

VoIPTG - v2.1.1

VoIP TRAFFIC GENERATOR

Contents

1	VoIPTG-2.1.1 - Overview and User Manual	1
1.1	VoIPTG-v2.1.1 - VoIP Traffic Generator	1
1.2	Example	2
1.3	Installing VoIPTG-2.1.1	4
1.4	Configuration parameters for VoIPTG-v2.1.1	4
1.4.1	Parameter for Configuring Output File name and location	7
1.4.2	Parameter for Configuring User Sets	8
1.4.3	Parameter for Generating Users' Address	10
1.4.4	Parameter for configuring Traffic Variance and Intensity	11
1.4.5	Parameter for Interarrival Time Distribution	14
1.4.6	Parameter for Call Duration Distribution	14
1.4.7	Parameters of call distribution over the caller	17
1.4.8	Parameters of call distribution over the Receiver	21

Chapter 1

VoIPTG-2.1.1 - Overview and User Manual

1.1 VoIPTG-v2.1.1 - VoIP Traffic Generator

The objectives of releasing this version is to generate a combination of good and bad traffic instead of only good or only bad. This becomes possible by defining multiple caller sets with different call generation patterns. In VoIPTG-v2.1.1 application calls are generated by multiple Caller sets with different call pattern where the calls are directed to a universal Receiver set. This idea replicates the real world traces which may be generate by various callers such as home users, commercial users, spammers .etc.

The central idea of this version can be summarized as follows:

- Multiple caller sets and call generation patterns can be defined through multiple configuration files.
- Generated traffic by different caller sets can be categorized as good or bad. Generated calls from various caller sets are directed to one good receiver set.
- In the traffic generation phase VoIPTG-v2.1.1 application instantiates new traffic generation phases with the defined configuration files to generate traces.
- Finally the generated traces are combined together in a file and thus we get the merged file which is a combination of good and bad traffic.

1.2 Example

In this scenario, we consider the user domain of a telecom provider “A”. Its user domain can be divided into two overlapping sets defined as “Caller” and “Receiver” found in figure 1.1. Calls generated from users from “Caller” set are redirected to

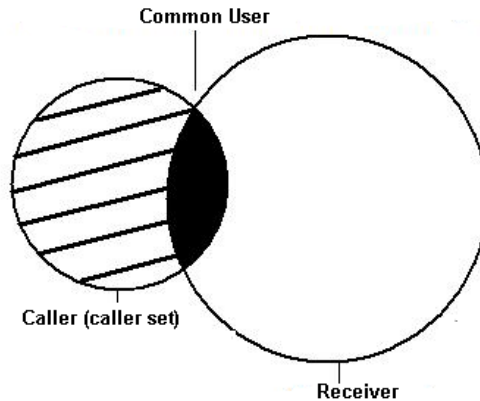


Figure 1.1: Good user domain of a telecom provider.

“Receiver” set and these calls can be defined as good calls. Again, there may be some attackers who generate spam calls (SPIT) or flooded the server with calls (Call Flooding -DoS attack). These callers can be gathered into two sets defined as “SPIT Attacker” and “DoS Attacker”. Both the SPIT call generators and the Dos attackers are aimed to attack the “Receiver” group which is the good users of telecom provider “A”. So calls from the “SPIT Attacker” and “DoS Attacker” are also directed to “Receiver” set figure 1.2 . We try to capture this scenario of multiple callers set with universal receiver set through VoIPTG-v2.1.1. Here we figure out three different caller sets (“Caller”, “SPIT Attacker” and “DoS Attacker”) and only one “Receiver” set. To model the above scenario, we need three configuration files (may be named as “input.properties”, “input1.properties” and “input2.properties” and they should be stored under the folder named “/input” in the location where the .jar file stored) to define the characteristics of users in two caller sets. As the calls from users of “Caller” to “Receiver” are good calls so we should define “G=t” in the configuration file (“input.properties”). The other parameters of this input file should be configured according to the property of good (home, business) callers. Again, the calls from user of “SPIT Attacker” and

