Insider Threat Detection Using Time-Series-Based Raw Disk Forensic Analysis

Nicole Beebe, Lishu Liu, Zi Ye

Chapter 9

INSIDER THREAT DETECTION USING TIME-SERIES-BASED RAW DISK FORENSIC ANALYSIS

Nicole Beebe, Lishu Liu and Zi Ye

Abstract This research tests the theory that volitional, malicious computer use based on insider threat activity can be detected via a time-series-based analysis of data and file type forensic artifacts that reside on a raw disk. In other words, statistical profiling of allocated and unallocated space pertaining to the types of files accessed and the data browsed, acquired and processed incident to espionage, intellectual property theft, fraud or organizational computer abuse can help detect insider threats. The t-test approach is used to compare the means of two time windows using the split and sliding window methods along with first-order autoregressive modeling. Empirical testing against the nineteen-day snapshots of the M57-Patents case provides support for all three methods, but the results suggest that the first-order autoregressive modeling method is the most robust. Additionally, the autoregressive modeling approach is likely to generate more intuitive results for an analyst. Ground truth analysis confirms nearly all of the outliers that were detected. While the majority of the outliers were due to benign and easily explainable situations and system contexts and the minority were due to malicious activity, the approach does not yield an inordinate amount of search hits to examine and validate. This research thus provides a new computational approach for locating digital forensic evidence.

Keywords: Insider threat, anomaly detection, time series, profiling

1. Introduction

The trusted insider remains one of the most critical cyber security threats to organizations [3, 7, 17, 23]. In fact, some contend that insiders present greater risks to organizations than external attackers [19, 22]. Insiders vary along two major dimensions – malice and volition [5, 11,

24]. Malicious, volitional insiders are often characterized by their methods and motivation and placed into four categories: (i) espionage; (ii) intellectual property theft; (iii) fraud; and (iv) sabotage [1, 12]. Volitional, non-malicious insiders include users who knowingly subvert security measures to accomplish work goals and insiders who violate acceptable use policies for personal gain or satisfaction. This research focuses on volitional insiders with malicious intent, specifically those interested in espionage, intellectual property theft or fraud, as well as non-malicious, volitional insiders who abuse computing privileges for personal satisfaction (e.g., browsing pornography on the web). Both types of insiders often leverage institutional trust and system access privileges to facilitate their criminal or unauthorized computing activities [4, 24].

Current approaches for detecting insiders rely largely on behavioral heuristics based on past insider cases [18]. These approaches fall short in three important ways: (i) they fail to detect novel insider methodologies and attacks; (ii) they fail to detect large-scale data collection within the scope of authorized access permissions; and (iii) they fail to consider forensic traces of information-handling activity in unallocated space. Analyses of seven insider cases – Robert Hanssen (1979), Aldrich Ames (1985), Harold Nicholson (1994), Brian Reagan (1999), Leandro Aragoncillo (2004), Chelsea Manning (2010) and Edward Snowden (2013) – have revealed a single, common distinguishing characteristic: in preparing to exfiltrate data, an insider often browses, acquires and prepares data for exfiltration on a single system, typically his/her own workstation [1, 6, 8, 13, 15, 16].

This research posits that digital forensic traces of user activity, in both allocated and unallocated space, can signal impending exfiltration and unauthorized computer use for which information browsing, collection and/or handling are facilitating activities. Specifically, this research seeks to profile a workstation disk at the physical level based on the forensic artifacts that are left behind from user activity with respect to the types of data browsed, stored and handled. Following this, it attempts to detect statistical anomalies in the profile over time that signal nefarious user activity. Five types of features are considered, including file types, file classes, data types, email related features and string classes other than email-related strings. Table 1 shows examples of each feature type. In the case of string classes, the measures used include the total number of instances (hits) that match the type of string and the total number of unique instances (i.e., without repeated hits); in the case of email addresses and URLs, the measures used also include

Table 1. Feature types.

Feature Type	Examples
File Types	JPEG, Email, PDF, EXE
File Classes	Text, Video, Audio, System
Data Types	Compressed, Encrypted, Allocated
String Classes	Email, CCN, SSN, URL

the numbers of instances in specific most frequently occurring (high-frequency) domains (e.g., `gmail.com`).

2. Methodology

A time series analysis was conducted of four disks with a synthetic dataset (discussed below) that were snapshotted daily for nineteen days. Two classes of time series analysis were employed: (i) t-tests; and (ii) autoregressive analyses, both with varied set-ups and parameters. The t-tests involved two methods for establishing time series windows: (i) split window; and (ii) sliding window. A *post hoc* ground truth analysis was conducted to validate the statistically-detected anomalies by assessing the Type I error (false positives) and the Type II error (false negatives).

2.1 Sample Data

The sample data was taken from the M57-Patents dataset [9, 10] corresponding to a case involving four employees of a fictitious corporation, three of whom were involved in various types of criminal activity, including intellectual property theft, extortion and possession of illegal pornography. In producing the synthetic evidence, the scenario participants engaged in scripted and normal user activities every day for nearly three weeks. Researchers made forensic images of the user workstations at the end of each day. All the daily disk images from the case were analyzed using a data driven anomaly detection algorithm.

