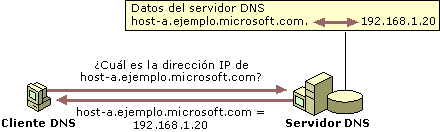
**SEGUNDO PARCIAL PRÁCTICO**

**DNS (Domain Name System – Sistema de Nombres de Dominio)**

Sistema para asignar nombres a equipos y servicios de red que se organiza en una jerarquía de dominios. La asignación de nombres DNS se utiliza en las redes TCP/IP para localizar equipos y servicios con nombres sencillos.

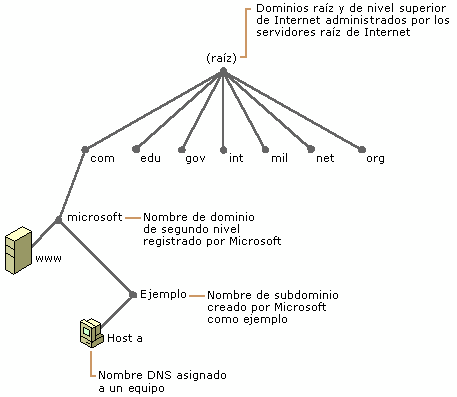
El gráfico siguiente muestra un uso básico de DNS, consistente en la búsqueda de la dirección IP de un equipo basada en su nombre.

En este ejemplo, un equipo cliente consulta a un servidor, preguntando la dirección IP de un equipo configurado para utilizar hosta.ejemplo.microsoft.com como nombre de dominio.

Como el servidor puede utilizar la base de datos local para responder la consulta, contesta con una respuesta que contiene la información solicitada, un registro de recursos de direcciones host (A) que contiene la información de dirección IP para host-a.ejemplo.microsoft.com.

**Nombres de dominio DNS**

* Espacio de nombres de dominio DNS, que especifica una jerarquía estructurada de dominios utilizados para organizar nombres.
* Registros de recursos, que asignan nombres de dominio DNS a un tipo específico de información de re curso para utilizar cuando se registra o se resuelve el nombre en el espacio de nombres.
* Servidores DNS que almacenan y responden a las consultas de nombres para los registros de recursos.
* Clientes DNS que consultan a los servidores para buscar y resolver nombres de un tipo de registro de recursos especificado en la consulta.



**Descripción del espacio de nombres de dominio DNS**

Se basa en el concepto de un árbol de dominios con nombre. Cada nivel del árbol puede representar una rama o una hoja del árbol. Una rama es un nivel donde se utiliza más de un nombre para identificar una colección de recursos con nombre. Una hoja representa un nombre único que se utiliza una vez en ese nivel para indicar un recurso específico.

Los clientes y los servidores DNS usan las consultas como el método fundamental para resolver los nombres en el árbol como información específica de los tipos de recurso.

En el proceso de resolución de un nombre, los servidores DNS funcionan frecuentemente como clientes DNS, consultando a otros servidores para resolver completamente un nombre consultado.

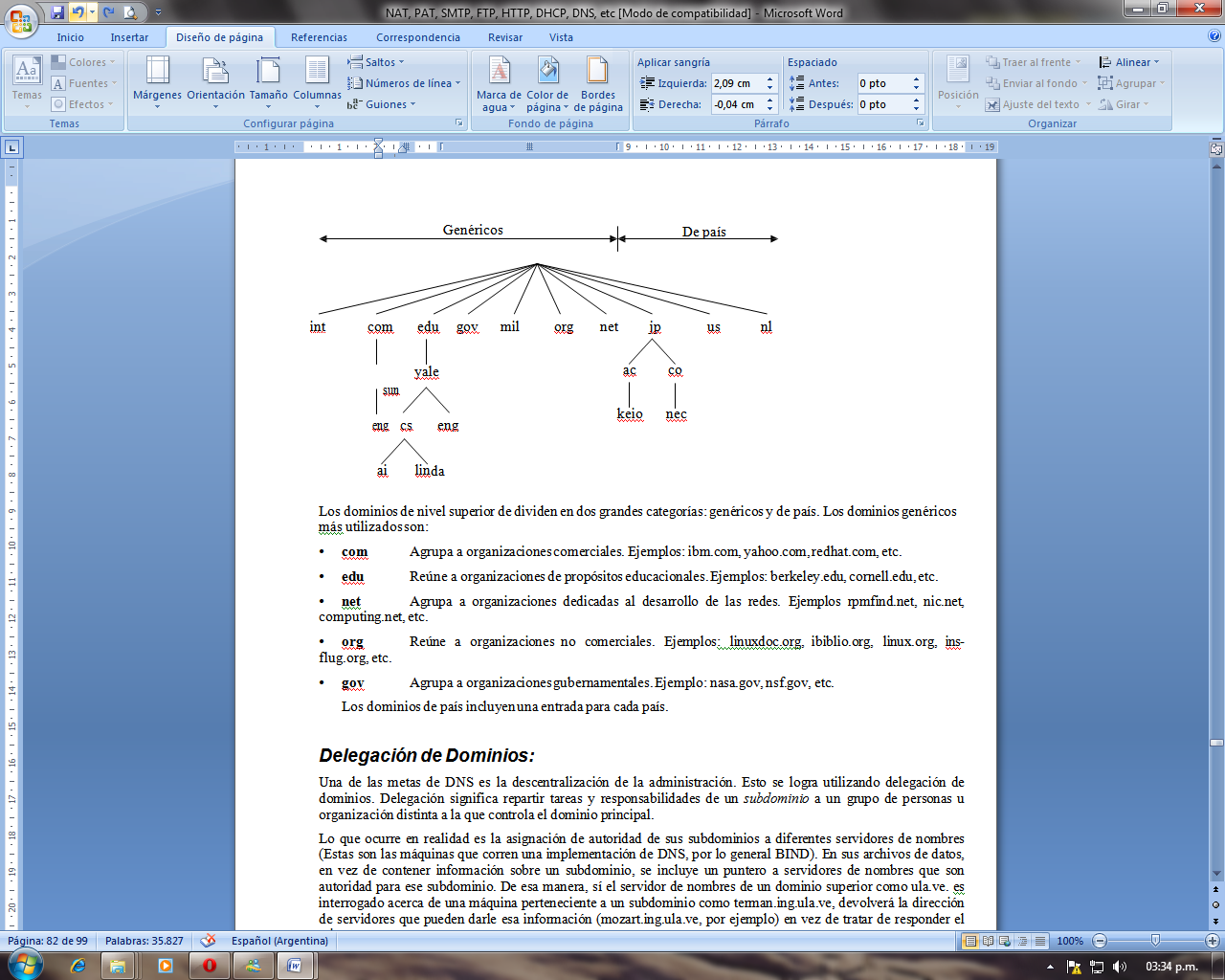
*Cómo se organiza el espacio de nombres de dominio DNS*

Hay cinco formas, según el nivel y la forma en que se utiliza normalmente un nombre. Por ejemplo, el nombre de dominio DNS registrado para Microsoft (microsoft.com.) se conoce como dominio de segundo nivel, ya que tiene dos partes (etiquetas) que indican que se encuentra dos niveles por debajo de la raíz o la parte superior del árbol. En los nombres se utilizan puntos para separar las etiquetas.

Otros niveles:

|  |  |  |
| --- | --- | --- |
| **TIPO DE NOMBRE** | **DESCRIPCIÓN** | **EJEMPLO** |
| Dominio Raíz | Parte superior del árbol que representa un nivel sin nombre; a veces se muestra como dos comillas vacías (“”), que indican un valor nulo. Cuando se utiliza en un nombre de dominio DNS, empieza con un punto (.) para designar que se encuentra en la raíz. En este caso, el nombre de dominio DNS se considera completo e indica una ubicación exacta en el árbol de nombres (Nombres de dominio completos - FQDN) | Un solo punto (.) o un punto usado al final del nombre, como: ejemplo.microsoft.com. |
| Dominio de Nivel Superior | Un nombre de dos letras que se utilizan para indicar un país o región, o el tipo de organización que usa un nombre. | “.com”, que indica un nombre registrado para usos comerciales o empresariales en Internet. |
| Dominio de Segundo Nivel | Nombres de longitud variable que un individuo u organización utiliza en Internet. Se basan en un dominio de nivel superior apropiado, según el tipo de organización o ubicación geográfica donde se utiliza el nombre. | “mocrosoft.com”, que es el nombre de dominio de segundo nivel registrado para Microsoft por el registrador de nombres de dominio DNS de Internet. |
| Subdominio | Nombres adicionales que puede crear una organización y se derivan del nombre de dominio registrado de segundo nivel. Incluyen nombres agregados para desarrollar el árbol de nombres de dominio DNS en una organización y que la divide en departamentos o ubicaciones geográficas. | “ejemplo.microsoft.com”, que es un subdominio ficticio asignado por Microsoft para utilizarlo en nombres de ejemplo de documentación |
| Nombre de Recurso o de Host | Nombres que representan una hoja en el árbol DNS de nombres e identifican un recurso específico. Normalmente, la etiqueta de la izquierda identifica un equipo específico en la red. Por ejemplo, si un nombre se utiliza en un RR de host (A), se utiliza para buscar la dirección IP del equipo según su nombre de host. | "hosta.ejemplo.microsoft.com.", donde la primera etiqueta ("host-a") es el nombre de host DNS de un equipo específico en la red |

La estructura de la base de datos del DNS posee una forma jerárquica tipo árbol invertido donde cada nodo representa un dominio. Los nodos a su vez pueden poseer varios subnodos hijos que constituyen subdominios en el DNS. Los nodos que no poseen hijos pueden verse como los nombres de los hosts que pertenecen al dominio definido por el nodo padre.



