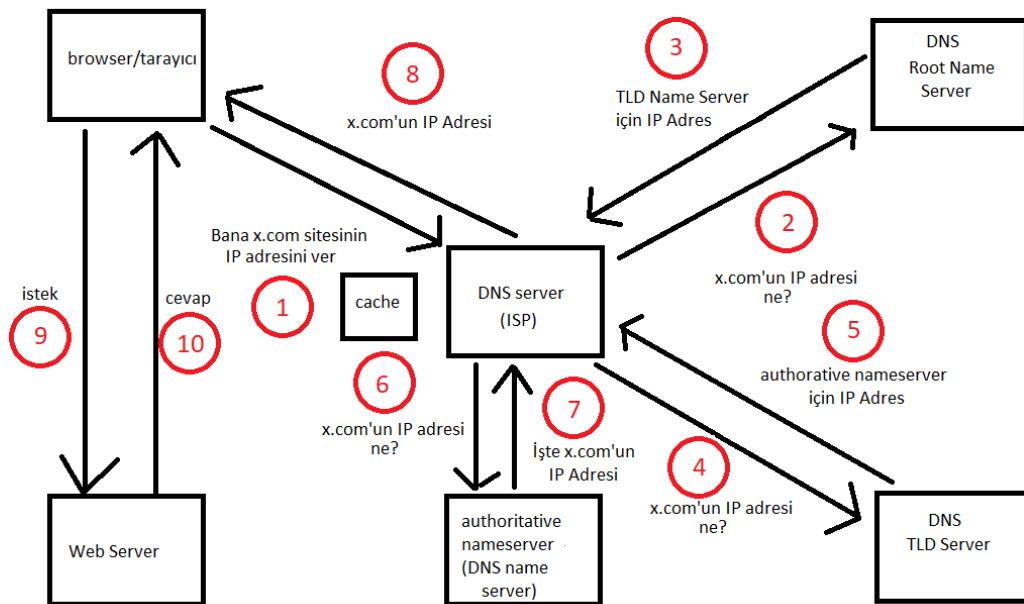


WEB & INTERNET & DNS



Hazırlayan

Ömer Faruk Kocaefe

İletişim ve Kanallar

Linkedin: <https://www.linkedin.com/in/omer-faruk-kocaefe/>

Youtube: <https://www.youtube.com/@sibernia9144>

Udemy: <https://www.udemy.com/user/sibernia/>

Açıklama

Bu dokümanda web, Internet ve DNS sistemlerinin çalışma mantığına değinilmiştir. Üzerinde durulan asıl konu DNS sistemleridir. Öncelikle, kısaca teorik olarak web sistemleri üzerinde durulmuştur. Daha sonra Internet'le birlikte biraz Network konuları anlatılmıştır. En sonunda DNS konusu ele alınmıştır. DNS içinde; DNS çalışma mantığı, DNS kayıt tipleri, DNS saldırıları ve DNSSec gibi çözümlerden bahsedilmiştir.

Bu doküman tamamen teoriktir ve öğrenilen bilgiler doğrultusunda uygulanacak olan zararlı işlemlerden Ömer Faruk Kocaefe sorumlu değildir.

Laboratuvar ortamlarında **LEGAL kalın!**

WWW (World Wide Web) Nedir?

Web, www ve W3 olarak da bilinen World Wide Web; tüm public web sitelere veya sayfalara tekabül eder. Web, bir bilgisayar ağı olan Internet üzerinde bilgi ve dokümanların paylaşıldığı, erişilebildiği ve gezilebildiği bir sistemdir. Web ile metinler, görüntüler, videolar, sesler ve diğer medya öğeleri birbirlerine bağlanır. Kullanıcılar kendi local cihazlarından Internet aracılığıyla web'e bağlanır.

Web, HTTP ve HTTPS gibi protokoller üzerinde çalışır ve web tarayıcıları aracılığıyla kullanıcıların web sitelerine erişmesine olanak sağlar. Web sayfaları; HTML, CSS ve JavaScript gibi diller kullanılarak oluşturulur ve bağlantılar aracılığıyla birbirlerine bağlanır. Bu bağlantılar, kullanıcıların farklı sayfalar arasında gezinmesini ve içeriklere erişmesini sağlar.

Web günümüzde çok yaygın olarak kullanılır. Bunun sebepleri: bilgiye erişmek, iletişim kurmak, e-ticaret yapmak, platformlara katılmak. Yani kısaca Web çevrimiçi etkinlik sağlayan iletişim aracıdır.

WWW Tarihçesi

➔ Kurulum Zamanı

CERN'de (İsviçre, Cenevre yakınlarında bir araştırma laboratuvarı) 1989 yılında Tim Berners-Lee bir teklif sundu. Bu teklif bilgi yönetim sistemi içeriyordu. Bu sistem ise bilgisayar ağları üzerinde kaynak ve bilgi paylaşımı sağlıyordu.

Sunduğu sistem git gide yayıldı ve World Wide Web (www) olarak adlandırılmıştı. WWW dünyada farklı amaçlarla kullanıldı.

➔ Web'in evrimi sürecindeki önemli dönem noktaları nelerdir?

1960'lar:

- Internetin temeli olan ARPANET oluşturuldu. Bu, bilgisayarların birbirine bağlandığı ilk ağıdı.
- Hypertext kavramı ortaya çıktı. Ted Nelson, Xanadu adlı projesiyle birbiriyle bağlantılı belgeleri tanımladı.

1980'ler:

- Internetin TCP/IP (Transmission Control Protocol/Internet Protocol) protokollerini kullanarak daha geniş çapta yaygınlaşmaya başladı.
- 1989'da Tim Berners-Lee, CERN'de World Wide Web'i (WWW) yaratmak için ilk adımları attı. İlk web tarayıcısı "WorldWideWeb" ve ilk web sunucusu da aynı yıl oluşturuldu.

1990'lar:

- 1991'de HTTP (Hypertext Transfer Protocol) tanıtıldı ve web sayfalarının sunulmasında kullanılmaya başlandı.
- Web tarayıcıları popülerleşti. Mosaic (1993) ve Netscape Navigator (1994) gibi tarayıcılar web kullanımını artırdı.
- 1994'te W3C (World Wide Web Consortium), web standartlarını belirlemek ve ilerletmek amacıyla kuruldu.
- 1995'te Microsoft, Internet Explorer tarayıcısını piyasaya sürdü ve web tarayıcısı rekabeti hızlandı.
- 90'ların sonu 2000'lerin başı blog yazıları popülerleşti.

2000'ler:

- Web 2.0 dönemi başladı (2004). Kullanıcıların içerik üretmesine ve paylaşmasına dayalı interaktif web siteleri gelişti.
- Sosyal medya platformları ortaya çıktı. Facebook (2004), YouTube (2005) ve Twitter (2006) gibi platformlar büyük popülerlik kazandı.
- Mobil cihazların yaygınlaşmasıyla birlikte mobil web kullanımı hızla arttı.
- 2009'da Bitcoin ve blockchain teknolojisi ortaya çıktı, web üzerinde finansal işlemleri ve dijital varlık transferini etkileyen bir alan oluşturdu.

2010'lar:

- Web tarayıcıları daha hızlı ve gelişmiş hale geldi. Google Chrome, Mozilla Firefox ve Safari gibi tarayıcılar önemli rekabetçi güçler haline geldi.
- Bulut bilişim hızla yaygınlaştı ve web tabanlı hizmetlerin altyapısını değiştirdi.
- Web uygulamaları ve çevrimiçi hizmetler, iş, iletişim, eğlence ve alışveriş gibi birçok alanda hayatımızın vazgeçilmez bir parçası haline geldi.

2020'ler:

- Yapay zeka, büyük veri analitiği ve nesnelerin interneti (IoT) gibi teknolojiler, webin daha da gelişmesine katkıda bulunuyor.

İlk web tarayıcıları ve nasıldı?

1990 – The WorldWideWeb: Tim Berners-Lee tarafından oluşturulan ilk tarayıcıdır (www/web ile karıştırılmamalı). Daha sonra Nexus ismi verildi. Bunun sebebi www ile karıştırılmamasını sağlamak. O zamanlar web'e erişim sağlamak için kullanılan tek tarayıcıydı.

1992 – Lynx: Lynx metin tabanlı (text-based) tarayıcıydı ve herhangi bir grafik içeriği yoktu.

1993 – Mosaic: Mosaic, metin tabanlı olmayan içerikleri, grafikleri ve tabloları görüntüleyebilen ilk popüler web tarayıcısı oldu ve Web'in geniş kitleler tarafından kullanılmasını sağladı.

1994 – Netscape Navigator: Mosaic, sadece metin tabanlı olmayan içerikleri, grafikleri ve tabloları görüntülemekteydi. Netscape Navigator ise daha gelişmiş bir tarayıcı olarak kullanıcılarına daha fazla özellik sunuyordu. Netscape Navigator, çerezleri (cookies) destekliyor, SSL (Secure Sockets Layer) gibi güvenlik protokollerini kullanabiliyor, e-posta istemcisi entegrasyonu sağlıyordu ve daha fazla etkileşimli öğeyi destekleyerek zenginleştirilmiş web deneyimi sunuyordu. Mosaic temelleri üzerine inşa edilse de içerisinde daha fazla özellik de barındırıyordu.

1995 – Internet Explorer: Internet Explorer Microsoft'un ilk web tarayıcısı olarak ortaya çıktı. İlk sürüm Mosaic tabanlıydı. Grafikleri ve tabloları görüntülemek için kullanılıyordu (1995). İkinci sürümde SSL desteği, yeni HTML özellikleri ve daha iyi performans gösterdi (1995). Üçüncü sürümde CSS desteği, çerezler ve masaüstü entegrasyonu gibi özellikleri içeriyordu (1996). 4,5,6,7,8,9,10 ve 11 gibi sürümleri bulunmakta olup Microsoft desteğini çekmiştir. Günümüzde ise kullanılmamaktadır.

1996 – Opera: Opera 1994 yılında araştırma projesi olarak ortaya çıkmıştı. 2 sene sonra halka açıldı. Ortaya çıkmasıyla Opera, Internet Explorer ve Netscape arasında ortalık rekabetçi ortama bürünmüştü.

2003 – Apple Safari: Macintosh bilgisayarlar için Netscape Navigator yerine piyasaya sunulmuştu.

2004 – Firefox: Mozilla, Netscape Navigator satın alarak Firefox'u ortaya çıkarttı. Şu an Netscape Navigator'ün varisi olarak kullanılmaktadır.

2007 – Mobile Safari: Mobil Apple cihazlar için ortaya çıkan Mobile Safari iOS marketi domine etmişti.

2008 – Google Chrome: Chrome ortaya çıktı ve tarayıcı pazarını eline geçirdi.

2015 – Microsoft Edge: Microsoft Edge, Google ile mücadele etmesi için 2015 yılında piyasaya sunuldu.

www Neden Kullanılır?

- Eğitim kurumları ve araştırma laboratuvarları Web'in ilk kullanıcılarından birisi oldular. Web'i doküman ve diğer kaynakların Internet üzerinde paylaşımı için kullandılar.
- Web'i kullanıcılar; posta servisleri, sanal fotoğraf albümleri ve global satışlar olarak kullandı. İletişim ve sosyal ağ olarak tercih etmektedirler.
- İşletmeler e-ticaret adı altında etkileşime girdiler. Bu şekilde ağ üzerinde satın alım ve satma ortamı oluştı. Aynı zamanda diğer işletmeler ile B2B (business-to-business) yoluyla iletişime girdiler.
- World Wide Web, araştırmacılar ve meraklılar için bilimsel makaleler, araştırma raporları, veritabanları ve diğer kaynaklarla doludur. Kullanıcılar, çeşitli konular hakkında bilgi edinmek, yeni bilgiler keşfetmek ve araştırma yapmak için webi kullanabilirler.
- Web, kullanıcılarla çeşitli eğlence kaynaklarına erişim imkanı sunar. Müzik dinleme, film ve dizi izleme, oyun oynama, çevrimiçi yayınıları takip etme gibi eğlence aktiviteleri web üzerinden gerçekleştirilebilir.

www nasıl çalışır?

Browser Nedir?

Tarayıcı (browser), bilgisayar, akıllı telefon, tablet ve diğer cihazlar aracılığıyla internet üzerindeki web sayfalarını görüntülemek ve etkileşimde bulunmak için kullanılan bir yazılımdır. Tarayıcılar, web içeriğini alırlar, HTML, CSS ve JavaScript gibi web teknolojilerini yorumlar ve kullanıcılarla görsel olarak sunarlar.

Web Server Nedir?

Web sunucusu (web server), istemcilere (genellikle web tarayıcılarına) web sayfalarını sağlayan, HTTP (Hypertext Transfer Protocol) üzerinden iletişim kurarak istemci taleplerini işleyen bir yazılım veya donanım sistemidir.

DNS Nedir?

DNS (Domain Name System), internet üzerindeki alan adlarını (örneğin, www.example.com) IP adreslerine ve IP Adresleri alan adlarına dönüştüren bir sistemdir. İnsanların anlayabileceği alan adlarını, bilgisayarların anlayabileceği IP adreslerine çevirerek iletişimini kolaylaştırır. Tam tersi işlemde ise doğrulama yapmak için kullanılır.

DNS Server Nedir?

DNS sunucusu (DNS server) ise DNS hizmetini sağlayan bir sunucu veya hizmettir. DNS sunucusu, alan adlarının IP adreslerine dönüştürülmesinden sorumludur. İstemciler (örneğin, web tarayıcıları), DNS sunucusuna bir DNS sorgusu göndererek bir alan adının IP adresini öğrenirler. Yani, DNS bir sistem veya protokolken, DNS sunucusu bu sistemi uygulayan ve alan adlarının IP adreslerine çözülmesinden sorumlu olan bir sunucudur. DNS sunucusu; DNS sorgularını işler, kaynak sunuculara yönlendirme yapar ve istemcilere alan adlarının karşılık geldiği IP adreslerini sağlar.

Cache nedir?

Cache, verilerin geçici olarak saklandığı bir bellek alanıdır (önbellek). İnternet ve bilgisayar sistemlerinde cache, sık kullanılan verilerin daha hızlı erişilebilir olmasını sağlamak için kullanılır.

DNS Root Name Server Nedir?

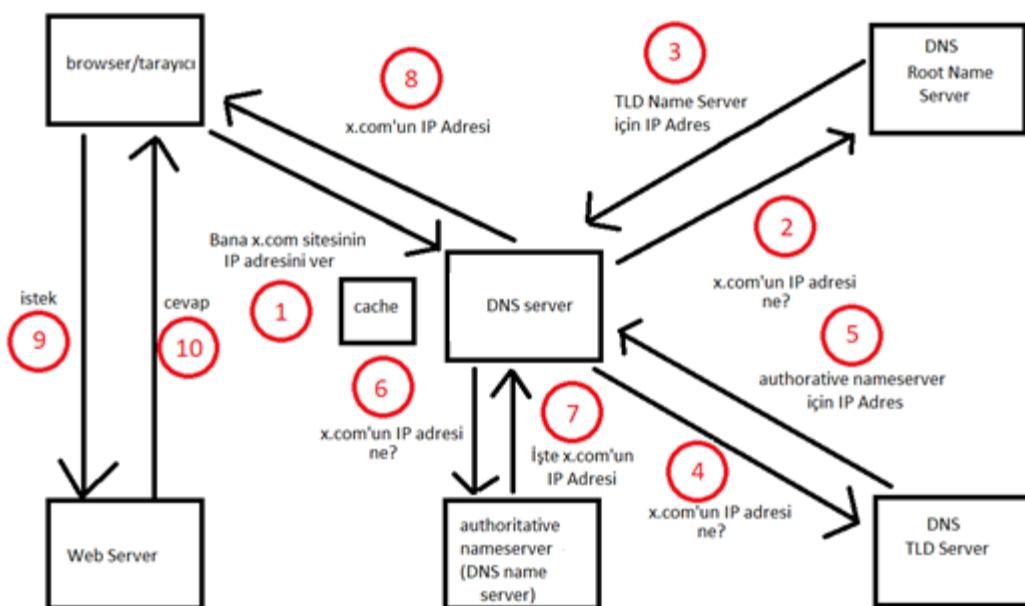
Kök ad sunucusu (root name server), İnternet'teki DNS (Domain Name System) hiyerarşisinin en üst düzeyinde yer alan sunuculardır. Kök ad sunucuları, DNS sisteminin temel yapı taşıdır. İnternet üzerindeki herhangi bir alan adına ilişkin bilgi, kök ad sunucuları tarafından sağlanmaz. Bunun yerine, kök ad sunucuları, tüm DNS sorgularının yönlendirildiği ilk noktalardır.

DNS TLD (Top Level Domain) Server Nedir?

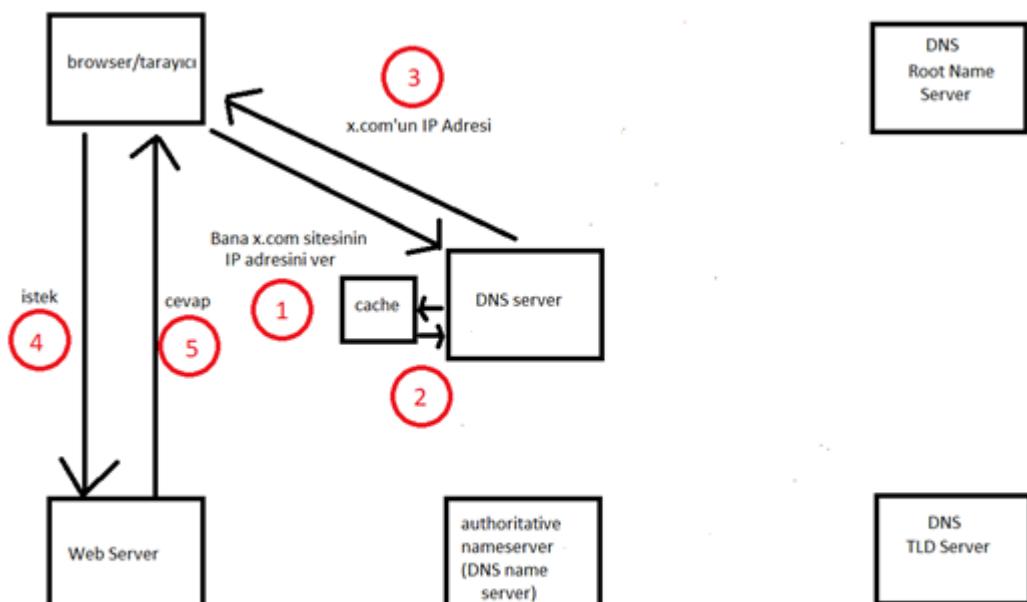
Üst düzey alan adı sunucusu (top-level domain name server), İnternet'teki DNS (Domain Name System) hiyerarşisinin üst düzey alan adlarına ilişkin bilgileri sağlayan sunuculardır. Her üst düzey alan adı (TLD), İnternet adreslerinin sonundaki en üst düzey etki alanlarını temsil eder. Örneğin, .com, .net, .org, .gov, .edu gibi popüler üst düzey alan adları vardır.

DNS Authoritative Server Nedir?

Authoritative name server (yetkili ad sunucusu), bir alan adının DNS kayıtlarını yöneten ve bu kayıtlara yönelik sorguları yanıtlayan sunucudur. Yetkili ad sunucusu, o alan adının DNS kayıtlarının kaynağıdır ve doğrudan bu kayıtlara erişebilir.



- 1- Tarayıcıya x.com yazdığımızda biz arka planda DNS server'a yönlendiriliyoruz. Amaç DNS server'dan x.com'un IP adresini almak. IP adresi doğrudan alamıyoruz. DNS server kendi içinde parçalara ayrılıyor. Bunlar: DNS Root Name Server, DNS Top Level Domain Server, DNS Authoritative Name Server
- 2- 3- Bu adımda x.com'un IP adresini DNS Root Server'da arıyoruz. DNS Root Server, Domain Name içeren soruyu kabul eder. Daha sonra cevap döndürür ve bu cevap domain'in uzantısına göre Top Level Domain Server'a gidecek şekilde şekillendirilir (.com, .net, .org, ...).
- 4- 5- Bu adımda DNS Root Server'dan gelen bilgileri, DNS TLD Server'a iletiyoruz. TLD Server, İnternet'teki DNS (Domain Name System) hiyerarşisinin üst düzey alan adlarına ilişkin bilgiler sağlayan sunuculardır. Her üst düzey alan adı (TLD), İnternet adreslerinin sonundaki en üst düzey etki alanlarını temsil eder (.com, .net, .org, ...). Daha sonra bize DNS TLD Server bize authoritative name server için gerekli bilgiler döndürüyor.
- 6- 7- Bu adımda Authoritative Name Server'a bir önceki adımdan gelen bilgileri iletiyoruz. Authoritative Name Server bir alan adının DNS kayıtlarını yöneten ve bu kayıtlara yönelik sorguları yanıtlayan sunucudur. Ayrıca hizmet verdiği alan adına (x.com) özgü bilgiler içerir. En sonunda IP Adresi Local DNS server'a yönlendirir ve IP Adres bu adımdan sonra DNS Server'da çözümlenmiş olur.
- 8- Bu adımda DNS Server'da çözümlenen IP Adres tarayıcıya döner.
- 9- Bu adımda DNS Server'dan gelen IP adresi kullanılarak tarayıcıdan web sunucusuna istek gönderilir.
- 10- Tarayıcıdan gelen istek doğrultusunda sayfa cevap olarak tarayıcıya geri döner. Cevap içerisinde linkler ve resimler gibi aynı sunucuda olan dosyalar bulunabilir.
- 11- Tabii ki bu aşamalar, IP Adresin cache (önbellek) içinde olmadığı zamanlarda geçerli. Eğer IP adres, tarayıcının ya da işletim sisteminin önbelleginde (cache) bulunuyorsa doğrudan IP Adres alınır ve tarayıcıdan web sunucusuna istek gönderilir. Yani IP Adres cache içinde bulunuyorsa DNS Server'ın 3 aşama ile iletişime geçmesine gerek yoktur. DNS server, bize doğrudan IP Adresi döndürür. Bu IP Adres ya browser'da bulunan cache ya da DNS Server'da bulunan cache üzerinden döner. Ancak cache geçici bir depolama alanıdır. Ayrıca, önbellette bulunan bir IP adresi her zaman doğru ve güncel olmayıpabilir. Dolayısıyla DNS Server içindeki 3 aşama tekrar gerçekleştirilecek IP Adres yeniden alınabilir.



Internet ve internet nedir?

Internet vs internet

Internet, dünya çapında bilgisayar ağlarının birbirine bağlılığı ve bilgi iletişimiminin gerçekleştiği global bir ağıdır. Internet, farklı coğrafi konumlar, bilgisayarlar, sunucular, cihazlar ve iletişim protokolleri aracılığıyla milyarlarca insanın ve bilgisayarın birbiriyle bağlantı kurmasını sağlar.

internet ise, birden fazla cihazların birbirleriyle bağlantılı olduğu ağı temsil eder. Yani Internet ve internet birbirlerinden farklı kavramlardır. Internet bir internet çeşididir. Buna ek olarak intranet ve extranet'te bir internet çeşididir. Internet ile karıştırılmamalıdır.

Extranet Nedir?

Extranet; bir şirketin veya kuruluşun, belirli dış paydaşlarla (tedarikçiler, müşteriler, iş ortakları vb.) paylaşılan bir ağ veya iletişim altyapısıdır. Extranet, şirket içi bilgi ve kaynakların, dış paydaşlarla güvenli bir şekilde paylaşılmasını sağlar.

Extranet, güvenlik önlemleriyle korunur ve genellikle kullanıcıları kimlik doğrulamasıyla sınırlar. Bu sayede, yalnızca yetkili kullanıcılar extranet üzerinden erişim sağlayabilir ve belirli verilere veya hizmetlere erişebilir. Ancak, extranetin gizliliği tamamen şirketin veya kuruluşun uyguladığı güvenlik politikaları ve önlemlerine bağlıdır.

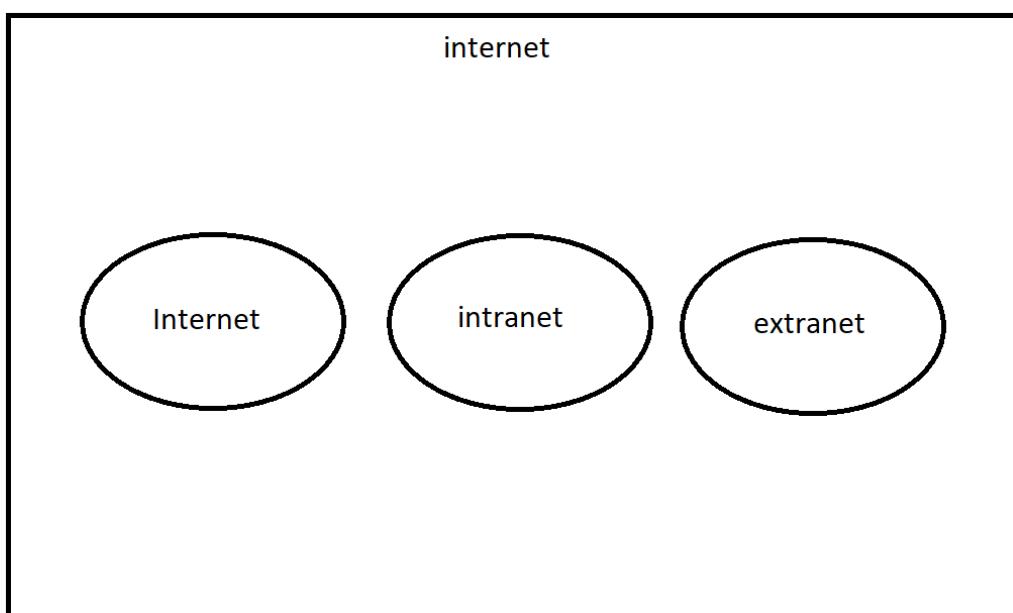
Intranet Nedir?

Intranet, bir şirketin veya kuruluşun iç ağı üzerinde çalışan, belirli bir kullanıcı grubuna özel olarak tasarlanmış bir iletişim ve bilgi paylaşım platformudur. Intranet; şirket içi iletişimini, işbirliğini, bilgi paylaşımını ve iş süreçlerini desteklemek amacıyla kullanılır.

Intranet, genellikle şirket içindeki çalışanların erişimine açık olan bir web tabanlı sistemdir. Şirketin iç ağı üzerinde çalışır ve kullanıcılar güvenli bir şekilde erişim sağlar. Kullanıcılar, bir tarayıcı kullanarak intranet üzerindeki web sayfalarına, belgelere, uygulamalara, iletişim araçlarına ve diğer içeriklere erişebilirler.

Intranet vs Extranet

Intranet şirket içindeki tüm çalışanlar için erişilebilirken, extranet hem şirket çalışanları hem de belirli dış paydaşlar (tedarikçiler, müşteriler, iş ortakları vb.) tarafından erişilebilir.



internet ⊃ Internet, intranet, extranet

Internet'in tarihçesi nedir?

Internet, günümüzde kullandığımız modern anlamda bir ağın ortaya çıkışının birçok farklı gelişme ve aşamadan geçmiştir. Internetin başlangıcı olarak kabul edilen tarih, 29 Ekim 1969'dur.

Bu tarihte, ABD'deki UCLA (University of California, Los Angeles) ve Stanford Research Institute (SRI) arasında ARPANET adı verilen bir bilgisayar ağı üzerinde ilk veri transferi gerçekleştirildi. Bu, iki bilgisayar arasında "LO" (Login) komutunun gönderilmesi amaçlanan bir deneydi.

Bu deneme, ARPANET projesinin temelini oluşturdu ve ARPANET, daha sonra geliştirilerek internetin temelini oluşturan TCP/IP protokolünün kullanıldığı bir ağa dönüştü.

ARPANET'in amacı ABD'nin askeriye açısından nükleer bir saldırıyla karşı merkezi bir sistemden dağıtık bir sisteme geçmek için finanse edilmiş bir sistemdir. Bunun yanında, bilgisayar bilimleri ve bilgi传递i alanlarında yeni teknolojilerin ve fikirlerin geliştirilmesini teşvik etmek amacıyla akademik araştırmaları da destekledi.

Dolayısıyla, internetin ortaya çıkış tarihi olarak 29 Ekim 1969 kabul edilir. Ancak, internetin yaygınlaşması ve günümüzdeki halini alması için daha birçok gelişme ve evrim geçirmesi gerekti. Bu nedenle, internetin tarihini tam olarak belirlemek zordur ve çeşitli kilometre taşıları ve önemli gelişmelerle ilgili birçok farklı tarih vardır.

Bir diğer kilometre taşı 1982 (1981'de işaret edilir) yılına işaret etmektedir. 1970'lerde oluşturulan ve kullanılmaya başlanan Transmission Control Protocol (TCP) ve Internet Protocol (IP) (TCP/IP) ARPANET için resmi protokol olarak tescillendi.

1960'lар: Internetin temelleri ABD'deki bilim adamları ve askeri araştırmacılar tarafından atıldı. ARPANET (Advanced Research Projects Agency Network), Amerika Birleşik Devletleri Savunma Bakanlığı tarafından finanse edilen bir proje idi ve bilgisayar ağları arasında veri iletişimini sağlamak amacıyla kullanıldı.

1970'ler: ARPANET, diğer üniversiteler ve araştırma kurumları tarafından da kullanılmaya başlandı. TCP/IP (Transmission Control Protocol/Internet Protocol) protokolü geliştirildi, bu protokol bugün hala internetin temelini oluşturur.

1980'ler: ARPANET ticari ağlara doğru genişlemeye başladı ve daha fazla insan tarafından kullanılmaya başlandı. E-posta, FTP (File Transfer Protocol) ve DNS (Domain Name System) gibi önemli internet hizmetleri ortaya çıktı.

1990'lar: World Wide Web (WWW) icat edildi ve internetin hızlı bir şekilde popülerleşmesini sağladı. Web tarayıcıları ve web siteleri oluşturuldu ve internet üzerinden bilgiye erişim daha kolay hale geldi.

2000'ler: Internet kullanımı büyük bir patlama yaşadı ve hızlı bir şekilde yaygınlaştı. Sosyal medya platformları, çevrimiçi alışveriş ve çevrimiçi video paylaşımı gibi internet hizmetlerinin popülerlik kazanmasıyla birlikte insanların internete erişimi daha da arttı.

2010'lar: Mobil internetin yükselişi gerçekleşti ve akıllı telefonlar aracılığıyla insanlar her zaman ve her yerde interneye erişebilme imkanına sahip oldu. Bulut bilişim, büyük veri ve yapay zeka gibi teknolojilerin gelişimi, internetin gücünü ve kullanımını daha da artırdı.

TCP/IP Nedir?

TCP, veri paketlerinin güvenilir bir şekilde iletilmesini sağlar. Veri paketlerinin doğruluğunu, eksiksizliğini ve sıralamasını kontrol eder. IP ise veri paketlerinin yönlendirilmesini ve internet üzerindeki farklı ağlar arasında iletilmesini sağlar.

TCP/IP'nin standartlaştırılması, ARPANET ve diğer bilgisayar ağları üzerindeki veri iletişimini etkin bir şekilde gerçekleştirmesini sağladı. Daha sonra, TCP/IP'nin kullanımı internetin büyümesiyle birlikte yaygınlaştı ve günümüzde hala internetin temel protokolü olarak kullanılmaktadır.

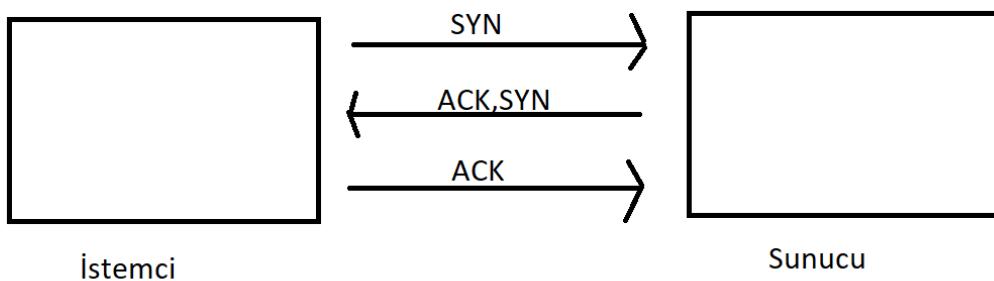
TCP/IP Nasıl Çalışır?

TCP Nasıl Çalışır?

TCP mesaj türleri

İleti	Açıklama
Syn	Bir bağlantı başlatmak ve kurmak için kullanılır. Sıra numaralarını cihazlar arasında senkronize etmenize de yardımcı olur.
ACK	Diğer tarafın SYN'yi aldığıni doğrulamaya yardımcı olur.
FIN	Bir bağlantıyı sonlandırmak için kullanılır.

Three way handshake (üç yollu el sıkışma)



1- İstemciden sunucuya durum kontrolü için Syn bayrağı gönderilir.

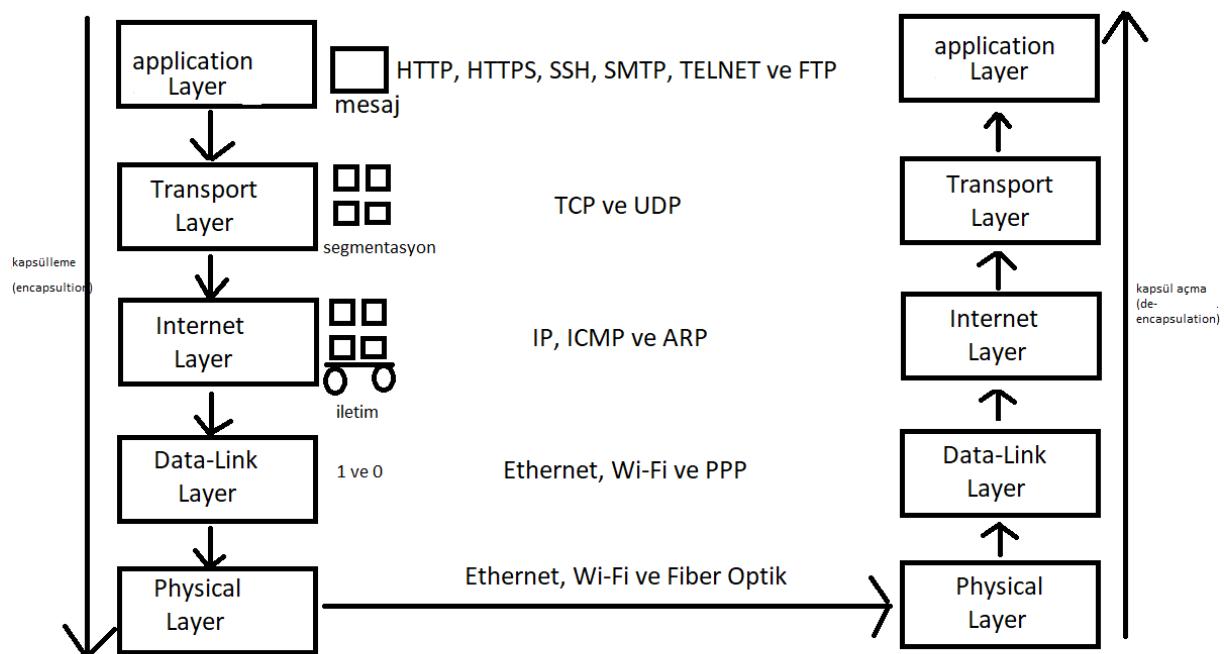
2- Server Syn bayrağını alırsa Ack bayrağı istemciye gönderir. Ardından bağlantı hazırlısa Syn bayrağı gönderilir (Sunucu -> İstemci).

3- Gelen bayrak istemciye ulaşırsa, istemci sunucuya ack bayrağı gönderir.

Three way handshake’te FIN bayrağı kullanılmaz. Bunun sebebi 3 way handshake bağlantı oluşturmak için kullanılır bağlantı sonlandırma işlemi yoktur. Bağlantı sonlandırmak için 4 way handshake kullanılır.

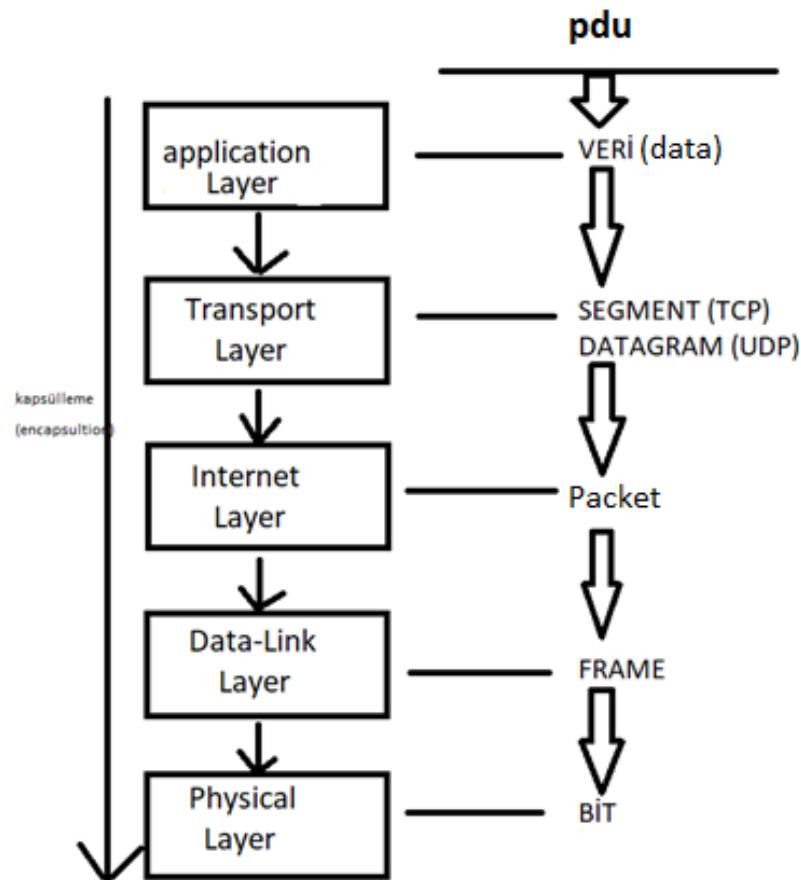
IP Nasıl Çalışır? (?)

TCP/IP Nasıl Çalışır? (TCP/IP üzerinden mesaj iletimi)



- 1- Kullanıcı uygulamaları ve ağ arasında iletişimini sağlar. Kullanıcıların ağa erişimini sağlayan protokoller ve hizmetler bu katmanda yer alır. Mesaj bu katmanda oluşturulur.
- 2- Veri iletiminin güvenilir ve hata kontrolü yapılarak gerçekleştirilmesinden sorumludur. Veri bölme, akış kontrolü ve hata kontrolü gibi işlemleri sağlar. Mesaj bu katmanda segmentasyon geçirir.
- 3- Veri paketlerinin kaynak noktasından hedef noktasına iletmesini sağlar. IP adreslemesi, yönlendirme ve paketleme işlemleri bu katmanda gerçekleştirilir.
- 4- Ağ üzerindeki fiziksel bağlantıyı sağlar ve veri paketlerini ağdaki cihazlar arasında iletmek için çerçevelere dönüştürür. Bilgisayar dili 0 ve 1'lerden oluşmaktadır ve bu katmandaki iletişim için veri paketleri 0 ve 1'lere dönüştürüllererek taşınır.
- 5- Fiziksel katmanda veriyle birlikte paketlenen çerçeve (frame); elektrik sinyalleri, optik ışık dalgaları veya radyo dalgaları gibi fiziksel ortam üzerinden ilettilir.
- 6- Veri karşıya iletildikten sonra aynı işlemler ters sırada gerçekleşerek veri uygulama katmanında insan için anlamlı hale döner.
- 7- Üç cihazlar (hostlar) arasında veri iletildikten sonra her katmandan ilerledikçe her katmanda belirli bilgiler eklenir. Gönderici bilgisayar için yukarı katmandan aşağı katmana doğru veri ilerler ve her bir katmanda belirli bilgiler eklenir ve aşağı katmana ilettilir. Bu işleme kapsülleme (encapsulation) denir. Aynı şekilde alıcı bilgisayar için aşağı katmandan yukarı katmana doğru veri ilerler ve her bir katmanda eklenmiş olan belirli bilgiler çıkarılır ve üst katmanlara ilettilir. Bu işleme de kapsül açma (de-encapsulation) denir. Verinin her bir katmanda aldığı biçimde PDU (Protocol Data Unit / Protokol Veri Birimi - PDU terimi,

katmanlar arasındaki veri aktarımında her katmanın kendi içinde taşıdığı veri birimini ifade eder) olarak adlandırılır. Kapsülleme işleminde her bir katman üst katmandan aldığı PDU'yu kullanılan protokole göre kapsüller. Ve her bir katmanda kapsüllenmiş bu PDU'nun ismi artık farklıdır.



- 8- Bazı yerlerde TCP/IP 4 katman bazı yerlerde 5 katman olarak gösteriliyor. Bunlar şu şekilde gösteriliyor:

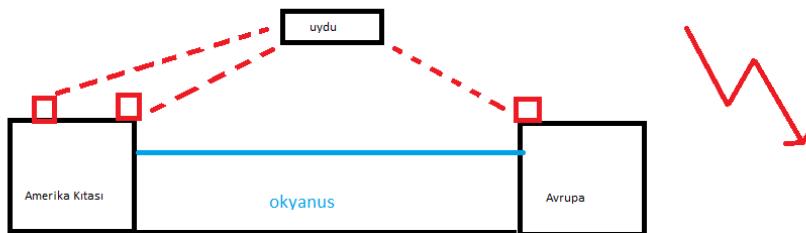
<table border="1"> <tr><td>Application</td></tr> <tr><td>Transport</td></tr> <tr><td>Internet (Network)</td></tr> <tr><td>Network Access</td></tr> </table>	Application	Transport	Internet (Network)	Network Access	<table border="1"> <tr><td>Application</td></tr> <tr><td>Transport</td></tr> <tr><td>Internet (Network)</td></tr> <tr><td>Data Link</td></tr> <tr><td>Physical</td></tr> </table>	Application	Transport	Internet (Network)	Data Link	Physical
Application										
Transport										
Internet (Network)										
Network Access										
Application										
Transport										
Internet (Network)										
Data Link										
Physical										

Internet neden kullanılır?

- 1- İletişim: Internet; Email, ani mesajlaşma, video konferansı, sosyal medya platformları ve VoIP (Voice over IP) servisleri gibi çeşitli iletişim kanalları sunmaktadır. Yani dünya üzerinde herhangi birisiyle iletişim kurmak için kullanılmaktadır.

- 2- Bilgi ve Araştırma: Internet herhangi bir konuda sanal olarak çok fazla bilgi kaynağı sunmaktadır. İnsanlar bilgiye erişmek, yeni yetenek kazanmak ve güncel kalmak için arama motorları, online kütüphaneler, haber web siteleri ve eğitim kaynakları kullanırlar.
- 3- Eğlence: Internet oldukça fazla eğlence seçeneği sunmaktadır. İnsanlar yayın servislerini film, dizi, Show ve müzik için kullanabilirler. Online oyunlar; multiplayer deneyim ve rekabetçi oyun sunmaktadır. İçerik paylaşmak, başkalarına bağlanmak ve yeni ilgiler keşfetmek için sosyal medya platformları kullanılabilir.
- 4- E-Ticaret: Internet, insanların alış veriş yapmasında devrim oluşturdu. Online market yerleri ve E-Ticaret siteleri, insanları ürün araştırmak ve almak konusunda evlerinden konfor alanları içerisinde gerçekleştirmesine izin verdi. Online bankacılık ve dijital ödeme sistemleri de bu alanda kullanıldı.
- 5- Eğitim ve Öğretim: Online kurslar ve eğitim platformları bilgiye erişmek ve yetenek geliştirmek için bir yöntem haline geldi. Şu an siz de bilgi öğrenmek için Youtube platformunda bu kursu izliyorsunuz. Bir başka örnek karantina döneminde eğitim ve öğretim online şekilde gerçekleştirilmiştir.
- 6- İş ve Üretkenlik: Uzaktan iş, telekomünikasyon ve sanal işbirliği araçları kişi ve takımlara herhangi bir yerden çalışmasına izin verdi.
- 7- Devlet Servislerine Erişim: Bir çok devlet servisleri ve bilgiler artık online erişilebiliyor (E-devlet gibi). Bu sayede zamandan tasarruf ve kolaylık sağlandı.

Internet nasıl çalışır?



Fiber optik kabloların alternatifleri:

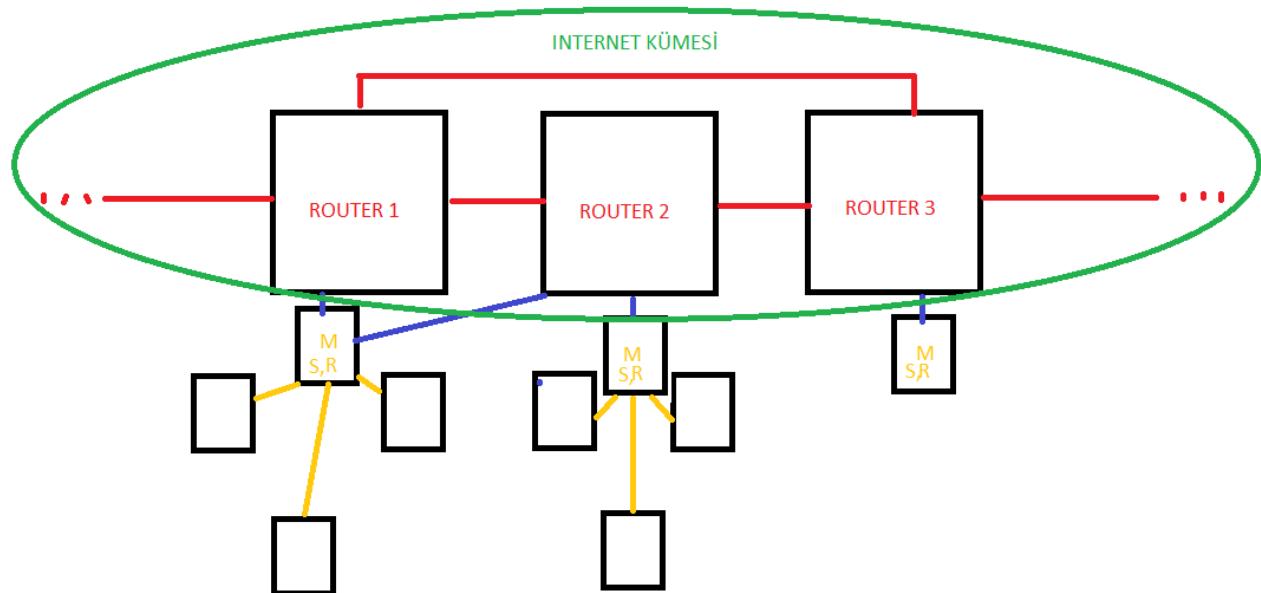
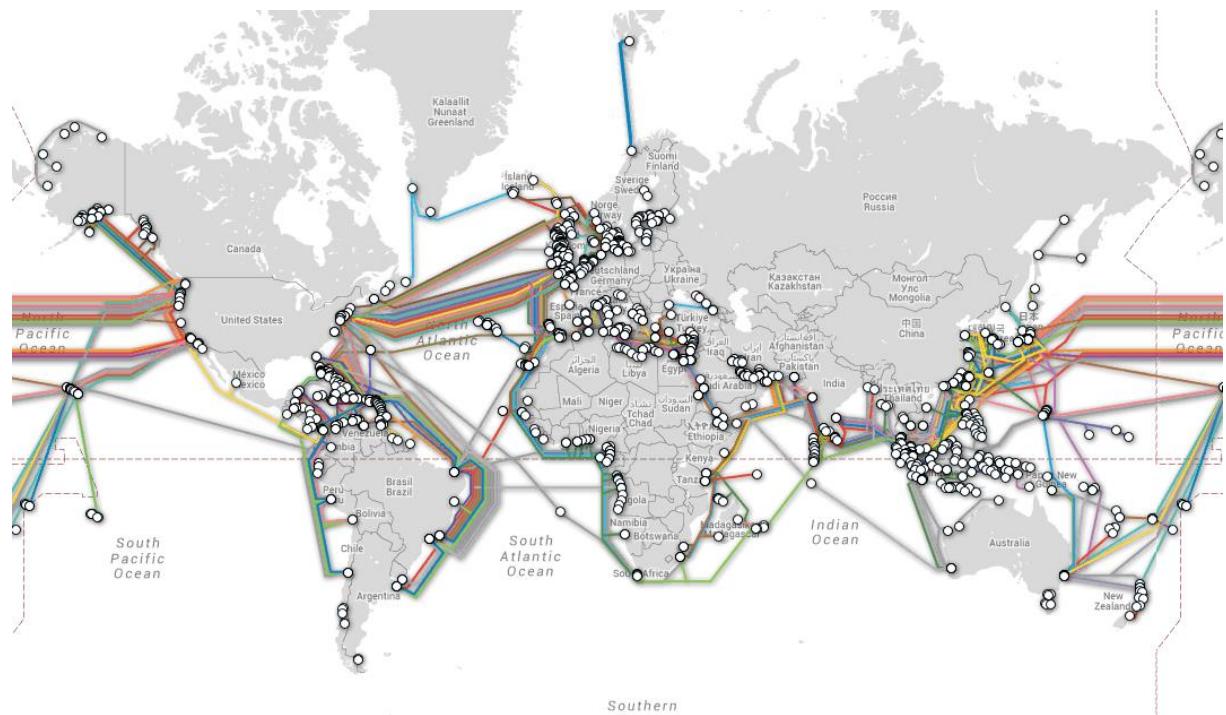
Kablosuz bağlantı (örnek-uydu): Uydu sistemlerinde kabloya gerek kalmadan bir bağlantı sağlanabilir. Ancak gecikmeler olacağı için (delay) bağlantı hızında sorun oluşabilir.

UTP - Bakır kablo: Birbirlerine dolandıkları için manyetik alan oluşturur. Oluşan manyetik alan başka bir kabloya değdiği zaman yine manyetik alan oluşur. Tersine Çevrilen Çiftler (Twisted Pair), Çapraz Konuşma (Crosstalk) Önleme, Yalıtım ve Örgüsüz Tasarım ve Güvenli Mesafe ile bu manyetik alan sorunu çözülebilir.

BNC - koaksiyel kablo: İçeride bakır tel ve dışarıda metal bulunur. Böylece sinyal dinlenmesinin önünde geçilir. Ancak yapısında yine bakır tel bulunduğu için fiber optik kablolardan çok daha yavaştır.

Görüleceği üzere güncel olarak aralarında en hızlı bağlantı fiber optik kablolarda bulunmaktadır. Bunun sebebi fiber optik kablolardan veriyi yüksek hızlarda ve uzun mesafelerde ışık sinyalleri olarak ileten bir iletişim kablo türüdür. Dolayısıyla fiber optik kablo karada, deniz ve okyanus altında kullanılabilir.

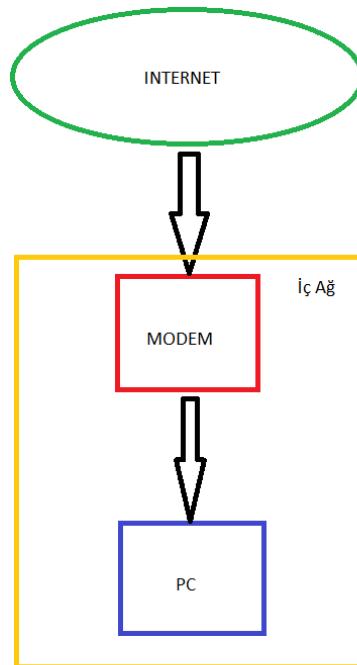
Deniz altında kullanılan fiber optik kablolardan:



Yukarıdaki resimde Internet'in merkezi değil de dağıtık bir sistem olduğunu göstermektedir. Internet'in bir router kümesi olduğunu düşünelim. Router 1 ve router 2 arasındaki ağ koptuğunda iletişim router 1 ve router 3 arasında bulunan ağ üzerinden gerçekleşecektir. Böylece ağda kopma yaşanmayacak. Fakat router 3 ile altındaki modem + router + switch aracılığıyla oluşan bir iç ağda mavi ağın koptuğunu düşünelim. Bu durumda o iç ağda kopukluk meydana gelir. Ancak router 1 altındaki mavi ağ ile bir kopukluk olduğu zaman router 2'den mesaj iletimi devam edeceği için router 1 altındaki iç ağda bir kopma olmaz.

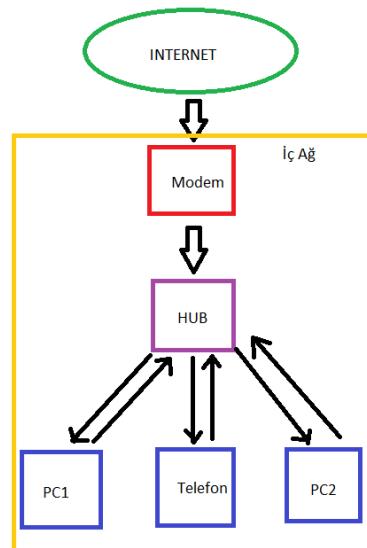
Modem vs Hub vs Switch vs Router

Modem:



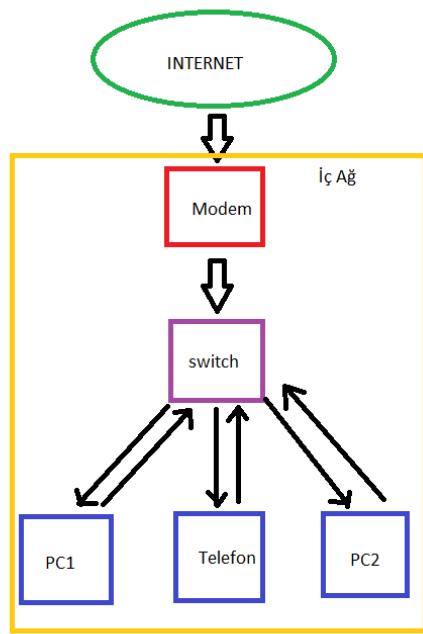
Modem dış ağdan yani Internet'ten gelen veriyi (sinyali) iç ağa yönlendiren bir köprü olarak çalışır. Eğer iç ağda tek bir cihaz var ise hub veya switch gibi cihazlara ihtiyaç duyulmaz. Ancak birden fazla cihaz varsa bu cihazlardan birine (switch, hubdan daha iyi bir çözüm) ihtiyaç duyulur.

Hub:



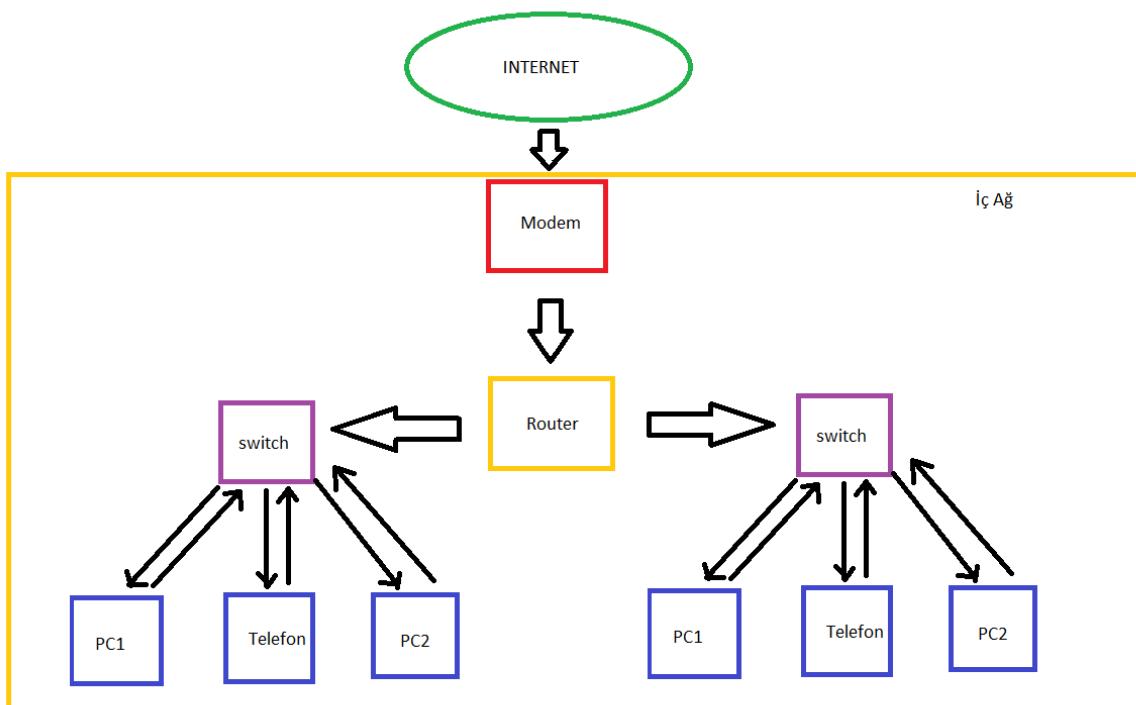
Dışarıdan gelen sinyal modem aracılığıyla iç ağa girdiği zaman hub ile karşılaşıyor. Burada hub gibi bir cihaz kullanmamızın sebebi birden fazla cihaz kullanmadan kaynaklanıyor. Gelen sinyalin herhangi bir cihaza gitmesi için hub gibi yönlendirici cihaz kullanmamız gereklidir. Ancak Hub gelen sinyali cihaz fark etmeksiz bütün cihazlara yönlendiriyor. Bu noktada karşımıza switch çıkıyor.

Switch:



Dışarıdan gelen sinyal modem aracılığıyla iç ağa girdiği zaman hub'da olduğu gibi switch ile karşılaşıyor. Burada switch kullanmamızın sebebi hub'un aksine gelen sinyal istenen cihaza yönlendiriliyor. Bunu mac adresleri (fiziksel adres) kullanarak yapıyor. Fakat modern switchler IP adres de kullanabilir.

Router:



Dışarıdan gelen sinyal ilk olarak modemden geçiyor. Router, modemden sonra eğer farklı iki veya daha fazla ağ varsa (farklı subnetler de olabilir) onları birbirlerine bağlar. Yani iki farklı ağ arasındaki trafiği yönetir. Switch1 ve Switch2'nin bulunduğu ağları iki farklı subnet kabul edelim (iki farklı ağ da olabilir). Switch1'deki PC1'den, Switch2'deki PC1'e mesaj iletebilmeyi sağlayan şey aradaki router'dır. Örneğin, Switch1'deki PC1'den Switch2'deki PC1'e bir mesaj gönderilmek istendiğinde, öncelikle hedef cihazın MAC adresi belirlenir. Daha sonra router, IP adresleri kullanarak paketi doğru subnetteki switch'e yönlendirir. Switch mac adres kullanarak mesaj ileter demistik. Router ise IP adres kullanarak aracılık yapar. Tabii ki bu eski sistemler için geçerli. Modern cihazlar birden fazla işlev sahip olabilir (Modern Switch'in IP ve MAC ile çalışması gibi).

$2^3 \ 2^2 \ 2^1 \ 2^0$



1011 (ikilik/binary) => 11 (onluk/decimal)



$$(2^3 * 1) + (2^2 * 0) + (2^1 * 1) + (2^0 * 1) = 8 + 0 + 2 + 1 = 11$$

SUBNET MASK

TYPE - A => 255.0.0.0

TYPE - B => 255.255.0.0

TYPE - C => 255.255.255.0

192.168.1.100 => 11000000.10101000.00000001.01100100

255.255.255.0 => 11111111.11111111.11111111.00000000

AND

11000000.10101000.00000001.00000000



192.168.1.0/24

A tipi, B tipi ve C tipi olmak üzere bizim 3 ana subnet maskimiz vardır. Aslına bakacak olursak IPv4 için 2^{32} ’den 4.294.967.296 subnet maske sahibiz. Neden 2^{32} diye soracak olursak: IPv4 8,8,8,8 olmak üzere 32 bit uzunluğa sahip ve bu bitler ya 1 ya da 0 değeri alacak. Yani 2 farklı değer alabilir. Bunu her bir bit ile çarparsa, yani 32 kere 2’yi çarparsa 2^{32} gibi bir değer ortaya çıkıyor. Teorik olarak 4.294.967.296 farklı subnet maskesi kombinasyonu bulunsa da, pratikte yaygın olarak kullanılan ve kullanışlı olan birkaç standart subnet mask vardır.

En yaygın kullanılan subnet masklerden; A tipi subnet mask 255.0.0.0, B tipi subnet mask 255.255.0.0 ve C tipi subnet mask 255.255.255.0 değerlerinden oluşuyor. Biz local IP adresimizi herhangi bir subnet mask ile and işlemeye tabi tutarsak elimize subnet adresimiz gelir. Eğer iki cihaz farklı subnet’lerde olursa, aralarında mesaj iletimi gerçekleşebilmesi için router gereklidir. Çünkü farklı alt ağdalar.