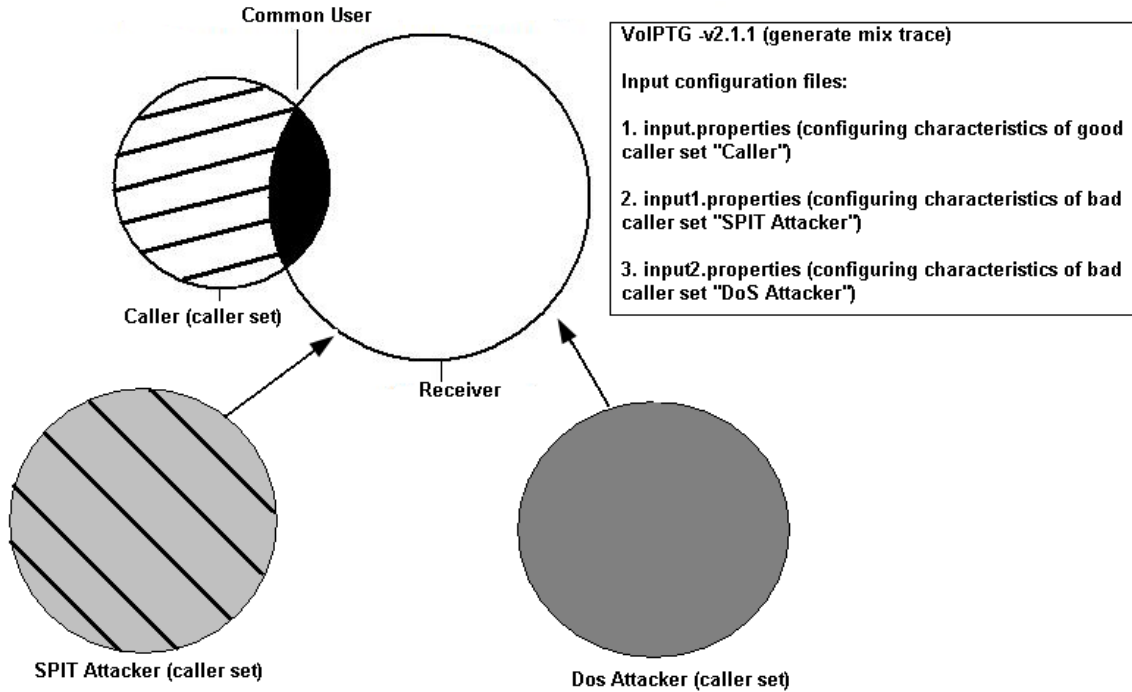


Figure 1.2: User domain of a telecom provider.

“DoS Attacker” to “Receiver” are supposed to be SPIT calls so the configuration file (“input1.properties” and “input2.properties”) should be include “G=f”. The other parameters of this input file should be configured according to the property of SPIT callers and Dos attackers. When the generator runs, it takes both the input files to create the user domain. It creates the “Receiver” set only for the input file (“input.properties”) which indicates good callers (“G=t”). For the other input files it only generates the caller set (that’s why we should have only one input file indicating “G=t” to generate the “Receiver” set).

In the trace generation phase, some parameters (e.g., observation time, starting time of observation, end time of observation) are found fixed for the whole traffic generation phase in spite of multiple caller sets and call patterns. These parameters are stored in another configuration file named ‘general-input.properties’.

1.3 Installing VoIPTG-2.1.1

Users can download VoIPTG-2.1.1 application from [here](#). Installation procedure of the application and the generation step of traffic is found [here](#).

1.4 Configuration parameters for VoIPTG-v2.1.1

After successful installation of the application, user can generate traffic. This version allows traffic generation with multiple configuration files indicating different set of callers while calls are directed to a single receiver set. Here some parameters should be remain same for the whole traffic generation process. For example, the traffic generation window should be same for all the callers to generate call. So the definition for the receiver set and the traffic generation window are considered as common parameters. Considering this VoIPTG-v2.1.1 maintains two categories of configuration/input files for traffic generation.

Configuration parameters for VoIPTG-v2.1.1 are stored in a file with ‘.properties’ extension which is used for files mainly used in Java related technologies to store the configurable parameters of an application. The general structure of the configuration parameter is as follows:

1. The parameters are defined in following format:

<PARAMETER NAME> = <VALUE>

2. The comments are begin with # character.

Description and guidelines for the configuration parameters are given in the following subsections.

1. This configuration file contains parameters which are common for the whole traffic generation phase in spite of multiple caller sets and call patterns. A sample configuration file containing these common parameters named ‘general-input.properties’ are found in the installation folder. So user should prepare an input file like ‘general-input.properties’ containing the following common configuration parameters :

Parameter Name : OUTPUT_FORMAT

- Description - This parameter allows to format the call record as desired with the necessary information such as the start time, end time, duration of call, caller and the receiver. Present VoIPTG implementation supports following two output formats.

UNITN- This format separates fields of call information by comma character (,). Start and end times of call are expressed in milliseconds while call duration in seconds. It arranges the generated call information in following fashion:

Caller, Receiver, Call Start Time, Call End Time, Call Duration, Response Code(200), Response String (OK).

NEC- This format separates fields of call information by comma character (,). Start, end times and call durations are expressed in seconds. It arranges the call information in following format:

Caller, Receiver, Call Start Time, Call End Time, Call Duration, Response Code(200), Response String (OK).

- Data Type - Integer
- Range - 1 : UNITN FORMAT, 2 : NEC FORMAT
- Default Value - 1
- Example - OUTPUT_FORMAT=1. This indicates that the generated calls are stored in the output file following UNITN output format.