2.2 Data Driven Algorithm Development

In this context, a statistical outlier means that the outlier media (e.g., an employee workstation) has a storage profile that is different from a historical perspective. The mathematical definition of what constitutes an inlier versus an outlier varies from dataset to dataset, especially when the central distribution violates conditions such as normality. In such cases, the central distribution is ideally identified by removing outliers

and then modeling the data. However, removing outliers may not be possible because they are not always known. Challenges to defining inlying user behavior include: (i) encompassing the full range of normality; (ii) normality that evolves over time; (iii) normality that varies across contexts; and (iv) difficulty in establishing a precise boundary between inlying and outlying behavior [21]. As a result, an outlier detection process cannot be easily separated from the process of identifying the normal storage profile.

Traditional statistical methods cannot be used when outliers cannot be eliminated from a dataset before determining the central distribution. Instead, robust statistical measures are required that are not significantly influenced by outliers. Otherwise, outlier masking occurs – the central distribution is skewed by outliers, causing failures in outlier detection [14, 20].

In a deployed application of this research, such as the ongoing monitoring of employees, an analyst would not know the ground truth *a priori* and would be unable to separate outliers before establishing a statistical profile of a workstation. Furthermore, the analyst would often be unable to ensure that outliers do not already exist when establishing a statistical profile. Accordingly, this research uses a robust data driven algorithm that is not as sensitive to outliers as traditional methods. The data distribution is characterized using a robust location parameter (center of the data) and a robust dispersion parameter (variability of the data around the center).

2.3 Time-Series-Based Anomaly Detection

In time-series-based anomaly detection, the storage profiles of daily disk snapshots are treated as time-ordered sequences. Anomalies are then detected by: (i) comparing means between two different time periods; or (ii) predicting future observations in a time series based on past values and declaring as outliers the actual values that deviate significantly from predicted values. The former is accomplished via unpaired t-testing whereas the latter is accomplished via autoregressive modeling.

Unpaired t-Test Approach. Outliers are found in time series data by comparing two periods of time, ΔT1 and ΔT2, for statistical differences between the periods. Toward this end, unpaired t-tests were conducted – unpaired because ΔT1 and ΔT2 occur at different times and the observations are not paired in the sense of a repeated measures design.

The basic outlier detection approach involves the following steps:

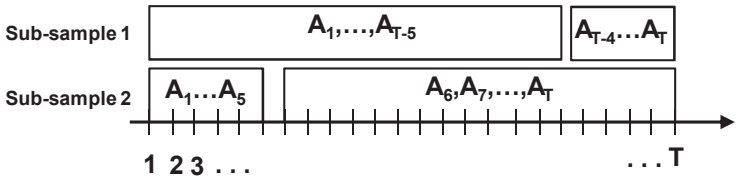


Figure 1. Unpaired t-test – Split window method.

- **Step 1:** From a complete time series $A_1, A_2, ..., A_T$, create several sub-samples where each sub-sample contains two sub-series, $X_{1i}, X_{2i}, ..., X_{Mi}$ and $Y_{1i}, Y_{2i}, ..., Y_{Ni}$, where $M, N < T$ and i is the index of a sub-sample.
- **Step 2:** Perform an F-test to test for the equality of the variances of the two sub-series in each sub-sample. If the p-value of the F-test is greater than 0.1, then the variances are considered to be equal ($\sigma_X^2 = \sigma_Y^2$).
- **Step 3:** Perform the appropriate t-tests based on variance equality and obtain a p-value for each sub-sample i. If a p-value is larger than a certain significance level, then the null hypothesis that the means of the two sub-series are equal ($\mu_X = \mu_Y$) is not rejected.
- **Step 4:** For each time series division point (split point) at which the p-value meets a specified significance threshold, declare an outlier at the split point. When a time series exhibits multiple outlying points, order the split points in ascending order of p-value significance to rank order the outlying points for further analysis.

Two methods for defining sub-series samples were employed: (i) split window method; and (ii) sliding window method:

- **Split Window Method:** In the split window method, each sub-sample contains the entire time series sequence split into two sub-series. Different sub-samples have successively different split points in the time series continuum beginning at t_2 (first observation is t_0) and ending at t_{n-2} because at least two points are needed in a sub-series sequence. In the example shown in Figure 1, the split point for sub-sample 1 ($i = 1$) occurs at the fifth to last time point A_{T-5}. The split point for the second sub-sample ($i = 2$) occurs at the sixth time point A_5. Continuing this procedure yields $T - 3$ sub-samples.

  As described above, $T - 3$ p-values $P_1, P_2, ..., P_{T-3}$ are computed. When the p-value is statistically significant, it can be concluded

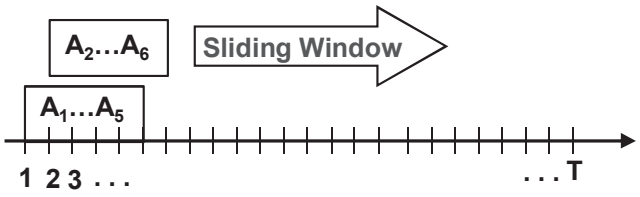


Figure 2. Unpaired t-test – Sliding window method.

that there is a difference between the means of the observations that occurred before and after the split point. This is referred to as a jump point or change point and the later observation is typically considered to be the outlying point (chronologically speaking). Using the terminology, the outlying observation for P_i is A_{i+2}.

