# Evaluation of Public Datasets for Intrusion Detection/Prevention System Benchmark*

Khalid Huseynov[1†] , KwangJo Kim[2‡] , Paul D.Yoo[3§]
[1 2]Dep. of Computer Science, Korea Advanced Institute of Science and Technology (Korea)
[3]Dep. of Electrical and Computer Engineering, Khalifa University (UAE)

## ABSTRACT

Thousands of new intrusion patterns appear everyday. Anti-virus industry soon may not be able to deal with the scale of signatures and ease of deployment. Therefore, tools like intrusion detection/prevention systems receive more acceptance nowadays. Even though misuse IDS use similar approach to AV industry, anomaly-based approaches exploit more ingenious algorithms borrowed from data mining field. Normally those algorithms require datasets for training and testing. However, openly published datasets are rarely found and usually subject to privacy issues. Therefore, in this paper, we focus our attention on the available labeled datasets. Firstly, we give description regarding generation of each dataset followed by analysis and feedback. This could be a useful reference point before choosing benchmark for the Intrusion detection/prevention system(IDS/IPS) development by research community.

**Keywords: evaluation dataset, survey, IDS/IPS benchmark, intrusion detection/prevention system, network security.**

## I. Introduction

Current escalating trend in malicious activities makes intrusion detection/ prevention systems(IDS/IPS) integral part in a toolset of any network administrator. Even though signature-based systems have been widely adopted in Anti-Virus(AV) industry, anomaly-based systems still haven't obtained enough trust. This is partially due to the significance of datasets in the training phase of anomaly-based systems. The same detection system can perform totally different in various environments (topology, traffic), and further that performance still changes with different datasets. Therefore, the choice of right dataset is crucial. On the other hand, the assortment of openly available labeled datasets is small, making evaluation of any system burdensome. In this paper, we survey all labeled and few additional datasets currently known to us, and that have been used by research community for Intrusion detection/prevention system training and benchmark. Firstly, this is KDD '99 dataset[1] that have been utilized

* *This research was supported by the KUSTAR-KAIST Institute, Korea, under the R&D program supervised by the KAIST.*

† khalid.huseynov@kaist.ac.kr
‡ kkj@kaist.ac.kr
§ paul.usyd@gmail.com

for long time, but currently isn't recommended to be used as the only evaluation dataset. The safe bet is to use KDD '99 with some recent traffic dataset. Another dataset introduced is UNB dataset[2] that was generated in 2010. This dataset contains mostly all up to date malicious activities as well as some old attack patterns. Recent appearance of botnets and botnet detection techniques launches demand in botnet traffic dataset as well. Thus, P2P botnet traffic[15] was created by ISOT research lab at University of Victoria. This dataset is a blend of datasets replayed on their network.

## II. Review and methodology

One of the important points of any dataset is its realistic behavior both network and traffic wise. Sometimes simulations on small testbeds may not capture the realistic behavior from network and traffic points of view. Another crucial point for IDS training and benchmark is labeling. Most of realistic datasets openly available are non-labeled. The process of labeling itself is laborious and requires expert intervention. Even with assiduous work of expert the results may not be accurate enough to satisfy the rate of false negatives/positives. However, non-labeled data can be used in combination with other datasets as a sample of non-malicious traffic.

For example, one of the publicly available sources of any kind of internet traffic is available through CAIDA[3]. The traffic is realistic but non-labeled including everyday normal activities as well as some DDOS attacks traffic. Endpoint Worm Scan Dataset[4] was originally collected at the University of Michigan network. It can be considered as labeled since it consists of worm and benign traces separated. The problem with this dataset is high anonymity, since only 6 fields of the packet are available with absent payload part. Internet Traffic Archive[5] - still can be considered another source of various non-labeled traffic, even though little old. Sometimes Defcon datasets[6] are used as a sample of malicious activity as well. However, Defcon traffic is too specific, and can not be substitute for real world traffic.

Further, our paper will cover three main labeled datasets of interest. Firstly, each dataset has description and method of generation. Then, pros and cons of each dataset are revealed followed by analysis and final statistics of data.