Parameter Name : NUMBER_RECEIVER

- Description - Number of users in the Receiver set. This is a common parameter as for all configuration files, as VoIPTG-v2.1.1 maintains single receiver set.
- Data Type - Integer
- Range - $0 < \text{NUMBER_RECEIVER} > 1.0^7$. The input for parameter 'NUMBER_RECEIVER' should be correlated with the call generation rate of the Traffic Generator. For example with a call generation rate of 10 call/sec, Receiver set containing only 100 receivers may be unrealistic. This is because it indicates on average 8640 calls will be received by a receiver which is very high rate.
- Default Value - 1000

- Example - NUMBER_RECEIVER=1000 indicates 1000 users in the Receiver set.

Parameter Name : CALLERS_RECEIVERS_SET_OVERLAPPING_PERCENTAGE

- Description - Ratio of the overlapping users from Caller set to receiver set.
- Data Type - Double
- Range - Min : 0 - Max : 1
- Default Value - 0.2
- Example - If NUMBER_CALLERS=1000, NUMBER_RECEIVER=2500 and CALLERS_RECEIVERS_SET_OVERLAPPING_PERCENTAGE=0.3, then from the overlapping percentage we can evaluate that 300 users are common in Caller and Receiver set. This number of common users is calculated over the Caller Set. The graphical representation of this example is found in figure 1.3.

Parameter Name : OBSERVATION_TIME

- Description - Time window / Duration (in week) of Traffic generation.
- Data Type - Double
- Range - It should be non-negative and greater than zero. OBSERVATION_TIME > 0
- Default Value - 1
- Measure Unit - Week.
- Example -] OBSERVATION_TIME = 2 indicates traffic generation will continue for 2 weeks.

Parameter Name : START_OBSERVATION_DAY

- Description - Starting day of traffic generation.
- Data Type - Integer
- Range - Min : 0 - Max : 6 , This parameter can hold a number from 0 to 6 indicating 7 days in a week. Here, 0 - Monday, 1-Tuesday, 2- Wednesday, 3 - Thursday, 4-Friday, 5- Saturday and 6- Sunday.
- Default Value - 0 (Monday)

- Example - START_OBSERVATION_DAY =1 indicates that traffic generation will start from Tuesday.

Parameter Name : START_OBSERVATION_HOUR

- Description - Starting hour of traffic generation.
- Data Type - Integer
- Range - Min : 0 - Max : 23 . This parameter can hold a number from 0 to 23 indicating 24 hours in a day.
- Default Value - 9
- Example - START_OBSERVATION_HOUR=9 indicates that traffic generation will start from 9 AM in the morning on specified starting day.

2. This configuration file contains parameters require to define the caller set and their call generation pattern. This configuration file includes a special parameter which is used to characterize generated traffic as good or bad.

Parameter Name : G

- Description - This parameter indicates the type of the generated traffic by the caller set. VoIPTG-v2.1.1 application generates the receiver set only for the configuration file indicating "G=t". Because this indicates the good users. And all the calls generated by the different caller sets are aimed to this good receiver set.
- Data Type - boolean
- Range - This parameter can only take two values. G= t/f. "G=t" indicates good traffic which means the caller set defines in the configuration file will generate good traffic.

The other parameters for defining caller sets and call generation pattern are found below:

1.4.1 Parameter for Configuring Output File name and location

Users of VoIPTG can choose the name and location of the output file to store the generated traffic information. Parameters for configuring format,

name and location of the output file are defined in this section.

Parameter Name : OUTPUT_DIRECTORY

- Description - Location/Path of the output file to be stored.
- Data Type - String
- Default Value - Path of the directory where the jar file of the application (Traffic Generator) is located.
- Example - C:\\TrafficGenerator\\NewCall\\ [Double \\ is needed to define the path.]

Parameter Name : OUTPUT_FILE_NAME

- Description - Name of the output file.
- Data Type - String
- Default Value - Calls.txt
- Example - Calls.txt

Parameter Name : RECEIVER_FILE_NAME

- Description - Name of the file containing the receivers.
- Data Type - String
- Default Value - Receiver.txt
- Example - Receiver.txt

Parameter Name : CALLER_FILE_NAME

- Description - Name of the file containing the callers.
- Data Type - String
- Default Value - caller.txt
- Example - caller.txt

1.4.2 Parameter for Configuring User Sets

VoIPTG maintains two user sets to indicate the callers who generate the calls and the receivers who receive the generated calls. These two user sets are not disjoint. The overlapping percentage between the caller and receiver set is defined by overlapping percentage parameter. Based on the overlapping

percentage, the total number of common users for both sets are calculated from the Caller set. Then the common users are selected randomly from the Caller set and added in the Receiver set. In other words, these users common to both Caller and Receiver sets can be considered as Internal Users for an Operator. User domain of VoIPTG can be explained with figure 1.3.