Yukarıdaki örneğe baktığımızda 192.168.1.100 local IP adresli cihazımızı C tipi subnet mask yani 255.255.255.0 ile and işlemi uyguluyoruz. Tabii ki bu işlemi binary code yani ikilik tabanda yapıyoruz. Bize gelen sonuç ikilik tabanda 192.168.1.0 değerine karşılık geliyor. Biz de bunu 192.168.1.0/24 ile temsil ediyoruz. 24 ile temsil etmemizin sebebi 255.255.255.0 subnet maskini kullanmamızdır. Yani 24 tane 1’den oluşmasıdır. Bu da 8 tane 1, 8 tane 1 ve 8 tane 1’den oluşmasıdır. Eğer 255.255.0.0 subnet maski ile oluşsaydı /16 ile gösterecektik. Yani 8 tane 1 ve 8 tane 1’den oluştuğu için. Yine farklı bir örnek 255.255.255.128’den oluşsaydı (11111111.11111111.11111111.10000000) /25 ile ifade edecektik.

Ayrıca fark ettiyseniz a,b ve c tipi subnet masklerde eğer 4 parçadan birisi (8 bit – 1 byte) 255 den oluşuyorsa, and işlemi uygulayacağımız IP adresin değeri aynı kalıyor. Yani 255 ile 192 and’lediğimizde 192, 255 ile 168 and’lediğimizde 168, 255 ile 1 and’lediğimizde 1 değeri alıyor. Eğer subnet mask’in 0 değeri ile and’lenirse ne olursa olsun subnet değerinde orası 0 oluyor. Yani 0 ile 100 and’lediğimizde 0 değerini elde ediyoruz. Tabii ki bu ezbere çok aldanmamak lazım ama kısaca hesaplamak istediğimizde bu sonuçları elde edebiliriz. Uzun yolda ise daha önce de belirttiğimiz gibi subneti elde etmek için, IP adres ile subnet mask’i ikilik değerde and işlemeye tabi tutmamız gerekiyor.

A tipi subnet mask ile
10.0.0.1 gibi bir IP adres
olmalı

B tipi subnet mask ile
172.16.0.1 gibi bir IP adres
olmalı

192.168.1.100	192.168.1.110	192.168.2.100	192.168.1.120
255.255.255.0	255.255.255.0	255.255.255.0	255.255.0.0
192.168.1.0/24	192.168.1.0/24	192.168.2.0/24	192.168.0.0/16

AND (binary code ile)

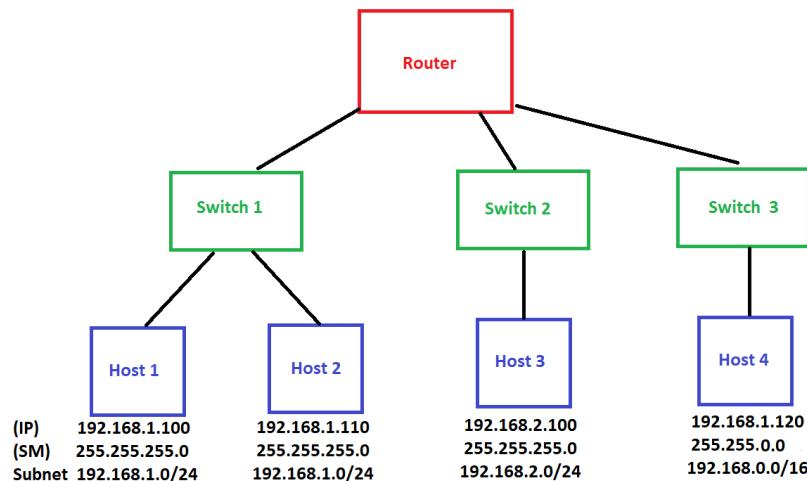
Bu örnekte ise farklı veya aynı subnet’tे olan cihazların iletişimini inceleyeceğiz. İlk örneğe baktığımızda 192.168.1.100 IP adresli cihazı C tipi subnet mask yani 255.255.255.0 ile and’lıyoruz. Sonuçta 192.168.1.0/24 subnetini elde ediyoruz. İkinci örnekte ise 192.168.1.110 IP adresli cihazı yine C tipi subnet mask yani 255.255.255.0 ile and’lıyoruz. Sonuç olarak 192.168.1.0/24 subnetini elde ediyoruz. Üçüncü örnekte bu sefer IP adresimizde küçük bir değişiklik oluyor. 192.168.2.100 IP adresini C tipi subnet mask yani 255.255.255.0 ile and’lıyoruz. Sonuç olarak 192.168.2.0/24 subnetini

elde ediyoruz. Son yani dördüncü örnekte ise 192.168.1.120 adresini bu sefer B tipi subnet mask yani 255.255.0.0 ile and'lıyoruz. Sonuç olarak 192.168.0.0/16 subnetini elde ediyoruz.

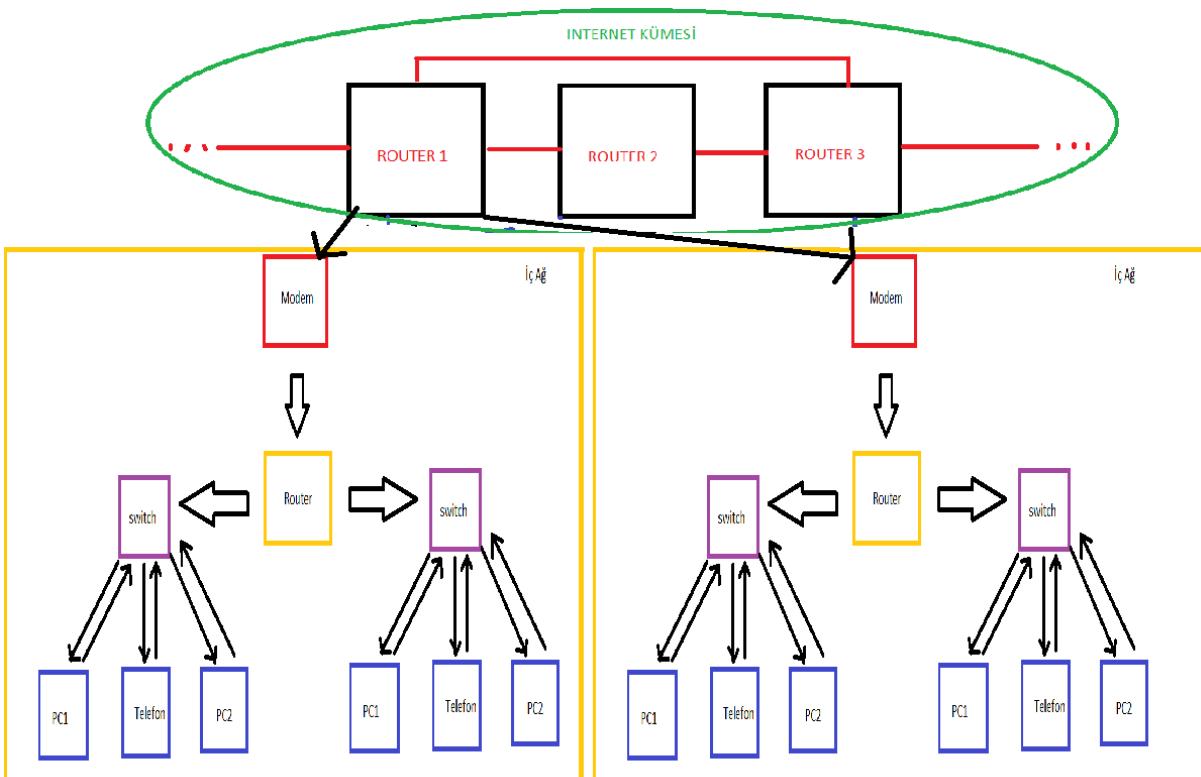
Aslında B tipi subnet mask'e sahip cihaz 192 ile başlayan bir IP adrese sahip olamaz. Ama bu örnekte anlaşılsın diye böyle gerçekleştirdik. Neden sahip olamaz diye soracak olursak cevap aşağıdaki tabloda yatıyor. Tabii ki %100 bu kural geçerli diye bir şey yok. Arada istisnalar olabilir.

Subnet Mask	Sahip olabileceği IP Aralığı
Class A	0.0.0.0 - 127.255.255.255
Class B	128.0.0.0 - 191.255.255.255
Class C	192.0.0.0 - 223.255.255.255

Sonuç olarak 1. Cihaz sadece 2. Cihaz ile aynı subnet'te olduğu için router'ın bir bacağından doğrudan mesajlaşabilir. 1. Cihaz, 3 ve 4 ile aynı subnet'te değil. Yine router ile iki bacaktan farklı subnet'lere çıkararak birbirleriyle mesajlaşabilirler. Aşağıdaki resimde örnekte anlatılanlar resmedilmiştir.



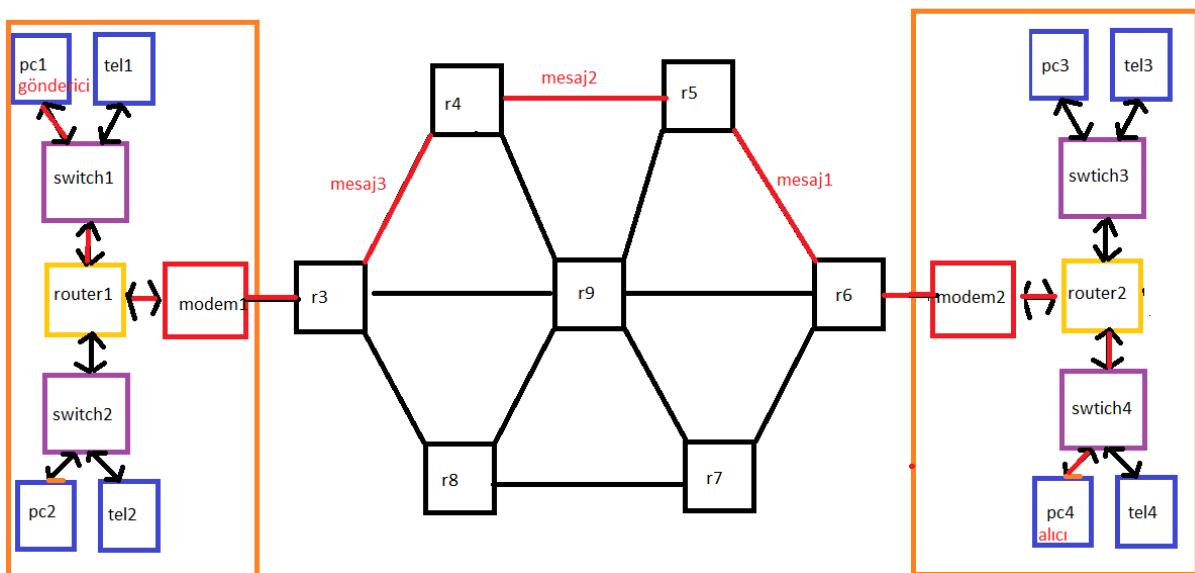
Dikkat! Bu örnekteki/görseldeki değerler daha iyi anlaşılsın diye farzi oluşturulmuştur.



Router konusuna geri dönecek olursak, yukarıdaki resimde görüldüğü gibi Router aynı zamanda modemden sonra da konumlanabilir. Yani iki farklı iç ağ birbirine bağlamak için kullanılabilir. Bu görselde router1 iki iç ağın modemine de bağlanarak iki ağ içerisinde köprü görevi görmektedir. Örneğin bu router ISP (Internet Service Provider – Turk Telekom, Vodafone ...) olabilir.

Circuit Switching vs Message Switching vs Packet Switching

Circuit Switching



Circuit Switching, veri iletimi için ayrılmış ve önceden belirlenmiş bir fiziksel bağlantı üzerinden gerçekleştirilen iletişim yöntemidir. Bu yöntemde, iki cihaz arasında veri alışverişi yapmak için bir bağlantı önceden oluşturulur ve tüm veri bütün bu bağlantı üzerinden iletilir. Veriler, birçok küçük paket yerine, bir sürekli devre üzerinden aktarılır. Yani bütün bir parça ile iletişim söz konusudur.

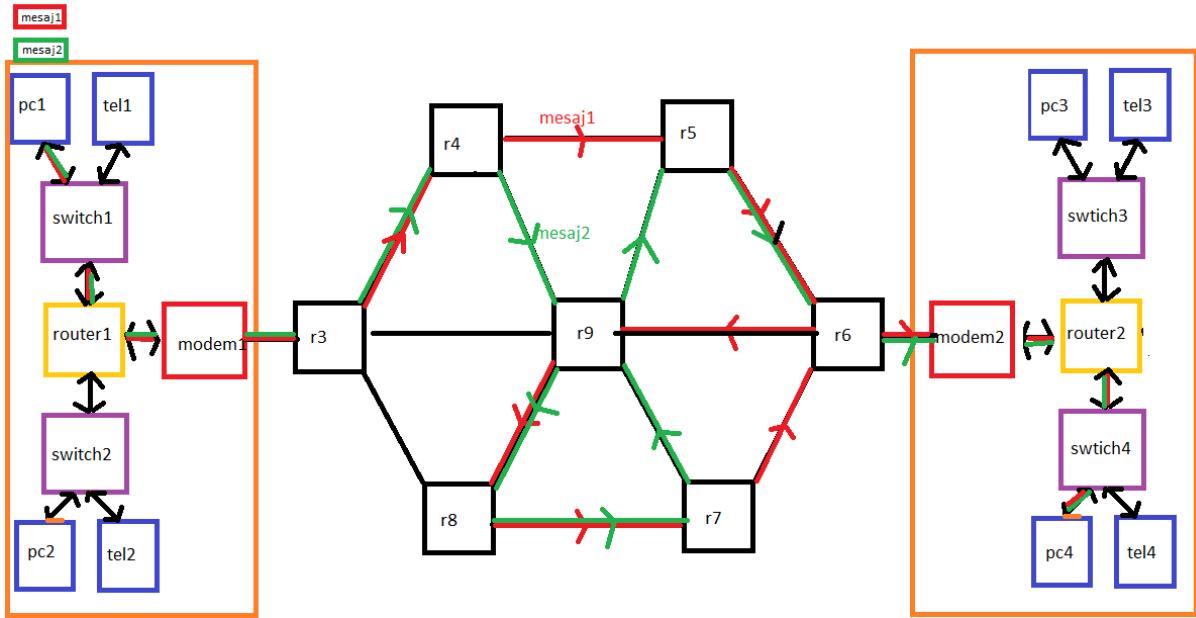
Çalışma evreleri

- 1- Bağlantı oluşturma: Önceden hazırlanan yol ile iki cihaz arasında bağlantı oluşturulur.
- 2- Devre iletimi: Bağlantı oluşturulduktan sonra, tüm veri bu devre üzerinden iletilir. Gönderilen veri paketleri, aynı yolu takip eder ve her seferinde aynı rotayı kullanır.
- 3- Bağlantı Sonlandırma: Veri alışverişi tamamlandıktan sonra, bağlantı sonlandırılır ve ayrılmış fiziksel bağlantı serbest bırakılır.

Telefon ağ sistemleri Circuit Switching için en bilinen örneklerden birisidir. Bir telefon görüşmesi sırasında, iki telefon arasında bir bağlantı oluşturulur ve görüşme boyunca bu bağlantı sabit kalır. Bu sayede, görüşme süresince ses veya veri kesintisiz bir şekilde iletilir. ((kesintisiz) çok iyi bir sistem olduğu için değil bütün iletişim olduğu için)

Verimli değildir. Çünkü tüm veri taşınır. Bu yönteme alternatif olarak message switching ortaya çıkmıştır.

Message Switching



Message Switching, veri iletimini bir mesajın tümü olarak gerçekleştiren bir iletişim yöntemidir. Bu yöntemde, veriler mesajlar halinde iletilir ve tüm mesaj ağ üzerinde farklı rotaları takip ederek hedefe ulaşır. Message Switching'in çalışma prensibi, verilerin tamamını bir mesaj olarak ilettiği için verilerin ağ üzerinde adım adım ilerlemesini ve bir ağ düğümünden diğerine geçerek hedefe ulaşmasını sağlar.

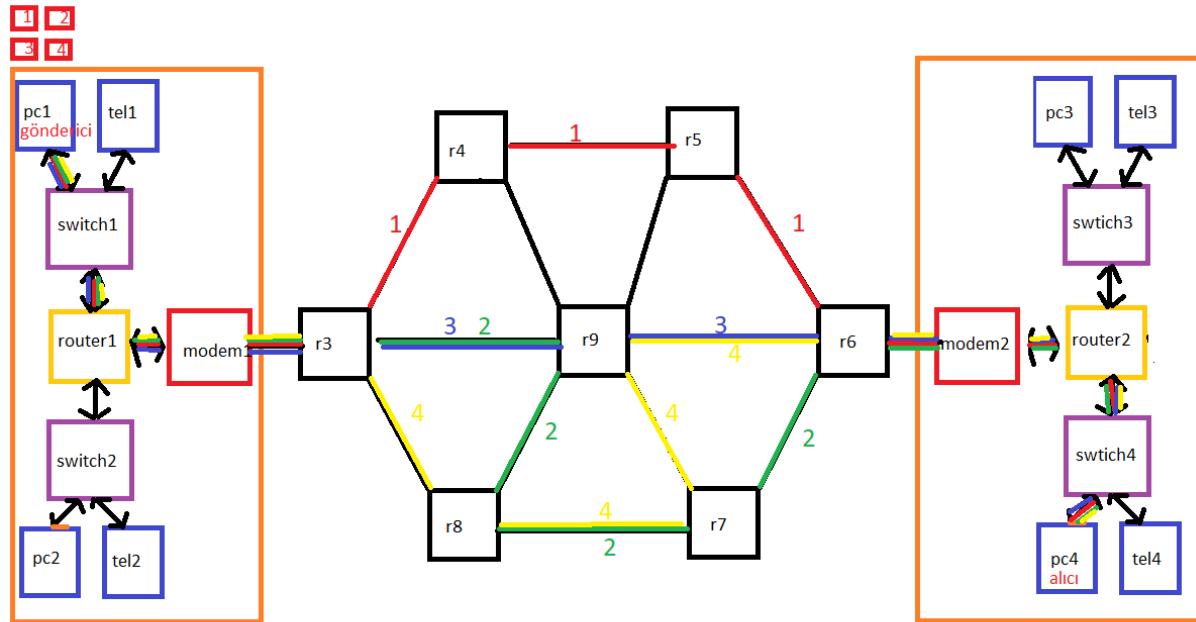
Çalışma Evreleri

- 1- Mesaj Parçaları: Gönderilecek veri, bir mesaj olarak bölümlenir ve mesaj parçaları oluşturulur. Her bir mesaj parçası, verinin bir parçasıdır ve tüm mesajın tamamını içeren bir parçadır.
 - 2- Yol Seçimi: Mesajlar, ağdaki trafik durumuna ve ağ düğümlerinin durumuna göre farklı yolları takip eder. Yol seçimi, mesajın tamamı için önceden belirlenmez ve dinamik olarak belirlenir.
 - 3- Adım Adım İletim (hop-by-hop): Mesaj parçaları, hop-by-hop yöntemiyle bir ağ düğümünden diğerine ilerler. Her bir ağ düğümü, aldığı mesaj parçasını tamamen işler ve bir sonraki ağ düğümüne yönlendirir. Mesaj her bir düğümde depolanır. Böylece bir sonraki düğüm doluya mesaj iletimi sekteye uğrar. Bu da yavaşlığa sebep olabilir.
 - 4- Mesajın Birleştirilmesi: Hedef cihaz, aldığı mesaj parçalarını birleştirerek tam mesajı elde eder. Tüm mesajın ağ üzerinde birleştirilmesi, hedef cihazda gerçekleşir.

Ancak, Message Switching'in de Circuit Switching'e göre bazı dezavantajları vardır. Message Switching, paketlerin tamamını gönderdikten sonra hedefte birleştirme gerektirdiği için iletişim süresini uzatabilir. Ayrıca düğümlerde depolama olduğundan dolayı o da yavaşlığa sebebiyet verebilir.

Bir diğer husus Message switching'de mesaj parçası her bir düğüme uğrayacağı için veri iletim hızı büyük ölçekli ağlarda yavaş olacaktır ve ağdaki kaynakları verimli kullanmaz. Bu yüzden alternatif olarak packet switching metodu çıkmıştır.

Packet Switching - Datagram

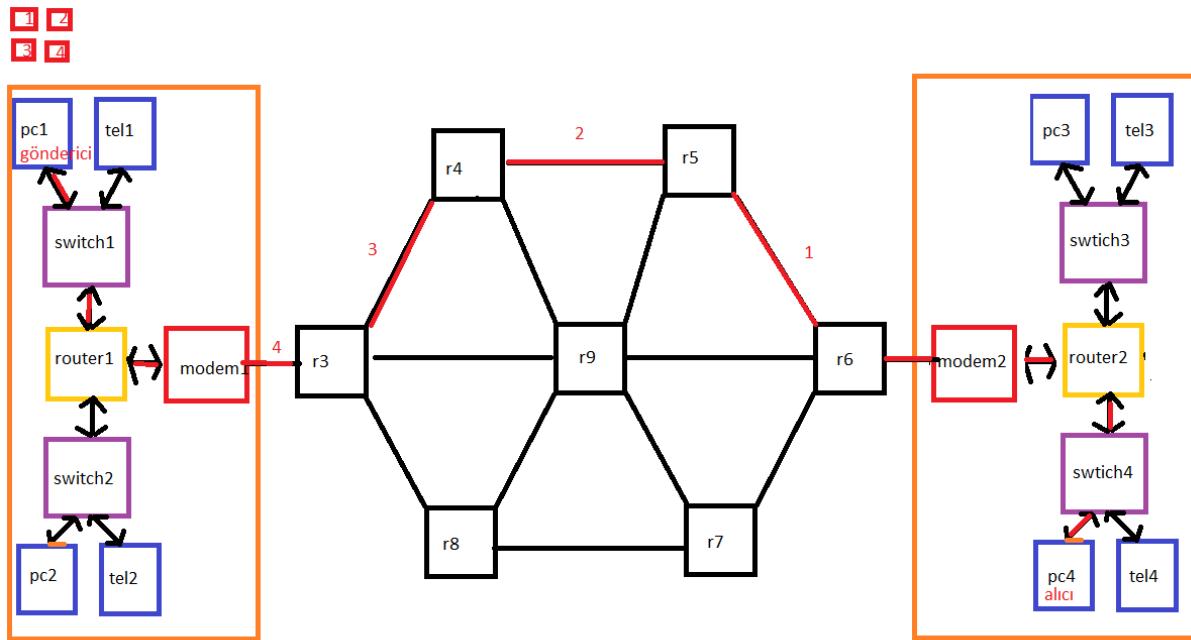


Datagram Packet Switching, Packet Switching yönteminin bir alt türüdür. Bu yöntemde, veri paketleri (datagramlar) bağımsız olarak ağ üzerinde iletilir ve her bir paket, hedefe ulaşmak için farklı yolları takip edebilir. Datagram Packet Switching, paketleri birleştirmeden (bir önceki paketlerle ilişkilendirme yapmadan) gönderir ve her bir paketin kendi başına hedefe ulaşması sağlanır. UDP protokolü Datagram Packet Switching'in bir örneğidir. Özellikle zaman duyarlı ve kayıpsızlık gerektirmeyen uygulamalarda kullanılır, örneğin ses ve video akışları (internet ve geniş alan ağları).

Çalışma Evreleri

- Bağımsız Paketler:** Veri, küçük veri paketlerine (datagramlara) bölünür ve her bir paket, kendi başına bağımsız bir şekilde ağ üzerinden iletilir. Paketler, birbirleriyle ilişkilendirilmeden, her birinin kendi hedefine ulaşması için farklı yolları takip edebilir.
- Rotalama:** Paketler, ağ üzerinde farklı rota ve bağlantılar kullanılarak hedefe ulaştırılırlar. Her bir paket, ağdaki trafik durumuna ve paketin gönderildiği sırada mevcut olan ağ durumuna göre belirlenen yolu takip eder. Paketler farklı pathler üzerinden iletilir.
- Yönlendirme Tablosu:** Rotalama işlemi, her ağ düğümünde bulunan yönlendirme tablosu sayesinde yapılır. Yönlendirme tabloları, paketlerin hedef adreslerine göre hangi yolun izlenmesi gerektiğini belirler. (Daha sonra buna değineceğiz)
- Birleştirme:** Datagram Packet Switching'de; her paketin kendisi hedefine ulaşır ve hedef cihazı, alınan paketleri doğru sıra ve hedef adresine göre birleştirerek orijinal veriyi elde eder.

Packet Switching – Virtual Circuit



Virtual Circuit Packet Switching (Sanal Devre Paket Anahtarlaması), Packet Switching yönteminin bir alt türüdür. Bu yöntemde, veri paketleri (datagramlar) bir sanal devre oluşturularak ağ üzerinde iletilir. Sanal devre, veri paketlerinin belirli bir sıra ve yönlendirme bilgisiyle iletimini sağlar. Virtual Circuit Packet Switching'e göre, paketlerin her biri kendi başına bağımsız bir şekilde ağ üzerinden iletilmez; bunun yerine, bir sanal devre boyunca belirli bir yol izlenir. Özellikle zaman duyarlı ve kayıpsızlık gerektiren uygulamalarda tercih edilir (ses ve video konferans).

Çalışma Evreleri

- 1- **Sanal Devre Oluşturma:** İlk iletişimde, kaynak cihaz ve hedef cihaz arasında sanal bir devre oluşturulur. Bu sanal devre, veri paketlerinin传递 için bir yol belirler. Sanal devre oluşturma aşamasında, her bir ağ düğümünde bulunan yönlendirme tabloları güncellenir ve belirli bir paketin sanal devre boyunca izleyeceği yol belirlenir.
- 2- **Sabit Yol:** Sanal devre boyunca, paketler belirli bir sıra ve yönlendirme bilgisiyle iletilir. Her bir paket, sanal devrenin belirdiği yolu ve aynı sanal devre üzerinden gönderilir. Bu sayede, paketlerin belirli bir sıra ve yönlendirmeyeyle传递 sağlanır.
- 3- **Yönlendirme Tabloları:** Sanal devre oluşturma aşamasında, ağdaki her bir düğümde yönlendirme tabloları güncellenir ve belirli bir paketin hangi sanal devre üzerinden gönderileceği belirlenir. Bu sayede, paketlerin sanal devre boyunca izleyeceği yol belirlenir.
- 4- **Veri传递:** Sanal devre oluşturulduktan sonra, veri paketleri belirli bir sıra ve yönlendirmeyeyle iletilir. Her bir paket, belirli bir sanal devre üzerinden gönderilir ve belirlenen yolu izler.
- 5- **Veri birimleri (datagram) hedef cihazda birleştirilir ve veri karşıya gitmiş olur.**

Internet kullanımında veya geniş çaplı ağlarda neden packet switching (datagram packet switching) kullanılır?

Circuit Switching yöntemine baktığımızda verinin iletimi Internet için kullanışız olacaktır. Bunun sebebi veri parçalanmayacak ve boyutu büyük olacak. Asıl sebep ise veri gönderilmeden önce yol (path) belirlendiği için alternatif yol oluşmayacak. Bu da kullanışsızlığa yol açacak.

Message switching ise baktığımız zaman packet switching'in özelliklerini taşıyor. Message switching neden olmasın diye sorarsak şu cevap ortaya çıkacak: Veri, mesaj parçalarına bölünüp teker teker yollanıyor fakat mesajlar paket kadar küçük olmayacağı. Dolayısıyla boyut problemi daha çok yaşanacak. Ancak asıl sorun bu değil. Veri mesajlar halinde düğümden düğüme hop-by-hop yöntemiyle taşınıyor. Bu da packet switching metoduna benziyor. Fakat sorun şu: Message switching yönteminde mesaj her bir düğüme uğruyor. Dolayısıyla veri iletim hızında düşüş meydana gelecek. Ayrıca mesajımız düğüm noktalarında depolandığı için depolama sıkıntısı ortaya çıkabilir (bu packet switching'de de var).

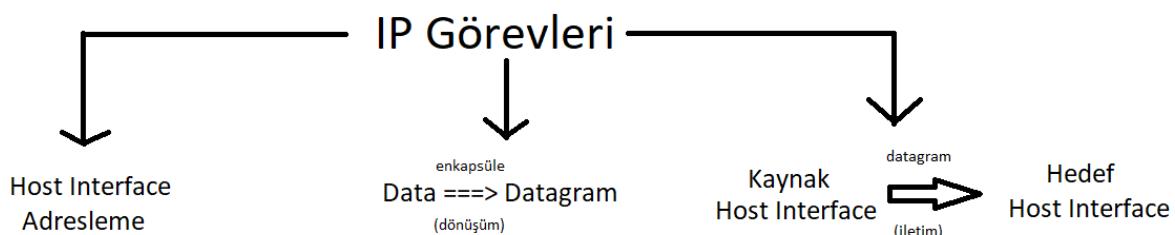
Bütün bunlar doğrultusunda Internet gibi geniş çaplı ağlarda diğerlerine göre daha yeni olan packet switching (datagram) metodu kullanılıyor.

IP (Internet Protocol) Nedir?

IP, "Internet Protocol" (İnternet Protokolü) teriminin kısaltmasıdır. IP veriyi enkapsüle eden paket yapısıdır. Internet üzerindeki bilgisayarların ve cihazların birbirleriyle iletişim kurmasını sağlayan bir iletişim protokolüdür. Bu işlem IP adresler ile gerçekleşir. IP adresleri, internet üzerindeki her bir cihazın benzersiz kimlik numarasıdır ve verilerin doğru hedefe yönlendirilmesine yardımcı olur.

IP, farklı sürümlere sahip olabilir. IPv4 (Internet Protocol Version 4) ve IPv6 (Internet Protocol Version 6) yaygın kullanılan IP sürümleridir. IPv4, 32 bitlik adresleri kullanırken, IPv6 128 bitlik adresleri kullanır ve böylece daha fazla benzersiz IP adresi sağlar, çünkü internetin büyümeye IPv4 adresleri tükenmeye başlamıştır. IPv6 adresleme sistemi, internetin büyümeye ve cihazların sayısındaki artışa uyum sağlamak için tasarlanmıştır. Bu sayede, her cihazın kendine özgü bir global IP adresi olabilir, böylece adres çatışmaları azalır ve IP adresleri daha verimli kullanılır.

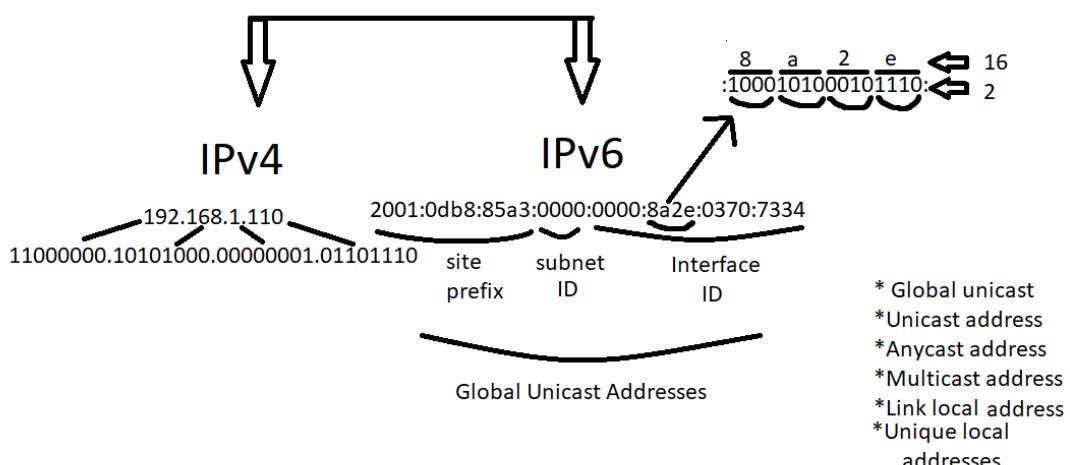
IP Nasıl çalışır?



Internet Protocol (IP); host interface'leri (bir bilgisayarın veya ağ cihazının (örneğin, bir router veya switch) dış dünya veya başka cihazlarla iletişim kurmasını sağlayan fiziksel veya mantıksal bağlantı noktalarını ifade eder) adreslemeden, veriyi (data) veri birimine (datagram, "Datagram", veri iletimi sırasında paketlenen ve ağ üzerinden gönderilen bir veri birimidir.) enkapsüle etmeden ve bir veya birden fazla IP ağları üzerinden kaynak host interface'den hedef host interface'e veri birimlerini (datagram) iletmekten sorumludur. Aslında hosttan hosta iletim söz konusu. Host ağa bağlı cihazı temsil ederken host interface o hostun mantıksal veya fiziksel bağlantı noktasını yani girişini/çıkışını temsil eder.

1- Host Interface Adresleme

1) HOST INTERFACE ADRESLEME



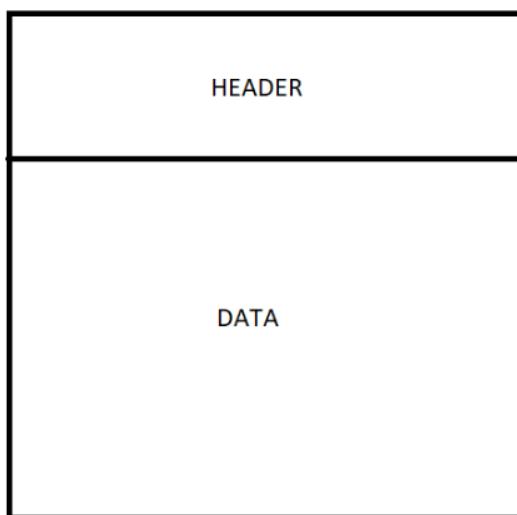
Host interfaceler, IPv4 veya IPv6 adresleme kullanılarak adreslenir. IPv4 adresleri 32 bit kullanır ve 4 parçaya ayrılır. Her bir parça 8 bit'i yani 1byte'ı temsil eder. Her bir byte, 8 bit değerinin binary şekilde yazılmıştır. Mesela 11000000, 192'yi temsil eder. IPv4 adreslerinin ihtiyaçları karşılayamama gibi bir problemleri düşünülmeye başlayınca IPv6 adresleri ortaya çıkmıştır.

IPv6 adresleri ise 128 bit kullanır ve 8 parçaya ayrılır. Her bir parça 16 bit'i yani 2 byte'ı temsil eder. 16'luk tabanda yazılarak gösterilir (IPv4 2'lük-binary). 8a2e parçasına bakalım. Bu kısım 16 bit'i yani 2 byte'ı oluşturur. İster tamamı için (8a2e), isterse de ayrı ayrı 4 bit için (8,a,2,e) 16'luk taban 2'lük taban şeklinde yazılır. Mesela 8 için 1000, a (10) için 1010, 2 için 0010 ve e (14) için 1110 şeklinde ifade ederiz ve bunları birleştiririz. Ortaya 1000101000101110 çıkar. Veya tamamı yani 8a2e'yi doğrudan binary şeklinde yazarız. Yine 1000101000101110 sonucu ortaya çıkar. Bu şekilde bir parçanın 16 bitten oluşanluğu görülür. Her 8 parça için bu işlemi yaptığımızda 16 bit'ten IPv6'nın 128 bitten oluşanluğu sonucunu elde ederiz.

IPv6, **Global Unicast Address** çatısı altında 3 parçaya ayrılır. İlk 48 bit site prefix, sonraki 16 bit subnet ID ve son 64 bit Interface ID'den oluşur. Tabii ki bu her IPv6 adresleri için geçerli değildir. Bizim global

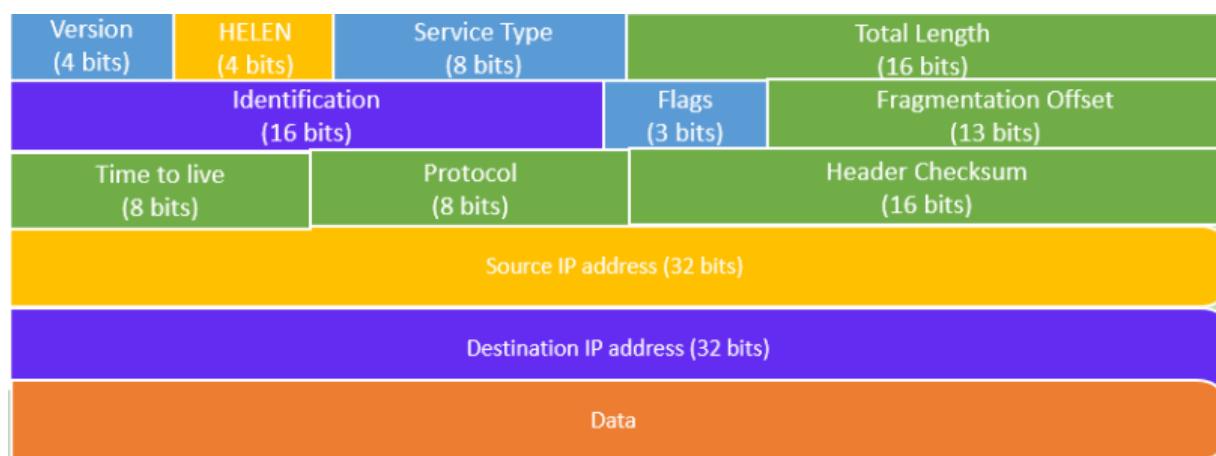
unicast, unicast, anycast, multicast, link local ve unique local olacak şekilde birden fazla adresimiz var. Her biri için farklı gösterim biçimleri vardır.

2- IP Datagram Elde Etme



Öncelikle veriyi yani datayı kısaca özetleyelim. IP verisi, Internet Protokolü (IP) üzerinde taşınan verileri ifade eder. Hatırlarsanız datayı datagrama enkapsülle işlemi ile çeviriyorduk. Ona sonra geleceğiz. Önce datagram yapısına bakalım.

Datagram ise farklıdır. **"Datagram"**, veri iletimi sırasında paketlenen ve ağ üzerinden gönderilen bir veri birimidir. Verinin iletilmesi için genellikle verinin datagram haline dönüştürülmesi gereklidir. Yani data'yı şekillendirip datagramı elde ediyoruz. Her bir datagram iki bileşen içerir. Header ve payload (data). IP header'ı kaynak IP adresi, hedef IP adresi ve veri biriminin (datagram) taşınmasına yardımcı olan diğer metadataları içerir. Payload ise taşınan veridir (data). Header ile paket içine veri payload'ını yerleştirme işlemeye enkapsülle (encapsulation) adı verilmektedir. Datagram yapısında, genellikle data (payload) bölümü, başlıktan daha çok yer kaplar. Başlık (header) kısmı, veriyi taşıyan pakete yönlendirme ve diğer kontrol bilgilerini ekleyen kısmı olduğu için genellikle daha küçüktür.



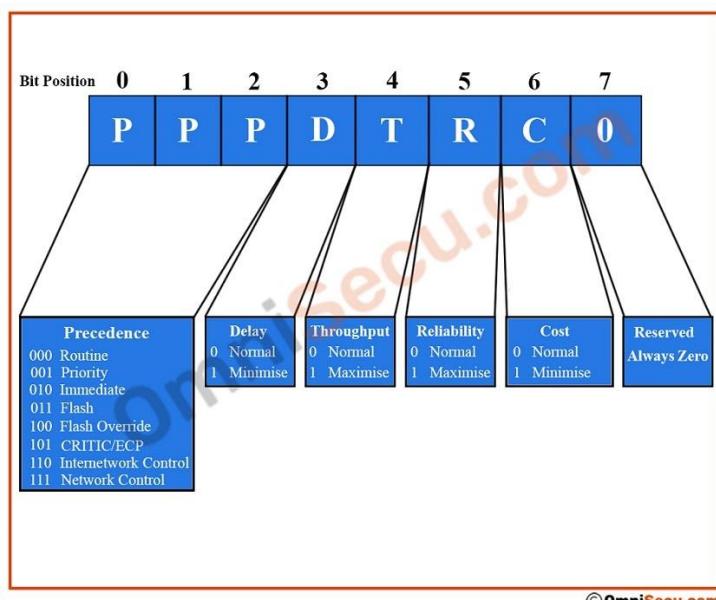
IP datagramının Header kısmını oluşturan parçalar sırasıyla Version, Header Length, Service Type, Total Length, Flags, Fragmentation Offset, TTL (Time to Live), Protocol, Header Checksum, Source IP Address, Destination IP Address. IP datagramının Payload kısmını ise Data (veri) oluşturur.

IPv4 Header

Version (Versiyon - 4bit): IP protokolünün sürüm numarası (örneğin, IPv4 veya IPv6). Eğer IPv4 protokolü kullanıyorsak version alanı 0100'dan oluşur. IPv6 kullanıyorsak 0110'dan oluşur.

Header Length (Başlık Uzunluğu – 4 bit): Header'ın ne kadar uzunlukta olduğunu belirtir. Header uzunluğu 20 byte olduğunu düşünelim (20 ila 60 bayt arasında değişim mümkündür). Verimiz ise 500 byte olsun. Bu durumda verimiz aslında 480 byte'dan oluşacak. Çünkü header kısmının kapladığı bir alan var.

Service Type (Servis Tipi – 8 bit): Paketin hizmet kalitesi (QoS – quality of service) ve önceliğiyle ilgili bilgileri içerir. Bazı IP datagramları diğerlerinden daha önemlidir. Dolayısıyla öncelik buradaki bilgiler doğrultusunda sağlanır. 8 bitten oluşur. Bu bitler şu şekildedir:



Precedence (Öncelik): 3 bitten oluşur ve bit değerine göre öncelik belirlenir.

Delay (Gecikme): Gecikmenin belirlendiği bit değeridir. 1 bitten oluşur. 0 normali 1 minimumluğu ifade eder.

Throughput (Verim): Verimin belirlendiği bit değeridir. 1 bitten oluşur. 0 normali 1 maksimumluğu ifade eder.

Reliability (Güvenirlilik): 1 bitten oluşur ve güvenirliliği belirler. 0 normali 1 maksimumu ifade eder.

Cost (Maliyet): 1 bitten oluşur ve maliyeti belirler. 0 normali 1 minimumu ifade eder.

0: Son 1 bit her zaman için 0'dan oluşur.

Total Length (Total Uzunluk – 16 bit): Header (Başlık) ve veri (Data - Payload) bölümlerinin toplam uzunluğunu belirtir. Yani datagramın uzunluğunu belirtir.

Identification (kimlik – 16 bit): Paketin parçalarına bölündüğü durumlarda, orijinal paketin parçalarını birleştirmek için kullanılır. Eğer verinin büyüklüğü bir datagramın içine sığamayacak seviyedeyse, IP katmanında veri paketlere bölünür.

Flag (Bayrak – 3 bit): Paketin parçalara bölünmesi durumunda yönetim bilgilerini içerir. Datagramın bölünebileceğini veya bölündüğünü gösterir.

Fragment Offset (Parça Numarası – 13 bit): Paketin parçalanması durumunda parçanın (fragment) başlangıç ofsetini belirtir. Parçalanma işlemi ise bir IP datagramını birden çok küçük datagramlara bölmeye işlemidir. Fragment Offset sayesinde bölünen parçalar doğru sırada toparlanır.

TTL (Time to Live – 8 bit): Paketin ağ üzerinde kaç yönlendirici tarafından geçebileceğini belirler ve paketin sonsuz döngülere düşmesini önerler. Bir router'dan datagram diğer router'a ulaştığında, router TTL boşluğunundaki değeri 1 azaltır. Bu değer 0'a ulaştığı zaman router datagramın daha ileri gidemeyeceğini bilir. Böylece sonsuz döngünün önüne geçilir.

Protocol (Üst Katman Protokolü – 8 bit): Taşınan verinin üst katman protokolünü belirtir (örneğin, TCP, UDP, ICMP, vb.).

Header Checksum (Başlık Kontrol Toplamları – 16 bit): Başlık bölümündeki hataları tespit etmek için kullanılan kontrol toplamıdır. Tüm datagram headerinin içeriğidir. TTL boşluğunundaki değer her değiştiğinde checksum boşluğu da değişir.

Source IP Address (Kaynak IP Adres – 32 bit): Kaynak IP adresini tutar. IPv4 için 32 bit IPv6 için 128 bit değeridir.

Destination IP address (Hedef IP Adres – 32 bit): Hedef IP adresini tutar. IPv4 için 32 bit IPv6 için 128 bit değeridir.

Ek Kısımlar

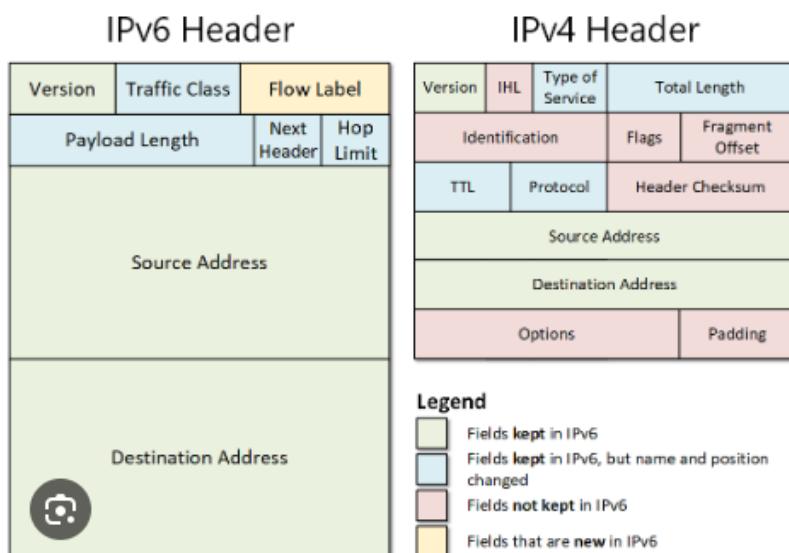
Options (Boyutu Değişebilir): Opsiyonel boşluktur. Test amaçlı öncelikli datagramlar için karakteristik özelliği berlirer.

Padding (Boyutu Değişebilir): Sadece 0'lardan oluşur. Options boşluğu opsiyonel ve değişken olduğu için header kısmının doğru boyutta olabilmesi için kalan kısımları 0 ile doldurur.

IPv4 Payload

IPv4 datagramında taşınan asıl veri yüküdür. Bu kısım, başlık kısmında belirtilen üst katman protokolüne ait verileri içerir. Örneğin, TCP verisi taşınacaksa, TCP segmenti burada yer alır.

Bu saydıklarımız IPv4 için geçerli. IPv6 için daha farklı bir header yapısı vardır.



Fark edildiyse IPv4 kaynak ve hedef adresleri 32 bit'ten oluşuyor ve sadece 32 bitlik 1 alan kaplıyor. Hatırlarsak IPv6 128 bit'ten oluşuyordu ve mantık olarak yine 32 bitlik 4 alan kaplıyor.

IPv6 datagramının header (başlık) kısmı 40 bayt uzunluğundadır ve paketin yönlendirilmesi ve hedefe ulaşması için gerekli bilgileri içerir.

IPv6 Header

Sürüm (Version): IP protokolünün sürüm numarası; IPv6 için değeri 6'dır.

Hizmet Sınıfı (Traffic Class): Paketin hizmet kalitesi ve önceliğiyle ilgili bilgileri içerir.

Etiket (Flow Label): Akış etiketidir, özellikle akışı gerektiren uygulamalarda kullanılır.

Payload Uzunluğu (Payload Length): Başlık ve veri bölümlerinin toplam uzunlığını belirtir.

Üst Katman Protokolü (Next Header): Taşınan verinin üst katman protokolünü belirtir (örneğin, TCP, UDP, ICMPv6, vb.).

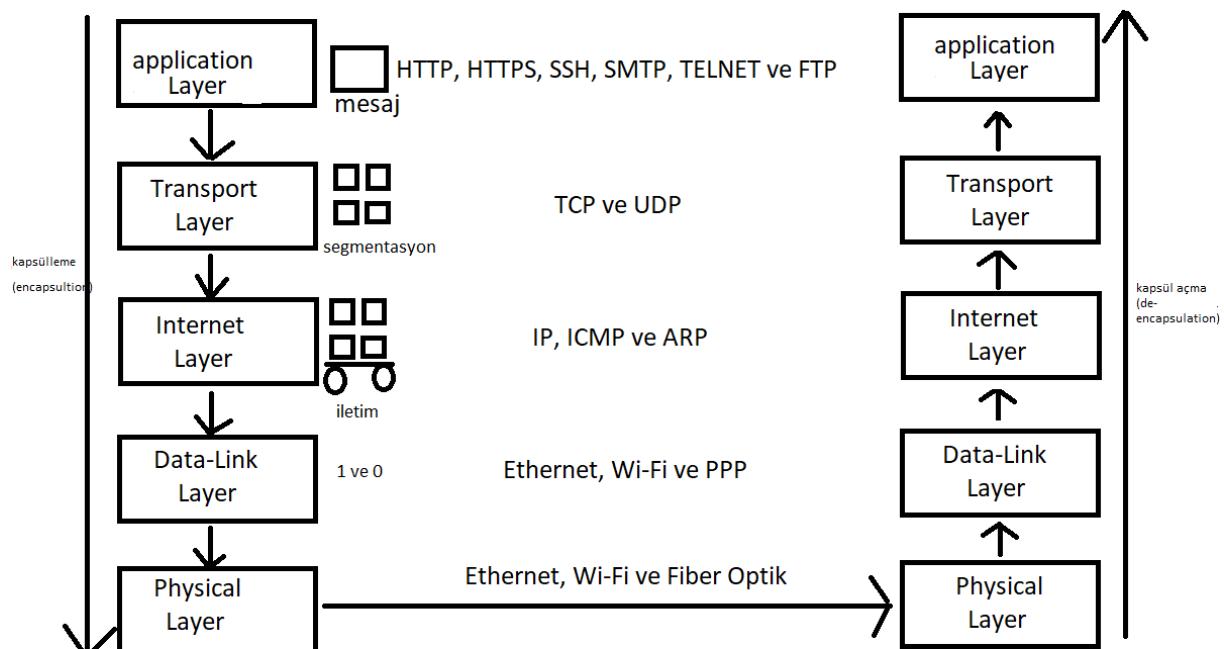
Geçerlilik Süresi (Hop Limit): Paketin ağ üzerinde kaç yönlendirici tarafından geçebileceğini belirler (IPv4'deki TTL'ye karşılık gelir).

Source IP Address (Kaynak IP Adres – 32 bit): Kaynak IP adresini tutar. IPv4 için 32 bit IPv6 için 128 bit değeridir.

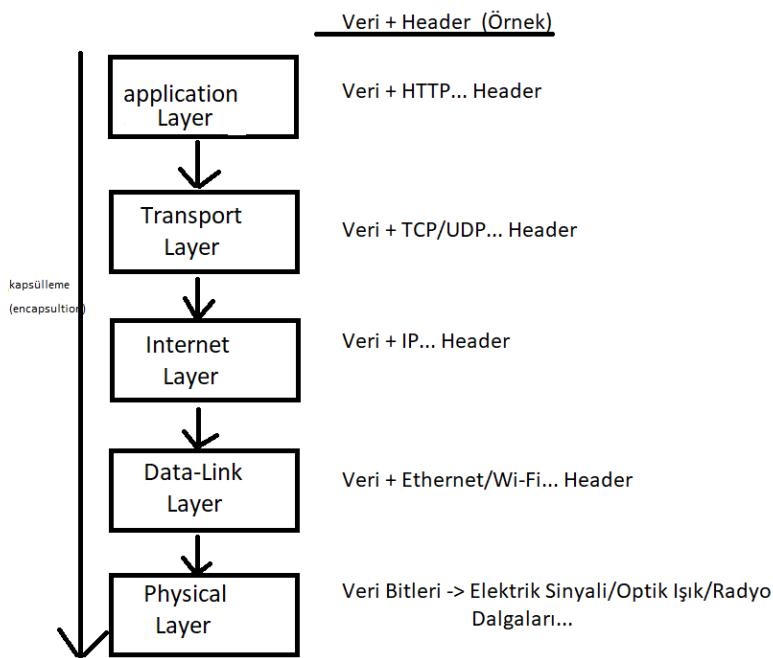
Destination IP address (Hedef IP Adres – 32 bit): Hedef IP adresini tutar. IPv4 için 32 bit IPv6 için 128 bit değeridir.

IPv6 Payload

IPv6 datagramında taşınan asıl veri yüküdür. Bu kısım, başlık kısmında belirtilen üst katman protokolüne ait verileri içerir. Örneğin, TCP verisi taşınacaksız, TCP segmenti burada yer alır.



Datagram ile enkapsülasyon (kapsülleme) işlemi ilişkili demistik. Datagram enkapsülasyonu, ağ iletişimi sırasında verinin katman katman paketlere yerleştirilmesi işlemine verilen isimdir. Bu süreç, veri iletimi için kullanılan protokol suitlerinin her bir katmanında gerçekleşir ve iletilen verinin farklı katmanlardaki başlık ve veri yapılarına yerleştirilmesini içerir.

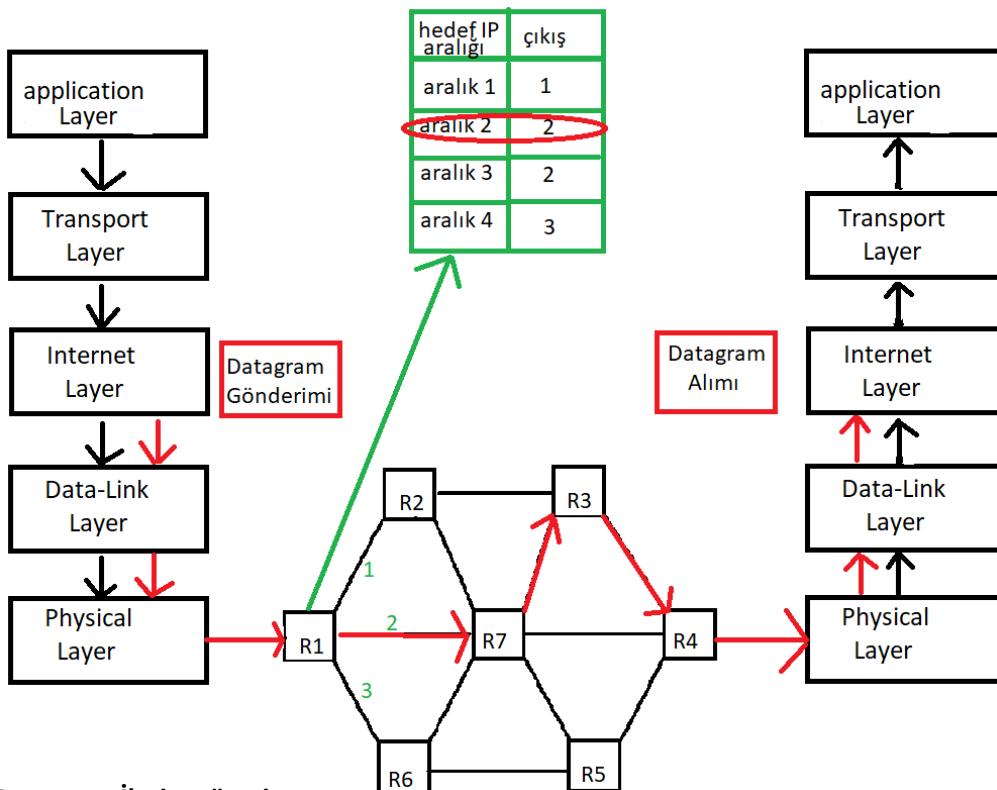


- Uygulama Katmanı: Uygulama katmanında veri ve başlık bir araya getirilir. Örneğin, HTTP uygulama katmanı protokolünde veri (web sayfası) ve HTTP başlığı bir araya getirilir.
- Taşıma Katmanı: Uygulama katmanından gelen veri, taşıma katmanında uygun protokolün (örneğin, TCP veya UDP) başlığı eklenerek ağ katmanına verilir.
- Ağ Katmanı (IP Katmanı): Taşıma katmanından gelen veri, IP katmanında IP başlığı eklenerek fiziksel katmana verilir. IP başlığı, kaynak ve hedef IP adreslerini içerir ve paketin yönlendirilmesi için gerekli bilgileri taşır. Bu aşamada datagram oluşturulmuş olur.
- Datalink Katmanı: Ağ katmanından gelen veri, datalink katmanında uygun protokolün (örneğin, Ethernet, Wi-Fi, vs.) başlığı ve sonlanma karakterleri eklenerek fiziksel katmana verilir.
- Fiziksel Katman: Datalink katmanından gelen veri, fiziksel katmanda uygun fiziksel sinyallere dönüştürülerek iletim için hazır hale getirilir. Bu katman, veri bitlerini fiziksel ortama uygun elektrik sinyalleri, optik ışık veya radyo dalgaları gibi biçimlere dönüştürür.

Her katman, üzerine yerleştirildiği katmanın verisini taşıyan bir başlık ekler. Başlık, verinin uygun bir şekilde yönlendirilmesi, doğrulanması ve hedefe ulaşması için gerekli bilgileri içerir. Bu başlıklar, verinin katmanlar arasında iletilesini ve uçtan uca iletişimini sağlamak için kullanılır.

Kapsülleme sırasında her katman, bir önceki katmandan aldığı veriyi kendi başlığıyla kapsüller. Her katman sadece kendi başlığını eklediğinden, aynı datagram üzerinde birden fazla katman başlığı (header) bulunmaz. Alıcı tarafta kapsül açma işlemi yapılır ve başlıklar her katmanda sırasıyla çıkarılır, böylece orijinal veri en üst katmana ulaşır ve kullanıcıya anlamlı hale gelir.

3- host interface'ler arasında datagram nasıl iletilir?



Datagram İletim süreci:

- ➔ **Veri Paketinin Oluşturulması:** Veri paketleri, gönderici cihazda (kaynak cihaz) oluşturulur. Veri paketleri, datagramlar olarak adlandırılır ve paketin içeriği verilerin yanı sıra hedef IP adresi, kaynak IP adresi ve diğer protokol bilgilerini içerir. Hatırlarsak data enkapsüle edilerek datagram oluşturuyordu.
- ➔ **Hedef IP Adresinin Belirlenmesi:** Gönderici cihaz, veri paketini göndermek istediği hedef cihazın IP adresini belirler. Bu işlem, veri paketinin yönlendirilmesi gereken yolu belirlemek için önemlidir. Datagram yeşil ile götserilen yönlendirme tablosundaki bilgiler doğrultusunda ilerler. Yani IP aralığı hangi çıkışa denk geliyorsa router'ın o çıkış yolu kullanılır. IP aralığı vermemizin sebebi Internet üzerinde yaklaşık 2^{32} 'den 4 milyar IP adres bulunmasından kaynaklıdır (IPv4 için konuşuyoruz). Çünkü her bir IP'yi tutmak imkansızı yakındır. Dolayısıyla router'ın yönlendireceği yol IP adres aralığına göre belirlenir.
- ➔ **Yönlendirme Kararı:** Gönderici cihaz, hedef cihazın IP adresini kullanarak bir yönlendirme tablosu aracılığıyla en uygun yolu belirlenmesini sağlar. Yönlendirme tablosu, çeşitli ağlar ve alt ağlar arasındaki bağlantılar ve en uygun router (yönlendirici) veya gateway (geçit) adreslerini içerir.
- ➔ **Datagramın İlerlemesi:** Gönderici cihaz, veri paketini yönlendirme kararı doğrultusunda uygun router'a (yönlendiriciye) veya gateway'e (ağ geçidine) iletir. Router (Yönlendirici), veri paketinin bir sonraki aşamada hangi çıkış arayüzüne yönlendirileceğini belirler.
- ➔ **Yönlendirme ve Aktarım:** Datagram, her router (yönlendirici) veya gateway (ağ geçidi) üzerinde uygun çıkış arayüzüne yönlendirilir ve sonunda hedef cihaza doğru ilerler. Bu işlem, veri paketinin tüm yönlendirme yollarında ve ağ geçitlerinde ilerlemesiyle tekrarlanabilir.
- ➔ **Varış:** Datagram, sonunda hedef cihaza ulaşır. Hedef cihaz, datagramı açar, içeriği verileri alır ve gerekli işlemleri gerçekleştirir.

DNS

domain name	IP Adres
www.google.com	↔ 8.8.8.8
www.facebook.com	↔ 208.65.153.238

Web üzerinde çok fazla web sayfası bulunmaktadır. Dolayısıyla insanlar bu IP adresleri ezberlemekte zorluk çekecektir. Çünkü bir web siteye gitmek istediğimizde IP adres ile gideriz. Bilgisayarlar binary (2'lik) dilinden anırlar. Bizim gibi değillerdir. Bu yüzden DNS'e (Domain Name System'e) ihtiyacımız var. DNS, Internet'te alan adlarını IP adreslerine çeviren bir sistemdir. Bu işlem tek yönlü değildir. DNS hem alan adlarını IP adreslerine çevirir (bu işleme alan adı çözümlemesi denir), hem de IP adresleri alan adlarına çevirir (bu işleme tersine çözümleme denir). DNS, bu çift yönlü dönüşümü sağlayarak internet trafiğinin doğru kaynaklara ulaşmasını mümkün kılar.

Bir web sitesinin (domain) birden fazla IP adresine sahip olması mümkündür. Bir web sitesi, birden fazla sunucu veya sunucu grubu tarafından barındırılabilir ve bu sunucuların her biri farklı IP adreslerine sahip olabilir.

