AN ALGORITHMIC APPROACH FOR DESIGNING OF TGREP SIMULATOR

1Jayashree Mohanty, 2Manas Ranjan Mishra

1 Assistant Professor, Dept. Of Computer Science & Engineering NM Institute of Engineering & Technology, Bhubaneswar, Odisha, India
2 Assistant Professor, Dept. Of Computer Science & Engineering, C.V. RAMAN College Of Engineering, Bhubaneswar, Odisha, India

1Email: jayashree.simi2k5@gmail.com

Abstract:

In a Telephony network the most important issue is to select the correct Gateway for the data transformation. The Gateways are generally required for the handling purpose of geographical diversity, capacity, and redundancy. For this issue TGREP (Telephony Gateway Registration Protocol) is used. TGREP is a route registration protocol for telephony destinations on a gateway. These telephony destinations could be prefixes, trunk groups or Carriers. The proposed theoretical implementation of the simulator should allow the user to closely monitor the network under TGREP and consequently decide the values of various parameters so that the network gives an optimum performance.

Based on the above goals, the simulator deals with the following aspects:
- Providing flexibility of defining the network by means of various parameters.
- For a given ITAD configuration the simulator evaluates the performance of TGREP.

1. Introduction

Gateways are generally required to handle geographical diversity, capacity, and redundancy. So in a VoIP networks, Internet Telephony Administrative Domains (ITADs) will deploy numerous gateways so that when a call arrives at the domain, it must be routed to one of those gateways. Gateways are reside in a POP. ITAD will break their network into geographic POPs. Along with gateways POPs also contain a proxy server element that fronts those gateways. The proxy server (Location Server) is responsible for managing the access to the POP, and also for determining which of the gateways will receive any given call that arrives at the POP. The gateway selection depends on many factors like availability, remaining call capacity and call success statistics to a particular PSTN destination. For the proxy to do this adequately, it needs to have access to this information in real-time, as it changes. This means there must be some kind of communications between the proxy and the gateways to convey this information.

The Telephony Gateway Registration Protocol (TGREP) is used to serve this purpose. It is a protocol for registration of routes (destinations) supported by the gateway to the Telephony Routing over IP (TRIP) Location Server. It can be considered an auxiliary protocol to TRIP. Routes learnt through TGREP can be injected into and further processed and/or propagated by the TRIP Location Server. The aim of this work is to give an algorithmic approach to construct the TGREP simulator, which will enable the user to simulate a real time environment consisting of a network which has TGREP deployed on it.

2. TGREP Attributes

A TGREP supports the following attributes:
1. WithdrawnRoutes
2. ReachableRoutes
3. NexthopServer
4. TotalCktCapacity
5. AvailableCkts
6. CallSuccess
7. Prefix
8. TrunkGroup
9. Carrier

2.1 WithdrawnRoutes

The WithdrawnRoutes specifies a set of routes that are to be removed from service by the receiving LS(s). The set of routes MAY be empty, indicated by a length field of zero.

2.2 ReachableRoutes

The ReachableRoutes attribute specifies a set of routes that are to be added to service by the receiving LS(s). The set of routes MAY be empty, as indicated by setting the length field to zero.

2.3 NexthopServer

NexthopServer indicates to the next-hop that messages of protocol be sent to it. This may or may not represent the ultimate destination of those messages.

2.4 TotalCktCapacity

The TotalCktCapacity identifies the total number of PSTN Circuits that are available on a route to complete calls.

2.5 AvailableCkt

The AvailableCkts identifies the number of PSTN Circuits that are currently available on a route to complete calls.

2.6 CallSuccess

The CallSuccess attribute is an attribute used ONLY between a gateway and its peer LS responsible for managing that gateway. If it is received in a route, it is not propagated. The CallSuccess attribute provides information about the number of normally terminated calls out of a total number of attempted calls. CallSuccess is to be determined by the gateway based on the Disconnect cause code at call termination.

2.7 Prefix

The Prefix attribute is used to represent the list of prefixes that the respective route can complete calls to. This attribute is intended to be used with the Carrier or Trunkgroup address family.

2.8 TrunkGroup

The TrunkGroup attribute is used to represent the list of trunk groups on the gateway used to complete calls.

2.9 Carrier

The Carrier attribute is used to represent the list of carriers that the gateway uses to complete calls. It enables providers to route calls to destinations through preferred carriers. This attribute is relatively static.

