Network Anomaly Detection Engines – a comprehensive summary of the state-of-the-art

Akhilesh Anand Undralla

4/21/2022

Anomaly, which is a deviation from normal observations that are suspicious are significant since this behavior can potentially detect critical cyber/network attacks originated from various sources. Network anomalies disrupt the transmission of legitimate traffic over the network. For example, several requests to connect to a web server outside business hours is an anomaly which is identified as Denial-of-Service (DoS). Similarly, there is a need to collect anomalies that essentially contribute to extensive data points on characteristics of all network based cyber attacks to be used for learning patterns.

Traffic collected through networks is difficult to detect anomalies in real-time since data collected can be noisy, high dimensional and voluminous. It is indeed necessary to detect these anomalies in an enterprise network in a short time interval since network wide traffic changes characteristics continuously.

Network anomaly detection approach can provide building an extensive state-of-the-art knowledge base that supersedes signature based approach where attack can only be detected based on known data points of threat signatures. Anomaly detection engines are designed to have high processing speed and greater accuracy for processing structured and unstructured data from several sources. This helps in detecting zero-day exploits which utilize vulnerabilities with no knowledge beforehand and with no recorded signature.

Apart from traditional network filtering engines which supplement each other such as Firewalls, Intrusion Detection Systems and Intrusion Prevention

System, It is becoming efficient to use state-of-the art network anomaly detection engines to mitigate advanced malware explicitly engineered by adversaries. While traditional network controls are aimed towards implicit denial of malicious traffic entering the network. Advanced engines are more aiming towards comprehensive scan of internal networks which can be deemed as a reactive approach which is becoming an essential part of network security to monitor outflow of streaming records of information from internal networks to the outside internet.

Network anomaly detection engines act as proactive network defense systems that can contribute to monitoring newer evidence of compromise, rather than relying on predefined attack profiles.

Unlike traditional network filtering engines, which reject bad activity traffic based on given rules and signatures, anomaly network detection engines use dynamic profile databases. After processing the influx of resulted traffic generated by intrusion detection system and matched with machine defined baseline profiles and updated constantly into the attack profile database that is dynamic and does not match dictated baseline.

Data processing engines use machine learning approaches outsetting grouping and separating input data points from various network endpoints (servers, devices, sensors, routers, switches etc.)in a form suitable for further processing. Training stage, data attributes put forward by the extracted data points are exposed to training/learning datasets. In the detection stage, a set of features finalized from the previous stage, solutions are identified based on features that are deviated from baselines (anomalies).