The limitations of the unpaired split window method are: (i) inability to detect outliers at the first two or last two time points in a time series because they cannot be split points; (ii) inability to conduct a t-test when there is no variance in the sub-series on either side of a split point (e.g., in the case of a step function); and (iii) sub-optimal level of robustness.

- **Sliding Window Method:** In the sliding window method, the entire time sequence in the composite of the two sub-series is no longer included in a single sub-sample. Furthermore, the window size W is held constant for all the sub-series in a sub-sample. After setting W, the window is moved incrementally along the entire time series, creating $T - W + 1$ sub-series of length W. Each sub-series is then paired with its successive sub-series to obtain $T - W$ sub-samples. While W remains fixed for an entire set of sub-samples, W could vary for alternate sub-sample sets. For a time series $A_1, A_2, ..., A_T$, the range of W is $2 \leq W \leq T - 2$. Small window sizes may bear too little information while large window sizes are limited from the standpoint of outlier detection sensitivity, similar to the split window method discussed above. Again, the p-values between sub-series within each sub-sample are computed and ranked outliers are considered based on statistically significant p-values. Figure 2 shows a graphical depiction of two sub-series in a single sub-sample using the sliding window method.

  The limitations of the unpaired t-test sliding window method are: (i) inability to detect outliers at the first $W - 1$ points or the last $W - 1$ points in a time series because they cannot be split points

(this is mitigated by a small window size); and (ii) inability to perform a t-test when there is no variance in the sub-series on either side of a split point (e.g., in the case of a step function). When there is a constant segment in the time series of length $\geq 2W$, the t-test cannot be performed for the segment because $s_X^2 = s_Y^2 = 0$ for the first $W + 1$ sub-samples. An anomaly detection system should test for constant segments and univariate step functions and the p-value should be set to one for these sub-samples because no outliers exist in constant value segments. Finally, this approach is not particularly robust because the sub-series means are influenced by outliers. However, the effect is less pronounced than in the split window method, especially when W is sufficiently small.

Autoregressive Model Method. Instead of comparing means between two sub-series in a time series sequence, the autoregressive (AR) model method predicts successive observations in a time-varying sequence as a linear model of its previous values. $AR(p)$ (p is the number of prior observations in the sequence) along with a noise term help predict the current observation. In an $AR(0)$ time sequence, the prior observation does not help predict the current observation. In an $AR(1)$ time sequence, the single prior observation helps predict the current observation, and so on. When a time series conforms to the autoregressive model assumptions and the model is $AR(p > 0)$, then outliers can be declared as the points whose actual values deviate statistically from the predicted values. An autoregressive model $AR(p)$ of order p is given by:

$$A_t = c + \sum_{j=1}^{p} \phi_j A_{t-j} + \varepsilon_t \tag{1}$$

where $\theta = (c, \phi, \sigma_2)'$ is the parameter vector and the error terms ε_t are independent and identically distributed and follow a normal distribution $\varepsilon_t = N(0, \sigma_2)$.

Since it is not possible to readily know the exact distribution of sub-series, it is necessary to first work with the simplest autoregressive model $AR(1)$, which is given by:

$$A_t = c + \phi A_{t-1} + \varepsilon_t \tag{2}$$

where $\theta = (c, \phi, \sigma_2)'$ is the parameter vector.

The parameters are estimated using the maximum likelihood estimation (MLE) method. Given an observed sample $a_1, a_2, ..., a_T$ of size T, the first step is to compute the joint probability density function:

$$f_{A_1,A_2,...,A_T}(a_1, a_2, ..., a_T; \theta) \tag{3}$$

This can loosely be considered to denote the probability of having observed the particular sample.

The maximum likelihood estimate $\hat{\theta}$ is the value for which the sample is most likely to have been observed. Specifically, it is the value of θ that maximizes the probability density function in Equation (3). Note that at least three observations are required to obtain an estimate using this approach.

Suppose that the three observations are a_1, a_2 and a_3, and the maximum likelihood estimate is:

$$\hat{\theta} = (\hat{c}, \hat{\phi}, \hat{\sigma}_2)' \tag{4}$$

It is possible to predict the next observation using the equation:

$$\hat{a_4} = \hat{c} + \hat{\phi} a_3 + \hat{\varepsilon}_4 \tag{5}$$

and to compute the residual between the actual and predicted values as:

$$res_4 = a_4 - \hat{a_4} \tag{6}$$

Continued iteration yields $T - 3$ residuals $res_4, res_5, ..., res_T$.