3. Operation

The operation is carried out by the activities like session establishment, updation of messages, dynamic information collection, error correction and notification, call routing information, behaviour of LS and Proxy, route consolidation, route aggregation etc. The TGREP gateway sends an OPEN message which includes a Send Receive Capability in the Optional Parameters. The Send Receive Capability is set by the gateway to Send Only. The OPEN message also contains the address families supported by the gateway. Once the peer session has been established, the gateway sends UPDATE messages to the TRIP LS with the gateway's entire reachability. TGREP gateway will never generate a NOTIFICATION message in response to an UPDATE message, irrespective of the contents of the UPDATE message. A TGREP gateway maintains one call routing database per peer LS. The database will contain the gateway's reachability information advertised to its peer LS. The TGREP receiver may receive routing information from one or more gateways. It is possible that multiple routes are available for the same destination. These different alternative routes may be received from the same gateway, or from multiple gateways. It is RECOMMENDED that the set of gateway routes for each destination be consolidated, before presenting a candidate route, to the LS. The motivation for this operation should be to define a route that can maximally represent the collective routing capabilities of the set of gateways, managed by this TGREP receiver. Whereas the aggregation operation is carried out for scaling enhancement used by a LS to reduce the number of routing entries that it has to synchronize with its peers. Aggregation may be performed by an LS when there is a set of routes...
{R1, R2, ...} in its database such that there exists a less specific route R where every valid destination in R is also a valid destination in {R1, R2,...} and vice-versa.

4. Architecture

5. Algorithm

1. Accept the ITAD and Parameter configuration files from the user.
2. Pass these configuration files to the simulator( ) function which is to used as the processing function for all the parameters.
3. The Simulator gives a call to the readFileAll ( ) function of Parser class. readFileAll ( ) is used to read all the input files and check all the availability.
4. ITAD config file provides values for databases at the GW (routes, prefixes) , databases at the LS and other global variables.
5. The connectivity matrix conMatrix is initialized by the Simulator.
   conMatrix: it is a 2D integer array. Its size is Number of rows i.e. number of LSs and Number of columns i.e number of Gateways.
   conMatrix[i][j] = 1, if the LS is connected to the GW. It is 0, otherwise.
6. Parameter config provides values for parameters updateFrequencyLevel, noOfUpdateRates and AllCallDuration.
7. The start( ) function is then called which simulates the protocol for given environment for a specific period of time. This function repeatedly calls the procFirstEvent( ) function of the Events class. The procFirstEvent( ) function process event at head of the queue which will order by time stamp values.
8. Then the Events are generated
9. Depending on the type of Event generated they are processed as follows.
   a. CallEvent: A prefix is selected randomly from the available pool of prefixes. The call is then forwarded via a randomly selected LS – GW pair which can terminate call to that particular prefix. The number of callsMade is incremented by one. Now, if the AvailableCkt value for that prefix is not zero then the callsSuccessful is incremented by one and AvailableCkt decremented by one else no changes are made.
   b. BreakCallEvent: The AvailableCkt value of that prefix is incremented by one.
   c. GWCallEvent: This Event emulates calls from PSTN into the GW.
   d. UpdateEvent: The DataBase is refreshed by the current values received from the GW.
   e. LogEvent: This Event causes data to be written to a Log file.

6. Future Implementation

The implementation of the simulator can be implemented in C++ programming language i.e an Object Oriented approach . The problem at hand can be identified as interactions between several entities which exist in a real environment (Gateways, Location Servers, etc.) each having a set of well defined properties.

7. Conclusion

The goal of the work is to implement a TGReP simulator, which would enable the user to simulate a real time environment consisting of a network which has TGReP (Telephony Gateway Registration
Protocol) deployed on it. To start with, the user first enters the topological and resource details with the help of a file. The dynamic parameters are also entered in another file. These files are read, and the database is populated accordingly. The values of the different parameters are logged into a file at regular time intervals. At the same time output files are generated. This simulation is called repeatedly with different values of call arrival rate and update rates, for the same topology, in order to plot relevant characteristics which help analyzing the performance of the protocol. The update rates for which characteristics are to be studied are specified by the user in a second input file. The user of the simulator can then select a suitable topology for deploying TGREP.

REFERENCES: