Technical Report CIDDS-001 data set

Markus Ring, Sarah Wunderlich and Dominik Grüdl

April 28, 2017

CIDDS-001 (Coburg Intrusion Detection Data Set) [2] is a labelled flow-based data set for evaluation of anomaly based intrusion detection systems. In this report, we provide an overview of the CIDDS-001 data set. We describe in detail the environment in which the data was captured as well as the labelling process of the data set. Further, we explain the structure and the additional published material of the data set. For the underlying ideas of this data set, we refer to our original publication *Flow-based benchmark data sets for intrusion detection* [2].

Table 1: Revision History

Version	**Description**	**Editors**
0.1	First Version of the technial report	Markus Ring, Sarah Wunderlich, Dominik Grüdl

1 Terms of Use

To facilitate reproducibility, further development and experiments, we make the CIDDS-001 data set as well as the generation scripts openly available to the community. If you publish material based on our CIDDS (Coburg Intrusion Detection Data Set) dataset or the generation scripts `https://github.com/markusring/CIDDS`, please cite our paper:

Ring, M., Wunderlich, S., Grüdl, D., Landes, D., Hotho, A.: "Flow-based benchmark data sets for intrusion detection.", In: Proceedings of the 16th European Conference on Cyber Warfare and Security (ECCWS), to appear. ACPI (2017)

Here is a BiBTeX citation as well:

```
@incollection{ring2017data,
  title={Flow-based benchmark data sets for intrusion detection},
  author={Ring, Markus and Wunderlich, Sarah and Gruedl, Dominik and
  Landes, Dieter and Hotho, Andreas},
```

```
  booktitle={Proceedings of the 16th European Conference on Cyber
  Warfare and Security (ECCWS), to appear},
  year={2017},
  publisher={ACPI}
}
```

2 What is the CIDDS-001 data set?

CIDDS-001 is a labelled flow-based data set for evaluation of anomaly-based network intrusion detection systems. For creation of the CIDDS-001 data set, a small business environment was emulated using OpenStack. This environment includes several clients and typical servers like an E-Mail server or a Web server. Python scripts emulate normal user behaviour on the clients.

The CIDDS-001 contains unidirectional *NetFlow* [1] data. Table 2 shows an overview of the attributes within the CIDDS-001 data set. The attributes 1 to 10 are default *NetFlow* attributes whereas the attributes 11 to 14 are added by us during the labelling process (see Section 5.1).

Table 2: Attributes within the CIDDS-001 data set. The second column provides the column names in the published files of the CIDDS-001 data set. The third column gives a short description of these attributes.

Nr.	Name	Description
1	Src IP	Source IP Address
2	Src Port	Source Port
3	Dest IP	Destination IP Address
4	Dest Port	Destination Port
5	Proto	Transport Protocol (e.g. ICMP, TCP, or UDP)
6	Date first seen	Start time flow first seen
7	Duration	Duration of the flow
8	Bytes	Number of transmitted bytes
9	Packets	Number of transmitted packets
10	Flags	*OR* concatenation of all TCP Flags
11	Class	Class label (normal, attacker, victim, suspicious or unknown)
12	AttackType	Type of Attack (portScan, dos, bruteForce, —)
13	AttackID	Unique attack id. All flows which belong to the same attack carry the same attack id.
14	AttackDescription	Provides additional information about the set attack parameters (e.g. the number of attempted password guesses for SSH-Brute-Force attacks)

3 Emulated Network Environment

3.1 Overview

Figure 1 provides an overview of the emulated small business environment in which the CIDDS-001 data set was captured.