## III. Datasets

### 3.1 KDD CUP 99 Dataset

#### 3.1.1 Description

KDD CUP '99 is a labeled dataset that was extracted from DARPA'98 IDS evaluation dataset[7], and was introduced as a benchmark on International Knowledge Discovery and Data Mining Tools Competition[8] in 1999. DARPA'98 dataset contained around 4 gigabytes of compressed network traffic in TCP dump format captured during 7 weeks, whereas testing set was captured during another 2 weeks. The attacks performed during simulations fall into one of the following four classes:

- *Denial of Service* (dos): is an attempt to make machine or network resource unavailable to its intended users, e.g. syn flood.
- *Remote to Local* (r2l): attempt to gain access to host from remote machine

without having account on host, e.g. guessing password.

- *User to Root* (u2r): attempt to gain root privileges without having account on victim's machine, e.g. buffer overflow vulnerabilities.
- *Probing:* attempt to gain information about the victim's machine, e.g. port scanning.

Note, training dataset contains only 24 attack types, whereas testing dataset has additional 14 attack types belonging to the mentioned classes.

Furthermore, feature extraction has been realized using Bro IDS[8], resulting in 41 features for a connection. By connection we mean a sequence of packets exchanged between source IP and destination IP starting at TCP handshake until teardown. The features can be categorized to the following four groups:

- Basic features: these features can be extracted directly from the TCP headers, without inspection of the payload (e.g. type of protocol, number of data bytes)
- Content features: these features are based on the inspection of the payload. Since DoS and Probing attack are too noisy, it's possible to detect them without looking at payload. However, R2L and U2R attacks don't exhibit any noise and usually realized through single connection. These attacks can be detected by analyzing the payload of the packet (e.g. number of failed logins).
- Traffic features: these features are computed using two-second time window interval and are divided into two groups:
  i. Same host features: calculate the statistics of the connections to the same destination host as the current connection within the last two seconds.
  ii. Same service features: calculate the statistics of the connections with the same service as the current connection within the last two seconds.

More detailed information about features and labeling of connections can be found in Appendix 1.

### 3.1.2 Pros and Cons

Although, KDD '99 dataset have been extensively used by research community for a decade, extensive critique in McHugh *et al.*[10] does not recommend to use it as the solely benchmark for IDS nowadays. According to Tavallaee *et al.*[11] about 78% of the train set and 75% of test set are duplicated, causing proclivity of learning algorithms towards more frequent records. Thus, less frequent but more harmful u2r attacks could go undetected. Another research[12] confirms that 98% of the training data is composed of normal, neptune and smurf type of attack. Furthermore, these attacks are highly related to certain features that make their classification easier, resulting in the total high accuracy of learning algorithm. Yet another substantial problem is that this data was generated more than a decade ago, and on the era of hourly AV updates, the use of decade old dataset for research purposes is highly questionable. Therefore KDD '99 dataset better be used as a double check dataset for old types of attack alongside with new benchmark containing up-to-date traffic. A modification of KDD '99 dataset was done with the purpose of removing redundant records in the so called NSL-KDD dataset [13]. This dataset is recommended to use instead of KDD '99 due to smaller size and less biased results.

## 3.2 University of New Brunswick (UNB) dataset.

### 3.2.1 Description

This is another relatively new dataset that was generated by ISCX (Information Security Centre of Excellence) of UNB (2010). The underlying idea is based on the concept of profiles. Thus, the whole traffic is divided into malicious (α profile) and non-malicious (β profile) traffic that can each be profiled in different ways and then use all the profiles to generate traffic in network.

In order to generate α profile unambiguous description of the attack should be given. This research used exploit language known as ADeLe[14] for the description of the attacks. On the other hand, β profiles were generated using statistical approach. First of all, ISCX network topology was chosen as a sample to extract non-malicious traffic for β profiling. The statistics from Figure 1 was observed.