Los dominios de nivel superior se dividen en dos grandes categorías: genéricos y de país. Los genéricos más utilizados son:

• com: Agrupa a organizaciones comerciales. Ejemplos: ibm.com, yahoo.com, redhat.com, etc.

• edu: Reúne a organizaciones de propósitos educacionales. Ejemplos: berkeley.edu, cornell.edu, etc.

• net: Agrupa a organizaciones dedicadas al desarrollo de las redes. Ejemplos rpmfind.net, nic.net, computing.net, etc.

• org: Reúne a organizaciones no comerciales. Ejemplos: linuxdoc.org, ibiblio.org, linux.org, insflug.org, etc.

• gov: Agrupa a organizaciones gubernamentales. Ejemplo: nasa.gov, nsf.gov, etc.

Los dominios de país incluyen una entrada para cada país.

**Delegación de Dominios:**

Es el proceso de repartir tareas y responsabilidades de un subdominio a un grupo de personas u organización distinta a la que controla el dominio principal. Se utiliza para descentralizar la administración.

Lo que ocurre en realidad es la asignación de autoridad de sus subdominios a diferentes servidores de nombres. En sus archivos de dato, en vez de contener información sobre un subdominio, se incluye un puntero a servidores de nombres que son autoridad para ese subdominio. Así, si el servidor es interrogado acerca de una máquina perteneciente a un subdominio, devolverá la dirección de servidores que pueden darle esa información en vez de tratar de responder el mismo.

**Interpretar un nombre de dominio DNS**

Ejemplo de un nombre de dominio completo (FQDN): host-a.ejemplo.microsoft.com. Se lee de izquierda a derecha, moviéndose desde su información más específica a su grupo de información más general. Niveles de dominio DNS:

1. El dominio "ejemplo", corresponde a un subdominio donde el nombre de equipo "host-a" está registrado para su uso.

2. El dominio "microsoft", corresponde al dominio principal que es la raíz del subdominio "ejemplo".

3. El dominio "com", corresponde al dominio de nivel superior designado para ser usado por empresas u organizaciones comerciales y es la raíz del dominio "microsoft".

4. El punto final (.),es un carácter de separación estándar que se utiliza para calificar el nombre de dominio DNS completo en el nivel raíz del árbol del espacio de nombres DNS.

**Servidores de nombres**

Programas encargados de agrupar y mantener disponible la información asociada a un espacio de nombres de dominio.

*Se clasifican en*:

* Primarios: toma los datos de las zonas para las que está autorizado desde ficheros presentes en la máquina donde se ejecuta.
* Secundarios: toma los datos de las zonas para las que está autorizado a través de otros servidores autorizados para la zona (primarios generalmente). El proceso de actualización de la información de una zona en los servidores secundarios se conoce como transferencia de zona.

**Cómo funcionan las consultas DNS**

Cuando un cliente DNS necesita buscar un nombre que se utiliza en un programa, consulta los servidores DNS para resolver el nombre. Cada mensaje contiene tres grupos de información:

. Un nombre de dominio DNS especificado, indicado como el nombre de dominio completo (FQDN)

. Un tipo de consulta especificado, que puede establecer un tipo de registro de recursos por tipo o un tipo especializado de operación de consulta

. Una clase especificada para el nombre de dominio DNS. Para servidores DNS de Windows, esto se debería especificar siempre como la clase Internet (IN).

Los servidores DNS pueden contener los siguientes *tipos de registros de recursos*:

* A (host): resuelve nombres de host name a IP. Contiene la información de asignaciones de nombre a dirección IP, que se utiliza para asignar un nombre de dominio DNS a una dirección IP de host en la red. También se conocen como registros de host.
* NS (name server): identifica el DNS server para cada zona. Designa los nombres de dominio DNS de los servidores que tienen autoridad en una zona determinada o que contienen el archivo de zona de ese dominio.
* CNAME (canonical name): resoluciones desde nombres de host a nombres de host. Permite proporcionar nombres adicionales a un servidor que ya tiene un nombre en un registro de recursos A. También se conocen como registros de alias.
* MX (mail exchanger): el server de mail. Especifica el servidor en el que las aplicaciones de correo electrónico pueden entregar correo.
* SOA (Start Authority): el primer registro en cualquier archivo de zona. Indica el punto de partida o el punto original de autoridad para la información almacenada en una zona. Es el primero que se crea cuando se agrega una zona nueva. Contiene también varios parámetros que utilizan otros equipos que emplean DNS para determinar cuánto tiempo utilizarán la información de la zona y con cuánta frecuencia hay que realizar actualizaciones.
* PTR (pointer): resuelve IP a nombres de host. Se utiliza en una zona de búsqueda inversa para designar una asignación inversa de una dirección IP de host a un nombre de dominio DNS de host.
* SRV (service): resuelve nombres de servers que proveen servicios. Lo registran los servicios para que los clientes puedan encontrar un servicio mediante DNS. También se conocen como registros de ubicación de servicio.

**Resolución**

Las consultas DNS se resuelven de diferentes formas:

* Consulta recursiva: aquella realizada a un servidor DNS en la que el cliente DNS solicita al servidor que proporcione una respuesta completa a la consulta. El servidor DNS comprueba la zona de búsqueda directa y la caché para encontrar una respuesta a la consulta.
* Consulta iterativa: aquella efectuada a un servidor DNS en la que el cliente DNS solicita la mejor respuesta que el servidor puede proporcionar sin buscar ayuda adicional de otros servidores. El resultado suele ser una referencia a otro servidor de nivel inferior en el árbol.

*Proceso de consulta recursiva:*

* Parte 1: el solucionador local: En un programa del equipo local se utiliza un nombre de dominio DNS. A continuación, la solicitud se pasa al servicio Cliente DNS para proceder a su resolución mediante la información almacenada en la caché local. Si el nombre consultado se puede resolver, se responde a la consulta y el proceso finaliza.

La caché del solucionador local puede incluir información de nombres obtenida de dos orígenes posibles:

. Si un archivo Hosts está configurado localmente, las asignaciones de nombre a dirección de host de ese archivo se cargan con anterioridad en la caché cuando se inicia el servicio Cliente DNS. Los registros de recursos obtenidos en las respuestas de consultas DNS anteriores se agregan a la caché y se mantienen durante un período.

. Si la consulta no coincide con una entrada de la caché, el proceso de resolución continúa con la consulta del cliente al servidor DNS para resolver el nombre.

* Parte 2: consultar un servidor DNS

El cliente consulta un servidor DNS preferido. Cuando el servidor DNS recibe una consulta, primero comprueba si puede responder la consulta con autoridad. Si el nombre consultado coincide con un registro de recursos, el servidor responde con autoridad y usa esta información para resolver el nombre consultado. Si no existe información, comprueba si puede resolver el nombre mediante la información almacenada en la caché local de consultas anteriores. Si aquí se encuentra una coincidencia, responde con esta información. Si el nombre consultado no encuentra una respuesta coincidente, el proceso de puede continuar hasta resolver completamente el nombre, implicando la asistencia de otros servidores DNS.

Para que el servidor DNS realice la recursión correctamente, primero necesita información de contacto útil acerca de los otros servidores DNS en el espacio de nombres de dominio DNS. Esta información se proporciona en forma de sugerencias de raíz, una lista de los registros de recursos preliminares que puede utilizar el servicio DNS para localizar otros servidores DNS que tienen autoridad para la raíz del árbol del espacio de nombres de dominio DNS. Los servidores raíz tienen autoridad para el dominio raíz y los dominios de nivel superior en el árbol del espacio de nombres de dominio DNS. Mediante el uso de las sugerencias de raíz, un servidor DNS puede completar el uso de la recursión.

Aunque puede usar muchos recursos, tiene algunas ventajas en el rendimiento para el servidor DNS. Por ejemplo, durante el proceso de recursión, el servidor DNS que realiza la búsqueda obtiene información acerca del espacio de nombres de dominio DNS, que se almacena en la caché y se puede utilizar para ayudar a acelerar la obtención de la respuesta de consultas subsiguientes que la utilizan o concuerdan con ella.