Bir web sitenin birden fazla IP adresi kullanması durumu, yüksek trafikli veya dağıtık bir web sitesi için yaygın bir uygulamadır. Bunun nedeni **Yük Dengeleme** (Yüksek trafikli bir web sitesi, kullanıcıların taleplerini karşılamak ve hızlı yanıtlar vermek için birden fazla sunucu kullanabilir. Bu sunucuların her biri farklı IP adreslerine sahip olabilir ve trafiği bu sunucular arasında dengelendirmek için yük dengeleme yöntemleri kullanılır.), **Yedekleme ve Yedek Sunucular** (Web sitesi sahipleri, yüksek kullanılabilirlik ve sürekli çevrimiçi olma ihtiyacını karşılamak için yedek sunucular kullanabilirler. Ana sunucuya aynı içeriği barındıran bu yedek sunucular, farklı IP adreslerine sahip olabilir ve ana sunucunun çalışmadığı durumlarda trafiği yönlendirebilirler.) ve **Coğrafi Dağılım** (Büyük bir web sitesi, farklı coğrafi bölgelerdeki kullanıcırlara daha iyi hizmet vermek için farklı veri merkezlerinde barındırılabilir. Her veri merkezi, farklı bir IP adresine sahip olabilir ve DNS, kullanıcıları en yakın veri merkezine yönlendirebilir.) olabilir.



DNS Nasıl Çalışır?

Browser Nedir?

Tarayıcı (browser), bilgisayar, akıllı telefon, tablet ve diğer cihazlar aracılığıyla internet üzerindeki web sayfalarını görüntülemek ve etkileşimde bulunmak için kullanılan bir yazılımdır. Tarayıcılar, web içeriğini alırlar, HTML, CSS ve JavaScript gibi web teknolojilerini yorumlar ve kullanıcılarla görsel olarak sunarlar.

Web Server Nedir?

Web sunucusu (web server), istemcilere (genellikle web tarayıcılarına) web sayfalarını sağlayan, HTTP (Hypertext Transfer Protocol) üzerinden iletişim kurarak istemci taleplerini işleyen bir yazılım veya donanım sistemidir.

DNS Nedir?

DNS (Domain Name System), internet üzerindeki alan adlarını (örneğin, www.example.com) IP adreslerine dönüştüren bir sistemdir. İnsanların anlamlandıracakları alan adlarını, bilgisayarların anlayabileceği IP adreslerine çevirerek iletişimini kolaylaştırır.

DNS Server Nedir?

DNS sunucusu (DNS server) ise DNS hizmetini sağlayan bir sunucu veya hizmettir. DNS sunucusu, alan adlarının IP adreslerine dönüştürülmesinden sorumludur. İstemciler (örneğin, web tarayıcıları), DNS sunucusuna bir DNS sorgusu göndererek bir alan adının IP adresini öğrenirler. Yani, DNS bir sistem veya protokolken, DNS sunucusu bu sistemi uygulayan ve alan adlarının IP adreslerine çözülmüşinden sorumlu olan bir sunucudur. DNS sunucusu, DNS sorgularını işler, kaynak sunuculara yönlendirme yapar ve istemcilere alan adlarının karşılık geldiği IP adreslerini sağlar.

Cache nedir?

Cache, verilerin geçici olarak saklandığı bir bellek alanıdır. İnternet ve bilgisayar sistemlerinde cache, sık kullanılan verilerin daha hızlı erişilebilir olmasını sağlamak için kullanılır.

DNS Root Name Server Nedir?

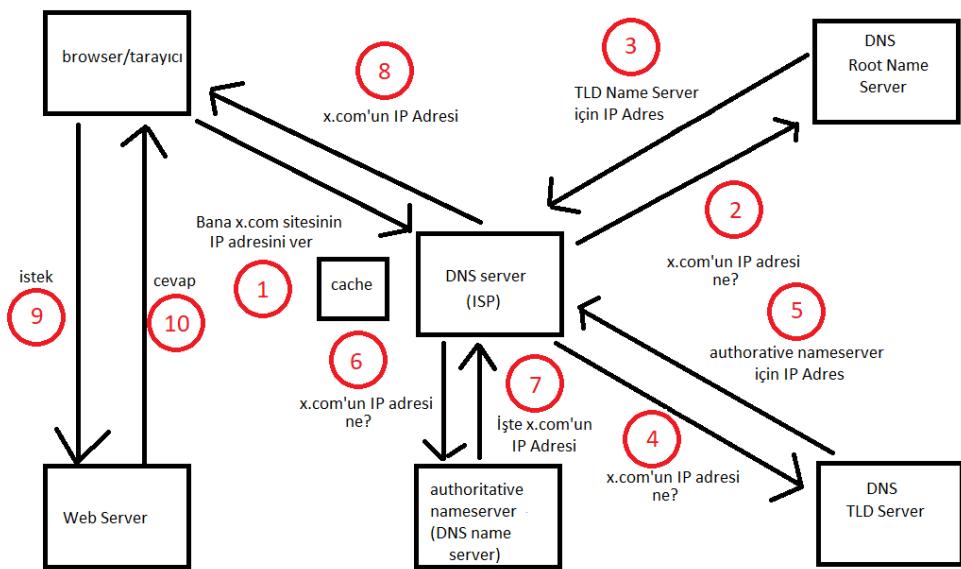
Kök ad sunucusu (root name server), İnternet'teki DNS (Domain Name System) hiyerarşisinin en üst düzeyinde yer alan sunuculardır. Kök ad sunucuları, DNS sisteminin temel yapı taşıdır. İnternet üzerindeki herhangi bir alan adına ilişkin bilgi, kök ad sunucuları tarafından sağlanmaz. Bunun yerine, kök ad sunucuları, tüm DNS sorgularının yönlendirildiği ilk noktalardır.

DNS TLD (Top Level Domain) Server Nedir?

Üst düzey alan adı sunucusu (top-level domain name server), İnternet'teki DNS (Domain Name System) hiyerarşisinin üst düzey alan adlarına ilişkin bilgileri sağlayan sunuculardır. Her üst düzey alan adı (TLD), İnternet adreslerinin sonundaki en üst düzey etki alanlarını temsil eder. Örneğin, .com, .net, .org, .gov, .edu gibi popüler üst düzey alan adları vardır.

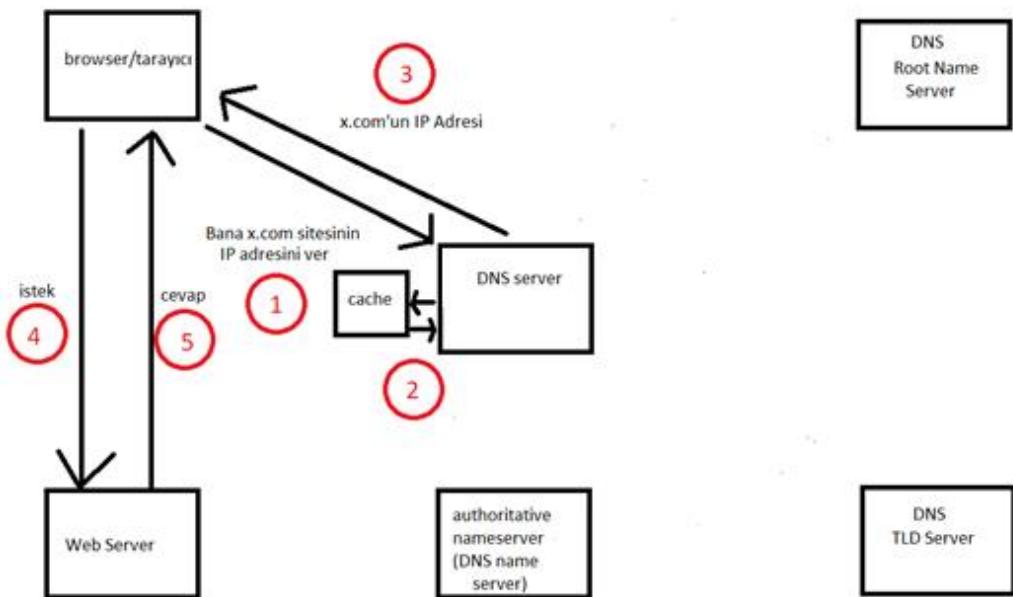
DNS Authoritative Server Nedir?

Authoritative name server (yetkili ad sunucusu), bir alan adının DNS kayıtlarını yöneten ve bu kayıtlara yönelik sorguları yanıtlayan sunucudur. Yetkili ad sunucusu, o alan adının DNS kayıtlarının kaynağıdır ve doğrudan bu kayıtlara erişebilir.



DNS Çalışması

- 1- Tarayıcıya x.com yazdığımızda biz arka planda DNS server'a yönlendiriliyoruz. Amaç DNS server'dan x.com'un IP adresini almak. IP adresi doğrudan alamıyoruz. DNS server kendi içinde parçalara ayrılıyor. Bunlar: DNS Root Name Server, DNS Top Level Domain Server, DNS Authoritative Name Server
- 2- 3- Bu adımda x.com'un IP adresini DNS Root Server'da arıyoruz. DNS Root Server, Domain Name içeren soruyu kabul eder. Daha sonra cevap döndürür ve bu cevap domain'in uzantısına göre Top Level Domain Server'a gidecek şekilde şekillendirilir (.com, .net, .org, ...).
- 4- 5- Bu adımda DNS Root Server'dan gelen bilgileri, DNS TLD Server'a iletiyoruz. TLD Server, Internet'teki DNS (Domain Name System) hiyerarşisinin üst düzey alan adlarına ilişkin bilgileri sağlayan sunuculardır. Her üst düzey alan adı (TLD), Internet adreslerinin sonundaki en üst düzey etki alanlarını temsil eder (.com, .net, .org, ...). Daha sonra bize DNS TLD Server bize authoritative name server için bilgiler döndürüyor.
- 6-7- Bu adımda Authoritative Name Server'a bir önceki adımdan gelen bilgileri iletiyoruz. Authoritative Name Server bir alan adının DNS kayıtlarını yöneten ve bu kayıtlara yönelik soruları yanıtlayan sunucudur. Ayrıca hizmet verdiği alan adına (x.com) özgü bilgiler içerir. En sonunda IP Adresi DNS server'a yönlendirir ve IP Adres bu adımdan sonra DNS Server'da çözümlenmiş olur.
- 8- Bu adımda DNS Server'da çözümlenen IP Adres tarayıcıya döner.
- 9- Bu adımda DNS Server'dan gelen IP adresi kullanılarak tarayıcıdan web sunucusuna istek gönderilir.
- 10- Tarayıcıdan gelen istek doğrultusunda sayfa cevap olarak tarayıcıya geri döner. Cevap içerisinde linkler ve resimler gibi aynı sunucuda olan dosyalar bulunabilir.
- 11- Tabii ki bu aşamalar, IP Adresin cache (önbellek) içinde olmadığı zamanlarda geçerli. Eğer IP adres, tarayıcının ya da işletim sisteminin önbelleginde (cache) bulunuyorsa doğrudan IP Adres alınır ve tarayıcıdan web sunucusuna istek gönderilir. Yani IP Adres cache içinde bulunuyorsa DNS Server'ın 3 aşama ile iletişime geçmesine gerek yoktur. DNS server, bize doğrudan IP Adresi döndürür. Bu IP Adres ya browser'da bulunan cache ya da DNS Server'da bulunan cache üzerinden döner. Ancak cache geçici bir depolama alanıdır. Ayrıca, önbellette bulunan bir IP adresi her zaman doğru ve güncel olmayıpabilir. Dolayısıyla DNS Server içindeki 3 aşama tekrar gerçekleştirilerek IP Adres yeniden alınabilir.



DNS kaydı nedir?

DNS kayıtları, bir alan adını web tabanlı bir hizmetle ilişkilendirir. DNS kaydının birincil amacı, web sitenizi internette aranabilir ve erişilebilir kılmaktır. Bu, bir alan adını sunucunuzun IP adresine yönlendiren bir dizi işlem aracılığıyla gerçekleşir. Bizim birçok DNS kayıt tiplerimiz vardır. Bunlardan bazıları şunlardır:

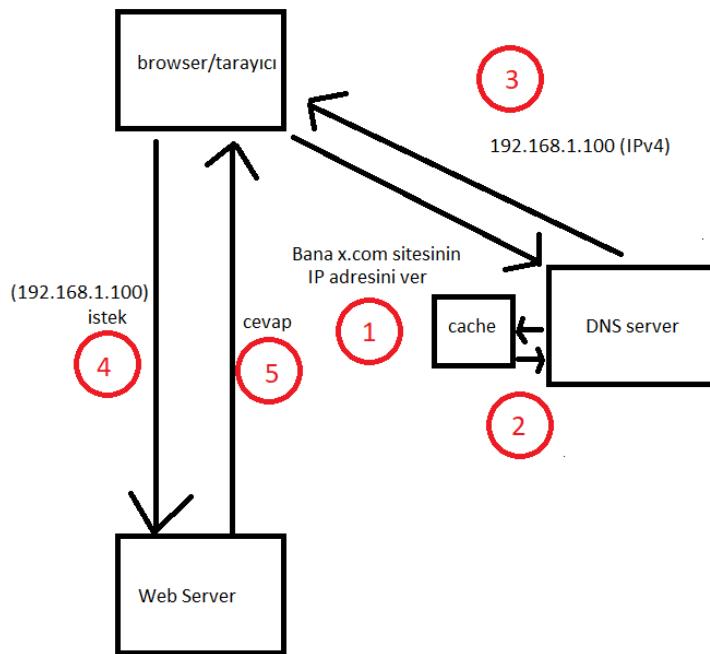
A	AAAA	CNAME	MX	SOA	NS	TXT	PTR	SRV
---	------	-------	----	-----	----	-----	-----	-----

A Tipi DNS Kaydı

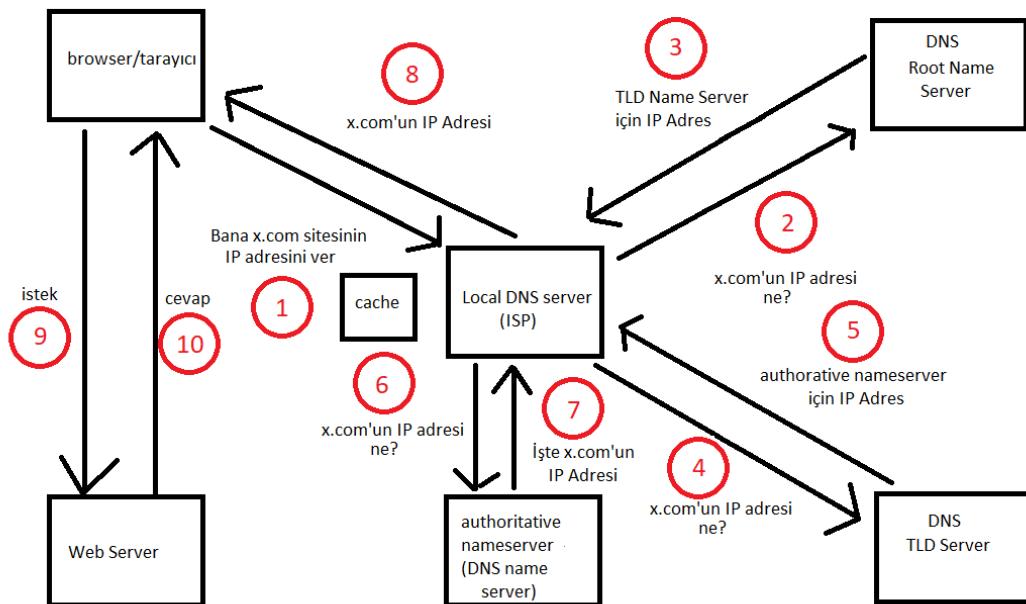
Type	Name	IP addr	TTL
A	x.com	192.168.1.100	3600

A tipi DNS kaydı (A record), bir alan adını (domain name) IPv4 adresiyle ilişkilendiren DNS kaydıdır. "A" harfi, "Address" kelimesinin baş harfidir ve bu kayıtlar alan adlarını IPv4 adreslerine dönüştürmek için kullanılır. Name adresin domain adını, IP ise adresin IP Adresini gösterir.

TTL (Time to Live) değeri, bu kaydın diğer DNS sunucularında ne kadar süreyle önbellekte saklanacağını belirleyen bir parametredir. TTL değeri, saniye cinsinden belirtilir ve her DNS kaydı için ayrı ayrı tanımlanabilir.



A Kaydı ve DNS Çalışma Mantığı



Örneğin siz x.com diye Internet üzerinde arattığınız zaman cihazınızdan çıkan veriler DNS Sunucuya x.com diye gider. Daha sonra bu sunucuda gerekli çevirme işlemi yapıp girdiğiniz domain name IPv4 adresi neyse ona çevrilir ve yönlendirme işlemi gerçekleşir. Bu işlemde 192.168.1.100 adresli web siteye istekte bulunuyoruz. Böylece biz x.com'a gitmek istediğimizde arka planda 192.168.1.100 adresine gitmiş oluyoruz. Sonrasında istek doğrultusunda cevap dönüyor ve web siteyi görüyoruz.

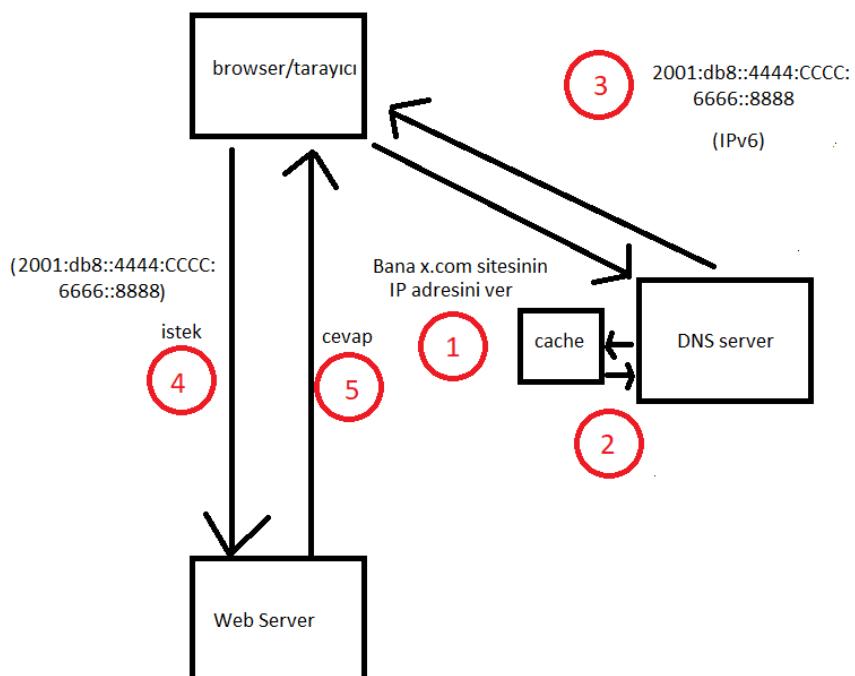
AAAA Tipi DNS Kaydı

Type	Name	IP addr (IPv6)	TTL
A	x.com	2001:db8::4444:CCCC: 6666::8888	3600

AAAA tipi DNS kaydı, bir alan adının (domain name) IPv6 adresiyle ilişkilendirildiği DNS kaydıdır. "AAAA" harfleri, "Address (IPv6)" kelimesinin baş harfleridir. Bu kayıtlar, IPv6 adreslerini alan adlarına dönüştürmek için kullanılır. A tipi DNS kaydı IPv4'ten oluşuyordu. Bu da hatırlarsak 32 bit'ti. 4 tane A'dan yani AAAA tipi DNS kaydı ise IPv6'dan oluşuyor ve 128 bit. Sağlamasını buradan da yapabiliriz. 1 a 32 biti temsil ederse 4 a 128 biti temsil eder.

AAA tipi DNS kayıtları, özellikle IPv6 adreslerinin kullanıldığı modern ağlar için önemlidir. Günümüzde, IPv6 adresleri, IPv4 adreslerinin tükenmesini önlemek ve daha fazla cihazı ve hizmeti desteklemek için giderek daha yaygın hale gelmektedir. Bu nedenle, AAAA tipi DNS kayıtları, IPv6 adreslemesi kullanan web siteleri ve hizmetler için önemli bir rol oynar.

Name adresin domain adını, IP ise adresin IP Adresini gösterir. TTL (Time to Live) değeri, bu kaydın diğer DNS sunucularında ne kadar süreyle önbellekte saklanacağını belirleyen bir parametredir. TTL değeri, saniye cinsinden belirtilir ve her DNS kaydı için ayrı ayrı tanımlanabilir.



Bu örnek yine IPv4 örneğine benzer. Yine x.com diye Internet üzerinde arattığımız zaman cihazımızdan çıkan veriler DNS Sunucuya x.com diye gider. Daha sonra bu sunucuda gerekli çevirme işlemi yapılmış girdiğiniz domain name IPv6 adresi neyse ona çevrilir ve yönlendirme işlemi gerçekleşir. Bu işlemde 2001:db8::4444:CCCC:6666::8888 adresli web siteye istekte bulunuyoruz. Böylece biz x.com'a gitmek istediğimizde arka planda 2001:db8::4444:CCCC:6666::8888 adresine gitmiş oluyoruz. Sonrasında istek doğrultusunda cevap dönüyor ve web siteyi görüyoruz.

CNAME Tipi DNS Kaydı

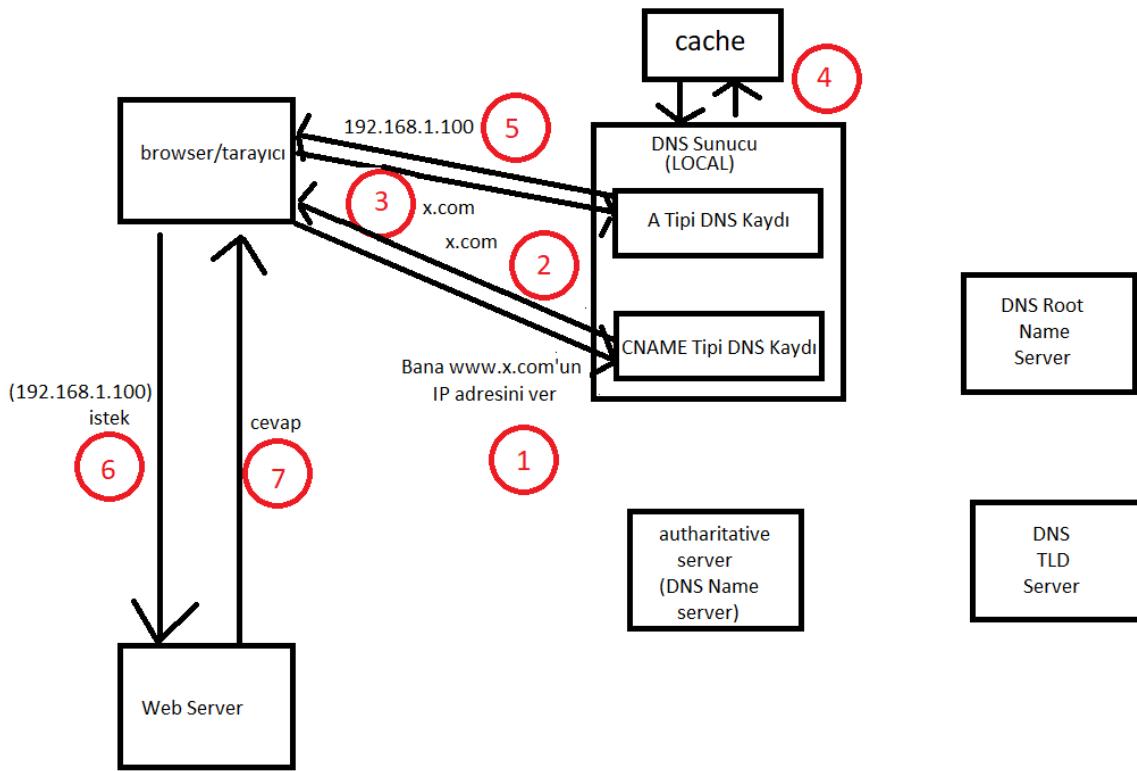
TYPE	NAME	NAME2	IPv4/IPv6	TTL
CNAME + A/AAAA	WWW.X.COM	X.COM	192.168.1.100	3600

CNAME (Canonical Name) tipi DNS kaydı, bir alan adının (domain name) başka bir alan adıyla ilişkilendirildiği DNS kaydıdır. CNAME kaydı, bir alan adının gerçek (kanonik) adını belirleyerek başka bir alan adına yönlendirilmesini sağlar.

CNAME kaydı, bir alan adının DNS çözümlemesinin, başka bir alan adına yönlendirilmesi gerektiği durumlarda kullanılır. Özellikle alt alan adları için yaygın olarak kullanılır. Alt alan adları, ana alan adının bir uzantısı veya bölümü olarak hizmet veren alt bölümlerdir.

Örneğin: x.com bir ana alan adıdır. mail.x.com ve blog.x.com, x.com'un alt alan adlarıdır. Mesela mail.x.com diye arattığımızda bizi direkt x.com'a yönlendirmesi CNAME kaydı sayesinde olmaktadır.

Name adresin subdomain adını, name2 adresin domain adını, IP ise adresin IP Adresini gösterir. TTL (Time to Live) değeri, bu kaydın diğer DNS sunucularında ne kadar süreyle önbellekte saklanacağını belirleyen bir parametredir. TTL değeri, saniye cinsinden belirtilir ve her DNS kaydı için ayrı ayrı tanımlanabilir.



Bu örnekte ise çok aşamalı DNS sunucu işlemi gerçekleşiyor `www.x.com` diye Internet üzerinde arattığımız zaman cihazımızdan çıkan veriler DNS Sunucusuna CNAME kaydına `www.x.com` diye gider. Daha sonra bu sunucuda gerekli çevirme işlemi yapılp girdiğimiz domain name sadece `x.com` olarak çevrilir. Daha sonra bu `x.com` verisi tarayıcıya döner. Tarayıcıya dönen bu domain (`x.com`) yeniden DNS sunucusuna gider (A tipi veya AAAA tipi DNS kaydına). Bu örnekte IPv4 adresi üzerinden gösterdik. Yani A tipi bir DNS kaydı kullanıldı. Eğer sitenin IP adresi IPv6 formatında olsaydı AAAA tipi bir DNS kaydı kullanacaktık. Bu adımdan sonra DNS sunucu bize A tipi DNS Kaydı sonucunda IPv4 formatında bir IP adres döndürdü (192.168.1.100). Bu IP adresi 4 numarada gösterildiği gibi cache (ön bellek) içinden aldı. Bunu tarayıcının ön belleğinden de alabilirdi. Veya cache içinde hiç bulunmasaydı root name server, tld server ve autharitative serverı dolaşıp bu IP adresi alacaktı. Daha sonra bu IP adresine sahip web sunucuya istekte bulunuyoruz. Sonrasında istek doğrultusunda cevap dönüyor ve web siteyi görüyoruz.

MX Tipi DNS Kaydı

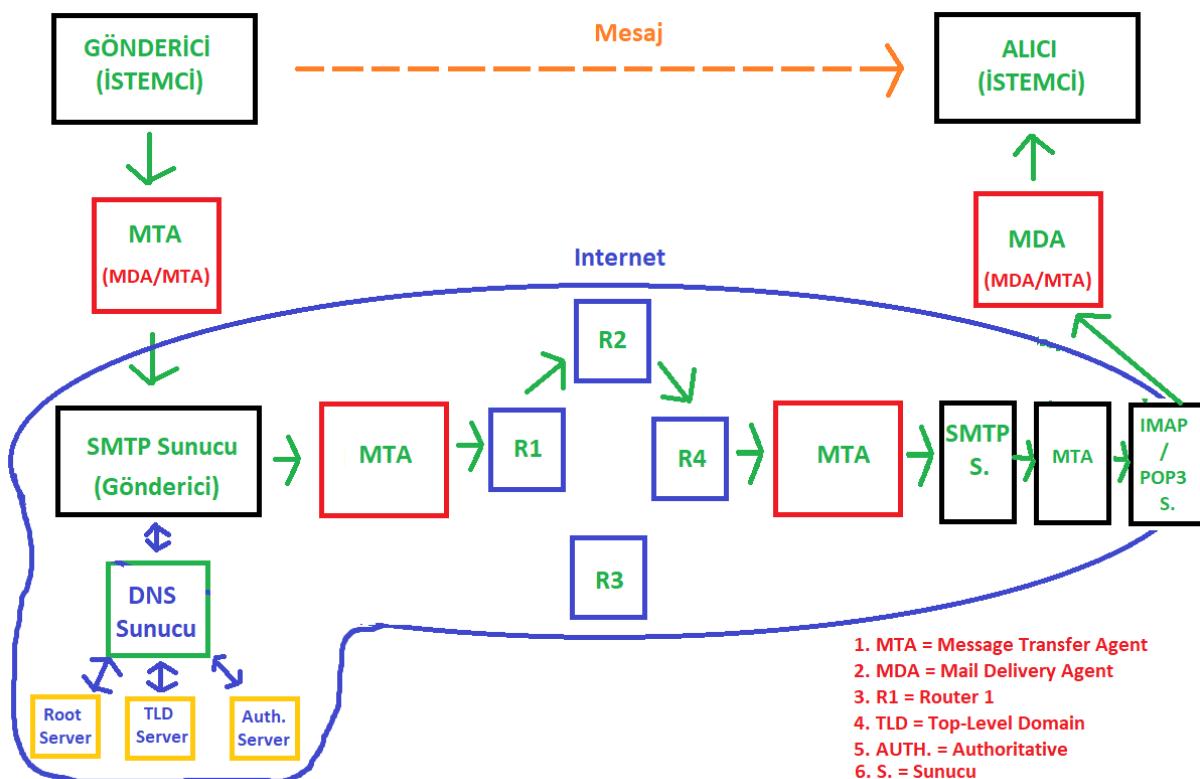
Type	Priority	Name	Value (Host)	TTL
MX	10	x.com	mail1.x.com	7200
MX	20	x.com	mail2.x.com	7200

MX (Mail Exchange) DNS kaydı, e-posta hizmetlerini yönlendirmek için kullanılan bir tür DNS kaydıdır. Bu kayıt, bir alan adına ait e-posta sunucularının IP adreslerini belirler. E-posta gönderimi yaparken, gönderilen e-postanın hedef alıcıya ulaşması için doğru e-posta sunucusuna yönlendirilmesini sağlar.

MX (Mail Exchange) DNS kaydı içindeki "priority" (öncelik) değeri, aynı etki alanına ait birden fazla MX kaydı olduğunda hangi e-posta sunucusunun öncelikli olarak kullanılması gerektiğini belirleyen bir sayısal değerdir. Düşük sayılar daha yüksek önceliği temsil eder.

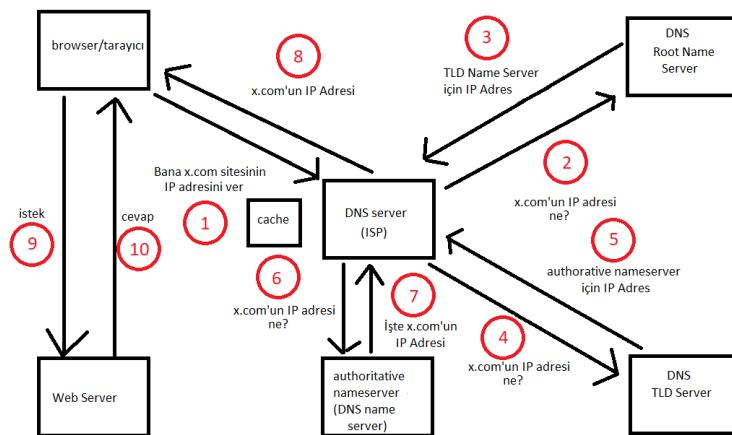
Name adresin domain adını, value ise mail sunucusunu gösterir. TTL (Time to Live) değeri, bu kaydın diğer DNS sunucularında ne kadar süreyle önbellekte saklanacağını belirleyen bir parametredir. TTL değeri, saniye cinsinden belirtilir ve her DNS kaydı için ayrı ayrı tanımlanabilir.

Bir mail gönderdiğimizi düşünelim. Aslında biz turuncu ile belirtilen "Mesaj" iletimini yapmak istiyoruz. Fakat mail gönderirken arka planda bazı olaylar olur. Bu olaylara resim üzerinden adım adım deşinelim.



- **Gönderici (İstemci):**
 - E-postayı gönderen kişi veya istemcidir. Bir mesaj göndermek ister.
- **MTA (Message Transfer Agent) - Gönderici:**
 - Gönderici MTA'sı, e-postayı gönderen istemcisinden alır ve SMTP sunucusuna iletir.
 - MTA ve MDA birbirine benzer görevler yapsa da amaçları farklıdır.
 - MTA, E-postaların bir sunucudan diğerine taşınmasından sorumludur.
 - MDA, E-postaların alıcının posta kutusuna teslim edilmesinden sorumludur.
- **SMTP Sunucusu (Gönderici):**
 - Göndericinin SMTP sunucusu, e-postayı alır ve internet üzerinden alıcının SMTP sunucusuna iletilmek üzere işler.

- Bu aşamada DNS sunucusuna başvurularak alıcının SMTP sunucusunun IP adresi belirlenir.
- **DNS Sunucusu:**
 - DNS (Domain Name System) sunucusu, alan adlarını IP adreslerine çevirir.
 - Gönderici SMTP sunucusu, alıcının SMTP sunucusunun IP adresini bulmak için DNS sunucusuna soru yapar. Çünkü SMTP sunucu alıcı SMTP sunucunun alan adını (domain name) bilir ihtiyacı olan şey IP adresidir. Bu alan adına karşılık gelen IP adresi DNS sunucuya sorar.
 - **DNS Sorgusu Nasıl Çalışır?**



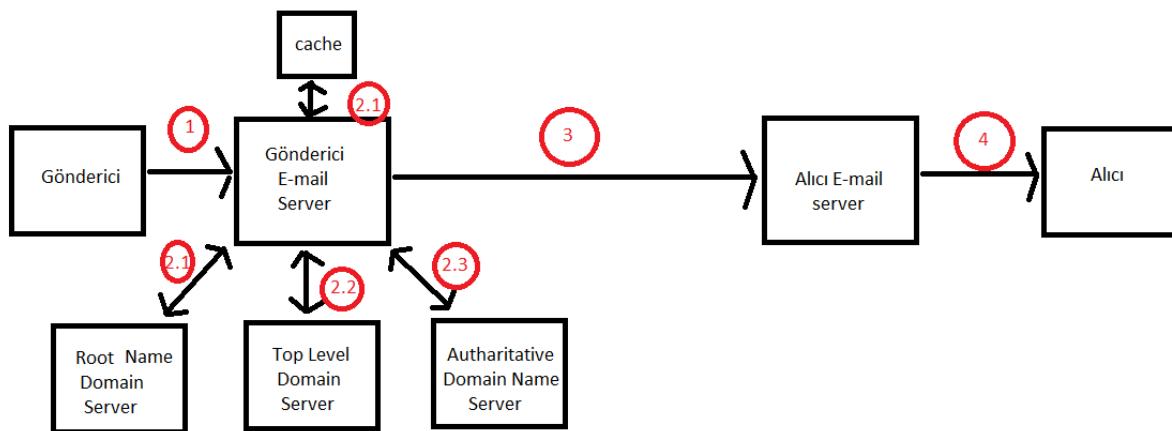
- Bu örnek web siteye erişirken arka planda dönen işlemlerin örneğidir. Bizim örneğimizde browser/tarayıcı yerine “SMTP Sunucu (Gönderici)”, Web Server yerine ise “SMTP Sunucu (Alıcı)” karşılık gelmektedir.
- Şimdi adımlara bakalım:
 - 1-Tarayıcıya x.com yazdığımızda biz arka planda DNS server'a yönlendiriliyoruz. Amaç DNS server'dan x.com'un IP adresini almak. Cache içinde IP adres bulunabilir ancak kalıcı olarak bulunmaz. Cache'IP adresi doğrudan alamadığımızı varsayıyalım. DNS server kendi içinde parçalara ayrılıyor. Bunlar: DNS Root Name Server, DNS Top Level Domain Server, DNS Authoritative Name Server. Bu aşamaları daha önce görmüştük. O yüzden burada atlıyoruz.
 - SMTP sunucuya DNS sunucudan bu şekilde IP adres döner. IP adresin yanında MX kaydı da iletilir. DNS sunucusu, SMTP sunucusuna IP adresi ve MX kaydı da dahil olmak üzere birden fazla bilgi döndürür. (MX kaydına yazının sonunda değineceğiz.)

- **Yönlendiriciler (Router) (R1, R2, R3, R4):**
 - E-posta, internet üzerinden çeşitli yönlendiricilerden (Router) geçerek alıcının SMTP sunucusuna ulaşır. Yönlendiriciler, e-postanın doğru yöne gitmesini sağlar.
- **SMTP Sunucusu (Alıcı):**
 - E-posta, alıcının SMTP sunucusuna ulaşır.

- Alıcının SMTP sunucusu, e-postayı alıcının posta kutusuna teslim etmek üzere işlemler yapar.
- **MTA (Message Transfer Agent) - Alıcı:**
 - Alıcının SMTP sunucusu, e-postayı alıcının posta kutusuna teslim ederken bir MTA kullanır.
 - Bu aşamada e-posta, alıcının posta kutusuna yerleştirilir.
- **IMAP/POP3 Sunucusu:**
 - Alıcı, e-postalarını almak için IMAP veya POP3 sunucusuna bağlanır. IMAP ve POP3, e-postaları bir e-posta sunucusundan bir e-posta istemcisine senkronize etmek için kullanılan iki farklı e-posta protokolüdür.
 - IMAP (Internet Message Access Protocol): E-postaları sunucuda tutar ve farklı cihazlardan erişim sağlar.
 - POP3 (Post Office Protocol version 3): E-postaları alıcının cihazına indirir ve genellikle sunucudan siler.
 - Bu aşamada başka protokoller de kullanılabilir. Ancak IMAP ve POP3 yaygın kullanılan iki ayrı protokoldür. Dolayısıyla bu iki protokolu ele aldık.
 - SMTP, e-posta iletilerinin gönderilmesi için kullanılan bir iletişim protokolüdür. POP3 ve IMAP ise e-posta alıcılarının, yani kullanıcıların, e-posta sunucusundan iletileri almak ve yönetmek için kullandıkları protokollerdir.
- **MDA (Mail Delivery Agent) - Alıcı:**
 - E-posta, alıcının posta kutusuna teslim edilir. Alıcının e-posta istemcisi, IMAP veya POP3 sunucusundan e-postaları alır.

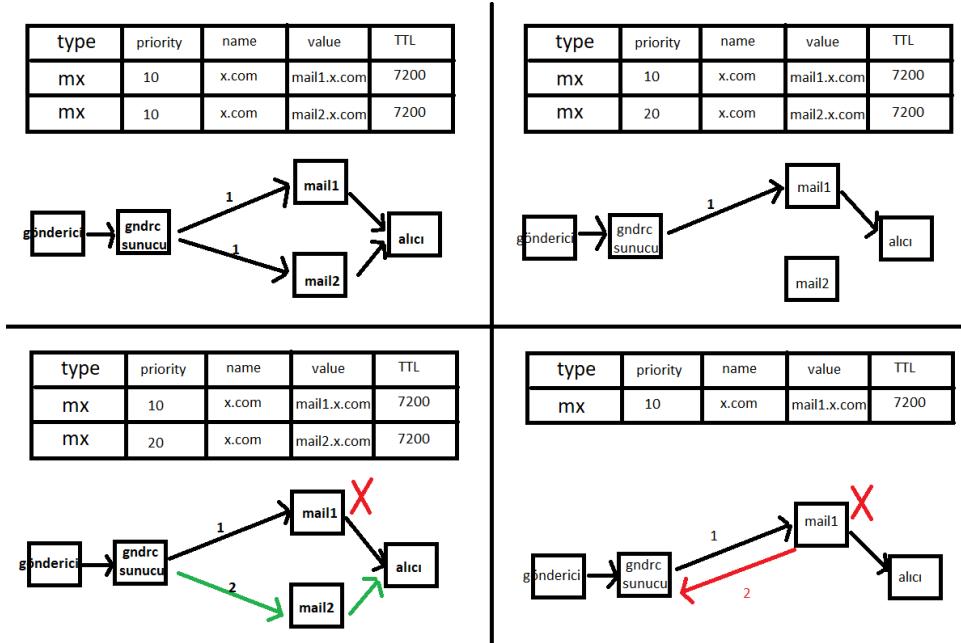
Dipnot! Bu örnekte MTA'ların ve Router'ların varlığı daha çok veya daha az olabilir. Bu görsel temsili bir resimdir.

İşte biz mail atarken arka planda bu aşamalar gerçekleşir. Şimdi tekrar DNS sunucu aşamasına dönelim. MX kaydı ve IP adres gibi bilgiler DNS sunucudan SMTP sunucusuna döner demıştık. MX Kaydını biraz daha inceleyelim.



Mx tipi DNS kaydına sahip DNS sunucunun çalışma prensibini inceleyelim. İlk olarak gönderici mail oluşturur ve gönderici mail server'a yönlendirir. Mail server DNS sunucuya gerekli bilgileri (MX kaydı, IP adres, ...) almak için başvurur. Eğer DNS sunucunun önbelleği (cache) içerisinde alıcı mail sunucunun adresi varsa mesaj direkt cache içindeki bilgi doğrultusunda alıcı mail sunucusuna gönderilir. Eğer cache içinde bu bilgi yoksa DNS'in çalışma prensibi ortaya çıkar. İlk önce root name'e hedefin adresini soruyoruz. O da bize sana bunu söyleyemem ama tld için bilgisini verebilirim diyor. Daha

sonra tld'ye (top level domain) hedefin adresi ne diye soruyoruz. Yine bize bilgiyi doğrudan vermiyor. Onun yerine authoritative name server için bilgiyi veriyor. En sonunda authoritative name server bize alıcı mail server'ın adresini veriyor ve mesaj o yöne doğru yönlendiriliyor. Tabii ki cache içinde hedefin adresi yoksa bu işlem gerçekleşiyor. Alıcı mail sunucu gelen mesajı alıcıya iletiyor.



Dikkat! Bu görseldeki mail1, mail1 sunucusunu ve mail2, mail2 sunucusunu belirtir.

PRIORITY (Öncelik)

1. örnekte (sol üst) göndericiden çıkan mail gönderici mail sunucuya ulaşıyor. Daha sonra mail1 sunucu ve mail2 sunucudan hangisinin öncelik değeri düşükse oraya yönlüyor. Fakat bu örnekte iki sunucuda öncelik değeri eşit. Bu durumda genelde rastgelelik durumu söz konusudur. Mesaj bir mail1 sunucusuna bir de mail2 sunucusuna rastgele ulaşıyor. Daha sonra mesaj mail sunucular doğrultusunda alıcıya gönderiliyor.

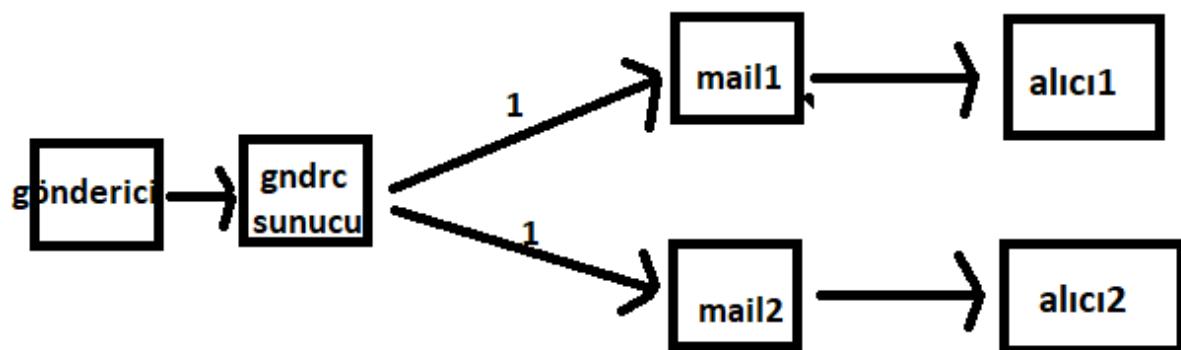
2. örnekte (sağ üst) en ideal durum ele alıyoruz. Göndericiden çıkan mail gönderici mail sunucuya ulaşıyor. Daha sonra mail1 sunucu ve mail2 sunucudan hangisinin öncelik değeri düşükse oraya yönlüyor. Görüldüğü gibi mesaj mail1 sunucusuna gidiyor. Çünkü öncelik (priority) değeri daha küçük. Daha sonra mesaj mail1 sunucusu doğrultusunda alıcıya ulaşıyor. Peki mail1 sunucusu kapasitesini doldurursa (overwhelm) veya offline olursa ne olur? Bu sorunun cevabı 3. örnekte yatıyor.

3. örnekte (sol alt) göndericiden çıkan mail gönderici mail sunucuya ulaşıyor. Daha sonra mail1 sunucu ve mail2 sunucudan hangisinin öncelik değeri düşükse oraya yönlüyor. Yine mesaj mail1 sunucusuna gidiyor. Çünkü öncelik (priority) değeri daha küçük. Fakat belli bir süre sonra mail1 sunucusunun kapasitesini doldurduğunu (overwhelm) veya offline olduğunu varsayıyalım. Bu durumda mesajlar artık mail2 sunucusuna yönlüyor. Daha sonra mesaj mail sunucular doğrultusunda alıcıya gönderiliyor. Peki tek bir alıcı mail sunucusu olsaydı ne olurdu? 4. örnekte ise bunu inceliyoruz.

4. örnekte (sağ alt) sadece 1 alıcı mail sunucusu var. Mesaj her zamanki gibi gönderici, gönderici sunucu ve alıcı mail1 sunucuya gidiyor. Fakat bir süre sonra overwhelm oluyor veya mail1 sunucusu offline duruma geçiyor. Bu durumda artık mesaj kabulü olmayacak ve gelen mesajlar aynen geri yönlendiriliyor. Böyle bir şey gerçekleşirse ortaya iletim hatası çıkıyor.

Not!

- Mail1 ve mail2 sunucuları, aynı alıcıya e-posta gönderebilirler. Bu durumda, MX kaydındaki öncelik sırasına göre e-posta hangi sunucuya yönlendirileceği belirlenir. (**Yukarıdaki görsel bu madde ile ilgili**)
- Mail1 ve mail2 sunucuları, farklı alıcılara e-posta gönderebilirler. Bu durumda, her alıcının kendine ait bir MX kaydı olması gereklidir ve bu kaytlara göre e-posta hangi sunucuya yönlendirileceği belirlenir. (**Aşağıdaki görsel**)
- Yani MX kaydındaki priority özelliği, birden fazla alıcı sunucu ve sadece bir alıcı (istemci) olduğunda kullanılır. Aşağıdaki görselde iki farklı alıcı ve birbirinden bağımsız sunucular bulunur. Bu durumda, priority özelliğinin kullanımı gereklidir. (mail1= mail1 sunucusu, mail2= mail2 sunucusu)



SOA Tipi DNS Kaydı

Type	MName	RName	Serial	Retry	TTL
SOA	ns1.x.com	admin.x.com	2023081401	3600	7200

SOA (Start of Authority) tipi DNS kaydı, bir DNS zone'un (bölgesinin) temel ayarlarını belirleyen bir tür kayittır. Hatırlarsak DNS (Domain Name System), alan adlarını IP adresleriyle eşlestiren bir sistemdi. SOA tipi DNS kaydı, belirli bir DNS zone (bölgesi) için yönetim yetkisini ve diğer önemli ayarları içerir.

Burada **Type** SOA tipi DNS kaydını göstermektedir. **Mname** ise master (primary, usta) sunucuyu temsil eder. Master ve slave sunucuya (primary, secondary) az sonra degeneceğiz. **Rname** bölgenin yönetiminden sorumlu kişinin e-posta adresini temsil eder. Bu alan, bölgenin yöneticisinin iletişim bilgilerini içerir. **Serial** bölgedeki kayıtlarda yapılan değişikliklerin sırasını belirten bir numaradır. Bu numara, bölgedeki herhangi bir kayıt değiştiğinde arttırılır. Serial numarası, diğer DNS sunucularına bölgedeki güncellemelerin sırasını ve yeni değişiklikleri iletmek için kullanılır. **Retry** ikincil (Secondary) DNS sunucularının birincil (Primary) sunucudan bölge verilerini almayı yeniden deneyecekleri zaman aralığını belirtir. Bu birincil ve ikincil yani master slave sunuculara az sonra degeneceğiz. **TTL** DNS kayıtlarının önbellekte ne kadar süreyle saklanacağını belirten bir değeri temsil eder. TTL değeri, saniye cinsinden ifade edilir ve bir DNS sunucusunun önbellekte tutulan kayıtları ne kadar süre boyunca kullanabileceğini belirler.

Dediğimiz gibi SOA tipi DNS kaydı, belirli bir DNS zone (bölgesi) için yönetim yetkisini ve diğer önemli ayarları içerir. Bunun için ilk önce DNS Zone'u öğrenelim.

DNS Zone Nedir?

DNS bölgesi (DNS zone), belirli bir alan adının veya bir alan adı grubunun yönetildiği ve bu alan adlarının IP adresleriyle ilişkilendirildiği bir alandır. DNS, domain isimlerini IP adreslerine çevirmek için kullanılan bir sistemdir. DNS zone, bu isim-IP çevirme işlemini gerçekleştirmek için gerekli kayıtları içerir.

Start of Authority (SOA) DNS kaydı, DNS zone'un temelini oluşturan ve bölgenin yönetimini sağlayan en önemli kayıtlardan biridir. SOA kaydı, bir DNS bölgesinin başlangıcını belirtir ve bölge yönetiminin nasıl yapılandırılacağı tanımlar. Bu nedenle, SOA kaydı DNS zone ile sıkı bir ilişkiye sahiptir.

DNS Zone görevleri:

Alan Adının Yönetimi: DNS zone, belirli bir alan adının yönetimini sağlar. Bu alan adı, internet üzerinde veya özel bir ağda bulunabilir. Alan adının alt alan adları ve bu alt alan adlarının kayıtları bu bölgede düzenlenir. Baktığımızı zaman DNS Zone'un en temel görevi alan adı yönetimi diyebiliriz.

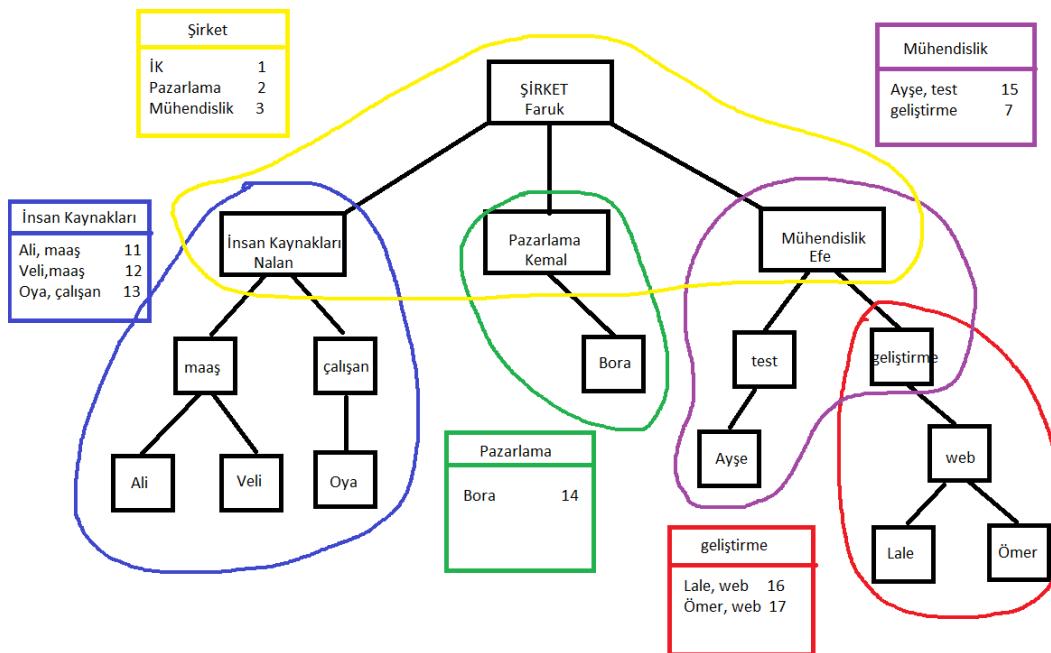
DNS Kayıtlarını İçerir: DNS zone, alan adının IP adresleriyle ilişkilendirildiği DNS kayıtlarını içerir. Bu kayıtlar, A kayıtları (IPv4), AAAA kayıtları (IPv6), MX kayıtları (e-posta sunucuları), CNAME kayıtları (kanonik ad), PTR kayıtları (ters çevrim), SOA kayıtları ve daha fazlasını içerebilir.

Yönetim Yetkisi: DNS zone, bölgeyi yönetme yetkisine sahip otoriter DNS sunucularını belirtir. Bu sunucular, alan adının kayıtlarını saklar ve diğer DNS sunucularına dağıtır.

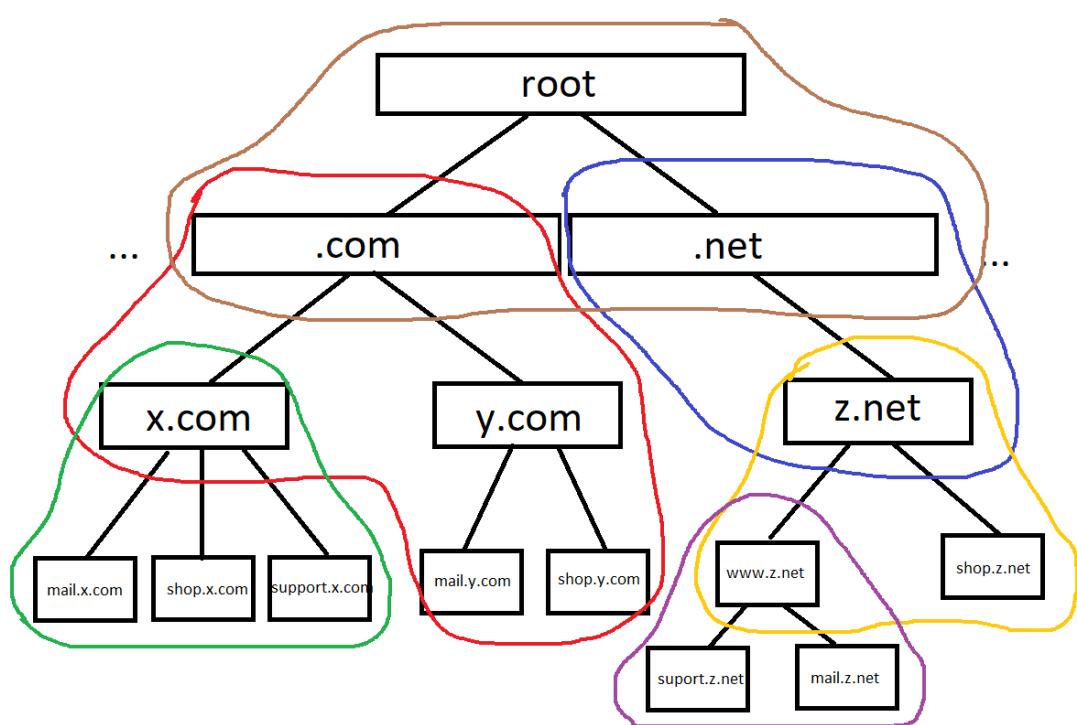
Güncelme ve Senkronizasyon: DNS zone, kayıtlarda yapılacak güncellemeleri yönetir. Yeni bir alan adı veya alt alan adı oluşturulduğunda, IP adresleri değiştiğinde veya diğer değişiklikler yapıldığında, bu değişiklikler zone üzerinde yapılır ve diğer DNS sunucularına senkronize edilir.

DNS Hiyerarşisinin Parçası: DNS zone, genel DNS hiyerarşisinin alt seviyelerini oluşturur. Alan adları ve alt alan adları, bu hiyerarşi içinde farklı DNS bölgelerini temsil eder.

Alan Adı Dağıtımı: DNS zone, alan adlarının IP adresleriyle ilişkilendirilmesini sağladığı için kullanıcıların alan adlarını IP adreslerine çözümlemelerini mümkün kılar. Bu sayede internet kullanıcıları, alan adlarına dayalı olarak web sitelerine, e-posta sunucularına ve diğer hizmetlere erişebilirler.

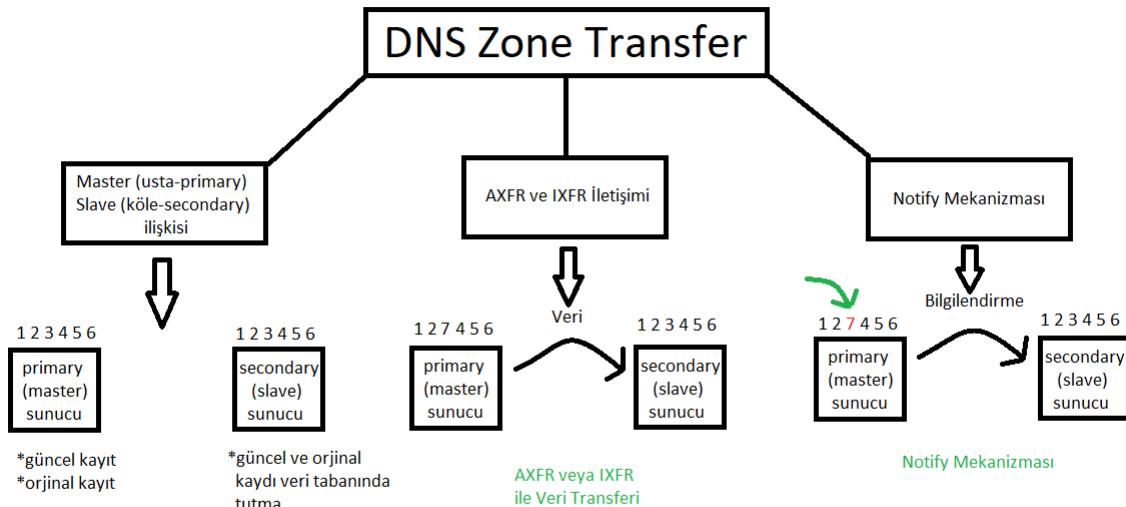


Şimdi DNS Zone'u anlamaya çalışalım. Bir şirket düşünelim. Mesela elimizde 5 tane zone (bölge) olsun. Görüldüğü gibi dikdörtgen kutularda iletişim bilgileri bulunmaktadır. Bunun yanı sıra DNS Zone'larda o zone ile ilgili yönetim bilgileri, iletişim bilgileri veya daha fazla bilgi bulunur. Bu şirketörneğinde sadece iletişim bilgisini gösterdik. Örneğin Sarı renkli zone'da bulunan şirket sahibi Faruk, insan kaynaklarından sorumlu Nalan ile görüşecek diyalim. Bu işlemi doğrudan yapabilir çünkü aynı zone içindeler. Ama pazarlama biriminde çalışan Bora ile iletişimde bulunmak isteseydi, Faruk pazarlama biriminin başındaki Kemal ile görüşecekti ve bilgileri ondan alacaktı. Çünkü Bora'nın bulunduğu zone'daki kilit nokta Kemal.



Resimde görüldüğü gibi Domain adlarını zone'lara ayıralım. DNS zone'lar arası iletişim biraz daha karmaşıktır. Bunu için DNS Zone Transfer kullanılabılır.

DNS Zone Transfer'i anlamak için bazı terimleri bilmekte fayda vardır. Bu terimleri 3 farklı terimi anlamak lazım: Bunlar master-slave ilişkisi, AXFR ve IXFR iletişimini, Notify Mekanizması.



Master-Slave ilişkisi: DNS zone'lar arası iletişimde master-slave (ustaca-köle) ilişkisi yer alır. Master (usta) DNS sunucusu (Primary Zone), bölgenin orijinal ve güncel kayıtlarını saklar. Slave (köle) DNS sunucuları ise (Secondary Zone) master sunucusundan bu kayıtları düzenli olarak alır ve kendi veritabanlarında saklar. Slave sunucular, master sunucudan gelen güncellemelere tepki verir ve senkronize olur.

Primary Zone (Birincil Bölge): Bir Primary Zone, bir DNS bölgesinin asıl ve güncel kayıtlarını içeren bölgedir. Bu zone, bölgenin orijinal kayıtlarını saklar, güncellemeleri kabul eder ve yönetim yetkisine sahiptir. Primary Zone sahibi, bu bölge için kayıt ekleme, düzenleme ve silme işlemlerini gerçekleştirir. Kayıtlarda yapılan güncellemeler, Primary (master-usta) Zone'da gerçekleştirildikten sonra Secondary Zone'lara (köle-slave) aktarılır.

Secondary Zone (İkincil Bölge): Bir Secondary Zone, Primary Zone'dan (birincil bölge) veri kopyalayarak güncelleyen bölgedir. Secondary Zone, yedeklenmiş bir kopya olarak hizmet verir ve bu kopya, Primary Zone'dan düzenli aralıklarla veya bildirim mekanizmalarıyla güncellenir. Secondary Zone, DNS sunucuları arasında yük dengesi ve yedeklenmiş verilerin sağlanması için kullanılır. Eğer Primary Zone'daki sunucu çevrimdışı kalırsa, Secondary Zone hala güncel verileri sunabilir.

