SAND2011-1610P

Statistical Modeling of Malware Behavior to detect New Threats
Julie Ard

I. Motivation

The Stuxnet worm presented a unique challenge in detection world-wide. It remained undetected for six months because it did not follow known patterns of propagation and selectively infected only the hosts it was designed for(Matrosov). Stuxnet has been described as the world's first precision-guided cyber munition and is expected to influence future malware. My goal is to test whether statistical methods can detect never-before-seen malware.

II. Problem Statement

The main methods of malware detection fall in to two basic categories: signature scanning and anomaly detectors. Signature scanning does not protect against zero-day attacks, while anomaly detectors are plagued with high false alarm rates and can be fooled by gradual changes in activity.

III. Related Work

Stuxnet has been analyzed in detail by ESET and Symantec in the reports listed in the references section. Because Stuxnet has been so thoroughly analyzed by security companies, signatures for each variant have been identified and incorporated in to AntiVirus software. Security firms account for future threats through their ability to detect polymorphic variants of the Stuxnet worm. Additionally, the network and file system behavior of Stuxnet has been documented but, to the author's knowledge, has not been used in the manner outlined in this report.

IV. Methodology

My methodology used known malicious and benign software behavioral data to derive coefficients based on several different models, validate them, and then apply them to Stuxnet behavioral data. After data was split in to training and test data partitions, coefficients were derived using seven different models. Model validation consisted of multiplying the test data matrix with the coefficients and analyzing the test response vector. Stuxnet testing was then performed by multiplying the coefficients with the four Stuxnet data records which were not present in the training or test data.

IV.a. Model Discussion

Seven behavior models were developed. The first uses multilinear regression, and the others are variations on a generalized linear model using the Normal, Binomial, and Poisson distributions, both with and without constant coefficients. Gamma and Inverse Gaussian distributions were also available but would not run because the response vector may include negative responses. For all models, the vector of predicted responses consisted of 1's for known malware and 0's for benign files.

The multilinear regression model outputs coefficients for each variable using multilinear regression. If $X_1,\ldots,X_{10}$ are the data vectors for each variable and Y is the predicted response vector, the linear regression model will output coefficients $b_1,\ldots,b_{10}$ such that

$$b_1*X_1 + b_2*X_2 + \ldots + b_{10}*X_{10} = Y$$

The generalized linear models output coefficients for each variable (plus a constant term, if used) for a generalized linear regression using the specified probability distribution. We used the normal, binomial, and Poisson distributions, both with and without constant coefficients. In the case that constant coefficients were not used, the above relation is accurate but with the derivation of the coefficients differing according to the distribution. In the cases that constant coefficients were used, if $X_1,\ldots,X_{11}$ are the data vectors for each variable and Y is the predicted response vector, the linear regression model will output coefficients $b_0,b_1,\ldots,b_{11}$ such that

$$b_0 + b_1*X_1 + b_2*X_2 + \ldots + b_{11}*X_{11} = Y$$

In summary:

Model ID	Description	Distribution	Constant Coeff
MLR	Multilinear Regression	n/a	No
GLMN	Generalized Linear Model	Normal	No
GLMB	Generalized Linear Model	Binomial	No
GLMP	Generalized Linear Model	Poisson	No
GLMNC	Generalized Linear Model	Normal	Yes
GLMBC	Generalized Linear Model	Binomial	Yes
GLMPC	Generalized Linear Model	Poisson	Yes

Table 4-1: Model Summary

IV.b. Data

The benign samples consist of Word and pdf documents, Windows system executables, and a variety of Portable Apps. The malicious samples were obtained from Arbor Networks, a proprietary database of known malware. All of these were analyzed by a proprietary tool based on an analysis engine which runs a piece of suspected malware on a wide variety of commercial and governmental off the shelf tools and provide results to security analysts(Van Randwyk, Chiang and Lloyd). I obtained 32 Stuxnet samples from Offensive Computing and found that 4 of them were analyzable using this tool.

The database consists of 5398 total samples. 4697 were known malware, 4 were Stuxnet, and 697 were benign. One thing that is important in deriving the coefficients is to not let the large quantity of malware dominate over the smaller number of benign samples. To that end, before processing the data each time, I took a random sampling of the malware, so that I had equal numbers of malicious and benign files to draw statistics from.

The data is first split in to training data and test data. The extraction of test data from training data was random based on a random permutation of indices. 90% of the data was specified to train the model and 10% to be used for testing. Because there was much more malicious data than benign data, early models were biased towards malicious scoring, so the script was then constrained to use the same number of records for both malware and benign data. Accordingly, all benign data was used with the same number of further randomly selected malicious data. For ease of analysis, the test data consists of two partitions of random sets of known malicious and benign files. This does not affect the validation results since "testing" consists simply of matrix multiplication between each test data row and the coefficient column.

IV.c. Variables

These ten variables are used in the models:

# of files created or modified in the C:/WINDOWS directory (excluding system32)
# of files created or modified in the C:/WINDOWS/system32 directory
# of files created or modified in the C:/Program Files directory
# of files created or modified in the C:/Documents and Settings directory
# of files created or modified in the root C:/ directory
# of registries read, created, or modified
# of DNS queries
# of tcp connections
# of http connections
# of udp connections

Although the outputs of several AntiVirus tools, a Rootkit Revealer, and Autorun detection were available, we decided to exclude these results as they would highly influence our model and diminish its ability to predict new threats.

The descriptive statistics for malicious, benign, and stuxnet raw data appear in the following tables.

Malicious	**Max**	**Min**	**Mean**	**Std**
WINDOWS	447	0	3.1467	26.8818
system32	48	0	0.4935	2.4674
Program	111	0	1.0089	8.0935
Docs and Set	103	0	3.3072	5.0403
C:/	613	0	8.3596	38.6070
Registries	1314	0	77.3181	186.2408
DNS	247	0	1.8473	12.6743
TCP	2	0	0.0009	0.0413
HTTP	149	0	1.5024	4.7356
UDP	17	0	0.1880	0.7854

Table 4-1: Malicious Data Descriptive Statistics by Variable

Benign	**Max**	**Min**	**Mean**	**Std**
WINDOWS	7	0	0.9971	0.5723
system32	49	0	0.1349	1.8995
Program	88	0	0.1994	3.6114
Docs and Set	23	0	3.1277	2.7122
C:/	54	0	5.0115	3.6758
Registries	109	0	0.9010	6.5985
DNS	3	0	0.0230	0.1993
TCP	4	0	0.0201	0.2265
HTTP	3	0	0.0115	0.1604
UDP	3	0	0.0100	0.1363

Table 4-2: Benign Data Descriptive Statistics by Variable

Stux	**Max**	**Min**	**Mean**	**Std**
WINDOWS	8	5	7	1.4142
system32	5	2	3.5	1.2910
Program	0	0	0	0
Docs and Set	5	0	1.5	2.3805
C:/	4	0	2	2.3094
Registries	36	12	20.75	10.5
DNS	3	1	2.5	1
TCP	3	0	1	1.4142
HTTP	0	0	0	0
UDP	4	1	3.25	1.5

Table 4-3: Stuxnet Data Descriptive Statistics by Variable

IV.d. Data Visualization

MATLAB provides excellent tools for multivariate visualization. For visualization purposes, the variables were separated in to two basic categories, Files & Registries and Connections. The following two figures are scatterplots of these two categories for all 5394 data, both benign and malicious.