Respuestas de consultas alternativas

Se supone que el proceso finaliza con una respuesta positiva devuelta por el cliente. Sin embargo, también pueden devolver:

• Una respuesta con autoridad: respuesta positiva devuelta al cliente y entregada con el bit de autoridad activado en el mensaje DNS para indicar que la respuesta se obtuvo de un servidor con autoridad directa para el nombre consultado

• Una respuesta positiva: puede estar formada por el registro de recursos consultado o por una lista de registros de recursos que se ajusta al nombre de dominio DNS consultado y el tipo de registro especificado en el mensaje de la consulta

• Una respuesta de referencia: referencia contiene registros de recursos adicionales no especificados por el nombre o el tipo en la consulta. Si el proceso de recursión no se admite, se devuelve al cliente este tipo de respuesta. Contiene datos adicionales como registros de recursos (RR) distintos de los del tipo consultado. Por ejemplo, si el nombre de host consultado era "www" y no se encontró ningún registro de recursos de dirección (A) para este nombre en esta zona pero, en su lugar, se encontró un registro de recursos de CNAME para "www", el servidor DNS puede incluir esa información cuando responda al cliente

• Una respuesta negativa: puede indicar que se encontró uno de los dos resultados posibles mientras el servidor intentaba procesar y resolver de forma recursiva la consulta completa:

* + Un servidor con autoridad informó de que el nombre consultado no existe en el espacio de nombres DNS.
  + Un servidor con autoridad informó de que el nombre consultado existe, pero no existen registros del tipo especificado para ese nombre.

Nota: Generalmente, se deshabilita el uso de la recursión en un servidor DNS cuando los clientes DNS se limitan a la resolución de nombres en un servidor DNS específico, o cuando no puede resolver nombres DNS externos y los clientes esperan migrar tras el error a otro servidor DNS para la resolución de estos nombres.

*Proceso de consulta iterativa*

Una solicitud iterativa informa al servidor DNS de que el cliente espera la mejor respuesta que el servidor DNS pueda proporcionar inmediatamente, sin entrar en contacto con otros servidores DNS.

Un servidor DNS responde al cliente en función de su propio conocimiento específico acerca del espacio de nombres, sin tener en cuenta los datos de los nombres que se están consultando. Por ejemplo, si un servidor DNS recibe una consulta de un cliente local, podría devolver una respuesta de su caché de nombres. Si el nombre consultado no está almacenado, podría proporcionar una lista de registros de recursos de dirección (A) y de servidor de nombres (NS) para otros servidores DNS que estén más cerca del nombre consultado por el cliente (reenviadores).

Cuando se proporciona una referencia, el cliente DNS asume la responsabilidad de continuar efectuando consultas iterativas a otros servidores DNS configurados para resolver el nombre.

**Cómo funciona el almacenamiento en caché**

Cuando los servidores DNS procesan las consultas de los clientes, adquieren información acerca del espacio de nombres DNS, y la almacenan en la caché.

El almacenamiento aumenta el rendimiento de la resolución DNS para las consultas subsiguientes de nombres muy utilizados, y reduce sustancialmente el tráfico de las consultas relativas a DNS en la red.

Cuando los servidores DNS realizan consultas recursivas en nombre de clientes, almacenan en caché temporalmente los registros de recursos. Éstos contienen información obtenida de los servidores DNS que tienen autoridad para los nombres de dominio DNS aprendidos mientras se hacen las consultas. Cuando otros clientes realizan consultas nuevas que solicitan información de un registro de recursos que coincide con los registros de recursos almacenados en las caché, el servidor DNS puede utilizar esa información para responderlas.

Cuando la información se almacena en la caché, se aplica el valor Tiempo de vida (TTL) a todos los registros de recursos almacenados. Mientras no caduque, un servidor DNS puede seguir almacenando y utilizando el registro al responder a consultas de sus clientes que coincidan con estos registros de recursos. Al valor de los TTL se le asigna el TTL mínimo (predeterminado) que se utiliza en el registro de recursos de inicio de autoridad (SOA) de la zona. De forma predeterminada es de 3600 segundos (1 hora), pero se puede ajustar o se pueden establecer tiempos de vida individuales para cada registro de recursos.

**Búsqueda inversa**

Permite a los clientes utilizar una dirección IP conocida durante la búsqueda de un nombre y busca un nombre de equipo en función de su dirección.

Un problema observado es la diferencia en la forma en que el espacio de nombres DNS organiza e indexa los nombres, y cómo se asignan las direcciones IP. Para resolver este problema, en el estándar DNS se definió y reservó el dominio in-addr.arpa. Al crear el espacio, los subdominios se crean con el orden inverso de los números en la notación decimal con puntos de las direcciones IP.

Este orden inverso es necesario porque cuando se leen las direcciones IP de izquierda a derecha se interpretan al contrario. Cuando se lee una dirección IP de izquierda a derecha, se va desde su información más general a la más específica que contienen los últimos octetos. Con esta colocación, la administración de las ramas inferiores del árbol DNS in-addr.arpa se puede dejar a las organizaciones ya que se les asigna un conjunto de direcciones IP específicas o limitadas dentro de las clases de direcciones definidas en Internet.

Finalmente, el árbol del dominio in-addr.arpa requiere que se defina un tipo de registro de recursos adicional, el registro de recursos de puntero (PTR), que se utiliza para crear una asignación en la zona de búsqueda inversa que corresponde a un registro de recurso de dirección de host (A) con nombre para el nombre del equipo DNS de un host en su zona de búsqueda directa.

*Consultas inversas*

Se propusieron para buscar un nombre de host en función de su dirección IP. Utilizan una operación de consulta DNS no estándar y su uso se limita a algunas versiones anteriores de Nslookup.

En el caso de Windows 2000 Server, el servicio DNS reconoce y acepta los mensajes de consultas inversas y los responde con una respuesta de consulta inversa falsa. En los servidores DNS que ejecutan Windows NT Server 4.0, esta compatibilidad está disponible de forma predeterminada si el equipo servidor se ha actualizado con el Service Pack 4 o posterior.

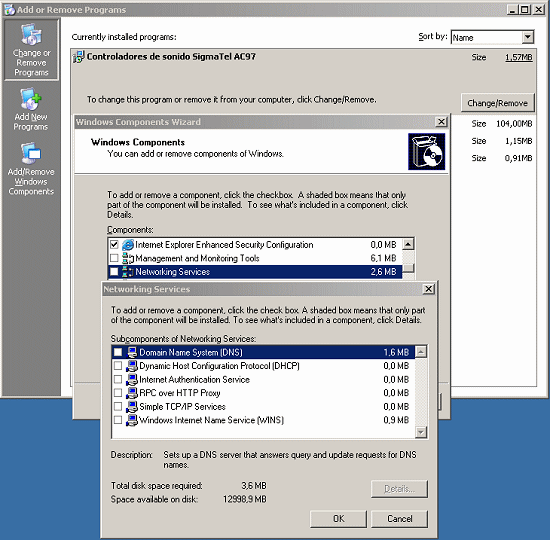
Nota:

• La configuración de registros de recursos de puntero (PTR) y zonas de búsqueda inversa para identificar hosts mediante consultas inversas constituye meramente una parte opcional de la implementación DNS estándar. No tiene por qué utilizar las zonas de búsqueda inversa, si bien algunas aplicaciones de red las necesitan, se utilizan para realizar comprobaciones de seguridad.

**INSTALAR Y CONFIGURAR DNS PARA INTRANET/INTERNET EN WINDOWS SERVER 2003**

Un servidor DNS nos va a servir para la resolución de nombres cuando queramos acceder a las Web que tenemos alojadas en nuestra intranet. Si el volumen de sitios Web es grande, instalando un DNS podremos utilizar una misma IP y tener multitud de webs en un mismo servidor IIS diferenciados por nombre. Si disponemos de IP suficientes esta opción nos servirá para acceder a los webs por un nombre y no por una dirección IP que son más difíciles de recordar.

La misma filosofía se aplica si el servidor lo tenemos en Internet, si instalamos un servidor DNS con una sola IP que nos proporcione nuestro proveedor podremos alojar multitud de dominios en un mismo servidor IIS.

**Instalación del servidor DNS.**

Para instalar un DNS Server:

• Inicio > Panel de control > Agregar o quitar programas

Una vez ejecutada esta opción aparece una ventana donde seleccionaremos:

• Agregar o quitar componentes de Windows

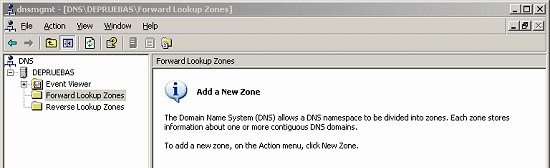
• Buscamos en la ventana que nos aparece la opción servicios de red, la seleccionamos y pulsamos el botón detalles

• De la nueva ventana que aparece seleccionamos la opción sistema de nombres de dominios (DNS), la seleccionamos y aceptamos todas las pantallas y pulsamos siguiente. Este proceso de instalación pedirá el CD de Windows debido a que necesita instalar varios componentes.

Una vez terminado el proceso de instalación podemos acceder al DNS desde:

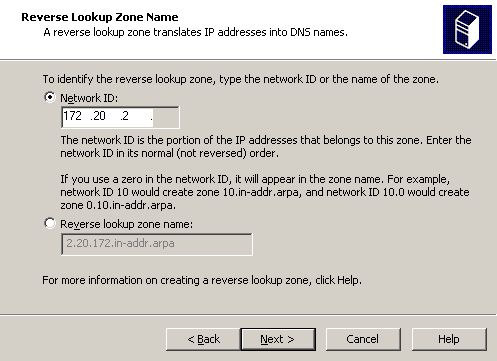
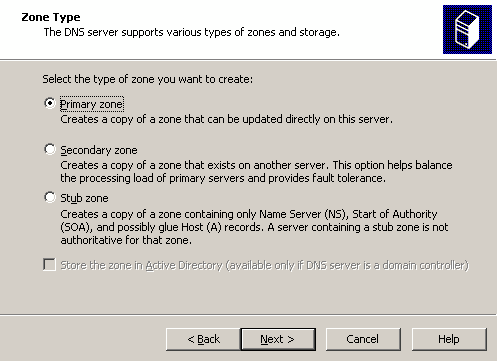
• Inicio > herramientas administrativas > DNS

Hacemos clic en DNS y aparecerá la siguiente figura, y lo tendremos que configurar.