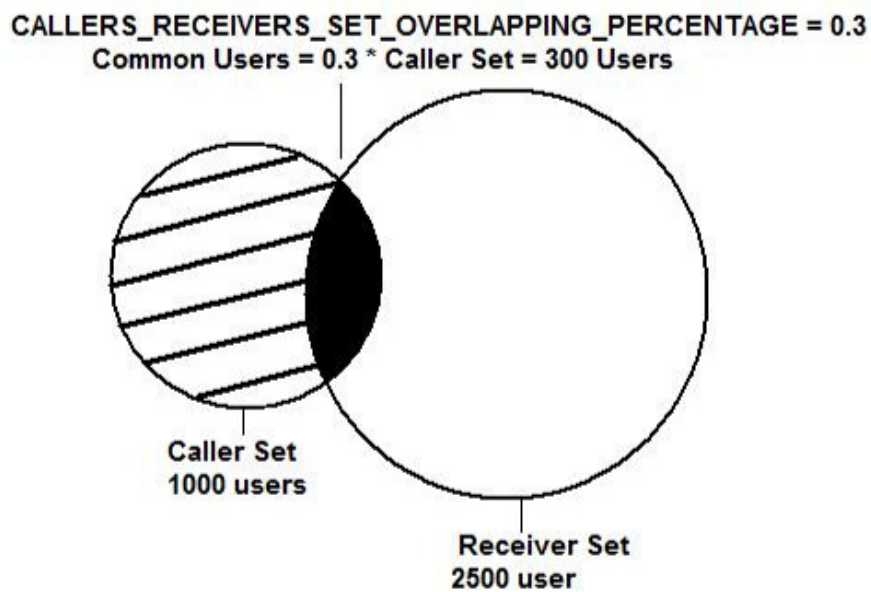


Figure 1.3: VoIPTG User Domain

The configuration parameters to control the User Domain are as follows:

Parameter Name : NUMBER_CALLERS

- Description - Number of users in the Caller set.
- Data Type - Integer
- Range - $0 < \text{NUMBER_CALLERS} > 1.0^7$. The input for parameter 'NUMBER_CALLERS' should be correlated with the call generation rate of the Traffic Generator. For example with a call generation rate of 10 call/sec, Caller set containing only 100 callers

may be unrealistic. This is because it indicates around 8640 calls per day by each user which is very high call generation rate.

- Default Value - 1000
- Example - NUMBER_CALLERS=1000. This indicates 1000 users in the Caller set.

1.4.3 Parameter for Generating Users' Address

VoIPTG generates users following the standard internet mailing address naming convention:

username@subdomain.domain.top-level-domain

The parameters to generate the users' address are as follows:

Parameter Name : NUMBER_DOMAINS

- Description - Number of Domains.
- Data Type - Integer
- Default Value - 1000
- Example - NUMBER_DOMAINS = 1000

Parameter Name : SUBDOMAINS_ENABLED

- Description - Boolean parameter to choose whether subdomain is used in forming users' address.
- Data Type - Boolean
- Range - true, false
- Default Value - true
- Example - SUBDOMAINS_ENABLED=true indicates that sub-domain name will be used in forming the users' address.

Parameter Name : MAX_CHARS_USERS

- Description - Maximum number of character used to generate user name part of the user's address.
- Data Type - Char

- Range - Each character of the username is chosen randomly from a set of characters [a-z, A-Z, 0-9].
- Default Value - 10
- Example - MAX_CHARS_USERS=8 indicates that at most 8 characters will be used to generate user name in the user address.

Parameter Name : MAX_CHARS_DOMAINS

- Description - Maximum number of character used to generate domain name of the user's address.
- Data Type - Char
- Range - Each character for the sub-domain is chosen randomly from a set of characters [a-z, 0-9].
- Default Value - 10
- Example - MAX_CHARS_DOMAINS=9

Parameter Name : MAX_CHARS_SUBDOMAINS

- Description - Maximum number of character used to generate sub-domain name of the user's address.
- Data Type - Char
- Range - Each character for the sub-domain is chosen randomly from a set of characters [a-z, 0-9]
- Default Value - 4
- Example - MAX_CHARS_SUBDOMAINS =3

1.4.4 Parameter for configuring Traffic Variance and Intensity

VoIPTG allows traffic variance over weeks and days and considers that traffic intensity degrades during the weekends and also during the off-peak hour in a day. Configuration file allows setting parameters for defining the traffic degradation in weekends by defining the weekly traffic reduction percentage. And the daily traffic variance is considered by setting the value of peak and off-peak hour in a day. VoIPTG users set the parameter for the starting and ending time of the peak hour where a day is divided in 24 (0 to 23) hour

slots. So a value of peak hour start time as 9 and peak hour end time as 21 means the peak hour starts from 9 am and ends at 9 pm. Parameters for getting traffic variance over weeks and also during the day are follows:

Parameter Name : HIGH_DAILY_RATE

- Description : Traffic Intensity/Traffic generation rate at peak hours in a day.
- Data Type : Double
- Range : HIGH_DAILY_RATE > 0, It should be non-negative and greater than zero.
- Default Value : 0.2
- Measure Unit : Call per second.
- Example : HIGH_DAILY_RATE =0.2 indicates that during peak hours in a day VoIPTG will generate 0.2 calls per second.

