Loway presents

Automatic queue recalls with WombatDialer and QueueMetrics

If you run a call center, serving clients in a timely way is often very complex, as it requires having enough people available to handle traffic spikes.
The number of callers that disconnect because they have been waiting too long in a queue is then an important driver of the quality of your work, and these frustrated callers are the focus of much attention and scheduling/planning efforts in all call centers.
This is because in a traditional setting doing inbound calling you basically had no other way of servicing the client but waiting for the person to call in.

With an Asterisk-based PBX and using digital lines, this scenario changes a bit, as:

- Your average caller has an associated caller-id that often matches a physical phone in their proximity
- Telephone traffic is very cheap compared to the cost of agent time for call handling
- You have ample means of programming the PBX to suit your exact needs

So it is now a conceivable scenario to improve the services you are offering by adding an automated call-back option, so that you search the logs of lost calls and you actively schedule recalls on them in order to get back to people who hung up in frustration.

Copyright ©Loway 2017 · www.loway.ch

The plan: automatic queue recalls

In this article, we explain how to implement a basic call-back scenario using QueueMetrics and WombatDialer. What we do is very easy, as in:

- We periodically run a script to gather the caller-ids of lost calls that were handled on a queue
- We check each caller-id as to be sure is a valid number
- We check that there is no subsequent successful call on the queue from the same caller-id (as to prevent recalling people who already retried themselves)
- We schedule those calls for dialing no more than once per number per day

As our dialing schedule happens on a WombatDialer campaign, we can control the flow of calls through it by adding and removing agents supposed to handle outbound traffic, or pausing it completely during periods of high inbound traffic.

Step 1. Configuring QueueMetrics

In order to gather information from QueueMetrics to an external script, we need to enable XML-RPC access credentials. This is usually very easy to do, as QueueMetrics ships with a (disabled) ROBOT login that allows external access.

Enabling it is very easy: log in as an administrator, click on “Edit users”, edit the “robot” user and set “Enabled” to yes. While you are at it, take a second to change the default password.

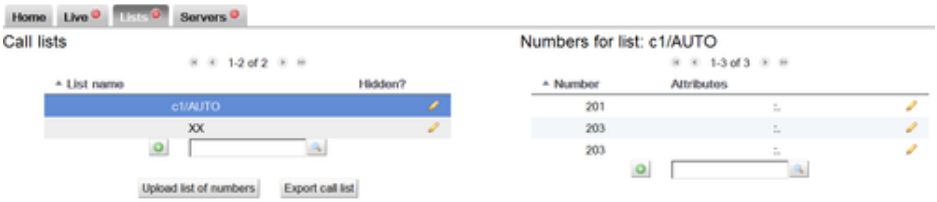
Step 2. Configuring WombatDialer

Set up WombatDialer with a queue end-point (as described for example in “Elastix Queue call-backs with WombatDialer”) and make sure everything is working.

Create a new campaign for calling back people - set its “Idles on termination” property to yes and make the logging QueueMetrics-compatible. This way the campaign can run until needed, waiting for more numbers to be added when idle. Do not add any call list as we will load numbers to be called through the WombatDialer APIs.

Before you start scheduling recalls, your campaign should look like the following one:

Copyright ©Loway 2017 · www.loway.ch



You might also want to pause it, so you can decide when to run it.

Step 3. The script

Scripting QueueMetrics and WombatDialer is really easy. It can be done in any language - we chose PHP as it is well known, has good XML-RPC support to query QueueMetrics and is very simple to edit and customize.

We created a sample script that can easily be downloaded from GitHub - as you will likely edit this to suit your needs, feel free to fork a new project and work on that. Our script is available from https://github.com/Loway/WombatDialerExamples in a folder named “AutoRecall”.

The following parameters should be edited in the script:

```
$qm_server = "10.10.5.11";

$qm_port = 8080;

$qm_webapp = "queuemetrics";

$qm_login ="robot";

$qm_pass = "robot";
```

These parameters specify the XML-RPC connector of your QueueMetrics instance.

```
$wbt_url = "http://10.10.5.18:8080/wombat";

$wbt_cmp = "c1";
```

These parameters specify the URL of WombatDialer and the campaign that calls should be added to. The dialer must be running when the calls are added and the campaign should be active (usually IDLE). Note that the campaign you use for call-back might be paused so that call-backs are actually deferred during periods of high activity.

```
$queue = "300";
```


Copyright ©Loway 2017 · www.loway.ch

```
$lookback = 3600 * 8 ; // in seconds

$allowedPatterns = array(

        "/^555..../",

        "/^0041.+/"

);
```

These parameters decide which set of queue(s) should be scanned and how long is to look back for the current day. Multiple queues can be used, separated by the pipe character.

The last parameter is a set of regexps that will be used to check the numbers read from QueueMetrics. At least one regexp must match for the number to be queued. This is used to avoid queueing invalid numbers or - worse - malicious numbers.

Step 4. Putting it all together

In order to run the script periodically, you could crete a cron job that runs it every 20 minutes. As number are never recalled more than once and the script keeps an history files of numbers already dialed, you can safely run it over and over again.

Once tested, a crontab entry like the following one will automate the running:

```
*/20 * * * * /usr/bin/php /root/WombatDialerExamples/AutoRecall/autoRecall.php
```

This is how a simple run looks like - the scripts logs its activity to STDOUT, so you may want to redirect it to some log file for the keeping.

```
$&gt;php autoRecall.php

Finding applicable period

Loading call log file for 2013-01-24

Looking for data between 2013-01-24.07:54:33 and 2013-01-24.15:54:33

  on server '10.10.5.25' - queues '300'

Query took 0 seconds.

# 201 - Last call lost @ 2013-01-24.15:46:39 - Scheduling.
```

 Copyright ©Loway 2017 · www.loway.ch

```
Adding 201 to campaign c1 on WombatDialer.

Saving call log
```

After running this, you should see that new numbers are added to an AUTO call list like the one shown in the following screenshot; and if the campaign is not paused and agents are available on the recall queue, calls will be dialed as needed.

Improving the solution

In order to run this solution in a real-life scenario, you should edit the campaign in order to:

- set up a time window that matches your agents' presence and when it is customarily allowed to recall. For example, even if a call is queued at 11 PM on a Saturday night, a recall might be acceptable only on Monday morning. This of course depends on what you are doing and the local customs.
- set up reschedule rules in order to handle calls unanswered and busy lines correctly. It would be too bad not to be able to recall just because the caller's phone was busy at the moment
- it could also be useful to connect the caller to a reverse-IVR first, so that they get a message like "Hello, we are calling you back because of your call made at 10.30AM. If you'd like to talk to one of our agents, please press 1 now" before being routed to an agent
- a simple addition that could be made to the script would be to set up a minimum wait time to qualify calls; that is, you would recall only people who waited in queue for more than 10 seconds.
- using a technique very similar to the one explained here, it would be trivial to set up campaigns for quality assessment or customer satisfaction, run as reverse IVRs.

Test WombatDialer for 30 days with the 100 channels Free Trial

Copyright ©Loway 2017 · www.loway.ch