AXFR ve IXFR İletişimi: Zone Transfer (Bölge Aktarımı) olarak da adlandırılan AXFR (Full Zone Transfer) ve IXFR (Incremental Zone Transfer), master sunucudaki DNS kayıtlarının slave sunuculara aktarılmasını sağlar. AXFR, bütün bölge kayıtlarının tam bir kopyasını slave sunucuya ileterek güncelleme sağlar. IXFR ise yalnızca değişen veya eklenen kayıtları slave sunucuya ileterek daha verimli bir güncelleme sağlar.

Notify Mekanizması: Master sunucu, kayıtlarda bir değişiklik olduğunda slave sunucuları hızlıca bilgilendirmek için Notify mekanizmasını kullanır. Slave sunucular, bu bildirimi aldıktan sonra güncellemeleri almak için gerekli işlemi başlatır.

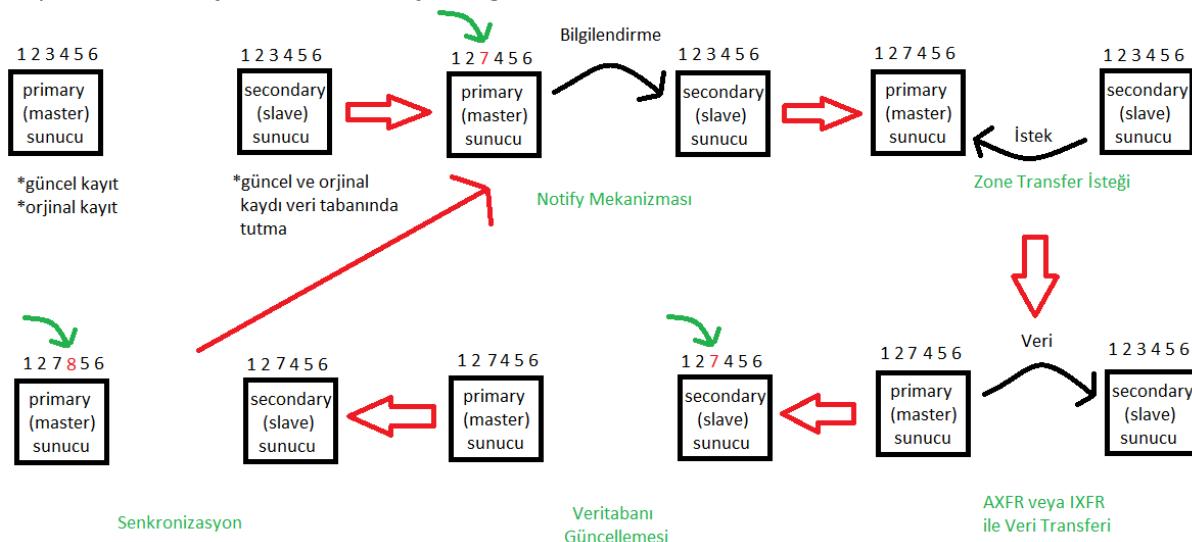
Notify mekanizmasının DNS Zone transferlerindeki rolünü yerine getiren bazı alternatifler şunlar olabilir: Periodic Zone Transfer (Düzenli Bölge Transferi), DNS Kaynak Sorgulaması (DNS Resource Query), Daha Düşük TTL Değerleri, Kendi Kendine Güncelleme (Self-Notify). Unutulmaması gereken önemli bir nokta, Notify mekanizmasının hızlı ve etkili bir şekilde bölge transferlerini sağlayan yaygın olarak kabul görmüş bir yöntem olduğunu söylemektedir. Diğer alternatif yöntemler, duruma ve gereksinimlere bağlı olarak kullanılabilecek seçeneklerdir ancak genellikle Notify mekanizması daha hızlı ve verimli sonuçlar elde etmeye yardımcı olur.

Master (Primary) Sunucu Örnekleri:

ns1.example.com: Örneğin, "example.com" alan adının Primary (Master) sunucusu olan ns1.example.com sunucusu bölgenin orijinal ve güncel verilerini içerir. Bu sunucu üzerinde yapılan değişiklikler, diğer sunuculara iletilir.

Slave (Secondary) Sunucu Örnekleri:

ns2.example.com: "example.com" alan adının Secondary (Slave) sunucusu olan ns2.example.com sunucusu, ns1.example.com'dan bölge kayıtlarını kopyalayarak günceller. Bu sunucu, yedeklenmiş ve yedeklenmemiş verilere hızlı erişim sağlar.



Master (Primary) ve Slave (Secondary) Sunucular: DNS Zone Transfer işlemi, bir Primary Zone (Master) sunucusu ile bir veya daha fazla Secondary Zone (Slave) sunucusu arasında gerçekleşir. Primary sunucu, bölgenin orijinal ve güncel kayıtlarını içerirken, Secondary sunucular bu kayıtları kendi veritabanlarında saklayarak yedek ve hızlı yanıt sunarlar.

Notify Mekanizması (İsteğe Bağlı): Primary sunucu, kayıtlarda bir değişiklik olduğunda Notify mekanizması ile Secondary sunucuları hızlıca bilgilendirir. Bu mekanizma, güncellemelerin daha hızlı iletilmesini sağlar, ancak zorunlu değildir.

Zone Transfer İsteği: Secondary sunucular, belirli aralıklarla veya Notify mekanizmasıyla Primary sunucudan bölge verilerini almak üzere Zone Transfer isteği gönderirler.

AXFR veya IXFR Mekanizması: Primary sunucu, gelen Zone Transfer isteğine AXFR (Full Zone Transfer) veya IXFR (Incremental Zone Transfer) protokollerinden birini kullanarak yanıt verir.

AXFR (Full Zone Transfer): Primary sunucu, bütün bölge kayıtlarını tam bir kopya olarak Secondary sunucuya gönderir. Bu, yeni bir Secondary Zone oluşturulduğunda veya büyük değişiklikler olduğunda kullanılır.

IXFR (Incremental Zone Transfer): Primary sunucu, yalnızca bölgelerdeki değişen veya eklenen kayıtları Secondary sunucuya gönderir. Bu şekilde, bölgelerdeki küçük değişiklikler veya günlük güncellemeler gibi durumlar için daha verimli bir seçenektedir.

Veri Transferi: Primary sunucu, AXFR veya IXFR mekanizması kullanarak bölge verilerini Secondary sunucuya aktarır.

Veritabanı Güncellemesi: Secondary sunucu, gelen bölge verilerini kendi veritabanına entegre eder ve güncelleme işlemi tamamlanır.

Senkronizasyon: Secondary sunucu, belirli aralıklarla veya Notify mekanizması ile güncellemeleri kontrol eder ve gerektiğinde tekrar Zone Transfer isteği göndererek bölge verilerini güncel tutar.

SOA Tipi DNS kaydından neden DNS Zone'a geldik diye sorarsak şu cevapla karşılaşırız: SOA Tipi DNS kaydını daha iyi anlamak için DNS Zone'u bilmemiz gereklidir. Çünkü SOA (Start of Authority) kaydı, bir DNS zone'un temel bilgilerini içeren ve bu zone'un yönetimini tanımlayan bir kayıttır. SOA kaydı, birincil (Primary, master, usta) DNS sunucusunun adını, e-posta adresini, seri numarasını, güncelleme sıklığını, yeniden deneme süresini, geçerlilik süresini (TTL) ve diğer bölge yönetimi parametrelerini içerir. Bu yüzden DNS Zone ve DNS Zone Transferi anlamaya çalıştık.

NS Tipi DNS Kaydı

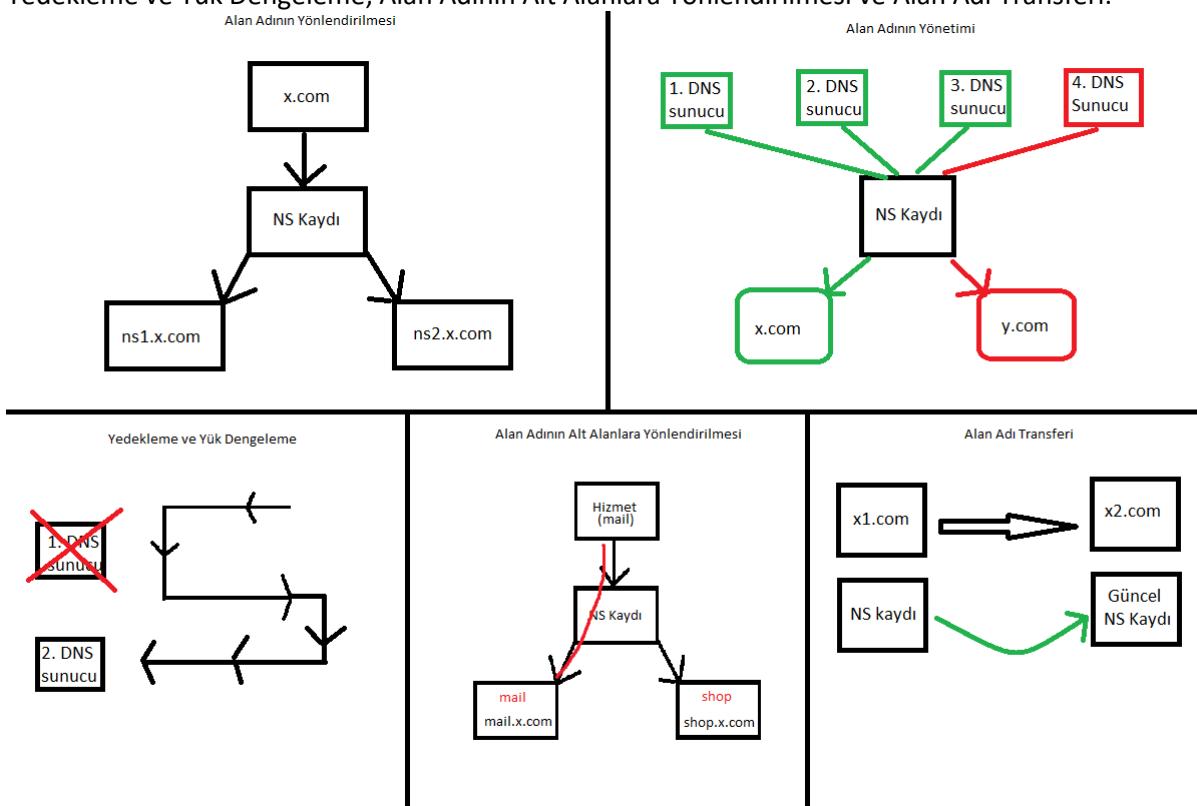
TYPE	Value	Name	TTL
NS	ns1.x.com	x.com	7200
NS	ns2.x.com	x.com	7200

NS tipi DNS kaydı (Name Server Record), bir alan adının (domain adı) DNS (Domain Name System) hizmetini sağlayan sunucuların adreslerini belirtir. Bu kayıtlar, hangi DNS sunucularının belirli bir alan adının IP adresini çözümlemesine yardımcı olduğunu gösterir. NS kayıtları, bir alan adının DNS yapılandırması ve yönlendirmesi için önemlidir.

Bu örnekte **Type** ns tipi DNS Kaydını belirtiyor. **Value** ise sunucuların adlarını gösteriyor. Mesela burada ns1.x.com primary (master) sunucu, ns2.x.com ise secondary (slave) sunucu olabilir. ns1.x.com ve ns2.x.com adreslerini subdomainler ile karıştırmamak lazım. Burada bahsettiğimiz şey primary ve secondary sunucu. Hatırlarsak Master ve Slave sunucularına DNS Zone (SOA Tipi DNS Kaydı) başlığı altında dephinmişтик. **Name**, ana domain adını temsil ediyor. **TTL** DNS kayıtlarının önbellekte ne kadar süreyle saklanacağını belirten bir değeri temsil eder. TTL değeri, saniye cinsinden ifade edilir ve bir DNS sunucusunun önbellekte tutulan kayıtları ne kadar süre boyunca kullanabileceğini belirler.

Alan Adının Yönlendirilmesi	Alan Adının Yönetimi	Yedekleme ve Yük Dengeleme	Alan Adının Alt Alanlara Yönlendirilmesi	Alan Adı Transferi
-----------------------------	----------------------	----------------------------	--	--------------------

NS tipi DNS Kaydının bazı görevleri vardır. Bunlar: Alan Adının Yönlendirilmesi, Alan Adının Yönetimi, Yedekleme ve Yük Dengeleme, Alan Adının Alt Alanlara Yönlendirilmesi ve Alan Adı Transferi.



- 1- NS (Name Server) kayıtları, bir alan adının DNS hizmetini sağlayan yetkili DNS sunucularının adreslerini belirtir. NS kayıtları, bir alan adının hangi DNS sunucularının bu alan adının DNS hizmetini sağladığını belirler. Yani, NS kayıtları domain adını, DNS sunucularına (primary, secondary) yönlendirmesi için kullanılır.
- 2- Her alan adı, bir veya daha fazla yetkili DNS sunucusu tarafından yönetilir. Hangi DNS sunucusunun hangi alan adını yöneteceğini NS kayıtları belirler. Bu sunucular, alan adının DNS kayıtlarını saklar ve günceller.
- 3- NS kayıtları, birden fazla DNS sunucusu kullanarak yedekleme ve yük dengeleme sağlamak için kullanılabilir. Böylece, bir sunucu hizmet dışı kalmadığında veya yoğunluk arttığında, trafik diğer DNS sunucularına yönlendirilir.

- 4- Büyük organizasyonlar veya hizmet sağlayıcıları, farklı alt alanlara farklı hizmetleri yönlendirmek için NS kayıtlarını kullanabilir. Örneğin, "mail.x.com" alt alanı e-posta hizmeti için, "shop.x.com" ise alış veriş için yönlendirilebilir.
- 5- Bir alan adının sahibi veya hizmet sağlayıcısı değiştiğinde, NS kayıtları da güncellenir. Bu şekilde, alan adının yeni sahibi veya hizmet sağlayıcısı, DNS hizmetini sağlayan sunucuları kontrol edebilir.

TXT Tipi DNS Kaydı

TXT DNS kaydı (TXT record), DNS (alan adı sistemi) üzerinde metin tabanlı verileri (string) depolamak için kullanılan bir tür DNS kaydıdır. İçerisinde metin verileri (string) bulunur ve gereken işleme göre kullanılır.

TXT DNS kaydında belirli değerler vardır. **Type** TXT tipinde olduğunu belirtir. **Name** domain adını gösterir. **Value** ise diğer DNS kayıt tiplerinden farklı olarak burada bulunmaktadır ve TXT DNS kaydı içindeki string değere işaret eder. **TTL** DNS kayıtlarının önbellekte ne kadar süreyle saklanacağını belirten bir değeri temsil eder. TTL değeri, saniye cinsinden ifade edilir ve bir DNS sunucusunun önbellekte tutulan kayıtları ne kadar süre boyunca kullanabileceğini belirler.

TYPE	Name	Value	TTL
TXT	x.com	Bu x.com domain'i için bir string değeridir.	7200

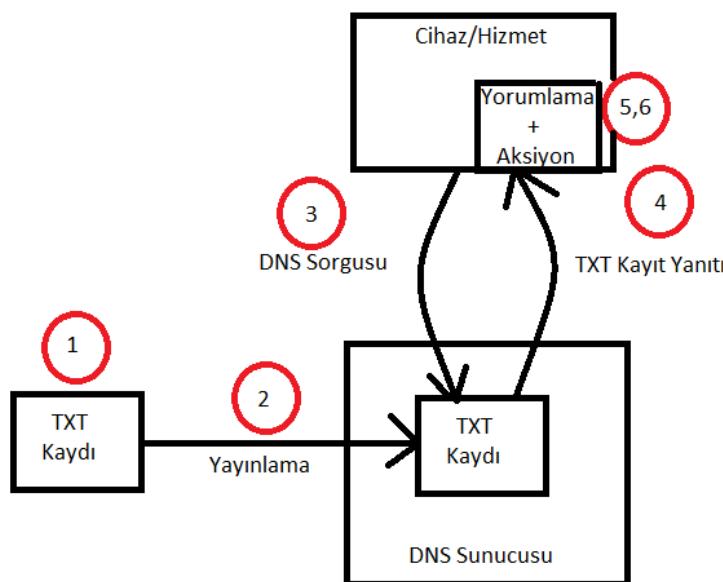
E-posta Doğrulama (SPF, DKIM, DMARC)	Alan Adı Sahipliği Doğrulama	Hizmet Keşfi ve Kimlik Doğrulama
Genel Bilgi Depolama	Belirli Protokol Uygulamaları	Belirli Servislerin Yönlendirilmesi

Bazı yaygın kullanım alanları vardır:

- 1- **E-posta Doğrulama (SPF, DKIM, DMARC):** TXT kayıtları, e-posta doğrulama protokollerini kullanır. SPF (Gönderen Politikası Çerçeve) kayıtları, bir alan adının hangi IP adreslerinin e-posta gönderebileceğini belirler. DKIM (Alan Adı Anahtarını Tanımlama) kayıtları, e-posta mesajlarının kimliğini doğrulamak için kullanılır. DMARC (Alan Adı Tabanlı Mesaj Kimliği Doğrulaması ve Uyarısı) ise SPF ve DKIM doğrulamasını birleştirerek sahte e-postaları tespit etmek için kullanılır.
- 2- **Alan Adı Sahipliği Doğrulama:** Bazı hizmetler, alan adının sahibi olduğunuzu doğrulamak için TXT kayıtlarını isteyebilir. Bu, alan adınızı doğrulayarak çeşitli hizmetlerin kontrolünü ele geçirmenizi sağlar.

- 3- **Hizmet Keşfi ve Kimlik Doğrulama:** TXT kayıtları, bir hizmetin doğru kimlik bilgilerini sağlama veya belirli hizmetlerin varlığını bildirme amaçlarıyla kullanılabılır. Örneğin, bir hizmet sağlayıcısı, OAuth veya OpenID kimlik doğrulama süreçlerini desteklemek için TXT kayıtları kullanabilir.
- 4- **Genel Bilgi Depolama:** TXT kayıtları, genel bilgi depolamak veya paylaşmak için de kullanılabilir. Örneğin, bir web sitesinin yönetici, ziyaretçilere veya otomatik sistemlere belirli bilgileri sağlamak için TXT kayıtlarını kullanabilir.
- 5- **Belirli Protokol Uygulamaları:** Bazı protokoller, TXT kayıtlarını belirli amaçlarla kullanabilir. Örneğin, Bitmessage protokolü, mesaj gönderme ve alımını sağlamak için TXT kayıtlarını kullanır.
- 6- **Belirli Servislerin Yönlendirilmesi:** TXT kayıtları, belirli hizmetlerin veya alt alan adlarının (subdomain) IP adreslerine yönlendirilmesi için kullanılabilir. Bu, özel hizmetlerin veya alt alan adlarının farklı sunucularda barındırılması durumunda kullanışlı olabilir.

TXT kayıtları genellikle insanlar tarafından okunması daha zordur ve çoğunlukla otomatik sistemler veya belirli protokoller tarafından yorumlanır. Bu nedenle, TXT kayıtlarını oluştururken dikkatli olunması ve gerektiğinde belirli protokollere uygun şekilde formatlanmış olması gereklidir.



TXT DNS kaydının çalışma süreci:

TXT Kaydı Oluşturma: Bir alan adı sahibi veya yönetici, DNS yönetim araçları veya hizmet sağlayıcısının kontrol paneli aracılığıyla bir TXT kaydı oluşturur. Kayıt içeriği, metin tabanlı veriyi içerir. Bu veri genellikle belirli bir protokol veya hizmet tarafından belirlenmiş bir yapıya sahiptir.

Sunucularına Yayınlama: TXT kaydı, alan adınızın DNS sunucularına yayınlanır. DNS sunucuları, alan adınızın IP adresi ile ilişkilendirilmesi gereken isteklere yanıt verirler. Bu yanıtlar, DNS sorgularını gönderen cihazlara veya hizmetlere döner.

DNS Sorgusu: Bir cihaz veya hizmet, belirli bir alan adının TXT kaydını almak için DNS sorgusu yapar. Bu sorgu, alan adının TXT kaydını içeren DNS sunucularına yönlendirilir.

TXT Kaydı Yanıtı: DNS sunucusu, TXT kaydını içeren yanıt döner. Yanıt, genellikle bir dizi metin satırından oluşur.

Protokol veya Hizmet Yorumlaması: TXT kaydının içeriği, belirli bir protokol veya hizmet tarafından yorumlanır. Örneğin, SPF veya DKIM gibi e-posta güvenliği protokolleri, TXT kaydındaki metin veriyi kullanarak e-postaların doğruluğunu veya kimliğini doğrular.

Aksiyon Alınması: TXT kaydının içeriğine bağlı olarak, çeşitli aksiyonlar alınabilir. Örneğin, SPF kaydı ile gönderici sunucuların doğruluğu kontrol edilirken, DMARC kaydı sahte e-postaları tespit etmek için kullanılır.

PTR Tipi DNS Kaydı

TYPE	IP addr (IPv4/IPv6)	Name	TTL
PTR	1.2.3.4	x.com	3600

PTR DNS kaydı, "Pointer" kısaltması ile ifade edilen ve genellikle "Reverse DNS" olarak da adlandırılan bir tür DNS (Domain Name System) kaydıdır. Geleneksel DNS kayıtları IP adreslerini alan adlarına çevirmek için kullanılırken, PTR Tipi DNS Kaydı IP adreslerini alan adlarına çevirmek için kullanılır.

Bu, özellikle e-posta sunucularının gelen e-postaların gönderen sunucularını doğrulamasına yardımcı olmak için kullanılır. Yani spam gibi mailler bu sayede anlaşılabılır. Bir IP adresi alındığında, PTR kaydı ile bu IP adresine karşılık gelen alan adını bulmak mümkün hale gelir.

Özetle, PTR DNS kayıtları, IP adreslerini alan adlarına çevirmek için kullanılır ve genellikle ağıdaki cihazların tanımlanmasında ve güvenlik denetimlerinde kullanılır.

PTR DNS kaydında belirli değerler vardır. **Type** PTR tipinde olduğunu belirtir. **IP addr** domain'in IP adresini belirtir. IPv4 veya IPv6 formatında olabilir. Bu örnekte IPv4 görüyoruz. **Name** domain adını gösterir. **TTL** DNS kayıtlarının önbellekte ne kadar süreyle saklanacağını belirten bir değeri temsil eder. TTL değeri, saniye cinsinden ifade edilir ve bir DNS sunucusunun önbellekte tutulan kayıtları ne kadar süre boyunca kullanabileceğini belirler.

Güvenlik ve Doğrulama	Ağ Tanımlama ve İzleme	Güvenlik Analizi	İnternet Servis Sağlayıcıları	Ters IP Adresi Çözümlemesi
--------------------------	---------------------------	------------------	----------------------------------	-------------------------------

PTR kayıtları genellikle aşağıdaki amaçlarla kullanılır:

Güvenlik ve Doğrulama: E-posta sunucuları, gelen e-postaların gönderen sunucularını doğrulamak için PTR kayıtlarını kullanabilir. E-posta sunucusuna gelen bir e-postanın kaynağına dair güvenilir bilgi elde etmek amacıyla PTR kaydı kontrol edilir. Eğer PTR kaydı bulunmuyorsa veya uygun değilse, bu e-postalar spam veya kimlik avı gibi kötü amaçlı e-posta türleri olabileceği düşünülerek reddedilebilir.

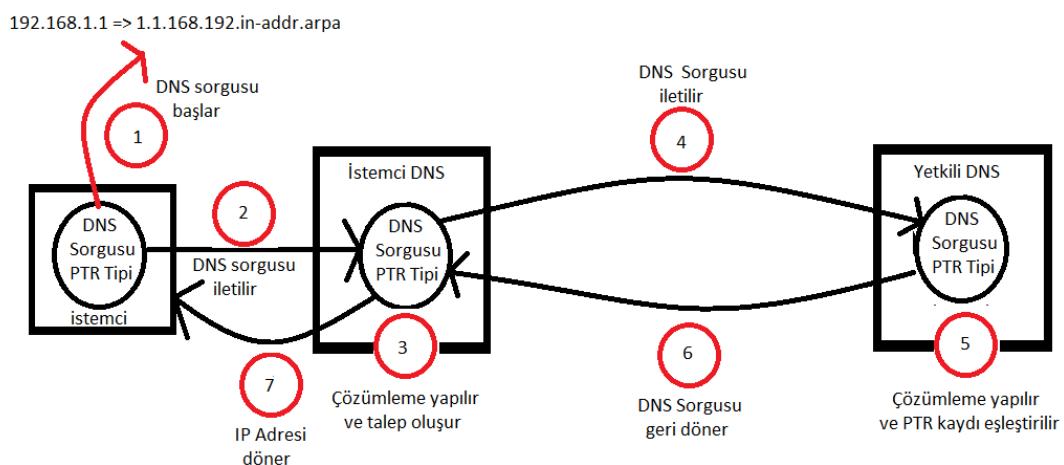
Ağ Tanımlama ve İzleme: PTR kayıtları, ağdaki cihazların IP adreslerine karşılık gelen alan adlarını içerir. Bu, ağ yöneticilerinin hangi cihazın hangi IP adresine sahip olduğunu daha kolay bir şekilde belirlemelerini sağlar. Ağda izleme, sorun giderme veya güvenlik denetimleri yaparken bu bilgilerden yararlanılabilir.

Güvenlik Analizi: Güvenlik analistleri, PTR kayıtlarını kullanarak belirli IP adreslerini alan adlarına çevirerek ağ trafiğini izleme ve anlama yeteneklerini artırabilirler. Bu, güvenlik olaylarını tespit etme ve yanıtlama süreçlerini destekler.

İnternet Servis Sağlayıcıları: İnternet hizmet sağlayıcıları, PTR kayıtlarını kullanarak ağlarındaki IP adreslerini alan adlarına çevirerek kullanıcılarının doğru şekilde tanımlandığından emin olabilirler.

Ters IP Adresi Çözümlemesi: PTR kayıtları, IP adreslerini alan adlarına çevirerek ağ yöneticilerine ve servis sağlayıcılara IP trafiği ve kaynaklarını daha iyi anlama ve izleme imkanı sunar.

Çalışma Prensibi şu şekildedir:



- 1- İstemci, alan adını veya IP adresini alan adına çevirme ihtiyacı doğduğunda, kendi DNS sunucusuna bir sorgu gönderir. Bu sorgu, genellikle otomatik olarak oluşturulur ve PTR sorgusu olduğunu belirtir. Cihazın IP adresi "192.168.1.1" ise, PTR sorgusu "1.1.168.192.in-addr.arpa" şeklinde bir alan adı oluşturur.
- 2- İstemci DNS sunucusu, PTR sorgusunu alır
- 3- İstemci DNS sunucusu bu sorguyu çözümlemeye çalışır. PTR sorgusu, IP adresini alan adına çevirme talebini içerir. İstemci DNS sunucusu, PTR sorgusundaki IP adresini temel alarak

gerektiğinde PTR kaydını içeren yetkili (authoritative) DNS sunucusunu belirler. Bu ad sunucusu, ters DNS alanını yöneten ve PTR kayıtlarını içeren sunucudur.

- 4- İstemci DNS sunucusu, PTR sorgusunu yetkili DNS sunucusuna ileterek çözümleme işlemini gerçekleştirir.
- 5- Yetkili DNS sunucusu, aldığı PTR sorgusunu çözümleyerek ilgili PTR kaydını bulur ve cevabı istemci DNS sunucusuna gönderir. Yetkili DNS sunucusu, sorgulanın IP adresi için bir PTR kaydı bulursa, bu kaydı cevap olarak istemciye döner. Yani yerel DNS sunucusu, PTR kayıtlarını içeren zone dosyasını sorgulanmış alan adıyla eşleştirir ve sonuçları döndürür.
- 6- İstemci DNS sunucusu DNS sorgusunu geri alır.
- 7- İstemci DNS sunucusu, aldığı cevabı kendi istemcisine ileterek IP adresini alan adına çevirme işlemini tamamlar.

SRV Tipi DNS Kaydı

TYPE	Name	Priority	Weight	Port	TTL	Target
SRV	_sip._tcp.x.com	5	5	5060	7200	sip.x.com
SRV	_smtp._tcp.x.com	10	0	587	7200	smtp.x.com

SRV (Service) tipi DNS kaydı, belirli bir hizmetin (service) bir alan adındaki kaynaklarını tanımlayan bir tür DNS (Domain Name System) kaydıdır. SRV (Service) tipi DNS kaydı, bir alan adındaki belirli bir hizmetin (service) konumunu ve iletişim parametrelerini tanımlayan bir tür DNS (Domain Name System) kaydıdır. Genellikle VoIP (Voice over IP), IM (Instant Messaging), e-posta (e-mail) gibi uygulamaların, sunucuların ve hizmetlerin yerlerini belirlemek için kullanılır.

SRV DNS kaydında belirli değerler vardır. **Type** SRV tipinde DNS kaydı olduğunu belirtiyor. **Name** hizmet, protokol ve domain adını tutmakta. İlk SRV Kaydına baktığımızda hizmet adı _sip, protokol adı _tcp ve domain adı x.com olarak belirlemiştir. **Priority** öncelik değerini ifade eder. Hatırlarsak priority özelliğini MX tipi DNS kaydında da görmüştük. priority (öncelik) değeri, aynı etki alanına ait birden fazla SRV kaydı olduğunda hangi sunucunun öncelikli olarak kullanılması gerektiğini belirleyen bir sayısal değerdir. Düşük sayılar daha yüksek önceliği temsil eder. **Weight** önceliği (priority) aynı olan kayıtlar arasında trafiği nasıl dağıtılabileceğini belirtir. Daha yüksek ağırlık, daha fazla trafiği çekme olasılığını ifade eder. **Port** Hizmetin çalıştığı port numarasını belirtir. Bu, istemci ve sunucu arasındaki iletişimde kullanılacak portu tanımlar. **TTL** DNS kayıtlarının önbellekte ne kadar süreyle saklanacağını belirten bir değeri temsil eder. TTL değeri, saniye cinsinden ifade edilir ve bir DNS sunucusunun önbellekte tutulan kayıtları ne kadar süre boyunca kullanabileceğini belirler. **Target**, hizmetin fiziksel konumunu veya IP adresini belirtir. Bu, istemcilerin doğru sunucuya yönlendirilmesini sağlar.

Yük Dengelemesi	Yedekleme ve Yüksek Erişilebilirlik	Protokol ve Port Esnekliği
Alan Adı Değişiklikleri ve Taşıma	Karmaşık Ağ Yapıları	Uygulama ve Hizmet Keşfi

SRV DNS kaydının kullanım nedenleri

Yük Dengelemesi: SRV kayıtları, bir hizmetin birden fazla sunucusunda çalıştığı durumlarda yük dengelemesi sağlamak için kullanılır. Örneğin, bir VoIP hizmetinin birden fazla sunucusu varsa, istemciler SRV kayıtları aracılığıyla farklı sunuculara yönlendirilerek yük dengelemesi gerçekleştirilebilir.

Yedekleme ve Yüksek Erişilebilirlik: SRV kayıtları, hizmetlerin yedeklenmesi ve yüksek erişilebilirlik gerektiren durumlarda kullanılır. Eğer ana sunucu çökerse veya hizmet verilemez hale gelirse, SRV kayıtları sayesinde istemciler otomatik olarak yedek sunuculara yönlendirilebilir.

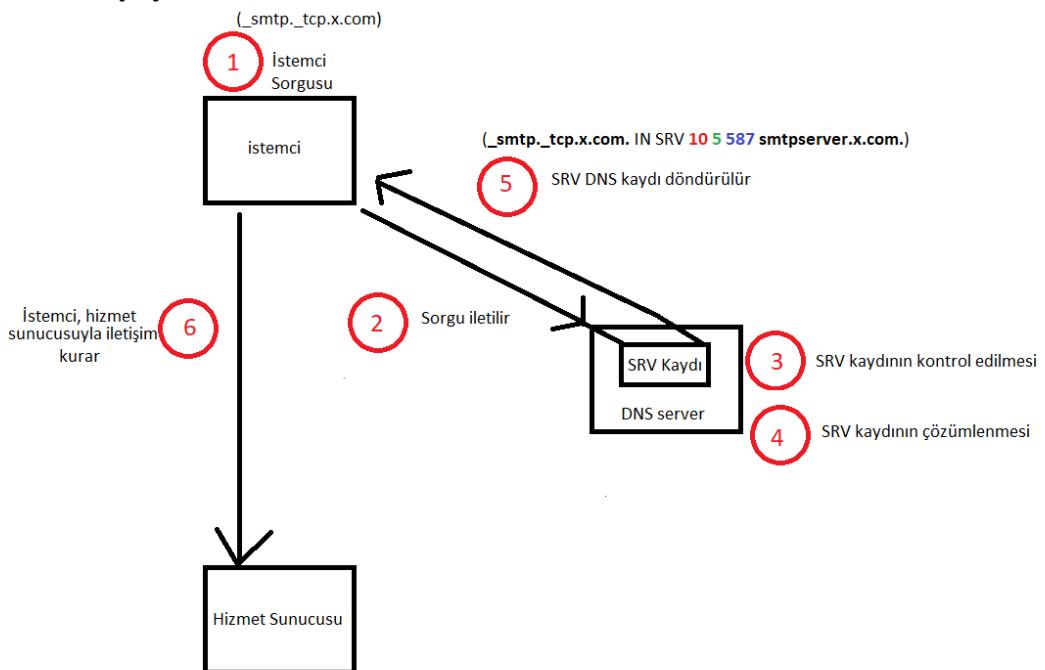
Protokol ve Port Esnekliği: SRV kayıtları, hizmetin çalıştığı protokol ve port numarasını içerir. Bu sayede, hizmetin protokol veya port numarası değiştiğinde bile istemciler hizmeti kolayca bulabilir.

Alan Adı Değişiklikleri ve Taşıma: Hizmetlerin taşınması veya alan adı değişiklikleri sırasında, SRV kayıtları güncellenerek istemcilerin yeni konuma sorunsuzca yönlendirilmesi sağlanabilir.

Karmaşık Ağ Yapıları: Büyük ve karmaşık ağ yapıları içinde, farklı yerlerde bulunan hizmet sunucularını koordine etmek ve yönlendirmek için SRV kayıtları kullanılır. Bu tür ağlarda, istemcilerin en yakın veya en uygun sunucuya yönlendirilmesi önemlidir.

Uygulama ve Hizmet Keşfi: Uygulamalar ve hizmetler, SRV kayıtları aracılığıyla otomatik olarak keşfedilebilir. İstemci uygulamalar, SRV kayıtlarını kullanarak uygun hizmet sunucusunu dinamik olarak bulabilirler.

Çalışma Prensibi şu şekildedir:



İstemci Sorusu: Bir istemci (örneğin, bir kullanıcı veya uygulama), belirli bir hizmeti kullanmak istediğiinde DNS sorusu gönderir. Bu soru, hizmetin adını (örneğin "_sip" veya "_smtp"), protokolünü (örneğin "tcp" veya "udp") ve hedef alan adını içerir. Bu örnekte _smtp._tcp.x.com SRV sorusunu belirtir.

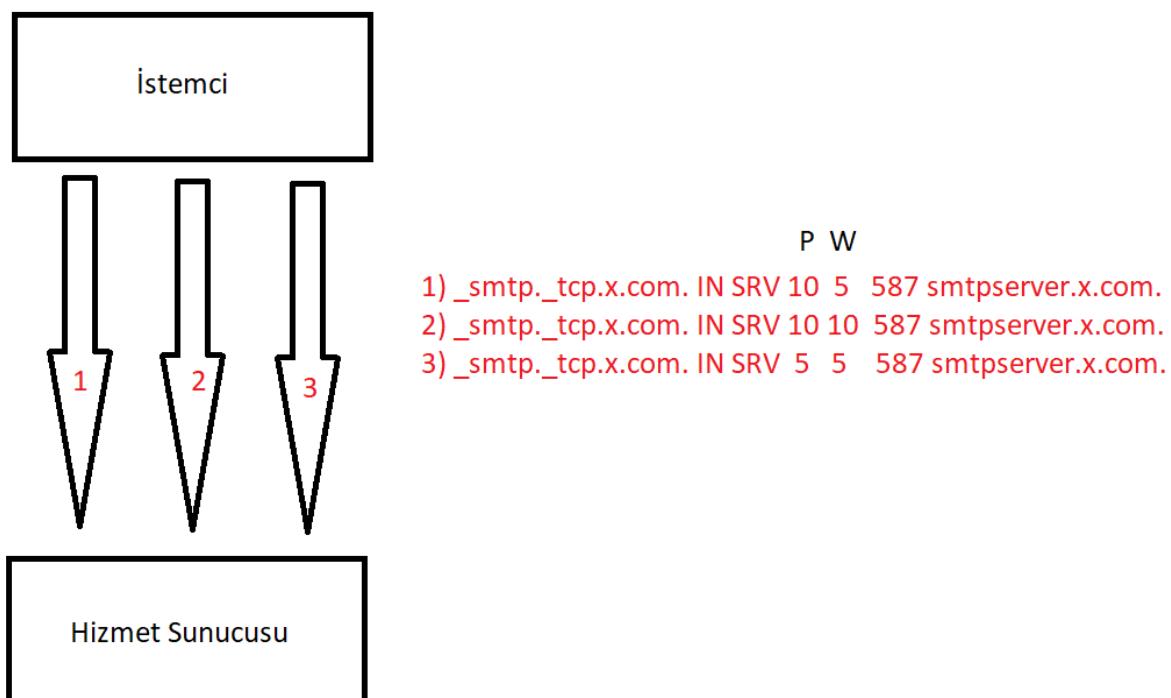
DNS Sunucusu Soruyu Alır: İstemci tarafından gönderilen DNS sorusunu, istemci tarafından belirlenen veya yapılandırılan bir DNS sunucusuna ulaşır.

SRV Kaydının Kontrol Edilmesi: DNS sunucusu, aldığı soruyu inceleyerek SRV kaydının varlığını kontrol eder. Eğer ilgili SRV kaydı varsa, devam eder.

SRV Kaydının Çözümlenmesi: DNS sunucusu, bulunan SRV kaydını çözümleyerek hizmetin erişilebilir olduğu sunucunun bilgilerini çıkarır. Bu bilgiler öncelik (priority), ağırlık (weight), port numarası ve hedef (target) bilgilerini içerir.

SRV Kaydının Döndürülmesi: SRV DNS kaydı DNS sunucudan istemciye doğru formatta geri döner. Bu format _smtp._tcp_x.com IN SRV 10 5 587 smtpserver.x.com şeklinde olabilir. _smtp hizmet adını, _tcp protokolü, x.com domain adını, 10 priority (öncelik), 5 weight (ağırlık), 587 port numarasını ve smtpserver.x.com hedefi gösterir. Ayrıca _smtp._tcp.x.com SRV sorusunu belirtir.

İstemci ve Hizmet Sunucusu Arasındaki Bağlantı: DNS sunucusu, çözümlenen SRV kaydındaki bilgilere göre istemciyi doğru hizmet sunucusuna yönlendirir. Eğer birden fazla SRV kaydı varsa (örneğin yedek sunucular), öncelik ve ağırlık bilgileri kullanılarak istemci trafiği yük dengelemesi yapabilir. İstemci, aldığı yönlendirme sayesinde hizmet sunucusuna bağlanır. SRV kaydı içinde belirtilen port numarası ve hedef bilgilerini kullanarak iletişim kurar (**_smtp._tcp_x.com IN SRV 10 5 587 smtpserver.x.com bilgileri**).



Diyelim ki 3 tane SRV DNS kaydı döndü. Bunlardan hangisinin Hizmet Sunucuna gideceğini priority (öncelik) ve weight (ağırlık) değerleri karar verir.

Öncelik Kontrolü (priority): İlk olarak, öncelik değeri daha düşük olan SRV kayıtları öncelikli olarak değerlendirilir. Düşük öncelik, daha yüksek önceliği temsil eder. Bu nedenle, önceliği daha düşük olan kayıtlar, öncelikli olarak hizmet sunucusuna yönlendirme şansı elde eder.

Ağırlık Kontrolü (weight): Eğer aynı önceliğe sahip birden fazla SRV kaydı varsa, bu kayıtların ağırlık değerleri hesaba katılır. Daha yüksek ağırlık değeri, daha fazla trafiğin o hizmet sunucusuna yönlendirilmesi anlamına gelir.

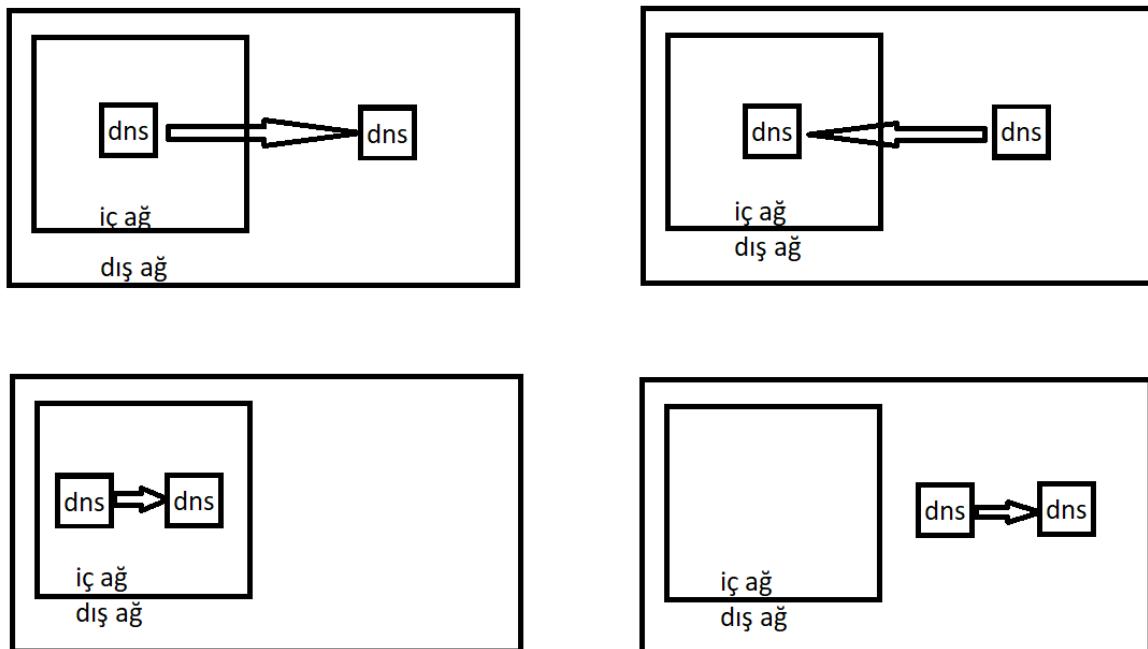
Rastgele Seçim: Eğer aynı öncelik ve ağırlığa sahip birden fazla SRV kaydı varsa, bu kayıtlar arasında rastgele bir seçim yapılır. Bu, yük dengelemesi yapılmasını sağlar.

Bu örnekte ise ilk olarak hizmet sunucusuna gidecek olan SRV Kaydı 3 numara olacak. Çünkü priority değeri en küçük. Priority değerleri eşit olan 1 ve 2 numaralı kayıttan 2 numaralı kayıt daha fazla trafik gönderecek. Çünkü ağırlığı daha fazla. Burada 2 numara 1 numaraya göre daha önce gönderilecek demiyoruz daha fazla trafik gönderilecek diyoruz. Ağırlık ve öncelik aynı işlevi görmüyor.

DNS SALDIRILARI

DNS Yönlendirmesi (DNS forwarding)

DNS Yönlendirmesi (DNS forwarding) Nedir?



DNS yönlendirme (DNS forwarding), bir DNS sunucusunun gelen sorguları başka bir DNS sunucusuna ettiği bir yapıdır. DNS yönlendirme, bir organizasyonun veya ağını içindeki DNS sunucularının dış dünyadaki DNS sunucularına sorguları yönlendirmesine olanak tanır. DNS forwarding (yönlendirme) işlemi iç DNS sunucusundan dış DNS sunucusuna olmak zorunda değildir. İç DNS'ten dış DNS'e, dış DNS'ten iç DNS'e, iç DNS'ten iç DNS'e veya dış DNS'ten dış DNS'e gerçekleşebilir. Hepsinin kullanım amacı ve yaygınlığı farklılık gösterebilir. Örneğin içten dışa işlemi iç ağın Internet'e bağlarken, dıştan dışa VPN gibi uygulamalar için çözüm olabilir.

DNS Yönlendirmesi (DNS forwarding) Neden Kullanılır?

Performans İyileştirmesi	Yük Dağıtımı	İç ve Dış Ağlar Arasında Geçiş	Filtreleme ve Güvenlik	Merkezi Kontrol	İnternet Servis Sağlayıcıları (ISP) İçin	İnternet Hizmet Sağlayıcıları (CDN) İçin
--------------------------	--------------	--------------------------------	------------------------	-----------------	--	--

Performans İyileştirmesi: DNS yönlendirme, bir organizasyonun kendi DNS sunucusunun yanıtlama yükünü azaltarak daha hızlı DNS yanıtlarımasına olanak tanır. Bu, kullanıcıların web sitelerine daha hızlı erişmelerini sağlar.

Yük Dağıtımı: DNS yönlendirme, DNS sorgularını farklı DNS sunucularına yönlendirerek yükü dağıtmak için kullanılabilir. Bu, trafik yükünü eşit şekilde paylaştırarak ağın performansını artırabilir.

İç ve Dış Ağlar Arasında Geçiş: Organizasyonlar, iç ağlarının güvenliğini korumak için dış ağdaki DNS sunucularına erişimi kısıtlayabilirler. DNS yönlendirme, iç ağdaki kullanıcıların dış dünyadaki kaynaklara erişmelerini sağlar.

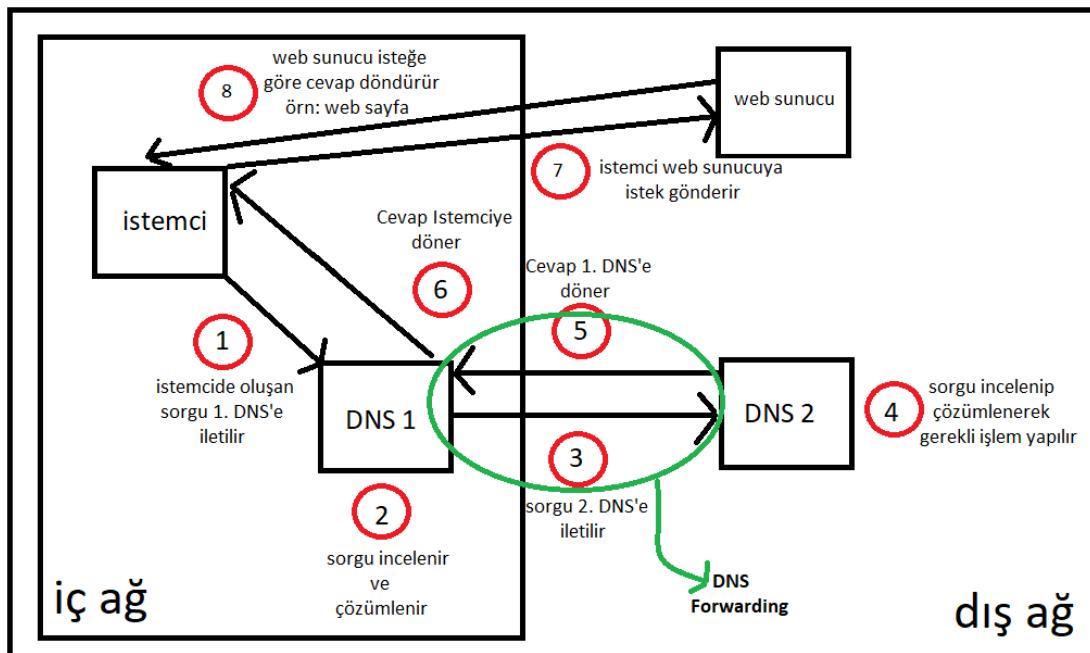
Filtreleme ve Güvenlik: DNS yönlendirme, zararlı veya istenmeyen web sitelerine erişimi engellemek için kullanılabilir. Bu, organizasyonların ağ güvenliğini artırmalarına yardımcı olur.

Merkezi Kontrol: DNS yönlendirme ayarları, organizasyonun merkezi bir konumdan yapılandırılabilir. Bu, ağ yöneticilerinin DNS trafiğini kontrol etmelerine ve yönlendirmeleri gerektiğinde değiştirmelerine olanak tanır.

Internet Servis Sağlayıcıları (ISP) İçin: Internet servis sağlayıcıları, kendi DNS sunucularını kullanarak müşterilerine DNS hizmeti sunarlar. DNS yönlendirme, ISP'lerin DNS sorgularını daha büyük ve daha güçlü DNS sunucularına yönlendirmelerine yardımcı olabilir.

Internet Hizmet Sağlayıcıları (Content Delivery Networks - CDN) İçin: CDN sağlayıcıları, içeriklerin daha hızlı teslim edilmesini sağlamak için coğrafi olarak yakın sunucuları kullanır. DNS yönlendirme, kullanıcıları en yakın CDN sunucusuna yönlendirmek için kullanılabilir.

DNS Yönlendirmesi (DNS forwarding) Nasıl Çalışır?



- 0- Baktığımız zaman istemci, DNS 1 ve web sunucu trafiği bize neyi hatırlatıyor? Benzediği mantık tamamen DNS'in çalışma mantığı. DNS çalışma mantığını bir sonraki DNS'e yani DNS 2'ye aktarma işleminde göreceğiz. Böylece DNS Forwarding işlemini anlamış olacağız.
- 1- **İlk DNS Sunucusunun Sorgu Alması:** İşlem, bir kullanıcının tarayıcısının veya cihazının bir web sitesine erişmeye çalıştığından başlıyor. Örneğin, kullanıcı "www.x.com" adresini ziyaret etmek istesin. Sorgu DNS için oluşturulur ve DNS 1'e sorgu iletılır.
- 2- **İlk DNS Sunucusunun Sorguyu İncelemesi:** Normalde DNS çalışma mantığında "DNS Root Name Server, DNS TLD Server ve DNS Authoritative Server" sunucularına sırayla gider ve IP Adres döner. Burada DNS 1 sunucusu bu işlemi yapmaz. Onun yerine gelen sorguyu inceler ve çözümler. DNS sorgu işlemi (root, tld, authoritative) dışındaki DNS 2'de gerçekleşir (Gerekirse DNS 3 adında bir DNS'de oluşturulur vs DNS 2 sorguyu bu sefer DNS 3'e yönlendirir (VPN)). DNS sunucusu, alan adını çözmek için gereken IP adresini aramaya başlar.
- 3- **DNS Yönlendirme Ayarlarının Kontrol Edilmesi ve DNS Sorgusunun Yönlendirilmesi:** İlk DNS sunucusu DNS 1, DNS yönlendirme ayarlarını kontrol eder. Eğer DNS yönlendirme etkinleştirilmişse, bu sunucu gelen sorguyu başka bir DNS sunucusuna yani DNS 2'ye iletmeye karar verir. İlk DNS sunucusu DNS 1, sorguyu belirtilen ikinci DNS sunucusuna DNS 2'ye yönlendirir. İşte biz bu işlemeye **DNS Forwarding** diyoruz.
- 4- **İkinci DNS Sunucusuna Yönlendirme:** İkinci DNS sunucusu, bu sorguyu çözmek için gereken IP adresini aramaya başlar. Burada gerekirse DNS sorgusu gerçekleştirir. Yani DNS Root Name Server, DNS TLD Server ve DNS Authoritative Server arasında sırasıyla sorgu gidip gelir ve IP Adres elde edilir.
- 5- **İkinci DNS Sunucusunun Cevap Vermesi:** İkinci DNS sunucusu yani DNS 2, çözülen IP adresini bulduğunda veya gerekiğinde sorguyu DNS 1'e yönlendirir.
- 6- **Cevap İstemciye Döner:** DNS 1'e gelen çözülmüş IP adresi (cevap) istemciye gider.
- 7- **İstemci Web Sunucuya İstek Gonderir:** İstemci DNS sunucusundan gelen cevap yani IP Adres ile web sunucusuna bir istekte bulunur. Bu örnekte [www.x.com'a](http://www.x.com) gitmek istediğini belirtmişтик.
- 8- **Web Sunucu, İstemciye Cevap Döndürür:** Web sunucu istek doğrultusunda cevap döndürür. Bu cevap HTML, JSON, XML, resim dosyası ve benzeri formattadır. Dolayısıyla kullanıcının cihazı bu IP adresini kullanarak hedef web sitesine erişir. Bu DNS sorusunda A Tipi DNS Kaydını kullandık. Yani IPv4 adresi ile ilgilendik.

Ağ Seviyesinde Yönlendirme Saldırıları

Ağ seviyesinde yönlendirme kullanarak gerçekleşen saldırılar, kötü niyetli bir saldırganın ağ trafiğini manipüle etmesini sağlar. Bu, saldırganın DNS isteklerini değiştirerek kullanıcıları yanıltmasına neden olabilir. DNS için ağ seviyesinde yönlendirme saldırılarının bazılarına şimdi özetle ilerde detaylı bakacağız.

Ağ seviyesinde yönlendirme Saldırıları Nelerdir?

Man-in-the-Middle (MitM) Saldırıları: Bu tür saldırılar, saldırganın ağ trafiğini izlemesine ve hatta değiştirmesine olanak tanır. Saldırganlar, veri paketlerini yakalar, inceleyebilir, değiştirebilir ve yeniden iletебilirler. Bu, gizli bilgilere erişim kazanmalarını sağlar.

DNS Spoofing: DNS spoofing, saldırganların DNS yanıtlarını manipüle etmelerini içerir. Saldırganlar, hedef DNS istemcilerini yanıltmak için sahte DNS yanıtları gönderirler, bu da kullanıcıları yanıltıcı web sitelerine yönlendirebilir.

BGP (Border Gateway Protocol) Saldırıları: BGP, internet üzerindeki veri trafigini yönlendiren bir protokoldür. Saldırganlar, BGP saldırıları aracılığıyla internet trafigini yönlendirmek ve hedeflenen kaynaklara erişim sağlamak amacıyla BGP güzergahlarını manipüle edebilirler.

Rerouting Saldırıları: Bu tür saldırılar, trafigi istenmeyen bir yönde yönlendirmek için kullanılır. Saldırganlar, hedef kaynakları veya hedeflere erişimi engellemek veya kullanıcıları yanıltmak amacıyla trafigi başka bir yöne yönlendirebilirler.

ARP (Address Resolution Protocol) Zehirlemesi: ARP zehirlemesi, bir ağdaki bilgisayarların IP adreslerini MAC adreslerine eşlemek için kullanılan ARP tablolarını manipüle etmek anlamına gelir. Saldırganlar, hedef bilgisayarları yanıltmak ve trafigi kendi aralarında yönlendirmek için ARP zehirlemesi kullanabilirler.

SYN Flooding: SYN flooding, bir ağ sunucusuna çok sayıda SYN isteği göndererek sunucunun kaynaklarını tüketme amacıyla taşır. Bu tür saldırılar, sunucuların erişilemez hale gelmesine neden olabilir.

IP Spoofing: IP spoofing, saldırganların sahte IP adresleri kullanarak ağ trafigini manipüle etmelerini sağlar. Bu, ağ güvenliği duvarlarını aşmak ve izini kaybettirmek amacıyla kullanılabilir.

DNS Spoofing

DNS Spoofing Nedir?

DNS spoofing, kötü niyetli bir kişinin veya grupun, DNS (Domain Name System) sistemi üzerinde sahte veya yanıltıcı bilgiler sunarak Internet trafigini manipüle etmeye çalıştığı bir saldırı türüdür. Yani amaç Internet trafigini manipüle etmektir. Birden fazla yöntemi vardır.

DNS Spoofing Yöntemleri Nelerdir?

DNS Spoofing Bazı Yöntemler					
DNS hijacking (DNS korsanlığı)	DNS Cache Redirection (DNS Cache Yönlendirme)	Cache Poisoning (Ön Bellek Zehirlenmesi)	DNS Server Spoofing	Man In The Middle	Wi-Fi Saldırıları

Belirli DNS Spoofing yöntemleri: DNS korsanlığı (DNS hijacking), DNS Cache Yönlendirme (DNS Cache Redirection), Cache Poisoning (Ön Bellek Zehirlenmesi), DNS Server Spoofing, Man In The Middle ve WiFi Saldırıları olarak gösterilebilir. Bunların hepsine ayrı ayrı değineceğiz.

DNS Spoofing Neden Kullanılır?

Phishing	Kimlik Hırsızlığı	Kötü Amaçlı Yazılım Dağıtımı	Veri Çalma	Engelleme veya Sansür	Yönlendirme	Şaka veya Kötü Niyet
----------	-------------------	------------------------------	------------	-----------------------	-------------	----------------------

DNS Spoofing saldırısının temel amacı, DNS (Domain Name System) sistemi üzerindeki zayıf noktaları kullanarak internet trafiğini manipüle etmek veya kullanıcıları yaniltmak için DNS yanıtlarını değiştirmektir. Bu saldırının bazı sebepleri bulunabilir:

Phishing: DNS Spoofing, saldırganların kullanıcıları sahte web sitelerine yönlendirerek hassas bilgileri (örneğin, kullanıcı adları, şifreler, kredi kartı bilgileri) çalmak için kullanabileceği phishing (kimlik avı) saldırının bir parçası olarak kullanılabilir.

Kimlik Hırsızlığı: Saldırganlar, DNS Spoofing'i kullanarak kullanıcıların kimlik bilgilerini ele geçirebilirler. Bu bilgiler daha sonra kimlik hırsızlığı veya dolandırıcılık amaçları için kullanılabilir.

Kötü Amaçlı Yazılım Dağıtımı: Saldırganlar, DNS Spoofing'i kullanarak kullanıcıları kötü amaçlı yazılım indirmeye veya güncellemeye zorlayabilirler. Kullanıcılar sahte güncelleme veya yazılım indirme sitelerine yönlendirilerek kötü amaçlı yazılımların kurulumuna izin verebilirler.

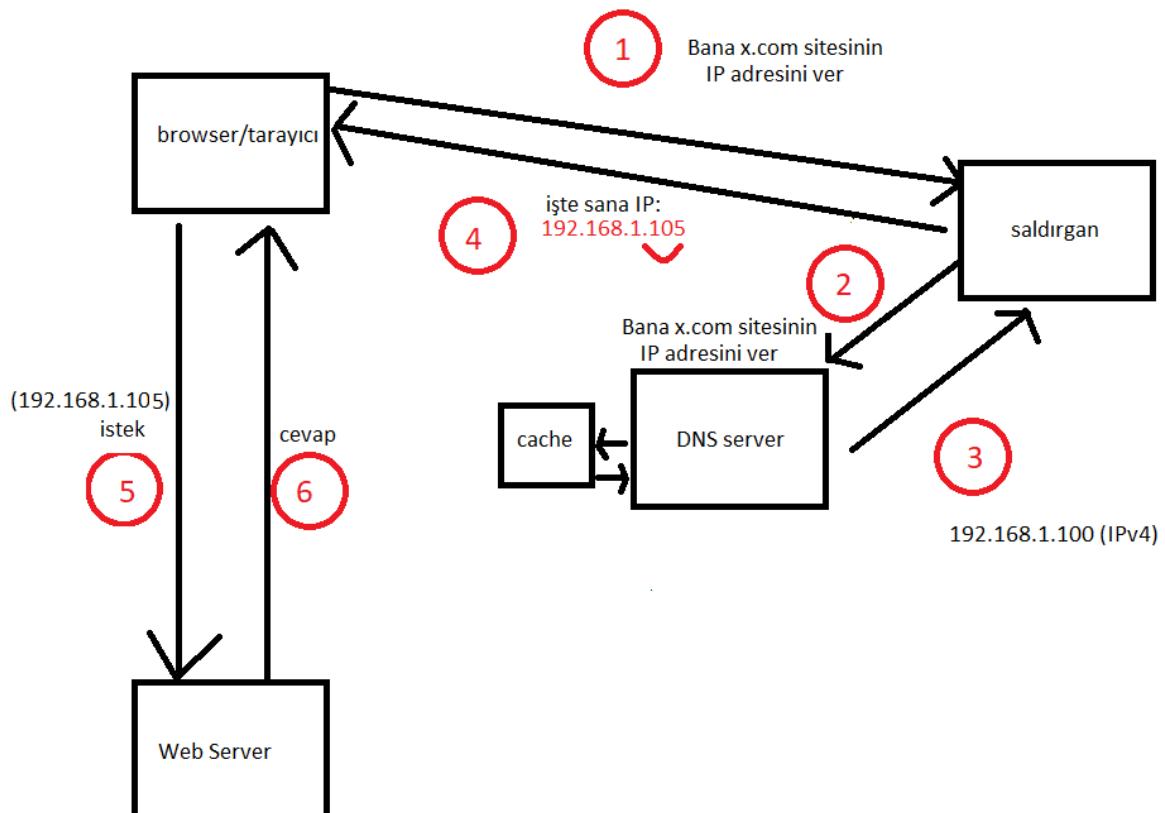
Veri Çalma: DNS Spoofing, saldırganların internet trafiğini izleyerek veya manipüle ederek hassas verileri ele geçirmesine yardımcı olabilir. Bu veriler daha sonra kötü amaçlı amaçlar için kullanılabilir.