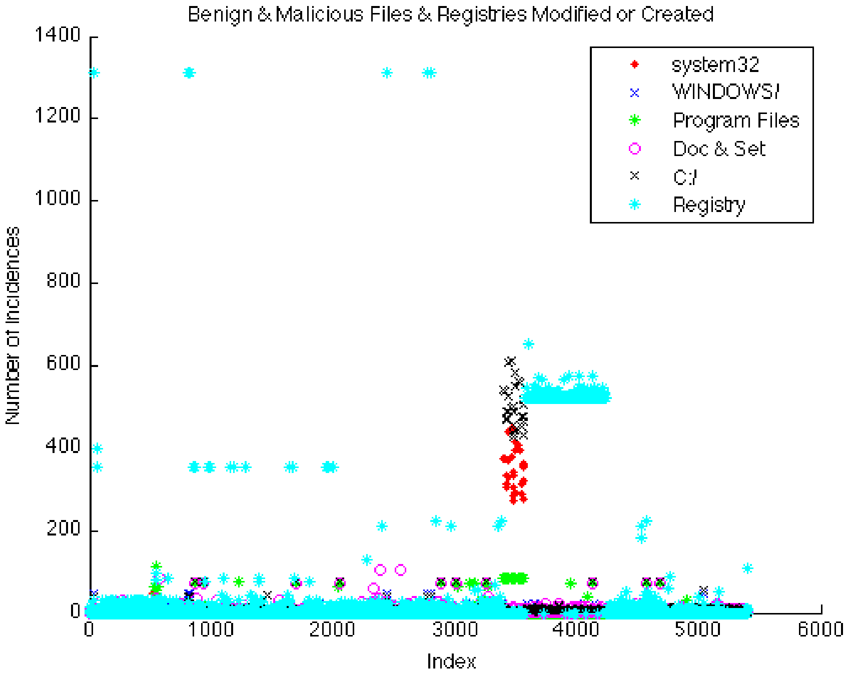

Figure 4-1: File & Registry behavior for all data

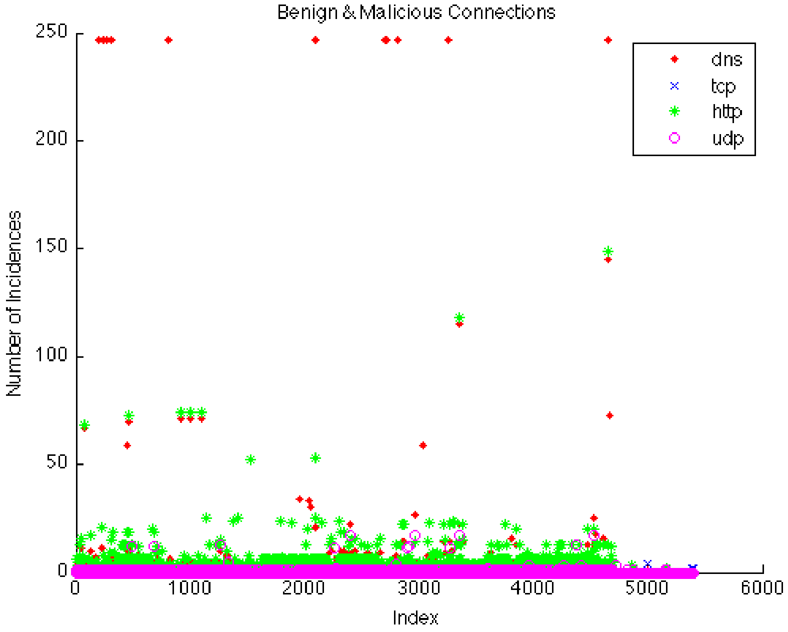

Figure 4-2: Connection behavior for all data

Now, compare the File & Registry activity between malicious and benign data. You can see, for example, that Registry creation and modification behavior is less prevalent in benign data than in malicious data.