Parameter Name : LOW_DAILY_RATE

- Description - Traffic Intensity/Traffic generation rate at off-peak hours in a day.
- Data Type - Double
- Range - LOW_DAILY_RATE > 0, It should be non-negative and greater than zero.
- Default Value - 0.1
- Measure Unit - Call per second.
- Example - LOW_DAILY_RATE =0.1 indicates that during off-peak hours in a day the call generation rate will degrades and then VoIPTG will generate 0.1 calls per second.

Parameter Name : TRAFFIC_VOLUME_REDUCTION_PERCENTAGE_IN_WEEKEND

- Description - This parameter holds the percentage of decrease in traffic volume in the weekends. Generally in weekend the traffic volume degrades. This parameter allows us to define the traffic degradation percentages based on traffic volume in the week days.
- Data Type - Double
- Range - Min : 0 - Max : 1.

- Default Value - 0.5
- Example - TRAFFIC_VOLUME_REDUCTION_PERCENTAGE_IN_WEEKEND = 0.5 indicates that in weekend (Saturday and Sunday) traffic volume reduces to 50% of the traffic volume in the week days.

Parameter Name : HIGH_WEEKLY_VOLUME_RATIO

- Description - Upperbound of the weekly traffic rate ratio (will be multiplied by the daily call rate)
- Data Type - Double
- Range - Min : 0 - Max : 1.
- Default Value - 1.0
- Example - HIGH_WEEKLY_VOLUME_RATIO = 1.0

Parameter Name : HIGH_DAILY_RATE_START

- Description - Starting of Peak-Hour in a day. This parameter holds the starting time of peak hour where a day is divided in 24 (0 to 23) hour slots.
- Data Type - Integer
- Range - Min : 0 - Max : 23 . This parameter can hold a number from 0 to 23 indicates 24 hours in a day.
- Default Value - 9
- Example - HIGH_DAILY_RATE_START = 9 indicates that peak hour starts from 9 AM in the morning on each day. And during peak hour the traffic generation rate follows the HIGH_DAILY_RATE.

Parameter Name : HIGH_DAILY_RATE_END

- Description - Ending time of peak hour/ Starting time of off-peak hour. This parameter holds the ending hour of peak time where a day is divided in 24 (0 to 23) hour slots.
- Data Type - Integer
- Range - Min : 0 - Max : 23 . This parameter can hold a number from 0 to 23 indicates 24 hours in a day.
- Default Value - 16
- Example - HIGH_DAILY_RATE_END = 16 indicates that peak hour ends at 4 PM in the afternoon everyday and the off-peak

Parameter	Value
HIGH_DAILY_RATE	1 call/sec
LOW_DAILY_RATE	0.5 call/sec
TRAFFIC_VOLUME_REDUCTION_PERCENTAGE_IN_WEEKEND	0.5%
HIGH_DAILY_RATE_START	0 (12 AM)
HIGH_DAILY_RATE_END	12 (12 PM)

Table 1.1: Configuring parameters for Traffic Variance and Intensity

hour starts from 5 PM in the afternoon. During the off-peak hour the traffic generation rate follows the LOW_DAILY_RATE.

Figure 1.4 shows traffic distribution over the a week following the configuration parameter for Traffic Variance and Intensity. The parameter configuration for this example is found in table 1.1.

1.4.5 Parameter for Interarrival Time Distribution

In present implementation of VoIPTG, the interarrival time of the calls are exponentially distributed.

Parameter Name : INTERARRIVAL_DISTRIBUTION_FUNCTION_TYPE

- Description - This parameter holds the choice for distribution type for the interarrival time of the calls.
- Data Type - Integer
- Range - 1 : Exponential Distribution
- Default Value - 1
- Example - INTERARRIVAL_DISTRIBUTION_FUNCTION_TYPE=1.

1.4.6 Parameter for Call Duration Distribution

VoIPTG is able to generate call following different distributions for the call duration. Present implementation supports exponential and Log Normal distribution for call duration. Users first need to define the choice for distribution for call duration. The parameter for choosing distributions is following:

