Adam will explain how we get our data

Thanks Maddie.

In order to access all the data you see we make a number of API calls from the Users Browser or server. And in order to bring you live updates, we need to be making these calls continuously.

So, what is an API call? Imagine you’re at a restaurant and you ask your waiter for a drink. The waiter goes off to the bar, grabs the drink, and brings it back to you. That is essentially an API call. We are sending out requests for specific information to the cryptocurrency exchanges, and other data sources, and wait for the data to come back, at which point we are able to process the data and display it on our site.

Now imagine 1000 people all ask the same waiter for a beer at the same time. That waiter would be so overwhelmed they wouldn’t be able to process the requests. This is a **DISTRIBUTED** denial-of-service, or DDoS attack. People can use these attacks to target crypto currency exchanges in attempt to change the market price of coins.

Exchanges have put prevention measure to negate these DDoS attacks. As such we are only allowed to make an API call roughly every 10 seconds. However, given that API calls can be made from either the User side (the browser), or the Server side. If we were to make these API requests to the exchanges from the Users browser, then were are making an API request for each user, 1 user 1 request every 10 sec 2, 3 etc., 1,000,000. If we had 1,000,000 users or more that would be 1,000,000 requests made to each exchange.

Therefore, we need a way to make API calls to our exchanges, that would not increase the number of calls as the number of Users increase. In order to achieve this, we implemented a websocket. A websocket sets up a continuous connection from our server to our users, which we can pass data. By using a websocket we will only ever make 1 API call per exchange, every 10 seconds. Whether we have 1 user or 1 million users, the amount of API calls made from our server will remain the same.

Now not every API call from Deltage uses the websocket. We do have API calls that are made from the users browser. Which Peter will talk to more as he describes the implementation of our charts.

A few API calls are made as the user clicks about the page. These calls don’t have potential disrupt the crypto currency market and therefore are not needed to be made on the server. These calls are also not live updates, and are only triggered by user interaction. Therefore only providing the user with data they wish to see.

are able to set up our server to make 1 API call to each exchange (the orange lines), roughly every 10 seconds

and remain within the API rate limits in our case 5x/minute. We then send our data through the websocket to our users. No matter how many users we have, it could be 1, 1000, or 1,000,000, the websocket allows us to still only make 1 API call per exchange, remain in the API rate limit of the exchange, and therefore help prevent DDoS attacks with the use of Deltage.

Not every API call from Deltage requires a the use of the websocket. A few API calls are made as the user clicks about the page

Peter will now provide more detail towards are graphs.

API calls can either be made from the User (the browser) or from the server.

We didn’t want our website to be able to used in these types of attacks, so we implemented the use of a websocket. A websockket sets up a continuous stream of data from our server to our users, the purple lines behind me. Therefore, we are able to set up our 1 API on our server to each exchange (the orange lines), and remain within the API rate limits in our case 5x/minute. We then send our data through the websocket to our users. No matter how many users we have, it could be 1, 1000, or 1,000,000, the websocket allows us to still only make 1 API call per exchange, remain in the API rate limit of the exchange, and therefore help prevent DDoS attacks with the use of Deltage.

Not every API call from Deltage requires a the use of the websocket. A few API calls are made as the user clicks about the page

Peter will now provide more detail towards are graphs.