



DEPARTMENT OF THE NAVY

OFFICE OF COUNSEL
NAVAL UNDERSEA WARFARE CENTER DIVISION
1176 HOWELL STREET NEWPORT RI 02841-1708

IN REPLY REFER TO

Attorney Docket No. 210938
12 October 2022

The below identified patent application is available for licensing. Requests for information should be addressed to:

TECHNOLOGY PARTNERSHIP OFFICE
NAVAL UNDERSEA WARFARE CENTER
1176 HOWELL ST.
CODE 00T2, BLDG. 102T
NEWPORT, RI 02841

Serial Number 17/900,982
Filing Date 1 September 2022
Inventor Makia S. Powell

Address any questions concerning this matter to the
Technology Partnership Office at (401) 832-3339.

DISTRIBUTION STATEMENT
Approved for Public Release
Distribution is unlimited

REAL-TIME NETWORK INTRUSION DETECTION SYSTEM

STATEMENT OF GOVERNMENT INTEREST

[0001] The invention described herein may be manufactured and used by or for the Government of the United States of America for governmental purposes without the payment of any royalties thereon or therefor.

CROSS REFERENCE TO OTHER PATENT APPLICATIONS

[0002] None.

BACKGROUND OF THE INVENTION

(1) Field of the Invention

[0003] The present invention is directed to a method for detecting an intrusion in a packet switched network in real time.

(2) Description of the Related Art

[0004] The need for cybersecurity has increased recently with the both the increase of society's reliance upon computer equipment and increase of cyberattacks on commercial systems, governmental systems and the infrastructure. However, the amount of cybersecurity data available on systems easily leads to operator overload, reducing the effectiveness of standard cybersecurity systems. To aid cybersecurity operators, there is a need to merge machine learning with the correct

features/variables to make better decisions and/or aid them in maintaining networks and systems. (Bresnicker, K., Gavrilovska, A., Holt, J., Milojicic, D., & Tran, T. (2019). Grand Challenge: Applying Artificial Intelligence and Machine Learning to Cybersecurity. *Computer*, 45-52). A lack of cybersecurity can put the data contained on the systems in danger and disrupt operations as adversaries gain access to critical information and systems (Rose, S., Borchet, O., Mitchell, S., & Connelly, S. (2020, August 11). SP 800-207 *Zero Trust Architecture*. From Information Technology Laboratory Computer Security Resource Center: <https://csrc.nist.gov/publications/detail/sp/800-207/final>).

[0005] The consequences of a lack of cybersecurity can be seen in recent events. Relevant examples of the need for cybersecurity are the recent cyberattacks against the United infrastructure in the form of the attack against a gas pipeline in the southeast United States, Colonial Pipeline, which disrupted the gas supply. (Benner, K., & Perlroth, N. (2021, June 7). U.S. Seizes Share of Ransom From Hackers in Colonial Pipeline Attack. From The New York Times: <https://www.nytimes.com/2021/06/07/us/politics/pipeline-attack.html>), and an attack on a beef processor, JBS, which threatened beef availability (Bastista, F., Hirtzer, M., & Doming, M. (2021, June 1), <https://www.bloomberg.com/news/articles/2021-05->

31/meat-is-latest-cyber-victim-ashackers-hlt-top-supplier-jbs).
All of JBS's U.S. Beef Plants Were Forced Shut by Cyberattack.).

[0006] These attacks could have been prevented via a relatively new approach known as zero trust architecture, combined with machine learning cybersecurity. There are many other systems which can benefit from this integrated approach. Machine learning cybersecurity is a combination of data science and machine learning which analyzes network and system data, which may be terabytes per day, recognizes anomalies using machine learning algorithms and then acts based upon the data. The common sources of information include logs, network traffic headers, network packet information, and data length. By sorting through terabytes of data, the alerts generated can either result in the network automatically taking steps to secure itself, such as shutting down traffic between routers, or alerting an administrator to act (Rose.). This is often part of a zero trust architecture whose main tenet is to not inherently trust any network, resource, or user but instead verify the identity of each actor each time a resource is requested (Id.). Zero trust assumes that a network has already been infiltrated and thus takes preemptive steps to protect data. This prevents not only initial attacks but reduces lateral movement of an adversary once they gain access to a network (Id.). Although zero trust and machine learning cybersecurity work well together

to secure systems and data, there is still a need to determine what data is needed from the system and networks and a standardized approach to both gather data and train machine learning cybersecurity systems (Bresnicker.). These data needs are the primary focus of this research.

