InsightGuard

Development of Threat Rules in ELK stack for detecting APTs

Walchand College of Engineering, Sangli

Team name : ReLU or Not to ReLU
Members : Shreeyash Dongarkar
Om Kulkarni
Tanmay Shingde
Vrushali Sangale
Yashraj Rane
Vaishnavi Yadav

Feedback and Iterations

Mentoring Round -1:

Review:

- Mostly on correct track, ELK Setup was one of the aligned things earlier
- What NTRO expects was breifed and clarified
- Documentation of the entire workflow was also needed

Suggestions/Feedback:

- We should complete what we define the scope of the problem statement
- We could also look up to other various types of APTs, threat's and vulnerabilities at the system, or in a network

Feedback and Iterations

Mentoring Round -1:

Implementation of Suggestion:

- Defined the scope of the Problem Statement and attacks that should be demonstrated and Rules needed to be created.
- Created documentation as per the Implementation manual of ELK & Network server, configuration, attack & test scripts, victim machine setup.
- Implementation using GeoIP an open source tool available in ELK
- Implemented 3 attacks sucessfully on ELK stack with proper simulation

Feedback and Iterations

First Round of Evaluation

Review:

- Attacks simulated were correct
- ML methodoloy was also verified

Suggestions/Feedback:

- We should simulate more attacks and APT Rules to cover almost of the cases.
- Lacking in terms of implementation part, theoritical and documentation part can be done to the end.

Feedback and Iterations

First Round Of Evalution

Implementation of Suggestion:

- Implemented winlogbeat
- Implemented more APT rules in ELK stack
- Worked on ML based APT detection

Feedback and Iterations

Mentoring Round -2:

Review & Feedback:

- We should focus on a complete kill chain implementation and by simulating attacks that could exploit the vulnerabilities.

- We can use open source libraries more to simulate the attacks and develop the APT rules

Feedback and Iterations

Mentoring Round -2:

Implementation:

- Referred Atomic Red Team's repository of the APT attacks
- Divided it into 4 stages:

Reconnaissance

• T1082 : System Information Discovery: Collected OS, hardware, and configuration details using systeminfo, hostname, and enumeration commands.

• T1083 : File & Directory Discovery: Enumerated files, directories, and system paths via PowerShell Get-ChildItem.

Privilege Escalation

• T1548.002 : Bypass UAC: Used an Event Viewer–based UAC bypass to launch a high-integrity PowerShell/cmd session.

Feedback and Iterations

Mentoring Round -2:

Persistence

- T1053.005 :Scheduled Task: Created a scheduled task with schtasks.exe to maintain persistence across reboots.

Exfiltration

- T1041 : Exfiltration Over Web/C2: Exfiltrated staged data using Invoke-WebRequest via HTTP POST.

Feedback and Iterations

Mentoring Round -2:

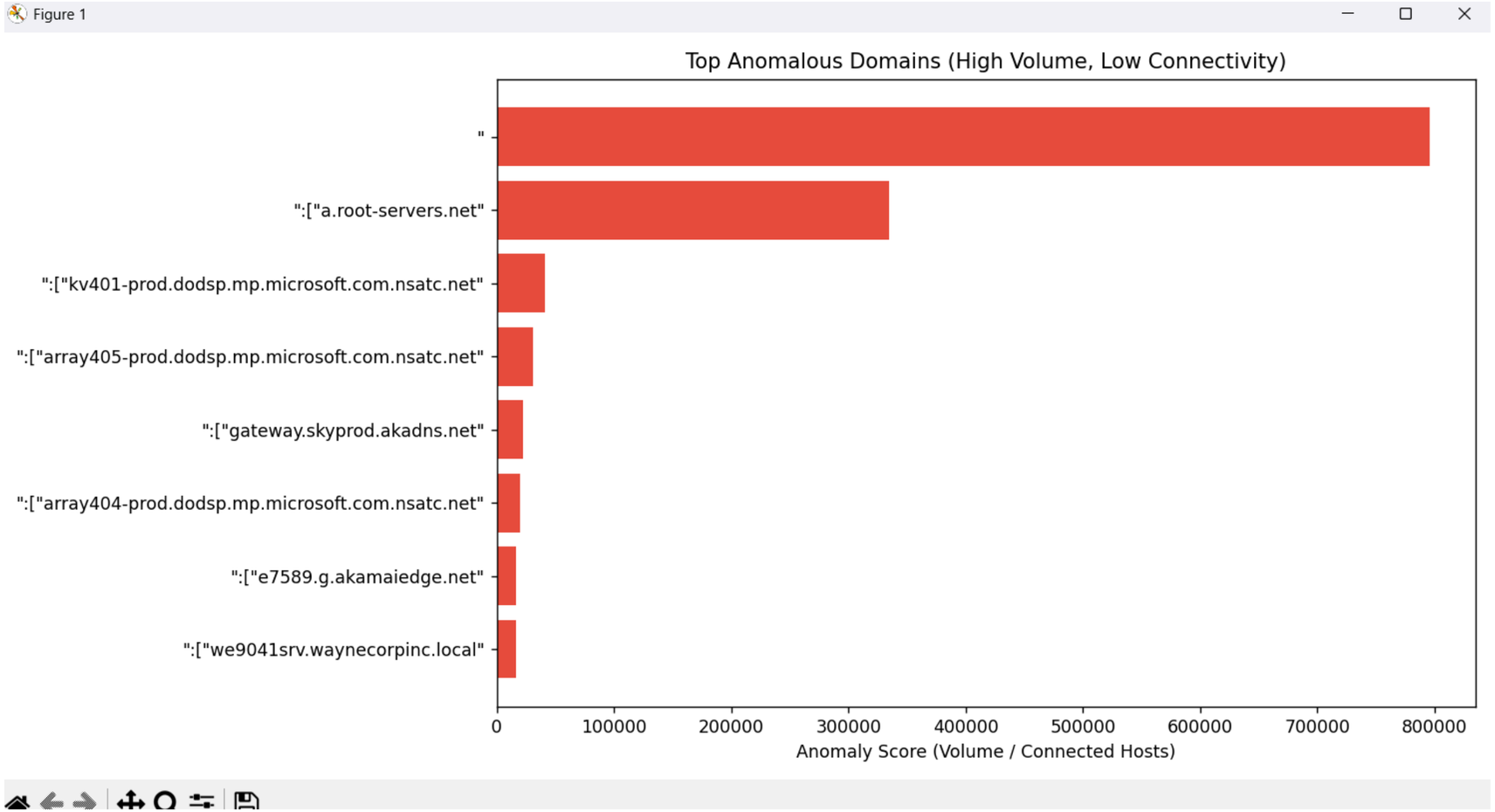
Implementation: Segugio System

- Dataset : Splunk stream: DNS (Open Source)
- Compares multiple domains being suspicious or not
- Calculates: Volume of Requests(Degree) / Connected hosts

Feedback and Iterations

Mentoring Round –2:

Output graph after successfully comparing multiple domains



Feedback and Iterations

Mentoring Round -2:

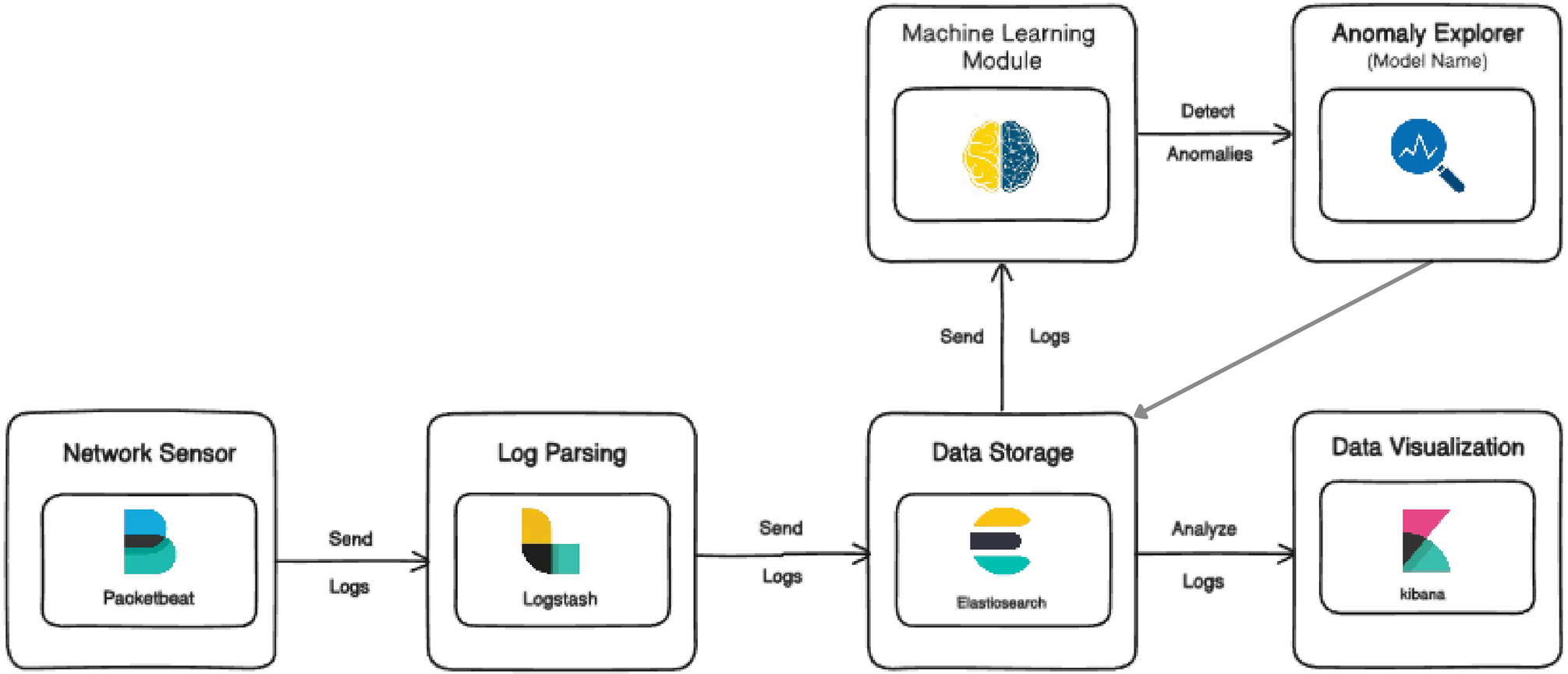
Implementation: Traffic Analysis

- Attackers dynamically change their packet size or timing to stay just below “highsum” threshold → Evade static threshold
- To overcome → DRL Agent learns a policy for e.g.: CIC-IDS2017 Dataset.
- If traffic volume drops but the frequency increases → Example of evasion reference: DRL outperforms static neural network in terms of acccuracy and bandwidth

Feedback and Iterations

Mentoring Round -2:

Architecture:



Feedback and Iterations

Mentoring Round –2:

Contribution of XAI to security system

DRL is a complex neural network → XAI provides the essential **trust** and provide **context** for security operation and centre

- **Trust and Auditability** : By showing exact feeature contribution
- **Reduced Alert Fatigue** : Traige info speeding up investigation process
- **Debugging and Retraining** : To know which feature needs normalization and re-weighting in DRL training phase

Feedback and Iterations

Mentoring Round -3:

- Simulate real world APT36
- Run ML model on real traffic

Feedback and Iterations

Mentoring Round -3:

Implementation:

- APT36 : Created python script to simulate rule as well as tried using Atomic red team APT36
- Real time traffic analysis using packetbeat