Engelleme veya Sansür: Bazı hükümetler veya kuruluşlar, DNS Spoofing'i kullanarak belirli web sitelerine veya hizmetlere erişimi engellemek veya sansürlemek için kullanabilirler.

Yönlendirme: Saldırganlar, DNS yanıtlarını değiştirerek kullanıcıları farklı web sitelerine veya hizmetlere yönlendirebilirler. Bu, kullanıcıların istedikleri web sitesine değil, sahte veya kötü niyetli bir siteye yönlendirilmesine neden olabilir.

Şaka veya Kötü Niyet: Bazı kişiler veya gruplar, sadece eğlence veya zarar verme amacıyla DNS Spoofing'i kullanabilirler. Kullanıcıları yanıltıcı web sitelerine yönlendirerek kafa karıştırmak veya zarar vermek isteyebilirler.

DNS Spoofing Nasıl Çalışır? (Man In the Middle üzerinden)



DNS spoofing saldırısında net bir işlem vardır diyemeyiz. Birden fazla DNS Spoofing yöntemimiz var. Mesela bu örnekteki saldırırda DNS Spoofing yöntemi olan Mitm ele alalım.

- 0- Saldırgan kendini istemci ve sunucu arasına konumlandırır.
- 1- İstemci istek gönderir. İstekte bana “x.com’un IP adresini ver” der.
- 2- Saldırgan bu isteği olduğu gibi DNS Sunucuya gönderir.
- 3- Sunucu x.com’un IPv4 adresini saldırgana döndürür. (192.168.1.100)
- 4- Saldırgan IP Adreste değişiklik yapar. (192.168.1.105) Değiştirdiği IP’yi istemciye yönlendirir.
- 5- IP adresi eline ulaşan istemci isteği web sunucuya gönderir.
- 6- Bu istege karşı web sunucudan istemciye bir cevap döner ve sayfa yüklenir. Aslında İstemci 192.168.1.100’e bağlanmak istiyordu fakat saldırgan araya girerek isteği manipüle etti.
- Dolayısıyla istemci artık 192.168.1.105’e bağlandı.

İşte bu bir DNS Spoofing yöntemi idi. Yani mitm kullandık. Demek istediğim birden fazla DNS Spoofing yöntemi olduğundan, DNS Spoofing saldırısının nasıl çalıştığını tek bir şeye indirgeyemeyiz. DNS Spoofing yöntemlerinden bazıları neydi? Daha önce bahsetmiştik: DNS hijacking, DNS Cache Redirection, DNS Cache Poisoning, DNS Server Spoofing, Man In The Middle, Wi-Fi Saldırıları.

DNS korsanlığı (DNS hijacking) Nedir?

DNS Hijacking, kötü niyetli saldırganların DNS sorgularını manipüle ederek kullanıcıları yanıltmaya çalıştığı bir siber saldırı yöntemidir. Saldırganlar, DNS sorgularını ele geçirir veya değiştirir ve yanıltıcı DNS yanıtları göndererek kullanıcıları yanıltmaya çalışır. Amaçlar arasında kullanıcıların hassas bilgilerini çalmak, kimlik hırsızlığı yapmak veya kullanıcıları yaniltmak yer alabilir. DNS Hijacking, kötü amaçlı bir DNS spoofing yöntemidir.

DNS Hijacking Yöntemleri Nelerdir?

Cache Poisoning (Ön Bellek Zehirlenmesi): Bu yöntemde, saldırganlar hedef DNS sunucusunun önbelleğine yanlış DNS kayıtları eklerler veya mevcut kayıtları güncellerler. Böylece, DNS sunucusu yanıtları yanıltıcı bilgilerle doldurulur ve kullanıcılar yanlış IP adreslerine veya web sitelerine yönlendirilir.

Man-in-the-Middle (MitM) Saldırısı: Saldırganlar, kullanıcıların DNS sorgularını yakalarlar ve hedef DNS sunucusu ile kullanıcı arasında bir köprü oluşturarak DNS trafigini izler veya manipüle ederler. Bu yöntemle, saldırganlar DNS yanıtlarını değiştirerek kullanıcıları yanıltabilirler.

DNS Sunucusu Ele Geçirme: Saldırganlar hedef DNS sunucusunun kontrolünü ele geçirebilirler. Bu, saldırganların DNS kayıtlarını değiştirmelerine ve yanıt olarak yanıltıcı bilgiler göndermelerine olanak tanır. Saldırganlar, bir DNS hizmet sağlayıcısının altyapısını hedef alabilirler. Bu, büyük ölçüklü DNS hijacking saldırılara neden olabilir ve milyonlarca kullanıcıyı etkileyebilir.

DNS Sunucu Yazılımı Zayıfları: DNS sunucuları çeşitli yazılım zayıflarılarına sahip olabilir. Saldırganlar, bu zayıfları kullanarak DNS sunucularına sızabilir ve kontrolü ele geçirebilirler. Bu, DNS kayıtlarını manipüle etmek için kullanılabilir.

Kötü Amaçlı Yazılım Kullanımı: Kötü amaçlı yazılım veya kötü amaçlı kodlar, kullanıcıların DNS sorgularını değiştirmek veya DNS sunucularına sızmak için kullanılabilir. Kullanıcıların bilgisayarlarına veya ağlarına bulaşan kötü amaçlı yazılım, DNS Hijacking'e yol açabilir.

Ağ Seviyesinde Yönlendirme: Saldırganlar, ağ cihazları veya yönlendiriciler üzerinden DNS sorgularını ele geçirip yönlendirerek kullanıcıları yanıltabilirler. Bu, ağ seviyesindeki DNS trafiğini manipüle etmek için kullanılır. DNS Hijacking'de "Ağ Seviyesinde Yönlendirme" (Network-Level Redirection), genellikle iç ağlarda gerçekleşir. Çünkü bu tür saldırular, ağ altyapısı üzerinde doğrudan kontrol sağlamayı gerektirir. Yani iç ağdaki bir dns sunucusuna saldırı gerçekleştirebilir.

DNS korsanlığı (DNS hijacking) Neden Kullanılır?

Phishing Saldırıları: Saldırganlar, DNS Hijacking'i kullanarak sahte web sitelerine kullanıcıları yönlendirirler. Bu sahte siteler, kullanıcıların hassas bilgilerini (örneğin, kullanıcı adları, şifreler, kredi kartı bilgileri) çalmak amacıyla oluşturulur. Saldırganlar, kullanıcıları gerçek bir web sitesi gibi görünen sahte bir giriş sayfasına yönlendirerek bu bilgileri ele geçirebilirler.

Kimlik Hırsızlığı: DNS Hijacking'i kullanarak saldırular, kullanıcıların kimlik bilgilerini çalmayı hedefleyebilirler. Bu, kullanıcıların giriş bilgilerini ele geçirerek veya kişisel kimlik bilgilerini çalarak gerçekleştirilebilir.

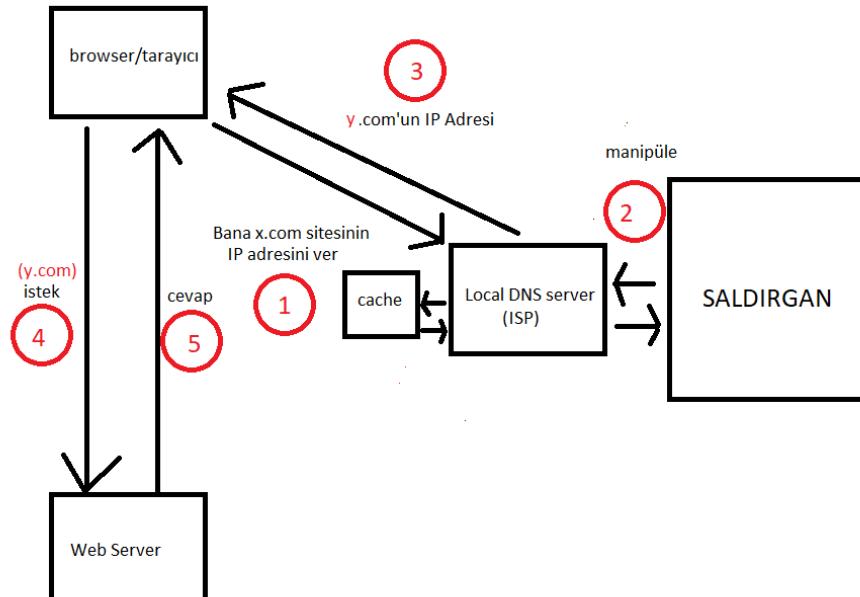
Kötüçül Yazılım Dağıtımı: Saldırganlar, DNS Hijacking'i kullanarak kullanıcıları kötüçül yazılımları indirmeye veya kötü amaçlı bağlantılar tıklamaya teşvik edebilirler. Bu, kötü niyetli yazılımları cihazlara bulaştırarak bilgisayarları ve verileri tehlikeye atabilir.

Sansür ve İnternet Kontrolü: Bazı ülkeler veya kuruluşlar, DNS Hijacking'i kullanarak belirli web sitelerine veya içeriklere erişimi engellemek veya sansürlemek amacıyla kullanabilirler. Bu, internet kullanıcılarının özgür ve açık internet erişimini kısıtlayabilir.

Veri Çalma ve İzleme: DNS Hijacking, saldıruların internet traafiğini izlemelerini ve kullanıcıların çevrimiçi davranışlarını takip etmelerini kolaylaştırabilir. Bu, kullanıcıların çevrimiçi faaliyetlerini izlemek ve veri toplamak isteyen kötü amaçlı kişiler veya kuruluşlar için bir tehlke oluşturabilir.

Web Sitesi Yönlendirmeleri: Saldırganlar, DNS Hijacking'i kullanarak kullanıcıları yanıltıcı veya kötü amaçlı web sitelerine yönlendirebilirler. Bu, kullanıcıları yanlış bilgilendirme veya kötü amaçlı içeriklere maruz bırakma amacıyla taşıyabilir.

DNS korsanlığı (DNS hijacking) Nasıl Çalışır?



- 1- **DNS Sorgusu:** Bir kullanıcı, web tarayıcısı veya uygulama aracılığıyla bir web sitesine veya hizmete erişmeye çalıştığında, cihazı, istenen alan adını IP adresine çevirmek için bir DNS sorgusu gönderir. Örneğin, kullanıcı "x.com" adresini ziyaret etmek isterse, cihaz bir DNS sorgusu yapar ve "x.com" alan adının karşılık geldiği IP adresini öğrenmeye çalışır. Kullanıcının cihazı, bu DNS sorgusunu kullanıcının genellikle internet servis sağlayıcısının (ISP) DNS sunucusuna veya yerel ağdaki bir DNS sunucusuna gönderir. DNS sunucusu, bu soruyu çözmek için işleme başlar.
- 2- **Saldırganın Müdahalesi:** DNS Hijacking burada devreye girer. Saldırganlar, kullanıcının DNS sorgusunu ele geçirir veya bu sorguya müdahale eder. Bu aşamada, saldırınların farklı yöntemleri olabilir. Bunlar:
 - Cache Poisoning (Ön Bellek Zehirlenmesi)
 - Man-in-the-Middle (MitM) Saldırısı
 - DNS Sunucusu Ele Geçirme
 - Kötü Amaçlı Yazılım Kullanma
 - DNS Sunucu Yazılımı Zafiyetleri
 - Ağ Seviyesinde Yönlendirme
 - Biz bu örnekte cache poisoning yöntemini görüyoruz. Hijacking saldırısı yöntemlerine az önce kısaca değinmiştik. İleride bu aşamalara daha detaylı bakacağız.
- 3- **Yanıltıcı Yanıt:** Saldırgan, ele geçirilen veya manipüle edilen DNS sorgusuna karşılık olarak istemciye yanıltıcı bir DNS yanıtını gönderir. Bu yanıt, hedef alan adının yanlış bir IP adresine veya sahte bir web sitesine işaret edebilir. Kullanıcı, bu yanıltıcı bilgiyi kabul eder ve sonraki adımda sahte web sitesine yönlendirilir.
- 4- **5- Kullanıcı Yanıltılması:** Kullanıcının cihazı, aldığı yanıltıcı DNS yanıtını doğru bilgi olarak kabul eder. İlk önce sahte web sitesine istek gönderilir. Bu isteği cevap döner ve sahte web site yüklenir. Sonuç olarak, kullanıcı sahte web sitesine yönlendirilir ve bu sahte siteye giriş yaparak hassas bilgilerini ifşa edebilir veya kötü amaçlı yazılım indirebilir.

Cache Poisoning (Ön Bellek Zehirlenmesi):

Ön Bellek Zehirlenmesi (Cache Poisoning), bilgisayar sistemlerinde veya ağlarda kullanılan bir güvenlik saldırısı türündür. Bu saldırısı, özellikle web tarayıcıları, sunucular veya HTTP önbellek hizmetleri gibi ön bellek mekanizmalarının güvenliğini etkileyebilir. Ön bellek, genellikle sık kullanılan verileri hızlı bir şekilde erişilebilir hale getirmek için kullanılır, böylece web siteleri veya uygulamalar daha hızlı yanıt verebilir.

Cache poisoning saldırıları, bu ön bellek mekanizmalarını yanıltmayı veya manipüle etmeyi hedefler. Bu tür saldırılar, genellikle kötü niyetli bir saldırının, kullanıcılarına yanlış veya tehlikeli veriler sunarak bir web uygulamasının güvenliğini tehditiye atmasını sağlar.

Birden çok cache poisoning (ön bellek zehirlemesi) bulunmaktadır. Bunlar:

- DNS Önbellek Zehirlemesi (cache poisoning)
- HTTP Önbellek Zehirlemesi (cache poisoning)
- Proxy Önbellek Zehirlemesi (cache poisoning)
- CDN (İçerik Dağıtım Ağı) Önbellek Zehirlemesi (cache poisoning)
- İçerik (Content) Önbellek Zehirlemesi (cache poisoning)
- Frame Zehirlemesi (poisoning)
- Tarayıcı Önbellek Zehirlemesi (cache poisoning)

- Sunucu Önbellek Zehirlemesi (cache poisoning)
- Flash Önbellek Zehirlemesi (cache poisoning)
- Web Uygulama Güvenlik Zehirlemesi (poisoning)

Bu saylıklarımız dışında daha fazla cache poisoning türü de bulunur. Ancak biz bu kursta sadece DNS Önbellek Zehirlemesi (cache poisoning) üzerinde duracağız.

DNS Cache Positioning Neden Kullanılır?

Kullanıcı Yönlendirmesi: Saldırganlar, DNS cache poisoning'i kullanarak kullanıcıları yanlış veya tehlikeli web sitelerine yönlendirebilirler. Bu, kullanıcıların hassas bilgilerini ifşa etmelerine veya dolandırıcılık faaliyetlerine maruz kalmalarına neden olabilir.

Kimlik Hırsızlığı: Saldırganlar, sahte bir web sitesi üzerinden kullanıcıların kimlik bilgilerini veya giriş bilgilerini çalmak için DNS cache poisoning'i kullanabilirler. Bu bilgiler daha sonra başka kötü amaçlı faaliyetler için kullanılabilir.

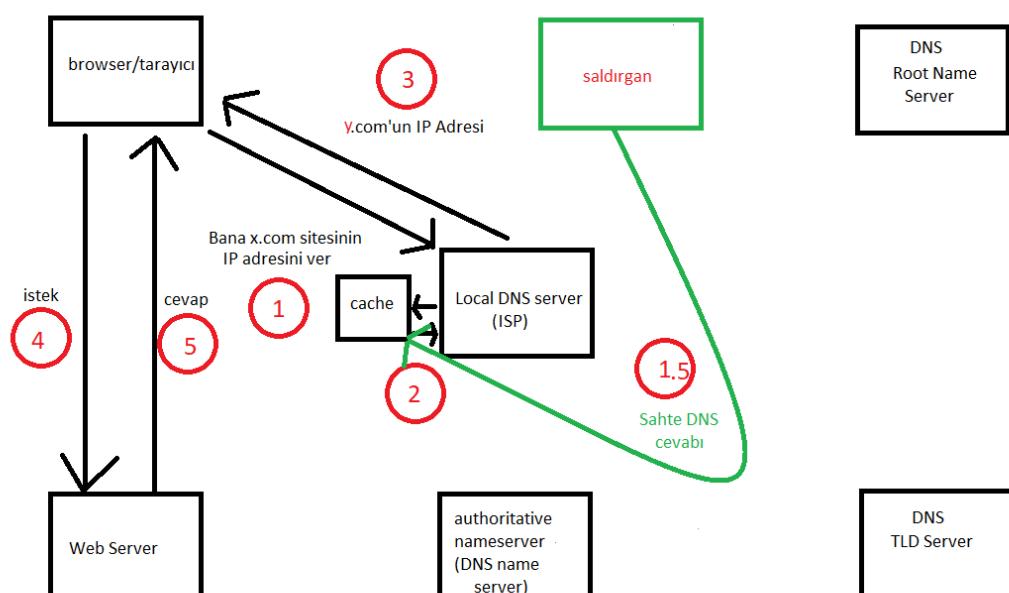
Kötü Amaçlı Yazılım Dağıtımı: DNS cache poisoning, kullanıcıları kötü amaçlı yazılım indirmeye veya kötü amaçlı web sitelerini ziyaret etmeye yönlendirmek için kullanılabilir. Bu, saldırganların hedef bilgisayarlara zararlı yazılımlar bulaştırmamasına olanak tanır.

DoS (Denial of Service - Hizmet Engelleme) Saldırıları: Saldırganlar, DNS cache poisoning'i kullanarak hedef web sitelerini veya hizmet sağlayıcılarını hedef alarak DoS (hizmet engelleme) saldırıları düzenleyebilirler. Bu, web sitelerinin kullanıcılarına erişimini kesmeye veya hizmetlerini bozmaya yönelik bir taktik olabilir. Saldırganların bu tür saldırıları başlatmalarının ana hedefi, hizmet kesintilerine neden olarak hedef kurumun veya web sitenin erişilemez hale gelmesini sağlamaktır. DDoS saldırıları, birçok endüstri dalında ciddi finansal ve itibari zararlara yol açabilir. Bu nedenle hedef DNS sunucusunun güvenliğini artırmak ve DDoS koruma önlemleri almak önemlidir.

Veri Manipülasyonu: Saldırganlar, DNS cache poisoning'i kullanarak veri manipülasyonu yapabilirler. Örneğin, e-posta iletişimini yönlendirebilirler, böylece kullanıcıların e-postalarını ele geçirebilir veya değiştirebilirler.

Amaç Dışı Yönlendirmeler: Saldırganlar, DNS cache poisoning'i kullanarak kullanıcıları zararlı içeriklere yönlendirebilirler. Bu, pornografi, terör propagandası veya diğer yasa dışı veya zararlı içeriklere yönlendirme amaçlarına hizmet edebilir.

DNS Cache positioning nasıl çalışır?



- 1- **Hedef DNS Sunucusu Belirlenir:** Saldırgan, hedef olarak bir DNS sunucusu seçer. Bu DNS sunucusu genellikle bir hizmet sağlayıcının sunucusu veya belirli bir alan adını çözmek için kullanılan bir kamu DNS sunucusu olabilir.
- 2- **Alan Adı İsmi Sorgulanır:** Saldırgan, hedef DNS sunucusunu yaniltmaya çalıştığı bir alan adı belirler. Bu alan adı, genellikle kullanıcıları yanlış bir web sitesine yönlendirmek veya kötü amaçlı bir uygulama sunmak için kullanılır.
- 3- **Sahte Cevap Hazırlanır:** Saldırgan, sahte bir DNS cevabı hazırlar. Bu cevap, belirli bir alan adının IP adresini belirlemek için kullanılan kayıtları içerir.
- 4- **Sahte Cevap Gönderilir:** Saldırgan, sahte cevabı hedef DNS sunucusuna gönderir. Bu işlem sırasında, saldırganın gönderdiği sahte cevap, hedef DNS sunucusunun önbelleğine eklenmeye çalışılır.
- 5- **Önbellek Zehirlenir:** Hedef DNS sunucusu, saldırganın gönderdiği sahte cevabı kabul eder ve önbelleğe ekler. Bu ekleme işlemi Ettercap gibi araçlarla yapılabilir. Bu, daha sonraki kullanıcı sorguları için sahte IP adresine yönlendirmeleri anlamına gelir.
- 6- **Kullanıcılar Yanıltılır:** Hedef DNS sunucusunun önbelleği zehirlendiğinde, kullanıcılar yanlış veya tehlikeli bir web sitesine yönlendirilir. Örneğin, kullanıcılar banka web sitesi gibi güvenilir bir kaynağa erişmeye çalışıklarında, sahte bir web sitesine yönlendirilirler ve hassas bilgilerini ifşa edebilirler.
- 7- **Saldırganın Amaçları Gerçekleştirilir:** Salırgan, kullanıcıları yanıltmak, hassas bilgilere erişmek veya kötü amaçlı yazılım dağıtmak gibi amaçlarını gerçekleştirmeye çalışır.

Görüldüğü gibi kullanıcı x.com'a gitmek isterken birden y.com'a yönlendirildi. Bu saldırıyı önlemek için bazı seçeneklerimiz olabilir. Bunlar:

- **DNSSEC (DNS Security Extensions) kullanımı:** DNSSEC, DNS sorgularının güvenliğini sağlamak ve sahte DNS cevaplarına karşı koruma sağlamak için kullanılır.
- **Güncellemelerin izlenmesi:** DNS sunucuları, güncellenen kayıtları kabul etmeden önce kaynakları dikkatle izlerler.
- **Randomized Query ID kullanımı:** Rastgele sorgu kimlikleri, saldırganların tahmin edilebilir sorgu kimliklerini kullanmasını zorlaştırır.
- **İzleme ve günlük kayıtları:** DNS sunucuları, şüpheli etkinlikleri izlemek ve tespit etmek için günlük kayıtlarını kullanabilirler.

Man-in-the-Middle (MitM) Saldırısı

Man in the Middle Nedir?

"Man-in-the-Middle" (MitM), Türkçe'de "Ara Katman Saldırısı" veya "Ortadaki Adam Saldırısı" olarak adlandırılan bir siber güvenlik terimidir. Bu tür bir saldırı, bir iletişim kanalı üzerindeki verilerin izlenmesi, değiştirilmesi veya çalınması amacıyla gerçekleştirilir. Man-in-the-Middle saldırıları, genellikle şifrelenmemiş veya zayıf şifrelenmiş iletişim kanallarını hedefler.

"man-in-the-middle atak" saldırının var olan iletişim veya veri transferini kesip yeni bir iletişim hattı oluşturan bir nevi kulak misafiri atak şeklidir. Salırgan ortadaki adam olduğu zaman her iki katılımcı gibi davranır. Bu sayede salırgan her iki taraftan gelen bilgi ve veriyi tutar. Basit olarak atak, iki kişinin görüşme yaptığı ve salırganın araya girerek her iki tarafın sözlerini aldığı telefon görüşmesi olarak düşünülebilir.

Man in the Middle Neden Kullanılır?

Hassas Bilgi Çalma: Saldırganlar, MitM saldırularıyla kullanıcıların kişisel bilgilerini, kimlik bilgilerini, finansal bilgilerini ve şifrelerini çalmak için hassas verilere erişebilirler. Bu bilgiler daha sonra dolandırıcılık, kimlik hırsızlığı veya finansal suçlar için kullanılabilir.

Veri Manipülasyonu: MitM saldıruları, iletişim kanalı üzerindeki verileri değiştirme veya manipüle etme amacıyla kullanılabilir. Saldırganlar, verileri istedikleri gibi değiştirerek yanıltıcı bilgiler veya komutlar gönderebilirler.

Kimlik Hırsızlığı: Saldırganlar MitM saldırularıyla kullanıcıların oturum açma bilgilerini çalabilirler. Bu bilgilerle hesaplara erişebilirler ve kullanıcıların kimliğini çalabilirler.

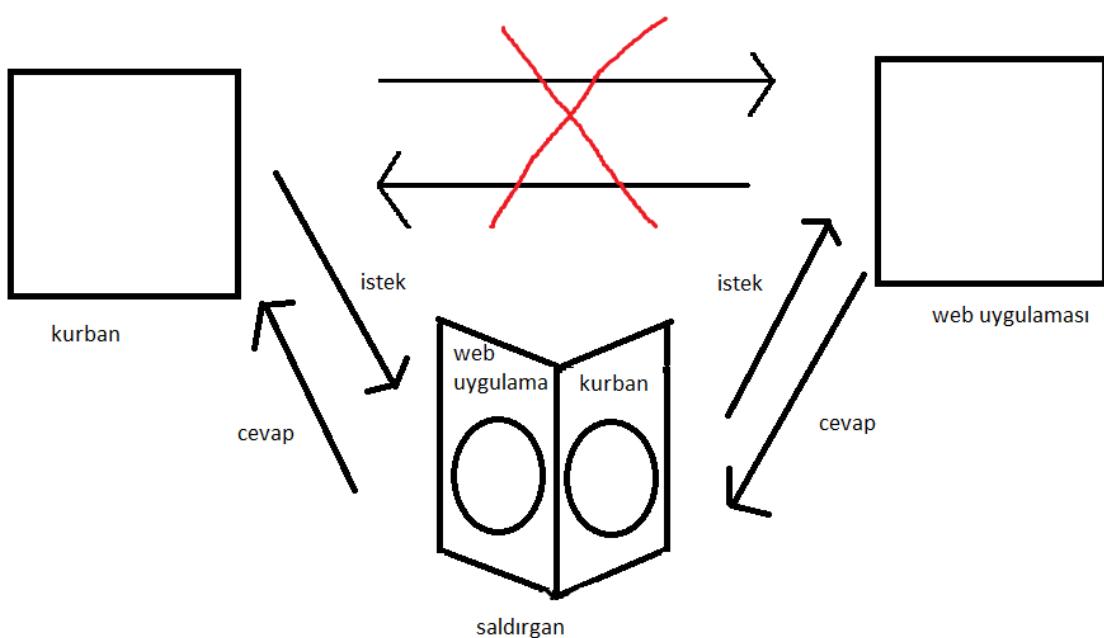
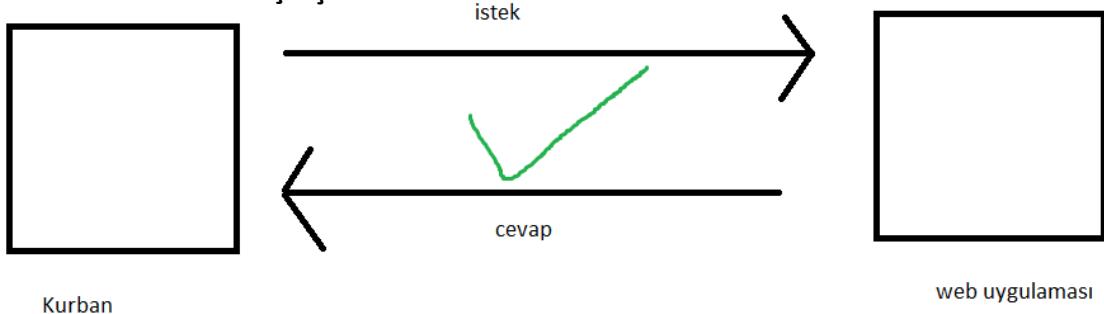
Şifre Kırmá: Saldırganlar, şifrelenmiş veriyi ele geçirerek daha sonra şifreyi kırmak veya çözmek için saldırı yapabilirler. Bu, özellikle zayıf şifrelemeye sahip bağlantılarda etkilidir.

Gözetleme ve İzleme: MitM saldıruları, iletişimi izlemek ve izlemek amacıyla kullanılabilir. Bu, kullanıcıların özel veya duyarlı konuşmalarını veya verilerini izlemek için casusluk amacıyla gerçekleştirilebilir.

Bilgi Toplama: Saldırganlar, MitM saldırularıyla bilgi toplama amacı güdebilirler. Örneğin, bir şirketin iç ağına sızarak hassas kurumsal bilgilere erişebilirler.

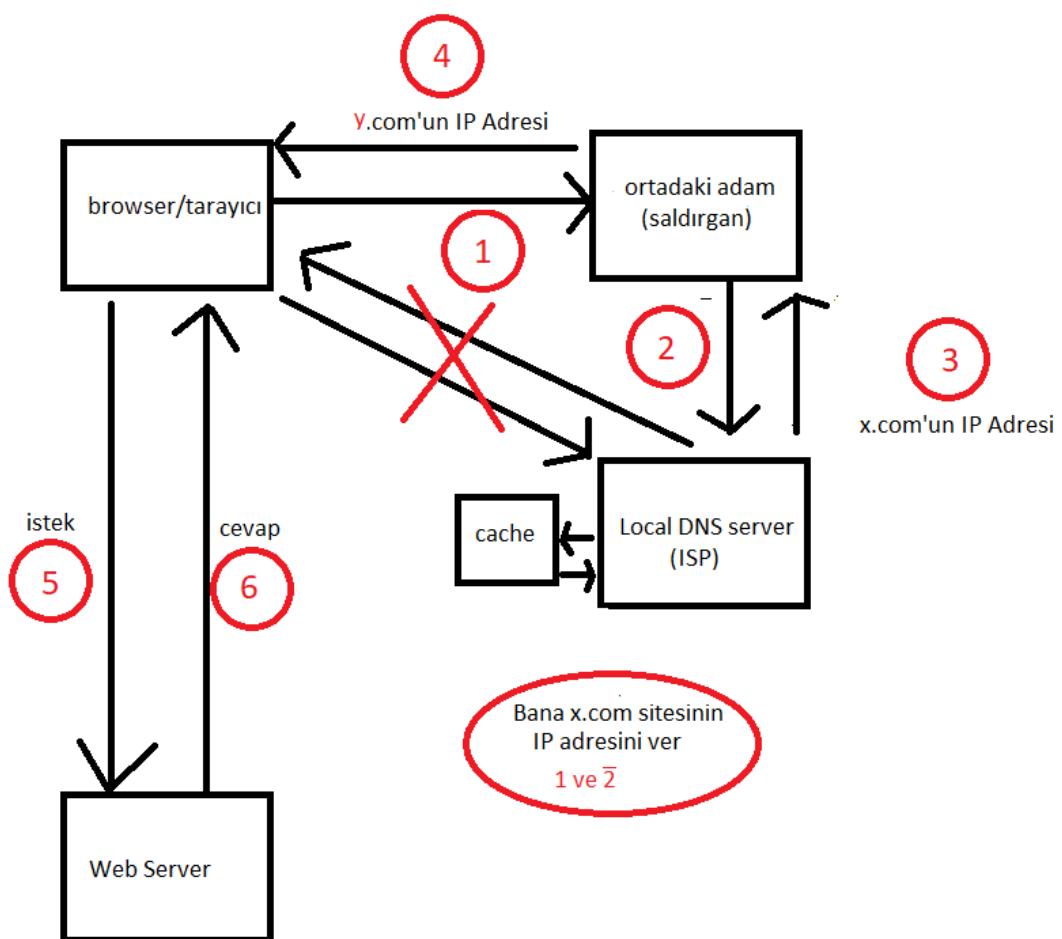
Güvenlik Zayıflıklarını Sömürme: MitM saldıruları, bir ağ veya sistemdeki güvenlik zayıflıklarını tespit etmek ve sövmek için kullanılabilir. Bu, daha büyük bir siber saldırının bir parçası olarak gerçekleştirilebilir.

Man in the Middle Nasıl Çalışır?



- 1- Normalde istemci (burada kurban) sunucuya (web uygulamasına) istekte bulunur. Bunu daha önce webin, DNS ile birlikte çalışma mantığında görmüştük. İstek gider bunun sonucunda cevap döner.
- 2- Fakat diğer görsele baktığımız zaman istemci (kurban) ile sunucu (web uygulaması) arasında iletişim artık olması gereği gibi değil.
- 3- Araya bir saldırgan giriyor ve saldırgan kurbana web uygulaması olarak gözükmek, web uygulamasına ise kurban olarak gözükmektedir. Yani sahte bir kimliğe sahip olup nasıl görünmek istiyorsa öyle görünüyor.
- 4- Bu sayede kurban ve web uygulaması saldırgana güveniyor.
- 5- Kurban isteği web uygulamasına göndermek istiyor – ki burada saldırgan web uygulaması olarak gözükmektedir – ve isteği gönderiyor. İstek saldırgan üzerinden web uygulamasına gönderiliyor.
- 6- Cevap iste web uygulamasından kurbana gönderiliyor – ki burada yine saldırgan manipüle ediyor. Yani web uygulaması, saldırganı kurban olarak görüyor – ve cevap saldırgan üzerinden kurbana gönderiliyor.
- 7- İşte bu noktada mesaj şifresiz gönderilirse yani HTTPS yerine HTTP protokolü tercih edilirse, mesaj okunup anlaşılabilir. Yani mesajın şifrelenmesi lazım.

DNS Sorgusu MitM ile Nasıl Sömürülür?



- 0- **Ağa Sızma:** Saldırgan, hedef ağına veya kullanıcının ağına sızar. Kendisini ortadaki adam olarak konumlandırır.
- 1- **2-İsteğin DNS Sorgusu İçin Gönderilmesi:** DNS soru isteği 1. ve 2. adımda DNS sunucuya gönderilir.
- 3- **DNS Sorgularının Yakalanması:** Saldırgan, ağıda dolaşan DNS sorgularını izler veya ele geçirir. Kullanıcının DNS sunucusuna gönderdiği sorguları veya DNS sunucunun kullanıcının sorgularına verdiği yanıtları yakalar.
- 4- **DNS Yanıtlarının Değiştirilmesi:** Saldırgan, kullanıcının DNS sorgularına verilen yanıtları manipüle eder. Örneğin, bir kullanıcı bir web sitesine erişmeye çalıştığında, saldırıcı bu web sitesinin IP adresini sahte bir IP adresiyle değiştirebilir.
- 5- **6-Kullanıcının Yanıltılması:** Kullanıcı, sahte IP adresi tarafından işaret edilen yanıltıcı bir web sitesine yönlendirilir yani istek gönderilir. Bu yanıltıcı site, orijinal web sitesinin bir kopyası gibi görünebilir. Web site cevap döndürerek web sayfayı istemciye sunar.
- 6- .
- 7- **Bilgi Çalma veya Casusluk:** Kullanıcı, yanıltıcı web sitesine girdiğinde, saldırıcı bu site üzerinden kullanıcının verilerini toplayabilir veya iletişimini izleyebilir. Bu, kimlik hırsızlığı, şifre çalma veya hassas bilgilerin ele geçirilmesi amacıyla kullanılabilir.

DNS sorgularının MitM saldırılarına karşı korunmanın bazı yolları şunlar olabilir:

- **Güvenilir DNS Sunucularını Kullanmak:** Bilgisayarlarınız ve cihazlarınız için güvenilir DNS sunucularını kullanarak DNS sorgularınıza güvendiğiniz kaynaklardan yanıtlar alabilirsiniz.
- **DNSSEC Kullanmak:** DNSSEC (DNS Security Extensions) gibi güvenlik önlemleriyle DNS sorgularınızı şifreleyerek ve doğrulayarak güvence altına alabilirsiniz.
- **HTTPS Kullanmak:** İnternet tarayıcınızda veya uygulamalarınızda HTTPS kullanarak iletişimini şifreleyebilirsiniz.
- **Güvenlik Duvarları ve Güvenlik Yazılımları:** Bilgisayarlarınıza veya ağına güçlü bir güvenlik duvarı ve güvenlik yazılımı eklemek, kötü amaçlı DNS sorgularını engelleyebilir.

DNS Sunucusu Ele Geçirme

DNS Sunucusu Ele Geçirme Nedir?

DNS (Domain Name System) sunucusu ele geçirme, bir saldırının hedeflenen bir DNS sunucusunu fiziksel olarak kontrol etme veya manipüle etme işlemidir. DNS sunucularının ele geçirilmesi, çeşitli zararlı amaçlar için kullanılabilir, örneğin:

- **Yönlendirme saldırısı:** Saldırganlar, ele geçirdikleri DNS sunucusunu kullanarak trafiği istedikleri yerlere yönlendirebilirler. Bu, kullanıcıların güvenilir gibi görünen sitelere yönlendirilerek hassas bilgilerin çalınmasını sağlayabilir.
- **Man-in-the-Middle saldırısı:** Saldırganlar, ele geçirilmiş DNS sunucusunu kullanarak iletişimleri izleyebilir ve hedeflenen bilgileri çababilirler. Bu, kullanıcıların iletişimlerinin gizliliğini tehlikeye atar.
- **Kimlik avı (phishing):** Ele geçirilen DNS sunucuları, kullanıcıları kötü amaçlı sitelere yönlendirerek kimlik bilgilerini veya hassas verilerini çalmak için kullanılabilir.

DNS sunucusu ele geçirme genellikle şunlar gibi güvenlik açılarının veya zayıf konfigürasyonların kullanılması yoluyla gerçekleştirilir:

- Zayıf şifreler veya kimlik doğrulama yöntemleri
- Yazılım güvenlik açıkları
- Sosyal mühendislik saldırıları
- Fiziksel erişim elde etme

DNS Sunucusu Ele Geçirme Neden Kullanılır?

Yönlendirme Saldırıları: Ele geçirilen DNS sunucusu, kullanıcıları güvenilir gibi görünen web sitelerine yönlendirilebilir. Bu, kullanıcıların bankacılık bilgileri, kimlik bilgileri veya diğer hassas bilgilerini girmeleri için sahte web sitelerine yönlendirilerek bu bilgilerin çalınmasına neden olabilir.

Man-in-the-Middle Saldırıları: DNS sunucusu ele geçirilirse, saldırganlar iletişimleri izleyebilir veya manipüle edebilir. Bu, kullanıcıların gizli bilgilerini çalmak veya iletişimlerini dinlemek için kullanılabilir.

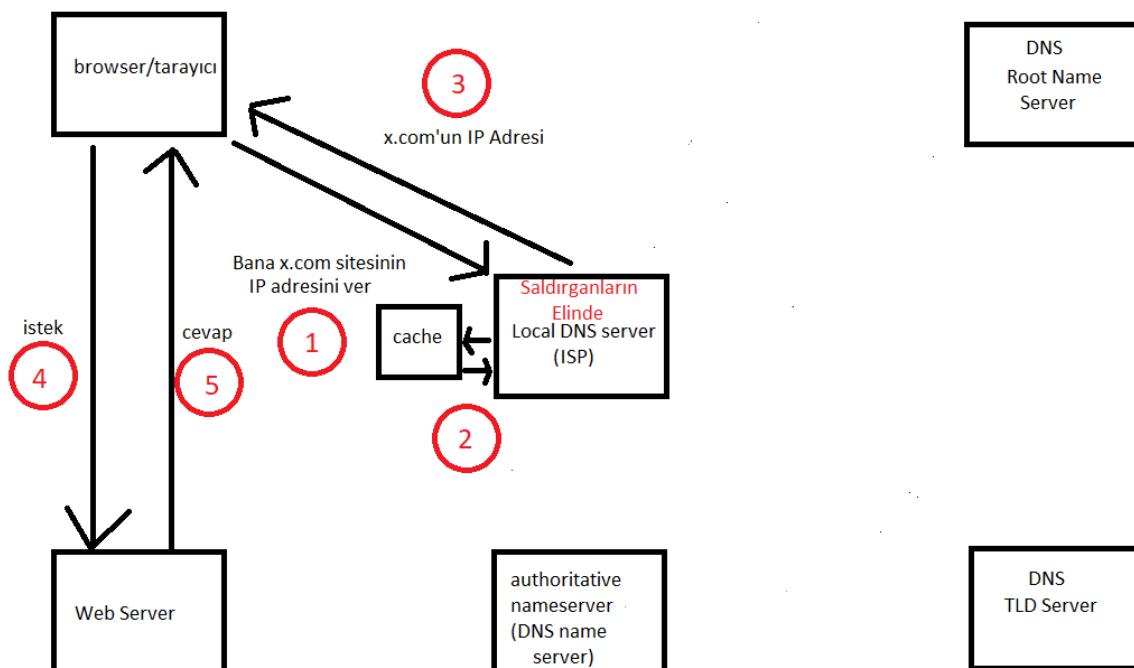
Kimlik Avı (Phishing): Ele geçirilen DNS sunucuları, sahte web sitelerine kullanıcıları yönlendirmek için kullanılabilir. Saldırganlar, kullanıcıları banka, e-posta veya diğer çevrimiçi hesaplarının kimlik bilgilerini çalmak için bu sahte siteleri kullanabilir.

Servis Kesme Saldırıları: Ele geçirilen DNS sunucuları, hedeflenen web sitelerini veya çevrimiçi hizmetleri kullanılamaz hale getirmek için kullanılabilir. Bu, hedefin çevrimdışı kalmasına veya hizmetlerinin kesilmesine neden olabilir.

Bilgi Toplama: DNS sunucusu ele geçirme, ağda geçen trafiği izlemek ve analiz etmek amacıyla kullanılabilir. Bu, hedef ağını yapısını ve etkinliklerini anlamak için casusluk amacıyla yapılabilir.

Devlet veya Kurumsal İstihbarat: Bazı devlet istihbarat ajansları veya büyük kurumsal organizasyonlar, DNS sunucularını ele geçirerek bilgi toplama veya siber casusluk faaliyetlerini yürütmek için kullanabilirler.

DNS Ele Geçirerek Saldırı Nasıl İşler?



- 1- DNS sorgulama işlemi normal bir şekilde işlemeye devam etmek ister. DNS sunucusu artık saldırganların fiziksel kontrolünde olduğu için sorgu saldırganların isteği doğrultusunda gerçekleşecektir. Yani sorgu sonucu saldırganlar dilediği saldıruları gerçekleştirebilecek (yanlış ip adres döndürmek gibi).

DNS Sunucu Yazılımı Zafiyetleri

DNS Sunucusu Yazılım Zafiyeti Saldırısı Nedir?

DNS sunucusu yazılım zafiyeti saldırısı, DNS (Domain Name System) sunucularının üzerinde bulunan yazılımın güvenlik açılarını hedef alan bir tür siber saldırıdır. Eğer DNS sunucu yazılımı güvenlik açıları içeriyorsa, kötü niyetli kişiler bu açıları kullanarak çeşitli zararlı eylemlerde bulunabilirler.

DNS sunucusundan kaynaklı yazılım zafiyeti saldıruları şu şekillerde gerçekleştirilebilir:

- **Yükseltme Saldıruları (Elevation of Privilege):** Saldırganlar, DNS sunucusu yazılımindaki bir güvenlik açığını kullanarak daha yüksek bir kullanıcı ayrıcalığına (privilege) sahip olabilirler. Bu, sunucunun kontrolünü ele geçirmelerine ve daha fazla zararlı işlem yapmalarına olanak tanır.
- **Düzelte (Patch) Yokluğundan Yararlanma:** Eğer DNS sunucusu yazılımının güncellemeleri yapılmamışsa ve güvenlik açıları giderilmemişse, saldırganlar bu açıları kullanabilirler. Bu nedenle yazılımın düzenli olarak güncellenmesi önemlidir.
- **Zehirli DNS Kaydı Ekleme:** Saldırganlar, DNS sunucusu yazılımının açılarını kullanarak sahte DNS kayıtları ekleyebilirler. Bu, kullanıcıları yanlış IP adreslerine yönlendirerek phishing (kimlik avı), kötü amaçlı yazılımların yayılması veya diğer zararlı faaliyetler için kullanılabilir.
- **DNS Saldıruları:** DNS sunucusu yazılımindaki zafiyetler, DNS sorgularının bozulmasına veya engellenmesine neden olabilir. Bu, hizmet kesintilerine yol açabilir.

Bu tür saldırılara karşı korunmak için aşağıdaki önlemleri almak önemlidir:

- DNS sunucusu yazılımının güncel ve en son sürümde olduğundan emin olun.
- Güçlü şifreler ve kimlik doğrulama kullanarak sunucu erişimini koruyun.
- Güvenlik yazılımları ve güvenlik duvarları kullanarak DNS sunucunuza koruyun.
- Güvenlik güncellemelerini ve yamalarını düzenli olarak uygulayın.
- DNS trafigini izlemek ve anomalilikleri tespit etmek için güvenlik izleme araçlarını kullanın.

DNS Sunucusundan Kaynaklı Yazılım Zafiyeti Saldıruları Neden Kullanılır?

Kullanıcıları Yanıltma (Phishing): Saldırganlar, DNS sunucusundaki zafiyetleri kullanarak sahte web sitelerine yönlendirebilirler. Bu, kullanıcıların kişisel bilgilerini veya kimlik bilgilerini çalmak için phishing saldırularına olanak tanır.

Kötü Amaçlı Yazılımların Yayılması: DNS sunucu yazılımı zafiyetleri, kötü amaçlı yazılımların hızlıca yayılmasını sağlayabilir. Saldırganlar, kullanıcıların bilgisayarlarına zararlı yazılım indirmeleri için sahte güncellemeler veya sahte yazılım indirme bağlantıları sunabilirler.

Veri Hırsızlığı: Saldırganlar, DNS sunucusundaki zafiyetleri kullanarak hassas verilere erişebilirler. Bu, ticari sırlar, müşteri verileri, finansal bilgiler veya diğer hassas bilgilere erişim sağlayabilir.

Hizmet Kesintileri: DNS sunucusuna yapılan saldırılar, hizmet kesintilerine yol açabilir. Bu, bir şirketin veya kurumun çevrimiçi hizmetlerinin geçici olarak kullanılamaz hale gelmesine neden olabilir, bu da finansal kayıplara ve itibar kaybına yol açabilir.

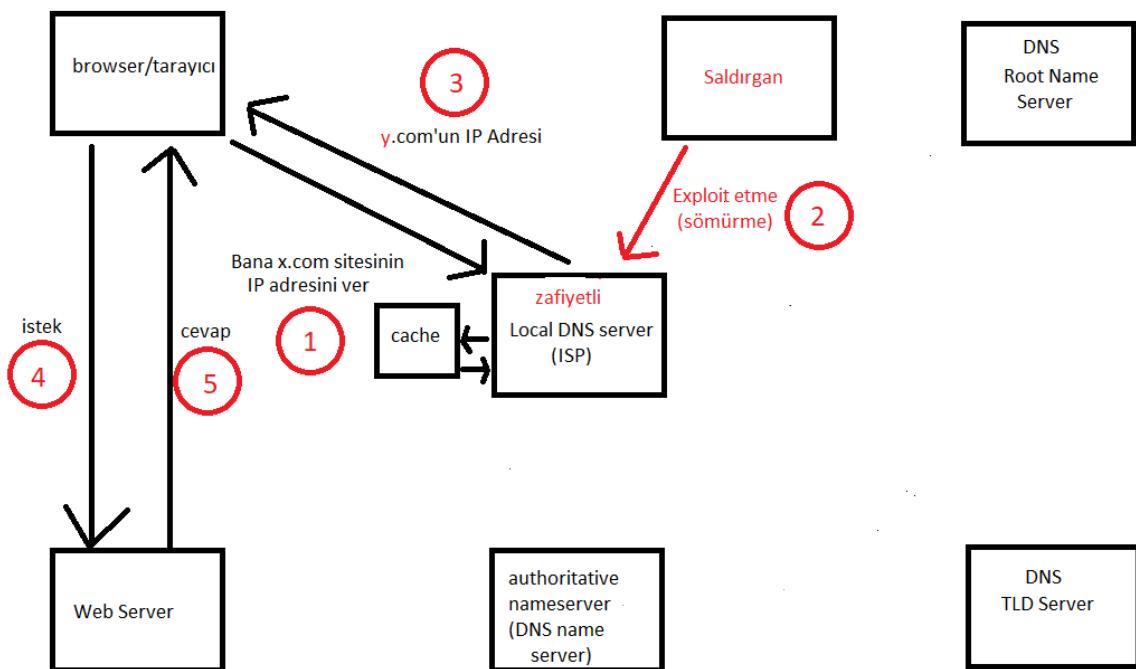
Dağıtık Hizmet Engelleme Saldırıları (DDoS): Saldırganlar, DNS sunucusundan kaynaklanan zayıflıkları kullanarak büyük ölçekli DDoS saldırıları başlatırlar. Bu, ağları, sunucuları ve hizmetleri aşırı yükleyerek kullanılamaz hale getirme amacını taşırlar. Saldırganların bu tür saldırıları başlatmalarının ana hedefi, hizmet kesintilerine neden olarak hedef kurumun veya web sitenin erişilemez hale gelmesini sağlamaktır. DDoS saldırıları, birçok endüstri dalında ciddi finansal ve itibari zararlara yol açabilir. Bu nedenle hedef DNS sunucusunun güvenliğini artırmak ve DDoS koruma önlemleri almak önemlidir.

Veri Manipülasyonu: DNS sunucularındaki zayıflıklar, veri manipülasyonuna olanak tanır.

Saldırganlar, DNS kayıtlarını değiştirerek kullanıcıları yanlış web sitelerine yönlendirebilir ve bu yolla bilgi çalabilirler.

Casusluk ve İzleme: Saldırganlar, DNS sunucusundan kaynaklı zayıflıkları kullanarak ağ trafigini izleyebilir ve casusluk yapabilirler. Bu, hassas verilerin çalınması ve kullanıcıların çevrimiçi etkinliklerinin izlenmesi amacıyla kullanılabilir.

DNS Sunucusu Yazılım Zayıflığı Saldırısı Nasıl İşler?



- 1- DNS sorgulama işlemi normal bir şekilde işlemeye devam etmek ister. DNS sunucusunda bulunan zayıfından dolayı kontrol saldırısına geçecektir. Dolayısıyla sorgu saldırısının isteği doğrultusunda gerçekleşecektir. Yani sorgu sonucu saldırılardan dilediği saldırıları gerçekleştirebilecek (yanlış ip adresi döndürmek gibi).

DNS Sunucusuna Kötü Amaçlı Yazılım Yükleme

DNS Sunucusuna Kötü Amaçlı Yazılım Yükleme Nedir?

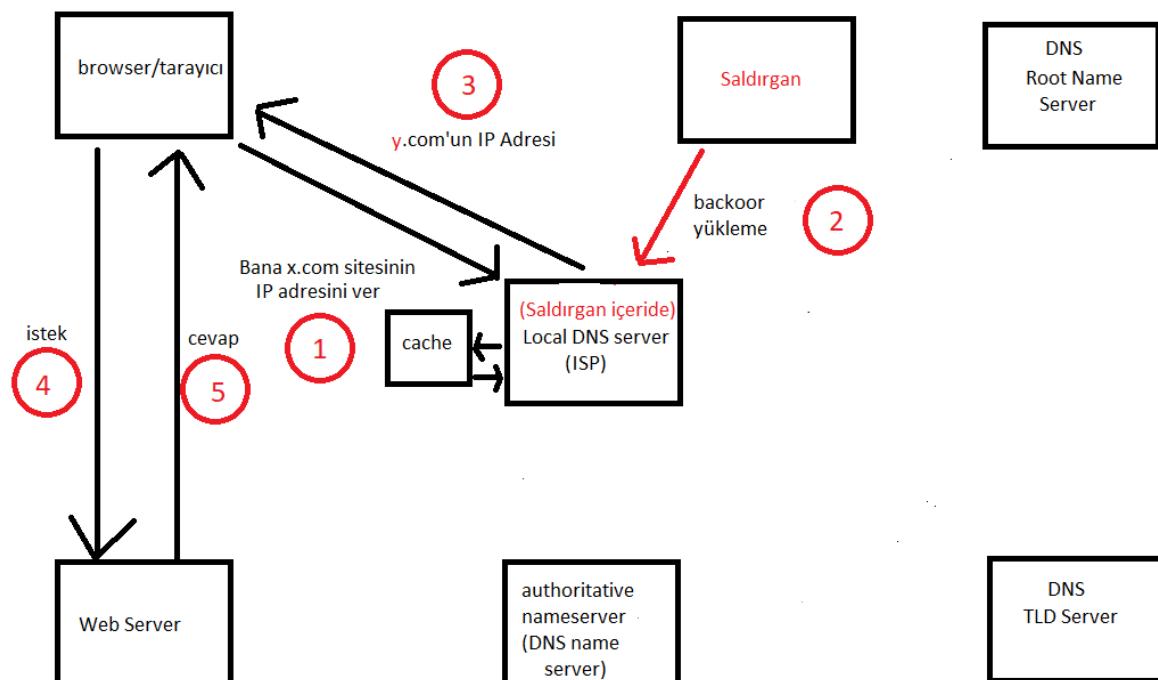
DNS sunucusuna kötü amaçlı yazılım yükleme, bilgisayar korsanlarının veya kötü niyetli kişilerin hedeflenen bir DNS (Domain Name System) sunucusuna zararlı yazılım eklemek amacıyla gerçekleştirdikleri bir siber saldırıdır.

DNS sunucusuna kötü amaçlı yazılım yükleme, bilgisayar güvenliği ve internet altyapısının bütünlüğü için ciddi bir tehdit oluşturur. Bu tür saldırıları önlemek için güçlü güvenlik önlemleri ve güncel yazılım kullanımı önemlidir. DNS sunucularınızı korumak için ayrıca güvenlik duvarları, güvenlik yazılımı, DNSSEC (DNS Security Extensions) gibi güvenlik önlemleri ve güncellemeleri uygulamak gereklidir.

DNS Sunucusuna Kötü Amaçlı Yazılım Yükleme Neden Yapılır?

- Veri Hırsızlığı:** Kötü niyetli kişiler, DNS sunucularına zararlı yazılım ekleyerek, kullanıcıların internet trafigini yönlendirme veya izleme yeteneği elde edebilirler. Bu sayede kullanıcıların kişisel bilgilerini veya hassas verilerini çalabilirler.
- DDoS Saldırıları:** Kötü niyetli kişiler, DNS sunucularına kötü amaçlı yazılım yükleyerek, büyük ölçekli DDoS (Distributed Denial of Service) saldırıları düzenleyebilirler. Bu, hedeflenen DNS sunucusunu devre dışı bırakabilir ve internet hizmetlerinin kesilmesine neden olabilir.
- Kimlik Avı (Phishing):** DNS sunucularına sızan kötü niyetli kişiler, hedef kullanıcılarla sahte web siteleri aracılığıyla kimlik avı (phishing) saldırıları düzenleyebilirler. Kullanıcıları yanıltarak kişisel bilgilerini çalmak amacıyla sahte web sitelerine yönlendirebilirler.
- Zararlı Yazılım Dağıtımu:** Kötü amaçlı yazılım geliştiricileri, DNS sunucularını kullanarak zararlı yazılım dağıtabilirler. Bu, kullanıcıların bilgisayarlarına kötü amaçlı yazılımların bulaşmasına yol açabilir.

DNS Sunucusuna Kötü Amaçlı Yazılım Yükleme Saldırısı Neden Kullanılır?



- 1- DNS sorgulama işlemi normal bir şekilde işlemeye devam etmek ister. DNS sunucusuna Backdoor gibi zararlı yazılım yüklenir. Böylece kontrol saldırgana geçer. Dolayısıyla soru saldırganların isteği doğrultusunda gerçekleşecektir. Yani soru sonucu saldırganlar dilediği saldıruları gerçekleştirebilecek (yanlış ip adres döndürmek gibi).

DNS Flooding

DNS Flooding Nedir?

DNS flood, genellikle DNS sunucularına aşırı miktarda DNS isteği veya soru gönderilmesiyle gerçekleştirilen bir saldırı türüdür. Bu isteklerin yoğunluğu nedeniyle DNS sunucusu üzerindeki kaynaklar tükenir ve sunucu hizmet veremez hale gelir. Bu tür saldırular, DNS sunucularının işlevsiz hale gelmesi veya hizmet kesintilerine neden olma amacını taşırlar.

DNS Flooding Neden Kullanılır?

Hizmet Kesintileri: DNS Flood saldıruları, hedef DNS sunucularının yanıt verme yeteneklerini aşırı yüklemeyi amaçlar. Bu, hedef organizasyonun web siteleri, e-posta sunucuları ve diğer internet hizmetlerine erişimi kesintiye uğratır. Saldıruları düzenleyenler, hizmet kesintileri nedeniyle rakiplerini zayıflatma veya hedeflerine zarar verme amacıyla taşıyabilirler.

Veri Çalma: Bazı DNS Flood saldıruları, ağ trafigini karıştırmak veya hedef DNS sunucusunun güvenlik açıklarını sövmürmek amacıyla kullanılabilir. Bu, saldırganların hassas verilere erişim sağlamak veya veri çalmak istedikleri durumlarda kullanılabilir.

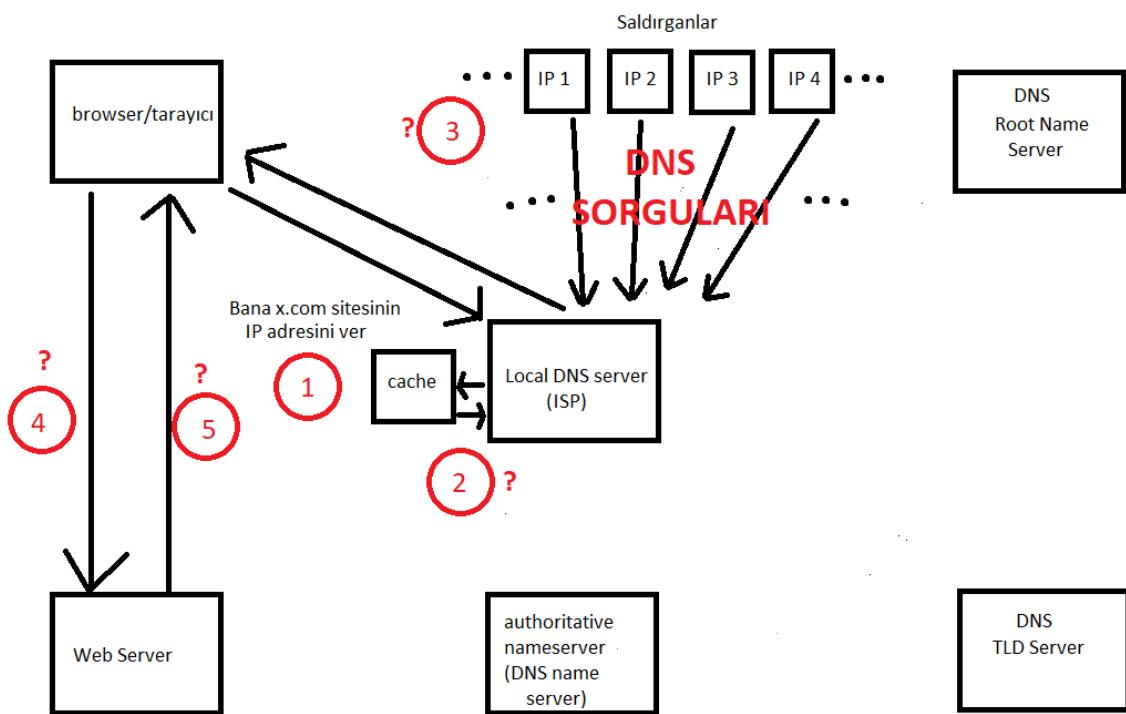
Distraksiyon: DNS Flood saldıruları, bir organizasyonun siber güvenlik ekibini meşgul etmek ve dikkatlerini başka bir yerden uzaklaştmak amacıyla kullanılabilir. Bu sırada gerçek saldırının başka bir yönde gerçekleştirilmesi planlanabilir.

Siber Güvenlik Erken Uyarı Sistemi Yok Etme: DNS Flood, saldırının başlangıcında erken uyarı sistemlerini veya güvenlik önlemlerini devre dışı bırakabilir. Bu, organizasyonun siber saldırının hedefini daha geç fark etmesine neden olabilir. (Siber Güvenlik Erken Uyarı Sistemleri: Bu tür sistemler, bilgisayar ağlarını izler ve kötü amaçlı yazılımlar, siber saldırular veya siber tehditlerin belirtilerini algılar. Erken uyarı sistemleri, bu tehditleri tanımlayarak bilgisayar sistemlerinin ve ağların güvenliğini artırır.)

Siber Savaş ve Rakip Saldıruları: DNS Flood, siber savaşların bir parçası olarak kullanılabilir veya rakip organizasyonları zayıflatmak için kullanılabilir.

Hacker Eğlencesi: Bazı siber saldırganlar, DNS Flood gibi saldıruları sadece eğlence veya boş zamanlarında zarar verme amacıyla gerçekleştirebilirler.

DNS Flooding Nasıl Çalışır?