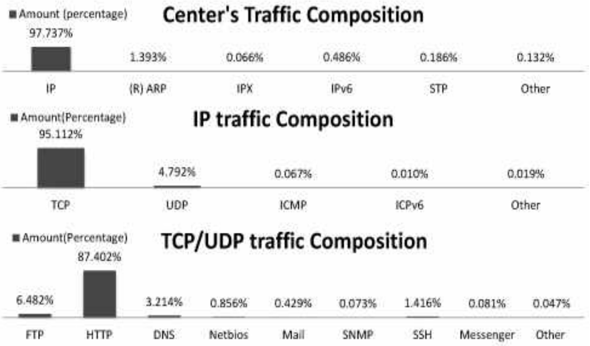


Fig.1 Composition of protocols and applications in non-malicious network

Further, the following protocols were chosen to be described in β profiles: HTTP, SMTP, POP3, IMAP, SSH, and FTP. Afterwards, each of the protocols was described using observed distribution in terms of number and time stamp of requests based on a weekday basis as in Figure 2.

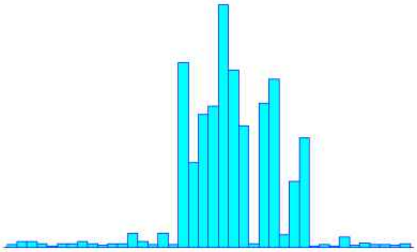


Fig. 2. Histogram of HTTP request made by an agent

Further, this distribution was described mathematically and used in β profile of HTTP traffic.

So, the intuition behind the approach is to enable researchers to generate datasets from profiles that can be combined to generate new traffic types. Thus, each profile can be related to specific feature and portably be used in different scenarios.

### 3.2.2 Attacks implemented

The malicious traffic generated was generated using scenario based attacks.

*Scenario 1: Infiltrating the network from the inside*

1. Querying the DNS for resource records using network administrative tools like *nslookup* and *dig*.
2. Exploit Adobe Reader *util.printf()* buffer overflow vulnerability using Metasploit and Meterpreter, by further establishing backdoor connection on port 5555.
3. Upload Nmap to exploited machine

through port 5555 using Meterpreter. Scan network for host with vulnerabilities.

4. Use SQL injection attack to the server since only port 80 is open.

*Scenario 2: HTTP Denial of Service*

Slowloris is used as the main tool for attack. Vulnerable SMB authentication protocol on port 445 is exploited.

*Scenario 3: Distributed Denial of Service using an IRC Botnet.*

An Internet Relay Chat bot was written from scratch and sent as an attachment for an update message for testbed users. Detailed execution of attack is given in the corresponding reference to paper.

*Scenario 4: Brute Force SSH.*

Brutessh is used to run dictionary brute force attack. The dictionary is composed of over 5000 alphanumerical entries. Account credentials were accessed in 30 min with successful login.

### 3.2.3 Dataset statistics

The dataset was generated during a week with the following settlement of attacks:

| Day | Date | Description | Size (GB) |
|---|---|---|---|
| Friday | 11/6/2010 | Normal Activity. No malicious activity | 16.1 |
| Saturday | 12/6/2010 | Normal Activity. No malicious activity | 4.22 |
| Sunday | 13/6/2010 | Infiltrating the network from inside + Normal Activity | 3.95 |
| Monday | 14/6/2010 | HTTP Denial of Service + Normal Activity | 6.85 |
| Tuesday | 15/6/2010 | Distributed Denial of Service using an IRC Botnet | 23.4 |
| Wednesday | 16/6/2010 | Normal Activity. No malicious activity | 17.6 |
| Thursday | 17/6/2010 | Brute Force SSH + Normal Activity | 12.3 |

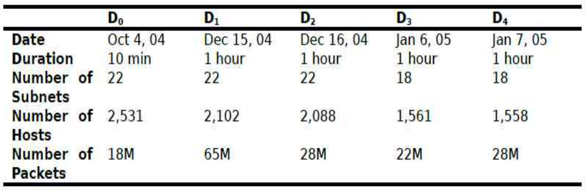
## 3.3 ISOT Dataset

### 3.3.1 Description and statistics

ISOT dataset was created by Information security and object technology (ISOT) research lab at the University of Victoria[15]. Basically, it's a mix of several existing open (malicious and non-malicious) datasets. The malicious traffic that was included in ISOT dataset comes from French chapter of honeynet project[16] and includes Storm and Waledac botnets. Storm botnet had its peak in 2007 -2008 with more than a million infected bots. Further, Waledac was considered as a successor of Storm with more distributed P2P style communication protocol. Unlike overnet used by Storm, Waledac utilizes HTTP communication and fast-flux DNS network.

Furthermore, non-malicious traffic was collected from two sources. One part of everyday usage traffic was obtained from the Traffic Lab at Ericsson Research in Hungary[17]. This traffic was further incorporated with second dataset from Lawrence Berkeley National Lab (LBNL) [18]. This combination is crucial since Ericsson Lab dataset includes general traffic from a variety of applications as well as HTTP web browsing, World of Warcraft traffic, and traffic from Azureus bittorent client. On the other hand, LNBL traffic is from a medium-sized enterprise network and consists of five datasets ($D_0$ to $D_4$),shown in Table 1.

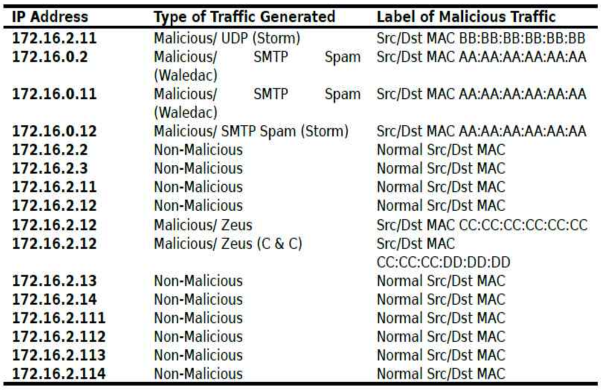
Table 1. LBNL dataset general information



| | $D_0$ | $D_1$ | $D_2$ | $D_3$ | $D_4$ |
|---|---|---|---|---|---|
| **Date** | Oct 4, 04 | Dec 15, 04 | Dec 16, 04 | Jan 6, 05 | Jan 7, 05 |
| **Duration** | 10 min | 1 hour | 1 hour | 1 hour | 1 hour |
| **Number of Subnets** | 22 | 22 | 22 | 18 | 18 |
| **Number of Hosts** | 2,531 | 2,102 | 2,088 | 1,561 | 1,558 |
| **Number of Packets** | 18M | 65M | 28M | 22M | 28M |

All these datasets were replayed using tcpreplay[19] and further can be labeled using Table 2. Whether the traffic is generated can be deducted simply from the source of the packet.

Table 2. Malicious hosts



| IP Address | Type of Traffic Generated | Label of Malicious Traffic |
|---|---|---|
| **172.16.2.11** | Malicious/ UDP (Storm) | Src/Dst MAC BB:BB:BB:BB:BB:BB |
| **172.16.0.2** | Malicious/ SMTP Spam (Waledac) | Src/Dst MAC AA:AA:AA:AA:AA:AA |
| **172.16.0.11** | Malicious/ SMTP Spam (Waledac) | Src/Dst MAC AA:AA:AA:AA:AA:AA |
| **172.16.0.12** | Malicious/ SMTP Spam (Storm) | Src/Dst MAC AA:AA:AA:AA:AA:AA |
| **172.16.2.2** | Non-Malicious | Normal Src/Dst MAC |
| **172.16.2.3** | Non-Malicious | Normal Src/Dst MAC |
| **172.16.2.11** | Non-Malicious | Normal Src/Dst MAC |
| **172.16.2.12** | Non-Malicious | Normal Src/Dst MAC |
| **172.16.2.12** | Malicious/ Zeus | Src/Dst MAC CC:CC:CC:CC:CC:CC |
| **172.16.2.12** | Malicious/ Zeus (C & C) | Src/Dst MAC CC:CC:CC:DD:DD:DD |
| **172.16.2.13** | Non-Malicious | Normal Src/Dst MAC |
| **172.16.2.14** | Non-Malicious | Normal Src/Dst MAC |
| **172.16.2.111** | Non-Malicious | Normal Src/Dst MAC |
| **172.16.2.112** | Non-Malicious | Normal Src/Dst MAC |
| **172.16.2.113** | Non-Malicious | Normal Src/Dst MAC |
| **172.16.2.114** | Non-Malicious | Normal Src/Dst MAC |

### 3.3.2 Pros and Cons

The problem of this dataset is no other malicious content except botnet traffic. Thus, it should be combined with another dataset in order to perform decently in the everyday networks. Furthermore, the Storm, and Waledac are P2P based botnets, however, C&C based botnets also exist. Therefore, the trained algorithm can behave biased toward detection of Storm and Waledac, and a possibility of non-detection of other botnets with different types of protocols still remains.