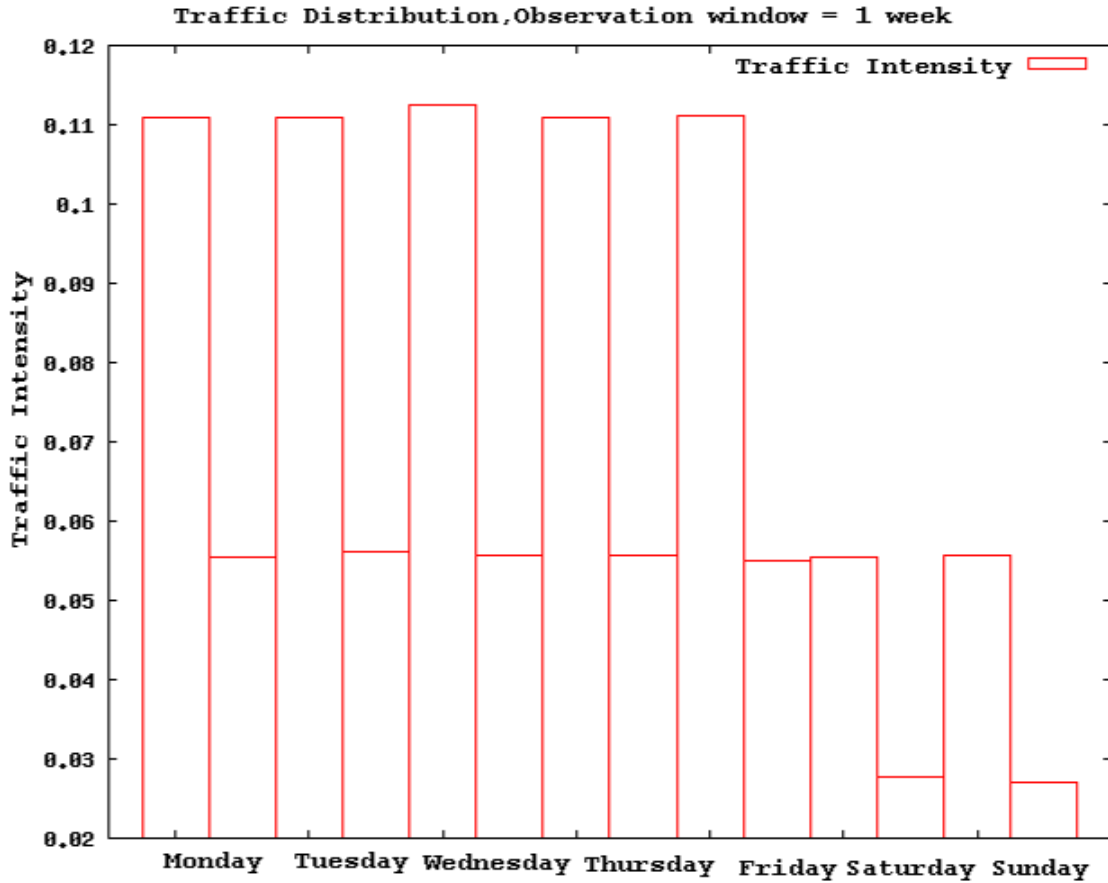


Figure 1.4: Traffic Distribution over a week.

Parameter Name : CALL_DURATION_DISTRIBUTION_FUNCTION_TYPE

- Description - This parameter holds the choice of distribution type for the call duration.
- Data Type - Integer
- Range - 1 : Exponential Distribution, 2 : Log Normal Distribution.
- Default Value - 1
- Example - CALL_DURATION_DISTRIBUTION_FUNCTION_TYPE = 1 indicates that the duration of the generated calls are exponentially distributed.

According to the choice of parameter 'CALL_DURATION_DISTRIBUTION_FUNCTION_TYPE' users need to define the configuration parameters for specific distribution.

The configuration parameters for other distributions can be ignored.

Parameter for Exponential distribution is follows:

Parameter Name: AVERAGE_CALL_DURATION

- Description - Average Duration of the calls in the observation time. It is used as Mean value for exponential distribution.
- Data Type - Double
- Range - $0 < \text{AVERAGE_CALL_DURATION} < 3600$ sec (1 hour).
- Default Value - 180.0
- Measure Unit - second
- Example - $\text{AVERAGE_CALL_DURATION} = 180$ indicates that the average duration of the generated calls is 180 seconds.

Parameters for Log Normal distribution are follows:

Parameter Name: LOGNORMAL_TRUNCATION_POINT_FOR_DURATION

- Description - This parameter defines the truncation point for Log Normal Distribution as the infinite support of Log Normal distribution is not needed (Due to finite duration of calls). Here the truncation point indicates the maximum duration of the generated call in the observation time.
- Data Type - Double
- Range - This parameter is used to restrict the heavy tail of the log normal distribution. We are not interested of calls with large duration as long duration calls are not so frequent, we should set the truncation point less than 60 minutes.
 $0 < \text{LOGNORMAL_TRUNCATION_POINT_FOR_DURATION} < 3600$ sec (1 hour)
- Default Value - 1800
- Measure Unit - second
- Example - $\text{LOGNORMAL_TRUNCATION_POINT_FOR_DURATION} = 1800$ indicates that the maximum duration of the calls can be 1800 seconds (30 min).