[0007] Increased cybersecurity is difficult because many attacks exploit newly discovered vulnerabilities and originate from new sources. Current intrusion detection methods require the cybersecurity operator to sift through gigabytes or terabytes of network traffic. This is infeasible with current technologies because these methods require on recognizing a known attack method or signature.

[0008] Machine learning has been used for identifying these zero-day attacks. Existing machine learning methods often require many features, which makes real time processing impractical. Reduction of the number of features is expected to reduce detection to around 50% which is unacceptably low. Machine learning and artificial intelligence methods have emerged which are able to detect and identify new attack traffic, however many need additional tools such as Zeek/Bro IDS to sift through and process the traffic, negating availability as a real-time system. For example, some prior art research requires analysis of 39 features, but raw, real-time network packet capture data commonly only provides 5 features.

(Moustafa, N., & Slay, J. (2015). UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). 2015 Military Communications and Information Systems Conference (MilCIS), 1-6.
doi:10.1109/MilCIS.2015.7348942).

SUMMARY OF THE INVENTION

[0009] The first object of the present invention is to provide a method for identifying a cyberattack in real time.

[0010] Another object is to identify the particular type of cyberattack in real time.

[0011] Accordingly, there is provided a method for detecting a cyberattack. A set of packet capture training data has data elements labeled as being normal or cyberattack data. Metrics in the data are identified that are associated with either cyberattack data or normal data. Statistical measures are developed from these metrics. The training data and statistical measures are used to train a machine learning network. Real packet capture data is obtained and statistical measures are developed for this real data. The trained machine learning network, real data and real statistical measures are utilized to classify the real data as cyberattack data or normal data. Users are alerted if the trained machine learning data identifies cyberattack data in the real packet capture data.

[0012] In another embodiment, multiple machine learning networks are trained on normal data and cyberattack data and statistical measures associated with a particular cyberattack. Real data is provided simultaneously to the multiple trained networks. Output from the networks is analyzed to determine if a cyberattack has occurred and the particular cyberattack. Users are alerted. Optionally, the network being monitored can take preprogrammed responses associated with the particular cyberattack.

BRIEF DESCRIPTION OF THE DRAWINGS

[0013] Reference is made to the accompanying drawings in which are shown an illustrative embodiment of the invention, wherein corresponding reference characters indicate corresponding parts, and wherein:

[0014] FIG. 1 is a diagram of a machine learning training process.

[0015] FIG. 2 is a diagram for utilizing the trained machine learning network to detect network intrusions.

[0016] FIG. 3 is a diagram for utilizing a plurality of trained machine learning networks to detect network intrusions.

DETAILED DESCRIPTION OF THE INVENTION

[0017] FIG. 1 shows a flowchart 10 for preparing and training the machine learning portion of the system. In step 12 the user

analyzes its own TCP/IP network to determine metrics related to the packets on the network. The data captured is referenced as packet capture (PCAP) data which is a set of network metrics. PCAP data is observed utilizing a network analyzer. Software such as Zeek, Netflow, sFlow or the like can be utilized to capture these metrics. Some of the available metrics are destination bits per second; source to destination packet count; destination to source packet count; source TCP window advertisement value; destination TCP advertisement value; source TCP base sequence number; destination TCP base sequence number; mean packet size transmitted by source; mean packet size transmitted by the destination; pipelined depth into connection of the http transaction; actual uncompressed content size of the data transferred; source jitter; destination jitter; record start time; record last time; source interpacket arrival time; destination interpacket arrival time; TCP connection setup roundtrip time; and others. The own network analysis should include statistical measures of these metrics such as mean, median and mode.

[0018] In step 14, a training data set is selected or developed has non-attack data that conforms with the statistical measures developed in step 12. Various training data sets are available to cybersecurity researchers. These data sets are PCAP data that includes labeled non-attack data and labeled

attack data. Cybersecurity attack data may also include a label for the type of cybersecurity attack. In the alternative, it is known to develop training data by collecting baseline data concerning one's own network as labeled non-attack data and then subjecting the network to attack with hacking tools. The data collected during the mock attack can be labeled as attack data. As yet another alternative, baseline data can be collected concerning one's own network and labeled as non-attack data. This can be combined with labeled attack data from an outside cybersecurity data set. In any case, the non-attack data should have statistical measures that are similar within a statistical tolerance to those collected in the own network analysis step 12.

