Goal

To incentivise the use of RSV. In the limited case, it's to incentivise users who are purchasing goods/services from merchants. In the broader case, it's to incentivise traders/market makers as well consumers of goods/services.

Initial proposal

Have a set pool of RSR that is to be given to users every epoch (a weekish?) based on their proportion of RSV transacted in that epoch out of all RSV transacted in that epoch by all users.

reward_for_individual = (amount_transferred_in_epoch / amount_everyone_transferred_in_epoch) * epoch_RSR_reward_amount

Pros:
1. It does incentivise increased usage of RSV (kind of)

Cons:
1. It doesn't always reward the target groups
2. The benefit to the system is limited
3. The amount rewarded to users can vary wildly
4. It's gamable to the point that the target groups get near-zero reward

Cons explained:
1. Regular users who send around some RSV to, say, trade for ETH on some DEX, will get rewarded even though they're not part of the target group. Since RSR is going to them that would've gone to the target group, the target group therefore has less incentive than it otherwise would have which reduces the effectiveness per $ spent on this scheme.

2. For example if $1m was given in rewards, which increased the 'value' of the system by $2m because of increased usage/network effects etc. And if increasing the rewards to $2m increased the value of the system by $3.5, it would be worth doing because going from the former to the latter spends an extra $1m but gained $1.5m, so there's still a net gain. If it's working, why stop?

3. If the goal is to reward, on average, 5% of the value of a transaction, but there are fewer transfers than expected for a given epoch reward, then the amount rewarded could end up being something crazy like 30% or >100%, which means wasted capital.

4. The problem with having a set pool that everyone competes for is that it's a tragedy-of-the-commons-like thing, where the incentive for each player at every point in time is to spend as much money on transfers until their marginal increase in RSR rewards == gas fee of another transfer tx.

Proof by example, imagine the following:
There are 2 profit-seeking yIeLD fARMerZ, A and B. They just finished a short season of toiling in the YAM fields and are looking for another crop to plant. They sold all their sweet juicy YAMs and both have $10k to farm with now.
The reward in RSR is $10,000 every week.

The amount 'naturally' (not for the purpose of yield farming) transferred in a week is 313,915 RSV (the amount transferred in the last week).
The gas cost of a transfer of WRSV (wrapped RSV) is probably around 50k if a regular RSV transfer (to an address that already had a non-zero balance) is around 30k. Assuming a gasprice of 80gwei, the ETH cost of a WRSV transfer == 0.004 ETH == 1.6 USD

The amount yielded for each farmer per epoch is:

$$\left(\frac{x}{313915+x}\right)*10000$$

where x is the total amount transferred by the farmer.
The amount spent by the farmer on transaction fees, assuming they always spend their max ($10k) in each transfer, is the number of transfers multiplied by the cost of each:

$$\frac{x*1.6}{10000}$$

We want to know how much each farmer will transfer before it's no longer profitable, which is when the margin yield == 1.6 USD

Say A starts to farm until it's no longer profitable. It's no longer profitable when the extra yield from another transfer equals the cost.

$$\left(\frac{x+10000}{313915+x+10000}*10000\right)-\left(\frac{x}{313915+x}*10000\right)=1.6$$

Plugging into Wolfram. Tractor go brrrrr:
x = 4.11×10^6

Amount spent = 1.6 * (2.81316×10^6 / 10000) = 656
Amount farmed = 9290.41
Total profit for A = 9290.41 - 656 = 8634.41
Nice. Solid harvest.