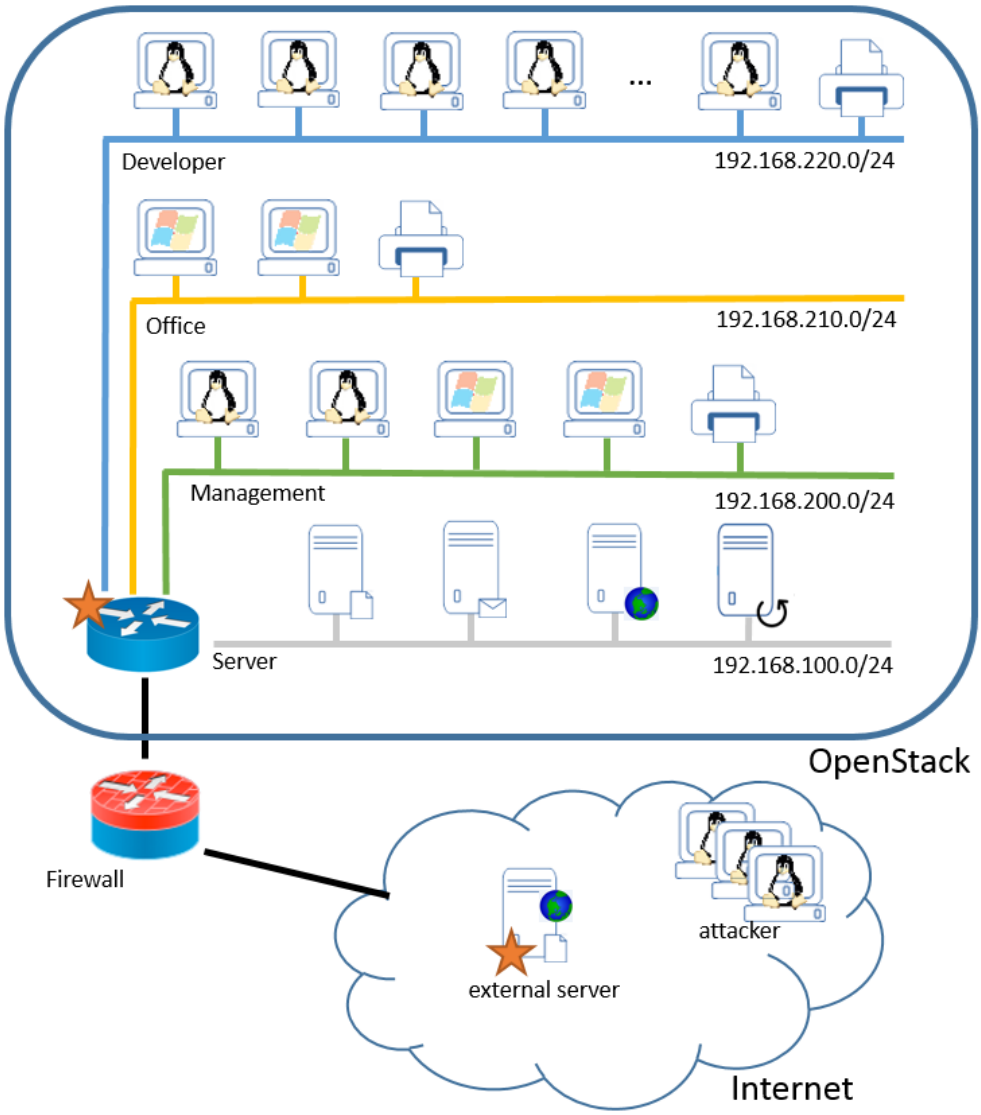

Figure 1: Overview of the small business environment [2]. The stars highlight the positions where the flow-based traffic was captured.

The unidirectional *NetFlow* traffic was captured at two different spots which are highlighted through stars in Figure 1. The emulated small business environment consists of four subnets. One subnet contains all internal servers (web, file, backup and mail).

The other three subnets represent client subnets. Each of them is assigned to a different department (*Developer*, *Office* and *Management*). Besides the *OpenStack* environment, an external server was deployed on the internet to capture real and up-to-date traffic from the internet. This server is called *external server* and it is directly accesssible from the internet. On this server, two different services (file-synchronization (Seafile) and web server) running for the internal clients.

3.2 Server subnet

The server subnet has the Subnet IP *192.168.100.0/24*. It contains four servers (Internal Web server, File server, Mail server and Backup server). The IP Addresses of the servers are shown in Figure 2.

