An anomaly detection framework for cyber-security data

Marina Evangelou
Department of Mathematics,
Imperial College London
and
Niall M. Adams
Department of Mathematics,
Data Science Institute,
Imperial College London

May 3, 2020

Abstract

Data-driven anomaly detection systems unrivalled potential as complementary defence systems to existing signature-based tools as the number of cyber attacks increases. In this manuscript an anomaly detection system is presented that detects any abnormal deviations from the normal behaviour of an individual device. Device behaviour is defined as the number of network traffic events involving the device of interest observed within a pre-specified time period. The behaviour of each device at normal state is modelled to depend on its observed historic behaviour. A number of statistical and machine learning approaches are explored for modelling this relationship and through a comparative study, the Quantile Regression Forests approach is found to have the best predictive power. Based on the prediction intervals of the Quantile Regression Forests an anomaly detection system is proposed that characterises as abnormal, any observed behaviour outside of these intervals. A series of experiments for contaminating normal device behaviour are presented for examining the performance of the anomaly detection system. Through the conducted analysis the proposed anomaly detection system is found to outperform two other detection systems. The presented work has been conducted on two enterprise networks.

Keywords: NetFlow, machine learning, regression models

1 Introduction

Online Trust Alliance named 2017 as another bad year of cyber attacks, with the estimated number of attacks to enterprises to have exceed 350,000 with only half of them reported or even noticed by the affected enterprises[1]. These numbers show the urgency of a new generation of cyber defence systems that can complement the existing intrusion detection systems. Most of the currently implemented systems are signature based tools, where signatures of known attacks are matched to the incoming signatures and blocked accordingly. Signature based systems are usually found at the level of firewalls, the first line of defence of an enterprise network separating the network from the outside world. Signature based systems are also found at the level of anti-virus software installed on individual devices.

Data-driven anomaly detection systems have been discussed in the literature as complementary detection systems. Such systems aim to detect any abnormal deviations from the **normal** activity of an enterprise network. They are based on statistical and machine learning methods that model the normal activity of the network. A number of anomaly detection systems have been proposed in the literature [Lakhina et al., 2004, Kind et al., 2009, Neil et al., 2013, Kimura et al., 2014, Gates et al., 2014, Juvonen et al., 2015, Whitehouse et al., 2016, Schon et al., 2017, Riddle-Workman et al., 2018]. Comprehensive reviews of existing methods can be found in Patcha and Park [2007], Liao et al. [2013] and in the book series by Adams and Heard [2014, 2016].

Although not a lot of work has been conducted in individual host-based monitoring there is a demand for it as a first step of a cyber-attack usually involves the infection of an individual host of the enterprise network. A host of an enterprise network describes any device that is connected to the network, such as personal computers, servers and printers.

[1] `https://www.otalliance.org/system/files/files/initiative/documents/ota_cyber_incident_trends_report_jan2018.pdf`

An intruder, by taking access to a device, ultimately aims to fully control the network by pivoting through devices of the network. As a device becomes infected, deviations from its normal behaviour occur, for example, an increased number of new connections.

In this manuscript, individual device behaviour is modelled at normal state and an anomaly detection system is presented that works at the level of individual devices within a network. Device behaviour is defined as the observed network traffic involving the device of interest, within a pre-specified time period. Network traffic is collected from observed NetFlow records from the device of interest (as a source device) or where the device has acted as a server (destination device). Each NetFlow record includes information about the time that a connection has been initiated, the duration of the connection, the protocol used, source and destination internet protocol (IP) numbers and ports, the number of packets and bytes exchanged during the connection [Collins, 2014, Kent, 2016]. IPs are numerical labels assigned to devices within networks, where it iis assumed that each IP is assigned to a unique device. For the rest of the manuscript the terms IP and device are used interchangeably.

In a network a variety of patterns of behaviours are observed as different devices and users operate them in quite distinct manners illustrated in Section 2. The illustrated behaviours correspond to devices from a subnet of Imperial College London's network, where NetFlow records over a certain period of time are presented. In Section 3, a number of statistical approaches are presented for modelling future device behaviour to depend on observed historic behaviour through a set of derived features. The features have been derived from the available NetFlow data information and by the observed device behaviour presented in Section 2.

Being able to predict how an individual device will behave in the future can form the basis of an anomaly detection framework that identifies any deviations from this behaviour.

The approaches compared include Poisson and Zero-Inflated Poisson, Regression Trees, Random Forests, Quantile Regression and Quantile Regression Forests. The methods were compared on a device basis, for identifying the optimal model that can be used for the majority of the devices across the network. The performance of the tested models are compared both on devices from Imperial College London's network and on another independent set of devices from the publicly available dataset from Los Alamos National Laboratory [Kent, 2015]. For both networks, the Quantile Regression Forests approach is found to have the best performance (Section 4).

In Section 5 an anomaly detection framework based on the Quantile Regression Forests approach is proposed. The proposed anomaly detection system identifies as anomalies any device behaviour that lies outside of an expected device behaviour prediction interval. One of the critical issues faced in the field of statistical cyber-security is the lack of labelled data. In this manuscript a series of experiments are presented that aimed towards compromising normal device behaviour and creating abnormal deviations (Section 5). The performance of the proposed anomaly detection system on these experiments is compared to two other detection systems, the change point detection system proposed by Killick et al. [2012] and a benchmark detector suggested in this manuscript.

The proposed anomaly detection system aims to capture deviations from the normal activity of a device for an anomaly as indicative of an infected device. In contrast to the network anomaly detection systems that will give a "red signal" for the whole network, the proposed system will answer the crucial question for network analysts, "Which devices need to be closely looked at for securing the rest of the network?".

2 Device Behaviour

NetFlow data collected from Imperial College London (ICL) network was analysed. ICL's network has approximately 40,000 devices connected to it daily. The average level of traffic flow on the network equates to approximately 1.3 billion NetFlow records per day [Heard et al., 2014]. NetFlow records for 55 randomly selected IPs from a subnet of the ICL network were analysed. NetFlow records for 13 consecutive days were included in the analysis. The observed period (T= 18,720 minutes) is discretized into 5 minutes time bins for obtaining a richer presentation of device behaviour.

Device behaviour is defined as the number of NetFlow events assigned to each time bin, both whether the device of interest is found as the source device as well as the destination device. A NetFlow record is assigned to a time bin if it has started, or if it has ended, or if it is a NetFlow record that runs over the time interval as seen previously in Evangelou and Adams [2016]. Individual device behaviour varies dramatically between devices, as a device's behaviour does not only capture the behaviour of the individual user operating the device but also all communications of the device with the network not generated by the user. Figure 1 illustrates the behaviour of four devices for the time period of 13 days. A value of zero indicates a time bin with no NetFlow events assigned to it.