- 1- Tarayıcı, Web siteye gidebilmek için bir DNS Sorgusu başlatır.
- 2- Fakat Saldırganlar tarafından aynı DNS sunucusuna DNS Flooding işlemi gerçekleştirilir. Saldırganlar, hedef DNS sunucusuna yani (ortak DNS sunucusuna) büyük miktarda sahte DNS isteği veya sorgu gönderirler. Bu sorgular, genellikle çok sayıda farklı IP adresinden gelir, bu da izini sürmeyi veya tespit etmeyi zorlaştırabilir. Hedef DNS sunucusu, bu yoğun isteklerle başa çıkmakta zorlanır. Sunucusu, bu istekleri işlemek için kaynaklarını tüketir ve işlem gücü sınırlarına ulaşır. Bu, DNS sunucusunun normal DNS isteklerine yanıt verme yeteneğini azaltır.
- 3- Dolayısıyla 2. ve 3. Adım sonuçsuz kalır. Çünkü DNS artık cevap veremeyecek hale geldi.
- 4- 2 ve 3. Adım sonuçsuz kalınca IP Adres tarayıcıya dönemmeyecek. Dolayısıyla 4 ve 5. adımlar da gerçekleşmeyecek. Çünkü IP adres olmadığından web sunucuya istek gönderilmeyecek ve bu doğrultuda cevap dönmeyecek.
- 5- Sonuç olarak DNS sunucusu işlevsiz hale gelir ve böylece hizmet kesintisi olur. Kullanıcılar gerçekleştirilmesi gereken işlemlerden mahrum kalır.

DNS Amplification

DNS Amplification Saldırısı Nedir?

DNS Amplifikasyon Saldırısı (DNS Amplification Attack), bir tür siber saldırıdır ve saldırganların DNS (Domain Name System) sunucularını kötüye kullanarak büyük miktarda trafik üretmeyi amaçladığı bir türdür. Bu tür saldırılar, DNS sunucularının büyük veri paketleriyle yanıt vermesini istismar eder.

DNS Amplifikasyon Saldırısı, hedeflenen sunucuya büyük miktarda trafik göndermek amacıyla taşır. Bu trafik, hedefin ağ kaynaklarını tüketerek hizmetlerini aksatma veya hedefe zarar verme amacını güder.

DNS flooding ve DNS amplification saldırısı benzer özelliklere sahiptir. Her iki saldırı türü de hedef DNS sunucusunu etkisiz hale getirme veya hizmet kesintilerine neden olma amacı taşıır, ancak farklı mekanizmaları ve teknikleri kullanırlar. DNS Amplification Saldırısı, yanıtların amplifikasyonuyla büyük miktarda trafik oluştururken, DNS Flooding Saldırısı sorguların yoğunluğuyla kaynakları tüketir. Güvenlik önlemleri alarak ve DNS sunucularını güvence altına alarak bu tür saldırılara karşı koruma sağlamak önemlidir.

DNS Amplification Saldırısı Neden Kullanılır?

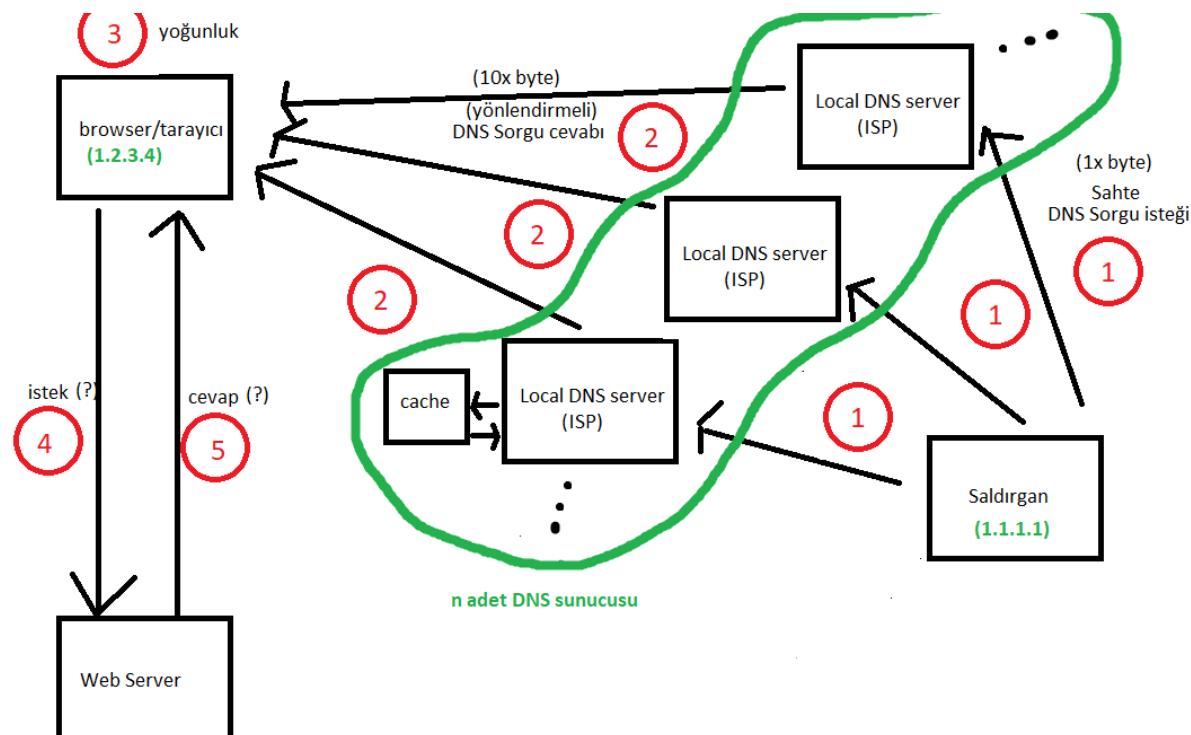
Hizmet Kesintileri: Saldırının en yaygın amacı, hedef DNS sunucusunu etkisiz hale getirmek ve hizmetlerini kesintiye uğratmaktır. DNS sunucuları, büyük miktarda amplifikasyon etkisiyle gelen trafik nedeniyle yavaşlayabilir veya çökebilir, bu da hizmetlere erişimde sorunlara neden olabilir.

Ağ Kaynaklarını Tüketme: DNS Amplifikasyon saldırısı, hedef organizasyonun ağ kaynaklarını tüketmeyi amaçlar. Bu, hedefin diğer ağ hizmetlerine erişimini engelleyebilir.

Distraksiyon: Saldırı, hedef organizasyonun siber güvenlik ekibini meşgul etmek ve dikkatlerini başka bir yönde uzaklaştmak amacıyla kullanılabilir. Bu sırada gerçek saldırının başka bir yönde gerçekleştirilemesi planlanabilir.

Siber Savaş ve Rakip Saldırıları: DNS Amplifikasyon Saldırıları, siber savaşların bir parçası olarak kullanılabilir veya rakip organizasyonları zayıflatmak için kullanılabilir.

DNS Amplification Saldırısı Nasıl Çalışır?



- 1- Saldırgan sahte DNS sorguları oluşturarak bunları n adet DNS sunucusuya gönderir. Sorgunun 1x byte kadar olduğunu düşünelim. Fakat sorgu içerisinde kaynak IP Adresi olarak kendi IP'sini kullanmaz (1.1.1.1). Onun yerine kurbanın yani tarayıcının IP adresini kullanır (1.2.3.4).

- 2- Böylece 10x byte'lık bir cevap dönebilir. Yani hedef DNS sunucusu, gelen sahte sorguya yanıt verirken büyük ve genişletilmiş bir yanıt gönderir. Bu yanıtın boyutu, orijinal sorgunun boyutundan çok daha büyük olabilir. Yanıtların bu kadar büyük olmasının nedeni, DNS sunucularının DNS kayıtlarının listelerini içeren verileri içermeleri ve bu verileri sorgulara dahil etmeleridir. DNS yanıtı, DNS Sorgusundaki kaynak IP adresine döner. Kaynak IP adresimiz 1.2.3.4 olduğundan cevaplar bu adres'e dönecektir.
- 3- 1.2.3.4 adresi tarayıcı olduğundan cevaplar buraya döner. Böylece tarayıcı tarafında yoğunluk oluşur.
- 4- 5- Bundan sonra tarayıcı tarafında yoğunluktan dolayı bir işlem gerçekleşmez.

DNS kullanılarak DRDoS

DoS/DDoS nedir?

DDoS/DoS Nedir?

Denial of service yani DoS saldırısında, saldırgan hedef sunucuya veya cihaza oldukça fazla bir trafik yollayan siber saldırı çeşididir. DoS atağı, web siteyi kullanılamaz hale getiren online ataktır. İşin özyü bu saldırında web site sunucusuna kapasitesinden fazla trafik gönderilir. Saldırı tek bir cihazdan yapılır.

DDoS (distributed denial of service) atağında ise birden fazla cihaz kullanılarak aynı saldırısı gerçekleştir. Birden fazla cihaz için genelde botnet kullanılır.

Botnet Nedir?

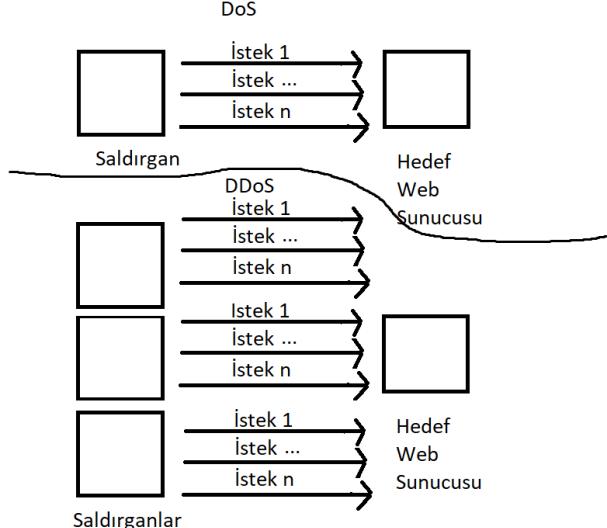
Botnet, bir grup internete bağlı cihazdır. Her biri bir veya birden fazla bot çalıştırır. DDoS saldıruları için kullanılabilir.

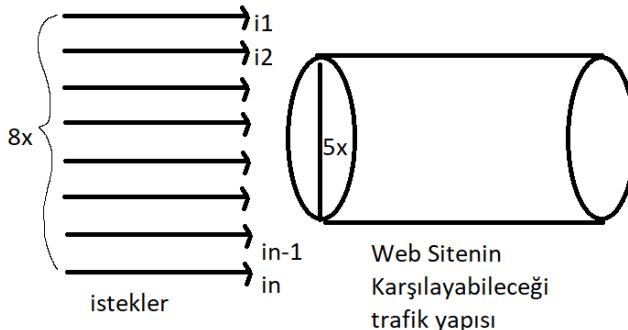


DDoS/DoS'a Neden Başvurulur?

Saldırganlar bir web siteyi etkisiz hale getirmek için DoS/DDoS saldıruları gerçekleştirebilirler. Bununla birlikte beyaz şapkalı hackerlar hedef sistemlerin DoS/DDoS saldırularına uygun olup olmadıklarını keşfetmek için ve bunun sonucunda planlama yapmak için bu saldıruları kullanabilir. Bir nevi gerçek saldıruları simüle ettikten sonra, belirli noktaları geliştirmek için kullanılır.

DDoS/DoS Nasıl Çalışır?





Dos/DDoS Nasıl Çalışır

- 0- **DoS:** 1 saldırıcı ve 1 kurbandan oluşur. Arada Zombie cihazlar veya birden çok saldırıcı bulunmaz. Saldırıcı doğrudan hedefe yönelir. Saldırıcı n adet isteği hedef web sunucusuna gönderir ve sunucuya kitlenmeye çalışır.
- 1- **DDoS:** x adet saldırıcı ve 1 kurbandan oluşur. Saldırıcıların veya zombie cihazların her birinden n adet trafik hedef web sunucusuna gönderilir. Bu noktada x.n adet istek hedef web sunucusuna yönlenmiş olur ve sunucuya kitlenmeye çalışır (x adet cihaz sayısından, n adet istek sayısından [her birinden n adet çıkacak diye bir kural yok. N-1 olur, n+1 olur.]).
- 2- **DDoS/DoS:** Bu noktada hedefin kitlenmesi nasıl olur diye soracak olursak, cevap yetersiz bandwith (bant genişliği) olacaktır. Mesela 8x adet isteği hedef sunucusuna yollayalım. Hedef sunucunun da 5x bandwith'e sahip olduğunu düşünelim. Sonuçta 3x istek artacak ve sıkışmaya neden olacak. Böylece web site yanıt veremez hale gelecek. İşte DoS ve DDoS saldırılarının mantığı budur.

DDoS/DoS Nasıl İndirgenir, Karşısında Ne Yapılır?

- 1- Network trafiğinizin kapasitesini (bandwidth) öğrenin ve ona göre hareket edin.
- 2- Bir Dos/DDoS saldırısı planı geliştirin. Organizasyonun başına böyle bir şey gelince ne yapılacağını planlayın.
- 3- Network bandwidth'ınızı genişletin. (Altyapınızı güçlendirin)
- 4- Anti-DDoS donanımı ve yazılımı edinin.
- 5- Trafiğinizi izleyerek DDoS atağının olup olmadığını anlayabilirisiniz ve duruma göre hareket edebilirisiniz.

İpucu

Dos saldırısında başarılı olabilmek için kendi ağımızın güçlü olması gereklidir (yüksek bandwith-bant genişliği). Bunun sebebi bir yere saldırırken paketler ilk önce bizin ağımızdan çıkacaktır. Dolayısıyla ağımızın iyi olması gereklidir. Ayrıca hedef sistemi alt etmek için yine yüksek seviyeli trafik gereklidir. Bu da sizin bant genişliğinizle doğru orantılıdır.

Bunun bir başka opsiyonu DDoS saldırısı yapmaktır. Böylece bir botnete sahip olabilir ve her bir cihazdan az da olsa bir trafik göndererek hedef sunucuya sıkıştırabiliriz (karınca kararınca + bir elin nesi var iki elin sesi var).

Bütün bunların laboratuvar ortamında gerçekleştirilmesi gereklidir. Veya hedef için yazılı bir izin belgenizin olması gereklidir. Bunun sebebi her bölümde olduğu gibi Dos/DDoS bölümünde de herhangi bir sisteme saldırı gerçekleştirirseniz hapis cezası veya para cezasıyla karşılaşabilirsiniz. LABORUTVAR ORTAMLARINDA LEGAL KALIN!

DRDoS Nedir?

Distributed Reflection Denial of Service (DRDoS), bir bilgisayar ağı saldırısı türüdür. Bu saldırının gerçek hedefine saldırmak yerine, genellikle yüksek bant genişliği sağlayan açık sunucuları veya cihazları kullanarak bir hizmeti hedef alır. Bu tip saldırılara genellikle DDoS (Distributed Denial of Service) saldıruları kategorisine girer, ancak DRDoS'teki belirgin bir özellik, saldırının başlatılan saldırının izini saklamak için yansıtma (reflection) tekniklerini kullanmasıdır.

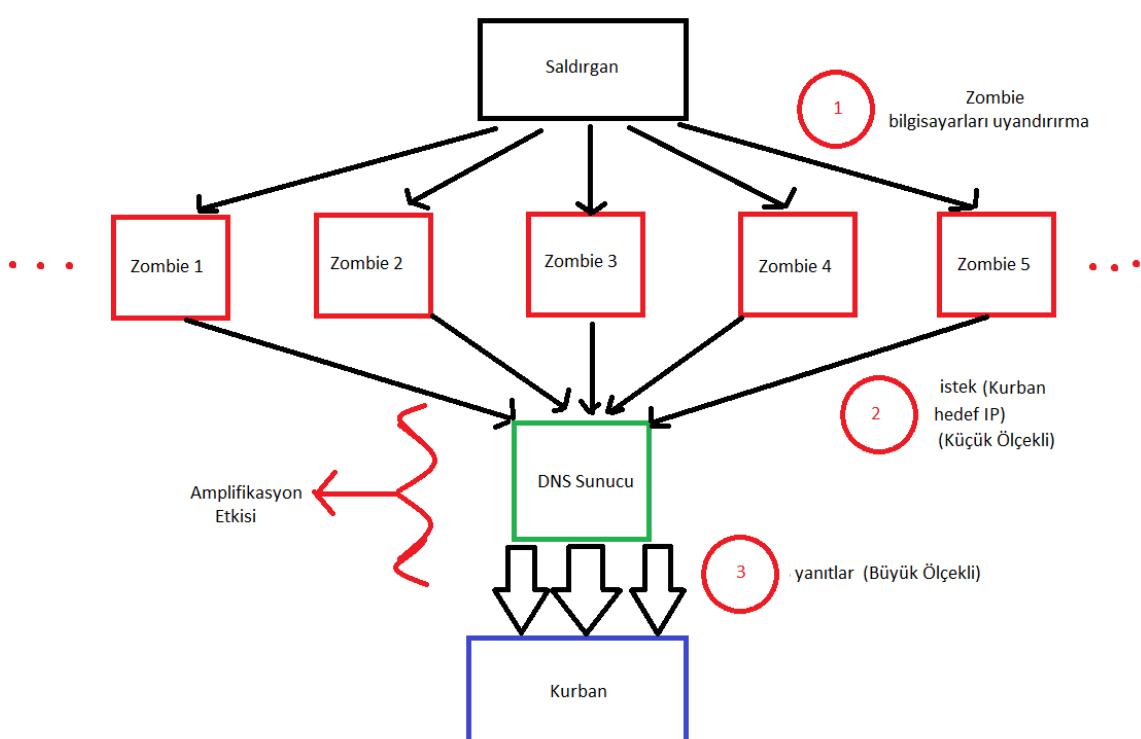
DRDoS saldırısı genellikle 4 ana bileşen içerir:

- 3- **Saldırgan**: Saldırgan, gerçek hedefine saldırmak yerine genellikle açık sunucuları veya cihazları hedef almak için yansıtma tekniklerini kullanır.
- 4- **Zombie**: Saldırgan daha önce zararlı yazılımlarla ele geçirdiği Zombie bilgisayarları kullanır.
- 5- **Yansıtma Sunucuları veya Cihazları**: Saldırgan, açık ve genellikle savunmasız sunucuları veya cihazları kullanarak bir dizi yansıtma saldırısını başlatır. Bu sunucular, saldırının trafiğini hedefe yansıtarak saldırının kaynağını gizlemeye yardımcı olur.
- 6- **Gerçek Hedef**: Salgın, gerçek hedefe yönlendirilen büyük miktarda trafik oluşturarak hedefin hizmetlerini aşırı yükleyerek bir hizmet reddi (denial of service) durumu yaratır.

DRDoS (Distributed Reflection Denial of Service) Neden Kullanılır?

Bu tür saldırılarda siber saldırı olarak kullanımının yanında, yansıtma sunucularını kullanarak saldırının izini saklamak ve saldırının dağıtmak için tasarlanmıştır. Yansıtma sunucuları genellikle geniş bant genişliğine sahip olduğundan ve kaynak IP adresini gizleyebildiğinden, saldırının tespit edilmesi daha zor olabilir.

Yansıtma Olarak DNS Kullanılarak DRDoS Nasıl Çalışır?



- 1- Saldırgan, her bir zombie bilgisayarları uyandırarak aktif rolde görev olmasını sağlar.
- 2- Zombie bilgisayarların her biri Kurbanın IP adresini kaynak olarak kullanarak DNS sorgusu gönderir. Bu sorgu istekleri küçük boyutludur. Örneğin her biri mx tipinde 1x byte boyutunda gönderdiğini düşünelim ve 5 zombie bilgisayarı var. Toplamda 5x byte gönderilecek. Tabii
- 3- Amplifikasyon etkisi (DNS Amplifikasyon Saldırısında degenmişlik) ile her biri 10 kat arttığını varsayılm. Yani 5 tane 10x byte'tan 50x byte boyutunda cevap donecek. Tabii ki cevap zombie bilgisayarlara değil Kurbana yönlendirilecek. Bunu nasıl sağlamıştık? Sorgu isteği gönderilirken Kurbanın IP adresini kaynak olarak kullandığımız için bu işlem gerçekleşmiş.
- 4- DRDoS'taki bu sorgu işlemi yansıtma olarak DNS sunucu kullanıldığı zaman gerçekleşir. NTP veya SNMP gibi farklı sunucular yansıtma olarak kullanılırsa işlem başka şekilde gerçekleşir. Biz DNS başlığı altında olduğumuz için işlemleri bu şekilde anlamaya çalışıyoruz.

DRDoS ve DNS Amplification Saldırılarının Farkı Nedir?

Her iki saldırı türü de, hedef sistem veya ağır kaynaklarını tüketerek hizmet reddi durumu oluşturma amacını taşır. Ancak, kullanılan teknikler ve hedeflenen servislerdeki farklılıklar, bu saldırı türlerini birbirinden ayırrı.

Teknik olarak, DRDoS saldırıları genellikle farklı protokollerdeki yansıtma sunucularının açıklarını kullanarak bir amplifikasyon etkisi oluşturmayı hedefler. DNS Amplification saldırıları ise genellikle DNS sunucularının özelliği olan büyük boyutlu yanıtları kullanarak amplifikasyon etkisi elde etmeye dayanır. Yani DRDoS'da DNS; NTP veya SNMP gibi sunucuları yansıtma olarak kullanılırken, DNS Amplification'da doğrudan DNS üzerinden saldırı gerçekleştirilir.

DNS Zone Transfer Saldırısı

DNS Zone Nedir?

DNS bölgesi (DNS zone), belirli bir alan adının veya bir alan adı grubunun yönetildiği ve bu alan adlarının IP adresleriyle ilişkilendirildiği bir alandır. DNS, domain isimlerini IP adreslerine çevirmek için kullanılan bir sistemdir. DNS zone, bu isim-IP çevirme işlemini gerçekleştirmek için gerekli kayıtları içerir.

Start of Authority (SOA) DNS kaydı, DNS zone'un temelini oluşturan ve bölgenin yönetimini sağlayan en önemli kaytlardan biridir. SOA kaydı, bir DNS bölgesinin başlangıcını belirtir ve bölge yönetiminin nasıl yapılandırılacağını tanımlar. Bu nedenle, SOA kaydı DNS zone ile sıkı bir ilişkiye sahiptir.

DNS Zone görevleri:

Alan Adının Yönetimi: DNS zone, belirli bir alan adının yönetimini sağlar. Bu alan adı, internet üzerinde veya özel bir ağda bulunabilir. Alan adının alt alan adları ve bu alt alan adlarının kayıtları bu bölgede düzenlenir. Baktığımızı zaman DNS Zone'un en temel görevi alan adı yönetimi diyebiliriz.

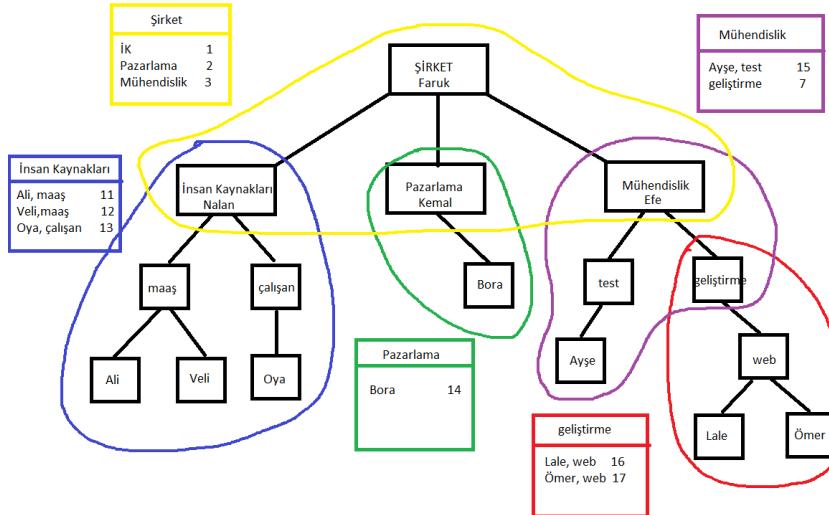
DNS Kayıtlarını İçerir: DNS zone, alan adının IP adresleriyle ilişkilendirildiği DNS kayıtlarını içerir. Bu kayıtlar, A kayıtları (IPv4), AAAA kayıtları (IPv6), MX kayıtları (e-posta sunucuları), CNAME kayıtları (kanonik ad), PTR kayıtları (ters çevrim), SOA kayıtları ve daha fazlasını içerebilir.

Yönetim Yetkisi: DNS zone, bölgeyi yönetme yetkisine sahip otoriter DNS sunucularını belirtir. Bu sunucular, alan adının kayıtlarını saklar ve diğer DNS sunucularına dağıtır.

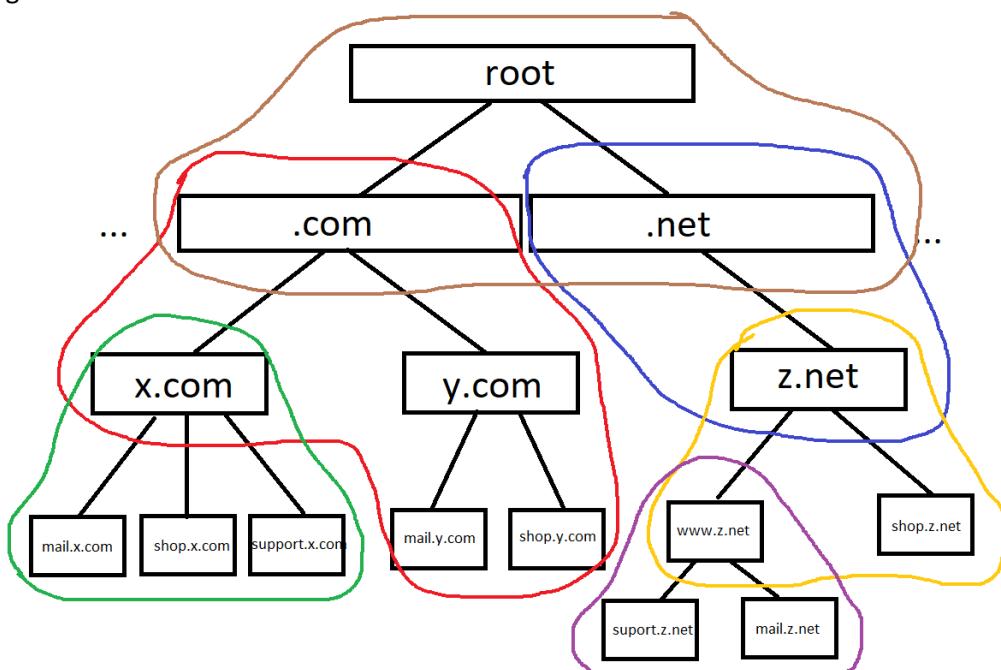
Güncelleme ve Senkronizasyon: DNS zone, kayıtlarda yapılacak güncellemeleri yönetir. Yeni bir alan adı veya alt alan adı oluşturulduğunda, IP adresleri değiştiğinde veya diğer değişiklikler yapıldığında, bu değişiklikler zone üzerinde yapılır ve diğer DNS sunucularına senkronize edilir.

DNS Hiyerarşisinin Parçası: DNS zone, genel DNS hiyerarşisinin alt seviyelerini oluşturur. Alan adları ve alt alan adları, bu hiyerarşi içinde farklı DNS bölgelerini temsil eder.

Alan Adı Dağıtımı: DNS zone, alan adlarının IP adresleriyle ilişkilendirilmesini sağladığı için kullanıcıların alan adlarını IP adreslerine çözümlemelerini mümkün kılar. Bu sayede internet kullanıcıları, alan adlarına dayalı olarak web sitelerine, e-posta sunucularına ve diğer hizmetlere erişebilirler.



Şimdi DNS Zone'u anlamaya çalışalım. Bir şirket düşünelim. Mesela elimizde 5 tane zone (bölge) olsun. Göründüğü gibi dikdörtgen kutularda iletişim bilgileri bulunmakta. Bunun yanı sıra DNS Zone'larda o zone ile ilgili yönetim bilgileri, iletişim bilgileri veya daha fazla bilgi bulunur. Bu şirketörneğinde sadece iletişim bilgisini gösterdik. Örneğin Sarı renkli zone'da bulunan şirket sahibi Faruk, insan kaynaklarından sorumlu Nalan ile görüşecek diyalim. Bu işlemi doğrudan yapabilir çünkü aynı zone içindeler. Ama pazarlama biriminde çalışan Bora ile iletişimde bulunmak isteseydi, Faruk pazarlama biriminin başındaki Kemal ile görüşecekti ve bilgileri ondan alacaktı. Çünkü Bora'nın bulunduğu zone'daki kilit nokta Kemal.



Resimde görüldüğü gibi Domain adlarını zone'lara ayıralım. DNS zone'lar arası iletişim biraz daha karmaşıktır. Bunu için DNS Zone Transfer kullanılabilir.

DNS Zone Transfer Nedir?

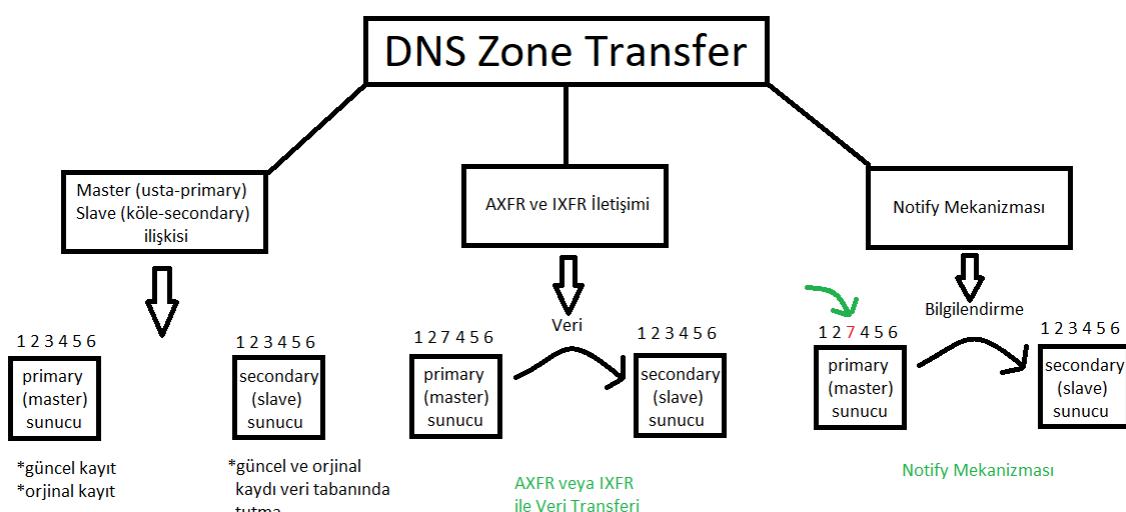
Zone Transfer, bir DNS sunucusunun, belirli bir alan adına ait tüm kayıtları diğer bir DNS sunucusuna kopyalama işlemidir. Bu genellikle birincil (primary) ve ikincil (secondary) DNS sunucuları arasında gerçekleşir. Birincil sunucu, alan adıyla ilgili kayıtları saklar ve günceller. İkincil sunucular ise bu bilgileri, belirli periyotlarla veya güncelleme talebi yapıldığında birincil sunucudan alarak kendi veritabanlarına kopyalarlar.

DNS Zone Transfer Neden Kullanılır?

- Yedekleme ve Redundancy (Yedekleme ve Yedek Sunucular):** Zone Transfer, birincil (primary) DNS sunucusundaki verilerin ikincil (secondary) sunuculara kopyalanmasını sağlar. Bu, veri kaybını önler ve servis sürekliliğini sağlar. Eğer birincil sunucu çökerse, ikincil sunucular hala erişilebilir olabilir.
- Dağıtık DNS Altyapısı:** Zone Transfer, farklı coğrafi konumlarda bulunan sunucular arasında veri dağıtımını ve senkronizasyonunu sağlar. Bu, kullanıcıların daha hızlı ve güvenilir DNS yanıtları olmasını sağlar.
- Veri Senkronizasyonu:** Büyük ölçekli ağlarda veya farklı alt alanlarda, DNS kayıtlarının senkronize edilmesi önemlidir. Zone Transfer, bu senkronizasyonu gerçekleştirerek tüm sunucuların güncel bilgilere sahip olmasını sağlar.
- DNS Yönetimi Kolaylığı:** Zone Transfer, DNS yöneticilerinin değişiklikleri tek bir sunucuda yapmalarına ve bu değişikliklerin diğer sunuculara otomatik olarak iletilmesine olanak tanır. Bu, yönetim kolaylığı ve veri bütünlüğü sağlar.

DNS Zone Transfer Nasıl Çalışır?

DNS Zone Transfer'i anlamak için bazı terimleri bilmekte fayda vardır. Bu terimleri 3 farklı terimi anlamak lazım: Bunlar master-slave ilişkisi, AXFR ve IXFR iletişimini, Notify Mekanizması.



Master-Slave İlişkisi: DNS zone'lar arası iletişimde master-slave (ustaca-köle) ilişkisi yer alır. Master (ustaca) DNS sunucusu (Primary Zone), bölgenin orijinal ve güncel kayıtlarını saklar. Slave

(köle) DNS sunucuları ise (Secondary Zone) master sunucusundan bu kayıtları düzenli olarak alır ve kendi veritabanlarında saklar. Slave sunucular, master sunucudan gelen güncellemelere tepki verir ve senkronize olur.

Primary Zone (Birincil Bölge): Bir Primary Zone, bir DNS bölgesinin asıl ve güncel kayıtlarını içeren bölgedir. Bu zone, bölgenin orijinal kayıtlarını saklar, güncellemeleri kabul eder ve yönetim yetkisine sahiptir. Primary Zone sahibi, bu bölge için kayıt ekleme, düzenleme ve silme işlemlerini gerçekleştirir. Kayıtlarda yapılan güncellemeler, Primary (master-usta) Zone'da gerçekleştirildikten sonra Secondary Zone'lara (köle-slave) aktarılır.

Secondary Zone (İkincil Bölge): Bir Secondary Zone, Primary Zone'dan (birincil bölge) veri kopyalayarak güncelleyen bölgedir. Secondary Zone, yedeklenmiş bir kopya olarak hizmet verir ve bu kopya, Primary Zone'dan düzenli aralıklarla veya bildirim mekanizmalarıyla güncellenir. Secondary Zone, DNS sunucuları arasında yük dengesi ve yedeklenmiş verilerin sağlanması için kullanılır. Eğer Primary Zone'daki sunucu çevrimdışı kalırsa, Secondary Zone hala güncel verileri sunabilir.

AXFR ve IXFR İletişimi: Zone Transfer (Bölge Aktarımı) olarak da adlandırılan AXFR (Full Zone Transfer) ve IXFR (Incremental Zone Transfer), master sunucudaki DNS kayıtlarının slave sunuculara aktarılmasını sağlar. AXFR, bütün bölge kayıtlarının tam bir kopyasını slave sunucuya ileterek güncelleme sağlar. IXFR ise yalnızca değişen veya eklenen kayıtları slave sunucuya ileterek daha verimli bir güncelleme sağlar.

Notify Mekanizması: Master sunucu, kayıtlarda bir değişiklik olduğunda slave sunucuları hızlıca bilgilendirmek için Notify mekanizmasını kullanır. Slave sunucular, bu bildirimi aldıktan sonra güncellemeleri almak için gerekli işlemi başlatır.

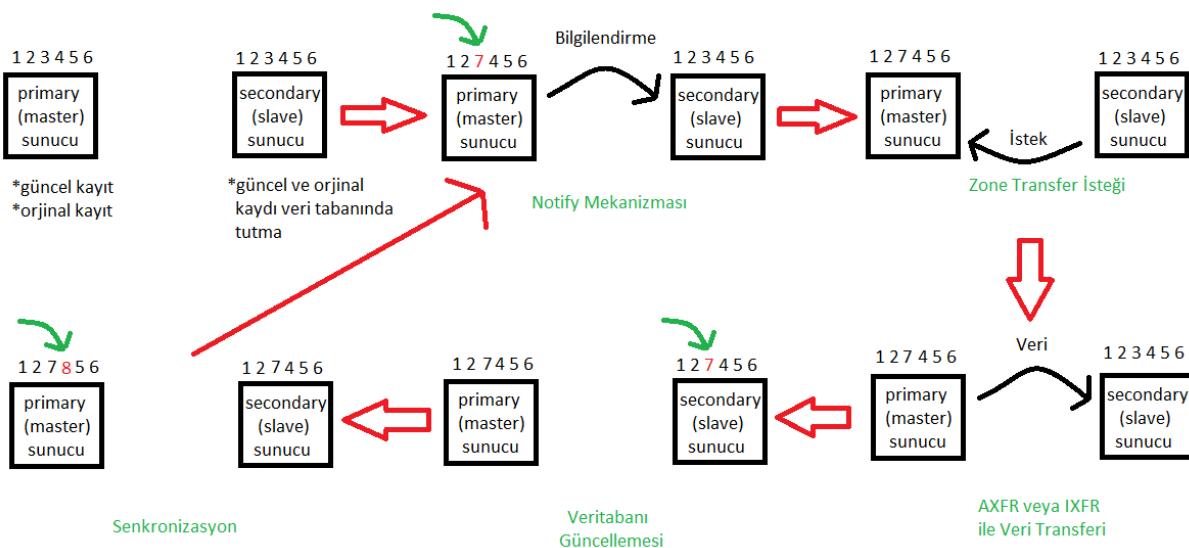
Notify mekanizmasının DNS Zone transferlerindeki rolünü yerine getiren bazı alternatifler sunlar olabilir: Periodic Zone Transfer (Düzenli Bölge Transferi), DNS Kaynak Sorgulaması (DNS Resource Query), Daha Düşük TTL Değerleri, Kendi Kendine Güncelleme (Self-Notify). Unutulmaması gereken önemli bir nokta, Notify mekanizmasının hızlı ve etkili bir şekilde bölge transferlerini sağlayan yaygın olarak kabul görmüş bir yöntem olduğunudır. Diğer alternatif yöntemler, duruma ve gereksinimlere bağlı olarak kullanılabilecek seçeneklerdir ancak genellikle Notify mekanizması daha hızlı ve verimli sonuçlar elde etmeye yardımcı olur.

Master (Primary) Sunucu Örnekleri:

ns1.example.com: Örneğin, "example.com" alan adının Primary (Master) sunucusu olan ns1.example.com sunucusu bölgenin orijinal ve güncel verilerini içerir. Bu sunucu üzerinde yapılan değişiklikler, diğer sunuculara iletilir.

Slave (Secondary) Sunucu Örnekleri:

ns2.example.com: "example.com" alan adının Secondary (Slave) sunucusu olan ns2.example.com sunucusu, ns1.example.com'dan bölge kayıtlarını kopyalayarak güncel tutar. Bu sunucu, yedeklenmiş ve yedeklenmiş verilere hızlı erişim sağlar.



Master (Primary) ve Slave (Secondary) Sunucular: DNS Zone Transfer işlemi, bir Primary Zone (Master) sunucusu ile bir veya daha fazla Secondary Zone (Slave) sunucusu arasında gerçekleşir. Primary sunucu, bölgenin orijinal ve güncel kayıtlarını içerirken, Secondary sunucular bu kayıtları kendi veritabanlarında saklayarak yedek ve hızlı yanıt sunarlar.

Notify Mekanizması (İstege Bağlı): Primary sunucu, kayıtlarda bir değişiklik olduğunda Notify mekanizması ile Secondary sunucuları hızlıca bilgilendirir. Bu mekanizma, güncellemelerin daha hızlı iletilmesini sağlar, ancak zorunlu değildir.

Zone Transfer İsteği: Secondary sunucular, belirli aralıklarla veya Notify mekanizmasıyla Primary sunucudan bölge verilerini almak üzere Zone Transfer isteği gönderirler.

AXFR veya IXFR Mekanizması: Primary sunucu, gelen Zone Transfer isteğine AXFR (Full Zone Transfer) veya IXFR (Incremental Zone Transfer) protokollerinden birini kullanarak yanıt verir.

AXFR (Full Zone Transfer): Primary sunucu, bütün bölge kayıtlarını tam bir kopya olarak Secondary sunucuya gönderir. Bu, yeni bir Secondary Zone oluşturulduğunda veya büyük değişiklikler olduğunda kullanılır.

IXFR (Incremental Zone Transfer): Primary sunucu, yalnızca bölgdedeki değişen veya eklenen kayıtları Secondary sunucuya gönderir. Bu şekilde, bölgdedeki küçük değişiklikler veya günlük güncellemeler gibi durumlar için daha verimli bir seçenektedir.

Veri Transferi: Primary sunucu, AXFR veya IXFR mekanizması kullanarak bölge verilerini Secondary sunucuya aktarır.

Veritabanı Güncellemesi: Secondary sunucu, gelen bölge verilerini kendi veritabanına entegre eder ve güncelleme işlemi tamamlanır.

Senkronizasyon: Secondary sunucu, belirli aralıklarla veya Notify mekanizması ile güncellemeleri kontrol eder ve gerektiğinde tekrar Zone Transfer isteği göndererek bölge verilerini güncel tutar.

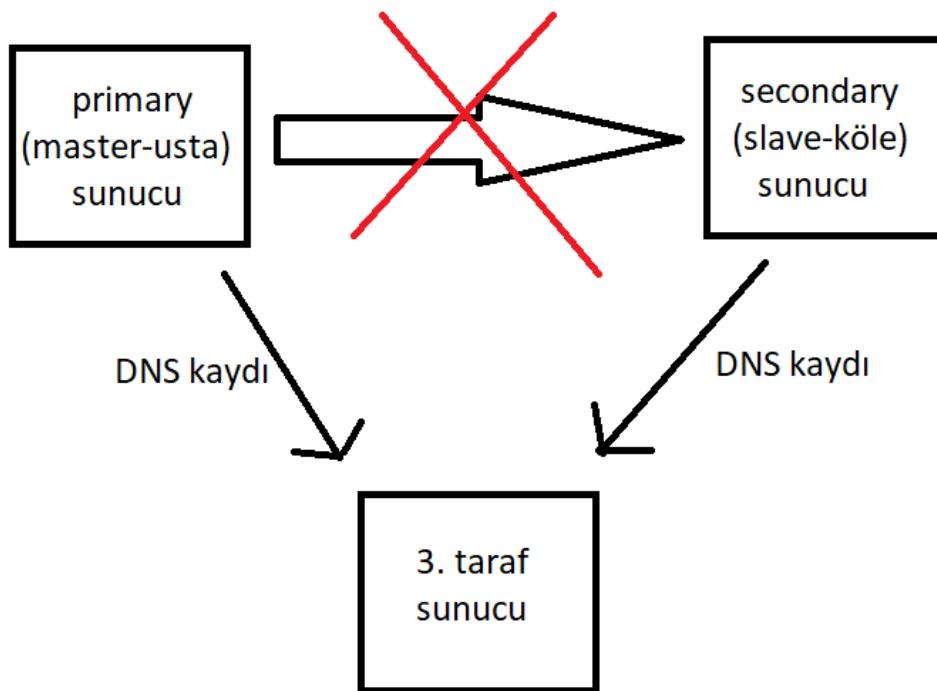
DNS Zone Transfer Saldırısı Nedir?

DNS Zone Transfer Saldırısı, bir saldırganın hedeflenen bir alan adının DNS sunucusundan tüm DNS kayıtlarını (zone) elde etmeye çalıştığı bir saldırı türüdür. Bu saldırı, genellikle zone transfer protokolünün yanlış yapılandırılmış veya açık bırakılmış bir DNS sunucusu üzerinden gerçekleştirilir.

DNS Zone Transfer Saldırısı Neden Kullanılır?

- **Ağ Topolojisi Keşfi:** DNS Zone Transfer, ağdaki sunucuların, alt alanların ve IP adreslerinin tam listesini sağlar. Saldırganlar bu bilgileri kullanarak ağın yapısını anlayabilir ve zayıf noktaları tespit edebilir. Örneğin, hassas sistemlere veya hedeflenen sunuculara erişmek için bir giriş noktası olarak kullanılabilir.
- **Saldırı Stratejisi Oluşturma:** DNS Zone Transfer ile elde edilen bilgiler, daha sofistike saldırı stratejileri için bir temel oluşturabilir. Saldırganlar, hedef ağa daha etkili bir şekilde sizme veya saldırısı gerçekleştirmek için bu bilgileri kullanabilirler.
- **Sosyal Mühendislik:** Zone Transfer ile elde edilen bilgiler, saldırının sosyal mühendislik saldırıcıları için kullanabileceğii değerli bilgiler içerebilir. Örneğin, bir şirketin iç ağ yapısı veya çalışanların e-posta adresleri gibi bilgiler, dolandırıcılık veya kimlik avı saldıruları için kullanılabilir.
- **Bilgi Toplama:** Saldırganlar, DNS Zone Transfer kullanarak hedeflenen alan adına ait tüm DNS kayıtlarını ele geçirerek bilgi toplayabilirler. Bu, hedef organizasyon hakkında daha fazla bilgi edinmek için kullanılabilir.

DNS Zone Transfer Saldırısı Nasıl Çalışır?



Eğer DNS Zone Transfer, primary sunucudan secondary sunucuya değil de primary veya secondary sunucudan 3. taraf bir sunucuya (saldırgana) doğru yapılrsa buna DNS Zone Transfer saldırısı denir.

DNS Dynamic Update Saldırısı

DNS Dynamic Update Nedir?

DNS Dynamic Update, DNS kayıtlarının dinamik olarak güncellenebilmesini sağlayan bir mekanizmadır.

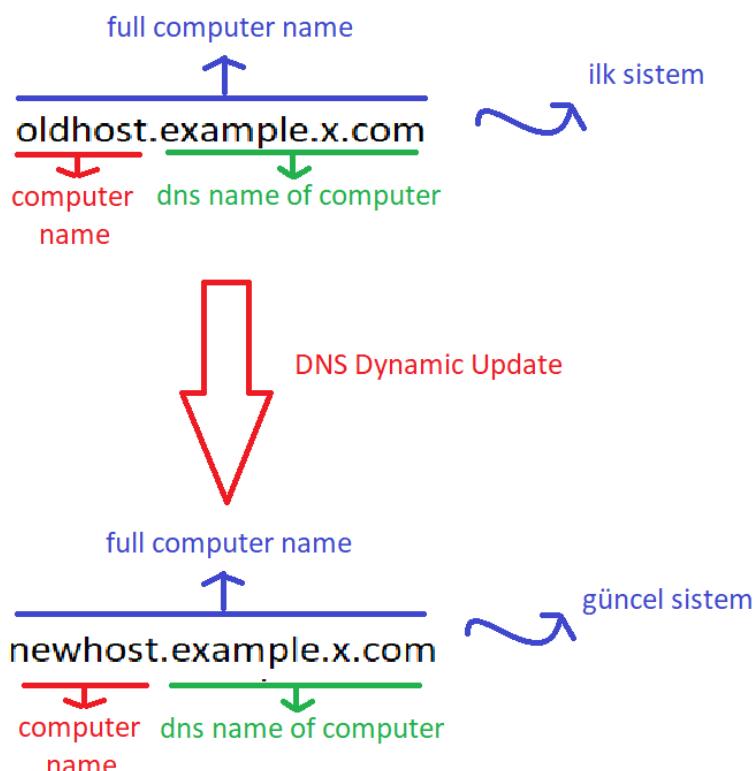
Geleneksel olarak, DNS kayıtları genellikle elle yönetilir ve bir sunucu üzerindeki kayıtların güncellenmesi gerektiğinde manuel olarak değiştirilir. Ancak DNS Dynamic Update, sistem yöneticilerinin otomatik olarak kayıtları eklemesine, değiştirmesine veya silebilmesine izin verir. Bu, özellikle büyük ağlar veya dinamik IP adresi atanması gibi durumlarda faydalı olabilir.

DNS Dynamic Update işlemi, genellikle DNS sunucuları ve DNS istemcileri arasında gerçekleşir. DNS Dynamic Update, DNS kayıtlarını dinamik olarak güncellemek için kullanılır. Ancak, her cihazın doğrudan bu işlemi gerçekleştirebilmesi mümkün değildir.

DNS Dynamic Update Neden Kullanılır?

- **Dinamik IP Adresleri:** Dinamik IP adreslerinin kullanıldığı durumlarda (örneğin, DHCP ile IP adresi atanması), DNS kayıtlarının otomatik olarak güncellenmesi gereklidir. DNS Dynamic Update, IP adresi değiştiğinde otomatik olarak DNS kaydını güncelleyerek kullanıcıların doğru şekilde hizmet almasını sağlar.
 - **Dinamik IP Adres Nedir?**
- **Büyük Ağlar:** Büyük ağlarda, sürekli olarak yeni cihazlar eklenir veya cihazların IP adresleri değişimlidir. DNS Dynamic Update, bu değişiklikleri yönetmek ve DNS kayıtlarını sürekli olarak güncellemek için kullanılır.
- **Dinamik Hizmet Kayıtları:** Bazı hizmetler, özellikle bulut tabanlı veya dağıtık sistemler, dinamik olarak kayıtlarını günceller. Örneğin, bir web sunucusunun IP adresi veya hedeflenen uygulama sunucusunun adresi değişikçe, DNS kayıtlarının güncellenmesi gereklidir.
- **Kullanıcı Doğrulaması:** DNS Dynamic Update, kullanıcıların kimlik doğrulama işlemleri için kullanılabilir. Örneğin, bir kullanıcının değişen IP adresi üzerinden DNS kayıtları güncellenecek erişim yetkisi sağlanabilir veya engellenebilir.

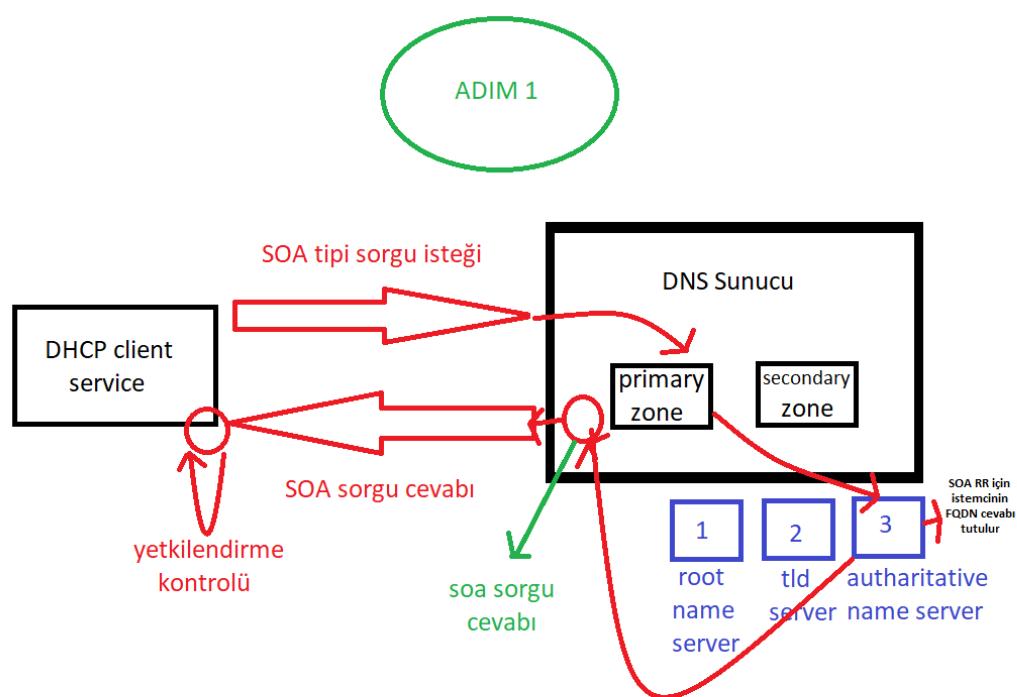
DNS Dynamic Update Nasıl Çalışır?



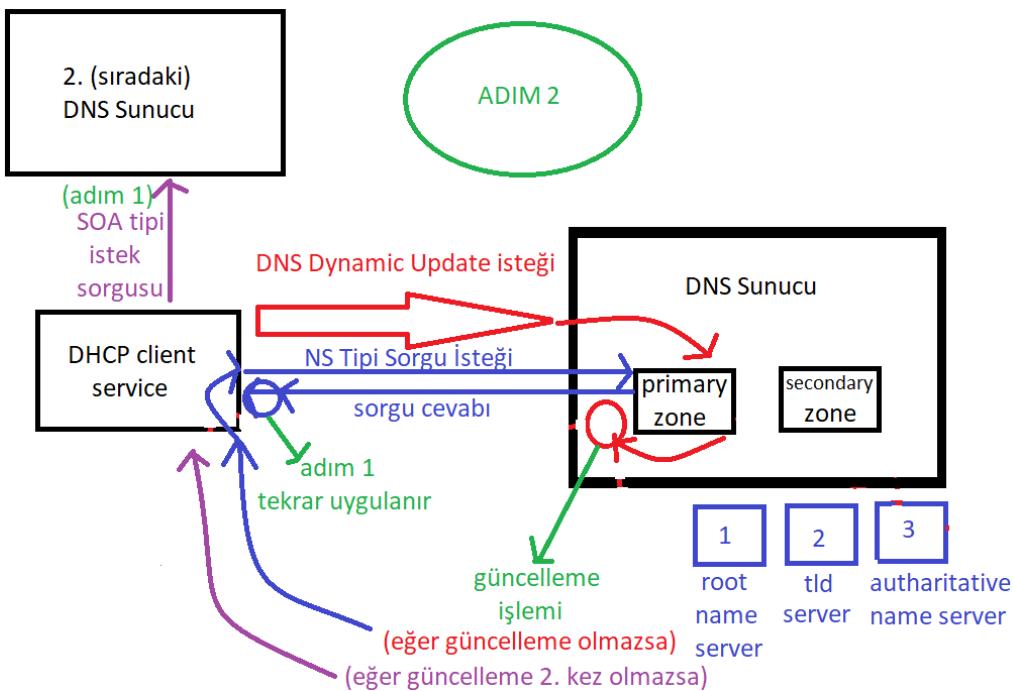
Elimizde bilgisayar adı (computer name), bilgisayarın dns adı (dns name of computer) ve tam bilgisayar adı (full computer name) olmak üzere 3 bileşenimiz var. DNS Dynamic Update işleminde genelde bilgisayar adı değişir. Buna bağlı olarak tam bilgisayar adı da değişir.

DNS Dynamic Update, genellikle belirli bir cihazın tam DNS adını değiştirmez, ancak o cihazla ilişkili olan kaydı (örneğin, IP adresi veya ana bilgisayar adı) günceller. Yani, "example.x.com" altında yer alan bir cihazın adı değiştiğinde, altında bulunan kayıt (örneğin, oldhost.example.x.com) "newhost.example.x.com" olarak güncellenebilir, ancak ana domain adı olan "example.x.com" değişmeyebilir. Bu, altındaki cihazların adlarının değişmesiyle ilgili bir güncellemeyidir.

DNS Dynamic Update çalışma prensibi 2 aşamadan oluşur diyebiliriz.



İlk aşamaya bakacak olursak, başlangıçta DHCP (Dynamic Host Configuration Protocol) istemcisinden SOA tipi sorgu isteği DNS sunucuya gönderilir. Daha sonra bu istek primary zone bölgesine iletilir. Primary zone ise isteği autharitative DNS sunucuya (autharitative name server) yönlendirir. Autharitative Name Server üzerinde, SOA kaydı için istemcinin tam domain adı (FQDN newhost.example.x.com gibi) cevap olarak hazırlanır. SOA sorgu cevabı DHCP client'a iletilir. Burada yetkilendirme kontrolü yapılır. Bu kontrolde IP adres ile doğru DNS sunucuda çalışıp çalışmadığımız belli olur. Eğer yetkilendirme başarılı olursa, yani doğru DNS sunucuda çalışıyorsak adım 2'ye geçilir.



Eğer yetkilendirme başarılı olursa bu sefer DHCP istemcisinden DNS sunucuya “DNS Dynamic Update” isteği gönderilir. Bu istek primary zone bölgесine iletılır. İletilen istek sonucunda güncelleme işlemi gerçekleşir. Eğer güncelleme işlemi başarılı olmazsa tekrar DHCP istemcisine gidilir. Buradan NS tipi Sorgu isteği primary zone bölgесine yönlendirilir. İsteğe karşı, istemciye sorgu cevabı döner. Buradan sonra tekrar adım 1 uygulanır. Yani tekrar bir SOA isteği ile yetkilendirme kontrolüne kadar işlemler gerçekleşir. Yetkilendirme kontrolü adımdan sonra ise DNS Dynamic Update isteği primary zone bölgесine iletılır. Primary zone bölgесinden çıkan cevap sonucu güncelleme işlemi yapılır. Eğer güncelleme işlemi 2. kez yine başarılı olmazsa DHCP istemciye yönleniriz. Bu sefer DNS Dynamic Update işlemi sıradaki (2.) DNS sunucu üzerinde yapılır. Önceki adımlar teker teker 2. DNS sunucu üzerinde gerçekleşir.

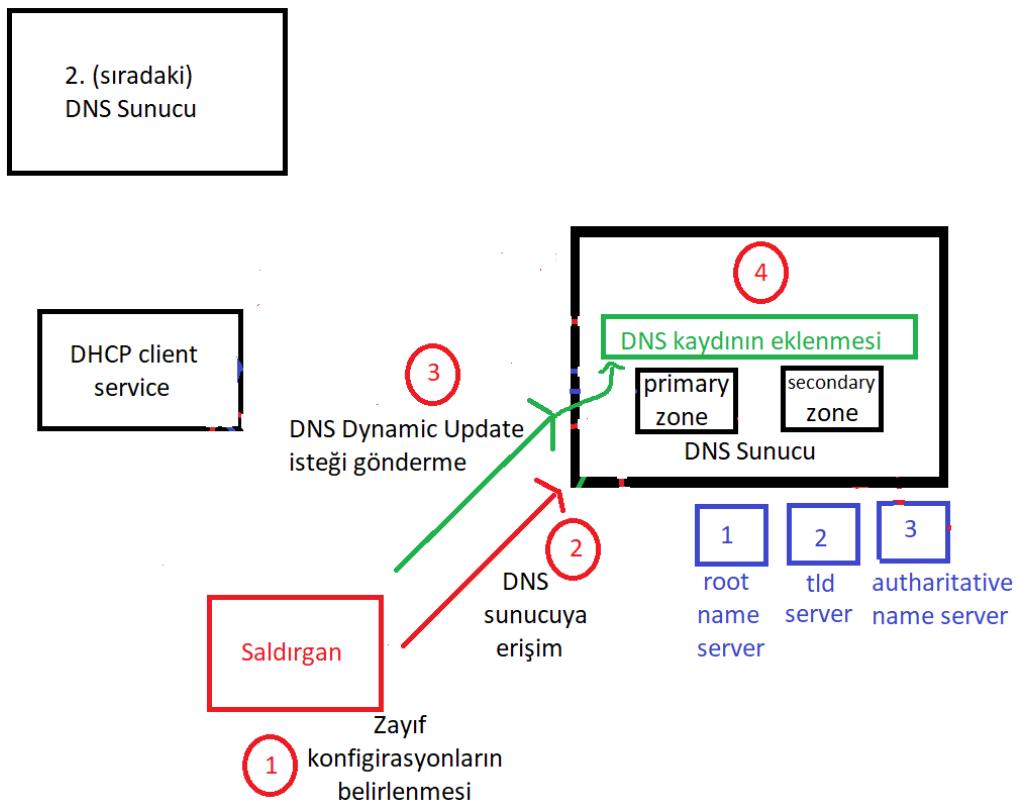
DNS Dynamic Update Saldırısı Nedir?

DNS Dynamic Update’ı öğrendik. DNS Dynamic Update saldırısı ise genellikle DNS sunucularının zayıf konfigürasyonlarını veya güvenlik açılarını hedef alır. Bu saldırı türünde, saldırganlar DNS sunucularına yanlış veya kötü niyetli DNS kayıtları ekleyerek, güvenilir olmayan IP adreslerine sahip domain adlarını yönlendirme yeteneğine sahip olabilirler.

DNS Dynamic Update Saldırısı Neden Kullanılır?

- **Phishing Saldırıları:** Saldırganlar, DNS kayıtlarını değiştirerek, hedeflerini sahte web sitelerine yönlendirebilir. Bu şekilde, kullanıcıları yaniltarak hassas bilgilerini çalmaya çalışabilirler.
- **Man-in-the-Middle (MITM) Saldırıları:** DNS Dynamic Update saldırıcıları, saldırılara ağ trafiğini yönlendirme yeteneği sağlar. Bu, saldırıcıların veri iletişimini dinleme veya değiştirme olasılığını artırabilir.
- **Servis Kesintileri:** Saldırganlar, DNS kayıtlarını değiştirerek hedefin servislerine erişimi engelleyebilir veya kesintiye uğratabilirler. Bu, hizmet dışı bırakma saldırılara (DoS) benzer bir etki yaratabilir.
- **Kötü Amaçlı Yazılım Dağıtıımı:** Saldırganlar, DNS kayıtlarını manipüle ederek kullanıcıları kötü amaçlı yazılım içeren sahte sitelere yönlendirebilirler. Böylece, bilgisayarlarına zararlı yazılım bulaştırabilirler.
- **Reputation Hijacking:** Saldırganlar, DNS kayıtlarını değiştirerek meşru bir hizmetin veya kuruluşun itibarını lekeleyebilirler. Örneğin, bir bankanın DNS kayıtları değiştirilerek kullanıcılar sahte bir web sitesine yönlendirilebilir, bu da müşteri güvenini zedeler. "hijacking" terimi genel bir kontrol ele geçirme eylemini ifade ederken, "Reputation Hijacking" özel olarak itibarın hedef alındığı siber saldıruları belirtir. Yani müşteri güvenini sarsmayı hedef alır.