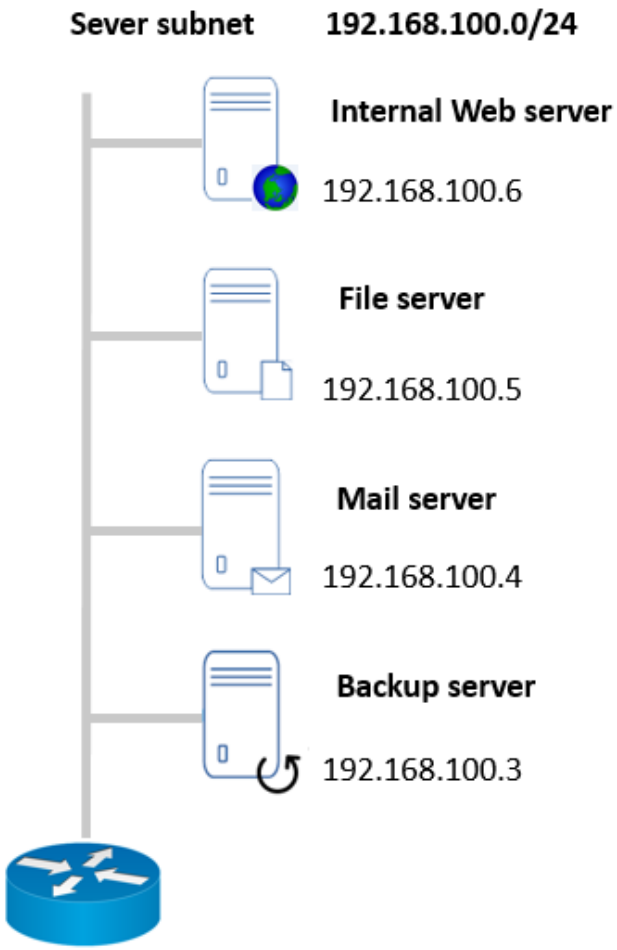

Figure 2: Overview of the *Server* subnet.

The clients occasionally communicate with the *Internal Web server* while surfing the

web. From time to time, the clients send and receive E-Mails. In these cases, clients communicate with the *Mail Server*. Further, the clients mount the shared folders from the *File server* as network drives. The backup server is only used by the other three servers. The *Internal Web server*, *Mail server* and *File server* create nightly backups and push them on the *Backup server*.

3.3 Management subnet

Figure 3 provides an overview of the *Management* subnet.