Parameter Name: LOGNORMAL_MEDIAN_FOR_DURATION

- Description - This parameter defines the median for Log Normal Distribution.
- Data Type - Double
- Range - $0 < \text{LOGNORMAL_MEDIAN_FOR_DURATION} < 3600$ sec (1 hour)
- Default Value - 87
- Measure Unit - second
- Example - $\text{LOGNORMAL_MEDIAN_FOR_DURATION} = 87$ indicates that the median of duration of the generated calls is 87 seconds.

1.4.7 Parameters of call distribution over the caller

VoIPTG provides the opportunity to switch between four distributions for call distribution over the user. These are call generation with delta probability, Uniform (Max-Min), Exponential and Zipf distribution. Users first need to choose the distribution for volume per user. The parameter for choosing distributions is following:

Parameter Name : CALLER_CALL_DISTRIBUTION_TYPE

- Description - This parameter holds the choice for distribution type for volume per caller.
- Data Type - Integer
- Range - 1 - Delta Probability, 2 - Uniform (Max-Min) Distribution, 3 - Exponential Distribution, 4 - Zipf Distribution.
- Default Value - 1
- Example - $\text{CALLER_CALL_DISTRIBUTION_TYPE} = 2$ indicates that the call generation by the users follows Uniform distribution bounded by a range from minimum to maximum calls.

According to ‘CALLER_CALL_DISTRIBUTION_TYPE’, users need to define the configuration parameters for specific distribution. The configuration

parameters for other distribution can be ignored. The configuration parameters for specific distributions are defined in following subsections.

Parameters of Delta Probability

To generate call with delta probability, it requires only the total number of caller in the caller domain to evaluate the delta probability for all the call generating calls. Here the call generation probability for all the caller is same.

Parameters of Uniform (Min-Max) Distribution

In Uniform(Min-Max) distribution, the users generate call within a range with equal probability. So it requires two parameters defining this range. Configuration parameters for Uniform (Min-Max) distribution are follows:

Parameter Name: UNIFORM_MAX

- Description - Maximum number of call generated by user in a week.
- Data Type - Double
- Range - This parameter indicates the upper range of Uniform distribution. It should be correlated with the global call generation rate of the generator. `HIGH_DAILY_RATE` contains the global call generation rate per second. From `HIGH_DAILY_RATE` we can evaluate the call generation rate per week and this value should be the average of `UNIFORM_MAX` and `UNIFORM_MIN`.
- Default Value - 90
- Measure Unit - Call per Week
- Example - `UNIFORM_MAX = 60` indicates that the users generates maximum 60 calls in a week.

Parameter Name: UNIFORM_MIN

- Description - Minimum number of call generated by user in a week.
- Data Type - Double

- Range - This parameter indicates the lower range of Uniform distribution. It should be correlated with the global call generation rate defined in parameter HIGH_DAILY_RATE. From HIGH_DAILY_RATE we can evaluate the call generation rate per week and this value should be the average of UNIFORM_MAX and UNIFORM_MIN.
- Default Value - 30
- Measure Unit - Call per Week
- Example - UNIFORM_MAX=90 and UNIFORM_MIN=30 indicate that 90 is the maximum number of call can be generated by the user in a week while the minimum number of generated call is 30 in a week. Again users can generate call in this rage from 30 to 90 with identical probability. Here users can generate on average 60 calls per week. So to get this desired output the generator must be able to generate on average 60 calls per week. This introduces the necessity to define the global call generation rate very carefully at the time of defining UNIFORM_MAX and UNIFORM_MIN. In this example the parameter HIGH_DAILY_RATE which indicates the global call generation per second would be 1. This means on average the generator is able to generate 60 calls per week. But if the user choose global call generation rate HIGH_DAILY_RATE=2 then it will surely contradict with the defined UNIFORM_MIN and UNIFORM_MAX value with average 120 calls per week and fails to produce desired output.

We should be careful to define the global call generation rate because this should be equal to the average call generation rate by the users. Sometimes due to daily and weekly traffic degradation users do not achieve desired result. For example, UNIFORM_MAX=90 and UNIFORM_MIN=30 indicates that the users will generate 60 calls per week. So the global call generation rate must be 60 calls/week to get the desired distribution. HIGH_DAILY_RATE = 1 call/sec indicates that on average VoIPTG will generate 60 calls/sec. But if we also introduce traffic degradation in the off-peak hours and in weekends then with call generation rate of 1 call/Sec will fails to give perfect result. This situation is found in figure 1.5. To avoid

this situation we need to increase global call generation rate to maintain the generation rate as 60 calls/week.

