A Comparative Study of Anomaly Detection Techniques in Web Site Defacement Detection

Giorgio Davanzo, Eric Medvet and Alberto Bartoli

Abstract Web site defacement, the process of introducing unauthorized modifications to a web site, is a very common form of attack. Detecting such events automatically is very difficult because web pages are highly dynamic and their degree of dynamism, as well as their typical content and appearance, may vary widely across different pages. Anomaly based detection can be a feasible and effective solution for this task because it does not require any prior knowledge about the page to be monitored. This study enables gaining further insights into the problem of automatic detection of web defacements. We want to ascertain whether existing techniques for anomaly intrusion detection may be applied to this problem and we want to assess pros and cons of incorporating domain knowledge into the detection algorithm.

1 Introduction

The defacement of web sites has become one of the most widely diffused security incident categories in the Internet and "continues to plague organizations" [4]. Anecdotal evidence about this phenomenon is abundant and steadily expanding (e.g., http://www.zone-h.org, http://www.serapis.net/), with more than 480.000 defacements stored at Zone-H, a public web-based archive devoted to gathering evidences of defacements, during 2007 alone. A defaced web site typically contains only a few messages or images, perhaps including disturbing contents and/or political messages.

We proposed a technology for implementing a *defacement detection service*, in which a single organization could monitor the web sites of hundreds or thousands of web sites of other organizations [1]. The crucial feature of our proposal is that it does not require any participation from the monitored site, in particular, it does not require the installation of any infrastructure at the monitored site, nor does it require the knowledge of the officially approved content of the web page. Our approach is

DEEI - Università degli Studi di Trieste

based on *anomaly detection*: we build automatically a profile of the monitored web page and then generate an alert to the relevant monitored organization whenever something "unusual" shows up.

In this work we broaden our analysis and compare our domain-knowledge based approach with anomaly detection techniques that have been proposed earlier for host/network-based intrusion detection systems. This study enables gaining further insights into the problem of automatic detection of web defacements. We want to ascertain whether techniques for anomaly-based intrusion detection may be applied also to anomaly detection of web defacements and we want to assess pros and cons of incorporating domain knowledge into the detection algorithm.

2 Experimental Evaluation

We provide only essential information about our monitoring framework. Full details can be found in the accompanying report [3]. We observe the monitored resource at regular intervals. Each reading of the resource is transformed into a numerical array of 1466 elements. This array is then classified by a detector as being *negative* (legitimate) or *positive* (anomalous).

We experimented with the following detectors, that implement techniquespreviously used for detecting intrusions in host or in network based IDSs. We used as baseline the detector based on domain-knowledge that we developed in our earlier work.

- **K-th Nearest** [9, 6] is distance-based, often computed using Euclidean metric; basically it measures the minimum distance required to include at least k points: an anomaly is detected when that distance is greater than a provided threshold.
- **Local Outlier Factor** [2, 6] is an extension to the k-th nearest distance, assigning to each evaluated point an outlying degree.
- **Mahalanobis Distance** [7, 6] is a measure based on the correlation between variables; an anomaly is detected when the distance of the inspected value from the mean is greater than that of the element composing the profile.
- **Hotelling's T-Square** [5, 10] is very similar to Mahalanobis distance; its main difference is that it considers the number of sampled elements.
- **Parzen Windows** [11] provide a method to estimate the probability density function of a random variable;we experimented with two basic distributions: Gaussian and Pulse.
- **Support Vector Machines** [8, 6] uses hyperplanes to maximally separate N classes of data. In anomaly detection, only $N = 2$ classes of objects are used, providing positive readings during the learning.

We use each technique to build a profile by passively observing the inspected resource—our experience shows that profiling for two weeks is enough. When the profile is complete, the system is then able to judge the content of the observed page by comparing it against the profile accordingly with the given technique.

