Report 2: Student Progress Report

Please complete all fields.

Name:	**Rahul Ethiraj**
Company:	Cisco Systems
Department/Team:	ASA Dev-US
Project Name:	Avatar - Next Generation Firewall – Network Security

1. Brief description of the project:

- Cisco Umbrella Branch solution aims at providing a single, more extensive cloud-based security solution by first examining DNS traffic and then client's suspicious HTTP(S) traffic.

- The Umbrella connector intercepts DNS packets and redirects the interesting DNS queries to the Umbrella resolver for resolution. Once the DNS response is received it forwards the response to the host.

- I work on Umbrella connector to test various possible failure and corner cases of this feature on integration with ASA. This includes testing of features and core functionality such as fail-open, local-domain bypass, umbrella-traffic, umbrella-config, registration, DNScrypt, handling network issues, multi-context, EDNS (datapath), existing Inspect DNS policy, NAT, VPN, and syslog messages.

2. Work done and progress made since Report 1:

- I started my work by understanding the existing topology and previously written scripts. All the scripts were written on Python 3.3 and automation of the test cases was done in an internal Cisco test framework called pyATS, using PyCharm as the GUI for scripting.

- I have started writing python scripts to test the above features of Umbrella since Report 1. I am concentrating extensively on features such as fail-open, local-domain-bypass, umbrella-traffic in both High Availability (HA) and Cluster network topologies.

- The scripts are run inside an internal Docker framework called Kick. I debug for any failures on the previously written scripts by analyzing the automation logs in Kick and look for errors in the debug messages of Adaptive Security Appliance (ASA) router. Upon a failure, I fix the script and do repeated manual testing to ensure stability.

- I have worked on a total of 6 scripts till now, since Report 1 and planning to work on two more for the rest of my internship. After completing writing the scripts, I plan to execute the scripts in Continuous Integration/Continuous Development (CICD) framework for full automation. Full automation includes automating the end points (Virtual machines), so that the test traffic and devices under test are tested on its own without manual intervention and the CICD platform returns the user with a summary of test cases passed and failed after regression and scaling tests.

3. Problems encountered:

- Fortunately, I don't have any problems relating to management, team or people.
- Due to data confidentiality, I have listed few of code & environment specific problems below that I have encountered at a very high level and how I have fixed them:

    - **Environment specific:**
        - Device logs were printed in PyCharm GUI Run window when the same script is run using pyats-4.1, but not when pyats-5.0 is used as a Docker Image.
        - Devices were not connected when the script is run in Debug mode, but successful when executed in Run mode.
    - **Scripts specific:**
        - **Consecutive probes:**
            - Description:
                - Consecutive queries are sent as probes when "send_dig()" is used in "localbypass_dig_traffic()".
            - Solution:
                - "send_dig()" sleeps for 5 seconds between consecutive dig queries and are thus sent as probes. Used "send_dig_fixed_srcport()" instead, which doesn't sleep between queries.
        - **config_Umbrella_prerequisite:**
            - Description:
                - On sequential runs of the script, it takes the previously configured bad resolver IP and uses for send_traffic_operational function, thinking that it is a good-resolver IP.
            - Solution:
                - Set good ipv4 & ipv6 resolver address in Umbrella pre-requisite, so that send_traffic_operational sends queries to correct Umbrella resolver at the initial run.
        - **Unique regex:**
            - Description:
                - Regex patterns such as "bypass", "inject" were not unique and was matching with the "show service-policy Inspect DNS detail" other than expected.
            - Solution:
                - Made a strict regex pattern to cover unique expected counters from "show service-policy Inspect DNS detail".
        - **HA pair not formed:**
            - Description:
                - "show failover | include host" was executed on the context mode in the peer Primary-Standby device, after the secondary device was made was Secondary-Active.
            - Solution:
                - Standby unit was pushed to system mode before checking failover.