**Configuración del servidor DNS.**

Para configurar el DNS y que nos resuelva las IP en nombres lo primero que deberemos crear es una zona de búsqueda inversa (dada una IP dice cuál es el dominio). Para realizar esta acción nos pondremos sobre esta rama de la consola del DNS y con el botón derecho seleccionamos nueva zona, nos aparece un asistente donde asignaremos:



• *Tipo de zona*: Primaria

• *Id. De red*: Será nuestra IP, para los ejemplos 172.20.2. Las siguientes pantallas las dejamos tal y como las pone el asistente.

Una vez creada la zona inversa tenemos que crear el nombre de domino que utilizaremos en nuestros webs. Para el ejemplo se crearon dos nombres ficticios "issdA" e "issdB". Se puede crear cualquier nombre que deseemos. Si se instala el servidor DNS en Internet, los nombres deben de ser reales y registrados en el InterNic que es la empresa final que otorga los dominios. En Internet se debe poner los nombres con el .com, .ar, etc … y no sólo el nombre como en la intranet.

Para crear los nombres de dominio pulsaremos con el botón derecho sobre zonas de búsqueda directa y seleccionamos nueva zona, aparece un asistente donde asignaremos:

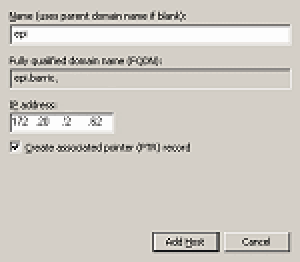
• *Tipo de zona:* Zonal principal. La zona secundaria se utiliza para añadir dominios que ya tienen una zona principal creada en otro servidor DNS, de esta forma cuando creamos una zona secundaria todos los registros que tengamos en la zona principal se replicarán a ésta.

• *Nombre de zona:* nombre de dominio, por ejemplo, "issdA".

Y las siguientes pantallas las dejamos como las pone el asistente. Para crear el segundo nombre de domino repetimos los pasos.

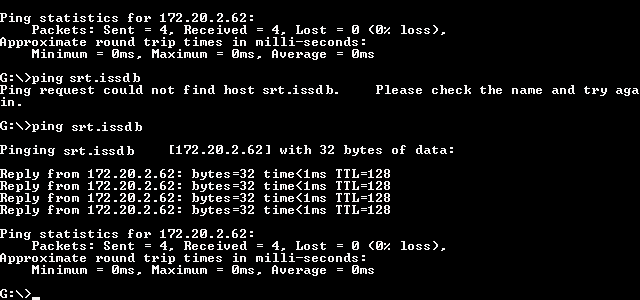
Con ésto ya tenemos dado de alta nuestro nombre de dominio para la intranet. Ahora queda crear los nombres de los subdominios con los que crearemos la dirección Web, es decir, accederemos a nuestro Web a través de

Nombre\_Subdominio.Nombre\_dominio

Para crear los subdominios pulsamos sobre la nueva zona creada (issdA)

Vemos que a la derecha nos aparecen los datos de este domino, pulsamos con el botón derecho sobre el nombre de dominio, seleccionamos #Host nuevo (A) y aparecerá una ventana:

Pondremos el nombre de subdominio, es el ejemplo "srt", y la IP que tiene asignada, normalmente la de la máquina y lo añadimos.

Si abrimos una consola (cmd.exe) y probamos hacer ping al dominio de intranet que acabamos de crear (srt.issdB) veremos que nos responde afirmativamente.

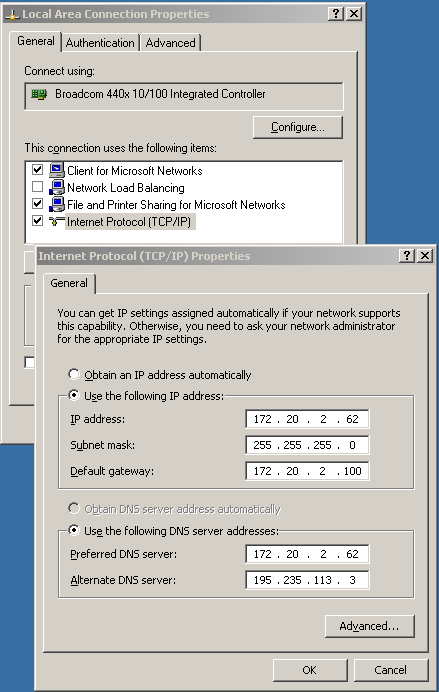
Si queremos crear más, solo tenemos que repetir los pasos.

Éstos serán los nombres que utilizaremos para acceder a los webs de la siguiente forma:

. <http://srt.issdB>

. <http://blas.issdA>

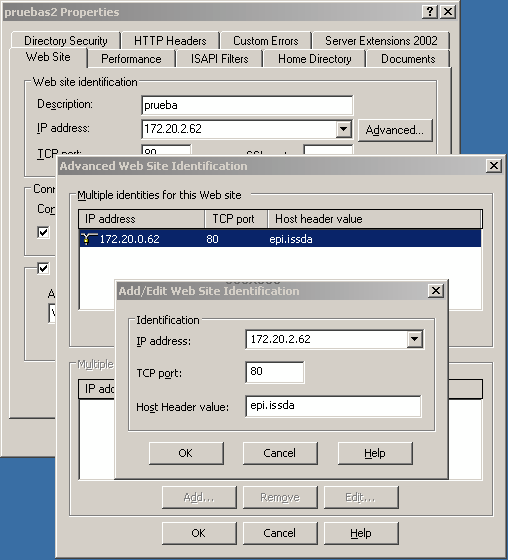
Ahora queda enlazar estos dominios con el IIS.



Un paso previo que es hacer accesible nuestro DNS a nuestro PC, es decir, en las opciones de red tenemos que poner nuestro DNS para que resuelva los nombres. Para realizar esta acción vamos a:

• Inicio > panel de control > conexión de red > conexión de área local

Pulsamos el botón propiedades de la ventana que aparece. Seleccionamos Protocolo de Internet (TCP/IP). Donde pone Servidor DNS preferido ponemos la IP de nuestro DNS, que suele ser la IP de la máquina, en el ejemplo seria 172.20.2.62

****

**Configuración del IIS.**

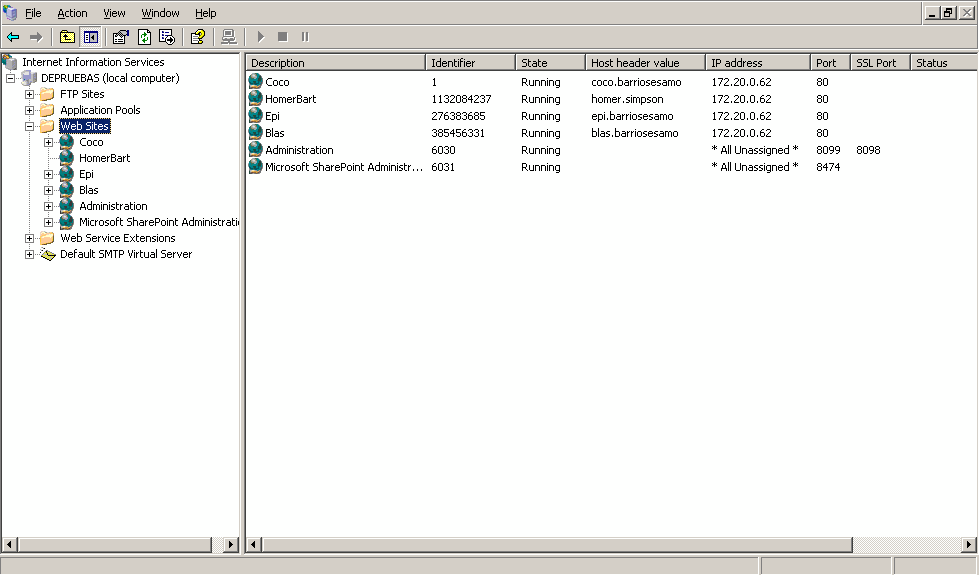
Ya tenemos el DNS funcionando, enlazado con nuestro servicio TCP/IP y los dominios dados de alta en el DNS, ahora nos queda configurar el IIS para que acepte estos nombres de dominio en lugar de nuestra IP y poder crear las webs que necesitemos sin necesidad de que todas estén dentro del wwwroot.

• Una vez creado el Web pulsamos sobre el botón derecho y seleccionamos propiedades.

• En la pantalla que aparece donde pone la dirección IP pulsamos el botón avanzadas y sobre la nueva pantalla seleccionamos la IP.

• Sale una ventana donde le pondremos la IP, el puerto y las cabeceras.