## IV. Discussion and conclusion

While dealing with all kinds of datasets, it's clear that no absolute ideal dataset exists. Every dataset has its pros and cons, and criteria such as labeling, realistic traffic behavior, or content of malicious activity are the ones that determine the quality of dataset. For example, dataset maybe realistic and reflecting up to date traffic, however, maybe non-labeled, decreasing quality of evaluation.

By examining a variety of currently existing datasets, we decided to focus our attention on three most useful, in our opinion. KDD ‘99 was the one in best condition in terms of fully available headers and payloads of the packets. However, the problem was its outdated patterns of attacks that do not reflect current situation. An alternative for KDD ’99 dataset was suggested in terms of NSL-KDD dataset, which has no redundant records. Further, UNB dataset was discussed. The traffic generation method used profiles. Statistical approaches have been utilized in order to profile non-malicious traffic. On the other hand, exploit description language was used in order to profile malicious traffic. This approach has the vision of being flexible with the profiles, giving power to anyone to generate similar traffic. Lastly, ISOT dataset was the topic of discussion. This dataset is generated for testing botnet detection techniques. Absence of another type of malicious traffic is one of the problems with this dataset.

Thus, according to our analysis in this paper, demand for the datasets is high in the research community. Privacy issues stop lots of sources to be openly published, making the topic of generation robust and realistic datasets a future topic of research.

## References

〔1〕 KDD '99 Cup Dataset, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html , Accessed: August 2013

〔2〕 A. Shiravi, H. Shiravi, M. Tavallaee, and A. A. Ghorbani "Towards Developing a Systematic Approach To Generate Benchmark Datasets for Intrusion Detection". Computers & Security 2011.

〔3〕 The Cooperative Association for Internet Data Analysis, http://www.caida.org/data/overview/ Accessed:July 2013

〔4〕 http://nexginrc.org/Datasets/DatasetDetail.aspx?pageID=24 Accessed: July 2013

〔5〕 Internet Traffic Archive http://www.sigcomm.org/ITA/ Accessed:July 2013

〔6〕 Defcon Datasets http://cctf.shmoo.com/ http://ddtek.biz/dc17.html Accessed: August 2013

〔7〕 Lincoln Laboratory http://www.ll.mit.edu/mission/communications/cyber/CSTcorpora/ideval/data/ Accessed: August 2013

〔8〕 KDD Task Description http://kdd.ics.uci.edu/databases/kddcup99/task.html Accessed:August 2013

〔9〕 V. Paxson, "Bro: A System for Detecting Network Intruder in Real-Time", Computer Networks, 31(23-24), pp. 2435-2463, 14 Dec. 1999

〔10〕 J. McHugh, "Testing Intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory", TISSEC 2000.

〔11〕 M.Tavallaee, E. Bagheri, and A.Ghorbani. "A Detailed Analysis of the KDD CUP 99 Data Set",CISDA 2009

〔12〕H.G. Kayacik, A. Nur Zincir-Heywood, and M.I. Heywood "Selecting features for intrusion detection: A feature relevance analysis on KDD 99 Intrusion Detection Datasets", WCECS 2010

〔13〕 NSL-KDD dataset

http://nsl.cs.unb.ca/NSL-KDD/ Accessed: August 2013

〔14〕 C. Michel and L. M´e, "ADeLe: an attack description language for knowledge-based intrustion detection", Proceedings of the 16th international conference on Information security: Trusted information, Kluwer Academic Publishers, Norwell, MA, USA. pp. 353 - 368.2001.