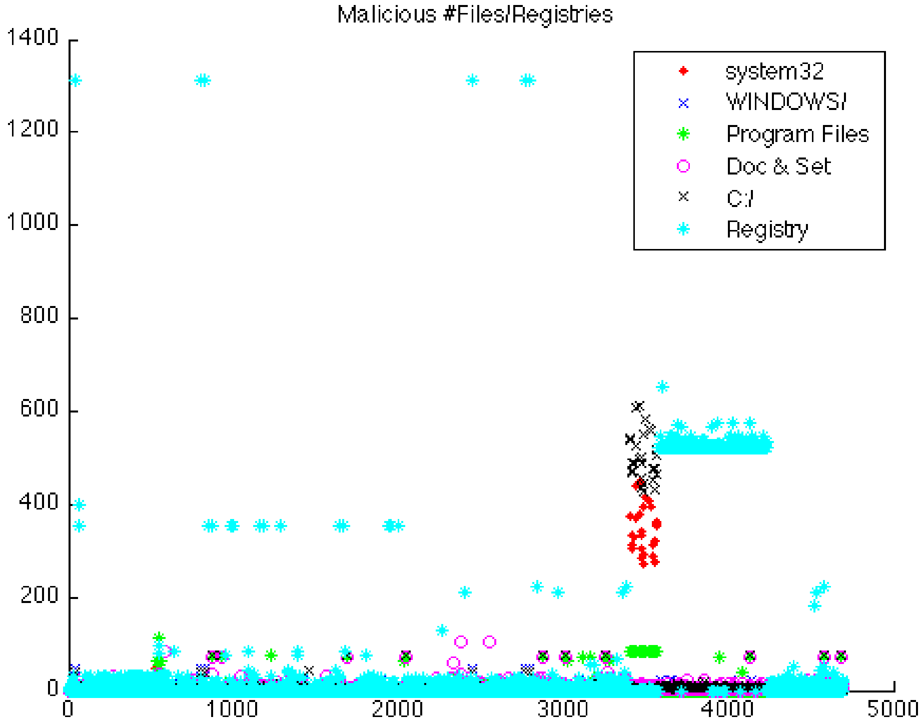

Figure 4-3: Files & Registries for malicious data

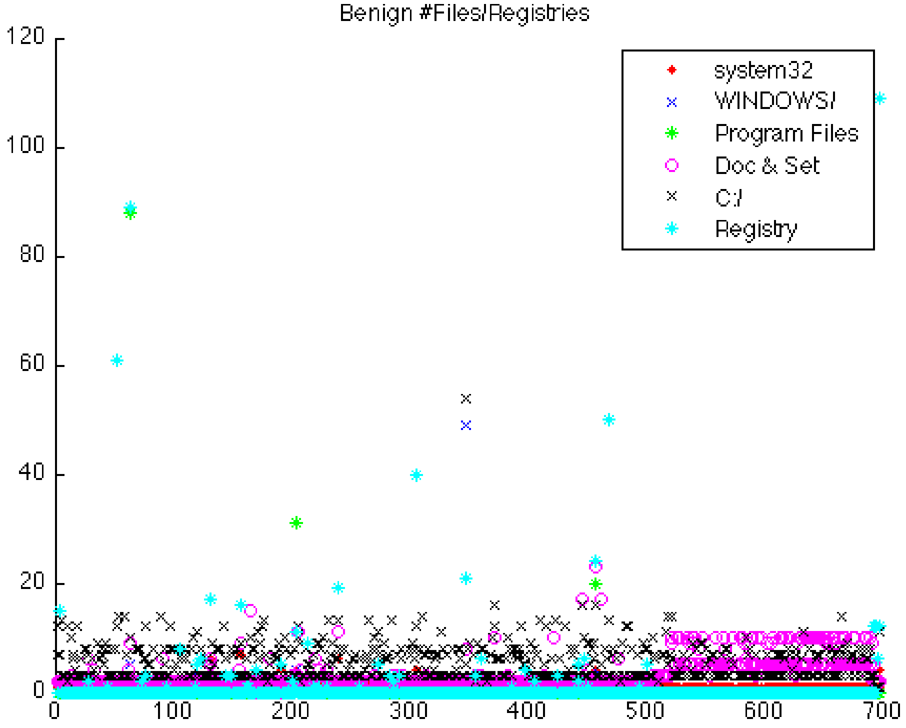

Figure 4-4: File & Registry behavior for benign data

We similarly consider benign and malicious Connection activity. DNS activity appears to be characteristic of malicious activity when all malicious data are considered. Note that only a randomly selected portion of this data is used in for model training and validation.

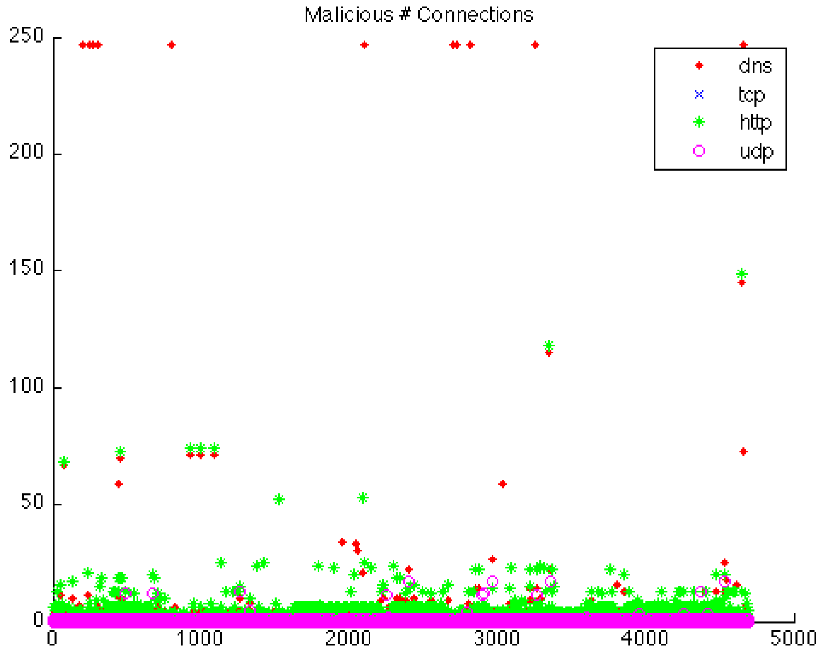

Figure 4-5: Connection behavior for malicious data

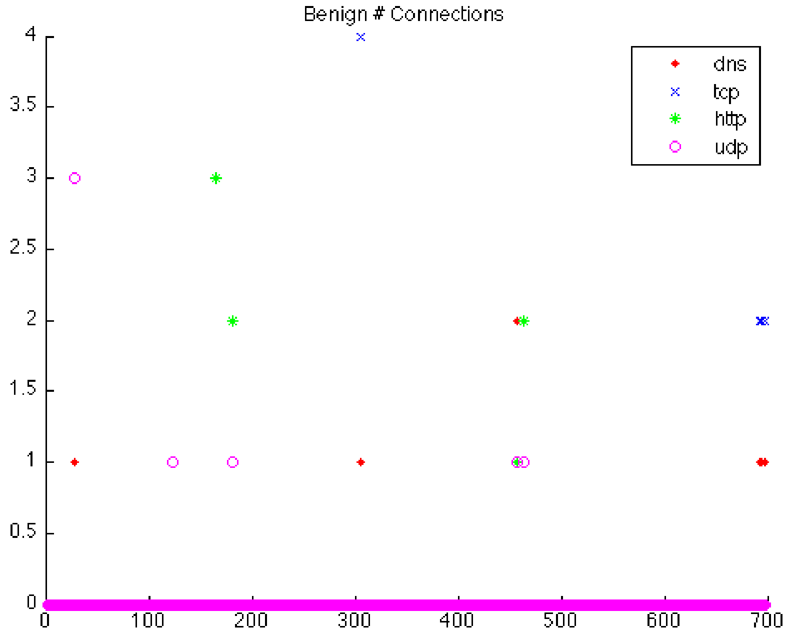

Figure 4-6: Connection behavior for benign data

Now let's look at the Stuxnet raw data. As you can see, the network activity is characterized by high UDP and DNS activity, and the file activity includes high registry activity.