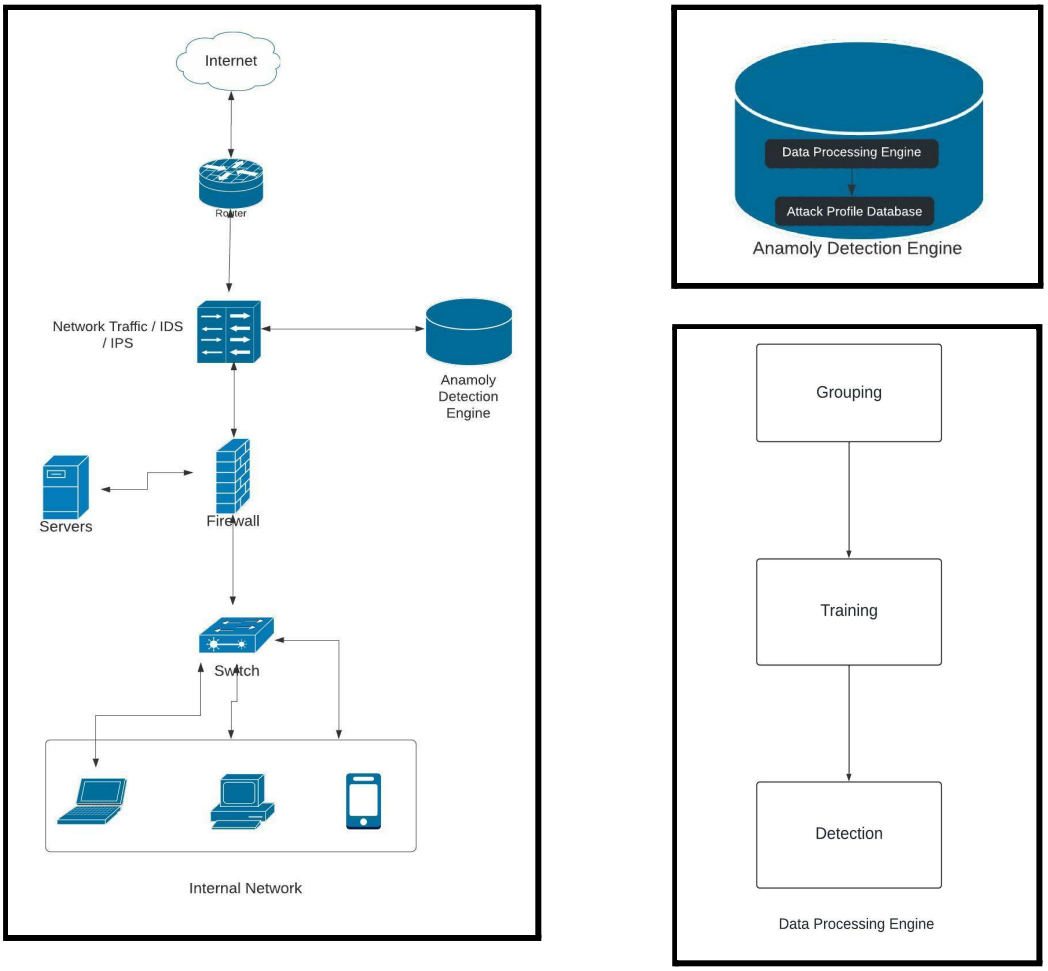

Figure: Network Topology representing position of Anomaly Detection Engine, Breakdown of a Anomaly Detection Engine and Data Processing Engine

Learning models in the training stage will gradually add trained attributes to detect a possible sophisticated attack that can be fed to the detection stage. A normal behavior of a network traffic (baseline) uses a database that self-updates using several data points using a framework fed with traffic patterns, various threat patterns, profiles, signatures etc. Given that several data streams are infused from various data sources, combining big data analytics with machine learning algorithms utilizes the volume and variety of influx of the data feeding the algorithms. This helps in drawing analyzed results such as hidden patterns and analytics, which assists in predictive modeling.

Network data streams towards anomaly detection engines may consist of a combination of captured network traffic, influx of IDS or IPS. Various industry advanced network capturing tools are wireshark, Nfdump, tcptrace, NfSen, Gulp, Cisco Netflow and various tools listed below

Tool	Description
wireshark	Packet capture
Nfdump	netflow data from the network
nfsen	Nfdump's graphical data
gulp	Linux based Lossless Gigabit Remote Packet Capture
nmap	Port Scanning
tcptrace	summarization of the connections from tcpdump
rnmap	Remote client-server port scanning
Cisco netflow	Cisco's network flow (proprietary)

Various attributes of network data streams that are aggregated to network data processing engines that can contribute to a network flow database which is dynamic i.e., scalable and cached.

- IP address - Source (Sender) and Destination (Receiver) of the traffic
- Ports - Source and destination ports that determines the application utilizing the traffic
- Class of the service - Inspect the order or priority of the traffic

- Timestamp
- Subnets
- Network packet size
- Location
- ID tags

Network flow database stores information such as packet size, operating system, bandwidth utilization. This information can be used to determine variable network behavior such as system transmits high volume of requests, unusual use of bandwidth, sensitive data transmission. These can be categorized into volume based anomalies, behavioral based anomalies.

Following technical controls can be implemented after setting up a network anomaly detection engine to make the internal network resilient.

1. Application proxy servers placed in demilitarized network zone to examine ingress traffic of web server, email server, ftp server
2. Security information and event management systems (SIEM) tools that collect and analyze security events across devices. SIEM tools generate alerts that can interfere with automation or human involvement.
3. In a cloud environment, operational data can be collected in the form of logs, metrics, events and utilize statistics based decisions on those metrics into the repository.
4. Technical controls such as network probes (NetFlow, NetStream, IPFIX, jFlow etc.) can utilize generated network traffic statistics from routers/switches to assist anomaly detection engines to detect malicious behavior.

Collecting huge amounts of data also comes with challenges of protecting or using sensitive data for learning data models. This can be addressed by efforts of

de-identification of data and publishing results with lower risk. This further poses another challenge, a draw-back of publishing restricted results that effects network attack traffic parameters and logs.