Both *IPs 1* and *31* generated a number of events within the 5 minutes time bins. *IP 1* behaviour suggests bursts of activity that might be suggestive of updates involving the device. No other IP was found to show a similar behaviour to *IP 31.* This IP is highly active on the first days whereas it becomes idle later, which suggests that this device might have been switched off and disconnected from the network (Figure 1 (b)). The maximum number of NetFlow events assigned to a single time bin for 5 minute time bins is 5,870 events. The time series of *IPs 55* and *40* appear to have two processes running in parallel,

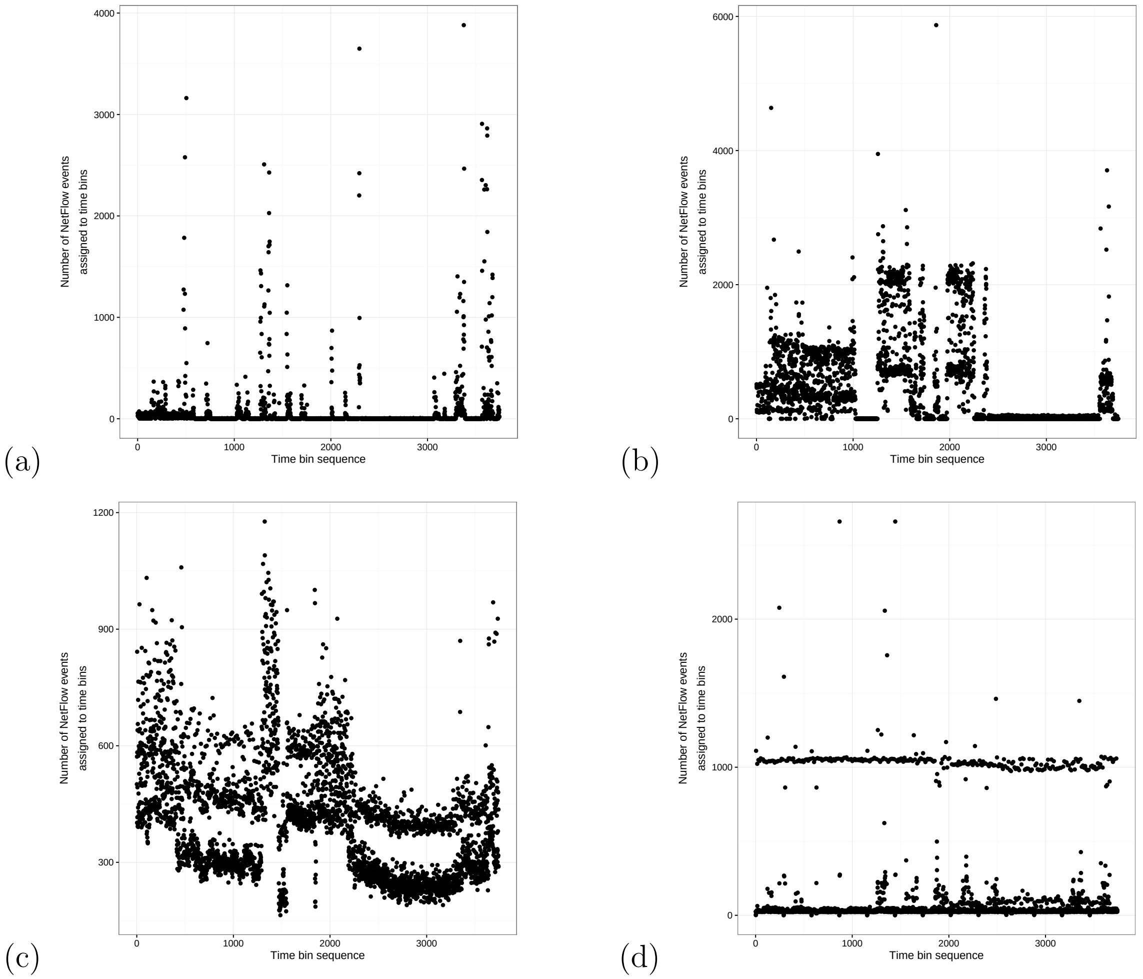


Figure 1: Distinct patterns of device behaviour are observed within a network. Device behaviour is defined as the number of NetFlow events assigned to 5 minute time bins. Device behaviour is illustrated for devices: (a) *IP 1*, (b) *IP 31*, (c) *IP 40*, (d) *IP 55*

with both of these IPs having no empty time bins. More than 3,000 time bins for *IP 55* have between 1-40 NetFlow events assigned to them (Figure 1 (d)).

It is generally unknown what defines normal activity when considering the number of NetFlow events, as one device may generate 5 events per 5 minutes whereas another may generate 100 events per 5 minutes. In addition to this, the two devices illustrated in Evangelou and Adams [2016] show completely different patterns of activity to those presented in Figure 1. A variety of factors can potentially explain a burst of activity of thousands of NetFlow records, including automatic processes such as software updates. Port scanning is another example that generates a large number of NetFlow connections. Port scanning is a procedure associated with the early surveillance stages of a cyber attack, where the intruder searches for open ports that can communicate within the network (Bhuyan et al., 2011)

3 Predicting Device Behaviour

The aim of the conducted work is the prediction of the future activity of an individual device. Being able to know how a device is expected to behave in the future can be used as a measure for identifying any abnormal deviations from this behaviour. The response of interest is defined as the number of NetFlow events assigned to the time bin $t + 1$.

A natural predictor of the response presents the number of currently assigned NetFlow events in time bin t named the Benchmark predictor [Evangelou and Adams, 2016]. Figure 2 presents the Benchmark predictions for the four devices illustrated in Figure 1. Optimal predictions are obtained when the predictions closely match the response, *i.e.* the data points lie close to the diagonal line. Only for *IP 31* the Benchmark predictions are good, with the points of Figure 2(b) being very close to the diagonal.

In addition to the Benchmark predictor, a number of statistical and machine learning models were compared for finding the best model for predicting the future activity of