〔15〕 S. Saad, I. Traore, A. A. Ghorbani, Bassam Sayed, David Zhao, Wei Lu, John Felix, and Payman Hakimian, "Detecting P2P botnets through network behavior analysis and machine learning", Proceedings of 9th Annual Conference on Privacy, Security and Trust (PST2011), July 19-21, 2011, Montreal, Quebec, Canada http://www.uvic.ca/engineering/ece/isot/datasets/index.php#section0-0 Accessed: August 2013

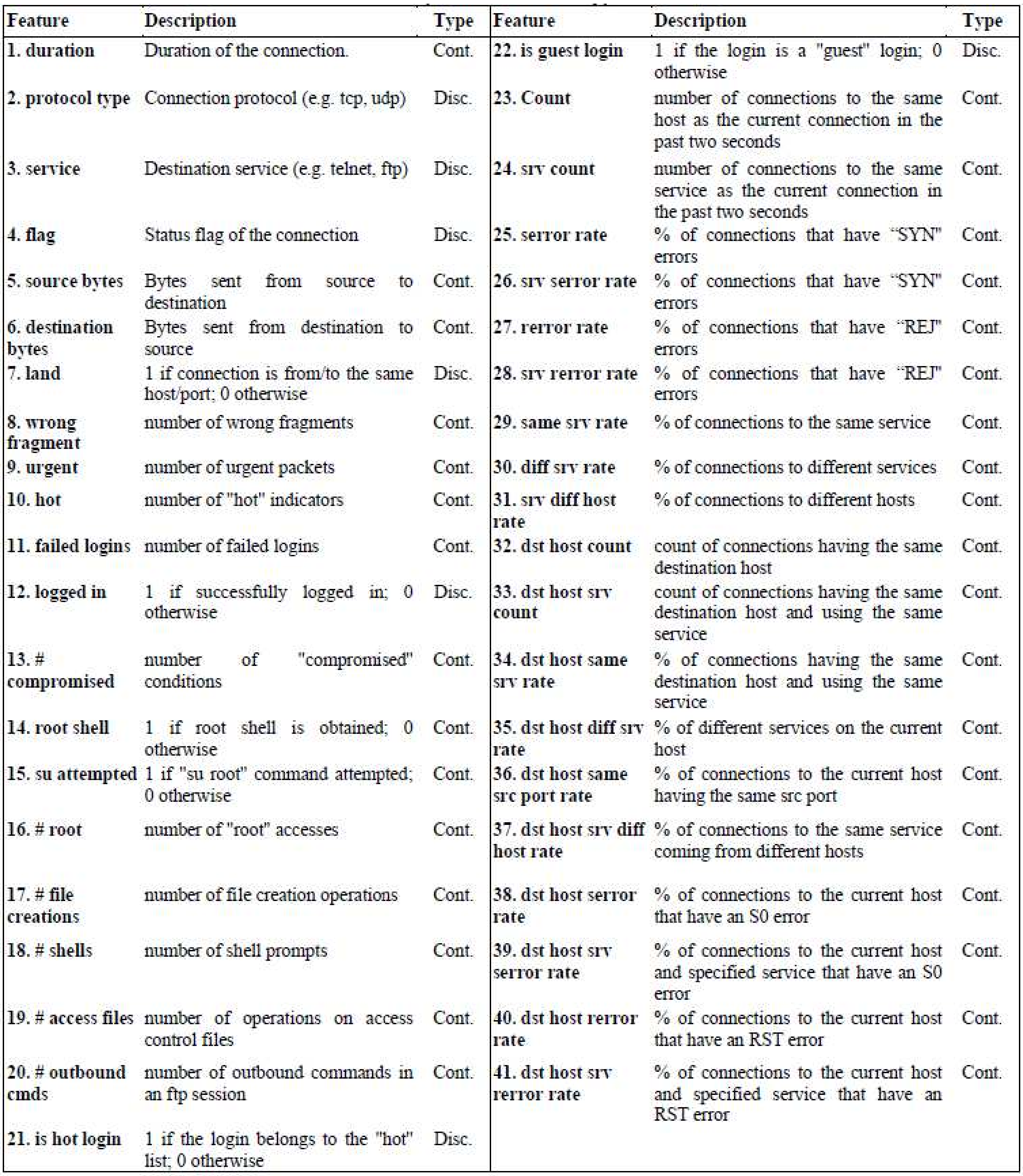
〔16〕 French Chapter of Honeynet: http://www.honeynet.org/chapters/france Accessed: August 2013

〔17〕 G. Szab´o, D. Orincsay, S. Malomsoky, and I. Szab´o, "On the validation of traffic classification algorithms," Proceedings of the 9th international conference on Passive and active network measurement, PAM'08, (Berlin, Heidelberg), pp. 72 - 81, Springer-Verlag, 2008.