Recent advances in cloud computing can make use of simulation techniques with secure test isolation instances to emulate live attacks using an influx of live data sets while emulating large complex physical infrastructure with several network data endpoints. This can generate more accurate data models using a combination of real and synthesized data. Cloud-based services adoption by large-scale networks/organizations also increased importance to protect enterprise network anomalies to reduce costs incurred by malicious attacks.

Anomalies with different data patterns translate to critical application design patterns. Network anomalies can be categorized into security related anomalies and performance based anomalies. Security anomalies occur due to malicious activities to enter the computer network by manipulating network traffic through DDoS attacks, IP masking etc. Security based anomalies generally occur due to misconfiguration of internal computer networks. Performance based anomalies can be identified as unusual traffic patterns such as small/large transfers of data nodes between networks, server failure data etc.

Various machine learning approaches such as supersived, semi-supervised, unsupervised and hybrid approaches are used to detect and separate anomalies. Supervised approach is built upon training datasets that contain collected data instances of both normal and attack profiles and uses a predictive model to group unknown data instances to appropriate profiles. Semi-supervised approaches only contain normal behavior profiles and this model detects anomalies based on normal data instances. Unsupervised approach does not utilize training data instances and can potentially produce errors in the results. Hybrid approach combines other

approaches to extract network anomalies on a large scale efficiently. Supervised and Semi-supervised approaches need huge amount of generated normal data instances while utilizing accurately labeled normal and attack profile data.

Datasets gathered must be evaluated and tested to justify the reliability of anomaly detection. This aids in tuning the parameters of the data, efficiency of different approaches and generating new approaches. Datasets can be generated in real life or can be obtained from public resources. Benchmark datasets are publicly available datasets generated using attack-prone simulated large networks. Real life datasets are created overtime or at scheduled intervals to collect network traffic potentially containing both normal and attack profiles. Real-life dataset generation is unbiased and imitates real-world attacks which can be achieved by setting up infrastructure or test network architectures. These network architectures are built to fit various network attacks such as DDoS, detecting server failures and various attack scenarios.

Based on type of algorithm used, network anomaly detection techniques are classified into 1) Statistical techniques and systems, 2) Classification-based techniques and systems, 3) Clustering and outlier-based techniques and systems, 4) Soft computing-based techniques and systems, 5) Knowledge based techniques and systems, and 6) Techniques and systems based on combination learners.

Since network traffic changes constantly, performance of each technique depends on the deployment point in the network and contributes to efficient handling of systems to mitigate growing threats of cyber attacks.

References:

Friedberg, Ivo, et al. “Cyber Situational Awareness through Network Anomaly Detection: State of the Art and New Approaches.” *E & I Elektrotechnik Und Informationstechnik*, vol. 132, no. 2, 30 Jan. 2015, pp. 101–105, 10.1007/s00502-015-0287-4. Accessed 8 Apr. 2022.

Sperl, Philip, et al. *A 3 : Activation Anomaly Analysis*.

Dutta, Vibekananda, et al. “A Deep Learning Ensemble for Network Anomaly and Cyber-Attack Detection.” *Sensors*, vol. 20, no. 16, 15 Aug. 2020, p. 4583, 10.3390/s20164583. Accessed 8 Apr. 20202.

“Network Anomaly Detection Software | Detect Network Anomalies - ManageEngine NetFlow Analyzer.” *www.manageengine.com*, www.manageengine.com/products/netflow/network-anomaly-detection.html. Accessed 11 Apr. 2022.

“What Are Network Anomalies and How to Detect Them | Flowmon.” *www.flowmon.com*, www.flowmon.com/en/blog/science-of-network-anomalies#active-passive-anomaly-detection. Accessed 11 Apr. 2022.

Merlo, Alessio, et al. “Anomaly Detection in Computer Networks: A State-of-The-Art Review.” *www.academia.edu*, www.academia.edu/15142118/Anomaly_Detection_in_Computer_Networks_A_State_of_the_Art_Review. Accessed 11 Apr. 2022.

Ahmed, Mohiuddin, et al. “A Survey of Network Anomaly Detection Techniques.” *Journal of Network and Computer Applications*, vol. 60, Jan. 2016, pp. 19–31, www.gta.ufrj.br/~alvarenga/files/CPE826/Ahmed2016-Survey.pdf, 10.1016/j.jnca.2015.11.016.

Bhuyan, Monowar H., et al. “Network Anomaly Detection: Methods, Systems and Tools.” *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, 2014, pp. 303–36. *Crossref*, https://doi.org/10.1109/surv.2013.052213.00046.