

# Self-Propagating Malware Detection using Information-Theoretic Tools



Syed Ali Khayam and Hayder Radha  
 Department of Electrical & Computer Engineering, Michigan State University (MSU)  
 khayam@ieee.org, radha@egr.msu.edu

## PROJECT OBJECTIVE

Network endpoints, comprising of client machines at homes and offices, are now serving as extremely potent and viable launch pads and carriers for malware infections. Thus it is important that real-time defenses be developed specifically for these endpoints.

The objective of this project is to design endpoint-based malware detectors that use sound theoretical tools to detect anomalous activities.

## MALWARE DETECTION USING INFORMATION DIVERGENCE OF NETWORK TRAFFIC

The first malware detector that we have developed in this work is a network-based detector which relies on the premise that the vulnerabilities targeted by self-propagating malware are associated with a small number of source or destination ports. Thus on a compromised machine, the distribution of source or destination ports on which a host communicates should be perturbed after infection. To quantify these perturbations, we use an information divergence measure, namely the *Kullback-Leibler (K-L) divergence*.

To effectively leverage K-L divergence for malware detection, we generate histograms of benign source and destination port usage from data collected on 13 uninfected endpoints. These histograms are compared against the real-time traffic observed on the endpoints to detect anomalies. For testing, malware traffic was mixed with an endpoint's benign traffic at random points. Fig. 1 shows the K-L perturbations caused by the malicious traffic.

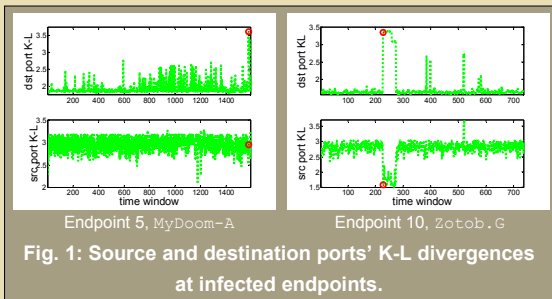


Fig. 1: Source and destination ports' K-L divergences at infected endpoints.

## MALWARE DETECTION USING SESSION-KEYSTROKE MUTUAL INFORMATION

The second malware detector developed in this work is a joint network-host anomaly detector, which exploits the observation that when a user is actively using his/her computer most of the benign traffic is triggered by a small subset of keystrokes and mouse clicks. Based on this observation, we propose to correlate the last input from the keyboard or mouse hardware buffer with every new network session to detect malicious activities.

Fig. 2 shows the usage histograms of keys that are used to initiate sessions and the keys that are generally used on an endpoint. Clearly, the two histograms are different from each other as network sessions are mostly initiated using a small subset of keys. To leverage the correlation between network sessions and session-initiation keys, we use the *mutual information* measure.

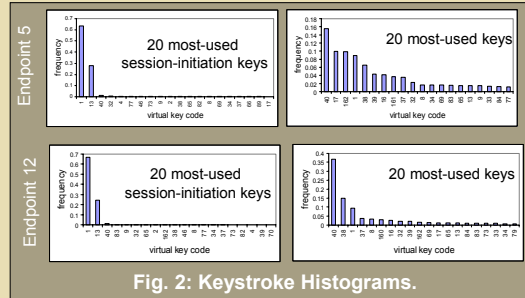


Fig. 2: Keystroke Histograms.

## Approach and Impact

### Salient Features

- Designed specifically for network endpoints
- Anomaly detection without relying on existing malware signatures or characteristics
- Low complexity
- Platform-independent, can be deployed on any endpoint with a graphical OS

### Research Impact

- Novel application of information theory to the anomaly detection problem
- Can be deployed on endpoints having dial-up to broadband connectivity
- Jointly exploits network and host benign features
- Collected data will be publicly available for future research

## Technical Details and Performance Evaluation

### BENIGN ENDPOINT DATA

Using the WinPcap API, we have written a low-complexity application for data monitoring and collection on operational endpoints. Using this application, we have been collecting benign data on 13 diverse endpoints for more than 12 months. Each entry of the collected data comprises the following fields: <session id, direction, transport protocol, src port, dst port, timestamp, virtual key code of session-initiation key>

The endpoints used for data collection included home, office, and research lab computers running different types (e.g. p2p, multimedia, gaming, SQL/SAS etc.) of applications.

### MALWARE DATA

For realistic experimentation, we collected active malware by connecting unpatched Windows machines to the Internet. The network administrators at MSU also provided us with some malware binaries. Moreover, we downloaded the source code and binaries of some malware from the Internet. Finally, we simulated some malware that we deemed important for testing of our proposed detectors. Details of the collected malware are provided in Table 1.