Using a forward (chronologically speaking) autoregressive model approach, it is not possible to identify whether the first three observations are outliers; this is because they are required for model building. However, unlike the unpaired t-test approach, a work-around is available. This simply involves backward (chronologically speaking) autoregressive modeling. When using the reversed sequence $a_T, a_{T-1}, ..., a_3$ as the observed time series values, the maximum likelihood estimates are obtained in the same manner as before. Specifically, the next future value is given by:

$$\hat{a_2} = \hat{c}^* + \hat{\phi}^* a_3 + \hat{\varepsilon_2} \tag{7}$$

and the residual is:

$$res_3 = a_2 - \hat{a_2} \tag{8}$$

Note that the residual for the third point is $a_2 - \hat{a}_2$ instead of $a_3 - \hat{a}_3$ because a reversed sequence is used. Also, if a_3 were an outlier, a very large difference between a_2 and $\hat{a}_2$ would be obtained by the backward procedure.

The residuals are $res_2, res_3, ..., res_T$. Defining the residual threshold for an outlier, however, is less straightforward than for unpaired t-tests because the magnitudes of the residuals can vary widely. Therefore, the residuals are standardized using the equation:

$$res_{sd(i)} = \frac{res_i - mean(res)}{var(res)} \tag{9}$$

and an observation whose absolute standardized residual is larger than two is defined as an outlier:

$$\left|res_{sd(i)}\right| \geq 2 \tag{10}$$

The sensitivity of this procedure can be tuned by defining a larger absolute standardized residual value (e.g., $\left|res_{sd(i)}\right| \geq 3$). However, the experiments conducted in this research suggest that it is better to use a threshold of two.

The primary limitation of this approach is that white noise ε_t is required to build a time series model. However, even in constant value segments, it is easy to add a small random noise term with the same mean as the sub-series and with very little variance to remove the constancy of the sub-series without modifying its underlying distribution.

3. Experimental Results

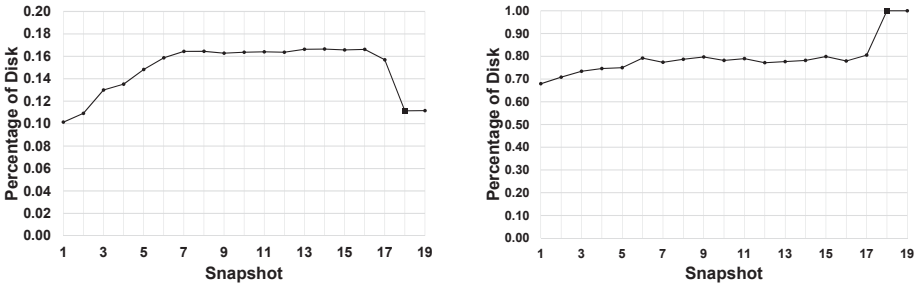
The three time-series-based anomaly detection methods were evaluated using the nineteen observation time series for the users in the synthetic M57-Patents dataset. While the intervals between observations in this data set are not identical, they are approximately equal (daily) and, hence, the observations were treated as having equal intervals.

Thirty-three of the 88 features have constant and/or zero values across all nineteen time intervals and were, therefore, removed from the sample, leaving 55 univariate, time series samples for testing. The constant and/or zero valued features included twelve credit card number features, twelve social security number features and the following file/data types: active server page files (`.asp`/`.aspx`), base64, base85, base16, URL encoded, postscript (`.ps`), tagged image file format (`.tif`/`.tiff`), configuration files (`.ini`) and link files (`.lnk`).

3.1 Unpaired t-Test/Split Window Method

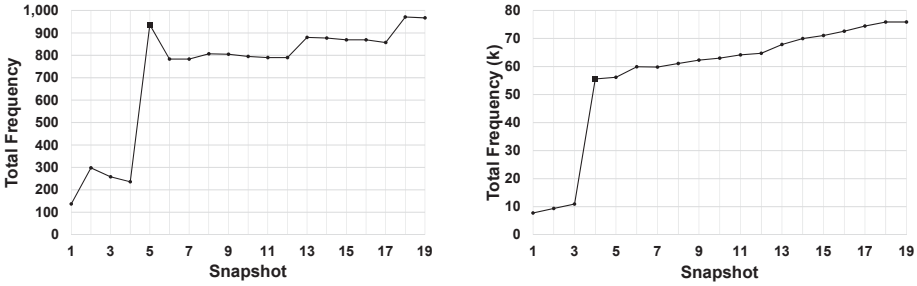
A p-value of 0.05 was selected as the significance threshold for outlier determination. The unpaired t-test with split window method was observed to work well for time series exhibiting sudden changes after sustained periods with low variance (Figures 3(a) and 3(b)) and for step functions (Figures 4(a), 4(b) and 5(a); the data type in Figure 4(a) is the top-third most frequent email domain). Note that all the experimental results described here pertain to user Charlie. Similar functions and outlier detection trends were realized for the other users in the dataset.