[0019] In step 16, features for analysis are selected from the list of metrics available from the network analyzer software. Differences between the mode, median, and mean of the attack data features and the non-attack data features can provide an indicator that the data is attack data. Mode can be used to provide an expected value. (This is especially useful when the feature is non-numeric.) Median can be used for establishing thresholds at the median value plus a percentage threshold and the median value minus a percentage threshold. The mean predicts the average value. A high standard deviation associated with the mean may indicate that the metric should not

be used for analysis. With these, the standard deviation gives uniformity of the data.

[0020] In order to select features, metrics for available features were compared using a training data set with labeled attack data and non-attack data. Large differences between the mean values for attack data when compared to non-attack data suggests usefulness of those feature means. In one case it was found that the source byte means, the destination byte means, and the mean number of packets differed significantly between the attack data and the non-attack data. The source byte mean for attack data was found to be 100% higher than that for the non-attack data. The mean of the destination bytes feature for attack data is nearly twice the mean of the destination bytes feature for non-attack data. The mean number of source packets in the non-attack data is 30% lower than the mean number of source packets in the attack data. In destination packets, the mean for attack data is over twice the mean for non-attack data. In this case, the source byte feature, the destination byte feature, and the destination packet feature were selected as features for training because these feature means were significantly different between the attack data and the non-attack data. This suggests a low P-value or a low likelihood that the differences occur by chance given the distribution of data. The University of New South Wales (USNSW) network data

set, cited above, has a non-normal skewed distribution. Methods tolerant of non-normally distributed data were used, and the P-value conclusions are true.

[0021] These three features (source bytes, destination bytes, and state) stood out from the other features as both independent and having the most statistically significant expected values in statistical analysis. Independence is necessary to use the features in machine learning algorithms since adding the data is expected to result in higher accuracy. The statistical significance makes it more likely that the machine learning algorithms will be able to differentiate the data labels, leading to higher accuracy. In other networks, different features and a different number of features may be independent and statistically significant. The features can be selected by setting a top number of features or by the number of features having high statistical significance.

[0022] While source packets showed a difference, the difference was not as significant as for these three features.

[0023] State was also analyzed as a feature because it is readily available and independent. State is the packet state. State depends on transaction protocol and has 16 values (ACC, CLO, CON, ECO, ECR, FIN, INT, MAS, PAR, REQ, RST, TST, TXD, URH, URN and '-' if not applicable). These were encoded as integers for compatibility with machine learning algorithms. Mode is

useful for this analysis because if the states are given numeric values, each state represents a different class. Means and medians are meaningless. The mode of the state differed between attack data and non-attack data. In one example, the mode of non-attack traffic is "FIN", and the mode of attack traffic is "INT." As a difference, this is a feature that can be utilized to train the machine learning algorithm.

[0024] Another criteria for selection of features is whether the features are independent and readily obtainable for packet capture data. Source bytes and destination bytes are readily available from packet capture data. Source packet number and destination packet number are not readily available and would need to be computed. These features should not be included unless they significantly improve attack detection.

[0025] Analysis should be performed to determine if the features are independent. Concerning source packet number and destination packet number, these features were compared with the source byte mean and the destination byte mean. A definite correlation was found between source byte mean and source packet number and also between destination byte mean and destination packet number. In view of this, source packet number and destination packet number features do not need to be considered because these features are not independent.

[0026] In step 18, the training data is prepared by extracting the selected features and the class. A supervised machine learning technique is selected in step 20 for classifying input data as either attack data or normal data. Supervised machine learning algorithms include neural network with two hidden layers, k-means clustering, Gaussian mixture clustering, random forest, extra trees, gradient boosting, histogram gradient boosting, voting classifier with random forest and logistic regression, bagging, Adaboost, and stacking classifier with random forest and logistic regression. In step 22, these machine learning techniques were trained utilizing the prepared training data.

[0027] After training, the trained network was tested with unmarked data in step 24. During testing, it was found that tree-based machine learning algorithms were most effective. These algorithms include random forest, extra trees, gradient boosting, histogram gradient boosting, voting classifier with random forest and logistic regression, bagging, Adaboost, and stacking classifier with random forest and logistic regression. Less accurate techniques include neural network with two hidden layers, and k-means clustering. Generally, these tree-based algorithms performed with an accuracy of 90% in distinguishing attack traffic from normal traffic. They also had a 97% accuracy in identifying specific cyberattacks.