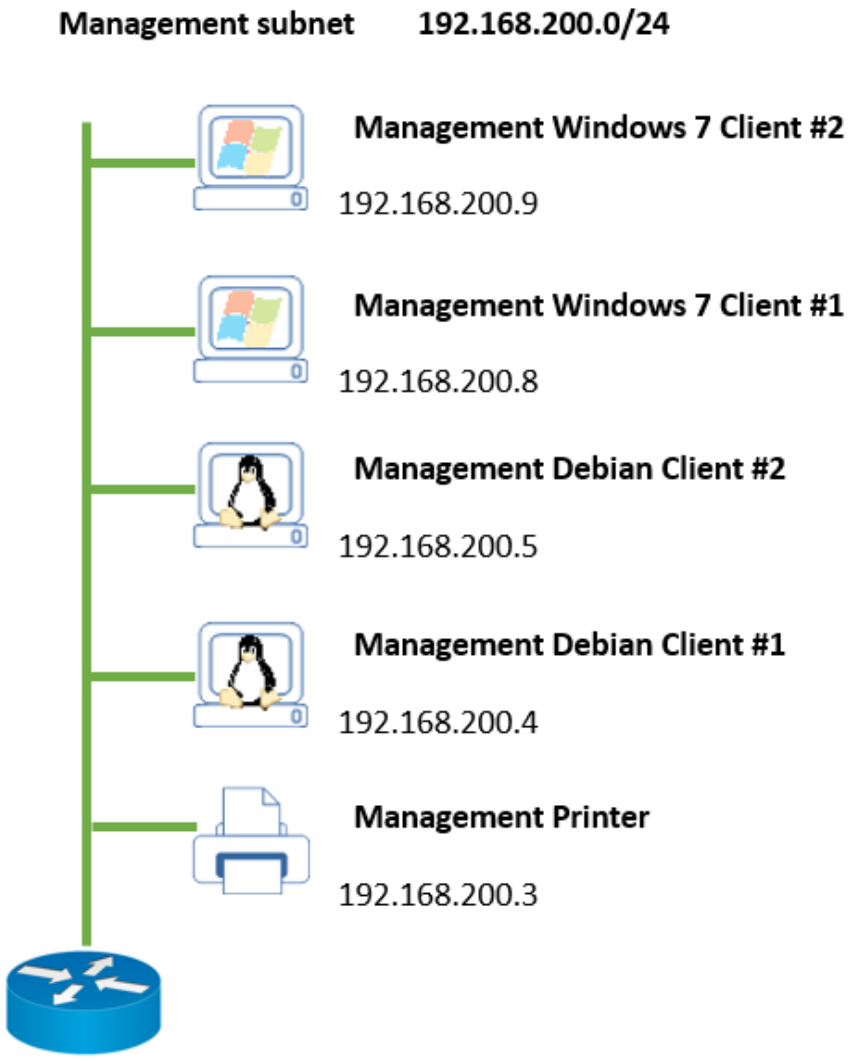

Figure 3: Overview of the *Management* subnet.

The *Management* subnet has the Subnet IP *192.168.200.0/24*. In our emulated environment, the management subnet has four clients and one network printer. All four

clients use the offered file synchronization service from the *external server*. The clients can be distinguished in two *Windows 7* and two *Debian 8* clients.

3.4 Office subnet

The office subnet has the Subnet IP *192.168.210.0/24*. It consists of two *Windows 7* clients and one network printer. In contrast to the clients of the other two subnets, the clients from this subnet do not use the offered file synchronization service from the external server. The IP Addresses for the devices are given in Figure 4.

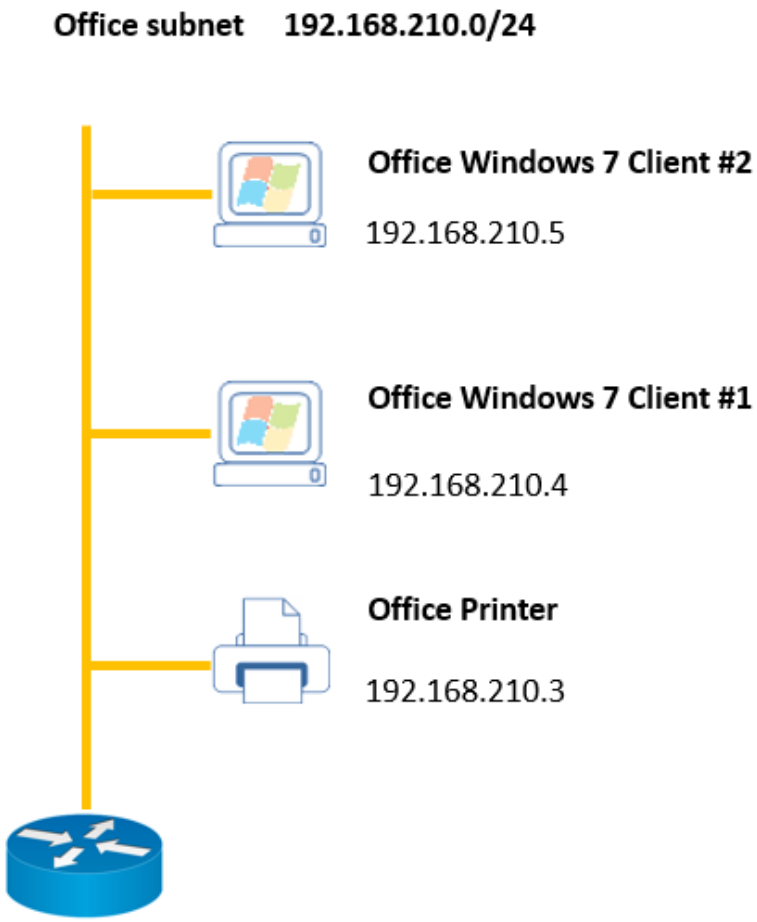

Figure 4: Overview of the *Office* subnet.

