





DNS: Doman Name System

INTRODUCCIÓN

Introducción (1)

- El protocolo IP asigna direcciones individuales a todos los hosts en una cierta red
- Estas direcciones son simplemente números binarios sin mayor estructura
- Para enviar tráfico IP de un host a otro esto es técnicamente mas que suficiente
- Sin embargo, para los usuarios de la red es prácticamente imposible recordar o manejar estos números, es preferible contar con identificadores textuales

Introducción (2)

- Encabezado IPv4

00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Version		IHL		TOS				Total length																							
Identification				Flags				Fragment offset																							
TTL				Protocol				Header checksum																							
Source IP address																Destination IP address															
Options and padding -																															

- Espacio de direcciones:
 - 32 bits en IPv4
 - 128 bits en IPv6

Introducción (3)

- DNS: *Domain Name System*
- Propósito básico:
 - Traducir números IP en nombres textuales mas amigables para los usuarios "humanos" de la red
- Propósitos adicionales:
 - Soporte a diferentes servicios a dar sobre la red
 - Correo electrónico
 - Sub-delegaciones de nombres
 - Resolución reversa
 - Reverso: correspondencia nombre -> número IP


Introducción (4)

- Requerimientos del sistema:
 - Apoyo a diferentes consultas y aplicaciones
 - Nombres directos, reversos, apoyo a aplicaciones
 - Distribución de la administración
 - En Internet no hay administración centralizada sino que todo es por naturaleza distribuido. El DNS debe soportar y apoyar esta forma de trabajo.
 - Performance adecuada
 - Las consultas deben responderse lo mas rápidamente posible.
 - Confiabilidad adecuada
 - El DNS es obviamente una pieza crítica de la infraestructura de Internet, por lo que debe ser altamente confiable.

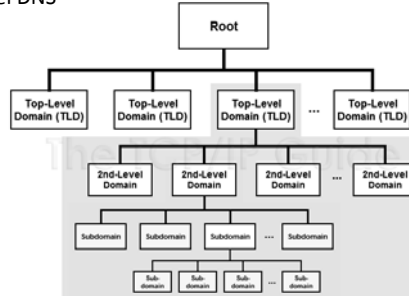
Introducción (4)


- El DNS como base de datos:
 - El objetivo principal del DNS es entonces almacenar información de mapeo entre nombres y números IP
 - Directa e inversa
 - El sistema opera entonces como una base de datos distribuida en la que existe la posibilidad de delegar la administración de sectores del espacio de nombres a diferentes organizaciones

Introducción (4)

- Estructura de los nombres de dominio:

- Comentarios:
 - Los niveles del árbol reflejan las divisiones administrativas
 - El root del arbol esta siempre presente de forma implícita
 - No hay restricciones a la cantidad de niveles
 - Los niveles superiores "delegan" hacia los inferiores

Introducción (5)

- El árbol del DNS


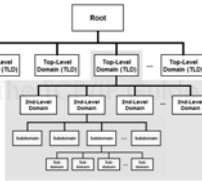


DNS: Doman Name System

CONCEPTOS BÁSICOS Y OPERACIÓN


Conceptos básicos

- Zonas
 - A cada dominio (*incluyendo siempre al root*) le corresponde lo que se denomina una zona de DNS
- Autoridad
 - Cada zona define una región de autoridad donde se le reconoce el derecho organización que administra la misma
 - Respuestas autoritativas



Conceptos básicos

- Registros (*Resource Records*)
 - La información en la base de datos del DNS está estructurada en un conjunto de *resource records*:
 - SOA, A, NS, MX, PTR, TXT, etc.
 - Cada RR representa un ítem de información en la base de datos de DNS que puede ser consultado
- SOA: “*Start of Authority*”
 - Delimita una zona
 - Incluye a todos los RRs de la misma



Conceptos básicos: Resource Records

- RR “A”: *Address*
 - Los registros A establecen las correspondencias entre direcciones IP y nombres de dominio
- RR “CNAME”: *Canonical Name*
 - Son el equivalente de los alias o de los links simbólicos.
 - Establecen una correspondencia entre dos nombres
 - En teoría para resolver completamente a la dirección hacen falta dos consultas
 - En la práctica los servidores ya devuelven el “A” correspondiente en la sección Additional de la consulta (ya lo vamos a ver)

