Flow-Data Compressibility Changes During Internet Worm Outbreaks

Arno Wagner Bernhard Plattner
Communication Systems Laboratory, Swiss Federal Institute of Technology Zurich
Gloriastr. 35, CH-8092 Zurich
Contact Author: Arno Wagner, wagner@tik.ee.ethz.ch

Abstract

During outbreaks of fast Internet worms the characteristics of network flow data from backbone networks changes. We have observed that in particular source and destination IP and port fields undergo compressibility changes, that are characteristic for the scanning strategy of the observed worm. In this paper we present measurements done on a medium sized Swiss Internet backbone (SWITCH, AS559) during the outbreak of the Blaster and Witty Internet worms and attempt to give a first explanation for the observed behaviour. We also discuss the impact of sampled versus full flow data and different compression algorithms. This is work in progress. In particular the details of what exactly causes the observed effects are still preliminary and under ongoing investigation.

1. Entropy and Compressibility

Generally speaking entropy is a measure of how random a data-set is. The more random it is, the more entropy it contains. Entropy contents of a (finite) sequence of values can be measured by representing the sequence in binary form and then using data compression on that sequence. The size of the compressed object corresponds to the entropy contents of the sequence. If the compression algorithm is perfect (in the mathematical sense), the measurement is exact.

On the theoretical side it is important to understand that not entropy is the relevant traffic characteristic, but Kolmogorov Complexity [16] of an interval of data. While entropy describes the average expected information content of a symbol that is chosen in a specific randomised way from a specific symbol set, Kolmogorov Complexity describes the specific information content of a specific object given, e.g. as a binary string of finite length.

2. Measurements

We are collecting NetFlow v5 [10] data from the SWITCH (Swiss Academic and Research Network [4], AS559) network, a medium-sized Swiss backbone operator, which connects all Swiss universities and various research labs (e.g. CERN) to the Internet. Unsampled NetFlow data from all four SWITCH border routers is captured and stored for research purposes in the context of the DDoSVax project [11] since early 2003. The SWITCH IP address range contains about 2.2 million IP addresses. In 2003 SWITCH carried around 5% of all Swiss Internet traffic [17]. In 2004, we captured on average 60 million NetFlow records per hour, which is the full, non-sampled number of flows seen by the SWITCH border routers.

In Figures 1,2,3 and 4 we plot the entropy estimations by compressibility over time for source and destination IP addresses and ports for the Blaster [9, 6, 15] and Witty [18, 20] worm. Both worms are relatively well understood and well documented. First observed on August 11th, 2003, Blaster uses a TCP random scanning strategy with fixed destination and variable source port to identify potential infection targets and is estimated to have infected 200'000...500'000 hosts worldwide in the initial outbreak. The Witty worm, first observed on March 20th, 2004, has some unexpected characteristics. Witty attacks a specific firewall product. It uses UDP random scans with *fixed* source port and *variable* destination port. Witty infected about 15'000 hosts in less than 20 minutes.

The y-axis in the plots gives inverse compression ration, i.e. lower values indicate better compressibility. The plotted time intervals start before the outbreaks to illustrate normal traffic compressibility characteristics. Samples taken from other times in 2003 and 2004 indicate that the pre-outbreak measurements, were source and destination figures are close together, are characteristic for non-outbreak situations. The outbreak times of both worms are marked with arrows.

The given measurements were done both on the full SWITCH flow set as well as on a 1 in 20 sample. Com-

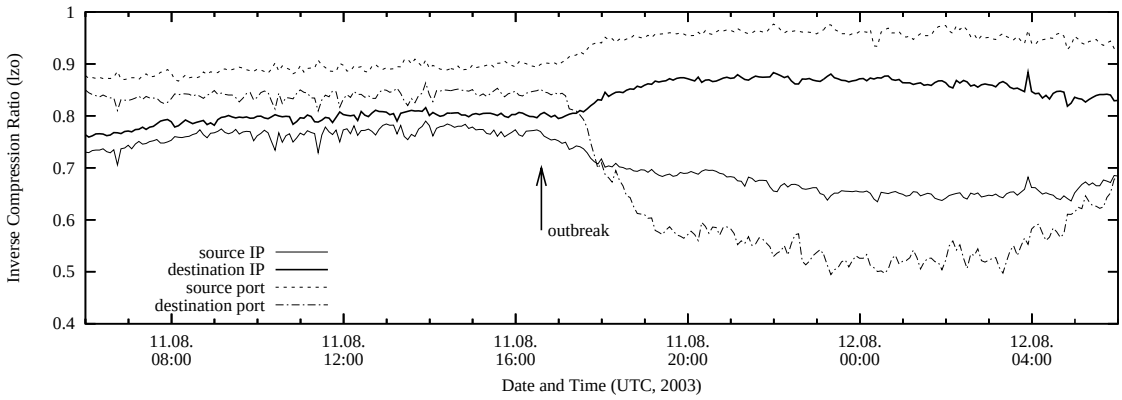

Figure 1. Blaster - TCP address parameter compressibility (lzo1x-1 algorithm)