3.5 Developer subnet

The *Developer* subnet has the most clients within the emulated business environment. 13 *Debian* clients and one network printer belong to the subnet IP *192.168.220.0/24*.

The clients *Dev. Debian Client #12* and *Dev. Debian Client #13* executed several attacks during *week1* and *week2* within the CIDDS-001 data set. Further, the clients *Dev. Debian Client #2* and *Dev. Debian Client #6* are configured as system administrators. These are the only two clients which open *SSH* connections for administration of the internal servers. The IP Addresses for the devices are given in Figure 5.

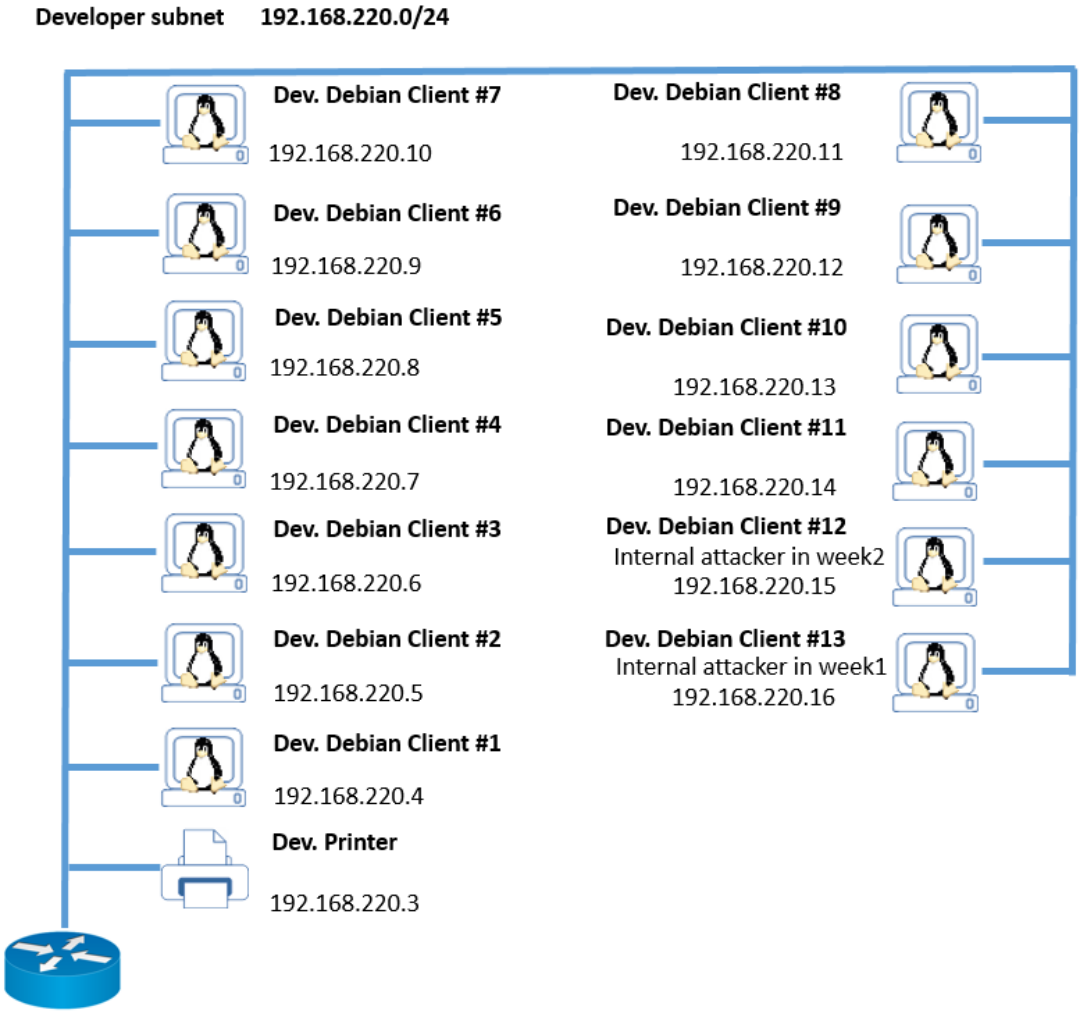

Figure 5: Overview of the *Developer* subnet.

3.6 External server

The external server is directly deployed on the internet (see Figure 6). It offers two services: a public reachable homepage for internal clients as well as for external interested people. Further, it provides a file synchronization service (Seafile) for the internal clients. This service is used by all clients from the *Developer* and *Management* subnet. It should