• Donde pone valor del encabezado host tenemos que poner nuestro nombre de domino, por ejemplo pondremos epi.issdA.

Podemos poner tantos webs como queramos y repetir este proceso por cada Web nuevo que deseemos instalar.

Muestra de cómo podría quedar con varios webs instalados:

Con este último paso ya tenemos todo configurado y sólo queda probarlo. Para eso ir al Internet Explorer y ponerhttp://epi.issdA y debería aparecer la pagina principal del Web.

**FTP (File Transfer Protocol – Protocolo de Transferencia de Archivos)**

Es el protocolo más utilizado dentro de la Internet y las redes en general para la transferencia de archivos entre nodos.

* Permite la transferencia de información en ambos sentidos
* Se establece una conexión TCP con el servidor remoto

El propósito es centralizar los archivos de uso común en una red, disponiéndolos en un servidor, donde se conectarán los usuarios para así poder trabajar sobre ellos. De acuerdo con los permisos que tengan podrán:

• Descargar los archivos

• Subir archivos

• Leerlos de forma remota

• Modificarlos de forma remota (en línea)

**Características de FTP**

• Acceso Interactivo: La mayoría de los sistemas operativos que lo soportan ofrecen comandos para efectuar las operaciones.

• Servicios de capa Presentación: solicitar la transferencia de los datos en forma de caracteres o en binario, y qué tipo de familia de caracteres se utilizará.

• Autenticación: diferentes tipos de acceso a diferentes tipos de usuario.

**Funcionamiento de FTP**

Para la transferencia de los datos utiliza una arquitectura Cliente-Servidor, usando dos puertos para manejar la conexión, ya que los clientes por sí mismos no pueden manejar toda la funcionalidad necesaria. Es por esto que se corren dos procesos en el cliente y dos en el servidor: el 20 para la transferencia de los datos en sí mismos y el 21 para el control de la comunicación.

* Conexión de Control:
  + Puerto 21
  + Conexión persistente
  + Permanece abierta TODO el tiempo hasta que el usuario se desconecta
  + Permite enviar comandos al servidor
  + El servidor mantiene el ESTADO del usuario
* Conexión de datos:
  + Puerto 20
  + Conexión no persistente
  + Se inicia cada vez que el servidor recibe un comando de transferencia de archivo desde el cliente
  + Se libera cuando se finaliza la transmisión del archivo

*Forma de establecer las comunicaciones*: El cliente asigna un puerto local sin uso, y envía la petición de conexión a un puerto conocido del servidor (20). Por medio de esta conexión, se mantienen en contacto coordinando la asignación de los puertos para la transferencia de archivos y mandar los comandos y mensajes. Así, cuando se solicite una transferencia de archivos, el cliente se dará a conocer en el puerto 21.

**Clientes FTP**

Ofrecen como mínimo un ambiente de línea de comandos, que puede ser accedido desde el prompt del sistema operativo, ya sea éste Windows (DOS), Linux, UNIX u otro.

**Servidores FTP**

Son programas más complejos, y su implementación dependerá del software. Características básicas que deberá satisfacer respecto a su configuración:

• Manejar una base de datos de usuario

• Cada usuario (o grupo) utilizará permisos (Lectura, Escritura, Modificación)

• Cada usuario (o grupo) podrá acceder a un set de archivos, esto es, a una estructura particular de directorios

• Se podrá configurar la opción de utilizar usuarios “Anónimos”

• Se podrá configurar la tasa de transferencia que se permita a usuarios (o grupo) en particular.

• Se podrá configurar la cantidad de sesiones simultáneas que soportará el servidor.

• Se podrá configurar la cantidad de sesiones simultáneas que se permitirán a cada cliente.

**Tipos de servicios:**

* FTP público (anonymous)
* FTP privado (nombre de usuario y contraseña)

**SERVIDOR DE CORREO**

Es una de las aplicaciones de uso más común, que contribuye a la comunicación entre las personas. Se distingue de las demás aplicaciones porque los usuarios trabajan sin conexión, lo que significa que las personas que se comunican no tienen por qué estar simultáneamente conectadas.

Se asemeja al correo postal tradicional. Cuando queremos mandar una carta escrita en papel a alguien, la redactamos tranquilamente en casa, la metemos en un sobre timbrado, escribimos la dirección del destinatario y la llevamos a una oficina postal. A partir de ahí, los servicios de correo se encargan de que la carta llegue a la oficina postal más cercana a la dirección del destinatario. Por último, un cartero intenta hacer llegar la carta a la dirección del destinatario. Si no la encuentra, la carta volverá al remitente, y si la encuentra, se pondrá en su buzón. El destinatario la leerá cuando quiera.

## Servicios básicos (funciones)

* Composición = proceso de crear mensajes y respuestas
* Transferencia = se envían mensajes al destino mediante el establecimiento de una conexión
* Generación del informe = se informa qué ocurrió con el mensaje
* Visualización = permite leer el correo (conversiones)
* Disposición = lo que realiza el usuario destino con el mensaje

Otras funcionalidades habituales:

* Buzones de correo
* Listas de correo
* Prioridad
* Cifrado

## Características avanzadas

* Reenvío automático de correo
* Manejo de listas de correo (grupos)
* Destinatarios alternativos
* Manejo de prioridad, seguridad

## DIRECCIÓN DE CORREO ELECTRÓNICO

Está formada al menos por el nombre de la maquina que maneja el correo del destinatario, y un identificador de usuario reconocido por esa máquina. Tienen el siguiente formato: nombre@maquina.dominio

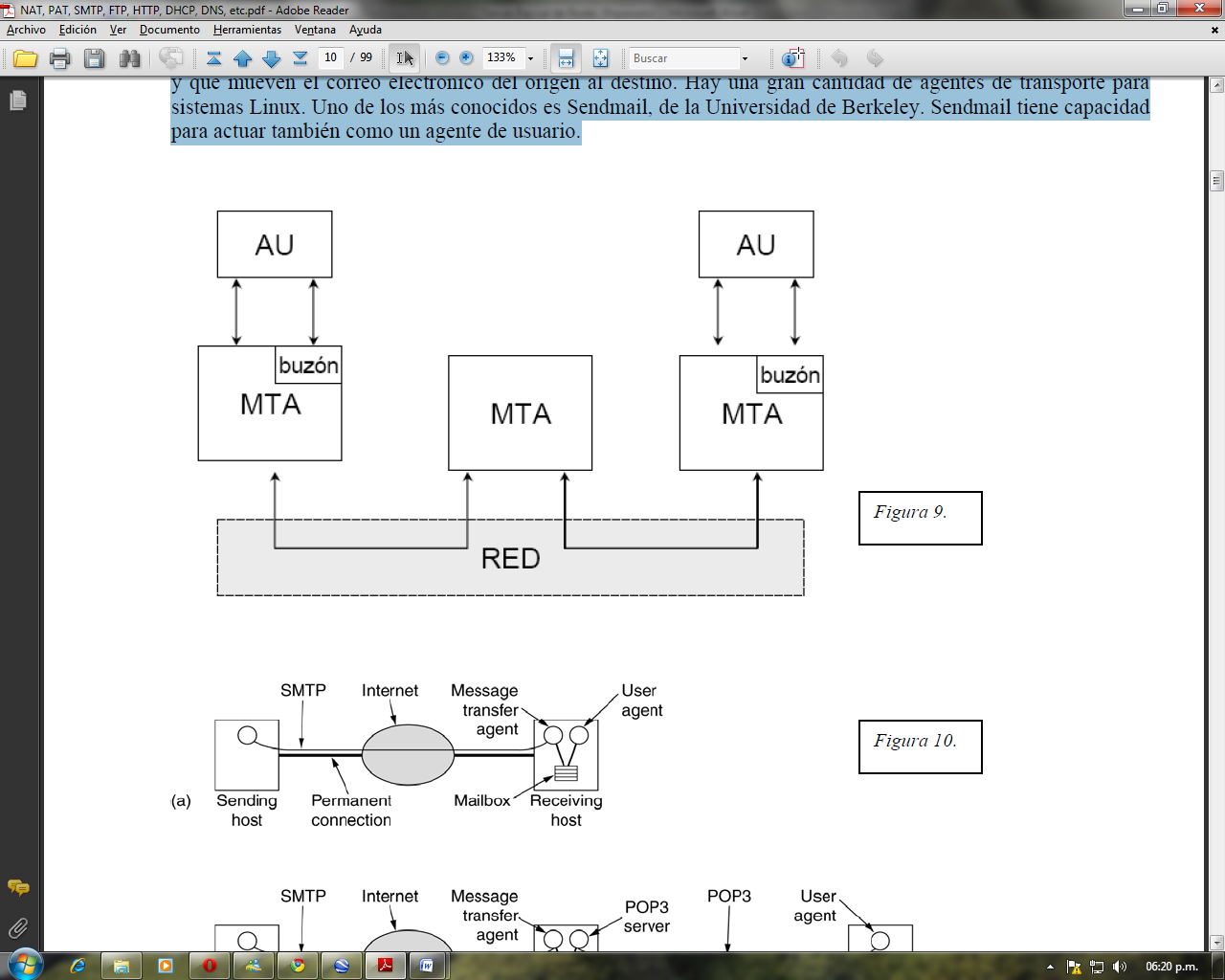
El arroba denomina a la preposición inglesa "at", que significa "en". Por “nombre” es el de una persona, o algún identificador de esa persona, conocido por la maquina perteneciente al dominio indicado.