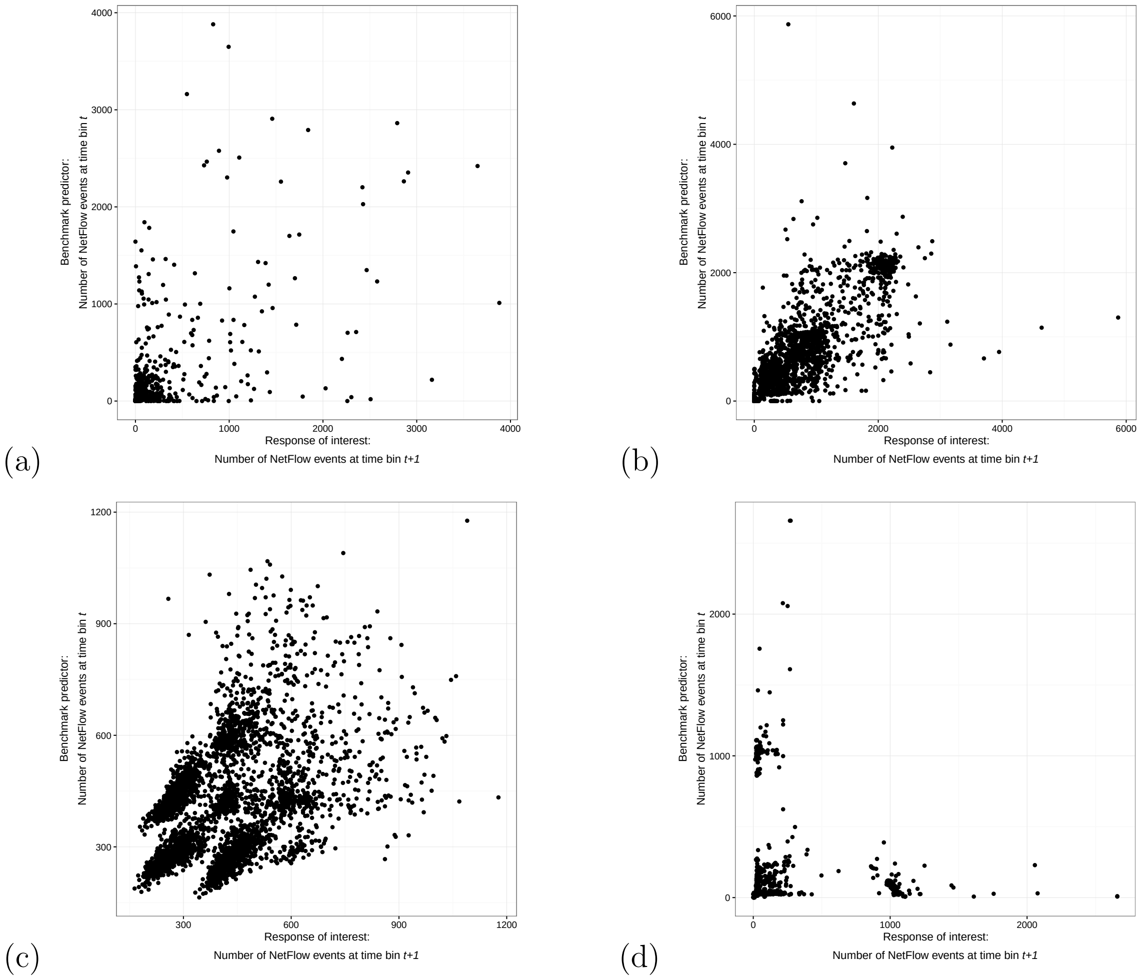


Figure 2: Benchmark predictions for 4 distinct devices: (a) *IP 1*, (b) *IP 31*, (c) *IP 40*, (d) *IP 55*.

each device (these methods/models are presented in Section 3.2). The constructed models depend on a set of features created from the observed historic NetFlow data of each device, *i.e.* the NetFlow events assigned to time bin t.

3.1 Feature Construction

As a number of NetFlow records were assigned to each time bin, a set of features were constructed from the available information of the events. In total 32 features were derived that can be classed into four main categories: *Time*, *Events*, *Characteristics* and *Nature* that describe time bin t. These features that characterise the immediate past behaviour of the device are linked with the future behaviour for better understanding device behaviour.

Time related features include information regarding the time of the time bin, for example start and end times of the time bin. A midpoint of each time bin recorded in minutes and repeated across the 13 days is also included. Through an exploratory analysis of the data, peaks of activity were observed across all devices between the first 20-30 minutes past the hour (Figure S1), therefore a peaks binary indicator was included in the set of features.

The *Events* set of features includes information about the number of events assigned to the time bin. This includes the number of events that only started in time bin and the number of events that only finished but did not start in the time bin. The *Characteristics* set of features includes summary statistics for the characteristics of the observed connections such as bytes, packets, ratio of bytes over packets and duration. A set of features related to the *Nature* of the observed connections is also considered that includes information regarding the entropy of the smallest port, counts of the most frequently observed protocols in the dataset, and a variable counting whether the tested device is a server or not. The complete list of features can be found in Table S2.

For the time bins with no NetFlow events assigned to them, their corresponding *Events*, *Characteristics* and *Nature* features were assigned with a -1. Further, as the features were constructed with no knowledge of what might be related with the response or between them, highly correlated features (with a Pearson's correlation coefficients greater than 0.9) were removed from the set of features for each device [Evangelou and Adams, 2016].

3.2 Predictive Models

This section presents the regression models and machine learning approaches for modelling the relationship between the response y_i and the features vector x_i of time bin i.

Poisson Regression

A natural parametric model is a Poisson regression model as the modelled response takes only positive integer values:

$$p_{Y_i}(y_i) = \frac{\exp(-\mu_i)(\mu_i)^{y_i}}{y_i!}$$

with $\log(\mu_i) = x_i^T\beta$ where x_i is the feature vector of time bin i and y_i presents the response of the same time bin.

As certain IPs exhibit an excess of zeroes, a zero-inflated Poisson (ZIP) model was also considered [Wang et al., 2015]. A ZIP model assumes that there are two zero generating processes and it combines a point mass at zero modelled by a Bernoulli process with a Poisson distribution:

$$p_{Y_i}(y_i) = \begin{cases} \pi_i + (1-\pi_i)'\exp(-\mu_i) & \text{if } y_i = 0 \\ (1-\pi_i)\frac{\exp(-\mu_i)(\mu_i)^{y_i}}{y_i!} & \text{if } y_i > 0 \end{cases}$$

where similarly with the Poisson model $\log(\mu_i) = x_i^T\beta$. The Bernoulli parameter π_i is also assumed to depend on the features such that $\log(\frac{\pi_i}{1-\pi_i}) = x_i^T\gamma$.

Regression Trees

Tree-based approaches, including classification and regression trees are a very popular approach in statistics and machine learning, where the feature space is partitioned and a simple model is fitted to each partition. In the case of Regression Trees (RT), the aim is to minimise the square error at each partition. RT in contrast to the Poisson regression models, include interactions between features. Regression trees (RT) were considered by Evangelou and Adams [2016] where it was shown that the trees perform better than the Benchmark predictor across the tested IPs for both performance errors assessed.

Random Forests (RF) were also considered for predictive device behaviour. RF build a large number of regression trees and then averages over them [Breiman, 2001]. The R libraries ***rpart*** and ***randomForest*** were used to fit RT and RF respectively, with the default settings of the corresponding functions used.

Quantile Regression

A τ^{th} quantile of a population is the number that indicates where the $\frac{100}{\tau}\%$ proportion of the population lies. For example, the median statistic is the 2^{nd} quantile and presents the middle value of the population that splits the sample into two sets of observations. Quantile Regression (QR) was introduced by Koenker and Bassett [1978] for the estimation of models in which the quantiles of the response are modelled to depend on the features. Through quantile regression the τ^{th} conditional on the features quantile is computed. For the purposes of this work the conditional median response of each device was modelled.

Linear programming approaches can be implemented for computing the τ^{th} conditional quantile. Quantile Regression Forests (QRF) proposed by Meinshausen [2006] combine the ideas of RF and QR. QRF is a tree-based approach for estimating conditional quantiles in contrast to the linear programming approaches. The key difference between RF and QRF is that for each node in each tree, RF only keeps the mean of the observations that lie in the node, whereas QRF hold the values of all the observations of the node [Meinshausen, 2006]. Similarly to QR, QRF was implemented for modelling the conditional median response of each device. The R libraries ***quantreg*** and ***quantregForest*** were used to fit QR and QRF respectively, with the default settings of the appropriate functions used.