However, the t-test with the split window method can be misleading. This is seen in Figure 5(b) when the change is more gradual (i.e., gradual change function with misleading outlier detected using the split window



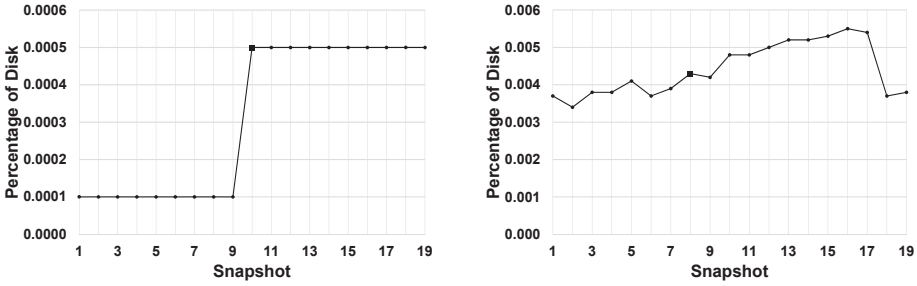
(a) Decrease (Data type: random). (b) Increase (Data type: allocated).

Figure 3. Time series exhibiting sudden changes.



(a) Data type: email. (b) Data type: URL.

Figure 4. Time series with pseudo-step function changes.



(a) Data type: GIF. (b) Data type: video.

Figure 5. Time series with step function (left) and gradual function changes (right).

method) and also in Figure 6 when the change is a spike function (i.e., temporary change returning to the previous relative, steady-state condition where the data type is the top-third most frequent email domain). When the change is more gradual, an outlier would be declared in the midst of the gradual change, making it difficult for an analyst to understand why the snapshot was deemed an outlier. The gradual change

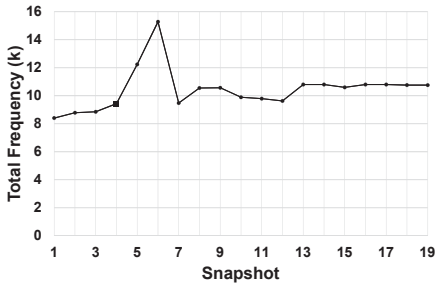


Figure 6. Time series with spike function change (Data type: email).

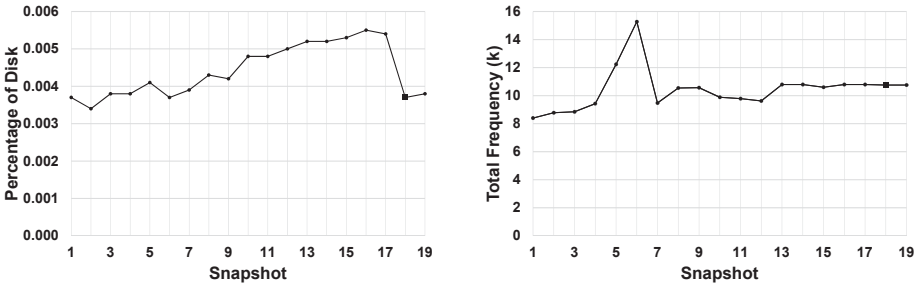
scenario is a concern because a patient and skilled insider may collect data gradually to specifically thwart detection efforts.

When the change is a spike function, the observation identified as an outlier is again misleading. The return to steady-state masks the true outlying observation point that occurs one or two intervals after the observation identified as the outlier. In this situation, without being alerted to the full nature of the time series, an analyst may only examine the identified outlying snapshot and erroneously declare it to be a false positive. A different conclusion may have been reached if the analyst had analyzed the snapshot(s) following the split point for a more complete context. The spike function scenario is a concern when an insider collects, exfiltrates and quickly wipes the collected data from the hard drive (i.e., allocated and unallocated space). A potential mitigation strategy is to design the system to detect significant changes in the wiped disk space.

In summary, using the unpaired t-test and split window method can identify outliers. However, an analyst would be able to make more informed analytical and investigative decisions if provided with the supporting time series function as a visualization aid.

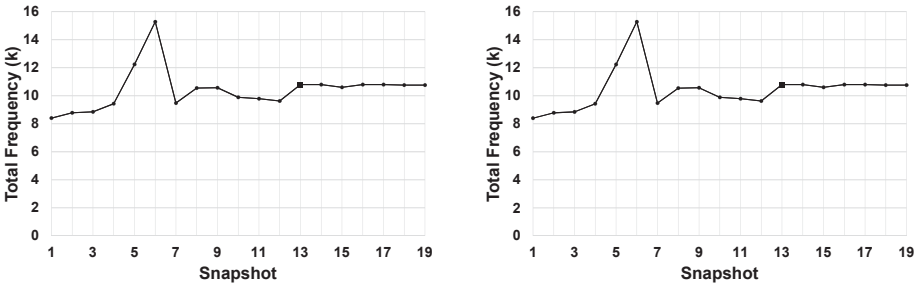
3.2 Unpaired t-Test/Sliding Window Method

Once again, a p-value of 0.05 was selected as the significance threshold for outlier determination, although this could be changed akin to a sensitivity setting. The results indicate that an unpaired t-test with the sliding window method works reasonably well at detecting sudden changes and step functions; to some extent, the sliding window method may be more sensitive at detecting small changes than the split window method. Also, it may occasionally provide more intuitive results to an analyst by identifying the outlying observation at the end of the change period as in Figure 7(a) (for the video data type) rather than during the



(a) Gradual change. (b) Spike function change.

Figure 7. Sliding window successful detection (left) and failure (right) for $W = 2$.