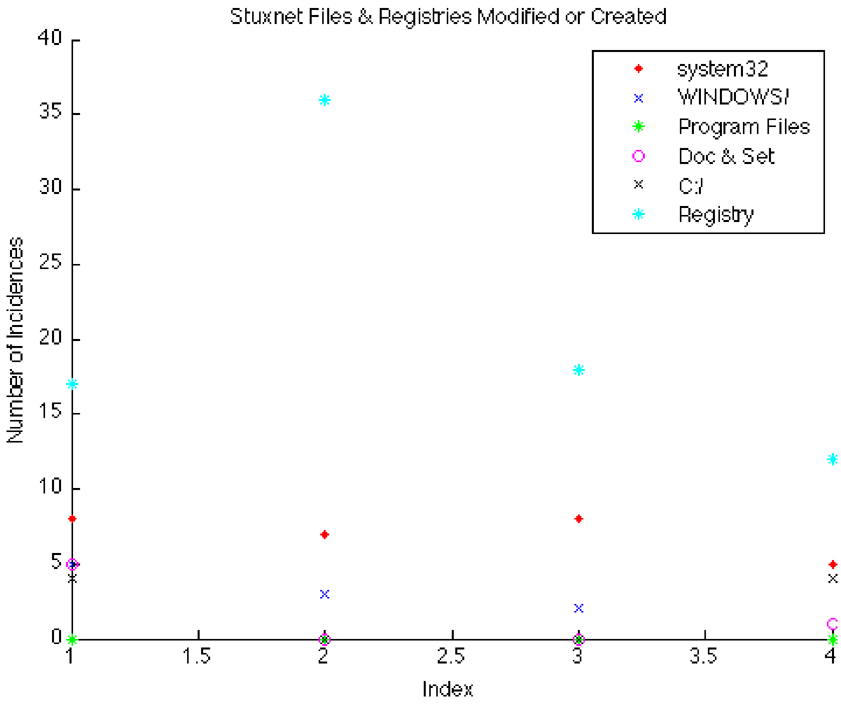

Figure 4-7: File & Registry behavior for Stuxnet data

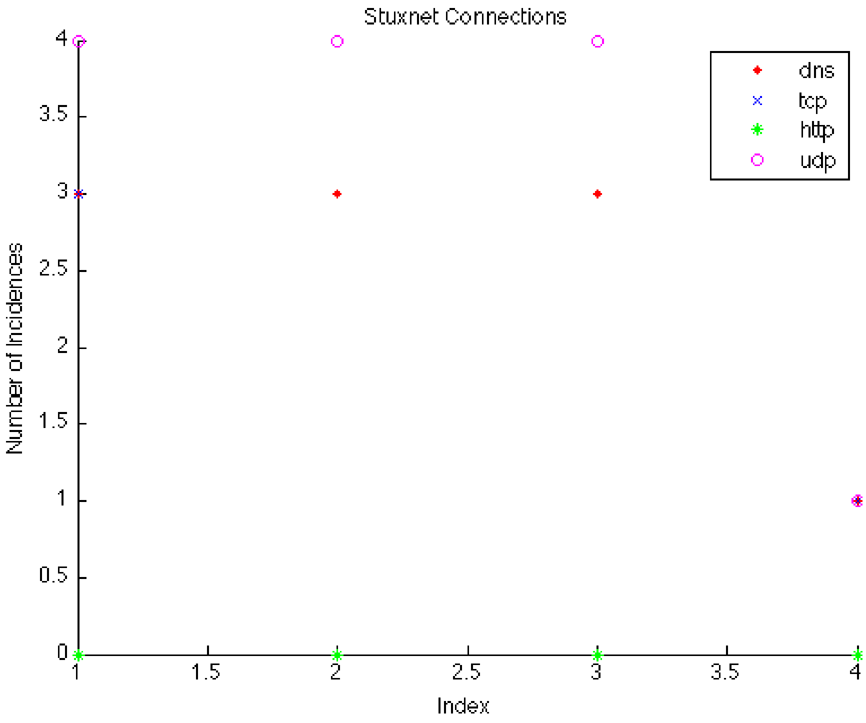

Figure 4-8: Connection behavior for Stuxnet data

V. Results

V.a. Model Coefficients

Using 1256 total randomly selected data records in the training set (with 628 malicious and 628 benign), the coefficients were:

	MLR	GLMN	GLMB	GLMP	GLMNC	GLMBC	GLMPC
Constant	0.0000	0.0000	0.0000	0.0000	0.4369	-0.1727	-0.8071
WINDOWS	-0.0620	-0.0620	-0.0234	0.1036	-0.0022	-0.0325	-0.0013
system32	-0.0445	-0.0445	-0.0851	0.0336	-0.0108	-0.0851	-0.0180
Program	-0.0112	-0.0112	0.0725	0.0176	0.0054	0.0494	0.0077
Docs & Set	0.0069	0.0069	-0.2032	-0.1078	-0.0118	-0.1843	-0.0220
C:/	0.0484	0.0484	0.0034	-0.0797	0.0016	0.0216	0.0013
Registries	0.0012	0.0012	0.0226	0.0010	0.0009	0.0263	0.0013
DNS	0.0039	0.0039	2.2020	0.0022	0.0027	2.1409	0.0039
TCP	-0.0721	-0.0721	-45.0792	-44.5639	-0.1748	-45.1588	-44.3407
HTTP	0.0733	0.0733	0.4093	0.0704	0.0615	0.3771	0.0775
UDP	0.1093	0.1093	0.0522	0.1354	0.1007	0.0675	0.1269

Table 5-1: Coefficients for all Models

You may have noticed that the MLR and GLMN coefficients are the same. They are actually similar up to 10 significant digits but differ beyond that. This led me to wonder if similar computational processes underlie the two models. MATLAB documentation does not say much at that level of detail, but it does state that residuals of the observed responses have a normal distribution in Multilinear Regression . Also, the default distribution for the Generalized Linear Regression function is the Normal distribution.

Pictorially, we can see the relative weight of each variable a little easier. The TCP coefficient is plotted separately so that you can still see the small coefficients.

