What's the role of a nonce in a transaction in A3…?

1. We can copy Ethereum's model, and require a tx with nonce N-1 to have been mined before accepting N. Note that we are not enforcing this in the protocol at the moment.

2. We can just require nonces to be unique, and leave ordering to the account abstraction, thus allowing out-of-order txs at the protocol level if the user wants to.

3. We can instead use nonces as a salt to hide a tx request. Since we'll be calculating a tx hash by hashing all the tx fields (see here), an attacker could brute force through common tx requests and compare them to the resulting hash to guess what the user intent was.

This is not possible since the attacker would not be able to predict the signature - we just need to make sure the tx signature is not leaked.

1. We can use the nonce just to let the user differentiate two txs with the same fields. So if the user wants to repeat an action, they just need to tweak the nonce to a different random value. Note that not making the pair (sender, nonce) be unique makes it impossible to replace or cancel in-flight txs, unless we add an explicit API for that.