CASE STUDY - NETWORKING
NETWORK INTRUSION DETECTION

BUSINESS CONTEXT:

With the enormous growth of computer networks usage and the huge increase in the number of applications running on top of it, network security is becoming increasingly more important. All the computer systems suffer from security vulnerabilities which are both technically difficult and economically costly to be solved by the manufacturers. Therefore, the role of Intrusion Detection Systems (IDSs), as special-purpose devices to detect anomalies and attacks in the network, is becoming more important.

The research in the intrusion detection field has been mostly focused on anomaly-based and misusebased detection techniques for a long time. While misuse-based detection is generally favored in commercial products due to its predictability and high accuracy, in academic research anomaly detection is typically conceived as a more powerful method due to its theoretical potential for addressing novel attacks.

Conducting a thorough analysis of the recent research trend in anomaly detection, one will encounter several machine learning methods reported to have a very high detection rate of 98% while keeping the false alarm rate at 1%. However, when we look at the state of the art IDS solutions and commercial tools, there is no evidence of using anomaly detection approaches, and practitioners still think that it is an immature technology. To find the reason of this contrast, lots of research was done done in anomaly detection and considered various aspects such as learning and detection approaches, training data sets, testing data sets, and evaluation methods.

BUSINESS PROBLEM:

Your task to build network intrusion detection system to detect anamolies and attacks in the network. There are two problems.

1. Binomial Classification: Activity is normal or attack
2. Multinomial classification: Activity is normal or DOS or PROBE or R2L or U2R

Please note that, currently the dependent variable (target variable) is not definied explicitly. However, you can use attack variable to define the target variable as required.

DATA AVAILABILITY:

This data is KDDCUP'99 data set, which is widely used as one of the few publicly available data sets for network-based anomaly detection systems.

For more about data: http://www.unb.ca/cic/datasets/nsl.html

LIST OF COLUMNS FOR THE DATA SET:

["duration","protocol_type","service","flag","src_bytes","dst_bytes","land",
"wrong_fragment","urgent","hot","num_failed_logins","logged_in",
"num_compromised","root_shell","su_attempted","num_root","num_file_creations",
"num_shells","num_access_files","num_outbound_cmds","is_host_login",
"is_guest_login","count","srv_count","serror_rate", "srv_serror_rate",
"rerror_rate","srv_rerror_rate","same_srv_rate", "diff_srv_rate",

"srv_diff_host_rate","dst_host_count","dst_host_srv_count","dst_host_same_srv_rate",
"dst_host_diff_srv_rate","dst_host_same_src_port_rate",
"dst_host_srv_diff_host_rate","dst_host_serror_rate","dst_host_srv_serror_rate",
"dst_host_rerror_rate","dst_host_srv_rerror_rate","attack", "last_flag"]

BASIC FEATURES OF EACH NETWORK CONNECTION VECTOR

1 Duration: Length of time duration of the connection
2 Protocol_type: Protocol used in the connection
3 Service: Destination network service used
4 Flag: Status of the connection – Normal or Error
5 Src_bytes: Number of data bytes transferred from source to destination in single connection
6 Dst_bytes: Number of data bytes transferred from destination to source in single connection
7 Land: if source and destination IP addresses and port numbers are equal then, this variable takes value 1 else 0
8 Wrong_fragment: Total number of wrong fragments in this connection
9 Urgent: Number of urgent packets in this connection. Urgent packets are packets with the urgent bit activated

CONTENT RELATED FEATURES OF EACH NETWORK CONNECTION VECTOR

10 Hot: Number of „hot" indicators in the content such as: entering a system directory, creating programs and executing programs
11 Num_failed _logins: Count of failed login attempts
12 Logged_in Login Status: 1 if successfully logged in; 0 otherwise
13 Num_compromised: Number of ``compromised'' conditions
14 Root_shell: 1 if root shell is obtained; 0 otherwise
15 Su_attempted: 1 if ``su root'' command attempted or used; 0 otherwise
16 Num_root: Number of ``root'' accesses or number of operations performed as a root in the connection
17 Num_file_creations: Number of file creation operations in the connection
18 Num_shells: Number of shell prompts
19 Num_access_files: Number of operations on access control files
20 Num_outbound_cmds: Number of outbound commands in an ftp session
21 Is_hot_login: 1 if the login belongs to the ``hot'' list i.e., root or admin; else 0
22 Is_guest_login: 1 if the login is a ``guest'' login; 0 otherwise