[0028] FIG. 2 shows a first embodiment of classifying attack data and normal data utilizing the trained machine learning network prepared as described with reference to FIG. 1. Raw data packets are captured from the network being analyzed in step 30. In step 32, the features selected in step 16 are computed and extracted from the captured packets. In step 34, these features are applied to the trained machine learning network prepared in step 22. The trained machine learning algorithm classifies the data as normal or as attack data. In step 36, data processing continues if the captured packets are normal and monitoring continues in step 30. When the packets are identified as attack data, the user is alerted in step 38.

[0029] FIG. 3 shows a second embodiment that allows for classifying the type of attack. Utilizing the procedures of FIG. 1, several machine learning networks are each trained on normal data and on specific cyberattacks. This is performed by preparing cyberattack data associated with the particular cyberattack and using that as the cyberattack data. Different cyberattacks can have different selected features, and each machine learning network can be trained using its own set of selected features. The method described in FIG. 1 then provides a machine learning network trained to identify one particular cyberattack.

[0030] In FIG. 3, data packets are captured from the network in step 40. All selected features required for identifying any trained cyberattack are computed in step 42. The computed features are provided to the plurality of trained machine learning networks 44-1, 44-2 through 44-N. Each of these networks 44-1-N determines whether the submitted features evidences the associated type of cyberattack. Machine learning network output is received at analysis routine 46. Analysis routine 46 determines if any of the networks 44-1-N show a cyberattack. Routine 46 further indicates which kind of cyberattack has been identified. If no cyberattack is shown, decision block 48 continues processing with step 40. If cyberattacks are shown, step 50 is executed for action. Action may include a suggested response with the identified cyberattack.

[0031] It will be understood that many additional changes in the details, materials, steps, and arrangement of parts, which have been herein described and illustrated in order to explain the nature of the invention, may be made by those skilled in the art within the principle and scope of the invention as expressed in the appended claims.

[0032] The foregoing description of the preferred embodiments of the invention has been presented for purposes of illustration and description only. It is not intended to be exhaustive, nor

to limit the invention to the precise form disclosed, and obviously, many modification and variations are possible in light of the above teaching. Such modifications and variations that may be apparent to a person skilled in the art are intended to be included within the scope of this invention as defined by the accompanying claims.

REAL-TIME NETWORK INTRUSION DETECTION SYSTEM

ABSTRACT OF THE DISCLOSURE

A method for detecting a cyberattack is provided. A set of packet capture training data has data elements labeled as being normal or cyberattack data. Metrics in the data are identified that are associated with either cyberattack data or normal data. Statistical measures are developed from these metrics. The training data and statistical measures are used to train a machine learning network. Real packet capture data is obtained and statistical measures are developed for this real data. The trained machine learning network, real data and real statistical measures are utilized to classify the real data as cyberattack data or normal data. Users are alerted if the trained machine learning data identifies cyberattack data in the real packet capture data.

