SARA and eMessenger in different box

# Changes:

## MySQL Server:

In existing system, eMessenger uses SARA’s MySQL server to maintain its eMessenger schema. For our proposed system, we need to install MySQL in that machine.

* If MySQL is not installed on the machine in which we are going to install eMessenger, we need to install MySQL while installing eMessenger.
* If MySQL already installed in that machine.

1. If respective machine installed with SARA previously, we can use the same instance of MySQL. (current behavior).
2. We need to ask log-in credentials for the DB, then eMessenger can create & use separate DB schema in same MySQL.
3. If the MySQL instance installed is not compatible to use with our eMessenger because of version conflicts or other issues, we can install and use new instance of MySQL.

## Communication URL:

In existing system, we are using “localhost” as URL for communication.

(Existing SARA URL: **http://localhost/utils/xml\_alarm.php)**

For our proposed system, the URL will point to SARA’s IP. eMessenger should be modified to get SARA’s IP at the time of installation or permit to skip the step during installation and provide a way to give it later from admin page. For this, we need to give an admin page to provide SARA’s IP later.

# Application Level Changes:

## Server down Intimation to Clients:

In our existing system, both desktop and iOS client checks only eMessenger server’s status and if server is down user is intimated that server is disconnected. If SARA and eMessenger are in two different machines, both the server’s status should be handled. A server down message will be displayed on the user side.

## Communication Update:

We need some interface to update eMessenger alive status to SARA. (Heartbeat update to SARA). For Example : To handle this functionality SIP App updates SARA’s DB after every x interval of time, based on which SARA decides whether it’s alive or not.

If eMessenger or SARA is down then there should be functionality in both the servers to understand the failure & intimate to respective end-users. In failures following can be done

**SARA can do the following process for new alerts:**

1. STOP generating new alerts for eMessenger.

2. Keep all alerts in stack and send all at once when eMessenger communication is back- up.

**eMessenger can do the following process for existing alive alerts:**

1. eMessenger may finish the alert processing without intimating SARA, so that emergencies can be handled by end user.
2. STOP processing till communication backed-up, so that SARA can keep track of all what happened with the alerts.
3. eMessenger finishes its process with all related transaction details in temporary storage, which can be delivered to SARA after communication is established – In this case we need to decide Interface by which SARA can accept huge data of multiple alerts and transactions.

## Audio Alert:

In present scenario, SARA sends audio URL & Local File Path to eMessenger and eMessenger uses this local file path & parses the file using SOX, to convert it to different codecs and forward the URL of converted file to user. This process is followed since the audio file produced by SARA is not supported in eMessenger.

If SARA and eMessenger are in different machine, after getting audio alert URL from SARA, eMessenger will download the file in its local machine & parse it & deliver the URL of the file to user which will result in delay in deliver time compared with the time it received alert from SARA.

To avoid delay on delivering audio alert following can be done: SARA can parse the file to eMessenger supported codec format and sends the URL; eMessenger can immediately deliver the URL to user.

eMessenger supported Audio codec:

1. For WAV format: PCM Audio, 8000Hz, 128 kb/s (1 chnl).
2. For Mp3 format: MPEG “2.5” Layer 3, 8000Hz, 64 kb/s(1 chnl).

# Environmental Issues:

## Dynamic IP Changes:

If SARA and eMessenger running in different machines. IP of both the machines may change based on network policies. In that case both servers should be updated with new IP-Address. To achieve this following can be done :

1. After every x time interval SARA has to listen for change in IP of eMessenger and eMessenger should listen for change in IP of SARA. If any changes found in IP it should be updated respectively.

2. After changing IP-Address there should be an feature to update other server.

Example: In eMessenger admin pages, IP-Address can be changed and updated, this feature can be used to send Update to SARA.

Similar feature should be there in SARA also.

If both the IP of SARA and eMessenger changes simultaneously, updating IP-Address will not be possible in this case & the communication broken mechanism should start as explained above in section of document.

## Network Issues:

Possible network types:

1. eMessenger & SARA on same machine (localhost).

2. eMessenger & SARA on same Local Area Network under same subnet (no intermediated router).

3. eMessenger & SARA on same Local Area Network with multiple routers in between.

4. eMessenger & SARA both on WAN/Cloud

5. eMessenger on WAN/Cloud & SARA connectivity to eMessenger via VPN.

Based on above, different type of network communication will work differently based on many parameters like: network policies, bandwidth etc. All parameters & issues in network to be considered to establish good connectivity between two servers.

## Communication Time Delay:

If SARA and eMessenger are in different networks, communication delay between networks plays a vital role. eMessenger after processing will send it to user and will receive acknowledgement from user and responds to SARA. Here each way of communication will take a time delay based on network lines, traffic and other network constraints.

Hence the response time-out parameter on both servers has to be decided considering all above network delays.