TIME RELATED TRAFFIC FEATURES OF EACH NETWORK CONNECTION VECTOR

23 Count: Number of connections to the same destination host as the current connection in the past two seconds
24 Srv_count: Number of connections to the same service (port number) as the current connection in the past two seconds
25 Serror_rate: The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in count (23)
26 Srv_serror_rate: The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in srv_count (24)
27 Rerror_rate: The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in count (23)
28 Srv_rerror_rate: The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in srv_count (24)
29 Same_srv_rate: The percentage of connections that were to the same service, among the connections aggregated in count (23)
30 Diff_srv_rate: The percentage of connections that were to different services, among the connections aggregated in count (23)

31 Srv_diff_host_ rate: The percentage of connections that were to different destination machines among the connections aggregated in srv_count (24)

HOST BASED TRAFFIC FEATURES IN A NETWORK CONNECTION VECTOR

32 Dst_host_count: Number of connections having the same destination host IP address
33 Dst_host_srv_ count: Number of connections having the same port number
34 Dst_host_same _srv_rate: The percentage of connections that were to the same service, among the connections aggregated in dst_host_count (32)
35 Dst_host_diff_ srv_rate: The percentage of connections that were to different services, among the connections aggregated in dst_host_count (32)
36 Dst_host_same _src_port_rate: The percentage of connections that were to the same source port, among the connections aggregated in dst_host_srv_count (33)
37 Dst_host_srv_ diff_host_rate: The percentage of connections that were to different destination machines, among the connections aggregated in dst_host_srv_count (33)
38 Dst_host_serro r_rate: The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in dst_host_count (32)
39 Dst_host_srv_s error_rate: The percent of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in dst_host_srv_count (33)
40 Dst_host_rerro r_rate: The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in dst_host_count (32)
41 Dst_host_srv_r error_rate: The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in dst_host_srv_count (33)

<u>Type Features:</u>
Nominal: Protocol_type(2), Service(3), Flag(4)

Binary: Land(7), logged_in(12), root_shell(14), su_attempted(15), is_host_login(21),, is_guest_login(22)

Numeric: Duration(1), src_bytes(5), dst_bytes(6), wrong_fragment(8), urgent(9), hot(10), num_failed_logins(11), num_compromised(13), num_root(16), num_file_creations(17), num_shells(18), num_access_files(19), num_outbound_cmds(20), count(23), srv_count(24), error_rate(25), srv_serror_rate(26), rerror_rate(27),srv_rerror_rate(28), same_srv_rate(29),diff_srv_rate(30), srv_diff_host_rate(31), dst_host_count(32), dst_host_srv_count(33), dst_host_same_srv_rate(34), dst_host_diff_srv_rate(35), dst_host_same_src_port_rate(36), dst_host_srv_diff_host_rate(37), dst_host_serror_rate(38), dst_host_srv_serror_rate(39), dst_host_rerror_rate(40), dst_host_srv_rerror_rate(41)

Attack Class	Attack Type
DoS	Back, Land, Neptune, Pod, Smurf,Teardrop,Apache2, Udpstorm, Processtable, Worm (10)
Probe	Satan, Ipsweep, Nmap, Portsweep, Mscan, Saint (6)
R2L	Guess_Password, Ftp_write, Imap, Phf, Multihop, Warezmaster, Warezclient, Spy, Xlock, Xsnoop, Snmpguess, Snmpgetattack, Httptunnel, Sendmail, Named (16)
U2R	Buffer_overflow, Loadmodule, Rootkit, Perl, Sqlattack, Xterm, Ps (7)

ATTACK CLASS:

1. DOS: Denial of service is an attack category, which depletes the victim‟s resources thereby making it unable to handle legitimate requests – e.g. syn flooding. Relevant features: “source bytes” and “percentage of packets with errors”

2. Probing: Surveillance and other probing attack‟s objective is to gain information about the remote victim e.g. port scanning. Relevant features: “duration of connection” and “source bytes”

3. U2R: unauthorized access to local super user (root) privileges is an attack type, by which an attacker uses a normal account to login into a victim system and tries to gain root/administrator privileges by exploiting some vulnerability in the victim e.g. buffer overflow attacks. Relevant features: “number of file creations” and “number of shell prompts invoked,”

4. R2L: unauthorized access from a remote machine, the attacker intrudes into a remote machine and gains local access of the victim machine. E.g. password guessing Relevant features: Network level features – “duration of connection” and “service requested” and host level features - “number of failed login attempts”