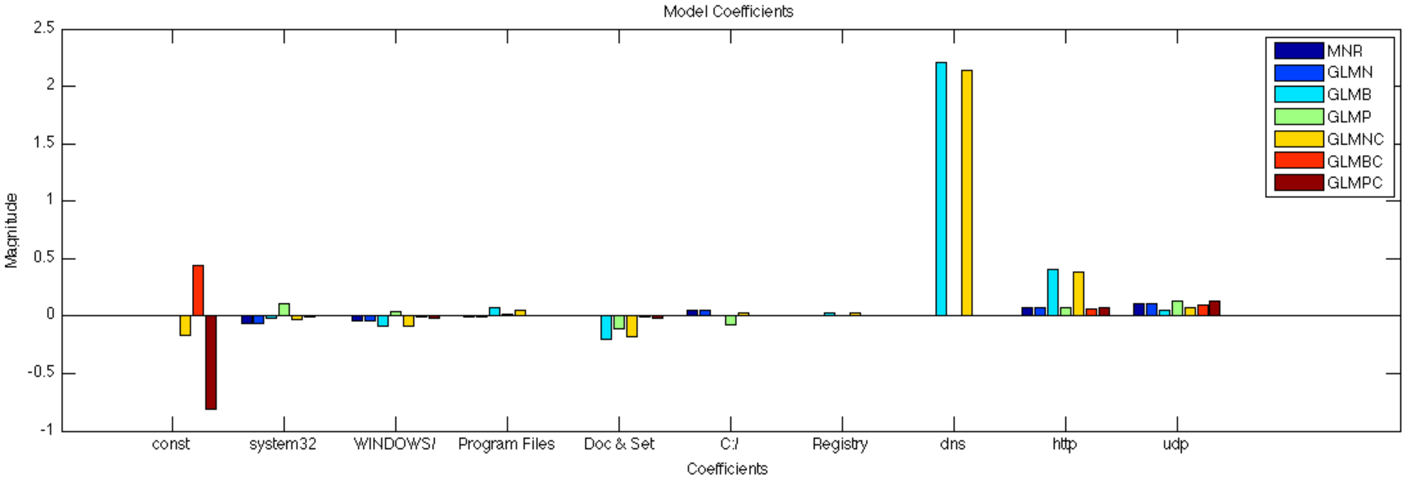

Figure 5-1: Model Coefficients (except TCP)

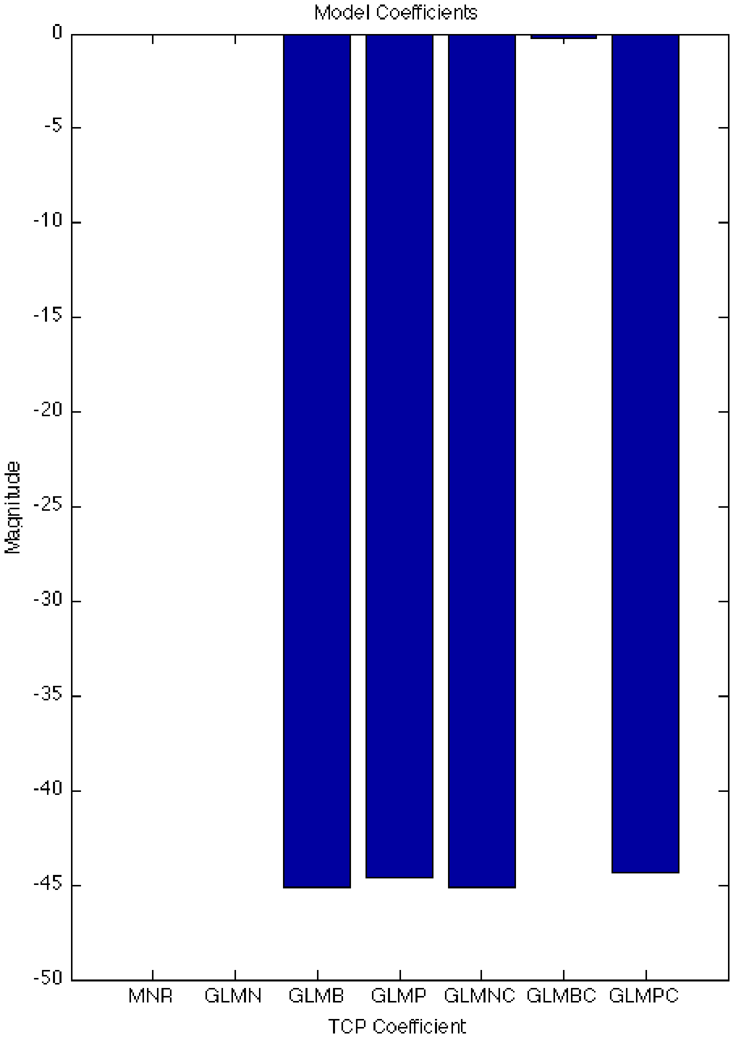

Figure 5-2: Model TCP Coefficients

Since the benign data are assigned a score of zero, we can conclude that negative coefficients indicate benign behavior. Note that 2 of 3 constant coefficients are negative, and most file system activity is negative while network activity is aligned with malicious behavior. Additionally, the Registry variable seems to have the least influence on any of the models.

The TCP coefficient produced by four of the models is greater than -40. These models are Generalized Linear Models with 1) Binomial distribution, 2) Poisson distribution, 3) Normal distribution with a constant coefficient, and 4) Poisson distribution with a constant coefficient.

The Binomial models are also the only models to assign DNS a relatively high coefficient, recognizing our early observation that DNS queries were higher among malicious data.

V.b. Response Descriptive Statistics

Basic descriptive statistics for both the malicious and benign test responses are displayed in the following table.

	Malicious				Benign			
	Max	Min	Mean	Std	Max	Min	Mean	Std
MLR	2.3972	-0.0401	0.4015	0.3982	0.6295	-0.7386	0.2072	0.2031
GLMN	2.3972	-0.0401	0.4015	0.3982	0.6295	-0.7386	0.2072	0.2031
GLMB	25.1586	-1.0190	4.0254	6.8845	6.0974	-0.8260	-0.2612	0.8210
GLMP	1.3691	-2.2755	-0.4613	0.4687	0.3508	-1.2274	-0.5274	0.3080
GLMNC	1.4751	0.3851	0.5694	0.2504	0.8455	0.3712	0.4309	0.0637
GLMBC	26.1577	-0.9970	4.1350	7.1759	4.5506	-0.8775	-0.3241	0.6565
GLMPC	0.3189	-0.9110	-0.6552	0.3263	-0.2941	-0.9329	-0.8314	0.0827

Table 5-2: Basic Descriptive Statistics for Malicious Reponses

In general, descriptive statistics confirm that the models correctly classify malicious and benign data. The means for benign data are lower than the corresponding means for malicious data. The malicious standard deviations for GLMB and GLMBC, both of which assigned TCP a very high coefficient, are too high, as are the malicious maximums. This implies that those models are not stable and may not perform well on data outside the data set.

V.c. Response ROC curves

Unfortunately, simple threshold filtering will not leverage these results. One item for future work is to incorporate conditional probabilities for better use of these results. As mentioned above, the MNR and GLMN coefficients are very similar. Accordingly, you cannot see the yellow line for the MNR ROC curve in the below figure because the magenta GLMN curve completely covers it.