3.3 Assessing the predictive accuracy of the methods

The predictive accuracy of the models was investigated by dividing the data into training and test sets. The models were firstly fitted to the training set and their predictive performance was assessed on the test set. The performance of the predicted responses ($\hat{y}_i$) in relation to the observed responses (y_i) of the test set were assessed by computing the square error (SE):

$$(\hat{y}_i - y_i)^2,$$

the absolute error (AE):

$$\frac{|\hat{y}_i - y_i| + 1}{|y_i| + 1},$$

and the symmetric version of the absolute error (SAE):

$$\frac{|\hat{y}_i - y_i| + 1}{|\frac{\hat{y}_i + y_i}{2}| + 1}.$$

Both the SE and the SAE equally penalise both situations of $\hat{y}_i > y_i$ and $\hat{y}_i < y_i$ whereas the AE assigns higher error to the former situation. As different behaviours have been observed across the IPs, both the mean and median summary statistics of the errors were computed across all samples of the test set, resulting to a total number of six computed errors. MSE, MAE and MSAE correspond to the mean versions of the three errors.

3.4 Application: ICL network

Prediction Accuracy

The data were split into training and test sets with the first 7 days in the training set and the last six days in the test set. Method comparison was done both at a device level and across all devices.

At a device level, the methods were ranked from 1 to 7 according to whether the method had the smallest prediction error, the second smallest and so-on. Ranks were computed for all six errors. Ties, *i.e.* methods with the same error, were given the smaller rank. Table 1 presents the mean ranking of the tested methods across the 55 IPs. The ZIP models did not fit for 13 of the 55 devices. For those IPs the ZIP method was penalised with a rank equal to 8. For the 5 minute time bins, it was observed that the QRF approach performs very robustly across the 55 IPs, as illustrated by the smaller ranks for Median AE, Mean and Median SAE. For the other three errors, the QRF is the second or third best method.

Error	Benchmark	Poisson	ZIP	RT	RF	QR	QRF
MSE	5.7273	3.6000	4.2182	2.7818	4.6000	3.3455	3.7091
MedianSE	1.7636	4.9273	5.2545	4.3455	6.4727	2.6909	1.8909
MAE	3.0727	4.6727	5.2364	4.9818	6.3091	1.6000	2.1091
MedianAE	1.6545	4.5455	5.0000	5.3636	6.4727	2.5455	1.2727
MSAE	2.8909	4.9091	5.5091	4.3818	6.4000	2.2364	1.6727
MedianSAE	1.6545	4.6545	5.1273	5.2545	6.3455	2.4909	1.1636

Table 1: The errors of the methods are ranked from the smallest to the largest for each device of the ICL network. The ZIP approach did not fit for 13 devices. The sample mean of the ranks across all devices are presented.

In addition the test set samples for each device were combined into a single test set of length 94,930 ($L = 55 \times 1726$) and all six errors were computed. Both Poisson and ZIP models have very large MSE and MAE errors, suggesting that there are outliers, either in the predictions or observed values, that lead to these extreme errors. The ZIP is found to have the smallest recorded Median SE. The QRF approach is found to minimise the mean

SE and SAE, and both median AE and SAE alongside the Benchmark predictor and QR.

Error	Benchmark	Poisson	ZIP	RT	RF	QR	QRF
MSE	10099.6822	Inf	Inf	9435.4471	9097.8144	66677.5533	8332.3215
MedianSE	1.0000	1.0326	0.3127	1.5673	2.2107	1.0000	1.0000
MAE	1.3005	Inf	Inf	2.0924	5.6682	1.1889	1.2309
MedianAE	1.0000	1.2249	1.3912	1.4164	1.6419	1.0000	1.0000
MSAE	0.9338	0.9879	1.0782	0.9903	1.0425	0.9270	0.9002
MedianSAE	1.0000	1.1630	1.1713	1.1744	1.2260	1.0000	1.0000

Table 2: Computed errors across all 55 devices with the test set combined as a single set.

Both at a device level and across all tested devices the QRF approach was found to minimise the majority of the prediction errors. These findings suggest that QRF is the best predictive approach for modelling device behaviour.

Selected Features

A secondary objective of the conducted analysis is the identification of features related to the response of interest. Similarly to RF, QRF also provides information about the importance of selected features across the trees. For each device, the selected features of QRF were ranked according to their importance. Tables 3 and S2 show the summary statistics of the observed rankings across the 55 IPs.

The features selected for all 55 tested devices include the *Time* related features: 1) start time, 2) midpoint, 3) week day indicator, 4) peaks indicator, 6) working hours indicator, with only the first two to have small ranks. In addition the *Events* related features: 1) number of events, 2) number of events that have only started in the time bin and 3) the number of events that have only stopped in the time bin, were selected for all 55 devices with only the first one to have a small rank. The sample mean of duration of the events was also selected by all devices.

Features with small ranks include the entropy of the smallest port, the sum of bytes and packets exchanged in the 5 minute bins. Also, the sum, SD and mean value of the ratio between bytes and packets exchanged during a time bin (Tables 3 and S2).

The list of features selected by the QRF approach is similar to the one reported by Evangelou and Adams [2016] chosen by the RT that included the smallest port entropy, the midpoint of each time bin and the sum of bytes. These findings suggest that *Time* related features play an important role in predicting device behaviour suggesting that there are seasonality patterns seen within each device. Characteristics that describe the observed connections (*bytes, ratio of bytes over packets, entropy of ports*) are selected highlighting that device behaviour is affected by them.

In the following section the analysis of the publicly available dataset released by Los Alamos National Laboratory is presented.

4 Los Alamos National Laboratory

Kent [2015] released anonymised data from multiple cyber-security data sources including NetFlow from the Los Alamos National Laboratory (LANL) network. A full description of the dataset can be found in Kent [2016]. The published anonymised NetFlow dataset includes information about the start time, duration, source and destination device and port numbers, bytes and packets exchanged at each NetFlow record. NetFlow events are recorded to the level of accuracy of 1 second. The data for 191 unique devices for the first 10 days from the LANL network were extracted and analysed. LANL's NetFlow data only includes connections between devices of the network. This is in contrast to the NetFlow

Feature	N	Mean	SD	Median	MAD
Start time of time bin	55	2.9455	2.8830	1.0000	0.0000
Middle value of time bin	55	3.7455	3.7477	2.0000	0.0000
Sum of bytes	54	3.8704	1.5546	3.0000	0.0000
Entropy of the smallest port	19	4.0000	1.9720	4.0000	1.4826
Sum of the ratio of bytes over packets	21	4.2857	1.0556	4.0000	0.0000
Sum of packets	3	6.3333	4.5092	6.0000	5.9304
Number of NetFlow events	55	6.6000	3.5984	7.0000	4.4478
Mean ratio of bytes over packets	50	6.7600	3.0140	6.0000	2.2239
IP is a Server	50	7.0200	3.6949	5.0000	1.4826
SD of ratio of bytes over packets	47	7.3404	2.3338	7.0000	1.4826

Table 3: The 10 most important features across all 55 devices are presented. Features were ranked according to their importance for each tested device. Summary statistics of these ranks across all 55 devices of ICL network are presented.

data of ICL that includes connections of the network devices both within and outside of the network.