Conceptos básicos: Resource Records

- RR "PTR": *Pointer*
 - Los registros PTR establecen enlaces o punteros entre nombres de DNS, es muy similar conceptualmente al CNAME
 - El principal uso es construir el dominio `in-addr.arpa` que contiene los reversos
- RR "SOA": *Start of Authority*
 - Establece el comienzo de una zona de DNS, tiene varios campos:
 - **MNAME, RNAME:**
 - dominio y mailbox del administrador
 - **SERIAL:**
 - número versión (32 bits), usado para saber si hay cambios
 - **RETRY:**
 - tiempo a esperar para reintentar una transferencia fallida
 - **EXPIRE:**
 - tiempo a esperar hasta considerar la zona no autoritativa
 - **MINIMUM:**
 - TTL mínimo que se exporta con cualquier RR que se responde sobre esta zona

Conceptos básicos: Resource Records

- RR "MX": *Mail Exchanger*
 - *Each MX record specifies a domain name (which must have an A record associated with it) and a priority; a list of mail exchangers is then ordered by priority when delivering mail. MX records provide one level of indirection in mapping the domain part of an email address to a list of host names which are meant to receive mail for that domain name. Critical part of the infrastructure used to support SMTP email. Defined in RFC 1035.*
- RR "NS": *Name Server*
 - Los glue records
 - *Specifies a host name (which must have an A record associated with it), where DNS information can be found about the domain name to which the NS record is attached. NS records are the basic infrastructure on which DNS is built; they stitch together distributed zone files into a directed graph that can be efficiently searched. Defined in RFC 1035.*
- Otros RRs:
 - Informativos: TXT
 - Locación geográfica: LOC
 - Seguridad: KEY, KX, TSIG, DNAME

Conceptos básicos

- Formato de paquetes DNS
 - Header
 - Encabezado del protocolo
 - Question
 - La pregunta que hacemos al DNS
 - Tuplas (*Name, Type, Class*)
 - Answer
 - RRs que responden la pregunta (si es que hay), también en (N, T, C)
 - Authority
 - RRs que apuntan a una autoridad (opcional)
 - Additional
 - RRs que a juicio del DNS pueden ser útiles para quien está preguntando

Header

Question

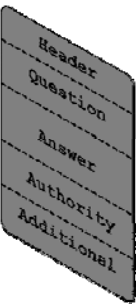
Answer

Authority

Additional

Conceptos básicos

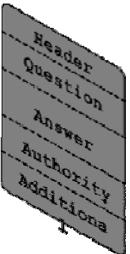
- Formato de paquetes DNS (II)
 - Class
 - Identifica la aplicación, en Internet es siempre **IN**
 - Type
 - El tipo de RR
 - SOA, MX, A, etc.
 - Name
 - El nombre completo por el que estamos preguntando o por el que se está respondiendo



Conceptos básicos

- Formato de paquetes DNS (III)
 - Flags:
 - Los paquetes de DNS llevan algunos flags que afectan a como el servidor interpreta la consulta o traen informacion sobre respuesta

bit 5	AA	Auth. Answer	[RFC1035]
bit 6	TC	Truncated Response	[RFC1035]
bit 7	RD	Recursion Desired	[RFC1035]
bit 8	RA	Recursion Allowed	[RFC1035]
bit 9		Reserved	
bit 10	AD	Authentic Data	[RFC4035]
bit 11	CD	Checking Disabled	[RFC4035]



Conceptos básicos

- Primarios y secundarios
 - Cada zona tiene que tener al menos un servidor de nombres que sea autoritativo para ella
 - Este es el primario de la zona
 - Por motivos de redundancia, se recomienda tener uno o más servidores secundarios para la misma
 - Los secundarios también son autoritativos
- Transferencia de zonas
 - Para no tener que configurar la misma información dos o tres veces, y para facilitar la operación, existe un protocolo de transferencia de zonas (AXFR)

Conceptos básicos

- Terminología sobre consultas
 - Consultas directas
 - De nombre a dirección
 - A
 - Consultas reversas
 - De dirección a nombre
 - PTR
 - Ojo con las “inversas”