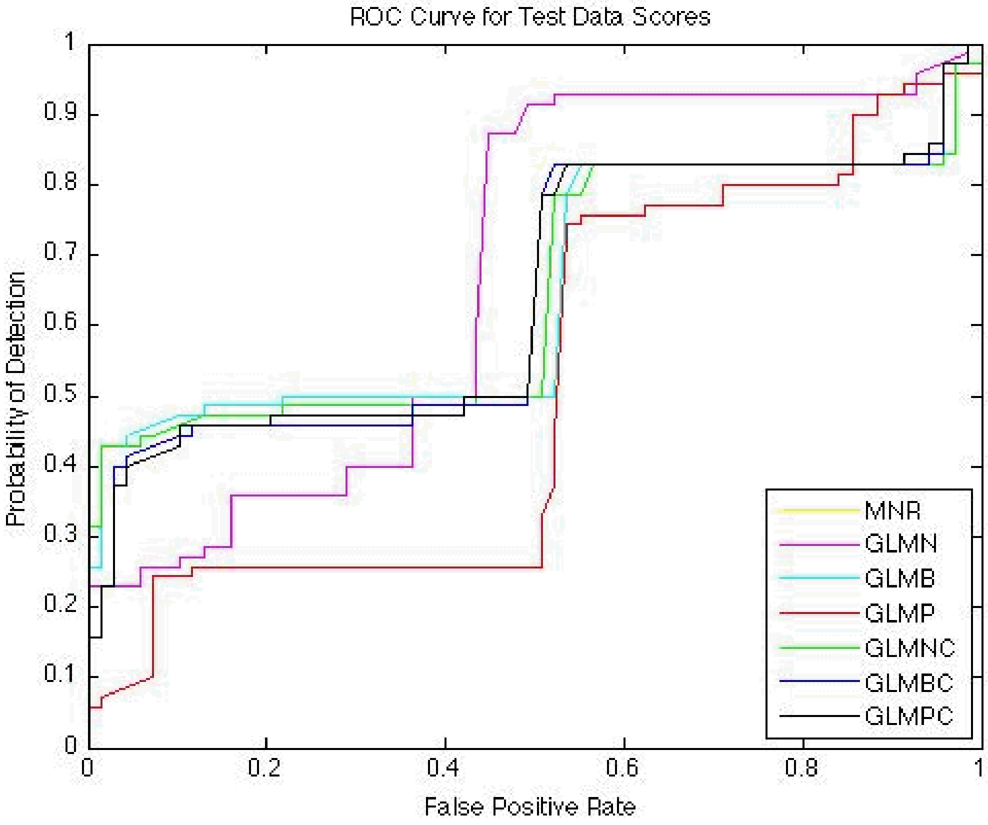

Figure 5-3: ROC Curves for all models

The GLMN/MNR models would perform the best with a simple threshold filter with a 91% detection probability can be achieved with a 48% false alarm rate.

V.d. Response to Stuxnet data

Now, all seven sets of coefficients are applied to the four stuxnet records to see whether the models will assign a malicious score to a previously unseen piece of malware.

V.d.i. Multilinear Regression

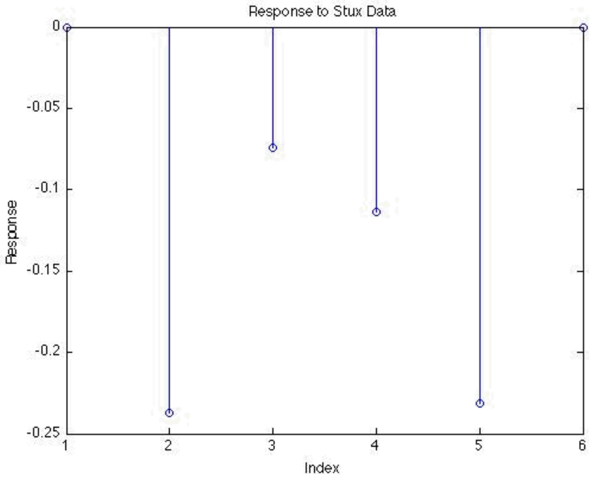

Figure 5-4: MLR Response to Stuxnet

MLR	Stux Scores	Result
1	-0.23686	B
2	-0.07425	B
3	-0.11393	B
4	-0.23158	B

Table 5-5: MLR Response to Stuxnet

MLR Stats	Stux	Malicious Test	Benign Test
Max	-0.07425	2.39716	0.62947
Min	-0.23686	-0.04013	-0.73860
Mean	-0.16416	0.40146	0.20716
Std	0.08253	0.39819	0.20309

Table 5-6: MLR Descriptive Statistics

We can see that the Multilinear Regression model assigned negative responses to all four Stuxnet data. Since the minimum score for malicious data produced by the MLR model is -0.04 and the mean of benign data is 0.2, the MLR model would score all of them as benign.

V.d.ii. Generalized Linear Model with Normal Distribution

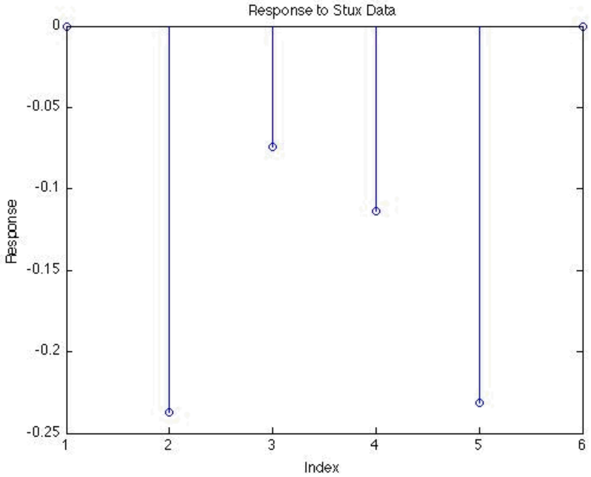

Figure 5-5: GLMN Response to Stuxnet

GLMN	Stux Scores	Result
1	-0.23686	B
2	-0.07425	B
3	-0.11393	B
4	-0.23158	B

Table 5-6: GLMN Response to Stuxnet

GLMN Stats	Stux	Malicious Test	Benign Test
Max	-0.07425	2.39716	0.62947
Min	-0.23686	-0.04013	-0.73860
Mean	-0.16416	0.40146	0.20716
Std	0.08253	0.39819	0.20309

Table 5-7: GLMN Descriptive Statistics

This model also assigned all negative scores to the known Stuxnet data. Its coefficients and descriptive statistics are very similar to the MLR model, so it also would not identify Stuxnet data as malware based on that score alone.