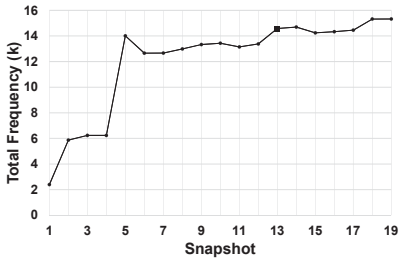
(a) Spike function change. (b) Spike function change.

Figure 8. Sliding window spike detection failure for $W = 3$ (left) and $W = 4$ (right).

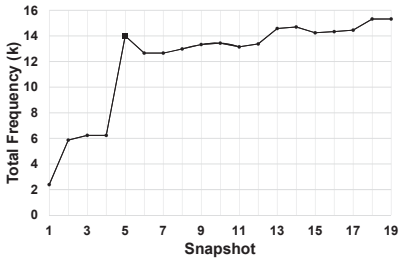
change period as in Figure 5(b). However, the sliding window approach appears to be even less able to detect very short duration spikes regardless of W as shown in Figures 7(b), 8(a) and 8(b) (for the top-third most frequent email domain data type).

Another problem with the sliding window approach is that a wide variety of results were obtained depending on the window size W. This is because there does not appear to be a single, universal objectively superior W that could be used. Two example sets are shown in Figures 9(a) through 9(c) and in Figures 10(a) through 10(c).

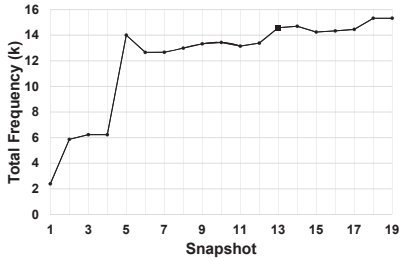
The empirical results indicate that the split window method should be preferred over the sliding window method. However, the impact that the time aperture may have on the split window method is a concern. The empirical time aperture was approximately nineteen days. Further empirical research is needed to ascertain the impact of a larger time aperture on the results.



(a) $W = 2$.

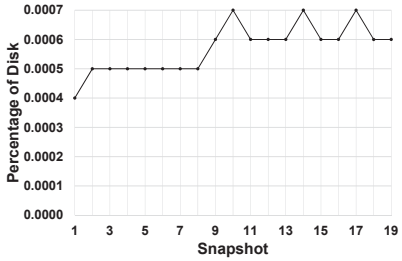


(b) $W = 3$.

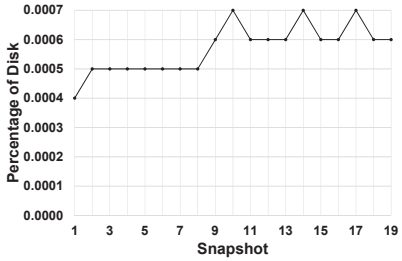


(c) $W = 4$.

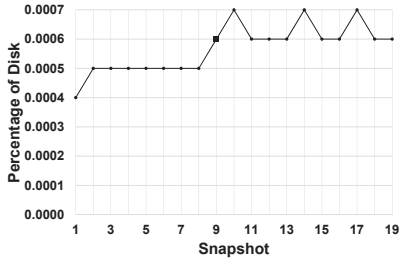
Figure 9. Outliers detected via sliding window (Data type: email).



(a) $W = 2$.

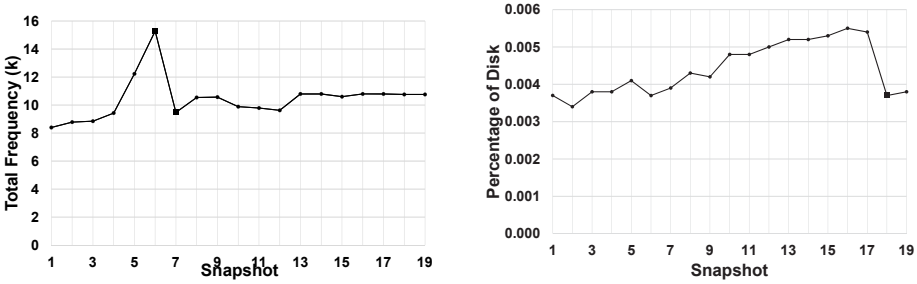


(b) $W = 3$.



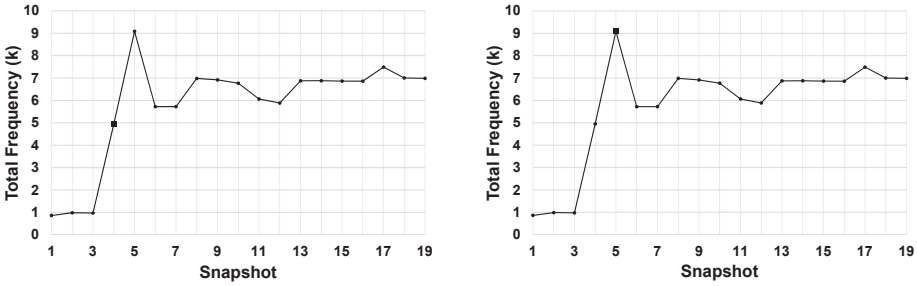
(c) $W = 4$.