Conceptos básicos: Transporte

- Clientes y servidores pueden elegir entre TCP y UDP, los servidores DEBEN atender en ambos
 - UDP puerto 53
 - TCP puerto 53
- UDP:
 - El preferido por clientes finales
 - Las consultas pueden truncarse (MTU)
- TCP:
 - El preferido para las consultas recursivas servidor-servidor
 - Las consultas no se truncan

Operación: Consultas

- Esquema de una consulta DNS

```
graph LR
    subgraph Client
        WB[Web browser  
URL: www.microsoft.com]
        DR[DNS client (resolver)]
    end
    subgraph Server
        DS[DNS server]
        ODS[Other DNS servers]
    end
    WB --> DR
    DR --> DR
    DR --> DS
    DS --> DR
    DS --> ODS
    ODS --> DS
```

Operación

- Esquema de una consulta DNS (II)
 - El PC final tiene un *resolver* local
 - Archivo */etc/hosts*
 - Si aquí hay una entrada, se responde desde aquí
 - Apunta a un servidor DNS
 - Cada DNS trata de responder de:
 - Sus *hints*
 - Su caché
 - Sus zonas autoritativas
 - Cada DNS cachea de forma agresiva todo los RRs que sean posibles
 - ¿Hasta cuando? Se guía por los tiempos establecidos en los registros SOA y en los TTLs

Operación: Consultas

- Esquema de una consulta DNS (III)
 - Una consulta puede ser:
 - No recursiva
 - Todos los servidores DEBEN soportarlas
 - La respuesta final podrá o no llegar
 - » Dependiendo de hints y cachés
 - Recursiva
 - Opcional
 - La respuesta final SIEMPRE es devuelta
 - Inversa
 - Cuando se busca el name dado un cierto RR
 - » NO confundir con la consulta "reversa" de numeros IP a nombres

Operación: Consultas

- Esquema de una consulta DNS (IV):
 - Recursión:
 - Este es el proceso a través del cual el árbol se forma
 - Si un servidor recibe una consulta por:
 - (IN, A, www.adinet.com.uy)
 - Debe:
 - Buscar la raíz
 - Buscar el "uy"
 - Buscar el "com.uy"
 - Buscar el "adinet.com.uy"
 - Buscar el "www.adinet.com.uy"
 - Cuando decimos buscar decimos:
 - Seguir la pista de la *autoridad*

Operación: Root Servers

- ¿Como arranca el proceso? Buscando la *raíz*
- Root servers
 - Son una serie de servidores *bien conocidos* repartidos en el mundo
 - Todos los servidores DNS cuando uno los instala vienen un *hint file* de los root servers
- ¿Como se sigue la *autoridad*?
 - Cada zona puede delegar sub-zonas a otros servidores
 - *Glue records*
 - Son registros NS (*name server*) que apuntan a una sub-zona, realizando una delegación de autoridad

Operación: Root Servers


- (Fuente: *Wikipedia*)

Letter	IP address	Old name	Operator	Location	Software
A	198.41.0.4	ns.internic.net	VeriSign	Dulles, Virginia, U.S.	BIND
B	192.228.79.201	ns1.isi.edu	USC-ISI	Marina Del Rey, California, U.S.	BIND
C	192.33.4.12	c.psi.net	Cogent Communications	distributed using anycast	BIND
D	128.8.10.90	tep.umd.edu	University of Maryland	College Park, Maryland, U.S.	BIND
E	192.203.230.10	ns.nasa.gov	NASA	Mountain View, California, U.S.	BIND
F	192.5.5.241	ns.nic.org	ISC	distributed using anycast	BIND
G	192.112.36.4	ns.nic.ddn.mil	Defense Information Systems Agency	Columbus, Ohio, U.S.	BIND
H	128.63.2.53	aos.arl.army.mil	U.S. Army Research Lab	Aberdeen Proving Ground, Maryland, U.S.	NSD
I	192.36.148.17	nic.nordu.net	Autonomica	distributed using anycast	BIND
J	192.58.128.30		VeriSign	distributed using anycast	BIND
K	193.0.14.129		RIPENCC	distributed using anycast	NSD
L	199.7.83.42		ICANN	distributed using anycast	NSD
M	202.12.27.33		WIIDE Project	distributed using anycast	BIND