DNS Dynamic Update Saldırısı Nasıl Çalışır?



- 1- **Zayıf Konfigürasyonların Belirlenmesi:** Saldırganlar, hedef DNS sunucularını belirlemeye çalışırken, zayıf konfigürasyonlara sahip veya güvenlik açıkları içeren sunucuları tespit etmeye çalışır.
- 2- **DNS Sunucusuna Erişim:** Saldırganlar, hedef DNS sunucusuna erişim sağlamak için çeşitli yöntemleri kullanabilir. Bu, sunucuya yetkisiz erişim elde etmek veya kullanıcı hesaplarını ele geçirmek anlamına gelebilir.
- 3- **Dynamic Update İsteği Gönderme:** Saldırganlar, elde ettikleri yetkilerle hedef DNS sunucusuna Dynamic Update isteği gönderirler. Bu istek, yeni DNS kayıtları eklemeyi, mevcut kayıtları güncelleme veya kaldırmayı içerebilir. Elde ettikleri yetkilerle diyoruz çünkü herhangi bir istemci DNS Dynamic Update isteği gönderemez. Bunun için işlemi gerçekleştirmek için gereken yetkilere sahip olması gereklidir.
- 4- **Kötü Niyetli DNS Kayıtlarının Eklenmesi veya Güncellenmesi:** Saldırganlar, Dynamic Update özelliğini kullanarak hedef DNS sunucusuna kötü niyetli DNS kayıtları ekleyebilir veya mevcut kayıtları değiştirebilirler. Bu kayıtlar, kullanıcıları sahte sitelere yönlendirmek veya veri çalmak için kullanılabilir.
- 5- **Yanılma ve Sosyal Mühendislik:** Saldırganlar, DNS kayıtlarını değiştirerek kullanıcıları yanıltabilirler. Örneğin, sahte bir banka web sitesine yönlendirilen kullanıcılar, hassas bilgilerini girmeye ikna edilebilir.
- 6- **Phishing ve Veri Çalma:** Kötü niyetli DNS kayıtları aracılığıyla saldırganlar, kullanıcıları phishing saldırılara maruz bırakabilir ve hassas bilgileri çalmak amacıyla sahte siteler oluşturabilirler.

DNS Saldırılarına Karşı Çözüm

DNSSEC (Domain Name System Security Extensions) nedir ve internet güvenliği açısından önemi nedir?

DNSSEC, DNS'yi DNS cache spoofing, spoofing, dnz zone transfer saldırısı da dahil olmak üzere çeşitli saldırırlara karşı daha güvenli hale getirir.

DNSSEC, DNS kayıtlarını dijital olarak imzalayarak çalışır. Bu, kayıtların değiştirilmemişinden veya sahte olmadığından emin olmanızı sağlar. DNSSEC ayrıca, kullanıcıların DNS sunucusunun yanıtlarının gerçekten yetkili bir kaynaktan gelip gelmediğini doğrulamasına olanak tanıyan bir kimlik doğrulama mekanizması da sağlar.

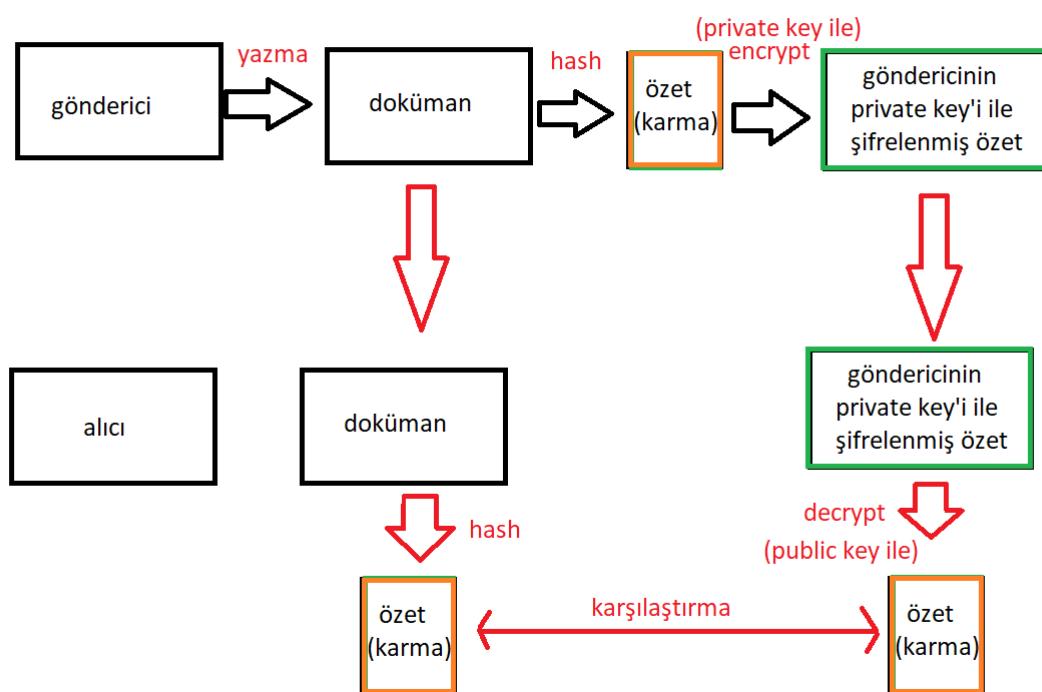
DNSSEC, İnterneti daha güvenli hale getirmeye yardımcı olabilecek önemli bir güvenlik teknolojisidir. Güvenlik ve daha iyi performans sunmaya yardımcı olur.

DNSSEC Nasıl Çalışır?

Dijital İmza Nedir?

Dijital imza, elektronik ortamda kimlik doğrulama ve belge bütünlüğü sağlamaya yönelik bir yöntemdir. El yazısı imza gibi, belgeyi imzalayan kişiyi tanımlayan dijital imzayı kısaca bir mesaja ya da belgeye eklenmiş bir kod gibi düşünübilirsiniz.

Dijital İmza Nasıl Çalışır?



- 1- Gönderici dokümanı yazarak oluşturur.
- 2- Doküman hashlenerek özet (karma) hale getirilir.
- 3- Özet, göndericinin private key'i ile şifrelenerek (encrypt) şifreli özet hale getirilir.
- 4- Alıcı, dokümanın orijinal halini ve şifreli özetini alır.
- 5- Dokümani hashler ve özet (karma) hale getirir. Aynı zamanda şifreli özetin şifresini göndericinin public key'i ile çözerek (decrypt) özet (karma) elde eder. (public key herkese açıktır ancak private key yalnızca imza sahibi tarafından tutulur.)
- 6- Eğer oluşan bu iki özet birbirinin aynısı ise belge imza sahibi tarafından imzalanmış ve değiştirilmemiş demektir.

Public Key vs Private Key

Public Key

Public key, kriptografi alanında kullanılan bir anahtar türüdür. İki parçadan oluşan asimetrik bir şifreleme sisteminin bir parçasıdır. Public key herkese açık bir şekilde dağıtılmakla birlikte, diğer parça olan private key ise gizli tutulmalıdır.

Public key'in işlevleri:

- Verileri şifrelemek:** Public key ile şifrelenen veriler, yalnızca private key ile çözülebilir. Bu sayede, veriler sadece yetkili kişiler tarafından okunabilir.
- Dijital imzalar oluşturmak:** Public key ile oluşturulan dijital imza, verilerin bütünlüğünü ve kaynağını doğrulamak için kullanılır.

Public key'in kullanım alanları:

- Güvenli iletişim:** E-posta şifrelemesi, mesajlaşma uygulamaları gibi güvenli iletişim protokolleri public key altyapısını kullanır.
- Dijital imzalama:** Yazılımlar, belgeler ve diğer dijital veriler public key ile imzalanarak, kim tarafından oluşturulduğu ve değiştirilmediği doğrulanabilir. (Bizim bu bölümde ilgilendiğimiz alan)
- SSL/TLS sertifikaları:** Web sitelerinde güvenli bağlantı sağlamak için kullanılan SSL/TLS sertifikaları public key altyapısına dayanır.

Private Key

Private key, kriptografi alanında kullanılan bir anahtar türüdür. Public key ile birlikte asimetrik şifreleme sisteminin iki önemli parçasından biridir. Bu sistemde, public key herkese açık bir şekilde dağıtılrken, private key ise gizli tutulmalıdır.

Private key'in işlevleri:

- Verileri çözmek:** Public key ile şifrelenen veriler, yalnızca private key ile çözülebilir. Bu sayede, public key ile size gönderilen şifreli mesajları yalnızca siz okuyabilirsiniz.
- Dijital imzalar oluşturmak ve doğrulamak:** Private key ile oluşturulan dijital imza, verilerin bütünlüğünü ve kaynağını doğrulamak için kullanılır. Siz bir belgeyi private keyiniz ile imzaladığınızda, herkes public keyiniz ile bu imzanın sizin tarafınızdan atıldığı ve belgenin değiştirilmediğini doğrulayabilir.

Private Key Kullanım Alanları:

Private key'in birçok kullanım alanı mevcuttur. Bunlar:

- Şifreleme (Verileri şifreleme, Dijital imza oluşturma)
- Kimlik Doğrulama (MFA)
- Onaylama ve Yetkilendirme
- Anahtar Yönetimi (Diğer anahtarları yönetme) ve Güvenli iletişim

Private key'in gizli tutulması çok önemlidir. Eğer private keyiniz çalınırsa, bunun birçok zararı olabilir:

- Verilerinizin açığa çıkması:** Private keyiniz ile şifrelenen veriler açığa çıkabilir.
- Sahtecilik:** Private keyiniz ile sizin adına dijital imzalar oluşturulabilir. Bu, önemli belgelerin değiştirilmesi veya sahte işlemler yapılması gibi sorunlara yol açabilir.

Private key: İmza anahtarıdır. Gizli tutulmalıdır.

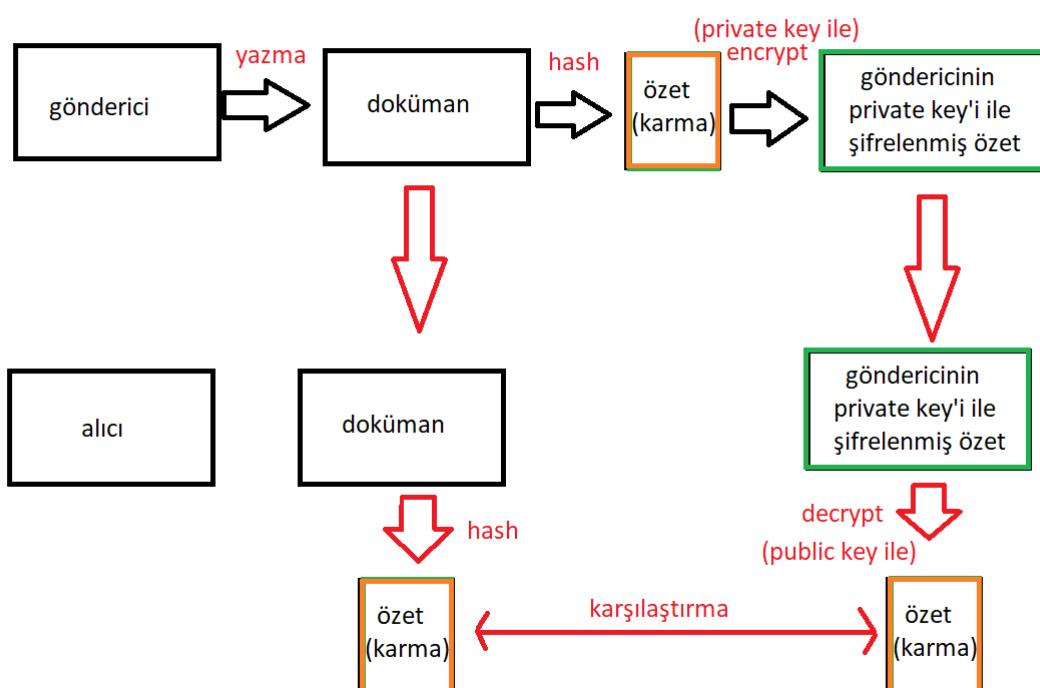
Public key: Doğrulama anahtarıdır. Herkese açıktır.

Özellik	Private Key	Public Key
Erişim	Gizli	Herkese açık
İşlev	Dijital imzalar oluşturmak	Dijital imzaları doğrulamak
Güvenlik	Daha hassas	Daha az hassas

Hangi anahtarın kullanılacağı:

- Bir belgeyi imzalamak için private key kullanılır.
- Bir belgenin doğruluğunu kontrol etmek için public key kullanılır.
- Örnek:
 - Bir e-posta gönderirken, e-postayı imzalamak için private key'inizi kullanırsınız.
 - E-postayı alan kişi, e-postanın doğruluğunu kontrol etmek için sizin public keyinize erişebilir.

Public ve Private Key'in nasıl kullanılacağına dijital imza örneğinde debynmişlik. Aşağıdaki görselden yukarıda belirtilenleri daha net anlayabiliriz. (Dijital imza konusunda görseldeki bilgiler anlatıldı)



Key Signing Key (KSK) ve Zone Signing Key (ZSK)

Key Signing Key (KSK) Nedir?

Kriptografi alanında, Key Signing Key (KSK), bir imza anahtarıdır. Asimetrik şifreleme sistemlerinde kullanılır ve genellikle bir üst düzey otorite (üst seviye zon yetkilisi gibi) tarafından tutulur. KSK, bir imza zinciri içinde üst düzeyde yer alır ve bu zincir ZSK'ları (Zone Signing Key) imzalamak için kullanılır. KSK ile aşağıdaki işlemler gerçekleştirilir:

- ZSK'ların güvenilirliğini sağlamak:** KSK, diğer anahtarların, özellikle Zone Signing Key'lerin (ZSK) imzalanması için kullanılır. ZSK'lar ise alt zonlardaki kayıtların imzalanmasından sorumludur.
- Güvenlik ve Yetki:** KSK, yalnızca yetkili kişiler tarafından kontrol edilir ve sıkı güvenlik önlemleri altında tutulur. Bu sayede, imzalama yetkisinin kötüye kullanılması engellenir.

Avantajları:

- Sahtecilik Koruma:** Alan (domain) bilgilerinin sahtecilik yapılmasını öner, böylece kullanıcılar gerçek internet sitelerine ulaştıklarından emin olabilirler.
- Veri Büyünlüğü:** Alan (domain) bilgilerinin değiştirilmeden doğru şekilde iletilmesini sağlar.
- Güvenli Bağlantı:** Internet kullanıcıları ile internet siteleri arasında güvenli bir bağlantı kurulmasını sağlar.
- Baktığımızda temel işlev güvenlik diyebiliriz.**

Dezavantajları

- Karmaşıklık:** DNSSEC protokolü ve KSK kullanımı teknik olarak karmaşık olabilir.
- Yönetim Zorluğu:** KSK'nın güvenli bir şekilde yönetilmesi ve güncellenmesi gereklidir.
- Performans Etkisi:** DNSSEC ve KSK kullanımı DNS çözümleme süresini biraz uzatabilir.

Zone Signing Key (ZSK) Nedir?

Kriptografi alanında, bir Zone Signing Key (ZSK), bir imza anahtarıdır. Asimetrik şifreleme sistemlerinde kullanılır ve genellikle bir alt zonun yetkilisi (alan adı sahibi veya alan adı yöneticiliği yapan kurum) tarafından tutulur. ZSK'nın temel işlevi, alt zon içerisindeki DNS kayıtlarının imzalanmasıdır. Bu imzalama sayesinde, DNS kayıtlarının bütünlüğü ve doğruluğu sağlanır.

ZSK'nın temel işlevi, alan bilgilerini digital olarak imzalayarak DNS sunucuları üzerinden güvenli bir şekilde dağıtmaktır. Bu sayede kullanıcılar, bir internet sitesine bağlandığında aslında doğru siteye ulaştıklarından emin olabilirler.

Avantaj ve dezavantaj olarak ZSK'da verdigimiz örnekleri burada da gösterebiliriz.

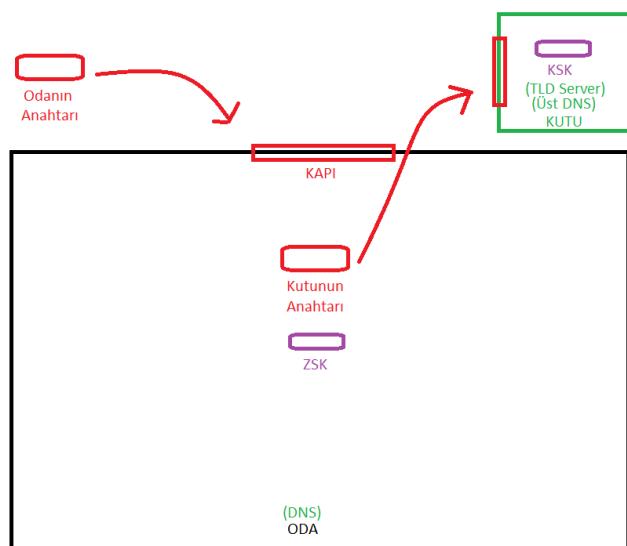
ZSK ile KSK (Key Signing Key) Arasındaki İlişki:

- ZSK doğrudan en üst düzey yetkiliye (KSK sahibi) iletilmez.
- Parent zone'daki Key Signing Key (KSK) kullanılarak ZSK imzalanır. Bu imza, ZSK'nın geçerliliğini ve güvenilirliğini sağlar.
- Bu sayede, bir alt zonun yetkilisi kendi ZSK'sini üretebilirken, bu anahtarın doğruluğu üst düzey yetkili tarafından kontrol edilmiş olur.

Baktığımız zaman her iki anahtarın görevleri benzer gibi görünebilir ancak bu doğru değil. Her ikisinin de farklı özellikleri bulunur.

Kısa bir örnekle gösterirsek, elimizde farklı konumlarda bulunan bir oda ve kutu olsun. KSK'yi korumak daha kritiktir. Dolayısıyla onu gizli bir kutuda saklarız. Bu kutuya erişmek için ise kutunun anahtarına ihtiyacımız var. Ancak kutunun anahtarının, odanın içerisinde saklandığını düşünelim. Dolayısıyla KSK'ya erişmek için hem oda hem de kutunun kilit anhtarına ihtiyacımız var. Fakat ZSK oda da bulunduğu için sadece onanın anahtarları işimizi görecek. İşte oda DNS sunucusu temsil ederken, kutu üst dns (TLD server) konumu temsil eder. Bu örnek tamamen KSK ve ZSK'nın konumlandırmasını açıklar.

Görevleri ise KSK, ZSK'yi imzalar. ZSK ise alan adının kayıtlarını imzalar.



Zone Singing Key ve Key Signing Key Farkları

Özellik	Zone Signing Key (ZSK)	Key Signing Key (KSK)
Görev	Alan adının kayıtlarını imzalar	ZSK'yi imzalar
Konum	Alan adının DNS sunucusunda	Üst veya üst üst DNS sunucusunda
İmzalama Yetkisi	Sadece kendi alan adının kayıtları	Diğer alan adlarının ZSK'ları da imzalayabilir
Geçerlilik Süresi	Kısa (haftalar veya aylar)	Uzun (yıllar)
Güvenlik Önemi	Yüksek (alan adının kayıtlarının bütünlüğü)	Kritik (tüm imza zincirinin güvenilirliği)

KSK ve ZSK Yapısı

KSK (Key Signing Key) ve ZSK (Zone Signing Key) asimetrik şifrelemede kullanılan private ve public keylerden oluşur. Asimetrik şifrelemede bir anahtar çifti kullanılır: biri gizli tutulan private key, diğer ise herkese açık olan public key.

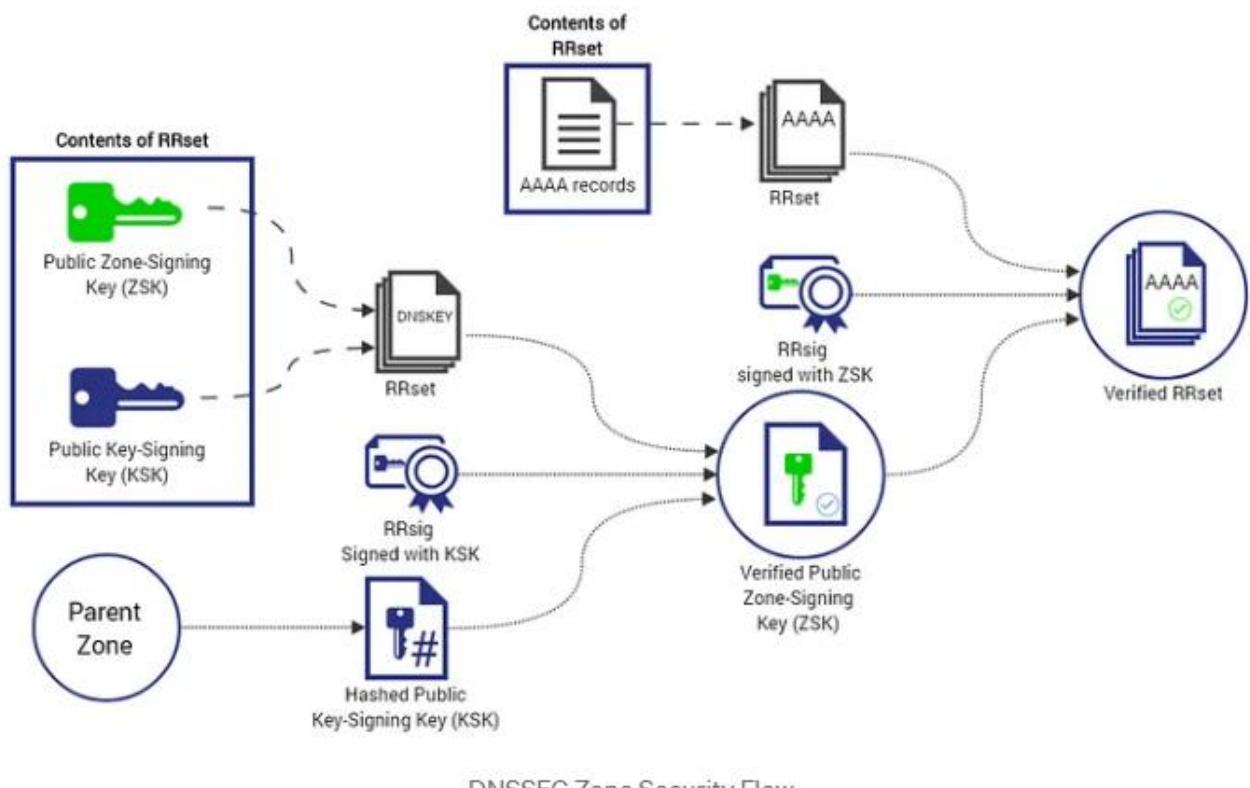
KSK (Key Signing Key):

- **Private KSK:** Üst düzey otorite (genellikle parent zone yetkilisi) tarafından sıkı güvenlik önlemleri altında tutulur.
- **Public KSK:** Normalde doğrudan dağıtılmaz. Bunun yerine, hashed KSK (KSK'nın hash değeri) bir Delegation Signer (DS) kaydı içerisinde parent zone'da saklanır. Alt zonlar, bu DS kaydına erişerek hashed KSK'yi öğrenebilirler.

ZSK (Zone Signing Key):

- **Private ZSK:** Alt zonun yetkilisi (alan adı sahibi veya alan adı yöneticiliği yapan kurum) tarafından güvenli bir şekilde saklanır.
- **Public ZSK:** Alt zonun DNS sunucusunda yayınlanır. Bu sayede, DNS istemcileri, ZSK ile imzalanmış kayıtların doğruluğunu doğrulayabilirler.

DNSSEC Güvenliği Akış Şeması



Public Zone Signing Key ve Public Key Signing Key

Hatırlarsak KSK ve ZSK public ve private olmak üzere ikiye ayrılmıştır. Bu görseldeki PZSK ve PKSK, aslında ZSK ve KSK'nın public key versiyonlarıdır. Yani farklı bir kavram yok. (Bir önceki görselin hemen üstünde anlatıldı)

RRSet Nedir? / DNSKEY RRSet

RRSet'ler, DNSSEC'nin güvenli bir şekilde çalışması için önemlidir. RRSet'lerdeki dijital imzalar sayesinde, internet kullanıcıları DNS kayıtlarının sahteciliğe ve manipülasyona karşı korunmuş olduğundan emin olabilirler.

RRSet (Resource Record Set - Kaynak Kayıt kümesi), DNSSEC (Domain Name System Security Extensions) protokolünde kullanılan bir terimdir. Bir kaynak kaydı kümesi anlamına gelir ve bir alan adına ait aynı türdeki tüm kaynak kayıtlarını içerir.

DNSKey RRSET:

- Alan adının DNSSEC ile korunması için gerekli olan bir kaynak kayıt setidir.
- İki ana bileşenden oluşur:
 - Public Zone-Signing Key (ZSK): Ana bölgeyi imzalamak için kullanılır.
 - Public Key-Signing Key (KSK): ZSK'yi imzalamak için kullanılır.
- Public ZSK:
 - Ana bölgedeki tüm kayıtların kimliğini doğrulamak için kullanılır.
 - Nispeten sık değiştirilir.
 - ZSK'nın güncel bir kopyası, alan adının DNS sunucularında bulunur.
- Public KSK:
 - ZSK'nın kimliğini doğrulamak için kullanılır.
 - Nadiren değiştirilir.
 - KSK'nın bir kopyası, alan adının kök sunucularında bulunur.

Birden fazla RRSet tipleri bulunabilir. Bunlar:

- A
- AAAA
- MX
- DNSKEY
- ...

Resim üzerinde Zone Signing Key ve Key Signing Key'in bir RRSet oluşturduğunu gördük. Bu RRSet tipi ise DNSKEY RRSet'tir. DNSKEY RRSet'in bazı görevleri bulunur. Bunlar:

- **Alan adının DNSSEC'ye imza yetkisini doğrulamak:** DNSKEY RRSet'teki kamu anahtarları, alan adının DNS kayıtlarının dijital olarak imzalanmasını ve doğrulanmasını sağlar. Bu sayede, internet kullanıcıları DNS kayıtlarının sahteciliğe ve manipülasyona karşı korunmuş olduğundan emin olabilirler.
- **Güvenli bir bağlantı kurmak:** DNSKEY RRSet'teki kamu anahtarları (public key), DNS sunucuları ile internet kullanıcıları arasında güvenli bir bağlantı kurulmasına yardımcı olur. Bu sayede, kullanıcıların gerçek DNS sunucularına bağlandıklarından emin olunur.

DNSKEY RRSet Yapısı

Name	Type	Class	TTL	RDLENGTH	FLAGS	Protocol	Algorithm	Public Key	lifetime
_acme.example.com.	DNSKEY	IN	3600	256	3	8	RSA/ SHA-256	AwEAAaz...	0 3

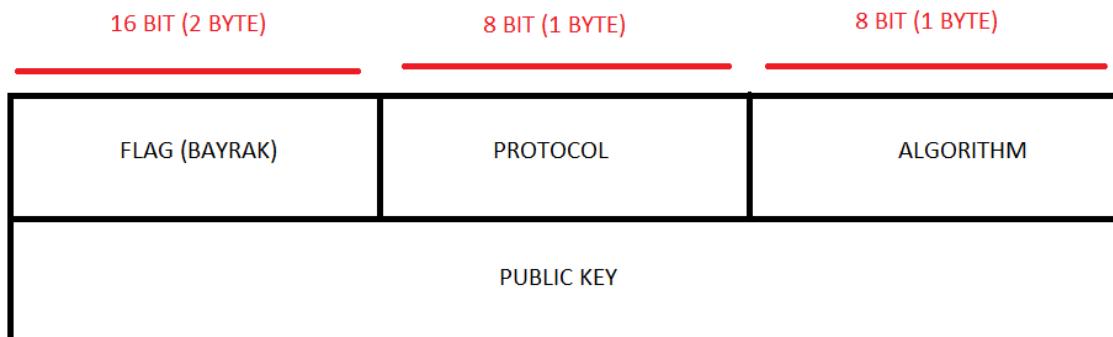
1. **_acme.example.com.:** Kaydın ait olduğu alan adı
2. **DNSKEY:** Kaydın türü
3. **IN:** Kaydın ait olduğu sınıf (internet).
4. **3600:** Kaydın önbelleğe alınma süresi (1 saat).
5. **256:** Anahtarın uzunluğu (256 bit).
6. **3:** Anahtarın nasıl kullanılacağına dair bilgiler (örneğin, imzalama)
7. **8:** Anahtarın hangi algoritmayla kullanılacağına dair bilgiler (RSA).
8. **RSA/SHA-256:** Anahtarın hangi kriptografik algoritmayla oluşturulduğuna dair bilgiler.
9. **AwEAAaz...:** Alan adının DNS kayıtlarını imzalamak için kullanılan kamu anahtarı.
10. **0 3:** Anahtarın kullanım ömrü.
11. Bizi asıl ilgilendirenler: FLAGS, Protocol, Algorithm ve Public Key.

DNSKEY RRSET Nasıl Oluşur?

1. **Algoritma Seçimi:** RSA, DSA veya ECDSA gibi bir algoritma seçmeniz gereklidir. Genellikle RSA algoritması tercih edilir.
2. **Anahtar Uzunlukları:** ZSK ve KSK için farklı anahtar uzunlukları seçebilirsiniz. Genellikle ZSK için 2048 bit ve KSK için 4096 bit anahtar uzunluğu önerilir. ZSK alan adının bilgilerini kilitler. KSK ise ZSK'nın kullandığı kilidin tutulduğu yeri kilitler. Böylece bu iki anahtar ile DNSKEY kaydı oluşturulur.
3. **Anahtar Çifti Oluşturma:** Her anahtar türü için (ZSK ve KSK) bir public key (genel anahtar) ve private key (özel anahtar) içeren bir anahtar çifti oluşturmanız gereklidir. OpenSSL gibi bir komut satırı aracı veya DNSSEC yönetim aracı kullanabilirsiniz.
4. **DNSKEY Kayıtları:** Her anahtar türü için bir DNSKEY kaydı oluşturmanız gereklidir. Bunu DNSKEY RDATA Wire Format biçiminde gerçekleştirebiliriz. DNSKEY kaydı, protocol, public key, algoritma ve anahtar flags (bayrakları) içerir.
 - o **DNSKEY DATA Wire Format**
 - **Wire Format**, verinin nasıl kodlandığını ve bir ağ üzerinden nasıl aktarıldığını belirleyen formattır. DNSKEY DATA (RDATA) için bu format, bayt dizileri

olarak belirlenir. Bir DNSKEY kayıt setindeki public key ve diğer bilgilerin nasıl kodlandığını ve aktarıldığını tanımlayan formatı ifade eder. Bu format bir standarda oturtulmuştur ve DNS yazılımları tarafından yorumlanabilir.

- **DNSKEY RDATA Wire Format ile Geliştirilen DNSKEY Kaydı Nelerden Oluşur?**



- 1. Bayrakları (Flags) belirleyin:
 - AD bayrağını, anahtarın DNSSEC ile imzalanmış kayıtları doğrulamak için kullanılcaksa ayarlayın.
 - CD bayrağını, anahtarın DNSSEC ile imzalanmamış kayıtları kontrol etmek için kullanılmayacaksanız ayarlayın.
 - UF bayrağını, anahtar güncellendiye ayarlayın.
 - RRSIG bayrağını, anahtarın RRSIG kayıtlarını imzalamak için kullanılcaksa ayarlayın.
 - NSEC bayrağını, anahtarın NSEC kayıtlarını imzalamak için kullanılcaksa ayarlayın.
 - Z (Zero) bayrağında, Şu anda atanmış bir işlevi yoktur.
- 2. Protokolü (Protocol) belirleyin:
 - Anahtarın hangi protokol ile kullanılacağını gösterir.
 - DNSSEC için protokol 3'tür.
- 3. Algoritmayı (Algorithm) belirleyin:
 - Anahtarın hangi kriptografik algoritma ile oluşturulduğunu gösterir.
 - RSA/SHA-1 için algoritma 5'tir.
 - RSA/SHA-256 için algoritma 8'dir.
 - ECDSA with SHA-256 için algoritma 13'tür.
 - Genellikle RSA algoritması tercih edilir.
- 4. Public Key'i (Genel Anahtar) edinin:
 - Public key, anahtarın türüne ve algoritmasına bağlı olarak değişen uzunlukta bir alandır.
 - Public key, verileri şifrelemek ve dijital imzalar oluşturmak için kullanılır.

```
<name> IN DNSKEY <flags> <protocol> <algorithm> <public key> <key id>
[<exponent> <modulus>] [<curve> <x coordinate> <y coordinate>]
```

```
example.com. IN DNSKEY 256 3 8 AwEAAaz... 12345 [65537 1234567890]
```

5. **DNS Sunucusuna Ekleme:** DNSKEY kayıtlarını alan adınızın DNS sunucusuna eklemeniz gereklidir. DNS sunucunuzun türüne bağlı olarak, farklı adımlar izlemeniz gereklidir. Peki DNSKEY kaydını DNS sunucusuna nasıl ekleriz? Kısaca:
- 1- Her anahtar için DNSKEY kaydı oluşturmuştu.
 - 2- Oluşan DNSKEY kayıtlarını alan adınızın zon dosyasına eklememiz gereklidir. Bunun için çeşitli araçlar kullanılabilir (BIND, PowerDNS, DNSSEC-Tools, OpenSSL).

(OpenSSL ile oluşan DNSKey kaydını DNS Zone'a eklemek):

ZSK için:

```
openssl genrsa -out example.com.zsk.key 2048
openssl dsaparam -out example.com.zsk.params 2048
openssl req -new -key example.com.zsk.key -out example.com.zsk.csr
openssl dsapub -in example.com.zsk.params -out example.com.zsk.pubpub
```

KSK için:

```
openssl genrsa -out example.com.ksk.key 4096
openssl dsaparam -out example.com.ksk.params 4096
openssl req -new -key example.com.ksk.key -out example.com.ksk.csr
openssl dsapub -in example.com.ksk.params -out example.com.ksk.pub
```

DNSKEY kayıtları:

```
example.com. IN DNSKEY 256 3 8 AwEAAaz... 12345 [65537 1234567890]
example.com. IN DNSKEY 257 3 8 AwEAAaz... 54321 [65537 1234567890]
```

DNSSEC Zone Dosyası:

```
$ORIGIN example.com.
$TTL 3600
```

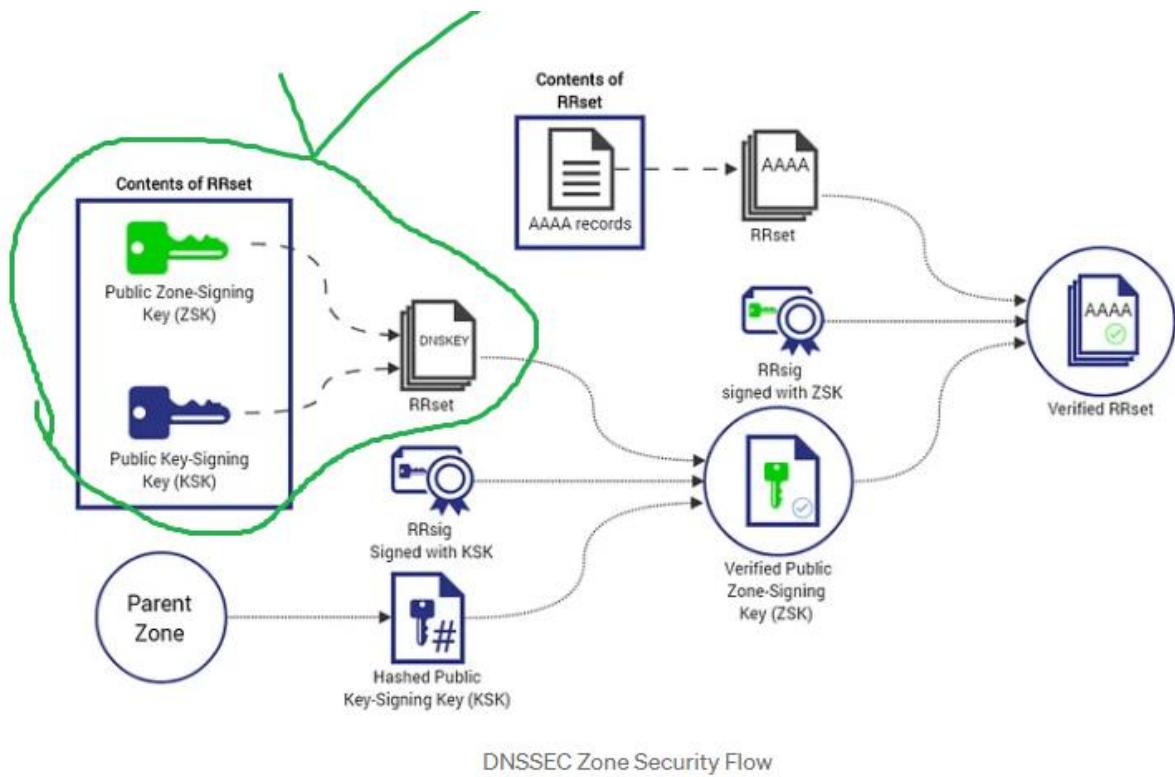
```
@ IN SOA ns1.example.com. hostmaster.example.com. (
    2023022300 3600 1800 604800 3600
)
```

```
@ IN NS ns1.example.com.
@ IN NS ns2.example.com.
```

```
ns1 IN A 192.0.2.1
ns2 IN A 192.0.2.2
```

```
example.com. IN DNSKEY 256 3 8 AwEAAaz... 12345 [65537 1234567890]
example.com. IN DNSKEY 257 3 8 AwEAAaz... 54321 [65537 1234567890]
```

- DNSKEY kaydı ekleme ile ilgili dikkat edilmesi gerekenler:
 - Public key'leri ve private key'leri güvenli bir şekilde saklayın.
 - KSK'yi daha az sıkılıkla değiştirin.
 - ZSK'yi daha sık değiştirin.
 - DNSSEC ile ilgili güncel bilgilere sahip olun.



Şu ana kadar DNSSEC çalışma mantığında ZSK, KSK ve DNSKEY RRset'e baktık. Sırada KSK ile imzalanmış RRsиг var.

RRsig - KSK ile İmzalanmış RRsig Nedir?

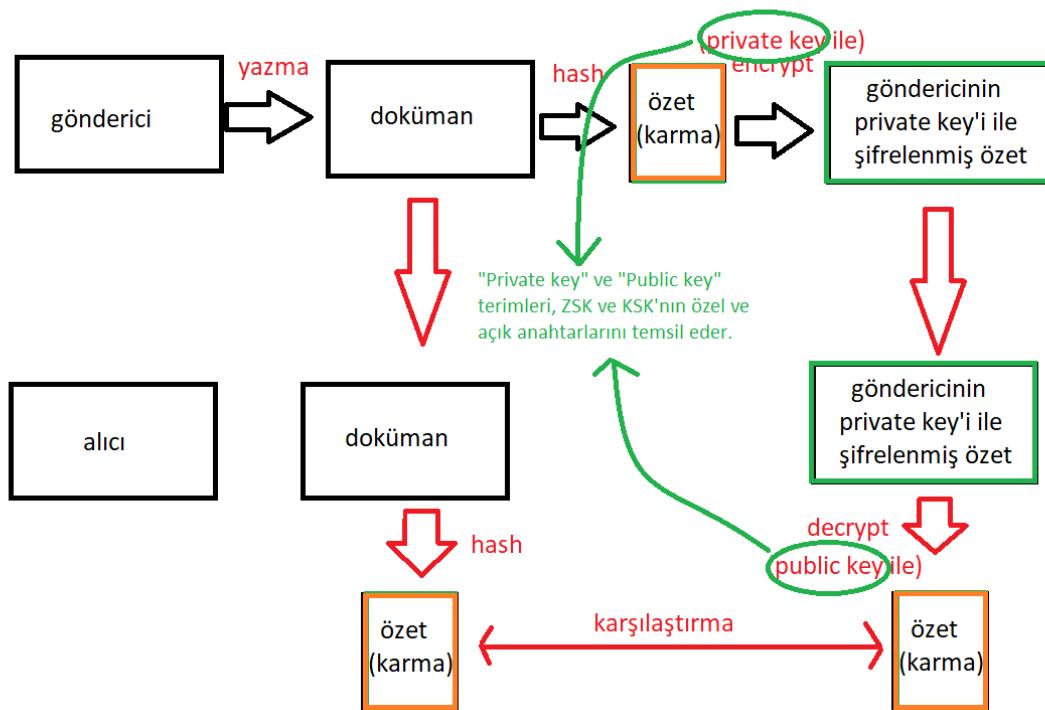
RRSig, bir DNS kaydınının yetkililiğini doğrulamak için kullanılan bir dijital imzadır. Bu imza, özel bir anahtarla imzalanır ve herkese açık bir anahtarla doğrulanabilir. Daha önce dijital imzanın ne olduğuna ve nasıl çalıştığını değinmiştık.

Key signing key (KSK) ise, RRSig'ları imzalamak için kullanılan özel bir anahtاردır. Bu anahtar, güvenli bir şekilde saklanmalı ve yetkisiz erişime karşı korunmalıdır. KSK'nın ne olduğuna bir önceki başlıklta bakmıştık.

RRSig'ların KSK ile imzalanmasının faydaları şunlardır:

- Artan güvenlik:** RRSig'lar, DNS kayıtlarının yetkililiğini doğrulamak için daha güvenli bir yöntemdir. Çünkü geleneksel yöntemlerden farklı olarak, kayıtlar yetkili olmayan kişiler tarafından değiştirilemez.
- Sahteciliği önleme:** RRSig'lar, sahteciliği önlemeye yardımcı olur. Çünkü bir saldırgan bir DNS kaydını değiştirmeye çalışırsa, imzayı da taklit etmek zorunda kalır.

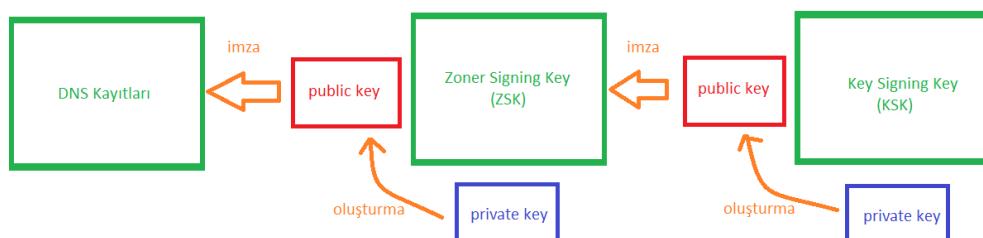
KSK ile İmzalanmış RRsig Nasıl Oluşur?



RRsig bir dijital imza demisitik. Dijital imzanın nasıl oluştuğuna yukarıdaki örnekteki gibi daha önce debynmişitik. Burada RRsig oluşumu için bakacak olursak private ve public key KSK (key signing key) kullanılarak imza oluşur.

Public ve private keylerin, KSK ve ZSK ile bağlantısını bilmek kafa karışıklığını ortadan kaldırabilir:

- KSK ve ZSK, public ve private anahtar çifti olarak kullanılır.
- Dijital imza işleminde private key encryption işlemini gerçekleştirirken, public key decryption işlemini gerçekleştirir. Ancak tek görevleri bunlar değildir. Private key kullanarak public key elde edebiliriz.
- KSK'nın private key'i public key'i oluşturmada yardımcı olur. KSK'nın public key'i, ZSK'nın imzalanmasını ve doğrulanmasını sağlar.
- ZSK'nın private key'i public key'i oluşturmada yardımcı olur. ZSK'nın public key'i ise DNS kayıtlarının doğrulanmasını sağlar.
- Bu cümlelerden anlaşılacağı üzere KSK ve ZSK'nın, public ve private keyleri bulunur.
- Private key, public key'in oluşturulmasında kullanılan kriptografik işlemlerin temelini oluşturur.
- Şu görsel KSK ve ZSK'ların mantığını bizlere anlatabilir.



Eğer Private anahtardan Public anahtarın nasıl olduğunu daha detaylı anlamak istersek matematiksel işlemlere inmemiz gereklidir. Bir örnek üzerinde bu işlemleri gerçekleştirelim. Ama önce kuralları belirleyelim. Kurallar:

- Prime Sayı Seçimi:
 - İki büyük asal sayı (p ve q) rastgele seçilir.
 - Asal sayıların büyüklüğü, istenen güvenlik seviyesine bağlıdır.
 - Asal sayıların birbirinden farklı olması önemlidir.
- Modulus Hesaplama:
 - Modulus değeri (n), p ve q 'nun çarpımıyla hesaplanır: $n = p * q$
- Euler Totient Fonksiyonu:
 - Phi (ϕ) fonksiyonu, n 'den küçük ve n ile asal olan tamsayıların sayısını hesaplar.
 - $\phi(n) = (p-1)(q-1)$
- Public Üs (e) Seçimi:
 - e , 1 ile $\phi(n)$ arasında rastgele seçilen bir tamsayıdır.
 - e ve $\phi(n)$ 'nin asal olması önemlidir.
- Private Üs (d) Hesaplama:
 - d , e 'nin modüler tersidir.
 - $d * e \equiv 1 \pmod{\phi(n)}$

Şimdi bir örnek ile gerçekleştirelim:

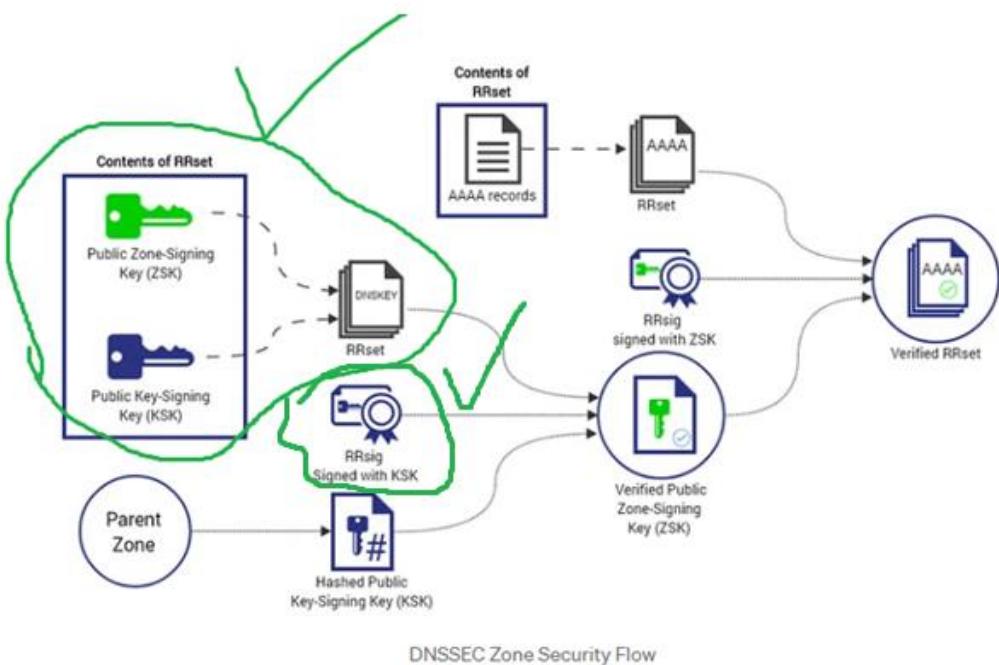
- Prime Sayı Seçimi:
 - Alice, rastgele iki büyük asal sayı seçer: $p = 17$ ve $q = 23$. Bu sayılar, mesajın güvenliğini belirler.
- Modulus Hesaplama:
 - Alice, modulus değerini hesaplar: $n = p * q = 17 * 23 = 391$.
- Euler Totient Fonksiyonu:
 - Alice, $\phi(n)$ değerini hesaplar: $\phi(n) = (p-1)(q-1) = (17-1)(23-1) = 360$.
- Public Üs (e) Seçimi:
 - Alice, 1 ile $\phi(n)$ arasında asal bir sayı seçer: $e = 7$.
- Private Üs (d) Hesaplama: (modüler ters hesaplama:
<https://www.youtube.com/watch?v=jXTWWtcGSck>)
 - Alice, d 'yi hesaplar: $d = e$ 'nin modüler tersi ($\text{mod } \phi(n)$) = 241.
 - $d * e \equiv 1 \pmod{\phi(n)}$ => $d * 7 \equiv 1 \pmod{360}$ => $d = 103$
 - $103 * 7 = 721$, $721 = 1 \pmod{360}$
- Public Key: $(n, e) = (391, 7)$
- Private Key: $d = 103$

Bu örnek RSA algoritması için temsili bir değerdir. RSA, asimetrik şifreleme algoritmasıdır. Bu algoritma, güvenli veri iletişimi ve dijital imza gibi birçok alanda yaygın olarak kullanılır.

Bu örnekte public keyden, private key elde etmiş gibi görünüyor ancak bu yanlış bir ifade olur. Eğer public key'den private key elde edilebilseydi, RSA algoritmasının güvenliği tehlikeye girecekti. Çünkü bir saldırgan, herhangi birinin public key'ini ele geçirerek bu yöntemle private key'i de elde edebilir ve şifrelenmiş mesajları çözerek gizli bilgilere erişebilirdi.

Yani özet geçersek:

- Kuralları açıklarken temel mantığı oluşturduk. Bu mantıktan private keyden public key elde edebiliriz.
- Daha sonra bu public key'i imza ile kullanarak KSK'dan ZSK, ZSK'dan DNS kaydı elde edebiliriz. (bundan bir önceki görsel)



Şimdi de Parent Zone'dan, Hashed Public KSK elde edilmesine bakalım. Öncelikle Parent Zone nedir buna değinelim.

Parent Zone – Child Zone

DNS'de, bir alan adının en üst seviyesini (top-level domain) temsil eden "root zone" veya "kök bölge" bulunur. Bu kök bölge, tüm alan adı sisteminin en üstündeki otorite noktasıdır ve tüm TLD'lerin (com, net, org, edu, gibi) kayıtlarını içerir. Diğer tüm alan adı bölgeleri, bu kök bölgesinden alt bölgelere (subzone) doğru hiyerarşik bir yapıda yer alır.

Hatırlarsak DNS içindeki hiyerarşik yapı kökten dışarıya sırasıyla Root Server, TLD Server ve Authoritative Name Server'dı. İşte bu sunucular arasında parent zone ve subzone ilişkisi vardır. "example.com." adlı bir domain düşünelim. Bu domainde ".." (Root Server'ı temsil eder), ".com" (TLD Server'ı temsil eder) için bir parent zone'dur. Aynı şekilde ".com", "example" (Autharitative Name Server'ı temsil eder) için bir parent zone'dur. Tam tersi durumda da Authoritative Name Server, TLD Server için subzone, TLD Server ise Root Server için bir subzone'dur.

Domain Name System'de (DNS) bir parent zone (üst zona), bir alt zonanın yönetimsel olarak üstünde yer alan, daha geniş kapsamlı bir zonadır. Bir bilgisayar ağacına benzetilebilecek hiyerarşik yapıda, parent zone ağacın gövdesi veya dalları konumunda olup alt zonalar ise yaprakları gibidir.

Parent zone'un üç önemli işlevi vardır:

- **Yetki devri:** Parent zone, alt zonaya kendi DNS kayıtlarını yönetmesine ve kendi kayıt sunucularına sahip olmasına yetkiyi devreder. Bu, domain adı sahibine daha fazla yönetim ve esneklik sağlar.
- **Delegasyon:** Parent zone, alt zonanın konumunu ve yetkili DNS sunucularını tanımlayan bir Delegation Set (Delegasyon Kümesi) kaydı aracılığıyla alt zonaya delege eder. Bu kayıt, sorugu doğru sunucuya yönlendirerek DNS sisteminin verimli çalışmasını sağlar.
- **Güvenlik (DNSSEC kullanılıyorsa):** Domain Name System Security Extensions (DNSSEC) kullanıldığında, parent zone, alt zonanın imza anahtarının doğruluğunu doğrulamak için bir Delegation Signer (Delegasyon İmzalayıcı) görevi üstlenebilir. Bu, güvenli bir iletişim ortamı sağlamaya yardımcı olur.

Hashed Public Key Signing Key (HPKSK)

Hashed PK SK, doğrulama anahtarının (public key) hash'ıdır. Doğrulama anahtarı (public key/genel anahtar), DNSSEC'de dijital imzaları doğrulamak için kullanılan genel anahtardır. Hashed PK SK ise bu anahtarın daha kısa ve sabit bir uzunluğa sahip bir değeridir.

DNSSEC'de, parent zone'dan (üst zona) alt zonanın DNSSEC imza anahtarlarının doğrulanması için hashed PK SK kullanılır. İşte resimde anlatılmak istenen, DNSSEC'de parent zone'dan hashed public key signing key (hashed PK SK) elde etmek için hem DS hem de NSEC3 kayıtları kullanılabilir. Bu işlem iki şekilde gerçekleştirilebilir:

- **Delegation Signer (DS Kaydı):** Delegation Signer (DS) kaydı, bir DNS zonunun alt zonanın DNSSEC imza anahtarlarının doğrulanmasını sağlayan bir DNS kayıt türüdür. Bu kayıt, parent zone'da (üst zon) bulunur ve alt zonanın hashed public key signing key'ini (hashed PK SK) içerir.
DNSsec üzerinde parent zone'dan, hashed public key signing key nasıl elde edilmesi yöntemlerinde doğrulama işlemi kullanılır.

DS Kaydı Nasıl İşler:

- Bir DNS istemcisi, bir alt zonadaki (örnek: tld server) bir kaydın imzasını doğrulamak ister.
- İstemci, parent zone'a (örnek: root server) erişerek alt zonanın DS kaydını arar.
- DS kaydı, alt zonanın hashed PK SK'sini içerir.
- İstemci, hashed PK SK'yi kullanarak alt zonanın imza anahtarının doğruluğunu kontrol eder.
- Doğrulama başarılı olursa, istemci alt zonadaki kaydın imzasının geçerli olduğundan emin olabilir.

DNSsec üzerinde parent zone'dan, hashed public key signing key nasıl elde edilir?

- Parent zone'a erişin: Alan adı yöneticinizin veya DNS sağlayıcınızın arayüzüünü kullanarak parent zone'un DNS kayıtlarına erişin.
- DS kaydını bulun: dig komutunu veya DNS sorgusu aracını kullanarak parent zone'da DS kaydını bulun. "Örnek: dig +short DS example.com."
- DS kaydını inceleyin: DS kaydında aşağıdaki bilgiler yer alır:
 - Key Tag: KSK'nın kimliği
 - Algorithm: Hash algoritması (örneğin, SHA-256)
 - Digest Type: Hash değeri türü (örneğin, SHA-256 256-bit)
 - Digest: KSK'nın hashed public key değeri

- Hashed public key'i ayıklayın: DS kaydındaki Digest alanını kopyalayın. Bu, parent zone'da bulunan KSK'nın hashed public key değeridir.
- Örnek:
 - ```
$ dig +short DS example.com
38477 13 2 256 3928159437
```
  - Bu örnekte, example.com alan adının parent zone'undaki KSK'nın hashed public key değeri: **3928159437**

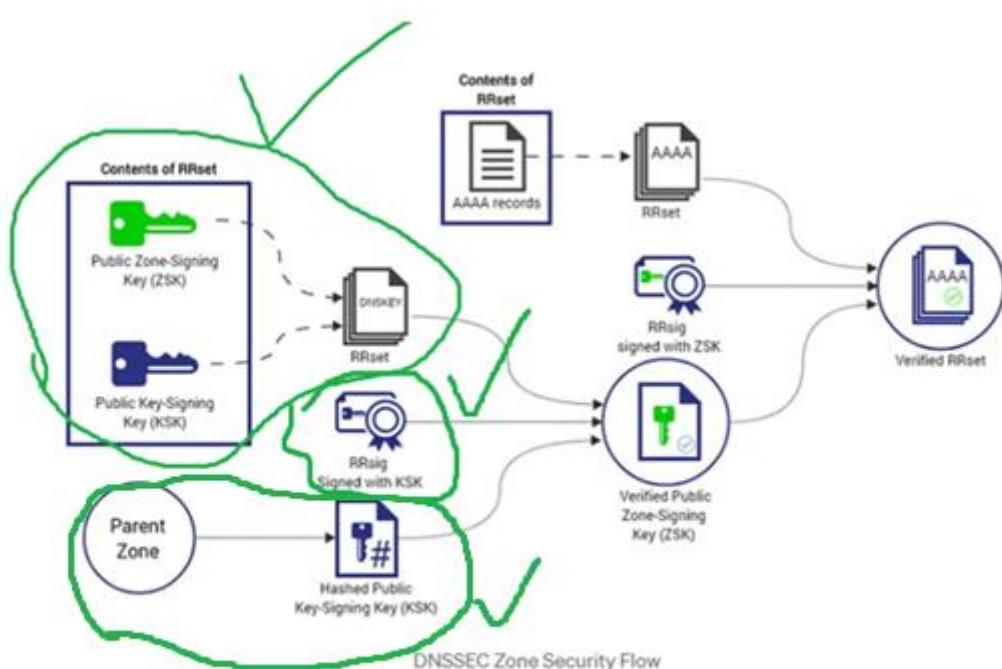
Yukarıda HPKSK elde edimini DS kaydının nasıl işlediğiyle gördük.

### Parent Zone'dan Hashed Public Key Signing Key elde etmek ne işe yarar?

- Güvenlik
  - Bir alt zonadaki bir kaydın imzasını doğrulamak için, DNS istemcisinin alt zonanın imza anahtarına (private key) ihtiyacı vardır. Private key'e doğrudan erişmek yerine hashed PK SK kullanmak daha güvenlidir. Bu sayede, private key'in sızdırılması riski azalır ve DNSSEC'nin güvenliği artar.
- Verimlilik ve Ölçeklenebilirlik
  - Her alt zone için ayrı bir private key yönetmek yerine, parent zone tek bir hashed PK SK'yi yönetebilir. Bu, DNSSEC anahtar yönetimini daha basit ve daha az karmaşık hale getirir. Özellikle büyük zonlarda bu durum önemlidir.
  - DS kaydı (Delegation Signer) kullanımıyla, parent zone alt zonanın hashed PK SK'sini sağlayabilir. Bu sayede, DNS istemcisi her alt zone için ayrı bir DNSSEC anahtarı aramak zorunda kalmaz. Bu da DNSSEC işlemlerini daha verimli hale getirir.

### DNSsec üzerinde parent zone'dan, hashed public key signing key nasıl elde edilir?

HPKSK, DS kaydı ile elde edilebildiğini küçük zonlar için görmüştük. Yani bu sorunun cevabını bir önceki konuda deincemiş olduk. Bunun yanı sıra DS kaydına erişimi olmayanlar için DLV (DNSSEC Lookaside Validation) kullanımı söz konusu olabilir. Ayrıca büyük zonlar için NSEC3 tipi kayıtlar da kullanılabilir. Yani başka yöntemler de bulunabilir. Biz DS kaydı ile HPKSK elde edimini gördük.



Artık elimizde "DNSKEY RRset", "KSK ile imzalanmış RRsig" ve "Hashlenmiş Public KSK" var. Şimdi bu üçü ile "Doğrulanmış PZSK" (Verified Public Zone-Signing Key) elde edimine bakalım.

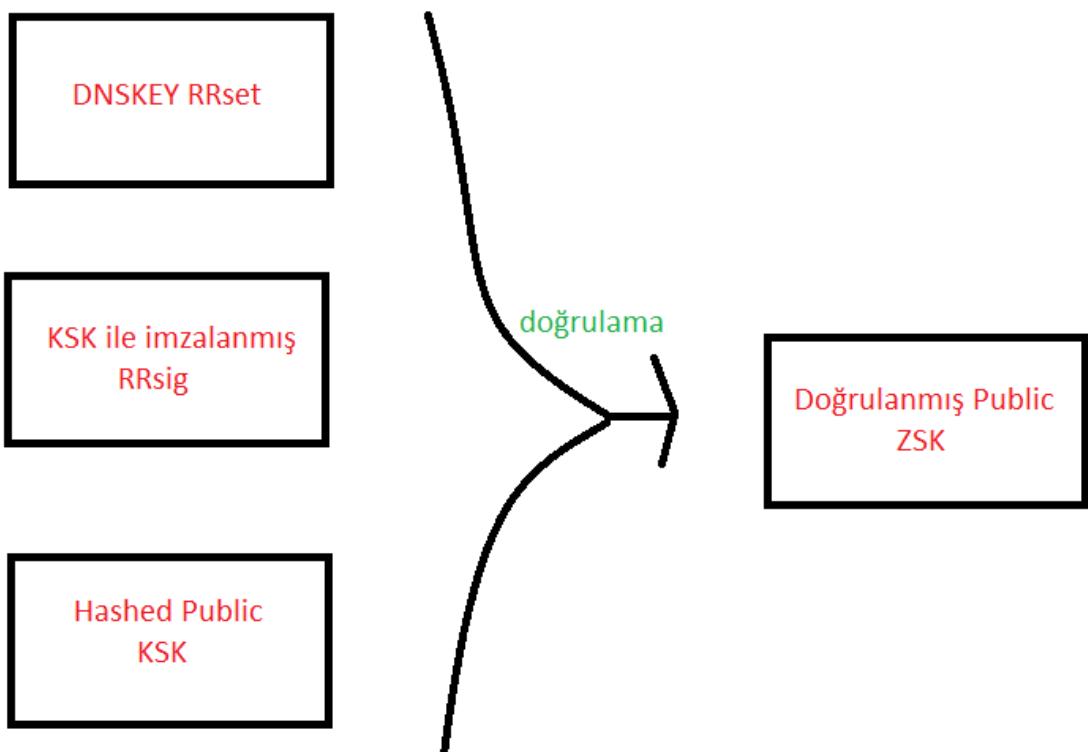
### Verified Public Zone-Signing Key

"Verified Public Zone-Signing Key" (Doğrulanmış Public Zone-Signing Key) terimi, bir alan adının imzalanmış ve doğrulanmış zone-signing key'inin (ZSK) kamu anahtarına karşılık gelir. Bu anahtar, alan adının (domain name'in) DNS kayıtlarının bütünlüğünü ve doğruluğunu doğrulamak için kullanılır.