〔18〕 LBNL Enterprise Trace Repository.
http://www.icir.org/enterprise-tracing
Accessed:July 2013

〔19〕 TCP Replay software
http://tcpreplay.synfin.net/
Accessed: July 2013

Appendix 1. Description of KDD 99 Intrusion Detection Dataset Features [1]



| Feature | Description | Type | Feature | Description | Type |
|---|---|---|---|---|---|
| **1. duration** | Duration of the connection. | Cont. | **22. is guest login** | 1 if the login is a "guest" login; 0 otherwise | Disc. |
| **2. protocol type** | Connection protocol (e.g. tcp, udp) | Disc. | **23. Count** | number of connections to the same host as the current connection in the past two seconds | Cont. |
| **3. service** | Destination service (e.g. telnet, ftp) | Disc. | **24. srv count** | number of connections to the same service as the current connection in the past two seconds | Cont. |
| **4. flag** | Status flag of the connection | Disc. | **25. serror rate** | % of connections that have "SYN" errors | Cont. |
| **5. source bytes** | Bytes sent from source to destination | Cont. | **26. srv serror rate** | % of connections that have "SYN" errors | Cont. |
| **6. destination bytes** | Bytes sent from destination to source | Cont. | **27. rerror rate** | % of connections that have "REJ" errors | Cont. |
| **7. land** | 1 if connection is from/to the same host/port; 0 otherwise | Disc. | **28. srv rerror rate** | % of connections that have "REJ" errors | Cont. |
| **8. wrong fragment** | number of wrong fragments | Cont. | **29. same srv rate** | % of connections to the same service | Cont. |
| **9. urgent** | number of urgent packets | Cont. | **30. diff srv rate** | % of connections to different services | Cont. |
| **10. hot** | number of "hot" indicators | Cont. | **31. srv diff host rate** | % of connections to different hosts | Cont. |
| **11. failed logins** | number of failed logins | Cont. | **32. dst host count** | count of connections having the same destination host | Cont. |
| **12. logged in** | 1 if successfully logged in; 0 otherwise | Disc. | **33. dst host srv count** | count of connections having the same destination host and using the same service | Cont. |
| **13. # compromised** | number of "compromised" conditions | Cont. | **34. dst host same srv rate** | % of connections having the same destination host and using the same service | Cont. |
| **14. root shell** | 1 if root shell is obtained; 0 otherwise | Cont. | **35. dst host diff srv rate** | % of different services on the current host | Cont. |
| **15. su attempted** | 1 if "su root" command attempted; 0 otherwise | Cont. | **36. dst host same src port rate** | % of connections to the current host having the same src port | Cont. |
| **16. # root** | number of "root" accesses | Cont. | **37. dst host srv diff host rate** | % of connections to the same service coming from different hosts | Cont. |
| **17. # file creations** | number of file creation operations | Cont. | **38. dst host serror rate** | % of connections to the current host that have an S0 error | Cont. |
| **18. # shells** | number of shell prompts | Cont. | **39. dst host srv serror rate** | % of connections to the current host and specified service that have an S0 error | Cont. |
| **19. # access files** | number of operations on access control files | Cont. | **40. dst host rerror rate** | % of connections to the current host that have an RST error | Cont. |
| **20. # outbound cmds** | number of outbound commands in an ftp session | Cont. | **41. dst host srv rerror rate** | % of connections to the current host and specified service that have an RST error | Cont. |
| **21. is hot login** | 1 if the login belongs to the "hot" list; 0 otherwise | Disc. | | | |