Then say B starts to farm.
The total transfers made by not-B at this point = 4.11×10^6 + 10000 + 313915 = 4433915

$$\left(\frac{x+10000}{4433915+x+10000}*10000\right)-\left(\frac{x}{4433915+x}*10000\right)=1.6$$

Tractor go brrrrr:
x = 1.21*10^7
Amount spent = 1.6 * (1.21*10^7 / 10000) = 1936
Amount farmed = 7324.33
Total profit for B = 7324.33 - 1936 = 5388.33

At this point, A's amount farmed now = (((4.1*10^6 / (313915 + 4.1*10^6 + 1.21*10^7)) * 10000) = 2482.75

A's profit is now 2482.75 - 656 = 1826.75, quite the reduction

Round 2:
A starts farming again.

$$\left(\frac{x+10000}{313915+1.21*10^7+x+10000}*10000\right)-\left(\frac{x}{313915+1.21*10^7+x}*10000\right)=1.6$$

Tractor go brrrrr:
x = 1.54*10^7 (this is the total amount spent, including what was spent in round 1)
Amount spent = 1.6 * (1.54*10^7 / 10000) = 2464
Amount farmed = 5536.79
Total profit for A = 3072.79

Then B starts to farm again:

(((x+10000) / (313915 + 1.54*10^7 + x+10000)) * 10000) - ((x / (313915 + 1.54*10^7 + x)) * 10000) = 1.6

$$\left(\frac{x+10000}{313915+1.54*10^7+x+10000}*10000\right)-\left(\frac{x}{313915+1.54*10^7+x}*10000\right)=1.6$$

Tractor go brrrrr:
x = 1.56199*10^7
Amount spent = 1.6 * (1.56199*10^7 / 10000) = 2499.18
Amount farmed = 4984.99
Total profit for B = 2485.81

At this point, A and B have collectively spent 2464 + 2499.18 = 4963.18 after only 2 rounds. As more rounds progress, each farmer's profit reduces, tending towards 0 ad infinitum.
Even if it didn't tend to 0 (which it does), we can see that the amount earned by 'honest' participants collectively after only 2 rounds is:

((313915 / (313915 + 1.54*10^7 + 1.56199*10^7)) * 10000) = 100.18

As more rounds progress, the reward for 'honest' participants also tends towards 0.

The only way this can be avoided in this dynamic is if A and B form a cartel so they don't have to keep spending money to compete against eachother and instead split the cost and profit. But as long as there is atleast 1 other farmer who isn't in this cartel, the situation devolves into the situation above where everyone makes nothing.
Tangent: a sneaky way around this competition between A and B is if B does nothing for the whole epoch until the last block before the epoch ends and fills up the whole block with their own transfers - thus B essentially 'snipes' the reward. However if A is watching out for this, they can compete again by

competing for block space with transaction fees, and the situation devolves into the above situation where everyone makes nothing again.

Solutions corresponding to cons above:

1&4. Whitelist which transactions get rewarded by only rewarding those that spend to registered vendors. This means having an admin key in WRSV to register vendors (otherwise farmers would just add register themselves as vendors), but this doesn't really add any negatives for decentralisation because it only affects who can get rewarded and doesn't affect the base system. Whitelisting addresses means that rewards are only paid to the target audience (which means proportionally greater rewards for those buyers and therefore a higher incentive, meaning more purchases, making the scheme more effective) aswell as preventing the above situation with farmers, except in the case where registered vendors buy from themselves. It's possible that this could happen, but it would be very obvious which vendors are abusing the system and they could therefore be de-registered as a result. In the ideal situation where the target groups are very large, it might be impractical to register all of them. However, this scheme is intended to bootstrap the network effects which is when the groups are small and so the scheme will presumably end before the groups are too large to handle efficiently.

2&3. Give rewards based on each transaction independently, e.g. a set % of the transfer amount. The amount transferred in a given epoch can also be capped. This way, users always know ahead of time exactly what reward they will get and removes any guesswork – if a user isn’t sure whether they want to make a purchase because they don’t know whether they’re going to get 1% or 10%, they might just avoid the purchase. Confidence is a key ingredient to encouraging economic activity. This also mirrors standard ‘cashback’ schemes on credit cards and will therefore reduce the friction for users because they already understand the scheme. If there’s less demand than expected, no capital is wasted giving people 30%. If there’s more demand than expected such that the cap is reached (if there is a cap) for that epoch, all subsequent transactions will not be rewarded – but this is arguably a worthy tradeoff.

Alternative Proposal A