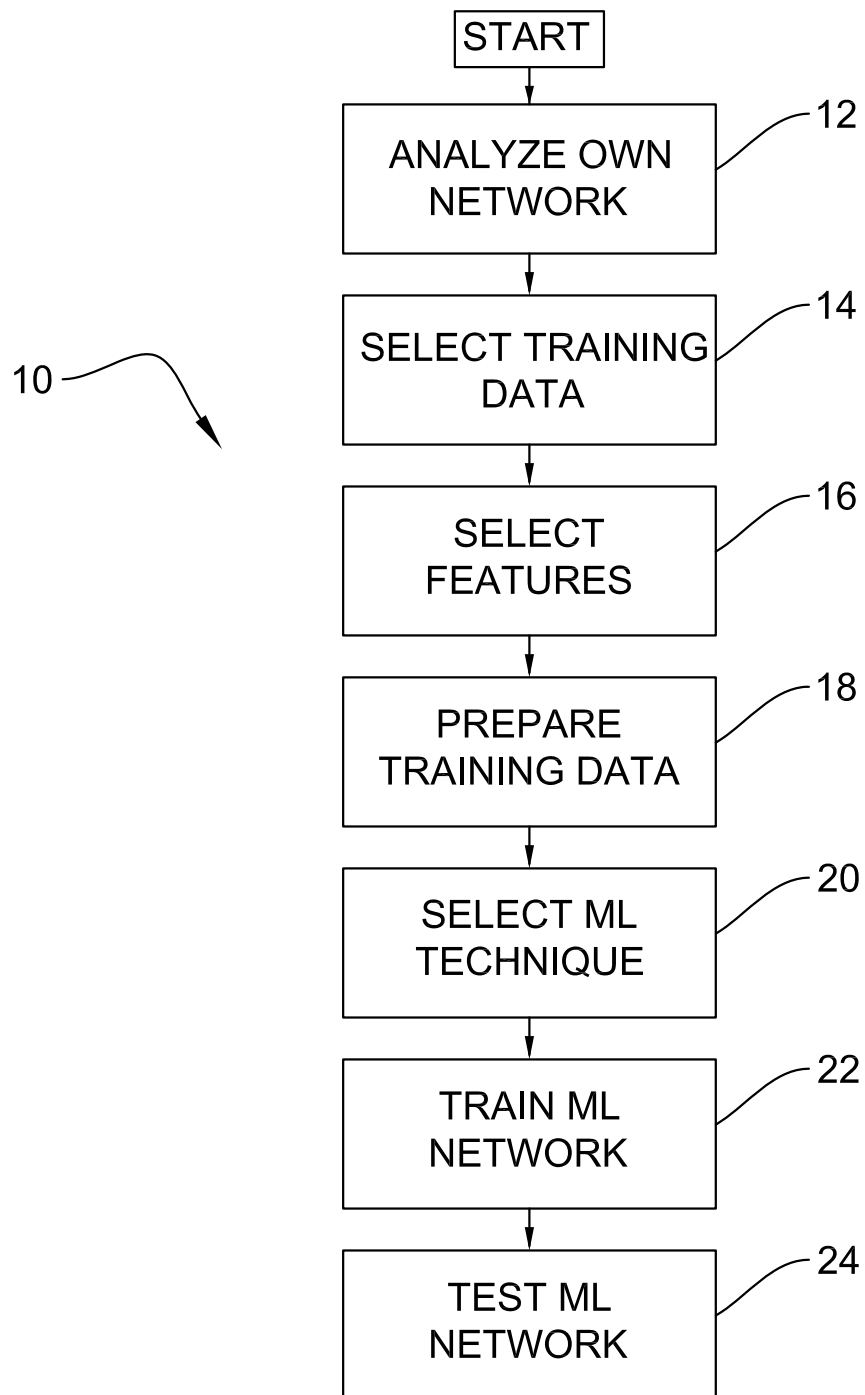


FIG. 1

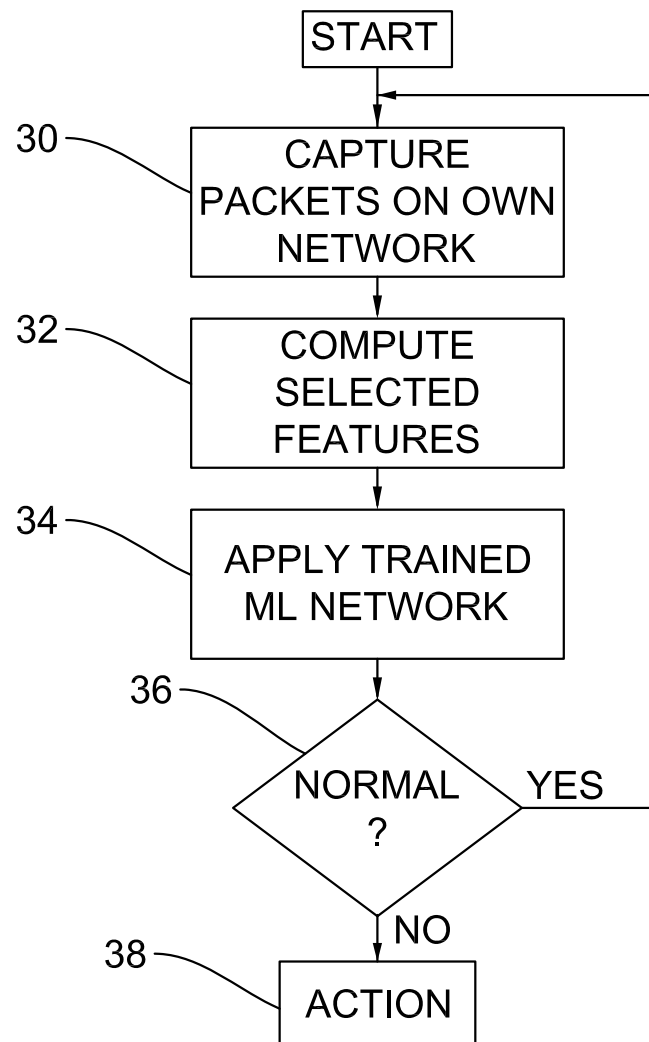


