (remember: `use some key words before the custom rule, so you can search`), just prefix the what we put in the rule (e.g., if our rule name is: `OURRULE_PREFIX: RULE_NAME`)

```
tail -100 phoenix.log | grep -i OURRULE_PREFIX
tail -100 phoenix.log | egrep -i 'OURRULE_PREFIX|ERROR'
```

- always test with `Run as Query` while creating the rule

Note:

- sometimes space may create some issue while writing `Attribute` . `Expression Builder` may say `Expression is Valid` . but, always test with `Run as Query` to test it.
- if it triggers the incident, there you can check `Triggering events` for that incident.
- in incidents it will also show `Resolution` if it's open or not
  - you can change in in `Actions` > `Resolve Incident` . then choose between 4 options
    - Open
    - In Progress
    - True Positive
    - False Positive
- you can also `Clear Incident`

Methodology

- To develop use cases or create your rule, you need to first define your objective then you search for logs of interest.
- After confirming logs are coming, you design your rule by first searching for an existing rule that may be there (You don't need to reinvent the wheel, just clone the rule and start editing it)
- Rules will have 3 Stages
  - here we can define a joining condition by defining two pattern and defining the relationships among them
  - it may be looks like complex rule that required us to define two patterns with the follow by operator and the relationships among the rules
  - in many of the rules, you just need to define one pattern or even two patterns and just define the AND or OR. and in `Actions` set some attributes.
  - Note that it is important and mandatory to define the relationships between `Incident attribute` and the `Filter attribute`

- and you can also store some attribute in watch list to use it or cross-reference this list with some other rules or even to generate some report
- you can also define `Clear` condition and also `Exception`