- Observación:
 - “*distributed using anycast*” significa que existen múltiples copias en el mundo de los mismos

Operación: Root Servers

- (Fuente: *Wikipedia*)



ANTEL
TELECOMUNICACIONES

CSIRT
CENTRO DE RESPUESTA DE EMERGENCIAS
MEMBRERA Y TELECOMUNICACIONES

DNS: Doman Name System

HERRAMIENTAS

Herramientas: Dig

- DIG

```

carlosm@evalon: ~/00_Wksp_Personal
carlosm@evalon:~/00_Wksp_Personal$ dig -t A www.adinet.com.uy
<>>> DIG 9.4.1-P1 <>>> -t A www.adinet.com.uy
;; global options: printcmd
;; Got answer:
;;->>>HEADER<>> opcode: QUERY, status: NOERROR, id: 39844
;; Flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 2
;; QUESTION SECTION:
; www.adinet.com.uy.                IN      A
;; ANSWER SECTION:
www.adinet.com.uy.                66488  IN      A      200.48.204.239
;; AUTHORITY SECTION:
adinet.com.uy.                    16546  IN      NS      ns1.anteldata.com.uy.
adinet.com.uy.                    16546  IN      NS      ns2.anteldata.com.uy.
;; ADDITIONAL SECTION:
ns1.anteldata.com.uy.             57925  IN      A      200.48.38.254
ns2.anteldata.com.uy.             69045  IN      A      200.48.229.254
;; Query time: 3 msec
;; SERVER: 172.16.16.16#53(172.16.16.16)
;; WHEN: Wed Dec 5 10:54:36 2007
;; MSG SIZE  rcvd: 129
carlosm@evalon:~/00_Wksp_Personal
  
```

Herramientas: Dig

- DIG: consultas reversas

```

carlosm@evalon: ~/00_Wksp_Personal
carlosm@evalon:~/00_Wksp_Personal$ dig -x 200.48.204.239
<>>> DIG 9.4.1-P1 <>>> -x 200.48.204.239
;; global options: printcmd
;; Got answer:
;;->>>HEADER<>> opcode: QUERY, status: NOERROR, id: 8135
;; Flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 0
;; QUESTION SECTION:
; 200.48.204.239.in-addr.arpa.        IN      PTR
;; ANSWER SECTION:
200.48.204.239.in-addr.arpa.        66488  IN      CNAME  200.224.20.204.48.200.in-addr.arpa.
200.224.20.204.48.200.in-addr.arpa. 66488  IN      PTR    malena.ingoipia.com.
;; AUTHORITY SECTION:
200.20.204.48.200.in-addr.arpa.     67156  IN      NS      ns1.ingoipia.com.
200.20.204.48.200.in-addr.arpa.     67156  IN      NS      ns2.ingoipia.com.
;; Query time: 13 msec
;; SERVER: 172.16.16.16#53(172.16.16.16)
;; WHEN: Wed Dec 5 11:13:47 2007
;; MSG SIZE  rcvd: 139
carlosm@evalon:~/00_Wksp_Personal
  
```

Seguridad en DNS

- Aspectos principales:
 - Caché Poisoning
 - DNS amplification attacks
 - Aseguramiento de las transferencias de zona
 - Certificación de autoridad
- Aspectos no directamente relacionados con el protocolo
 - Vulnerabilidades en el software que implementa DNS

Seguridad en DNS: *Caché Poisoning*

- El *caché poisoning* es una técnica por la cual es posible engañar a un servidor DNS y hacerle creer que recibió información auténtica y válida
- El servidor luego cachea esa información y la utiliza para responder otras consultas hasta la duración el TTL de los RRs cacheados
- De esta forma propaga el engaño downstream
- ¿Para qué?
 - Redirigir tráfico a sitios tomados, *pharming*
 - Robo de información