be mentioned, that all internal clients from the OpenStack environment communicate with the same public IP Address to the external server. Besides the external server, we had control over three other servers on the internet. From these servers, we executed several attacks on the *external server*.

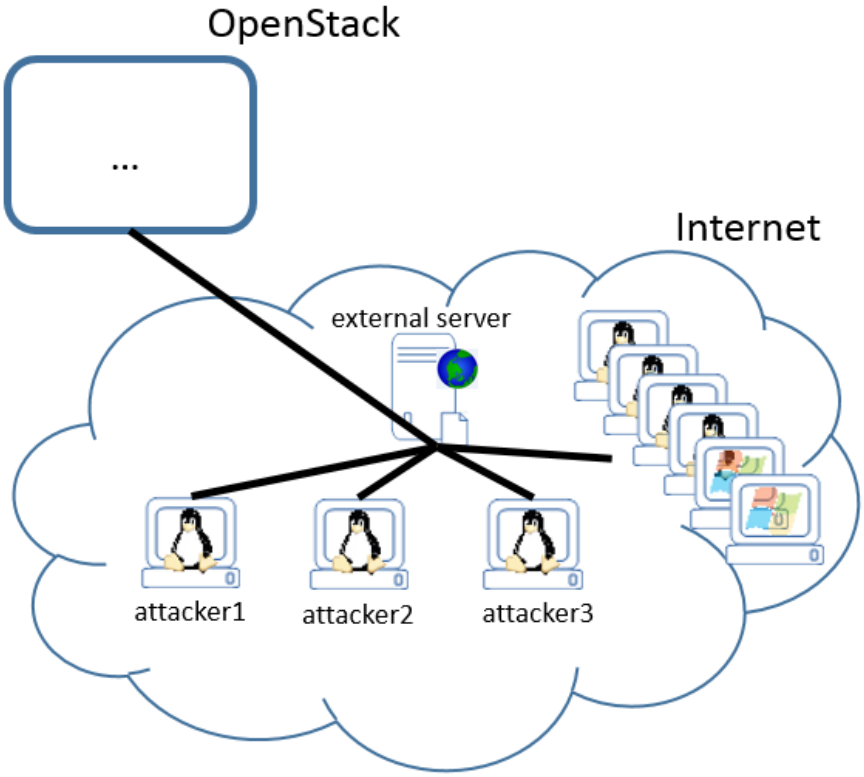

Figure 6: External server.

4 Published data of CIDDS-001 data set

This section describes the published material of the CIDDS-001 data set which can be downloaded at `https://www.hs-coburg.de/cidds/`. The archive *CIDDS-001.zip* includes the following four folders: *attack_logs*, *clients_confs*, *client_logs* and *traffic*.

4.1 attack_logs

The folder *attack_logs* contains two *CSV* files which provide information about the executed attacks. Executed attacks within the OpenStack environment are stored in the file *attack_logs_intern.csv*. The file *attack_logs_extern.csv* includes information about the attacks against the *external server*. Both files contain information like the *Source IP*

Addresses, the *start* and *end time* of the attacks as well as a short description of the attacks.

