Contracts vulnerabilities

Vulnerabilities list: #1

Involved contracts and level of the bugs

The present document aims to point out some vulnerabilities in the autonolas-registry contracts.

Vulnerabilities

1. `depositServiceDonationsETH` function

Severity: Low

The following function is implemented in the Treasury contract:

```
function depositServiceDonationsETH(uint256[] memory serviceIds, uint256[]
memory amounts) external payable
```

This service donating function calls another function from the Tokenomics contract that ultimately results in calling the internal function `_trackServiceDonations()`. The latter one checks whether agent and component Ids of each of the passed service Id exist, and if not, reverts with the `ServiceNeverDeployed()` error. The error arises from the fact that the service was never deployed, and its underlying component and agent Ids were not assigned (happens during the deployment of each service). However, if a specific service is deployed, then terminated, there is a scenario that it is updated with a different set of agent Ids, making the original donation distribution setup invalid. If this updated service is donated before it is re-deployed, the donations will be distributed between its old component and agent Ids owners.

We recommend not to donate to the service that is currently not in the `Deployed` or `TerminatedBonded` state. The state of the service can be easily checked via the ServiceRegistry contract view function `getService(uint256 serviceId)`.