Pueden expresarse de la forma: descripcion1<nombre@maquina.dominio>descripcion2

Las descripciones son ignoradas por los servicios postales, pero podría dárseles alguna utilidad, aunque no se recomienda. Dado que no se indica la ruta hasta la máquina de destino, sino el nombre (único) de dicha maquina, a esta notación se la suele llamar dirección absoluta.

Existe una notación especial para especificar la ruta de relays que obligatoriamente debe seguir el correo electrónico hasta llegar a la dirección del destinatario. Es: <@maquinaA.dominioA,@maquinaB.dominioB:nombre@dominioC.maquinaC>

La dirección anterior denota como destinatario al identificador de usuario nombre en la maquina C, indicando que se debe llegar a la maquina C a través de maquinaA y maquinaB, en ese orden. Este tipo de dirección se suele llamar una dirección Route-addr, de ruta (Route) y dirección (Address).



**ARQUITECTURA**

Los sistemas de correo electrónico están organizados en dos subsistemas: los *agentes de usuario de correo* (AU - mail User Agent, MUA) y los *agentes de transporte de correo* (mail Transport Agent, MTA).

Estos agentes se clasifican en:

• De distribución: utilizando el protocolo SMTP (Simple Mail Transfer Protocol) o SMTP extendido (ESMTP)

• De entrega final: permite al usuario gestionar su correo a través de una máquina remota, utilizando los protocolos POP3 e IMAP

**Agente de usuario (AU ó MUA)**

Son programas locales que aceptan una variedad de comandos para componer, recibir, y contestar mensajes. Algunos funcionan a base de comandos de un carácter desde el teclado, y otros asocian a cada uno de estos comandos un icono en la pantalla

• Es un programa (“lector de correo”) más o menos amigable.

• Comandos de una línea

• Interfaz gráfico con iconos

*Funcionalidad:*

• Enviar mensajes

• Componer mensaje

• Responder a mensajes

• Leer mensajes

• Manipular buzones

*Envío de correo electrónico:*

• Dirección destino

• Internet: xx@it.uc3m.es

• Lista (local, remota)

• Mensaje:

• Fichero(s) a transmitir, o entrar en editor.

• Otros: prioridad, seguridad, …

*Lectura del correo electrónico:*

• Busca en el buzón del usuario

• Anuncia número de mensajes

• Breve descripción de cada uno

• Admite comandos

**Agentes de transporte de correo (MTA)**

Son daemons del sistema que operan en segundo plano y que mueven el correo electrónico del origen al destino..

**BUZONES DE CORREO ELECTRÓNICO**

Es un lugar donde se van depositando el email que recibe una persona, y se almacena habitualmente en un fichero del ordenador donde el agente de transporte ha determinado que debe depositar los mensajes para el destinatario de los mismos. El ordenador en el que se colocan debe estar conectado a Internet.

**Acceso a buzones de correo electrónico**

El destinatario desea leer los mensajes que hay almacenados en el buzón. Si el buzón está en una maquina Linux, y el destinatario tiene acceso directo a esa máquina, puede acceder directamente a su buzón.

Lo habitual es que el destinatario tenga un acceso restringido al ordenador donde está depositado su buzón. Entonces puede recurrir a dos protocolos que permiten acceder a los buzones, POP3 e IMAP. Ambos requieren la presentación de un nombre de usuario y una contraseña. Una vez se ha conseguido el acceso, el comportamiento es diferente.

## Protocolos de entrega final de usuario

* *POP3 (Post Office Protocol*): permite que un usuario establezca una sesión, la termine, obtenga mensajes y los borre. Consiste en texto ASCII y se asemeja a SMTP. El objetivo es obtener correo electrónico del buzón remoto y almacenarlo en la máquina local del usuario para su lectura posterior.

El inconveniente es que, si se ha recibido un mensaje largo, habrá que esperar a que el agente lo traiga desde el buzón para poder ver de qué trata. Este es un problema de cómo se usa, no del protocolo en sí mismo.

Es un protocolo de transferencia, no permite editarlo.

* *IMAP (Interactive Mail Access Protocol*): La idea es que el servidor mantenga un depósito central al que puede accederse desde cualquier máquina. Se puede acceder, editar, leer, responder, etc. A diferencia del POP3, no copia el correo electrónico en la máquina personal del usuario. Es más potente y eficaz, pero su uso es conveniente cuando el costo de la conexión no va en función del tiempo y podemos estar leyendo los mensajes mientras estamos conectados. La gran ventaja es que los mensajes están siempre en el servidor.

Cuando el agente se conecta, obtiene la lista de las cabeceras de los mensajes y las muestra al usuario, quién puede ver asunto y remitente, permitiéndole leer selectivamente los mensajes y no tener que esperar a traer mensajes largos.

Otras características:

* + Puede mostrar los mensajes de llegada en función de cualquier atributo.
  + Capacidad de establecer filtros, que son reglas que se consultan cuando llega el correo electrónico.
  + Permite manejar múltiples buzones auxiliares en el mismo servidor.
  + Capacidad de instalar un daemon de vacaciones, que es un programa que examina cada mensaje de entrada y envía al transmisor una respuesta insípida con alguna información

## ENVÍO DE CORREO ELECTRÓNICO

Para enviar un mensaje, el usuario debe proporcionar el mensaje, la dirección de correo del destinatario y otras opciones como la prioridad o el nivel de seguridad.

El cuerpo de mensaje puede redactarse con un editor de textos independiente o con un editor incorporado en el agente de usuario. La dirección debe tener el formato expuesto. Una vez hecho esto, le pedimos al agente de usuario que envié el mensaje. Entonces, añade una cabecera y contacta con un agente de transporte para que lo transmita. Éste, que puede estar en otro ordenador distinto, usa algunos campos de la cabecera del mensaje para construir el sobre, e intenta transmitir el mensaje al agente de transporte donde se encuentra el buzón del destinatario. Si lo consigue, el agente de reparto eliminara el sobre del mensaje y lo depositara en el buzón del usuario destinatario. Si no lo consigue, el agente de transporte mantiene una cola de mensajes salientes donde se encuentran los que no se han podido enviar. Periódicamente procesa la cola e intenta enviarlos. Si pasan varias horas sin conseguir enviar un mensaje, se manda un aviso al remitente avisando de la situación.

## Estructura de un mensaje

Se compone de dos partes: un sobre y un mensaje. El sobre encapsula el mensaje. Contiene toda la información necesaria para transportar el mensaje: dirección de destino, prioridad y nivel de seguridad. Los agentes de transporte lo usan para enrutar el mensaje (igual que las oficinas postales).

Contenido:

• Cabecera: información de control para AU

• Cuerpo del mensaje: información para transportar el mensaje (MTA)

## MIME – EXTENSIONES MULTIPROPÓSITO DE CORREO INTERNET

* Inicios del correo 🡪 Sólo texto en inglés y en ASCII
* En la actualidad 🡪 TODO tipo de información, mediante nuevos MTA extendidos

La idea es continuar usando el formato primitivo de las cabeceras, pero agregando una estructura al cuerpo del mensaje para que contenga diversas partes y definiendo reglas para codificar con ASCII de 7 bits los mensajes que no tengan este formato. Al no cambiar las cabeceras, todos los mensajes pueden enviarse usando los agentes de transporte y protocolos de correo electrónico existentes. Todo lo que hay que cambiar son los agentes de usuario para que interpreten correctamente las cabeceras, algo que pueden hacer los usuarios mismos.

Lo que hace es agrupar el texto de a 7 bits. Arma el carácter que corresponda al ASCII y lo transmite. Cuando lo recibe, junta todo y lo extiende a 8 bits, por eso es tan importante la versión.

## MIME Seguro (Secure MIME, SMIME)

Cuando un mensaje de correo electrónico se envía entre dos sitios distantes, generalmente transitara por docenas de maquinas en el camino. Cualquiera de estas puede leer y registrar el mensaje para un uso posterior. La confidencialidad es inexistente, alguien podría modificar información o hacerse pasar por otra persona, con lo que tampoco podemos estar seguros de la autenticidad de la información.

La única forma de resolver ambos problemas es mediante el uso de la criptografía. Los dos protocolos criptográficos fundamentales empleados en la seguridad de Internet son SMIME para el correo electrónico y el SSL para la Web.

SMIME no es más que un tipo MIME que añade al MIME la capacidad de firmar y cifrar mensajes de correo electrónico, manteniendo toda la funcionalidad del MIME en nuestros mensajes seguros. Permite enviar mensajes firmados, con lo que el receptor estará seguro de que el mensaje proviene de nosotros (autenticidad) y enviar mensajes cifrados, de modo que solo el receptor puede leerlos (confidencialidad).