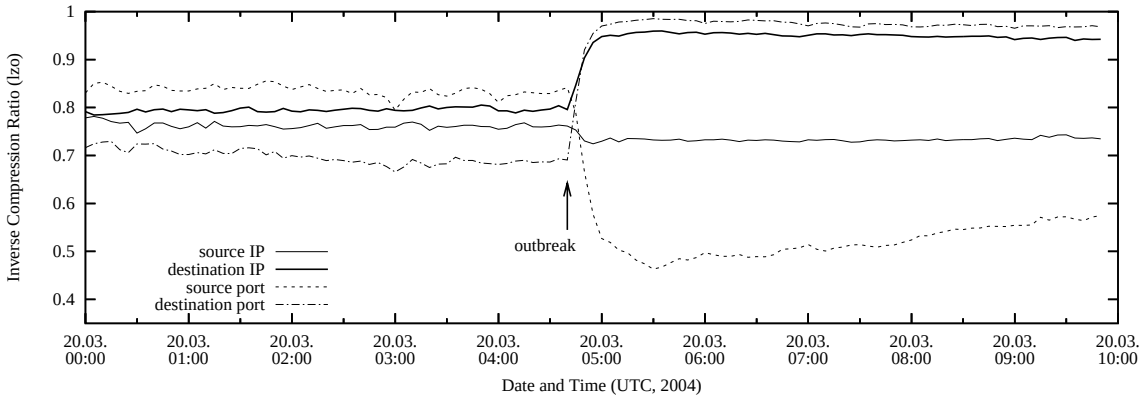

Figure 2. Witty - UDP address parameter compressibility (lzo1x-1 algorithm)

pression algorithm used is the fast lzo algorithm lzo1x-1 (see Section 4). It can be seen that in both cases the compressibility plots change significantly during the outbreak. Changes are consistent with the intuition that more random date is less compressible, while more structured date can be compressed better. The measurements on sampled data show a vertical shift, but still exhibit the same characteristic changes.

3. Analysis

In normal traffic there is roughly one return flow to a host for each flow it sends out as connection initiator. During a worm outbreak, most scanning flows do not have a return flow. This causes the changes in the overall flow data to be strongly dependent to the characteristics of the flows generated for scanning connection attempts. Note that the absence of an answering flow does not mean the absence of a host at the target address. It can also be due to firewalls, filters and not running services.

The connection between entropy and worm propagation is that worm scan-traffic is more uniform or structured than normal traffic in some respects and a more random in others. The change in IP address characteristics seen on a flow level is intuitive: few infected hosts try to connect to a lot of other hosts. If these flows grow to be a significant part of the set of flows seen in total, the source IP addresses of the scanning hosts will be seen in many flows and since they are relatively few hosts, the source IP address fields will contain less entropy per address seen than normal traffic. On the other hand the target IP addresses seen in flows will be much more random than in normal traffic. These are fundamental characteristics of any worm outbreak where each infected host tries to infect many others.

For ports, the behaviour is more variable. The typical scanning behaviour will be a random (from an OS selected range) or fixed source port and a fixed destination port. In the Blaster plots the impact of random source port and fixed destination port can be seen clearly. Witty is different. Because it did scan with fixed source port and random target port (because it attacked a firewall product that sees all network traffic), the port plots show exactly the opposite compressibility changes compared to Blaster.

At this time it in unclear how much weaker a topological worm (i.e. a worm that uses data from the local host to determine scanning targets and does not do random scanning) would influence the flow field compressibility statistics.

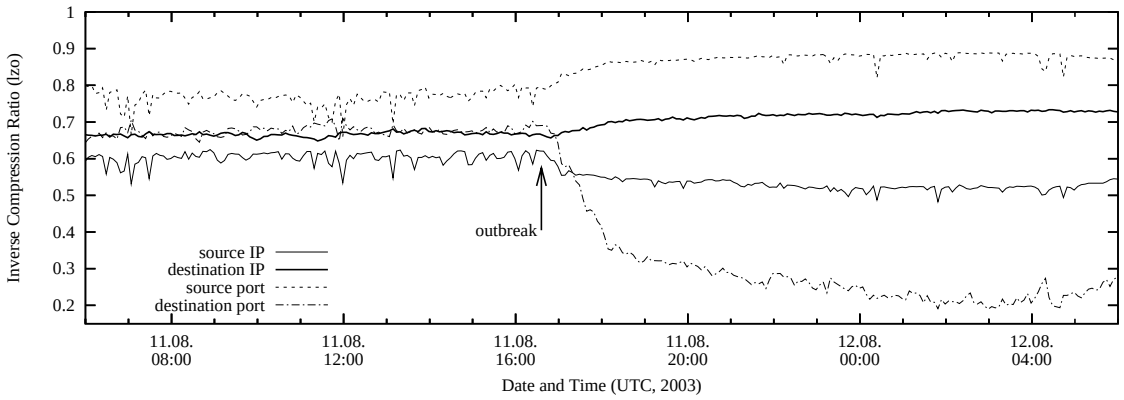

Figure 3. Blaster - TCP, randomly sampled at 1 in 20 flows (lzo1x-1 algorithm)