For the LANL dataset device behaviour is defined as the number of NetFlow records with a start time within the 5 minutes time bin. Similar features were created to the ones described in Section 3.1 including: the start time of each 5 minute time bin, the midpoint of each time bin, an indicator whether the time bin is within working hours or not (working hours defined between 8am-6pm), the number of events and number of events with duration less than 5 minutes (completed events). Summary statistics (sample mean, median, standard deviation and median absolute deviation) for duration, packets, bytes and the ratio of bytes over packets. The number of unique source and destination devices the tested device is connected to in each 5 minute time bin. Number of events with certain protocol numbers 1, 6 and 41. The entropy of the source and destination ports are included. The complete list of 31 features constructed is given in the Table S1. As with the previously conducted analyses, features that were highly correlated were extracted (Pearson's correlation coefficient $\rho \geq 0.9$) and the remaining features were scaled and centred before fitting any models. The first 7.5 days were treated as the training set and the 2.5 days as the test set.

Error	Benchmark	Pois	ZIP	RT	RF	QR	QRF
MSE	3.8429	4.9372	5.4607	2.8639	4.0576	3.7330	2.4241
MedianSE	3.0785	4.6911	5.3560	3.5812	5.5812	2.4869	1.8639
MAE	3.6126	5.1466	5.5288	3.4188	5.3560	2.4450	1.7539
MedianAE	2.6492	4.6335	5.1832	3.6492	5.3874	2.6178	1.8377
MSAE	3.9843	4.6492	5.3141	3.5183	5.1257	2.8272	1.8377
MedianSAE	3.4869	4.7644	5.1675	3.7696	4.9267	2.4241	1.5969

Table 4: The errors of the methods are ranked from the smallest to the largest for each device of the LANL network. The ZIP approach did not fit for 79 devices and QR did not fit for 74 devices. The sample mean of the ranks across all devices are presented.

The performance of the approaches was assessed by looking at the corresponding ranks of the approaches for the 191 devices as illustrated in Table 4. Similar to the analysis of the ICL network system, the QRF approach was found to be the best performing one by minimising the computed prediction errors.

The features selected by the QRF approach across the 191 devices were also explored. The most important features selected include the start time of the bin and the number of events in time bin t. Other features selected by the majority of devices include the midpoint of each time bin, the working hours indicator and the numeric value of the days, similarly to the results of the ICL network analysis. Other features with small ranks include the entropy of the source port, the average number of bytes, ratio of bytes over packets and duration, the standard deviation of bytes (Table S2).

5 Anomaly Detection System

The main goal of the statistical cyber-security field is the development of anomaly detection systems. Systems that detect any abnormal deviations from the normal activity and can be used to detect and prevent damage caused by cyber attacks. In the previous sections it was shown that the QRF model is the best performing one for predicting individual device behaviour. Our findings have been reproduced on two distinct enterprise networks. Figure 3 illustrates the QRF conditional median predictions for *IPs 1, 31, 40* and *55* of the ICL network, where for the first three IPs, the QRF forecasts are found to match the observed response for most time bins. For *IP 55* the QRF identifies mostly one of the two underlying processes.

Both QR and QRF can be used to build prediction intervals, where for every new observation of the response variable there is a high probability that it lies within the prediction interval [Meinshausen, 2006]. An anomaly detection system is proposed that utilises the prediction intervals computed through the QRF fitted models. The prediction intervals are computed through the calculation of the chosen conditional quantiles. For example, the lower point of the 95% prediction interval is the 2.5% conditional quantile, whereas its upper limit is the 97.5% conditional quantile.

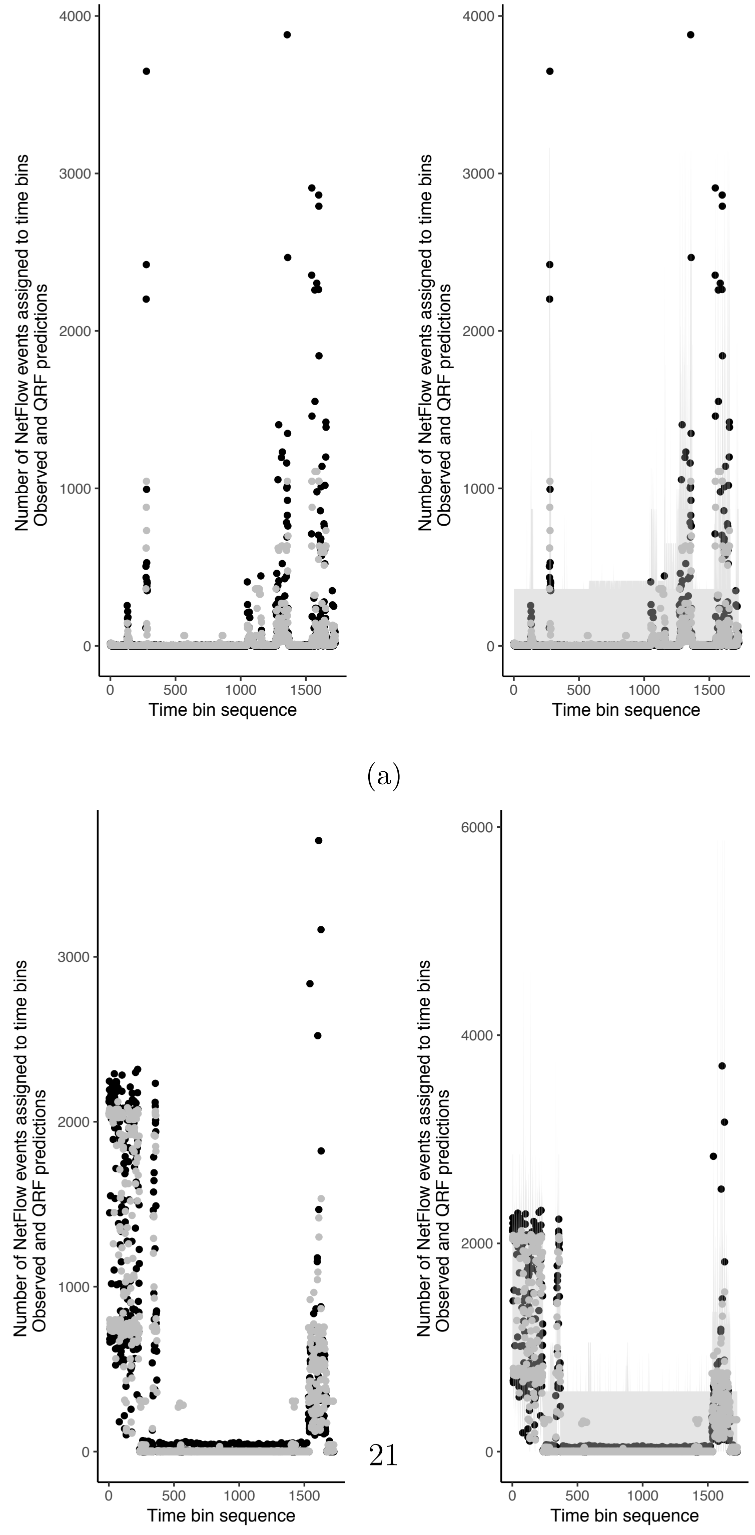
5.1 Proposed Anomaly Detector

The proposed anomaly detection system assumes a uniform distribution with endpoints as the lower and upper limits of the computed prediction intervals. Any observations outside of these limits are considered as abnormal. The observed response for time bin t is abnormal if either case is true:

$$\begin{cases} y_{observed} < y_{\eta/2\%\, level} \\ y_{observed} > y_{(100-\eta/2)\%\, level} \end{cases}$$