Seguridad en DNS: *Caché Poisoning (II)*

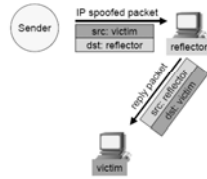
- ¿Cómo?
 - Recordar que los servers cachean agresivamente la sección *Additional* de la respuesta
 - Truco: devolver el engaño en esta sección
 - El atacante debe tener un DNS server bajo su dominio, con una zona autoritativa.
- Ejemplo:
 - Un cliente pregunta al DNS de good.org por el MX de bad.org
 - La respuesta trae el MX de bad.org y además trae, por ejemplo:
 - `ns.banco.com.uy IN A X.Y.Z.W`
 - X.Y.Z.W es la IP del DNS de bad.org o cualquier otro equipo malicioso

Seguridad DNS: *Caché Poisoning (III)*

- *Caché poisoning via DNS forgery*
 - Inyectar paquetes de respuesta antes que el verdadero servidor de nombres en una consulta recursiva
 - Implica adivinar un nonce de 16 bits
 - Es lo que se usa para atar preguntas con respuestas
 - Puede no ser tan complicado, es un ejemplo de un birthday attack

Seguridad en DNS: *DNS amplification*

- *DDoS via DNS amplification*
 - Es una forma de lograr DDoS aprovechando servidores DNS que responden consultas recursivas para cualquiera
 - Esto desde siempre se supo que no era una buena práctica, pero...
- Componentes:
 - Consultas con IPs spoofed
 - Generar reflexiones a otros lugares, clásico uso del spoofing
 - DNS amplification
 - ¿Cómo es posible? Usando UDP, ¡no hay 3-way handshake!




Seguridad DNS: *DNS amplification (II)*

- Primer paso:
 - Hacer que muchos servidores de nombres guarden en caché uno o mas registros grandes
 - Típicamente se usan TXTs que reside en un servidor bajo control del atacante. El largo máximo (RFC 1035) es de 255 caracteres, pero se pueden poner varios.
 - Esto se puede hacer consultándolos por ese registro, y esto funciona siempre que los servidores amplificadores soporten recursión
 - El paquete de consulta es pequeño, p.ej. 20 bytes
 - La respuesta puede llegar a ser de 1K a 4K!

Seguridad DNS: DNS amplification (III)

- Segundo paso:
 - Realizar las consultas *spoofed* con IP de origen la de la víctima
 - Las consultas se hacen por los registros TXT que esperamos esten en el caché de todos los recursivos intermedios
 - Reir!!!
- Algún numero:
 - 20 servidores recursivos
 - Amplificación 100X
 - 20 bytes -> 2 Kbytes
 - ADSL de 512 Kbps
 - Tráfico de ataque: **1 Gbps**
 - 512 Kbps * 100 * 20



Seguridad en DNS: AXFR's

- Transferencias de zona:
 - Necesario para implementar esquemas de replicación primario-secundarios
- Problemas:
 - Exposición de la información
 - Por defecto las zonas viajan en texto plano
 - Denegaciones de servicio
 - Si cualquiera puede disparar un AXFR, y como los AXFRs son potencialmente de muchos RRs, se pueden generar DoS locales
- Soluciones:
 - Buenas prácticas de filtrado
 - Cifrado en las transferencias
 - Propios del DNS
 - Establecer esquemas de VPN u otros túneles cifrados

Seguridad en DNS: Autoridad

- ¿Porqué?
 - Si un atacante logra mentir sobre la autoridad sobre una zona puede envenenar cachés y redirigir tráfico a voluntad
 - No solo en caso de ataques, sino también en casos de disputas comerciales, surge cada vez mas como una necesidad el poder certificar la autoridad de una organización sobre una zona
- Propuestas
 - Protocolo DNSSEC
 - Esquema de PKI apoyado en el propio arbol del DNS
 - Leito sabe mas que yo de esto ☺




DNS: Doman Name System

COMENTARIOS FINALES

Comentarios finales

- Esta presentacion apenas rasca un tema *muy* vasto
- Para completar un panorama completo del tema faltaria:
 - Hacer un pequeno tutorial sobre como configurar un DNS con BIND9
 - Repasando conceptos mas especificos de las implementaciones, como ser el de las *views*
 - Hablar mas en detalle de como se implementa el mapeo reverso y de la zona in-addr.arpa
 - Hablar mas en detalle de DNSSEC
 - Hablar mas en detalle de los posibles ataques al DNS, los que estan nombrados aca y los que no




DNS: Doman Name System

¡ GRACIAS POR SU ATENCION! ¿PREGUNTAS?
