Kickstarter

Soluciones

Gonzalo Martín Filón
gonzalofilon00@gmail.com

Solución al problema del despliegue del contrato

El problema del despliegue del contrato yo lo solucioné haciendo que no sea el manager quien despliega el contrato.

Mi aproximación es que este smart contract sería solamente la capa en blockchain de una aplicación más grande, por ejemplo, un sitio web. Por lo tanto, y para evitar la posible corrupción del contrato por parte del manager de la campaña, el contrato debería ser desplegado por la empresa, en este caso kickstarter.

La vulnerabilidad de que un usuario maliciosamente haga excesivos despliegues de contratos la mitigaría incorporando seguridad básica en la capa correspondiente de la aplicación abierta al público. Bastaría con un login seguro, tal vez verificación de identidad a través de algún método de los existentes y establecer un número máximo de contratos a desplegar sin pagar, o lo que la necesidad del negocio establezca.

Solución al problema del bug

Una alternativa que se me ocurrió (y que luego leí que existe) sería usar lo que yo pensé como **IContract**. Un IContract es un contrato inteligente que actúa como interfaz, es decir que no tiene implementación. Lo que tendría el IContract es por un lado, un address (modificable solo por el owner) que apunta a la implementación en vigor, y por otro lado la firma de los métodos. De esta manera, dentro de cada método del IContract lo único que hay es un llamado al método correspondiente de la implementación y su correspondiente retorno de valor en caso de que correspondiese. En caso de haber un bug, bastaría con desplegar un nuevo contract (ImplContract) y luego cambiar la implementación que está utilizando la interfaz.

El único dilema que plantea la utilización de este método es que no se acopla a la filosofía de blockchain, donde el pasado es inmutable. Por cuestiones en principio éticas, los IContracts deberían tener una advertencia sobre su naturaleza cambiante.