Optimierung der Amazon SES-Senderate mit Mautic

Es gibt verschiedene Möglichkeiten, wie man SES mit Mautic handhaben kann. Ich werde Ihnen meine Praxis mitteilen, die ich für alle unsere Kunden bei Friendly Automate 2 verwende und die wie ein Schweizer Uhrwerk funktioniert. Es gibt auch andere gute Lösungen, ich bin mir sicher, dass jemand Kommentare in diesem Thread posten wird.

Zu schnelles Senden Wenn Sie zu schnell senden, wird Ihre Kampagne von einer Klippe fallen. Das heißt, Sie beginnen zu senden, je nach Geschwindigkeit werden die ersten 500-1000 E-Mails zugestellt, und der Rest geht einfach ins Walhalla, da er weder zugestellt wird noch Sie eine Nachricht von Mautic erhalten, dass etwas nicht stimmt. Sie werden diese Nachrichten mit dem Status "zugestellt" in Mautic sehen, aber nie mit dem Status "empfangen" in Amazon SES, und es gibt keine Möglichkeit, herauszufinden, was verschickt wurde und was nicht (zumindest keine einfache Möglichkeit).

Ich möchte wie folgt vorgehen:

1. Die Warteschlange ist dein Freund

Da ich Tausende von E-Mails versende, verwende ich die Warteschlange. Sie müssen sie in den Mautic-Einstellungen einschalten. Konfiguration > E-Mail-Einstellungen > Wie soll E-Mail gesendet werden? (umschalten auf Queue)

Sie können die obigen Einstellungen verwenden, wir werden sie jedoch in der crontab überschreiben. (Siehe später)

Wenn Sie eine E-Mail erstellen und sie in Ihrem Browser versenden, sehen Sie eine Zeile, während der Versand abgeschlossen wird.

Wenn Sie die Verarbeitung in der Warteschlange nicht einstellen, werden Ihre E-Mails direkt von Ihrem Browser verschickt. Das ist ein guter Weg, um solche Listen zu verschicken, wenn Sie gerne mit dem Feuer spielen. Sie werden mit php-Timeouts, Amazon SES-Limit-Banns usw. konfrontiert.

Wenn Sie die Warteschlangenverarbeitung aktiviert haben, zeigt der Fortschrittsbalken die E-Mails an, die in Ihre Warteschlange verschoben wurden. Wenn Sie bereits in den 1990er Jahren Internetnutzer waren und gerne auf Fortschrittsbalken starren, dann tun Sie das ruhig, aber Sie können Ihre Sendung auch in die Zukunft verlegen und den Cronjob die Arbeit erledigen lassen.

2. Richten Sie die richtigen Cronjobs ein

- Lassen Sie den Cron den Fortschrittsbalken verwalten: bin/console mautic:broadcasts:send Dieser Cronjob wird die E-Mails in die Warteschlange stellen. (Sie finden diese Dateien in app/spool/defualt) Wenn Sie 500 E-Mails auf einmal versenden, müssen Sie diesen Prozess nicht einschränken. Wenn Sie jedoch 50.000 E-Mails versenden, müssen Sie einige Einstellungen vornehmen. (Mehr dazu später.)
- Der Cron, der an Amazon SES sendet, lautet: bin/console mautic:emails:send Dieser Cron überprüft den Warteschlangenordner und schiebt die E-Mails an Amazon SES. Sie sollten dies nicht schneller als 14/Sek. tun (oder was immer Ihr Limit ist).
- Es gibt auch noch einen dritten Aspekt. Sie wollen keinen Stau in Ihrem System verursachen. Das heißt, Sie wollen sicherstellen, dass das Auffüllen der Warteschlange und das Versenden der E-Mails kontrolliert abläuft, ohne dass ein Skript das andere blockiert. Sie

haben auch andere Cronjobs laufen: Segmente, Kampagnen-Trigger usw., die Sie ebenfalls laufen lassen müssen.