These files are used for the labelling process in Section 5.1.

4.2 client_confs

As already mentioned above, normal user behaviour of the clients is executed through configurable Python scripts. The *client_confs* folder contains the used configuration files for each client. The file names of the configuration files are constructed as follows:

IP–ADDRESS.conf

The *IP Address* within the file name allows us to map each configuration file to the corresponding client. E.g. the configuration for client *Dev. Debian Client #7 in Figure 5* is *192.168.220.10.conf*.

The configuration file controls – among other things – the different user behaviour of client as well as the working hours of the clients. The user behaviour is defined through the probabilities of the clients activities.

4.3 client_logs

The *client_logs* folder contains for each client a *CSV* file. The name structure of the log files is similar to the structure of the configuration file names:

IP–ADDRESS.log

Remember, the user behaviour of the clients is controlled by Python scripts. These Python scripts record their activities and store them in these log files. This allows users of the *CIDDS-001* data set to understand which client activities caused network traffic.

4.4 traffic

The *traffic* folder contains two sub-folders *ExternalServer* and *OpenStack*. These sub-folders contain several *CSV* files with the captured flow-based network traffic in unidirectional *NetFlow* format.

The file names in these sub-folders are constructed as follows:

CIDDS–version–origin–period.csv

All files start with *CIDDS-001*. Traffic recorded in the OpenStack environment is marked as *internal* origin. Traffic recorded on the external server is marked as *external* origin. The last part (*period*) provides information when the network traffic was recorded (*week1*, *week2*, *week3* and *week4*).

5 Labelling and Anonymization

We post-processed the recorded data by labelling the flows and by anonymizing public IP Addresses. At first, we describe our labelling process in Section 5.1. Following, the anonymization of all public IP Addresses is described in Section 5.2.

5.1 Labelling

We add four label attributes (*class, attackID, attackType* and *attackDescription*) to each flow during the labelling process. The first label attribute, called *class*, has five different emphasis: *normal*, *attacker*, *victim*, *suspicious* and *unknown*. Each flow is assigned to a predefined class.

The other three label attributes provide additional information about executed attacks. These attributes are only used if the flow belongs to the class *attacker* or *victim*. If the flow belongs to the class *normal*, *suspicious* or *unknown*, the value of the additional label attributes are set to a default value ("- - -"). The second label attribute is called *attackID*. In the CIDDS-001 data set, a unique ID is assigned to each executed attack. Consequently, all flows which belong to the same attack share the same value in this attribute.

The third label attribute is called *attackType* and gives more information about the executed attack type. Possible values are: *pingScan*, *portScan*, *bruteForce* or *dos*.

The fourth label attribute contains detailed information about the executed attacks. For example, this label attributes contains the parameters settings for *portScans* or the number of passwords guessed for *bruteForce* attacks.

5.1.1 Traffic within OpenStack environment

We have full control over the virtual machines and virtual networks within the *OpenStack* environment. This allows us to label all flows with their corresponding classes.

Since we know the exact timestamp, origin and target of executed attacks, we are able to label all flows which are caused by attacks. All remaining flows within the *OpenStack* environment are labelled as *normal*.

5.1.2 Traffic at external Server

The labelling of the traffic at the external server is more complicated. Therefore, we use a multi-stage labelling process for the flows captured at this server.

It should be mentioned that we didn't attack the external server from the OpenStack environment. Therefore, we label all flows of the external server which have their *origin* or *target* in the OpenStack environment as *normal*. Further, we have control over three servers that are directly deployed on the internet (see attacker1, attacker2 and attacker3 in Figure 6). These servers only exploit attacks to the *external server*. Since we know the origins, target and timestamps of the exploited attacks from these servers, we are able to label all corresponding flows with the additional class labels *attacker* or *victim*. The external server provides a homepage for interested people. Therefore, all traffic to the ports 80 and 443 could be normal traffic or intrusion attempts. As a consequence, traffic to the ports 80 and 443 on the external server is labelled as *unknown*.