Figure 10. Outliers detected via sliding window (Data type: JPEG).



(a) Data type: URL domain. (b) Data type: video.

Figure 11. Autoregression detection of spike (left) and at edge (right).



(a) Split window method. (b) Autoregressive method.

Figure 12. Outlier detection (Data type: top-ninth most frequent URL domain).

3.3 Autoregressive Method

The first-degree autoregressive model proved to be the most reliable of the three methods. It detected the most outliers, it was the most consistent in rank ordering outliers based on statistical significance and it does not appear to have some of the detection limitations of the other methods. In particular, when compared with the other methods, especially the split window method, it was better able to detect spikes in the time series (Figure 11(a)), outliers at the edges (beginning and ending observations in the time series in Figure 11(b) for the top-third most frequent URL domain data type). Also, it consistently identified as an outlier the more intuitive, successive observation, rather than the less intuitive, precipitory observation (Figures 12(a) and 12(b)). In both figures, the successive fourth and fifth observations are identified as outliers compared with the precipitory third observation.

3.4 Ground Truth Analysis

To establish ground truth and thereby evaluate the validity of detected outliers and identify false negatives, investigative interrogatories pertaining to the detected outliers as well as general investigative interrogatories pertaining to the case scenario to identify false negatives were developed. A trained digital forensic investigator analyzed the disk images using the interrogatories. The forensic analysis, when compared against the anomalies detected via time series analysis, identified nine true positives and two false positives. A true positive occurred when the forensic analysis confirmed that the drive snapshot did indeed contain an anomalous number of data/files of a specified type – whether benign or nefarious in nature. A false positive occurred when the results of the forensic analysis suggested that the drive should not have been flagged as anomalous by the outlier detection system.

The two false positives were identified as a result of issues with the outlier detection system design. First, it was determined that the file extension list for video files was overly broad and included extensions that are not exclusively used for video file types. This resulted in a statistical anomaly that would not have been anomalous if the video file type was defined more narrowly and reliably. Second, the approach failed to detect recycle bin content. If the recycle bin content had been detected, the second false positive anomaly would not have been statistically anomalous because the forensic traces of the data still existed on the disk; they were reported as missing because recycle bin content was omitted from the analysis.

Of the nine true positives that were identified, forensic analysis revealed that seven were benign anomalies. In other words, the anomalous activity was explained by legitimate circumstances (e.g., job role/task change) and activity (e.g., system activity related to infrequent system logging during the period of analysis). Two true positive cases were confirmed to be (synthetic) illegal behavior, specifically: (i) possession of illegal graphic images; and (ii) installation of a keylogger.

False negatives are somewhat challenging to define in this context. On the one hand, no false negatives were encountered from a statistical perspective. However, from an investigative perspective, the outlier detection method failed to detect two pieces of evidence that could have been detected via time-series-based analysis, if not for two extenuating circumstances. First, the same unauthorized keylogger that was detected on user Pat's machine via time series analysis of file types was not detected on user Terry's computer through the same means. This is likely because the keylogger stored its log files in HTML and Terry's

drive had a significant amount of HTML data as a result of much more web browsing activity than Pat. Second, Terry had a great deal of unauthorized screen captures of Pat's machine stored in the JPEG format, but these screen captures were missed by time-series-based anomaly detection. Again, this is likely because Terry's extensive web browsing activity masked this evidence from a time series perspective, given the large number of `.jpg` files stored in the web cache on the drive.

4. Conclusions

Time-series-based analysis, specifically first-order autoregressive modeling, successfully identified statistical anomalies with a direct investigative payoff. The number of true positives exceeded the number of false positives (nine versus two) and the false negatives were due to outlier detection system design errors, not problems with the anomaly detection method. While only two of the nine true positives were malicious, meaning that the number of investigatively-irrelevant true positives exceeded the number of investigatively-relevant true positives, this is nothing new in digital forensics. Text string searches typically yield 95% or more irrelevant search hits from an investigative perspective. They are not false positives from a search perspective; they simply are not germane to the investigation. Similarly, the false positives were indeed statistically anomalous; they simply were not germane to the investigation. Not only is the 70% rate of benign statistical anomalies an improvement over what is typically experienced in text string search (>95%), but it is also important to note that the total number of anomalies that have to be assessed for benign or malicious intent is a very small fraction of what text string search and other digital forensic techniques encounter. It is also important to remind users of the proposed method that the outliers are associated with p-values, which could be rank ordered to enable analysts to examine the more outlying observations first and analyze the less outlying observations as resources permit. Indeed, the results demonstrate that a time-series-based method for statistical disk profiling can detect insider threat activity with a manageable ratio of benign to malicious root causes and the ability to rank order the outliers.

Two key limitations of the dataset used in this research impact the research findings. First, the dataset is synthetic, which limits the external validity and generalizability of the research findings. Second, the data is limited in the number of observations. Approximately nineteen time series observations were available for each synthetic user. More observations would have been better, but suitable test datasets in the digital forensics field are difficult to come by. Robust synthetic digital forensic

cases are very rare and real-world datasets have access restrictions and the results are generally not reproducible by other researchers.

