

PollFic

Un sistema para organizar encuestas

AUTORES:

Nombre	Email
David Touriño Calvo	david.tourino@udc.es
Brais Muñiz Castro	brais.mcastro@udc.es
Manuel Daniel Gabín Brenlla	daniel.gabin@udc.es
Alejandro Matos García	alejandro.matos@udc.es
Juan Manuel Rey Escobar	j.manuel.rey@udc.es
Paloma Piot Pérez-Abadín	paloma.piot@udc.es

Descripción del sistema

A continuación se procede a analizar nuestro sistema informático de organización de encuestas, cuyo objetivo es permitir que los usuarios puedan crear sus propias encuestas para recabar información, o bien votar en encuestas ya existentes.

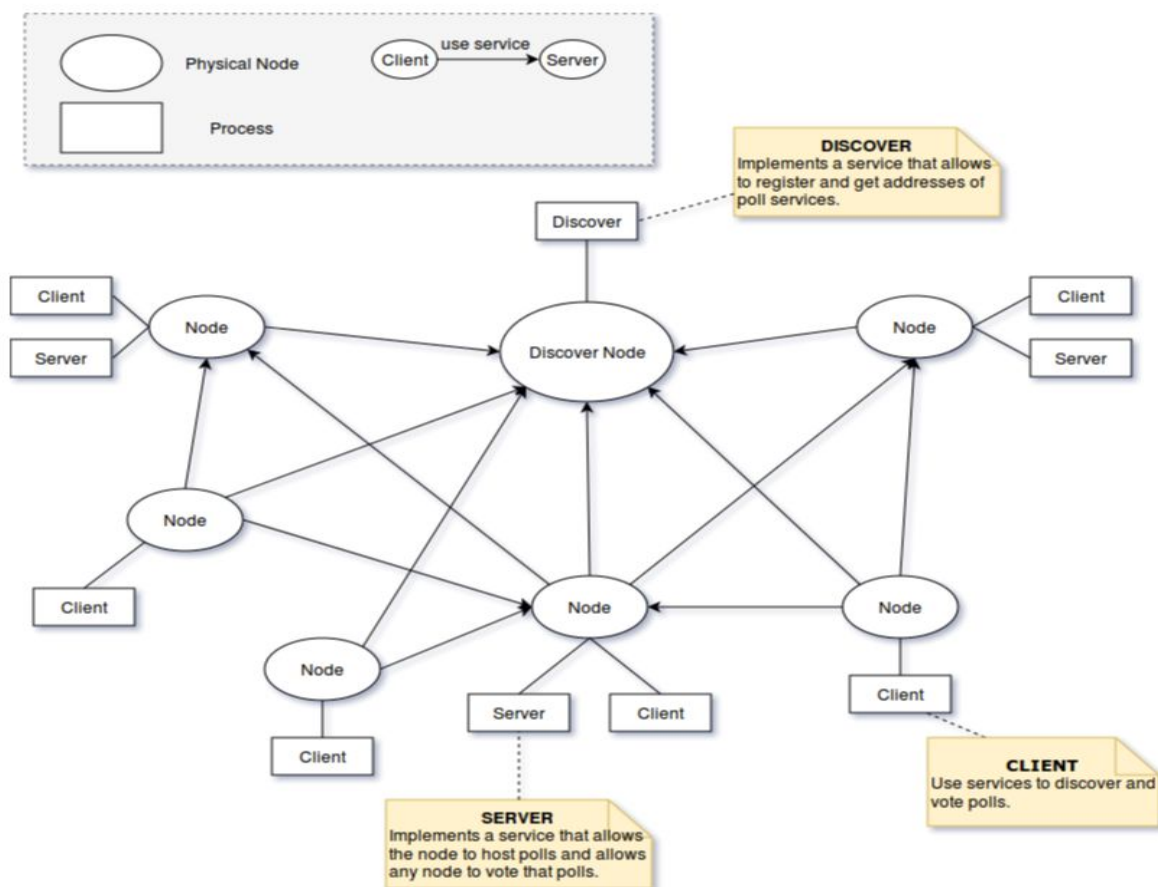
El usuario que inicia una encuesta ejecuta un proceso **servidor** que se encargará de recibir y procesar los votos para todas sus encuestas activas.

El usuario que quiera votar en una encuesta iniciará un proceso **cliente** que se encargará de emitir su voto a través de la red.

Para que los nodos votantes puedan encontrar a los nodos que organizan la encuesta es necesario que los primeros puedan descubrir a los segundos.

Como solución al problema, el sistema cuenta con un segundo tipo de nodo, llamado **discover**, en el cual se registra cada encuesta junto a los datos necesarios para llegar hasta el nodo que la organiza.