V.d.iii. Generalized Linear Model with Binomial Distribution

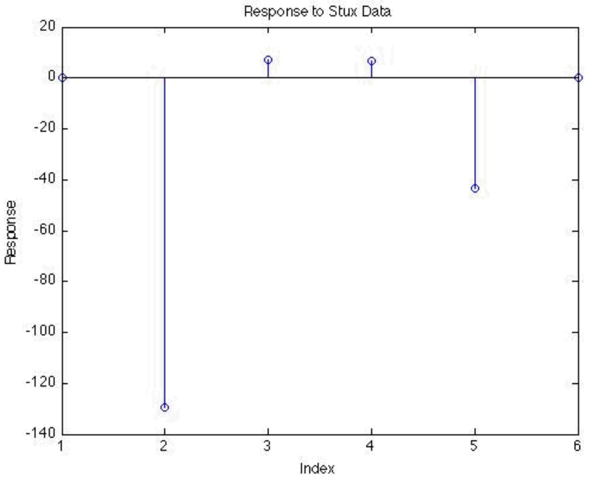

Figure 5-6: GLMB Response to Stuxnet

GLMB	Stux Scores	Result
1	-0.23686	B
2	-0.07425	M
3	-0.11393	M
4	-0.23158	B

Table 5-8: GLMB Response to Stuxnet

GLMB Stats	Stux	Malicious Test	Benign Test
Max	-0.07425	2.39716	0.62947
Min	-0.23686	-0.04013	-0.73860
Mean	-0.16416	0.40146	0.20716
Std	0.08253	0.39819	0.20309

Table 5-9: GLMB Descriptive Statistics

The GLMB model scored the stuxnet data with some better results. Its malicious mean is very high at +4, but its malicious min is -1. Its benign max is -0.8, so records # 2 & 3 are identified as malicious while #s 1 & 4 are not.

V.d.iv. Generalized Linear Model with Poisson Distribution

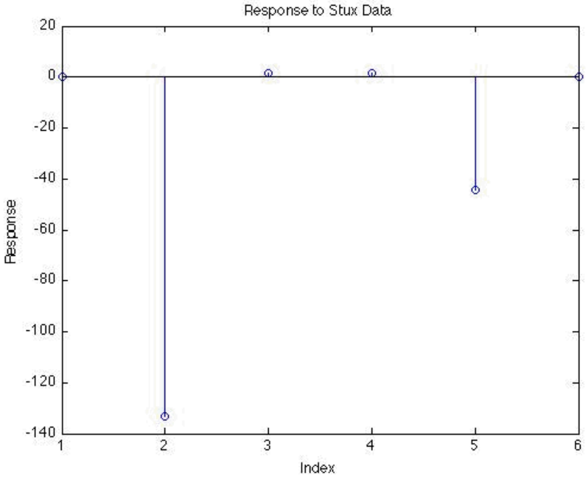

Figure 5-7: GLMP Response to Stuxnet

GLMP	Stux Scores	Result
1	-132.98776	B
2	1.40895	M
3	1.46155	M
4	-44.18880	B

Table 5-10: GLMP Response to Stuxnet

GLMP Stats	Stux	Malicious Test	Benign Test
Max	-132.98776	1.36906	0.35083
Min	1.40895	-2.27548	-1.22742
Mean	1.46155	-0.46127	-0.52743
Std	-44.18880	0.46873	0.30805

Table 5-11: GLMP Descriptive Statistics

The GLMP model would definitely score #1 & #4 as benign due to their high negative scores. It would score #2 & #3 as malicious since they exceed the maximum malicious score.

V.d.v. Generalized Linear Model with Normal Distribution and a Constant Coefficient

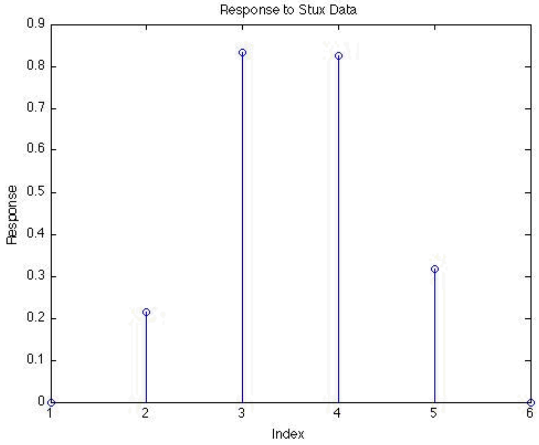

Figure 5-8: GLMNC Response to Stuxnet

GLMNC	Stux Scores	Result
1	0.21465	B
2	0.83313	M
3	0.82500	M
4	0.31687	B

Table 5-12: GLMNC Response to Stuxnet

GLMNC Stats	Stux	Malicious Test	Benign Test
Max	0.83313	1.47511	0.84546
Min	0.21465	0.38512	0.37122
Mean	0.54741	0.56944	0.43088
Std	0.32791	0.25042	0.06373

Table 5-13: GLMNC Descriptive Statistics

The GLMNC model would score #1 & 4 as benign since they are both less than the minimum malicious score. #2 & 3 would be scored as malicious since, although they are slightly less than the benign max, they are closer to the malicious mean than they are to the benign mean.

V.d.vi. Generalized Linear Model with Binomial Distribution and a Constant Coefficient

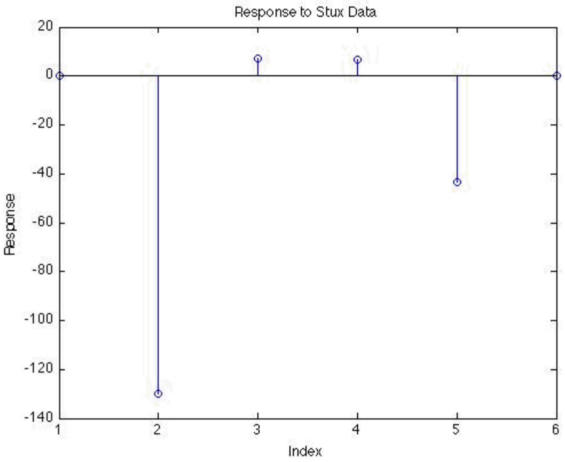

Figure 5-9: GLMBC Response to Stuxnet

GLMBC	Stux Scores	Result
1	-130.02974	B
2	6.98375	M
3	6.56331	M
4	-43.40840	B

Table 5-14: GLMBC Response to Stuxnet

GLMBC Stats	Stux	Malicious Test	Benign Test
Max	6.98375	26.15770	4.55059
Min	-130.02974	-0.99701	-0.87754
Mean	-39.97277	4.13500	-0.32409
Std	64.53057	7.17590	0.65649

Table 5-15: GLMBC Descriptive Statistics

The GLMBC model would score #1 & 4 as benign since they are both far less than the minimum malicious score. #2 & 3 would be scored as malicious since they are higher than the benign max.