We observed 15 web pages for two months, collecting a reading for each page every 6 hours, thus totaling a *negative sequence* of 240 readings for each web page. We visually inspected them in order to confirm the assumption that they are all genuine. The observed pages differ in size, content and dynamism and include pages of e-commerce web sites, newspapers web sites, and alike. We also collected an *attack archive* composed by 95 readings extracted from a publicly available defacements archive (http://www.zone-h.org).

We used False Positive Ratio (FPR) and False Negative Ratio (FNR) as performance indexes computing average, maximum and minimum values among web pages. For each page: (i) we built a *learning sequence* S^+ of positive readings composed by the first 20 elements of the attack archive; (ii) we built a sequence S^- of negative readings composed by the first 50 elements of the corresponding negative sequence; (iii) we trained the SVM detector with $S = S^- \cup S^+$ and all the other detectors with $S = S^-$. Then, for each page:(i) we built a *testing sequence* S_t by joining a sequence S_t^-, composed by the remaining 190 readings of the corresponding negative sequence, and a sequence S_t^+, composed by the remaining 75 readings of the attack archive; (ii) we fed each algorithm with each reading of S_t—as if it was the first reading to be evaluated after the learning phase—counting false positives and false negatives.

3 Results

A first crucial result is that only the detector based on domain-knowledge delivers acceptable results when fed with the entire array of 1466 features. A *feature selection* turned out to be necessary for all the other techniques. The results below have been obtained with detectors focussing only on 10-20 features selected as described in the companion report.

We provide FPR that we obtained experimenting first with the first 75 readings of S_t^- and then on all the 190 readings of S_t^-. In other words, we assessed the effectiveness of each technique separately on the *short term* and on the *long term*—about 19 and 48 days respectively. We also provide FNR computed on all readings of S_t^+.

Table 1 shows results obtained in the short term: FNR values suggest that all the algorithms proved to be effective when detecting defacements. On the other hand, they behaved differently when analyzing genuine readings.

Domain Knowledge performed well on many web pages, although on some of them it exhibited very high FPR; Mahalanobis, Hotelling and LOF did not score well, being unable to classify genuine pages for many pages. Both Parzen methods proved to be acceptably effective on many pages, although on some of them they worked as badly as Mahalanobis and Hotelling.

An excellent result comes from K-th Nearest and Support Vector Machines: both techniques managed to correctly classify all the negative readings, while still detecting a large amount of attacks.

Table 1 Short term results (75 readings).

Aggregator	FPR %			FNR %		
	AVG	MAX	MIN	AVG	MAX	MIN
Domain Knowledge	1.3	13.3	0.0	0.1	1.3	0.0
K-th Nearest	0.0	0.0	0.0	0.1	1.3	0.0
Local Outlier Factor	6.6	94.7	0.0	0.3	4.0	0.0
Hotelling	10.9	94.7	0.0	0.0	0.0	0.0
Mahalanobis	11.5	94.7	0.0	0.2	1.3	0.0
Parzen Pulse	1.2	16.0	0.0	0.0	0.0	0.0
Parzen Gaussian	4.1	40.0	0.0	0.0	0.0	0.0
Support Vector Machines	0.0	0.0	0.0	0.0	0.0	0.0

Table 2 shows results obtained in the long term. FNR is the same as Table 1, since both aggregator internal state and the positive testing sequence S_t^+ remain the same. Results in terms of FPR are slightly worse for all the evaluated techniques, as expected; the only aggregator that managed to perform almost as good as in short term is the one based on the Support Vetor Machines. As a matter of fact, both K-th Nearest and Pulse Parzen managed to maintain a low FPR, but raised many false alarms on some web pages.