Note that the views expressed in this chapter do not necessarily reflect the official policies of the Naval Postgraduate School nor does the mention of trade names, commercial practices or organizations imply an endorsement by the U.S. Department of Homeland Security or the U.S. Government.

Acknowledgement

This research was sponsored by the U.S. Department of Homeland Security Science and Technology Directorate, Cyber Security Division (DHS S&T CSD) via Contract No. N6600112WX01362 under a Cooperative Agreement No. N00244-13-2-0004 with the Naval Postgraduate School.

References

[1] S. Band, D. Cappelli, L. Fischer, A. Moore, E. Shaw and R. Trzeciak, Comparing Insider IT Sabotage and Espionage: A Model-Based Analysis, Technical Report CMU/SEI-2006-TR-026, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, 2006.

[2] V. Barnett and T. Lewis, *Outliers in Statistical Data*, John Wiley and Sons, New York, 1994.

[3] S. Boss, D. Galletta, P. Lowry, G. Moody and P. Polak, What do users have to fear? Using fear appeals to engender threats and fear that motivate protective security behaviors, *Management Information Systems Quarterly*, vol. 39(4), pp. 837–864, 2015.

[4] H. Chivers, J. Clark, P. Nobles, S. Shaikh and H. Chen, Knowing who to watch: Identifying attackers whose actions are hidden within false alarms and background noise, *Information Systems Frontiers*, vol. 15(1), pp. 17–34, 2013.

[5] D. Costa, M. Collins, S. Perl, M. Albrethsen, G. Silowash and D. Spooner, An Ontology for Insider Threat Indicators: Development and Application, *Proceedings of the Ninth Conference on Semantic Technology for Intelligence, Defense and Security*, pp. 48–53, 2014.

[6] D. Dishneau, Army general upholds Chelsea Manning's conviction, 35-year sentence in WikiLeaks case, *U.S. News and World Report*, April 14, 2014.

[7] F. Farahmand and E. Spafford, Understanding insiders: An analysis of risk-taking behavior, *Information Systems Frontiers*, vol. 15(1), pp. 5–15, 2013.

[8] J. Gallu, Snowden used "web crawler" to scrape NSA: New York Times, *Bloomberg Technology*, February 9, 2014.

[9] S. Garfinkel, M57-Patents Scenario, Digital Corpora (`digitalcorpora.org/corpora/scenarios/m57-patents-scenario`), 2017.

[10] S. Garfinkel, P. Farrell, V. Roussev and G. Dinolt, Bringing science to digital forensics with standardized forensic corpora, *Digital Investigation*, vol. 6(S), pp. S2–S11, 2009.

[11] K. Guo, Y. Yuan, N. Archer and C. Connelly, Understanding nonmalicious security violations in the workplace: A composite behavior model, *Journal of Management Information Systems*, vol. 28(2), pp. 203–236, 2011.

[12] M. Hanley and J. Montelibano, Insider Threat Control: Using Centralized Logging to Detect Data Exfiltration Near Insider Termination, Technical Note CMU/SEI-2011-TN-024, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, 2011.

[13] K. Herbig and M. Wiskoff, Espionage Against the United States by American Citizens 1947–2001, Technical Report 02-5, Defense Personnel Security Research Center, Monterey, California, 2002.

[14] P. Huber and E. Ronchetti, *Robust Statistics*, John Wiley and Sons, Hoboken, New Jersey, 2009.

[15] L. Kramer, R. Heuer and K. Crawford, Technological, Social and Economic Trends that are Increasing U.S. Vulnerability to Insider Espionage, Technical Report 05-10, Defense Personnel Security Research Center, Monterey, California, 2005.

[16] M. Maasberg, Insider espionage: Recognizing ritualistic behavior by abstracting technical indicators from past cases, *Proceedings of the Twentieth Americas Conference on Information Systems*, 2014.

[17] Mandiant, M-Trends 2015: A View from the Front Line, Threat Report, Alexandria, Virginia, 2014.

[18] A. Moore, D. McIntire, D. Mundie and D. Zubrow, The justification of a pattern for detecting intellectual property theft by departing insiders, *Proceedings of the Nineteenth Conference on Pattern Languages of Programs*, article no. 8, 2012.

[19] Ponemon Institute, 2015 Cost of Data Breach Study: Global Analysis, Ponemon Institute Research Report, Traverse City, Michigan, 2015.

[20] P. Rousseeuw and A. Leroy, *Robust Regression and Outlier Detection*, John Wiley and Sons, Hoboken, New Jersey, 2003.

[21] K. Singh and S. Upadhyaya, Outlier detection: Applications and techniques, *International Journal of Computer Science Issues*, vol. 9(1), pp. 307–323, 2012.

[22] Vormetric Data Security, 2015 Vormetric Insider Threat Report, San Jose, California, 2015.

[23] J. Wang, M. Gupta and R. Rao, Insider threats in a financial institution: Analysis of attack-proneness of information systems applications, *Management Information Systems Quarterly*, vol. 39(1), pp. 91–112, 2015.

[24] R. Willison and M. Warkentin, Beyond deterrence: An expanded view of employee computer abuse, *Management Information Systems Quarterly*, vol. 37(1), pp. 1–20, 2013.