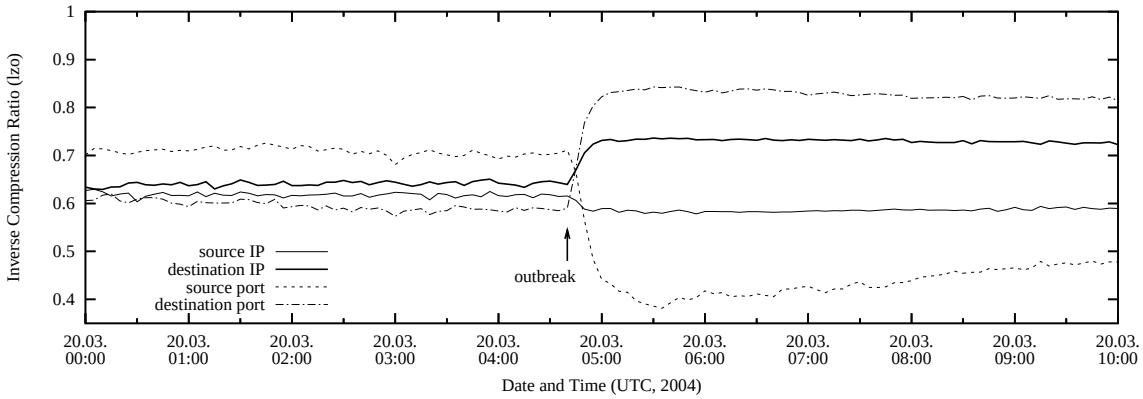

Figure 4. Witty - UDP, randomly sampled at 1 in 20 flows (lzo1x-1 algorithm)

4. Compressor Comparison

Method (Library)	CPU time / hour (60'000'000 flows/hour)
bzip2 (libbz2-1.0)	169 s
gzip (zlib1g 1.2.1.1-3)	52 s
lzo1x-1 (liblzo1 1.08-1)	7 s

Figure 6. CPU time (Linux, Athlon XP 2800+)

We compared three different lossless compression methods, the well-known bzip2 [2] and gzip [3] compressors as well as the lzo (Lempel-Ziv-Oberhumer) [1] real-time compressor. We did not consider lossy compressors. Bzip2 is slow and compresses very well, gzip is average in all regards and lzo family is fast but does not compress well.

Direct comparison of the three compressors on network data shows that while the compression ratios are different, the changes in compressibility are very similar. Figure 5 gives an example plot that compares the compression statistics for destination IP addresses before and during the Witty worm outbreak. Because of its speed advantage lzo1x-1 was selected as preferred algorithm for our work. Note that it is extremely fast (Table 6, non-overlapping measurement intervals of 5 minutes each, includes all overhead like NetFlow record parsing) and uses little memory (64kB for the compressor), making it far more efficient than other methods of entropy estimation, like for example methods based on determining the frequency of individual data values. Since we are only concerned with relative changes, the far from optimal compression ratio of the algorithm does not matter.

5. Related Work

The idea to use some entropy measurements to detect worms has been floating around the worm research community for some time. Yet we are not aware of any publication(s) describing concrete approaches, systems or measurements. The authors of this paper were prompted to investigate this idea by an observation on the Nachi [12, 7, 5] worm: Nachi generated about as many additional ICMP flow records as there were total flow records exported before the outbreak, yet the compressed size of the storage files increased only marginally.