Table 2 Long term results (190 readings)

Aggregator	FPR %			FNR %		
	AVG	MAX	MIN	AVG	MAX	MIN
Domain Knowledge	2.7	26.3	0.0	0.1	1.3	0.0
K-th Nearest	1.8	18.9	0.0	0.1	1.3	0.0
Local Outlier Factor	11.0	97.9	0.0	0.3	4.0	0.0
Hotelling	14.7	97.9	0.0	0.0	0.0	0.0
Mahalanobis	16.2	97.9	0.0	0.2	1.3	0.0
Parzen Pulse	1.3	15.3	0.0	0.0	0.0	0.0
Parzen Gaussian	4.5	40.0	0.0	0.0	0.0	0.0
Support Vector Machines	0.0	0.5	0.0	0.0	0.0	0.0

According to our experimental evaluation, almost all techniques (with the exception of LOF and Hotelling/Mahalanobis) show results, in terms of FNR and FPR, which are sufficiently low to deserve further consideration. In particular, most techniques achieve an average FPR lower than 4%, while being able to correctly detect almost all the simulated attacks (FNR $\simeq$ 1%). We remark that such a lower FPR is equivalent, in our scenario, to about 4 false positives raised every month. Such a finding suggests that, with most of the proposed techniques, the realization of a large-scale monitoring service is a feasible solution.

Support Vector Machines are the most promising alternative to our earlier Domain Knowledge proposal in terms of effectiveness. Since that technique requires

an archive of attacks, however, it may be useful to investigate more deeply the relation between quality of that archive and resulting performance.

On the other hand, we believe that a domain knowledge-based detection algorithm benefits from two important advantages:

First, an intrinsic feature of Domain Knowledge is that the framework can provide the human operator with meaningful indications in case of a positive. For example, the operator could be notified with some indication about an anomalous number of links in the page or about a tag that was not present in the page despite being supposed to be. These indications can hardly be provided using the other techniques.

Second, the Domain Knowledge aggregator does not require a feature selection. While increasing performance for the techniques we tested, thus definitely making defacement detection effective with them, feature selection introduces more opportunities for attacks that remain hidden within the analyzed profile. Any attack affecting only elements that are not taken into account after the feature selection, cannot be detected by a detection algorithm that requires feature selection. The practical relevance of this issue (i.e., which attacks indeed fall in this category) certainly deserves further analysis.

References

1. Bartoli, A., Medvet, E.: Automatic integrity checks for remote web resources. IEEE Internet Computing **10**, 56–62 (2006)
2. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: Lof: identifying density-based local outliers. SIGMOD Rec **29**, 93–104 (2000)
3. Davanzo, G., Medvet, E., Bartoli, A.: A comparative study of anomaly detection techniques in web site defacement detection - long version (2008). URL http://www2.units.it/bartolia/download/ComparativeStudyLong.pdf
4. Gordon, L., Loeb, M., Lucyshyn, W., Richardson, R.: CSI/FBI Computer Crime and Security Survey. Computer Security Institute (2006)
5. Hotelling, H.: The generalization of student's ratio. The Annals of Mathematical Statistics **2**, 360–378 (1931)
6. Lazarevic, A., Ertoz, L., Kumar, V., Ozgur, A., Srivastava, J.: A comparative study of anomaly detection schemes in network intrusion detection. Proceedings of the Third SIAM International Conference on Data Mining (2003)
7. Mahalanobis, P.C.: On the generalized distance in statistics. In: Proceedings of the National Institute of Science of India, vol. 12, pp. 49–55 (1936)
8. Mukkamala, S., Janoski, G., Sung, A.: Intrusion detection using neural networks and support vector machines. In: Neural Networks, 2002. IJCNN '02. Proceedings of the 2002 International Joint Conference on, vol. 2, pp. 1702–1707 (2002)
9. Ramaswamy, S., Rastogi, R., Shim, K.: Efficient algorithms for mining outliers from large data sets. SIGMOD Rec **29**, 427–438 (2000)
10. Ye, N., Chen, Q., Emran, S.M., Vilbert, S.: Hotelling t2 multivariate profiling for anomaly detection. Proc. 1st IEEE SMC Inform. Assurance and Security Workshop (2000)
11. Yeung, D.Y., Chow, C.: Parzen-window network intrusion detectors. In: Pattern Recognition, 2002. Proceedings. 16th International Conference on, vol. 4, pp. 385–388 vol.4 (2002)