FIG. 2

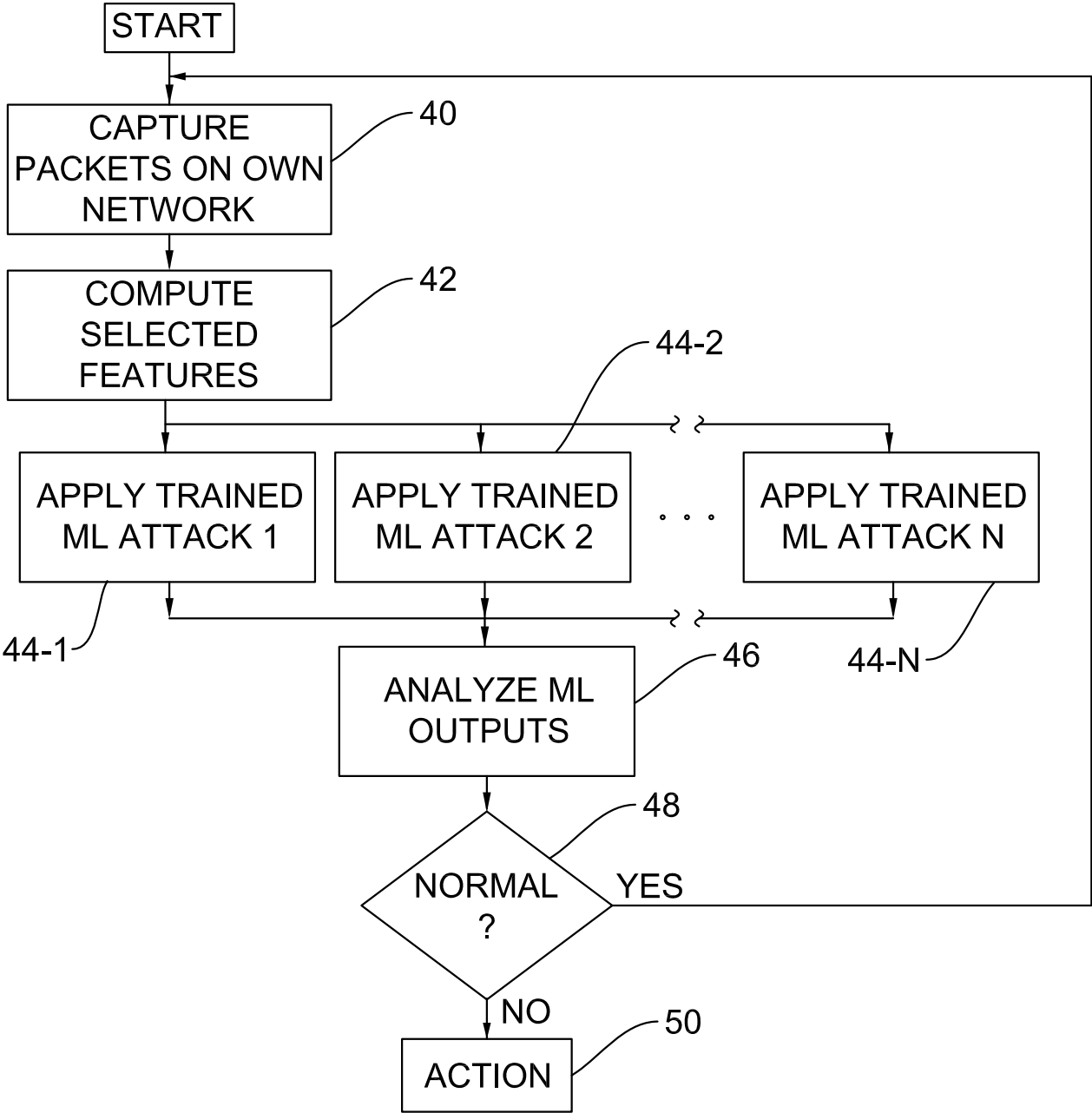


FIG. 3