

# BayesBitMe

Are you running multiple [SpamAssassin](#) systems, with only one primary MX record? You'll note that the primary MX box takes care of the majority of the email load, but the secondary box will still get hits from spammers trying to bypass spam filtering (expecting, I suppose, a lower level of protection. That'll show 'em.)

The primary box is benefitting from the lion's share the Bayes learning; unfortunately, that second (or tertiary) box pretty much gets nothing but spam, so its cynical view of the world is that almost all email is spam. Should your primary box fail and incoming email reverts to the secondary, almost all email will be slapped with BAYES\_99 and quarantined (or dropped) and you'll get scads of false positives. Once the primary box comes back up the problem goes away, but you've still quarantined or dropped a lot of good mail.

You could spend a lot of time trying to train your secondary box(es), but that's a lot of unnecessary work. Instead, use the Bayes knowledge that your primary box has generated by copying over all of the bayes\_\* files from the primary to the secondary boxes, making sure all the rights are correct, and restarting. You could do this on a regular basis manually, or you could get clever and figure out a way to do it automatically using clustering, FTP, or such. Regardless of how you do it, it's a must if you truly want fault tolerance in your [SpamAssassin](#)/Bayes email filtering system.

[SpamAssassin](#) 3.0.0 includes an SQL backend for Bayes – this would allow multiple servers to share a single Bayes DB efficiently. If you are running this configuration with SQL Bayes, please share your experiences here.