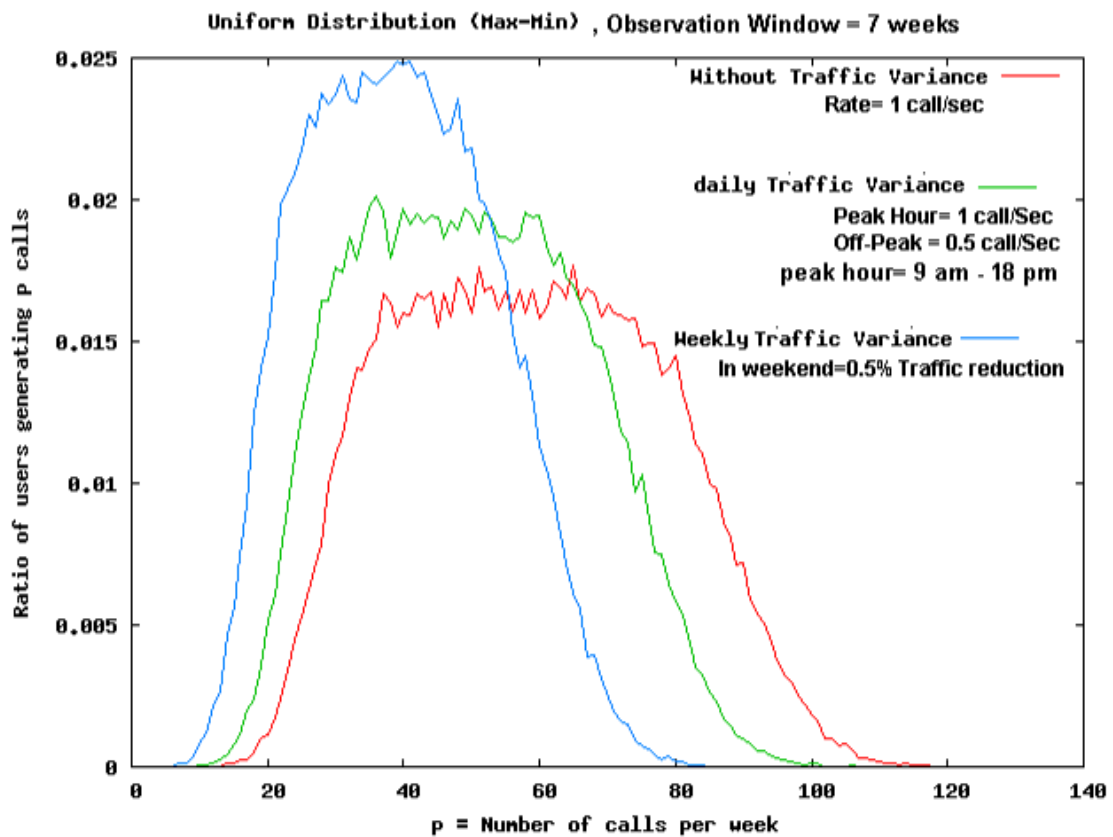


Figure 1.5: Uniform Distribution for volume per caller distribution.

Parameters of Exponential Distribution

In Exponential Distribution, the call generation by the users is exponentially distributed. So it requires one parameters defining mean value of the generated call by the user. The average call generated by the user need not be defined externally, the generator can calculate it through the call generation rate and the total number of callers.

Parameters of Zipf Distribution

For Zipf Distribution, we need to define the truncation point which is the maximum number of generated call by the user and the scale parameter.

Parameter Name: ZIPF_TRUNCATION_FOR_CALLERS_DIST

- Description - Maximum number of calls generated by user in a week.
- Data Type - Integer
- Range - $0 < \text{ZIPF_TRUNCATION_FOR_CALLERS_DIST} < ((\text{Call generation rate per week}) * 10)$.
- Default Value - 500
- Measure Unit - Maximum Number of generated calls in a week.
- Example - $\text{ZIPF_TRUNCATION_FOR_CALLERS_DIST} = 500$ indicates that one users can generate maximum 500 calls in 1 weeks observation time.

Parameter Name: ZIPF_SCALE_FOR_CALLERS_DIST

- Description - Scale parameter for Zipf Distribution. This parameter defines the spread of the zipf distribution.
- Data Type - Double
- Range - $0 < \text{ZIPF_SCALE_FOR_CALLERS_DIS} < 5$
- Default Value - 0.9
- Example - $\text{ZIPF_SCALE_FOR_CALLERS_DIST} = 0.9$

1.4.8 Parameters of call distribution over the Receiver

Present VoIPTG implementation only supports simplest distribution for call distribution over the receiver. The generated calls are distributed over the receiver with a fixed probability defined as delta. As in future we will implement other distributions for call distribution over the receiver, we introduce the following parameter to choose distributions.

Parameter Name : RECEIVER_CALL_DISTRIBUTION_TYPE

- Description - This parameter holds the choice for distribution type for volume per receiver.
- Data Type - Integer
- Range - 1 - Delta Probability
- Default Value - 1
- Example - RECEIVER_CALL_DISTRIBUTION_TYPE = 1 .

According to the choice of parameter ‘RECEIVER_CALL_DISTRIBUTION_TYPE’, users need to define the configuration parameters for specific distribution.

Parameters of Delta Probability

To distribute call over the receiver with delta probability, it requires only the total number of receiver in the receiver domain to evaluate the call receive probability. Here the call receiver probability for all the receiver are same which is defined as delta.