V.d.vii. Generalized Linear Model with Poisson Distribution and a Constant Coefficient

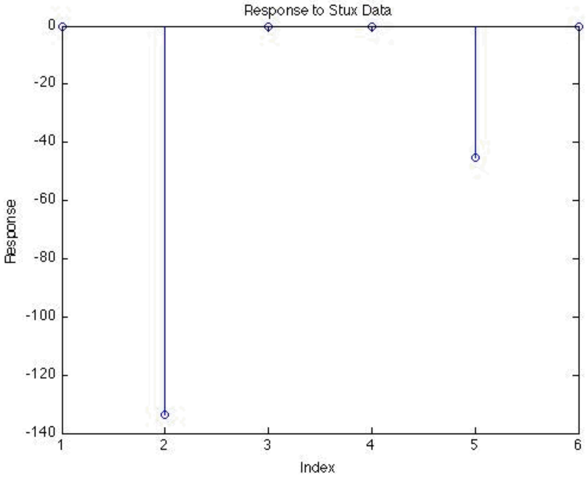

Figure 5-10: GLMPC Response to Stuxnet

GLMPC	Stux Scores	Result
1	-133.49340	B
2	-0.30389	M
3	-0.31105	M
4	-45.09684	B

Table 5-16: GLMPC Response to Stuxnet

GLMPC Stats	Stux	Malicious Test	Benign Test
Max	-0.30389	0.31889	-0.29413
Min	-133.49340	-0.91098	-0.93288
Mean	-44.80129	-0.65516	-0.83140
Std	62.78476	0.32629	0.08268

Table 5-17: GLMPC Descriptive Statistics

The GLMPC model would score #1 & 4 as benign since they are both far less than the minimum benign score. #2 & 3 would be scored as malicious since, while they are greater than the benign min, they are closer to the malicious mean than they are to the benign mean.

VI. Conclusions and Future Work

Overall, the models would score two known stuxnet malware files as malicious and two as benign.

Stux	MLR	GLMN	GLMB	GLMP	GLMNC	GLMBC	GLMPC
1	B	B	B	B	B	B	B
2	B	B	M	M	M	M	M
3	B	B	M	M	M	M	M
4	B	B	B	B	B	B	B

Table 5-18: Model overall Stuxnet classification

The association of UDP and DNS with malicious activity as seen in the raw data seems to have paid off since #2 & 3 were most often characterized as malicious.

The high negative value of 4 of 7 models' TCP coefficients can be attributed to the fact that the maximum malware TCP value was 2 and the maximum benign TCP value was 4. Since these values are quite low as compared to other variables, and few variables are associated with benign software, this coefficient was made to be quite influential in some models in order to meet the benign predicted response of 0.

A summary of the characterization of each variable is below.

Variable	Character
Constant	n/a
WINDOWS	n/a
system32	Malicious
Program	n/a
Docs and Set	Benign
C:/	Malicious
Registries	Malicious
DNS	Malicious
TCP	Benign
HTTP	n/a
UDP	Malicious

Table 5-19: Variable Summary

The small influence of the Registry variable, however, worked against detection of Stuxnet. The Stuxnet samples created or modified an average of 21 registries, while the training malware averaged 3 and benign software averaged 1. You may have noticed above that the overall malware average for this variable was 77; this is a consequence of using a subset of malicious data in the training set.

VI.a. Conclusions

A simple threshold filter would not be adequate to classify new threats as malicious. Overall the results reflect a 50% accuracy rate. However, even a 50% detection rate may have alerted us to the Stuxnet threat sooner.

The malicious data set was large and varied, but the benign data were not. Also, the analysis tool is better at analyzing malicious files thoroughly than benign files. Benign software is fairly inactive without user intervention. Though many commonplace pieces of software may be capable of making filesystem changes and network activity comparable to any average piece of malware, the reality is that most benign software only performs those actions after being instructed to do so by the user in some way. For example, the portable apps required installation and extraction prior to doing anything else. On the other hand, malware is known for not requiring any user intervention, and will quickly root itself in the filesystem and registry, and possibly start contacting other bots. What this means for our model is that it is good at detecting samples that are performing a variety of forensically significant actions autonomously (such as when run inside of the analysis tool), but it will not correctly classify software on a live network that has user intervention to direct it. More representative benign data would likely correct the false negative classification observed.

More realistic scores for training data, or perhaps variable scores based on the maliciousness of the malicious file, is an area of future consideration. One possible near-term solution to this problem is randomly generated scores between, for example, 0 and 0.5 for benign data and 0.5 and 1 for malicious data. This issue may also be solved with data centering and scaling in future work.

VI. Future Work

Future work will include more variables and string analysis so that non-numeric data can be analyzed. The ability to accommodate variable-length datasets would be desirable. For example, the connection data includes port numbers for each connection. Currently this and other data is not in our model because one record might contain four tcp connections with four sets of source and destination port numbers, while another record may contain one tcp connection with one set of port numbers.

More models are available to explore and, as we have seen in this project, an option such as the probability distribution significantly affects results. Future work would also examine assignment of scores used in the malicious and benign training data and/or data centering and scaling.

An area of additional improvement will also improve malicious data classification. This type of information was not available because the filenames consisted of MD5 hashes. I made an attempt to classify malware based on the AntiVirus output, but many of the malware samples are so new that AntiVirus did not detect it.

VII. Works Cited

Arbor Networks, <http://www.arbornetworks.com/>.

Falliere, Nicolas, et al. W32.Stuxnet Dossier. Symantec Security Response. Version 1.4 (February 2011).

Matrosov, Aleksandr, et al. Stuxnet Under the Microscope. ESET. Revision 1.3.1. <http://www.eset.com>.

Offensive Computing, <http://www.offensivecomputing.net/>.

PortableApps.com - Portable software for USB, portable and cloud drives. Rare Ideas, LLC. 6 3 2011 <http://portableapps.com>.

Van Randwyk, Jamie, et al. "Farm: An automated malware analysis environment." 42nd Annual IEEE International Carnahan Conference on Security Technology. IEEE ICCST, 2008. 321-325.