Universidad Autónoma de Nuevo león

Facultad Ciencias Fisco Matemático

DOO

LSTI

Ensayo Cliente y Servidor

Miguel Salazar

Edgar Vázquez Márquez

1657462 Aula: 413

Octubre de 2017

San Nicolas, Nuevo León

Cliente y Servidor

Primero que nada, veremos la definición de cliente y servidor. Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta.

Cliente

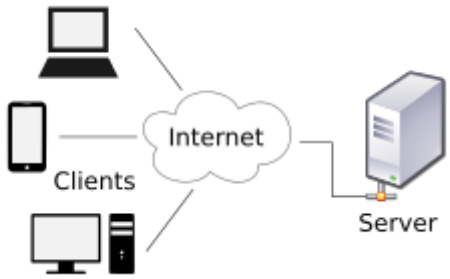
El cliente es el proceso que permite al usuario formular los requerimientos y pasarlos al servidor, se le conoce con el término front-end. Algunas funciones:

- Administrar la interfaz de usuario.
- Interactuar con el usuario.
- Procesar la lógica de la aplicación y hacer validaciones locales.
- Generar requerimientos de bases de datos.
- Recibir resultados del servidor.
- Formatear resultados.

Servidor

Es el proceso encargado de atender a múltiples clientes que hacen peticiones de algún recurso administrado por él. Algunas funciones:

- Aceptar los requerimientos de bases de datos que hacen los clientes.
- Procesar requerimientos de bases de datos.
- Formatear datos para trasmitirlos a los clientes.
- Procesar la lógica de la aplicación y realizar validaciones a nivel de bases de datos.



A otro punto del tema de cliente y servidor también hay diferentes técnicas. Algunas de las diferentes técnicas para comunicar datos entre el cliente y servidor:

cookies

Las cookies son algunos datos, almacenados en unos pequeños archivos de texto, en la computadora.

Cuando un servidor web ha enviado una página web a un navegador, la conexión se cierra y el servidor se olvida de todo acerca del usuario.

Las cookies fueron hechas para resolver el problema de recordarle al usuario cierta información como:

- Cuando un usuario visita una página web, su nombre puede almacenarse en una cookie.
- La próxima vez que el usuario visite la página, la cookie "recuerda" su nombre.

NOTA: JavaScript puede crear, leer y eliminar cookies con la propiedad **document.cookie** .

En seguridad las cookies tienen la capacidad de realizar seguimientos de los movimientos que realiza el usuario dentro de un sitio, lo cual puede ser recopilado y usado con fines ajenos a su propósito original.

Sesiones

Las variables de sesión solucionan este problema almacenando información de usuario para ser utilizada en varias páginas (por ejemplo, nombre de usuario, color favorito, etc.). De forma predeterminada, las variables de sesión duran hasta que el usuario cierra el explorador.

¿Cómo funciona?

La mayoría de las sesiones establecen una clave de usuario en el equipo del usuario que se parece a esto: 765487cf34ert8dede5a562e4f3a7e12

hidden inputs

El elemento “input”, teniendo el valor "hidden" en su atributo “type”, representa cualquier cadena de texto arbitraria que no está pensada para ser vista o editada por el usuario. Los controles ocultos son especialmente útiles para enviar datos al servidor definidos por el autor, basados o no en la interacción con el usuario.

Estos campos ocultos son representados por <input type=”hidden”>. Debido a malas prácticas de programación, algunas veces estos campos son utilizados para pasar información importante Información que debería estar almacenada únicamente en el servidor y no en el cliente. Normalmente los usuarios nunca ven estos datos, pero los hackers o usuarios maliciosos, fácilmente pueden descubrir estos datos y explotarlos.

Parámetros en la URL

Para construir un URL que inicie cualquiera de las aplicaciones web de Network Manager directamente desde un navegador web. Por ejemplo, puede crear una dirección URL para iniciar la vista de saltos que contiene un mapa de red predefinido.

Estos parámetros se pueden escribir directamente en la barra de direcciones del navegador. También, puede escribir una herramienta GUI web de Tivoli Netcool/OMNIbus para pasar valores de columna de un suceso a un script CGI. El script puede llamar a la aplicación web correspondiente con estos parámetros.

Las ventanas predeterminadas que se componen de varias aplicaciones web, como Vista de estado de red, por ejemplo, no se pueden abrir utilizando una URL.
En seguridad podría ser la manipulación de la URL.

Ventajas y desventajas

Ventajas:

- Servidor controla los accesos a sus datos protegiendo asi la integridad del sistema y facilitando la actualización de los datos.
- Escalabilidad.
- Fácil mantenimiento: al estar distribuidas las funciones y responsabilidades entre varios ordenadores independientes, es posible reemplazar, reparar, actualizar, o incluso trasladar un servidor, mientras que sus clientes no se verán afectados por ese cambio. Esta independencia de los cambios también se conoce como encapsulación.

Desventajas:

- Congestión: Cuando una gran cantidad de clientes envían peticiones simultáneas al mismo servidor, puede ser que cause muchos problemas para éste.
- El paradigma de C/S clásico es menos robusto que una red P2P.
- Se necesita software y hardware específico para que el servidor pueda satisfacer el trabajo. Por supuesto, esto aumentará el coste.
- El cliente no dispone información de los recursos que puedan existir en el servidor.

Conclusiones

Algunas de estas técnicas traen beneficiosos y también algunos fallos en la seguridad si mal lo manejas. El cliente y servidor siempre habrá una brecha donde un agente malicioso puede descubrir y explotar esas vulnerabilidades. La manera en que trabaja el cliente y el servidor es muy tedioso a la manera de trabajarlo.

Bibliografías

- http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/marquez_a_bm/capitulo5.pdf
- https://www.ibm.com/support/knowledgecenter/es/SSSHRK_3.9.0/com.ibm.networkmanagerip.doc_3.9/itnm/ip/wip/admin/reference/nmip_adm_parameterfortopoviz.html