VISTA FÍSICA

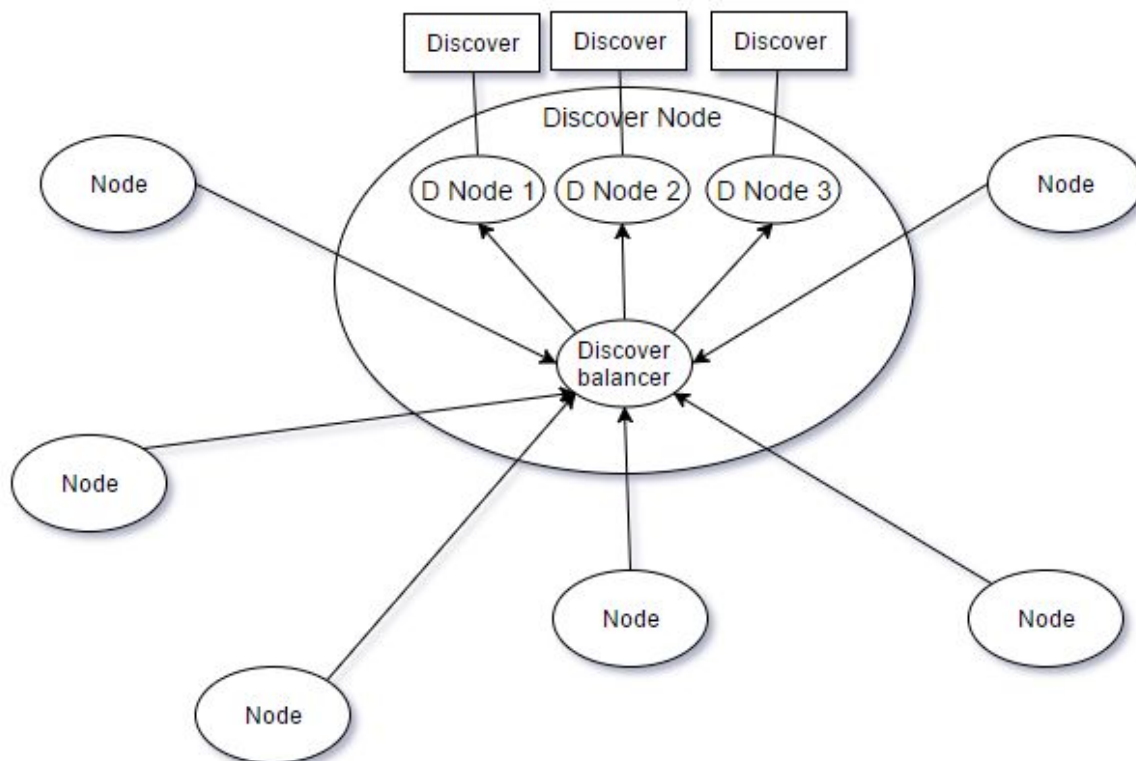


Requisitos no funcionales

Disponibilidad: replicación del Discover Node

El “Discover Node”, como se le denomina en el diagrama de despliegue, es un **punto de fallo único** en el sistema. Si dejase de funcionar, los nodos votantes no podrían descubrir la ubicación en la red de los nodos organizadores de encuestas.

Hemos solucionado este problema con la replicación del “Discover Node”, incorporando un **balanceador de carga** que se ocupa del rendimiento del sistema.



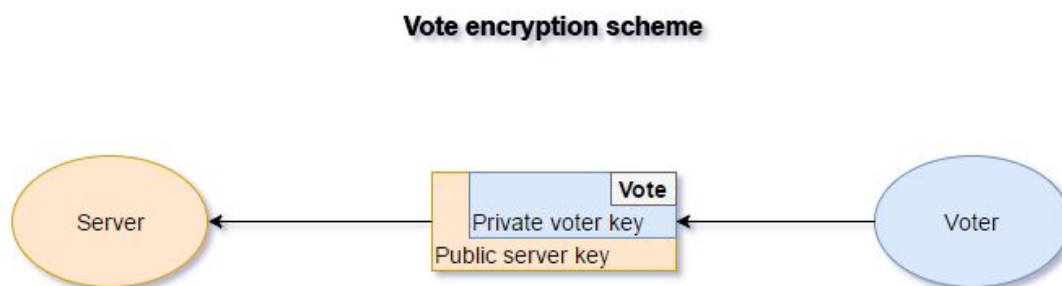
Seguridad: autenticación y confidencialidad

Para evitar que se pueda suplantar la identidad de un nodo al emitir un voto, y garantizar la integridad de los datos, hemos optado por utilizar un sistema de cifrado con clave pública y privada.

En nuestra red, el “Discover Node” actúa como autoridad certificadora para los nodos. Al votar, el voto se cifra con la clave privada del nodo votante. Solo es posible descifrar el voto con la clave pública del nodo votante original, así nos aseguramos de que el votante es quién dice ser.

Pero cualquier persona podría descifrar el voto con la clave pública del votante y acceder a su voto.

Si añadimos un segundo cifrado, por encima del primero, con la clave pública del nodo organizador de la encuesta a la que va dirigida el voto, podemos conseguir confidencialidad. Solo el nodo organizador de la encuesta podrá descifrar el voto con su clave privada.



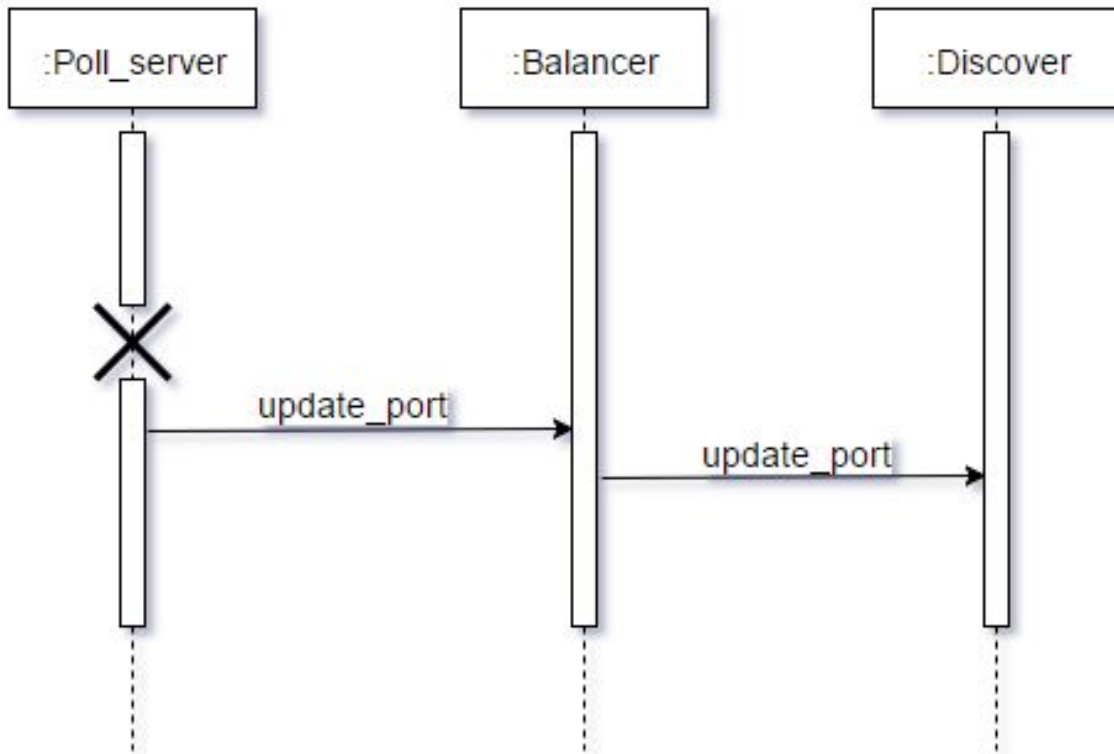
Por problemas de implementación, al final se ha decidido solo hacer cifrado a un nivel, utilizando la clave pública del servidor, para garantizar al menos la confidencialidad.

Recuperación ante fallos: recuperación de un nodo servidor de encuestas

Cuando se inicia una encuesta nueva, ésta se registra en la lista de encuestas del “Discover Node”, y se borra de la misma cuando la encuesta se cierra de forma normal a petición del usuario. Si el nodo organizador de una o más encuestas abortase su ejecución de forma anormal, sus encuestas seguirían listadas en el “Discover Node” pero votar no sería posible. Además, cuando el nodo reanudase su ejecución, su dirección en la red podría haber cambiado y la lista de encuestas reflejaría información incoherente.

Solución: cuando un nodo encuesta inicia su ejecución comprueba si hay encuestas activas pendientes, y si es así, se comunica con el “Discover Node” para actualizar su nueva dirección en la lista.

Cuando un poll server cae y se recupera



Comentarios sobre arquitectura

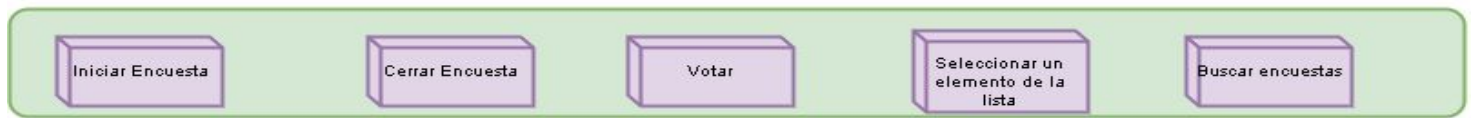
Aunque todos los nodos tienen la capacidad de ofrecer un mismo servicio, la arquitectura no encaja con un P2P, ya que en un mismo instante de tiempo los nodos no ofrecen el mismo servicio.

Además, en un estilo P2P semi-centralizado el nodo de descubrimiento solo revela la ubicación de los nodos vecinos, en nuestra aproximación también revela las distintas encuestas que contienen esos vecinos.

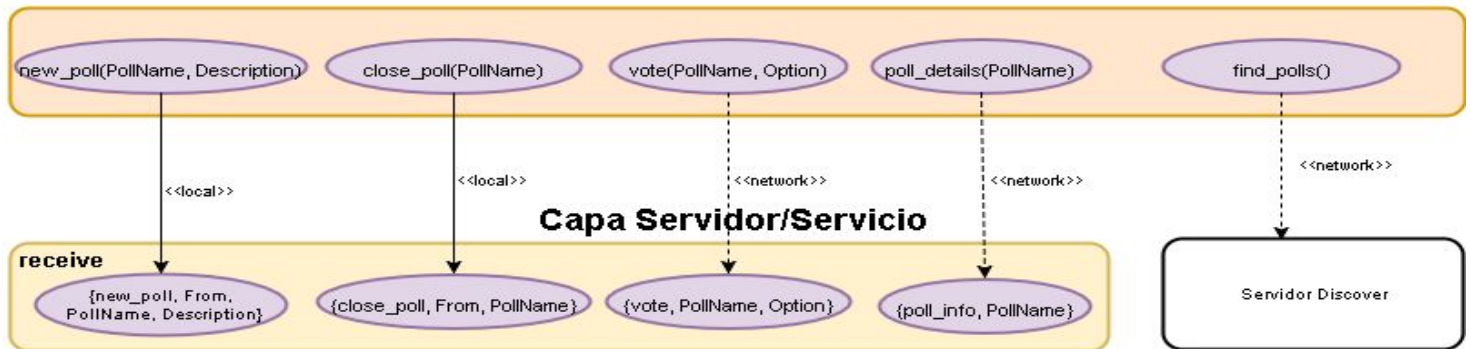
Nos encontraríamos por lo tanto con un cliente-servidor distribuido, donde los nodos servidores pueden ejercer de cliente, comunicados entre sí con un nodo de descubrimiento, que recuerda a estilos arquitectónicos de P2P.

Además, el sistema está organizado por “capas”, representadas en el siguiente diagrama..

Capa GUI



Capa Cliente / Acceso al servicio

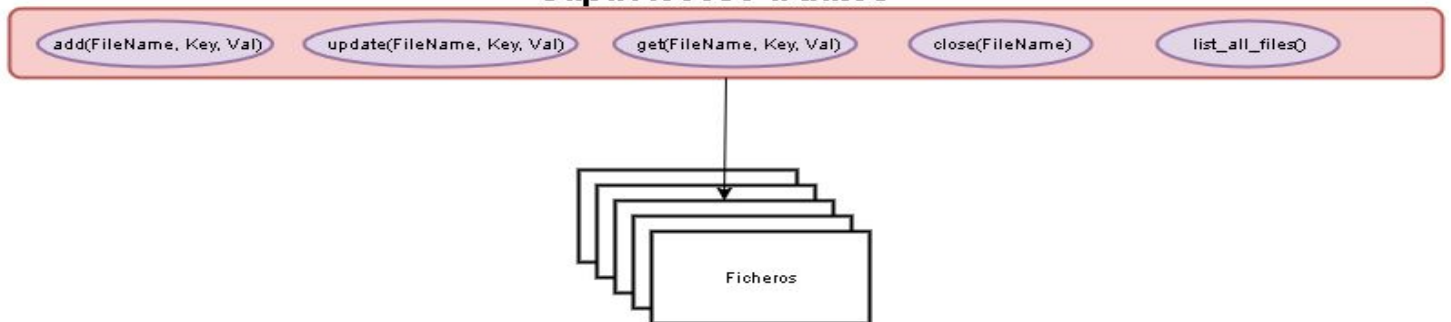


Capa Servidor/Service

Capa modelo



Capa Acceso a datos



VISTA DE DESARROLLO

Pollfic project files structure