In [19] the authors describe *behaviour-based clustering*, an approach that groups alerts from intrusion detection systems by looking at similarities in the observed packet header fields. The clusters are then prioritised for operator review. Principal Component Analysis is used in [14] to separate normal and attack

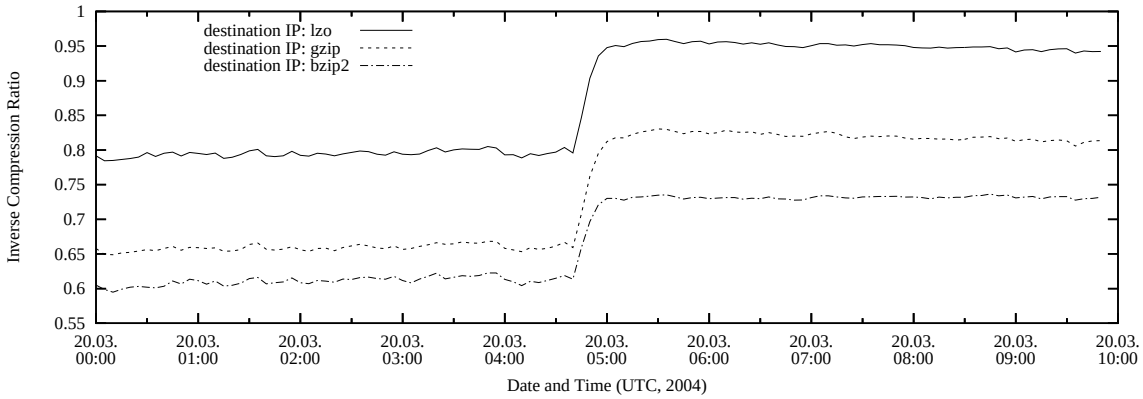

Figure 5. Witty - compressor comparison

traffic on a network-wide scale in a post-mortem fashion. Detection of exponential behaviour in a worm outbreak is studied in [8]. In [13] the authors study how worms propagate through the Internet.

6. Conclusion

We have presented measurements that indicate compressibility analysis of network flow data address fields can be used for the detection of fast worms. The approach is generic and does not need worm-specific parameterisation in order to be effective. It can generate first insights and is suitable for initial alarming, but has limited analytic capability. We are currently investigating how the entropy-based approach can help to generate a more detailed analysis of a massive network event.

References

[1] `http://www.oberhumer.com/opensource/lzo/`. LZO compression library.

[2] The bzip2 and libbzip2 official home page. `http://sources.redhat.com/bzip2/`.

[3] The gzip home page. `http://www.gzip.org/`.

[4] The swiss education & research network. `http://www.switch.ch`.

[5] Mcafee: W32/nachi.worm. `http://vil.nai.com/vil/content/v_100559.htm`, August 2003.

[6] Symantec Security Response - W32.Blaster.Worm. `http://securityresponse.symantec.com/avcenter/venc/data/w32.blaster.worm.html`, 2003.

[7] W32.welchia.worm. `http://securityresponse.symantec.com/avcenter/venc/data/w32.welchia.worm.html`, August 2003.

[8] C. C. Z. an L. Gao, W. Gong, and D. Towsley. Monitoring and Early Warning for Internet Worms. In *Proceedings of the 10th ACM Conference on Computer and Comminication Security*, 2003.

[9] CERT. Security Advisory: MS.Blaster (CA-2003-20). `http://www.cert.org/advisories/CA-2003-20.html`, 2004.

[10] Cisco. White Paper: NetFlow Services and Applications. `http://www.cisco.com/warp/public/cc/pd/iosw/ioft/neflct/tech/napps_wp.htm`, 2002.

[11] DDoSVax. `http://www.tik.ee.ethz.ch/~ddosvax/`.

[12] H. Gabor Szappanos VirusBuster. Virus Bulletin: Virus information and overview - W32/Welchia. `http://www.virusbtn.com/resources/viruses/welchia.xml`, Apr. 2004.

[13] J. Kim, S. Radhakrishnan, and S. K. Dhall. Measurement and Analysis of Worm Propagation on Internet Network Topology. In *Proceedings of the International Conference on Computer Communications and Networks*, 2004.

[14] A. Lakhina, M. Crovella, and C. Diot. Diagnosing network-wide traffic anomalies. In *SIGCOMM*, pages 219–230, 2004.

[15] R. Lemos. MSBlast epidemic far larger than believed. `http://news.com.com/MSBlast+epidemic+far+larger+than+believed/2100-7349_3-5184439.html`, 2004.

[16] M. Li and P. Vitanyi. *An Introduction to Kolmogorov Complexity and Its Applications*. Springer Verlag, second edition edition, 1997.

[17] O. Müller, D. Graf, A. Oppermann, and H. Weibel. Swiss Internet Analysis. `http://www.swiss-internet-analysis.org/`, 2004.

[18] C. Shannon and D. Moore. CAIDA: The Spread of the Witty Worm. `http://www.caida.org/analysis/security/witty/`, 2004.

[19] K. Theriault, D. Vukelich, W. Farell, D. Kong, and J. Lowry. Network traffic analysis unsing behaviour-based clustering. Whitepaper, BBN Technologies, `http://www.bbn.com/docs/whitepapers/NetTrafficAn-Clustering-Theriault10-02.pdf`.

[20] US-CERT. Vulnerability Note: Witty (VU#947254). `http://www.kb.cert.org/vuls/id/947254`, 2004.