Malware	Release Date	Avg. Scan Rate (scans per sec)	Port(s) Used
Blaster	Aug 2003	10.5	TCP 135, 4444, UDP 69
Dloader-NY	Jul 2005	46.84	TCP 135, 139
Forbot-FU	Sep 2005	32.53	TCP 445
MyDoom-A	Jan 2006	0.14	TCP 3217-3198
RBOT_OCC	Aug 2005	9.7	TCP 139, 445
Rbot-AQJ	Oct 2005	0.68	TCP 139, 769
Sdbot-AFR	Jan 2006	28.26	TCP 445
SoBig_E	Jan 2003	21.57	TCP 135, UDP 53
Zotob.G	Jun 2003	39.34	TCP 135, 445, UDP 137
Witty (simulated)	Mar 2004	357.0	UDP 4000
CodeRed II (simulated)	Aug 2001	4.95	TCP 80

### DETAILS OF THE DETECTORS

Let  $X_n = \{p_i^n, i \in S_n\}$  denote the normalized histograms of source ports observed in time-window  $n$ , and let  $X = \{p_i, i \in S\}$  denote the normalized histogram observed in the benign training data. Then the K-L divergence between these histogram can be computed as:

$$D(X_n || X) = \sum_{i \in S_n} \frac{p_i^n}{p_n} \log_2 \frac{p_i^n / p_n}{p_i / p}$$

where  $p = \sum_{i \in S} p_i$ . The K-L divergence of destination ports is computed similarly. In addition to K-L, we also experimented with the Jensen-Shannon,  $K$  directed, and Resistor Average information divergences, which are defined as:

$$J(X_n || X) = H(\pi_1 X_n + \pi_2 X) - H(\pi_1 X_n) - H(\pi_2 X), \quad H(\cdot) : \text{Entropy function}$$

$$\pi_1 + \pi_2 = 1$$

$$K(X_n || X) = \sum_{i \in S_n} \frac{p_i^n}{p_n} \log_2 \frac{p_i^n / p_n}{\frac{p_i^n + p_i}{2p_n}}, \quad R(X_n || X) \equiv \frac{1}{D(X_n || X)} + \frac{1}{D(X || X_n)}$$

These three measures, however, did not provide any performance improvement. We use benign and malicious K-L values to train support vector machines (SVMs), which are in turn employed to classify anomalous behavior.

To compute session-key mutual information, we define a session random variable (r.v.)  $X$  and a keystroke random variable  $Y$ , with marginal distributions  $p(x)$  and  $p(y)$ , respectively.  $X$  is a binary r.v. indicating the existence of a session in the current time-window, and  $Y$  is a r.v. computed by normalizing a keystroke histogram. Then the mutual information of these random variables is:

$$I(X; Y) = \sum_{x, y} p(x, y) \log_2 \frac{p(x, y)}{p(x)p(y)}$$

We use mean and standard deviation of benign mutual information values to set thresholds above and below which an alarm is raised.

Fig. 3 shows the accuracies of both detectors; results are averaged over 100 random infections per endpoint per malware.

### Published and Submitted Work Based on CyberTrust Funding

- [1] S. A. Khayam and H. Radha, "Using Session Keystroke Mutual Information to Detect Self Propagating Malicious Codes," accepted to IEEE ICC, June 2007.
- [2] S. A. Khayam and H. Radha, "Worm Detection at Network Endpoints using Information Theoretic Traffic Perturbations," submitted to IEEE ICDCS 2007.
- [3] S. Soltani, S. A. Khayam, and H. Radha, "Detecting Malware Outbreaks using a Statistical Model of Blackhole Traffic," submitted to IEEE ICDCS 2007.
- [4] S. A. Khayam and H. Radha, "Using Signal Processing Techniques to Model Worm Propagation over Wireless Sensor Networks," IEEE Signal Processing Magazine, (23(2), 164-169, March 2006. An earlier version of this paper appeared in IEEE ICDCS International Workshop on Security in Distributed Computing Systems (SDCS), October 2006.
- [5] S. A. Khayam and H. Radha, "Analyzing the Spread of Active Worms over VANET," ACM Mobicom International Workshop on Vehicular Ad Hoc Networks (VANET), September 2004.
- [6] S. A. Khayam and H. Radha, "Modeling Worm Propagation over Vehicular Networks," accepted to SAE Transactions. An earlier version of this paper appeared in SAE 2006 World Congress, April 2006.

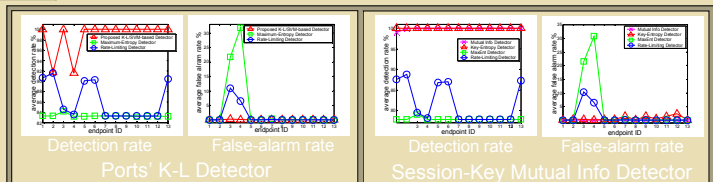


Fig. 3: Detection and false-alarm rates of the proposed detectors.