## SMTP (SIMPLE MAIL TRANSFER PROTOCOL)

*SMTP* transporta sencillos archivos de texto

* + Protocolo ASCII sencillo
  + El remitente actúa como cliente y establece una conexión TCP con el destinatario (servidor)
  + Puerto público del destinatario 🡪 el 25
  + Se inicia una sesión SMTP en dos sentidos 🡪 remitente y destinatario intercambian comandos y respuestas
  + El servidor acusa la llegada de un mensaje
  + Daemon SMTP 🡪 acepta conexiones de entrada y copia mensajes a los buzones
  + Finalizada la transferencia del mensaje 🡪 se libera la conexión
  + Está basado en la entrega punto-a-punto: un cliente contactará con el servidor del host de destino directamente para entregar el correo

## Servicios adicionales

* + Se puede saber cuándo se envió y cuándo llegó correo a nuestro equipo
  + SMTP añade marcas de tiempo al mensaje, informando todos los MTA por los que pasó el mensaje (fecha y hora)
  + Devolución de correo 🡪 destinatario desconocido

## Inconvenientes

* + Algunas implementaciones más viejas no pueden manejar mensajes mayores de 64 Kbytes.
  + Si el cliente y el servidor tienen temporizaciones distintas, uno de ellos puede terminar mientras que el otro continúa trabajando, terminando inesperadamente la conexión.
  + Pueden dispararse tormentas de correo infinitas cuando ambos servidores tienen una lista que incluye a la otra lista del otro servidor.
  + Solución: SMTP extendido (ESMTP): Los clientes que deseen usarlo deben enviar un mensaje EHLO, en lugar de HELO. Si el saludo se rechaza, código 500, esto indica que el servidor es un servidor SMTP normal y el cliente debe proceder de la manera normal

## SMTP y el DNS

Si la red usa el concepto de dominio, un SMTP no puede entregar simplemente correo a TEST.IBM.comando abriendo una conexión TCP éste. Primero debe consultar al servidor de nombres para hallar a que host(en un nombre de dominio) debería entregar el mensaje.

Para la entrega de mensajes, el servidor de nombres almacena los registros de recursos denominados MX. Mapean un nombre de dominio a dos valores:

• Un valor de preferencia. Como pueden existir múltiples registros de recursos MX para el mismo nombre de dominio, se les asigna una prioridad. El valor más bajo corresponde al registro de mayor preferencia. Esto es útil siempre que el host de mayor preferencia sea inalcanzable; el emisor SMTP intenta conectar con el siguiente host en orden de prioridad.

• Un nombre de host: el servidor de nombres responde con una lista vacía de registros de recursos MX. Esto significa que el nombre de dominio se halla bajo la autoridad del servidor, pero no tiene ningún MX asignado. En este caso, el emisor SMTP puede intentar establecer la conexión con el mismo nombre del host.

## MTAs DE REENVÍO

Conceptos previos:

• **DNS y registros MX**: son intercambiadores de correo, que reciben correo en nombre de otro servidor cuando el principal está fuera de servicio

• **Relay o reenvío**: indica si un servidor de correo acepta correo de otro servidor para reenviar.

• **SPAM**: envío masivo a un conjunto de direcciones gestionadas por un servidor. El servidor puede configurarse para marcarlas como SPAM.

Normalmente la conexión TCP no es directa entre MTA origen y destino, sino que se pasa por MTA intermedias. Tipos:

• MTA local: configuradas para enviar todo mail no local a MTA relay

• MTA relay: maneja todo el correo de entrada y salida de la organización

Dos razones para esta división:

• Simplifica la configuración de las MTAs

• El MTA relay da una visión unitaria de la organización

• Típicamente entre dos UA se pasa por cuatro MTA

**PASARELAS DE CORREO**

¿Qué sucede si el destino no pertenece a la Internet?

¿El origen 🡪 protocolo SMTP? ¿El destino 🡪 X.400?

¿Se puede realizar una comunicación directa?

Solución 🡪 **pasarelas de correo**

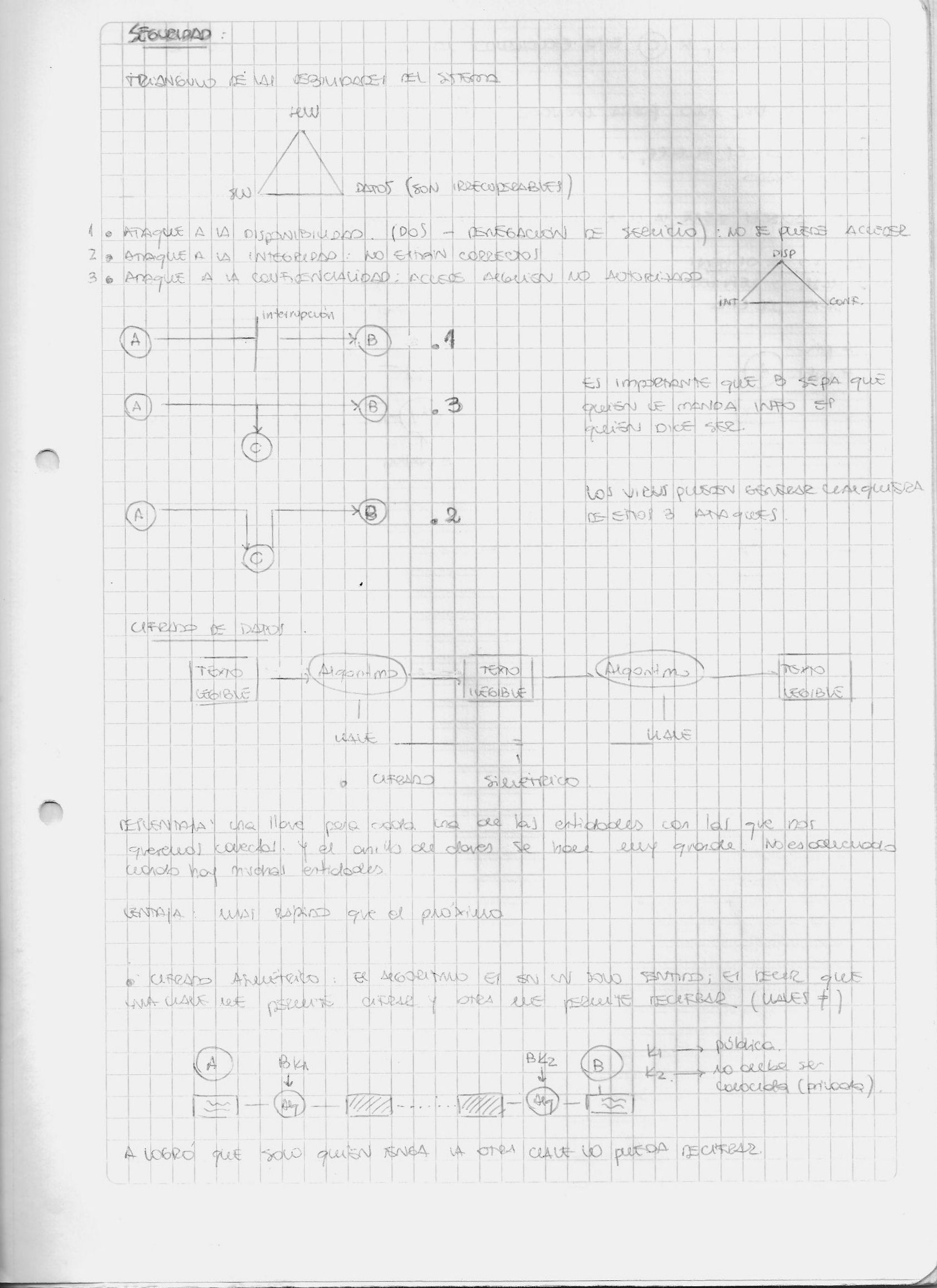
• Permiten intercambiar mensajes entre diferentes sistemas.

• Tienen muchas limitaciones

**SEGURIDAD**

*Triángulo de las debilidades del sistema:*

* Hw
* Sw
* Datos

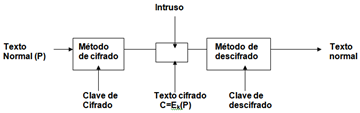
*Ataques:*

1. A la disponibilidad: no se puede acceder
2. A la integridad: los datos son incorrectos
3. A la confidencialidad: accede alguien no autorizado

Los virus pueden generar cualquiera de estos 3 ataques

**Cifrado de datos**

Un cifrado es fuerte, si conociendo el código, no puedo determinar el texto cifrado

* *Simétrico*
  + Algoritmo de clave secreta
  + Utilizan la misma clave para cifrado y descifrado
  + Objetivo: desarrollar un algoritmo muy complejo 🡪 cifrado de producto