3. Zeit für Ihre Skripte

Zunächst müssen Sie feststellen, wie oft Sie Ihre Cronjobs ausführen. Nehmen wir an, Sie lassen jede Minute 5 verschiedene Cronjobs laufen. Wenn Sie wollen, dass alles reibungslos abläuft, müssen Sie sicherstellen, dass jeder Cronjob beendet wird, bevor die 60 Sekunden abgelaufen sind.

Simulieren wir einen Sendevorgang: Schalten Sie Ihre Cronjobs aus, und führen Sie einen Sendevorgang durch, wie Sie es normalerweise tun würden. Schauen wir uns an, was im Hintergrund passiert. Öffnen Sie die Stoppuhr auf Ihrem Handy. (Im Ernst.) Starten Sie sie und geben Sie den Befehl: php path/console mautic:broadcasts:send. Finden Sie heraus, wie viele Sendungen Sie innerhalb von 50 Sekunden in die Warteschlange stellen. Schreiben Sie das auf.

Wenn Ihr Limit bei 14/Sek. liegt, sollten Sie nicht schneller als 850/Min. senden. Unser Grenzwert liegt also bei 800 pro Minute. Um sicherzugehen, dass Sie diesen Wert erreichen, machen Sie einen Test und sehen Sie, wie schnell Sie 800 E-Mails versenden. Gehen Sie zurück zu Ihrer Warteschlange, zählen Sie die Dateien (find . -type f | wc -l) und versenden Sie 800 E-Mails: bin/console mautic:emails:send --message-limit=800 Hat es unter 1 Minute geklappt? Wenn ja, dann müssen Sie dies begrenzen. Wenn nicht, dann haben wir noch etwas Arbeit vor uns.

4. Optimieren

Sie wollen die Warteschlange etwas schneller füllen, als Sie senden, um eine optimale Zustellgeschwindigkeit zu erreichen. Wenn Sie die Warteschlange aufbauen und mit 800/min senden können, dann nehmen Sie die entsprechenden Einstellungen vor:

bin/console mautic:broadcasts:send --limit=800 bin/console mautic:emails:send --message-limit=790

Wenn der Aufbau der Warteschlange langsamer ist, dann ist es sinnvoll, auch das Senden zu begrenzen. Zu schnelles Versenden ist ohnehin nicht gut für die Zustellbarkeit.

In diesem Beispiel werden Sie also 5 Cronjobs laufen lassen. Die ersten 3 (Segment Rebuild, Campaign Rebuild, Campaign Trigger) laufen jeweils 0-30 Sekunden, und die E-Mail-bezogenen Segmente laufen 50-55 Sekunden.

Wie Sie sehen können, sind Sie hier nicht effizient, Sie tun 3 Minuten lang fast nichts und arbeiten 2 Minuten lang hart. Kann das effizienter sein? Gut, dass du fragst.

5. Führen Sie Ihre Cronjobs sequentiell aus

Sie können Ihre Cronjobs in einer einzigen Bash-Datei zusammenfassen. Auf diese Weise wird, sobald ein Befehl beendet ist, der nächste ausgeführt.

Wenn Sie Ihre verketteten Cron-Befehle alle 5 Minuten ausführen, können Sie etwa so vorgehen:

Sie können dies auch jede Minute im Batch-Verfahren durchführen, aber stellen Sie sicher, dass Sie alle Ihre Crons begrenzen. Zum Beispiel (Auch hier hängt es von der Geschwindigkeit Ihres Servers ab.)

mautic:broadcasts:send --limit=150 mautic:emails:send --message-limit=150 mautic:campaigns:rebuild --batch-limit=300 mautic:segment:update --batch-limit=900 mautic:campaigns:trigger mautic:import --limit=500 mautic:webhooks:process mautic:reports:scheduler

Online URL:
https://kb.mautic.org/article/optimierung-der-amazon-ses-senderate-mit-mautic.html