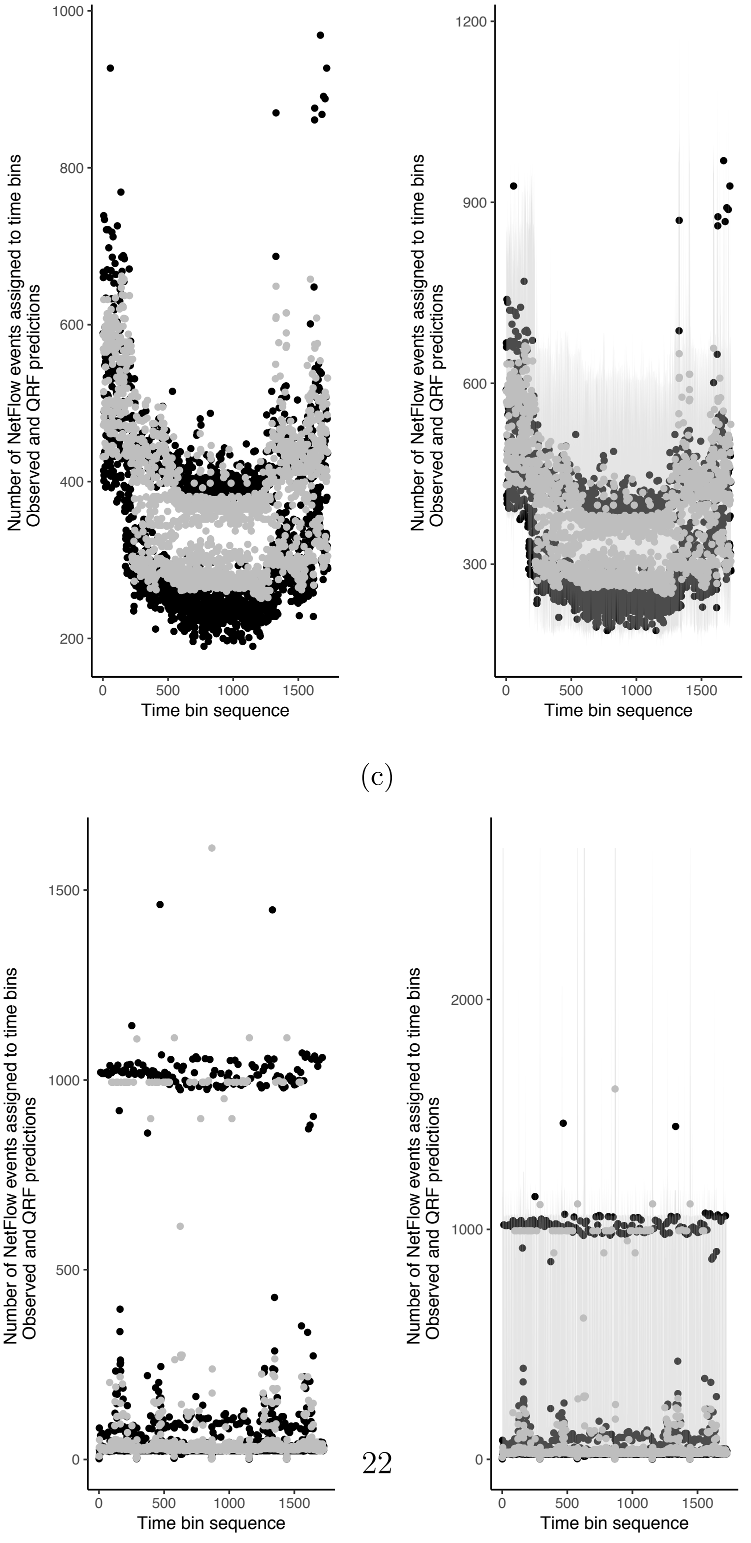
where η $(0 \leq \eta \leq 100\%)$ presents the chosen quantile level. The limits $y_{(100-\eta/2)\%\, level}$ and $y_{\eta/2\%\, level}$ represent the upper and lower conditional quantiles, respectively. A small chosen value of η will lead to a larger number of time bins predicted as abnormal.

Figure 3 illustrates the computed quantile conditional prediction intervals for $\eta = 95\%$. As expected most of the observed activity is included in the prediction intervals. This is expected as they were no reported cyber attacks occurred during the studied period. The prediction intervals for *IP 55* also include the second underlying process of the device. The proposed anomaly detection system has identified 10, 8, 7 and 14 abnormal time bins for *IPs 1, 31, 40* and *55*, respectively.


Number of NetFlow events assigned to time bins
Observed and QRF predictions
Time bin sequence
0
1000
2000
3000
4000
6000
500
1500

(a)

(b)



(c)

(d)

Figure 3: QRF predictions highlighted in grey with corresponding 95% prediction intervals. Observed response is illustrated in black. (a) *IP 1*, (b) *IP 31*, (c) *IP 40*, (d) *IP 55*.

5.2 Validation of the proposed detector

Deviations from the normal behaviour of a device are usually observed as part of a cyber attack. The intruder takes control of the infected device while the user of the device continues to use the device as before, therefore a mixture of the two behaviours is observed.

For validating the proposed anomaly detection system, a simulation study was conducted where deviations from the normal behaviour of a device were imposed. The normal behaviour of a device was modified to include a mixture of the normal behaviour of the device of interest with an abnormal behaviour: a device behaving in a different pattern from the device of interest. Both cases of changing only the response variable as well as changing both the response variable were explored and the corresponding features of each time bin. The first case corresponds to the phenomenon where the intruder runs a number of additional NetFlow events with similar characteristics as to the ones that the normal user runs (*e.g.* the features vector remains the same). Figure 4 illustrates an example of the carried validation experiments, where the left-hand side plot presents the normal behaviour of the device of interest. The right hand side plot presents the abnormal now behaviour of the device as 10% of it has been mixed with the device behaviour seen in the middle graph.

The performance of the proposed anomaly detection framework was compared to two alternative approaches. The first approach considered is the change-point detection algorithm proposed by Killick et al. [2012] called Pruned Exact Linear Time (PELT) method. Any changes to the mean value of the response variable for the test set are reported. The R library ***changepoint*** was used to run PELT with the default settings of the appropriate function used.

The second approach considered is a Benchmark anomaly detector that relies on intervals computed based on the quantiles of the observed device behaviour in the training set.

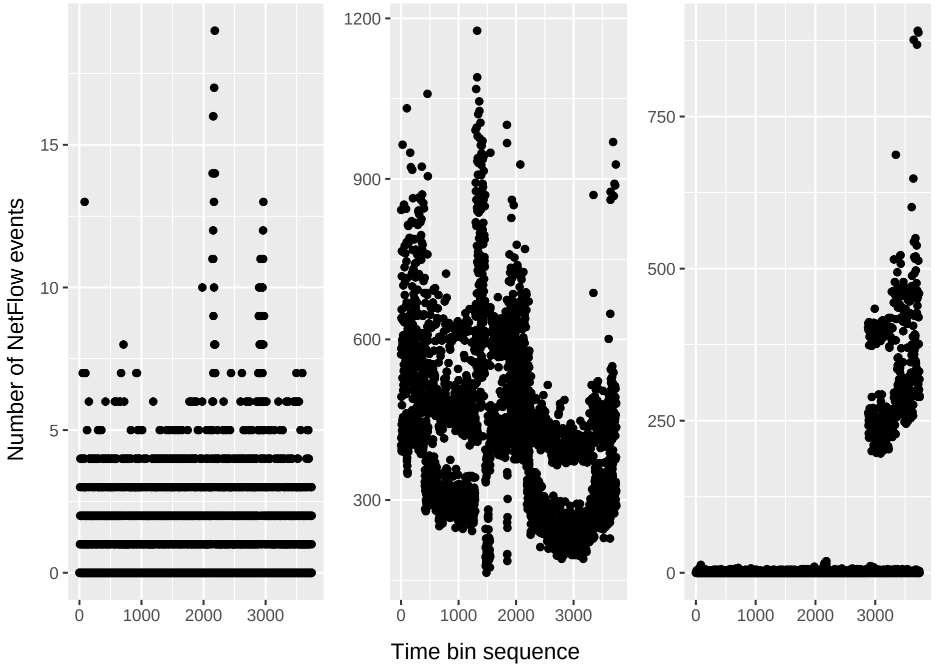


Figure 4: Mixing of two different device behaviours - 50% mixing. Mixed behaviours: *IP 29* and *IP 40* (ICL network)

Suppose that for an individual device for its first 10 days a response vector is observed with an $\eta/2\%$ quantile equal to b, and a (100-$\eta/2$)% quantile equal to a, any test set responses outside of the interval between a and b are considered abnormal. This interval is constant across all test set time bins as it depends only on the observed device behaviour recorded across the training set. Both the Benchmark predictor and PELT depend on the observed response and they do not take into account the features of each time bin as the proposed anomaly detector does. Therefore the second scenario considered where both features and response are changed does not affect the performance of these two detection systems and their performance is not affected by altering or not the features vector.

For both the proposed anomaly detection system and the Benchmark anomaly detection, any observed responses outside of the computed 95% quantile intervals are assigned with a p-value equal to zero and are considered to be abnormal. In contrast to the proposed anomaly detection system, the Benchmark anomaly detector has the same 95% quantile intervals for all time bins whereas the prediction intervals of the QRF are different for each time bin, as they depend on the features of each time bin.

For the validation experiments devices from the ICL network were utilised. Ten of 13 days were treated as the training data for the device of interest, with the last 3 days treated as the test data. The test data include time bins with response variables coming from a mixed distribution of the two devices. A mixing proportion was used that controlled how many abnormal time bins were included in the test set. Figure 4 illustrates an example of mixed behaviours, where half of the values of the response variable of the third graph come from the first device (*IP 29*) and the other half from the "contaminating" device (*IP 40*). The described process was repeated 100 times for each combination of two devices and for each mixing proportion considered.

Table 5 presents the average accuracy, false discovery rate (FDR) and true positive rate (TPR) of the three detection systems for 100 trials. For both cases of admixing (only the response variable and both features and response) the proposed anomaly detection framework performs better than the other two compared detection systems. Further, as more time bins are affected by the abnormal behaviour the performance of the method improves. The robustness of the method was validated across other combinations of device behaviours (Table S5).

Mixing Proportion	Approach	Accuracy	FDR	TPR
$\pi = 0.1$	Benchmark	0.9594	0.2893	1.0000
	PELT	0.8944	0.5144	1.0000
	QRF	0.9757	0.1962	1.0000
	Changing both features and response			
	QRF	0.9757	0.1962	1.0000
$\pi = 0.5$	Benchmark	0.9803	0.0379	1.0000
	PELT	0.7278	0.3495	0.9860
	QRF	0.9919	0.0160	1.0000
	Changing both features and response			
	QRF	0.9919	0.0160	1.0000
$\pi = 0.8$	Benchmark	0.9895	0.0129	1.0000
	PELT	0.8053	0.1755	0.9609
	QRF	0.9954	0.0057	1.0000
	Changing both features and response			
	QRF	0.9954	0.0057	1.0000

Table 5: Average accuracy, false discovery rate (FDR) and true positive rate (TPR) across 100 trials for different mixing proportions. π presents the mixing proportion, for example $\pi = 0.80$ corresponds to 80% of abnormal time bins. Mixed behaviours: *IPs 29* and *40*

6 Discussion

This manuscript focuses on individual hosts of enterprise networks as most attacks start by the infection of individual hosts. The conducted work comprises of two parts. In the first part of the study, a number of statistical and machine learning models are compared for identifying the optimal one for predicting device behaviour. The Quantile Regression Forests model was found to outperform the other approaches considered across the tested devices of two independent networks by minimising the computed prediction errors.

QRF advantageous over the other approaches for both the feature selection that it performs as well as the construction of prediction intervals. QRF performs variable selection by raking the features according to their importance on constructing the trees of the random forests approach implemented. The QRF selected features for the two networks include the start time and midpoint of each time bin, the number of events currently being observed, the entropy of the destination (server) port and the summary statistics of bytes and the ratio of bytes over packets (Tables 3, S2 and S4). These findings further emphasise that future device behaviour depends on time and features related to the nature and characteristics of the connections, outperforming the Benchmark predictor (defined as the number of events currently being observed). The second advantage of QRF is that it computes prediction intervals. Prediction intervals include the uncertainty of an individual new observation whereas confidence intervals provide uncertainty about the average response given the features.

The second part of the presented work focused on the development and validation of a data-driven anomaly detection system. Motivated by this a number of novel experiments for validating the proposed system are presented. As one of the challenges of the statistical cyber-security field is the lack of properly labelled data, *i.e.* whether the network is under an attack or not, or in this case if a device has been compromised or not, a series of novel experiments are presented in the manuscript for creating such data. The experiments involve the contamination of normal device behaviour with the behaviour of a different device. This is based on the expectation that as an intruder gets control of the machine, the user (or users) operating the device will not be aware of the device being infected and will continue operating as they usually do. Therefore in the conducted experiments different mixtures of contamination and normal behaviour were tested. The proposed anomaly detection system is based on the prediction intervals of QRF. Any time bins with observed responses outside of the corresponding computed prediction intervals are marked as anomalous. The proposed detection system when considered alongside alternative approaches, the named Benchmark anomaly detector and PELT, is found to outperform them suggesting that the proposed anomaly detection system is a promising system for operational implementation.