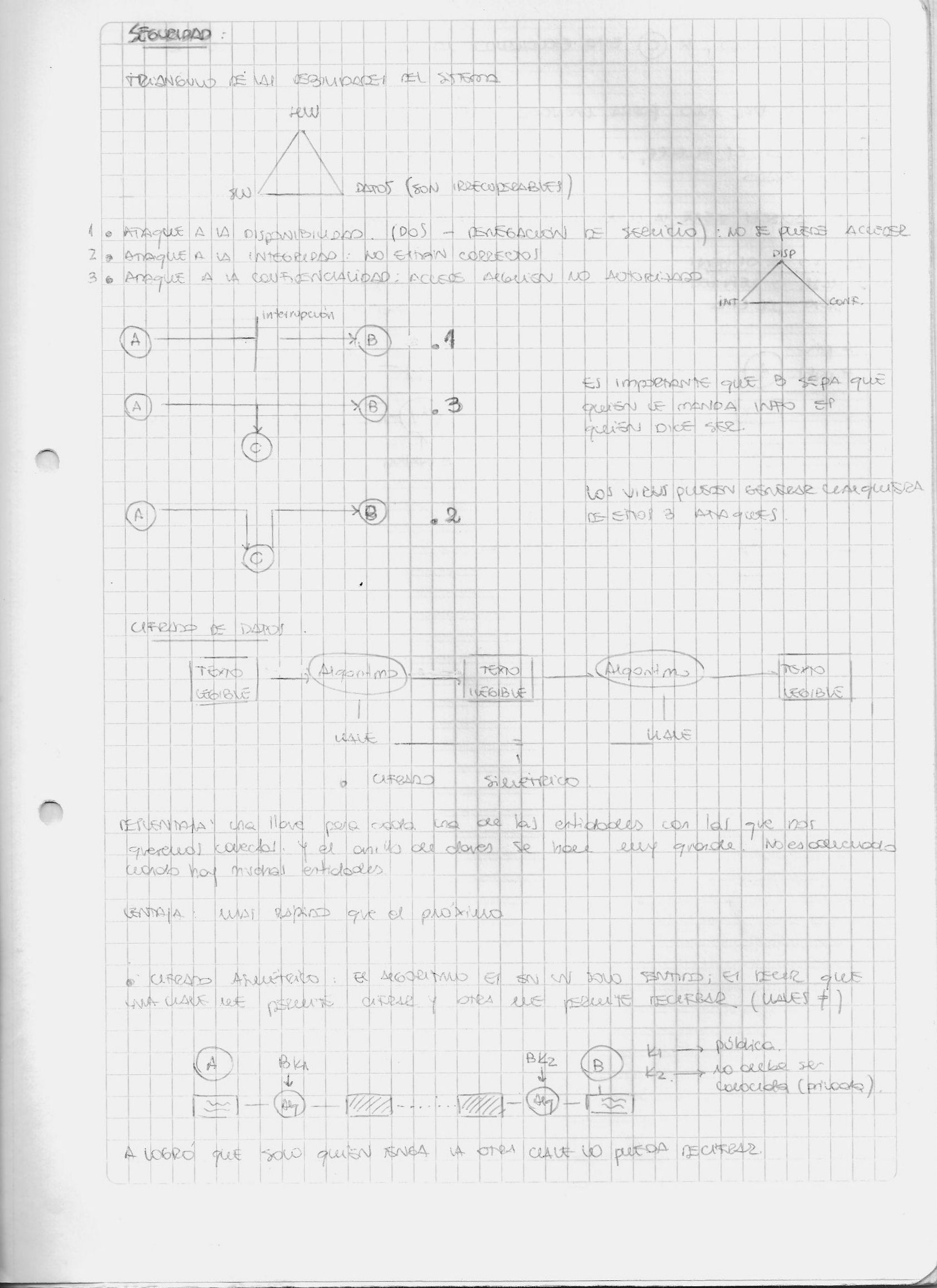
Desventaja: una llave para cada una de las entidades con las que nos queremos conectar. Con esto, el anillo de claves se hace muy grande, por lo que no es adecuado cuando hay muchas entidades

Ventaja: más rápido que el asimétrico

Distribución de claves simétricas

Si Ana necesita intercambiar mensajes confidenciales con N personas, requiere N claves distintas. Objetivo:

* + Solucionar la gran cantidad de claves diferentes para un gran número de usuarios
  + Distribuir eficientemente las claves entre los usuarios que desean confidencialidad
* *Asimétrico:* el algoritmo es en un solo sentido. Es decir, Cada usuario posee dos claves:
  + Clave pública para cifrar datos
  + Clave privada para descifrar datos



“A” logró que solo quien tenga la otra llave lo pueda descifrar

Con este método, si “C” está escuchando no va a poder saber qué se dicen. Si conoce BK1 solo podrá mandar mensajes privados a B.

Distribución de claves públicas

* Autoridad de certificación (CA = certification autority)
* Cuando Ana quiere enviar un mensaje a Bruno, lo encripta con la clave pública de Bruno y Bruno lo desencripta con su propia clave privada
* Las entidades necesitan estar seguros que tienen la clave pública de aquel con quien se están comunicando

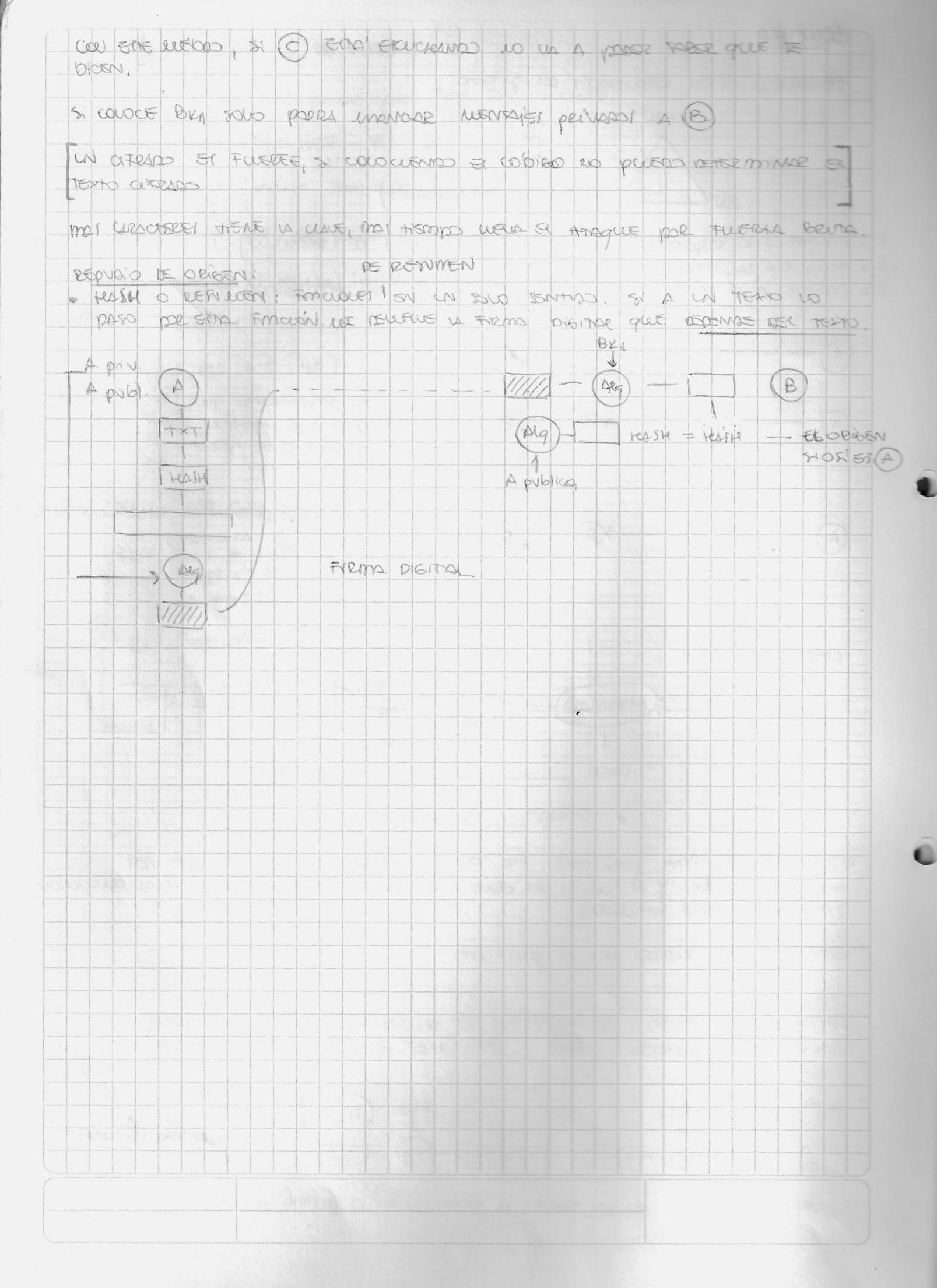
# Protocolos de validación de identificación

Validación de identificación (authentication): un proceso comprueba con quién se está comunicando

Autenticación ≠ Autorización

**Repudio de origen**

*Hash o resumen:* función de resumen en un solo sentido. Si a un texto lo paso por esta función, me devuelve la forma digital que depende del texto



*Firmas digitales*

Se espera:

1. El receptor pueda verificar la identidad del transmisor
2. El transmisor no pueda repudiar después el contenido del mensaje
3. El receptor no haya podido confeccionar el mensaje él mismo

Características:

* Bloque de caracteres que acompaña un documento
* El autor utiliza su propia clave secreta
* El autor queda vinculado al documento
* Cualquier persona puede verificar la validez de la firma si dispone de la clave pública del autor
* El SW del firmante aplica un algoritmo sobre el texto obteniendo un extracto de longitud fija (128 – 160 bits)
* Objetivo 🡪 firmar el resumen del mensaje 🡪 mayor rapidez
* Se cifra con la clave secreta del autor y se añade al final del mensaje