Course Code: COMP 421

Info Security (Industry Level)

Section: B

Project: TheHive

Komal Amjad Butt: 22-10134

Ahmed Yasser: 22-10216

Rifa Salman: 22-10067

Syed Ali Hamza: 22-11091

Muhammad Hassan Shakoor Rana: 22-10483

TheHive:
Project Description:

Introduction:

TheHive is a scalable, open source and free Security Incident Response Platform, tightly integrated with MISP (Malware Information Sharing Platform), designed to make life easier for Subsystem Operation and Checkout System **(SOCs)**, computer security incident response team **(CSIRTs)**, Controlled Environmental Regulatory Testing Services **(CERTs)** and any information security practitioner dealing with security incidents that need to be investigated and acted upon swiftly. One of the main functionality of TheHIve is that it allows us to access all the information about your team and their work conveniently.

There are different modules of TheHive software. **Our team will specifically work on the TheHive4Py Api** which is one of the main elements of TheHive Software.

What THEHIVE4PY does?

TheHive4py is a Python API client for TheHive, a scalable 3-in-1 open source and free security incident response platform designed to make life easier for SOCs, CSIRTs, CERTs and any information security practitioner dealing with security incidents that need to be investigated and acted upon swiftly. It allows analysts to create cases out of different sources such as email. For example, a SOC may ask its constituency to send suspicious email reports to a specific mailbox that a script polls at regular intervals. When a new email is received, the script parses it then calls TheHive4py to send an alert to TheHive. Analysts can then preview the alert and if deemed interesting, they can import it as a case and start working on it collaboratively

There is a class of THEHIVE4PY (TheHIveApi) and its constructor takes multiple arguments such as:

def __init__(self, url: str, principal: str, password=None, proxies={}, cert=True, organisation=None,version=Version.THEHIVE_3.value):

- Url (`URL of Thehive instance, including the port. Ex: `http://myserver:9000)`
- Principal (str) `The API key, or the username if basic authentication is used.`
- Password (str): `The password for basic authentication or None. Defaults to None`
- Proxies (dict): `The proxy configuration, would have `http` and `https` attributes. Defaults to {}`
- Cert (bool): `Wether or not to enable SSL certificate validation`

- Organisation (str): `The name of the organisation against which api calls will be run. Defaults to None`
- Version (int): `The version of TheHive instance. Defaults to 3`

On these arguments TheHive4Py authenticates and there are two types of authentications in TheHive4Py:

1. Basic Authentication
   Which authenticates only on Principal, password, organisation
2. Bearer Authentication
   Which authenticates only on principal and organisation

Proposed Method:
As we all know that TheHive4Py allows us to authenticate different things and identify suspicious mails or different security event management tasks. So, we can do following things:

1. Enhance the arguments to increase the authentication that will increase the security eventually.
2. We can reduce the code to one type of authentication instead of two different authentication as we can set password argument compulsory.