Through the experiments the anomaly detector identified any behaviour outside of the 95% prediction intervals as anomalous. The choice of the size of the prediction intervals depends on the operational capabilities of the network. Larger prediction intervals will mean that fewer time points per device will be identified as anomalous. As the proposed anomaly detection framework will run automatically for each device, the choice of the prediction intervals can be decided based on the status of each device. There are devices in an enterprise network that are more valuable assets than others. For these devices network analysts can consider setting more strict rules for identifying any abnormalities. The network analysts may also consider to flag abnormalities only above the upper limit of the prediction interval as this will also be more indicative of malicious activity or of a more abnormal behaviour than a device being turned off, or its frequent user generating a fewer number of NetFlow records.

An attack might not be apparent in only a single data source and vice-versa an anomalous behaviour in a single data source might not be indicative of a cyber attack. This suggests the need for a data fusion where information from multiple data sources are combined for identifying abnormal deviations (examples include the work of Whitehouse et al. [2016] and Turcotte et al. [2016]). Further developments of the proposed work include the extensions to multiple data sources and their combination for improving the chances of identifying an abnormal behaviour.

A novel statistical data-driven based anomaly detection system is proposed, that outperforms other detection systems, in spotting deviations from the normal activity of a device and subsequently of a network. As existing cyber-security defence systems are found to be inadequate in protecting enterprise networks, the development of data driven anomaly detection systems that can complement the existing defence systems is needed for improving the chances of identifying cyber attacks.

7 Conclusions

A novel statistical data-driven based anomaly detection system is proposed for spotting deviations from the normal activity of a device and subsequently of a network. The anomaly detection is based on a model that uses Quantile Regression Forests to characterise the relationship of future device behaviour with historic behaviour. Historic behaviour is described by a set of features derived from the traffic events of the device. The derive features characterise the behaviour of individual devices can be utilised for other analyses, as for example, for identifying clusters of similarly behaving devices. To our knowledge, this is the first study that illustrates the effectiveness of Quantile Regression Forests for the analysis of cyber-security datasets. The developed model is shown to have great performance in predicting device behaviour, even though individual devices have such distinct patterns between them. The developed model was found to have overall very good predictive performance when analysing devices from two distinct enterprise networks. This illustrates the model's robustness and reproducibility, which is a significant finding supporting the employment of the developed model as a novel anomaly detection system. In addition, a series of experiments are described in the manuscript for testing the performance of the proposed anomaly detection algorithm by contaminating normal individual device behaviour. In the experiments performed, the proposed anomaly detection algorithm is found to outperform two other detection algorithms, one of them being the widely known change-point detection algorithm proposed by Killick et al. [2012]. The experiments are designed in such a way to mimic what is currently being observed in recent attacks where intruders start by infecting individual devices in order to obtain full control of a network. As the existing cyber-security defence systems are found to be inadequate in protecting enterprise networks, the development of data driven anomaly detection systems that can complement the existing defence systems, such the one proposed in this manuscript, is needed for improving the chances of identifying cyber attacks.

References

N. M. Adams and N. A. Heard. *Data analysis for network cyber-security.* Imperial College Press, 2014.

N.M. Adams and N.A. Heard. *Dynamic networks and cyber-security.* World Scientific, 2016.

L. Breiman. Random forests. *Machine Learning*, 45 (1):5–32, 2001.

M. Collins. *Network Security through data analysis.* O'Reilly Media, Inc, 1005 Gravenstein Highway North, Sebastopol, CA 95472, 2014.

M. Evangelou and N. M. Adams. Predictability of NetFlow data. *IEEE Intelligence and Security Informatics*, 2016.

C. Gates, N. Li, Z. Xu, S.N. Chari, I. Molloy, and Y. Park. *Detecting Insider Information Theft Using Features from File Access Logs*. Springer, Cham, 2014.

N. Heard, P. Rubin-Delanchy, and D. Lawson. Filtering automated polling traffic in computer network flow data. *IEEE Joint Intelligence and Security Informatics Conference*, 2014.

A. Juvonen, T. Sipola, and T. Hämäläinen. Online anomaly detection using dimensionality reduction techniques for http log analysis. *Computer Networks*, 91(Supplement C):46 – 56, 2015.

A. D. Kent. Cyber-security data sources for dynamic network research. In N. M. Adams and N. A. Heard, editors, *Dynamic networks and cyber-security*, chapter 2, pages 37–64. World Scientific, 2016.

A.D. Kent. Comprehensive, multi-source cyber-security events. *Los Alamos National Laboratory*, 2015.

R. Killick, P. Fearnhead, and I.A. Eckley. Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association*, 107(500):1590–1598, 2012.

T. Kimura, K. Ishibashi, T. Mori, H. Sawada, T. Toyono., K. Nishimatsu, A. Watanabe, A. Shimoda, and K. Shiomoto. Spatio-temporal factorization of log data for understanding network events. In *Proceedings - IEEE INFOCOM*, pages 610–618. Institute of Electrical and Electronics Engineers Inc., 2014.

A. Kind, M. P. Stoecklin, and X. Dimitropoulos. Histogram-based traffic anomaly detection. *IEEE Transactions on Network and Service Management*, 6 (2), 2009.

R. Koenker and G. Bassett. Regression quantiles. *Econometrica*, 46 (1):33–50, 1978.

A. Lakhina, M. Crovella, and C. Diot. Characterization of network-wide anomalies in traffic flows. In *Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement*, IMC '04, pages 201–206, New York, NY, USA, 2004. ACM.

H. Liao, C.R. Lin, Y. Lin, and K. Tung. Intrusion detection system: A comprehensive review. *Journal of Network and Computer Applications*, 36(1):16 – 24, 2013.

N. Meinshausen. Quantile Regression Forests. *Journal of Machine Learning Research*, 7: 983–999, 2006.

J. Neil, C. Hash, A. Brugh, M. Fisk, and C.B. Storlie. Scan statistics for the online detection of locally anomalous subgraphs. *Technometrics*, 55(4):403–414, 2013.

A. Patcha and J. Park. An overview of anomaly detection techniques: Existing solutions and latest technological trends. *Computer Networks*, 51:3448–3 470, 2007.

E. Riddle-Workman, M. Evangelou, and N. Adams. Adaptive anomaly detection on network data streams. 2018.

C. Schon, N. Adams, and M. Evangelou. Clustering and monitoring edge behaviour in enterprise network traffic. *IEEE Intelligence and Security Informatics*, 2017.

M. Turcotte, J. Moore, N. Heard, and A. McPhall. Poisson factorization for peer-based anomaly detection. *IEEE Intelligence and Security Informatics*, 2016.

Z. Wang, S. Ma, and C. Y. Wang. Variable selection for zero-inflated and overdispersed data with application to health care demand in germany. *Biometrical Journal*, 57:867–884, 2015.

M. Whitehouse, M. Evangelou, and N. Adams. Activity-based temporal anomaly detection in enterprise-cyber security. *IEEE Intelligence and Security Informatics*, 2016.