The remaining network traffic is labelled as *suspicious*, since no further services are offered for public users.

5.2 Anonymization

We anonymized all public IP Addresses for privacy reasons. The IP Addresses of the internal OpenStack clients and servers are not affected during the anonymization process.

5.2.1 Special Treatments

The following IP Addresses are handled specifically during the anonymization process:

- All servers and clients from the OpenStack environment communicate with the same public IP Address to the external server. We replaced this public IP Address with *OPENSTACK_NET*.
- All virtual machines within the OpenStack environment use the same DNS server. We rename the IP Address of the DNS server as *DNS*.
- The IP Address of the external server is replaced with *EXT_SERVER*.
- The IP Addresses of the external attackers are replaced with *ATTACKER1*, *ATTACKER2* and *ATTACKER3*.

5.2.2 Other IP Addresses

We used the following anonymization process for the remaining public IP Addresses. The first three bytes of each IP Address are replaced with a randomly generated number. The fourth byte of the IP Address is kept. This allows us to retain information about network structures, since all IP Addresses from the same subnet are replaced with the same randomly generated number. Table 3 shows a few examples for this anonymization process.

Table 3: Anonymization process of public IP Addresses.

#	IP Address	Anonymized IP Address
1	8.8.8.8	4711_8
2	8.8.8.9	4711_9
3	8.8.8.18	4711_18
4	9.9.9.9	13107_9
5	9.9.9.173	13107_173
6	8.8.8.9	4711_9
7	7.7.7.7	2311_7

6 Traffic Characteristics

The CIDDS-001 data set was captured over a period of four weeks and contains nearly 32 millions flows. Thereof, about 31 millions flows were captured within the OpenStack environment. About 0.7 million flows were captured at the external server.

The CIDDS-001 data set includes 92 attacks. 70 attacks were executed within the OpenStack environment and 22 attacks targeted the external server. Table 4 provides more information about the executed attacks within CIDDS-001 data set.

Table 4: Overview of the number of executed attacks within the CIDDS-001 data set. Each row represents the attacks within a specific week. The columns describe the different types of attacks.

	OpenStack				External Server			
	PortScan	PingScan	DoS	BruteForce	PortScan	PingScan	DoS	BruteForce
week1	16	10	11	5	0	0	0	0
week2	8	6	7	7	2	0	0	4
week3	0	0	0	0	5	0	0	7
week4	0	0	0	0	1	0	0	3

As can be seen in Table 4, week one and week two contain traffic with benign behaviour as well as attacks. Weeks three and four contain traffic with solely benign behaviour in the OpenStack environment and mixed behaviour on the external server.

There are two peculiarities to be noted when analyzing the CIDDS-001 data set. The first peculiaritiy is one hour missing traffic between 01:00 AM and 03:00 AM on March 26th due to daylight-saving time. The second peculiarity is an OpenStack wide network failure on March 22th between the hours of 11:00 AM and 02:00 PM. Consequently, there is traffic missing during that time.

7 Further Information

For further information, we encourge you to read our paper *Flow-based benchmark data sets for Intrusion Detection* [2]. In addition to that, we also published our generation and labelling scripts in a github repository [3].

Acknowledgements

This work is funded by the Bavarian Ministry for Economic affairs through the WISENT project (grant no. IUK 452/002).

References

[1] Claise, B.: Cisco systems netflow services export version 9. RFC 3954 (2004)

[2] Ring, M., Wunderlich, S., Grüdl, D., Landes, D., Hotho, A.: Flow-based benchmark data sets for intrusion detection. In: Proceedings of the 16th European Conference on Cyber Warfare and Security (ECCWS), to appear. ACPI (2017)

[3] Ring, M., Wunderlich, S., Grüdl, D., Landes, D., Hotho, A.: Generation scripts for the coburg intrusio detection data sets (cidds) (Apr 2017), https://github.com/markusring/CIDDS