Verified Public Zone-Signing Key'in Önemi:

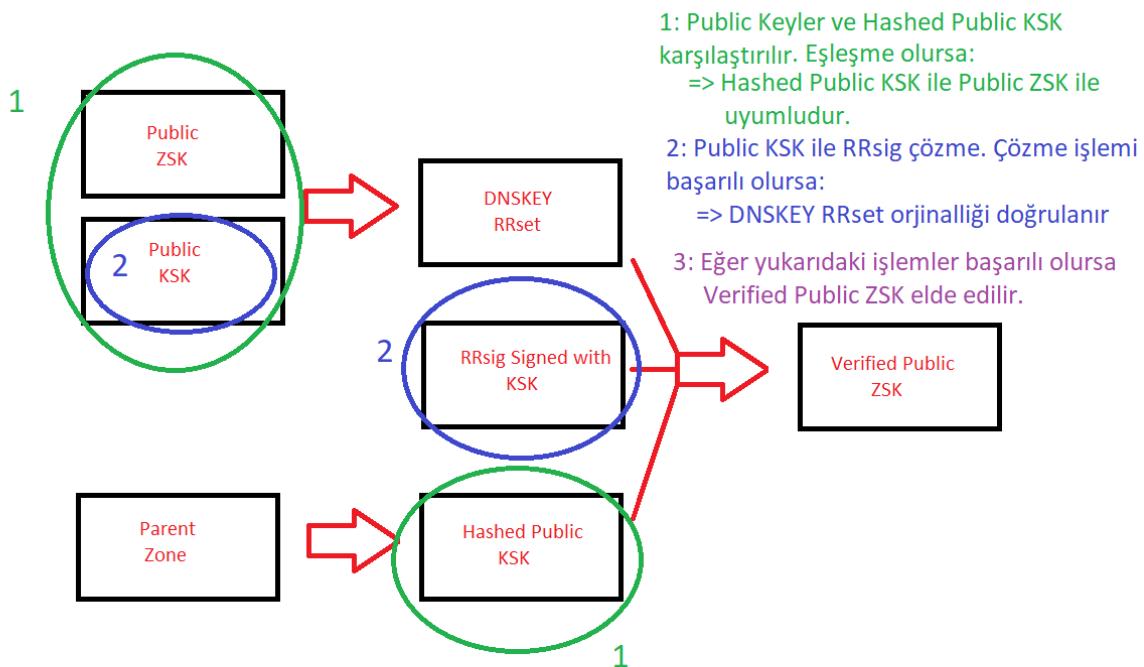
- **Sahteciliği Önler:** Verified Public Zone-Signing Key, bir saldırganın zone'a sahte kayıtlar eklemesini veya mevcut kayıtları değiştirmesini öner.
- **Veri Bütünlüğü Sağlar:** DNS kayıtlarının doğruluğunu ve bütünlüğünü garanti eder.
- **Güvenli İletişim Kurar:** İnternet üzerindeki iletişim güvenliğini artırır.

DNSKEY RRset, KSK ile imzalanmış RRsig ve Hashed Public KSK arasında belirli işlemler gerçekleşir. Bu işlemler amacı doğrulama gerçekleştirmektir. Doğrulama sonucunda "Doğrulanmış Public ZSK" elde edilir.

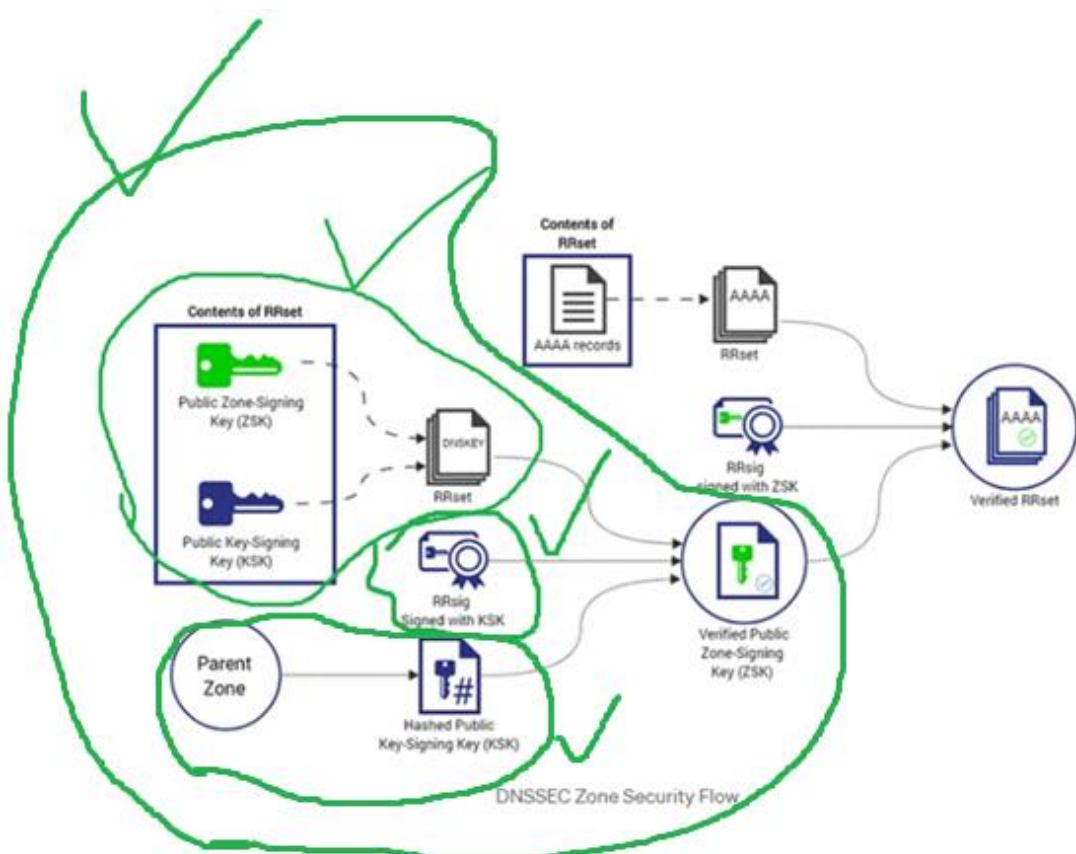


### Doğrulama adımları:

- DNSKEY RRset'indeki public key'i hashed public KSK ile karşılaştırın: Eşleşiyorsa, ZSK'nın parent zonun KSK'si ile uyumlu olduğu anlamına gelir.
- KSK ile imzalanmış RRSIG'i kullanarak DNSKEY RRset'inin sahte olmadığını doğrulayın: Bu, RRSIG'i KSK'nın public key'i ile çözerek yapılabilir.
- Doğrulama başarılı olursa, Verified Public Zone-Signing Key elde edilmiş demektir.



Göründüğü gibi Verified Public ZSK elde etmek için önceki adımlarda belirli işlemler gerçekleştirmemiz gereklidir. Örneğin Public ZSK ve Public KSK, Hashed public KSK ile karşılaştırılır. Eşleşme olması halinde Hashed Public KSK ile Public ZSK uyumlu olduğu anlaşılır. Bir diğer işlem Public KSK ile RRsig çözme işlemidir. Eğer çözme işlemi başarılı olursa DNSKEY RRset'in sahte olmadığı doğrulanır. Bu iki işlem başarılı olduğunda Verified Public ZSK elde ederiz.



Göründüğü üzere Verified Public ZSK'ya kadar oluşumu öğrendik. Şimdi sırada "ZSK ile imzalanmış RRsig" (RRsig Signed with ZSK) var. Hatırlarsak daha önce "KSK ile imzalanmış RRsig" (RRsig Signed with KSK) yapısına degenmişti. Bu ikisini karıştırmamak gerekiyor.

### ZSK ile imzalanmış RRsig (RRsig Signed with ZSK)

DNSSEC'de ZSK ile imzalanmış RRsig, Zone Signing Key (ZSK) ile imzalanmış bir Resource Record Signature (RRsig) kaydıdır. Bu kayıt, DNSSEC'nin önemli bir bileşenidir ve aşağıdaki işlevleri görür:

- Veri bütünlüğünü sağlar:** ZSK ile imzalanmış RRsig, DNS kayıtlarının yetkisiz taraflar tarafından değiştirilmemiğini veya bozulmadığını garanti eder. Bu, DNS'nin güvenilirliğini artırır.
- Sahetcilik ve kimlik avı saldırılardan korur:** ZSK ile imzalanmış RRsig, DNS sahteciliği ve kimlik avı saldırılardan tespit edilmesine ve önlenmesine yardımcı olur. Bu saldırılar, kullanıcıları sahte web sitelerine yönlendirmek veya hassas bilgilerini ele geçirmek için kullanılabilir.
- DNSSEC'nin güvenilirliğini ve işlevsellliğini artırır:** ZSK ile imzalanmış RRsig, DNSSEC'nin genel güvenilirliğini ve işlevsellliğini artırır. Bu, DNS'yi daha güvenli ve daha güvenilir bir hale getirir.

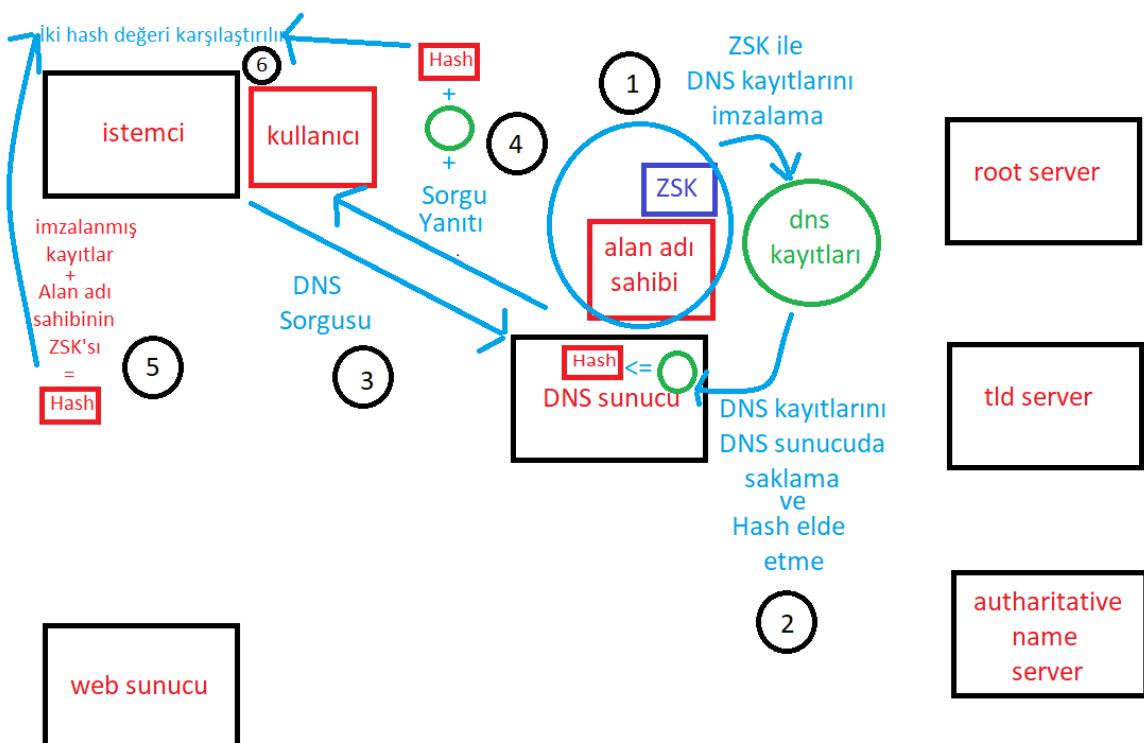
RRset Nedir?

RRset, aynı ad ve aynı DNS kayıt tipine sahip tüm kayıtların bir grubunu temsil eder ve DNSSEC'te verilerin doğruluğunu sağlamak için kullanılır.

RRsig Nedir?

RRsig kaydı, RRset için imza içerir. Bu imzada isim, sınıf ve tip gibi özellikler bulunur. RRsig kaydı, belirlenmiş RRset'i imzalar. Böylece RRset'in bütünlüğü ve kimliği doğrulanır.

ZSK ile imzalanmış RRsig (RRsig Signed with ZSK) Nasıl Çalışır? Kullanım Alanı Nedir?



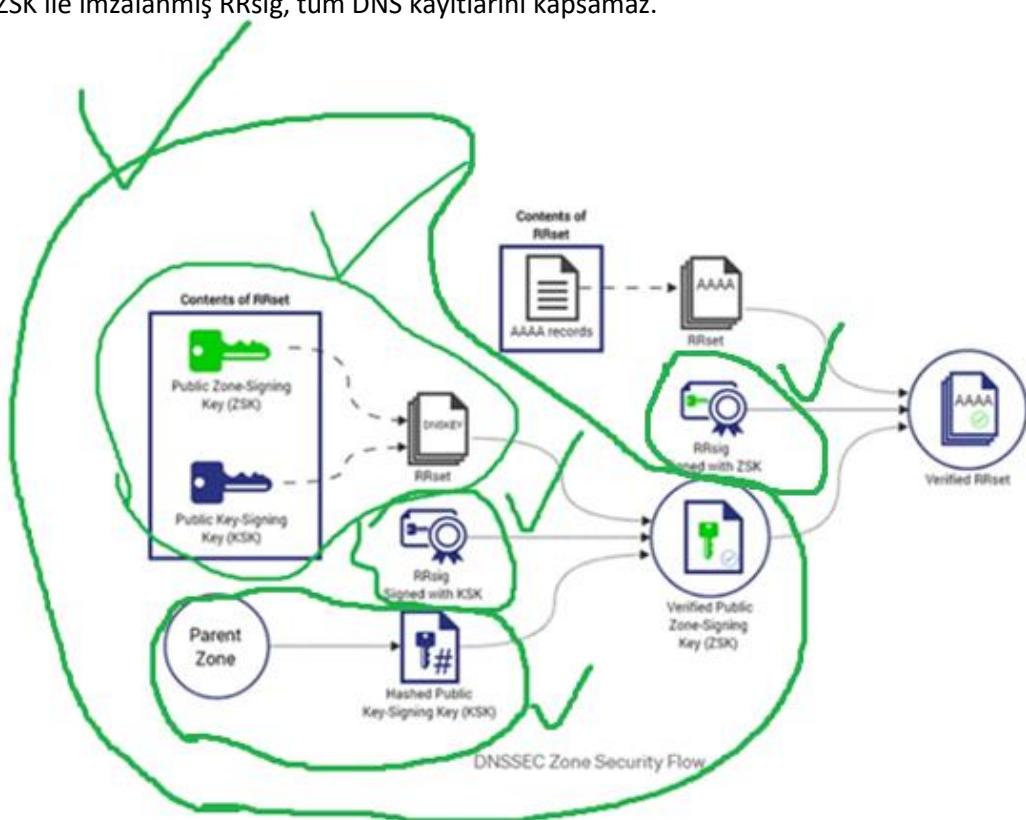
- Bir alan adı sahibi, alan adı sahibinin ZSK'si ile DNS kayıtlarını imzalar.
- İmzalanmış RRsig kayıtları, DNS sunucusunda saklanır. Bu kayıtlardan hash elde edilir.
- Bir kullanıcı DNS sunucusuna bir sorgu gönderir.
- Sunucu sorgu yanıt ile beraber, RRsig'leri ve hash değerini de gönderir.
- Sorgu cevabından dönen imzalanmış kayıtları ve alan adı sahibinin ZSK'nı alan istemci, bu ikisiyle bir hash oluşturur.
- Eğer istemci tarafında oluşturulan hash değeri ile DNS sunucudan dönen hash değeri karşılaştırımları sonucu eşitse doğrulama işlemi gerçekleşir.
- Hashler doğrulanırsa, imzalanmış kayıtların yetkisiz değişikliklere uğramadığını ve güvenilir olduğunu gösterir.

ZSK ile imzalanmış RRsig'in faydalari:

- Veri bütünlüğü ve güvenilirliği sağlar.
- Sahtecilik ve kimlik avı saldırılalarını önler.
- DNSSEC'nin güvenilirliğini ve işlevsellliğini artırır.

ZSK ile imzalanmış RRsig'in sınırlamaları:

- ZSK, bir alan adı sahibi tarafından kontrol edilir. Bu, ZSK'nin çalınması veya kaybolması durumunda DNS kayıtlarının güvenliğinin tehlikeye girebileceği anlamına gelir. Fakat istemcinin ZSK'yi alan adı sahibinden alması gerekiyor. Bu:
  - DNSSEC Look-Up
  - NSEC3 Look-Up
  - HTTPS
  - DS kaydı
  - Manuel
- Şekilde elde edilebilir.
- ZSK ile imzalanmış RRsig, tüm DNS kayıtlarını kapsamaz.



Böylece ZSK ile imzalanmış RRSig'e de dephinmiş olduk. Şimdi sırada AAAA RRset gibi, bir DNS kaydına ait RRset var.

### **AAAA gibi bir DNS kaydından RRset Oluşturma**

Resimde AAAA tipi DNS Kaydı ile RRset oluşturma örneği verilmiştir. DNSSEC güvenlik akışında AAAA tipi DNS Kaydı yerine A, CNAME, NS tipi DNS kayıtlar da kullanılabilir. Dediğimiz gibi biz AAAA üzerinden anlatımı gerçekleştireceğiz.

#### **AAAA Tipi DNS Kaydı**

AAAA kaydı, bir alan adını bir IPv6 adresine yönlendirmek için kullanılan bir DNS kayıt türüdür. IPv6, Internet Protokolü'nün (IP) altıncı sürümüdür ve IPv4'ten daha büyük adres alanı ve daha gelişmiş güvenlik özellikleri sunar.

Hatırlarsak A tipi DNS kaydı 32 bit uzunluktayken, AAAA tipi DNS kaydı 128 bit uzunluktadır. A tipi DNS Kaydı IPv4 adresleriyle ilişkiliyken, AAAA tipi DNS Kaydı IPv6 adresleriyle ilişkilidir. Daha detaylı bir şekilde IPv4 ve IPv6 konularında dephinmiştim.

#### **RRset**

RRset, açılımı Kaynak Kayıt Seti (Resource Record Set) olan bir terimdir. DNS (Domain Name System - Alan Adı Sistemi) içerisinde önemli bir kavramdır. Bir RRset, aynı alan adı için tek bir kayıt türüne ait tüm kayıt değerlerini içeren bir veri kümesidir.

Yani RRSet birden fazla DNS kaydını içeren bir kümedir. Aynı alan adı ve kayıt türü için tüm kayıt değerlerini içerir. Bu örnekte birden fazla AAAA tipi DNS Kaydının bir kümesi olarak karşımıza çıkıyor.

Ayrıca daha önce DNSKEY RRset'e dephinmiştim. DNSKEY RRset ile AAAA tipi RRset'lerin yapısı farklıdır.

- DNSKEY RRset: Public ZSK ve Public KSK'dan oluşur.
- AAAA RRset: AAAA tipi DNS kaydından oluşur.

#### **AAAA Tipi DNS Kaydından AAAA RRset Nasıl Oluşur?**

Bunun için bir DNS sağlayıcı kullanılabilir:

- DNS sağlayıcınızın arayüzüne giriş yapın.
- Alan adı seçin.
- "Yeni Kayıt" veya "Kayıt Oluştur" seçeneğine tıklayın.
- Kayıt türü olarak "AAAA" seçin.
- Kayıt değerlerini girin:
  - IPv6 adresi: Alan adını yönlendirmek istediğiniz IPv6 adresi.
  - TTL (Time to Live): DNS sunucularının bu kaydı ne kadar süreyle önbelleğe alacağını belirleyen değer.
- Kaydı kaydedin.

## AAAA RRset Nedir?

Özet olarak, bir AAAA RRset (Kaynak Kayıt Seti), **aynı alan adı için birden fazla IPv6 adresine yönlendiren AAAA kayıtlarının bir koleksiyonudur.**

AAAA kayıtları, bir alan adını IPv6 protokolü üzerinden erişilebilir hale getirir. AAAA RRset'i ise bu kayıtları organize eder.

AAAA RRset'in Bileşenleri:

- AAAA Kayıtları: Alan adını bir veya daha fazla IPv6 adresine yönlendirir.
- Alan Adı: RRset'in ilişkili olduğu alan adı.
- TTL (Time to Live - Yaşam Süresi): DNS sunucularının bir RRset'i ne kadar süreyle önbelleğe alacağını belirler.

AAAA RRset'in İşlevi:

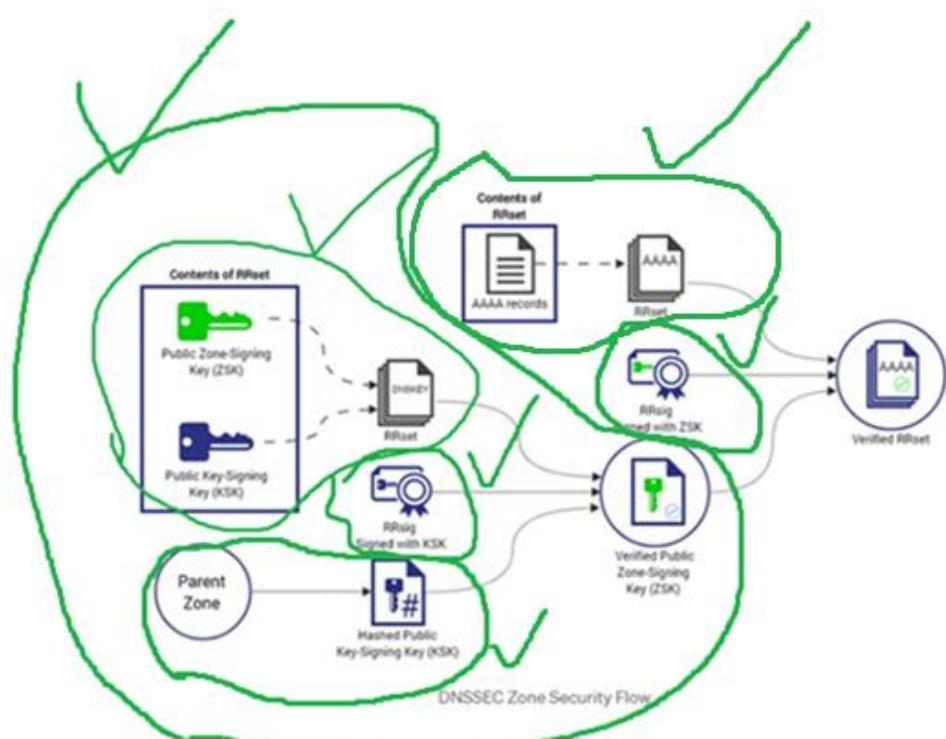
- Internet tarayıcıları ve diğer istemcilere, bir alan adına erişmek için kullanabilecek IPv6 adreslerini bildirir.
- IPv6 üzerinden web sitelerine, e-posta sunucularına veya diğer hizmetlere erişimi sağlar.

AAAA RRset Oluşturma Nedenleri:

- Bir web sitesini veya hizmeti hem IPv4 hem de IPv6 üzerinden erişilebilir hale getirmek.
- IPv6 desteği sunan sunucuları kullanmak.
- Yük dengeleme ve güvenlik gibi çeşitli avantajları kullanmak.
- Internet Protokolünün gelecekteki gelişimine uyum sağlamak.

AAAA RRset ile Arasında Farklar:

- AAAA Kaydı: Tek bir IPv6 adresi içerir.
- AAAA RRset: Aynı alan adı için birden fazla AAAA kaydı içerebilir



Bu adımda ise AAAA tipi DNS Kaydı ile RRset oluşturma örneğine deðindik. Dediðimiz gibi DNSSEC güvenlik akışında AAAA tipi DNS Kaydı yerine A, CNAME, NS tipi DNS kayıtları da kullanılabilir. Biz örneði AAAA tipi DNS Kaydı üzerinden gösterdik. (Tüm bu kayıtlara önceki sayfalarda deðindik **[DNS Kayıt Tipleri]**)

Şimdi son olarak eldeki tüm materyallerden doğrulanmış AAAA RRset (Verified AAAA RRset) elde edelim.

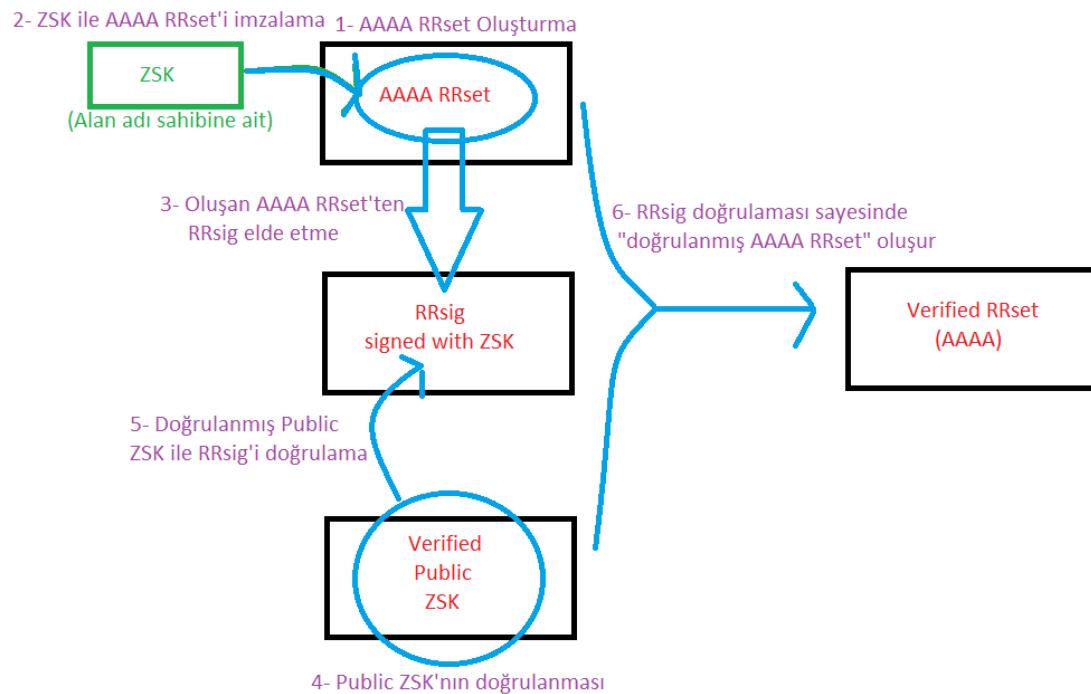
### Verified AAAA RRset

AAAA RRset, Alan adı için bir veya birden fazla IPv6 adresi içeren bir kayıt setidir. Doðrulanmış AAAA RRset'in ise, birden fazla özelliði vardır:

- Alan adının doğru IPv6 adresine yönlendirilmesini sağlar.
- DNSSEC'nin güvenliğini ve işlevsellliğini sağlar.
- Alan adı kayıtlarının sahteciliðini ve değiştirilmesini önler.

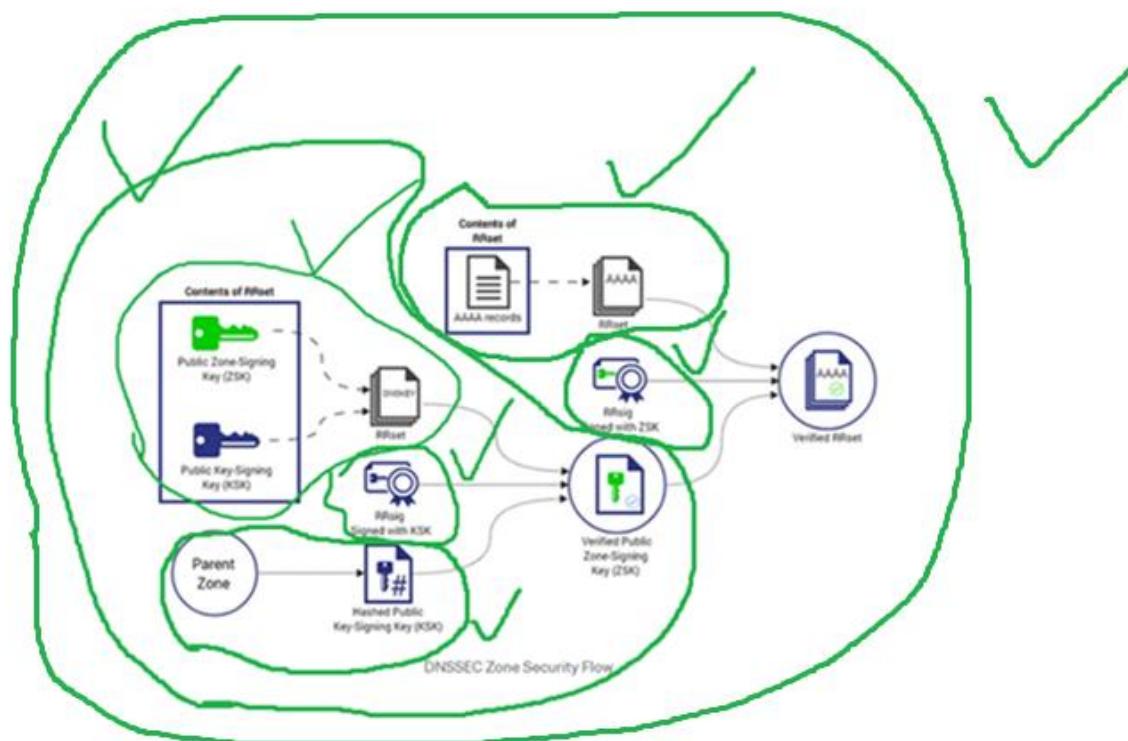
Yani temelde, alan adının IPv6'ya güvenli bir şekilde dönüþtürülmesini sağlar.

### Doðrulanmış AAAA RRset Oluşturma



1. AAAA RRset Oluşturma:
  - Alan adı için bir veya birden fazla IPv6 adresi içeren bir AAAA RRset oluşturun.
  - Kayıt türü AAAA, TTL değeri ve diğer gerekli bilgileri ayarlayın.
2. ZSK ile RRset İmzalama:
  - Alan adı sahibinin kontrolünde olan ZSK ile AAAA RRset'i imzalayın.
  - Bu işlem, RRset'in değiştirilmediğini ve DNSSEC'ye uyduğunu garanti eder.

3. İmzalanmış RRset (RRsig) Oluşturma:
  - ZSK ile imzalanmış AAAA RRset'ten bir RRsиг oluşturun.
  - RRsиг, DNS sunucularının RRset'in doğruluğunu doğrulamasına olanak tanır.
4. Public ZSK'nin Doğrulanması:
  - Güvenilir bir üçüncü taraf tarafından kontrol edilen KSK ile public ZSK'yi doğrulayın.
  - Bu işlem, public ZSK'nin güvenilir olduğunu ve alan adı sahibinin kontrolünde olduğunu garanti eder.
5. RRsig'in Doğrulanması:
  - Doğrulanmış public ZSK ile RRsig'i doğrulayın.
  - Bu işlem, RRset'in ZSK tarafından imzalandığını ve değiştirilmediğini garanti eder.
6. Doğrulanmış RRset'in Kullanımı:
  - RRsig, DNS sunucularının RRset'in değiştirilmediğini ve ZSK tarafından imzalandığını doğrulamasına olanak tanır. Doğrulanmış RRsig sayesinde, doğrulanmış AAAA RRset oluşur.
  - Doğrulanmış AAAA RRset'i DNS sunucularında yaylayın.
  - Bu işlem, alan adının doğru IPv6 adresine yönlendirilmesini garanti eder.



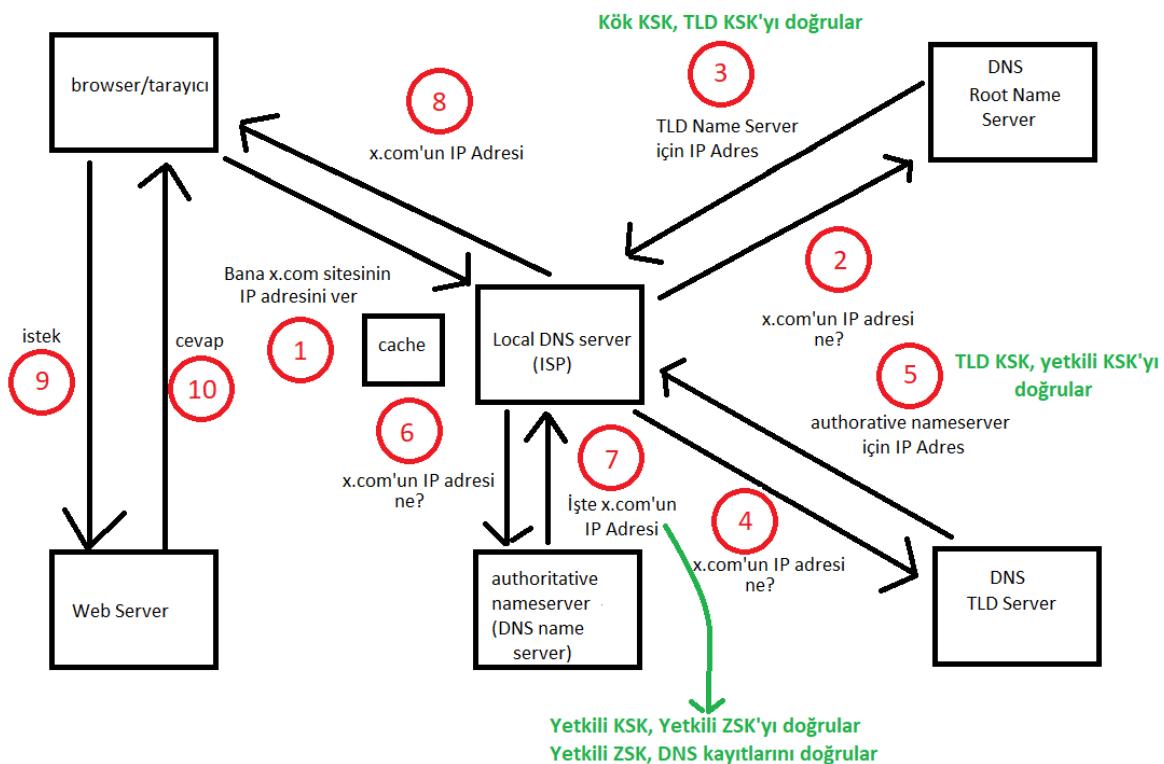
En sonunda DNSSEC güvenlik akışını anladık. Yani alan adının IPv4 ve IPv6'ya güvenli bir şekilde dönüştürülmesini gördük. Veya tam tersi de mümkün olabilir. Yani sadece A ve AAAA tipi DNS kayıtları değil de herhangi bir DNS kayıt tipinin güvenli bir şekilde işlevini yapmasını gördük. Biz bu işlemleri AAAA tipi DNS kaydı üzerinden gerçekleştirdik.

Şimdi sırada öğrenciklerimizi (adımları) DNS sorgusu üzerinde görmek var. Böylece DNSSEC'in çalışma mantığını iyice anlayacağız.

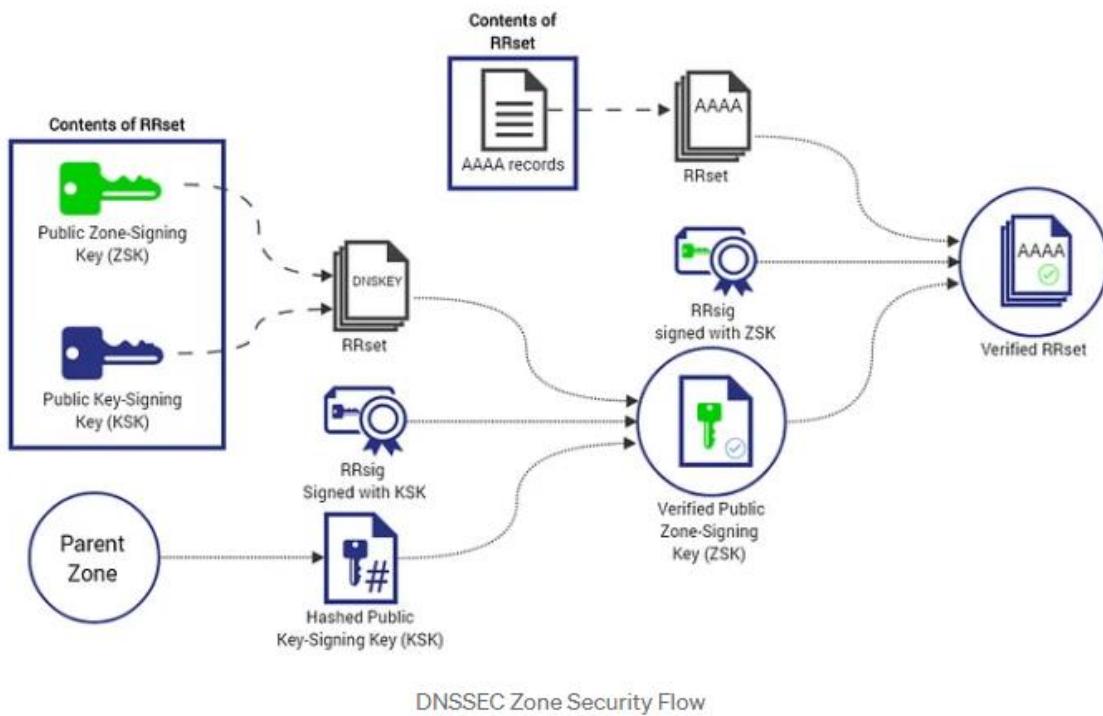
Bu mantığı öncelikle özet olarak açıklayalım:

1. Kullanıcı DNS sorgusu başlatır.
2. Çözümleyici (dns resolver), kök sunucusuna (root server) sorgu gönderir ve TLD sunucusunun adresini alır.
  - a. Kök KSK, TLD KSK'yi doğrular.
3. Çözümleyici, TLD sunucusuna sorgu gönderir ve yetkili ad sunucusunun adresini alır.
  - a. TLD KSK, yetkili KSK'yi doğrular.
4. Çözümleyici, yetkili ad sunucusuna sorgu gönderir ve DNS kaydını alır.
  - a. Yetkili KSK, yetkili ZSK'yi doğrular.
  - b. Yetkili ZSK, DNS kaydının imzasını doğrular.
5. Çözümleyici, doğrulanmış DNS kaydını kullanıcıya sunar. Kullanıcı uygulaması (örneğin, tarayıcı), bu güvenilir DNS kaydını kullanarak adıyla ilgili işlemeye devam eder (örneğin, web sitesine bağlanır).

Aşağıdaki resimde yukarıdaki adımlar anlatılmıştır.



Uzun halini açıklayacak olursak aşağıda gösterilen resimdeki DNSSEC çalışma prensibindeki her bir adım; root server, tld server ve autharitative name server için gerçekleşir diyebiliriz. Böylece root server aşamasında tld server için yetkilendirmeler gerçekleşir. Sonrasında tld server aşamasında autharitative name server için yetkilendirmeler gerçekleşir. Daha sonra autharitative name server aşamasında DNS kaydı için yetkilendirme gerçekleşir. Böylece yetkilendirme başarılı olur ve DNS sorusu başarılı hale gelir.



Detailed hali inceleyecek olursak:

### 1. Kök Sunucusu (Root Server)

Kök sunucuları, DNSSEC için en üst düzeyde bulunur ve TLD sunucularının anahtarlarını imzalar.

- **Kök Bölgesi (Root Zone):**
  - KSK ve ZSK Üretimi: Kök sunucusu, kendi KSK ve ZSK anahtarlarını üretir.
  - RRset İmzalama: Kök ZSK, kök bölgesindeki DNS kayıtlarını (RRset) imzalar.
  - KSK ile İmza: Kök KSK, kök ZSK'nin imzasını (RRSIG) üretir.
- **RRset ve RRSIG Yayını:**
  - Kök sunucusu, RRset ve ZSK tarafından imzalanmış RRSIG kaydını yayınlar.
  - Kök KSK, kök ZSK'nin doğruluğunu doğrular.

### 2. TLD Sunucusu (Top-Level Domain Server)

TLD sunucuları, kök sunucularından aldığı anahtarlarla kendi bölgelerini imzalarlar.

- **TLD Bölgesi (TLD Zone):**
  - KSK ve ZSK Üretimi: TLD sunucusu, kendi KSK ve ZSK anahtarlarını üretir.
  - RRset İmzalama: TLD ZSK, TLD bölgesindeki DNS kayıtlarını (RRset) imzalar.
  - KSK ile İmza: TLD KSK, TLD ZSK'nin imzasını (RRSIG) üretir.
  - Kök Sunucusundan Doğrulama: TLD KSK, kök sunucusunun KSK'si tarafından doğrulanır.

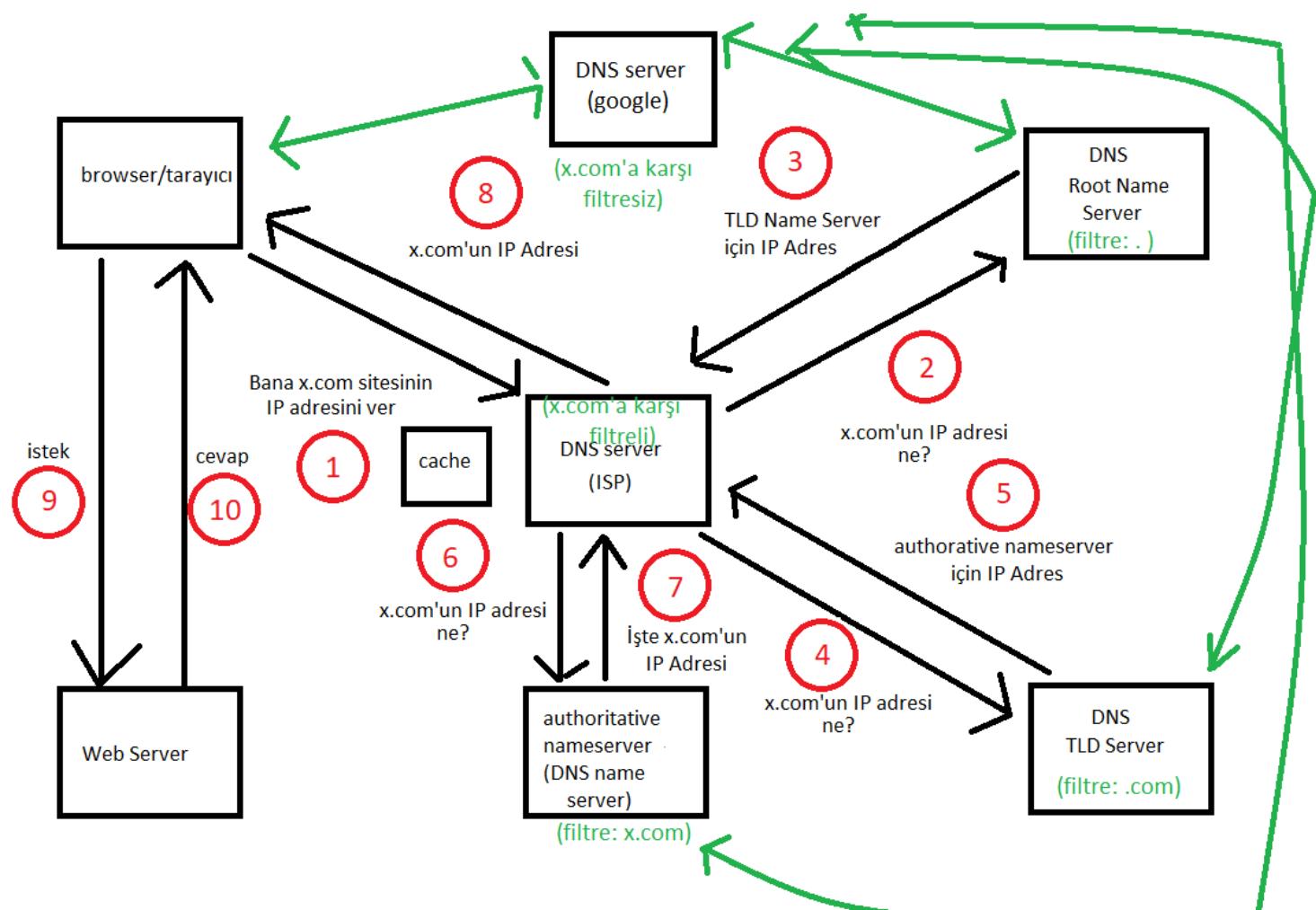
- **RRset ve RRSIG Yayınlama:**
  - TLD sunucusu, RRset ve ZSK tarafından imzalanmış RRSIG kaydını yayınlar.
  - TLD KSK, TLD ZSK'nin doğruluğunu doğrular.

### 3. Yetkili Ad Sunucusu (Authoritative Name Server)

Yetkili ad sunucuları, TLD sunucularından aldığıları anahtarlarla kendi bölgelerini imzalarlar.

- **Yetkili Bölge (Authoritative Zone):**
  - KSK ve ZSK Üretimi: Yetkili ad sunucusu, kendi KSK ve ZSK anahtarlarını üretir.
  - RRset İmzalama: Yetkili ZSK, yetkili bölgedeki DNS kayıtlarını (RRset) imzalar.
  - KSK ile İmza: Yetkili KSK, yetkili ZSK'nin imzasını (RRSIG) üretir.
  - TLD Sunucusundan Doğrulama: Yetkili KSK, TLD sunucusunun KSK'si tarafından doğrulanır.
- **RRset ve RRSIG Yayınlama:**
  - Yetkili ad sunucusu, RRset ve ZSK tarafından imzalanmış RRSIG kaydını yayınlar.
  - Yetkili KSK, yetkili ZSK'nin doğruluğunu doğrular.

### DNS IP Adresi Değiştirmek



Normalde DNS çalışma mantığında IP adrese ihtiyacımız olacak. Bu adresi DNS sorgusuyla elde ediyoruz demiştik. Sorgu ilk önce DNS Server'a oradan DNS Root Name Server'a gidiyor. DNS Root Name Server bize TLD server için gereken bilgileri sunuyor. Daha sonra TLD Server'a gidiyoruz. Burada ise bize Autharitive Name Server için gerekli bilgiler veriliyor. En sonunda Autharitive Name Server bize Domain adının IP adresini döndürüyor. Dönüşen adres tarayıcıya gidiyor ve web sunucuya IP adres ile erişebiliyoruz.

DNS IP Adresi değiştirmek bize yasaklı sitelere girme, güvenlik ve hız gibi etkenler sağlıyor. Genel kullanım amacı "abc.com" gibi yasaklı sitelere girmek için kullanılabilir. Burada, DNS Sunucu yani ISP (Internet Service Provider) x.com'a karşı filtreli olabilir. Eğer Root Name Server'da filtre olursa ., TLD Server'da filtre olursa .com. veya autharitive name server'da filtre olursa x.com. seviyesinde yasaklama gelir.

Bu yasaklardan kurtulmak için DNS Sunucumuzun IP adresini değiştirebiliriz (örnek: google). Google DNS Sunucu ise x.com'a karşı bir filtreleme sunmadığından dolayı, bu siteye erişim hakkına sahip oluruz.

Güvenlik anlamında ise DNS Sunucu güvenilir bir sunucu olursa (örneğin kendimizin yapılandırdığı güvenilir bir sunucu) bize güvenlik sağlayabilir. Ancak tam aksi durum da söz konusu olabilir. Çünkü DNS Sunucu güvenilir olmayıpabilir.

DNS Sunucunun IP adresini değiştirirken çok dikkatli olmak gereklidir.

## DNS Tünelleme

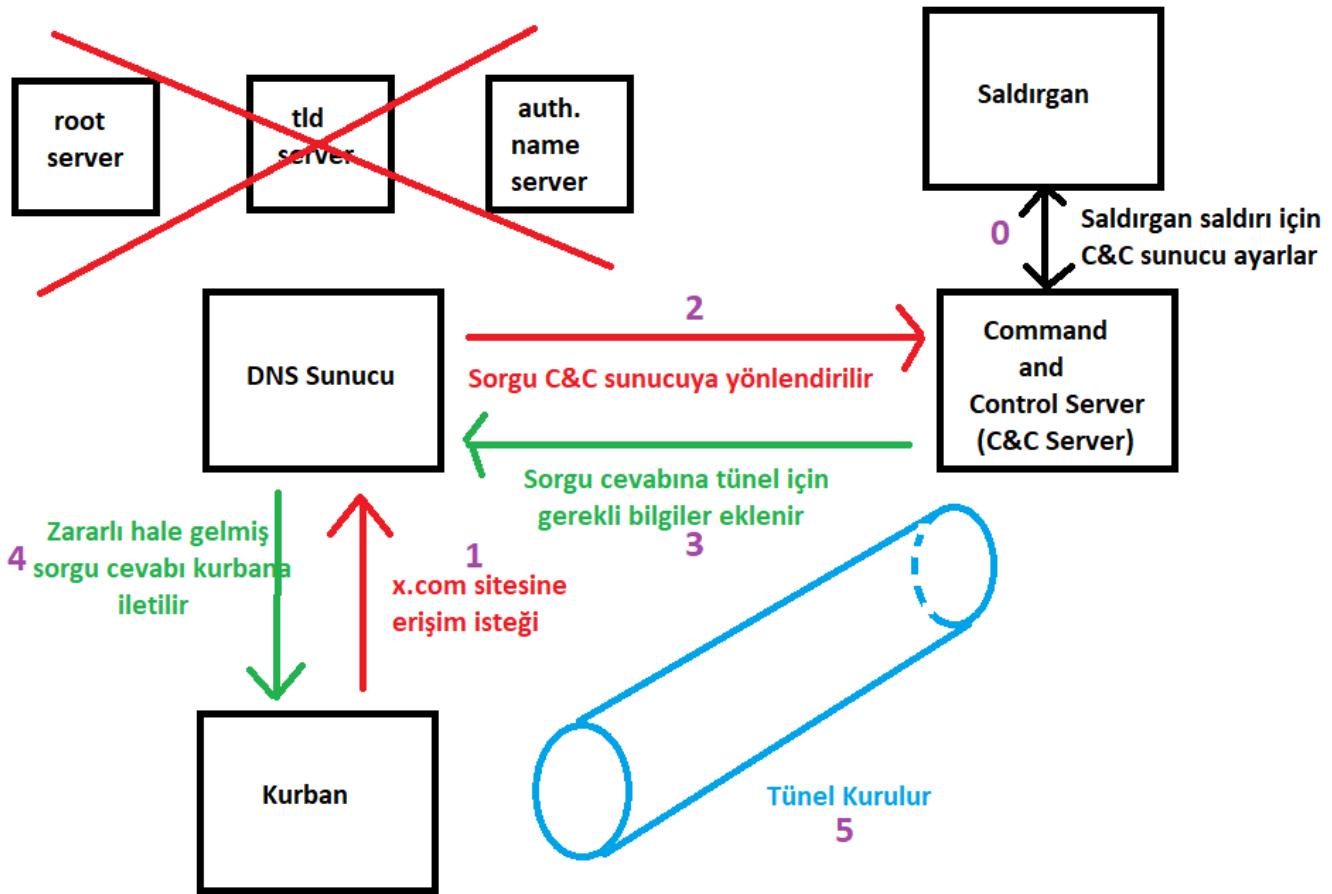
### DNS Tünelleme Nedir?

DNS Tünelleme, saldırganların DNS sorgularını kötü amaçlı yazılımları ve diğer verileri gizlice taşımak için kullandığı bir tekniktir. Bu yöntemde, saldırganlar DNS protokolünün normal işleyişini kullanarak, yasaklı veya izlenen protokollere (SSH, RDP, HTTP gibi) tünel açarlar.

### DNS Tünelleme Neden Kullanılır?

- **Güvenlik Kısıtlamalarının Aşılması:** Birçok kullanıcı, güvenlik duvarları veya ağ kısıtlamalarını aşmak için DNS tünellemeyi kullanır.
- **Veri Kaçakçılığı:** DNS tünelleme, hassas verilerin gizlice dışarı sızdırılması için kullanılabilir.
- **Ağ İzleme:** DNS tünelleme, bir ağın izlenmesi ve analiz edilmesi amacıyla da kullanılabilir.

## DNS Tünelleme Nasıl Çalışır?



DNS tünelleme saldırısı sayesinde kurban ile saldırgan arasında iletişimini sağlanacağı bir tünel oluşturulur. Adım adım çalışma prensibine bakacak olursak:

- İlk önce saldırgan bir Command and control (C&C) sunucusu ayarlar. Böylece saldırı için gerekli olan sunucu kurulur. Bu sunucu x.com için ayarlanır. DNS tünelleme saldırılarda saldırgan, genellikle kendi oluşturduğu alan adlarını kullanır (x.com).
- Daha sonra kurban x.com adlı siteye erişmek ister.
- DNS sunucuya giden sorgu isteği normalde root server, tld server ve authoritative name servera yönlendirilir. Ancak sorgu bu sunuculara gitmek yerine C&C sunucusuna iletılır. Bu birden fazla yöntemle gerçekleştirilebilir:
  - Saldırgan, kendi kontrolünde bir alan adı (örneğin, x.com) kaydeder. Saldırgan, bu alan adının authoritative DNS kayıtlarını kendi C&C sunucusuna yönlendirecek şekilde yapılandırabilir. Böylece kurban cihazı, x.com gibi bir alan adına DNS isteği gönderdiğinde, bu istek root server ve TLD server üzerinden geçerek saldırganın C&C sunucusuna ulaşır.
  - DNS Cache Poisoning saldırısı ile DNS önbelleği zehirlenerek yanlış DNS kayıtları oluşturur. Bu, meşru bir alan adı için yapılan sorguların saldırganın C&C sunucusuna yönlendirilmesini sağlar. Sorgu ro
- Cevap olarak gerekli bilgilerin yanında tünel açmak için bilgiler de eklenir.
- Zararlı hale gelen sorgu cevabı DNS sunucudan kurbana iletılır.
- Sonunda kurban ve saldırgan arasında bir tünel oluşur. Böylece bu ikili arasında bilgi alışverişi açık hale gelir.

## DNS ile İlgili Terimler

### DNS kesintisi (DNS outage) nedir ve neden oluşur?

DNS kesintisi, Internet'in temel altyapısı olan DNS sisteminin (Alan Adı Sistemi) çalışmayı durdurması veya erişilemez hale gelmesidir. Bu durum, kullanıcıların web sitelerine, e-postalara ve diğer çevrimiçi hizmetlere erişmesini engeller.

Nedenler:

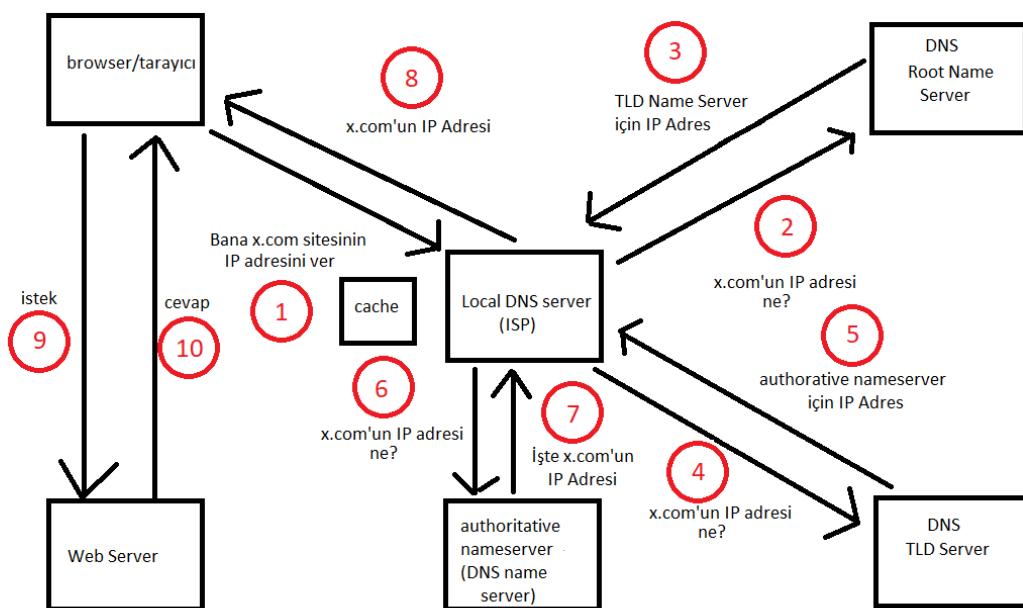
- **DDoS Saldırıları (Distributed Denial of Service):**
  - DNS sunucuları, büyük ölçekli DDoS saldırılarına maruz kaldığında, aşırı yüklenir ve hizmet veremez hale gelir. Bu tür saldırılar, sunucunun kapasitesini aşan sahte trafik göndererek sunucuyu devre dışı bırakır.
- **Teknik Arızalar:**
  - Donanım veya yazılım hataları, sunucuların çalışmasını durdurabilir. Özellikle yedekleme sistemlerinin olmaması durumunda, bu tür arızalar ciddi kesintilere neden olabilir.
- **Konfigürasyon Hataları:**
  - Yanlış yapılandırmalar, DNS sunucularının düzgün çalışmamasına neden olabilir. Bu, yanlış DNS kayıtları veya yapılandırma dosyalarındaki hatalar sonucu oluşabilir.
- **Internet Kesintileri:**
  - DNS sunucuları, internet bağlantısının kesilmesi veya yavaşlaması durumunda hizmet veremez hale gelebilir. Bu, fiber optik kabloların hasar görmesi veya servis sağlayıcılar arasındaki bağlantı sorunlarından kaynaklanabilir.
- **Yazılım Güncellemeleri:**
  - DNS sunucularındaki yazılım güncellemeleri sırasında veya sonrasında oluşabilecek uyumsuzluklar veya hatalar, kesintilere neden olabilir.
- **Siber Saldırılar:**
  - DNS sunucularına yönelik hedefli saldırılar, veritabanlarını bozarak veya ele geçirerek kesintilere yol açabilir.

### DNS çözümlemesi (DNS resolution) nedir ve nasıl yapılır?

DNS çözümlemesi (DNS resolution), bir alan adını (örneğin, www.example.com) bir IP adresine (örneğin, 192.0.2.1) dönüştürme sürecidir. Bu işlem, internet kullanıcılarının kolayca hatırlayabilecekleri alan adlarını kullanarak web sitelerine erişmelerini sağlar. Nasıl gerçekleştir sorusuna daha önce de geldik. Burada da deşimelim:

1. **Kullanıcının DNS Sorgusu Göndermesi:** Bir kullanıcı tarayıcısına bir web sitesi adresi girdiğinde, tarayıcı bu alan adının IP adresini öğrenmek için bir DNS sorgusu başlatır.
2. **Yerel DNS Önbelleği Kontrolü:** İlk olarak, bilgisayar yerel DNS önbelleğinde bu alan adına ait bir kayıt olup olmadığını kontrol eder. Eğer varsa, IP adresi buradan alınır ve süreç tamamlanır. Yoksa sorgu devam eder.
3. **DNS Resolver (Çözümleyici) Sunucusuna Sorgu Gönderme:** Yerel DNS önbelleğinde kayıt bulunamazsa, sorgu kullanıcının internet servis sağlayıcısının (ISP) DNS resolver sunucusuna gönderilir.

4. **Root DNS Sunucularına Sorgu:** Resolver sunucusu, sorguyu root DNS sunucularına gönderir. Root sunucular, alan adının hangi üst düzey etki alanı (TLD) sunucusuna ait olduğunu belirler (örneğin, .com, .org, .net).
5. **TLD Sunucularına Sorgu:** Root sunucuları, sorguyu ilgili TLD sunucusuna yönlendirir. TLD sunucusu, bu alan adının hangi yetkili ad sunucusunda bulunduğu bilir.
6. **Yetkili Ad Sunucusuna Sorgu:** TLD sunucuları, sorguyu ilgili yetkili ad sunucusuna yönlendirir. Bu sunucusu, alan adına karşılık gelen IP adresini içerir.
7. **IP Adresinin Geri Döndürülmesi:** Yetkili ad sunucusu, alan adının IP adresini resolver sunucusuna geri döndürür. Resolver sunucusu, bu bilgiyi yerel DNS önbelleğine kaydeder ve IP adresini kullanıcının bilgisayarına gönderir.
8. **Web Sitesine Erişim:** Kullanıcının tarayıcısı, aldığı IP adresini kullanarak hedef web sunucusuna bir bağlantı kurar ve web sitesi açılır.



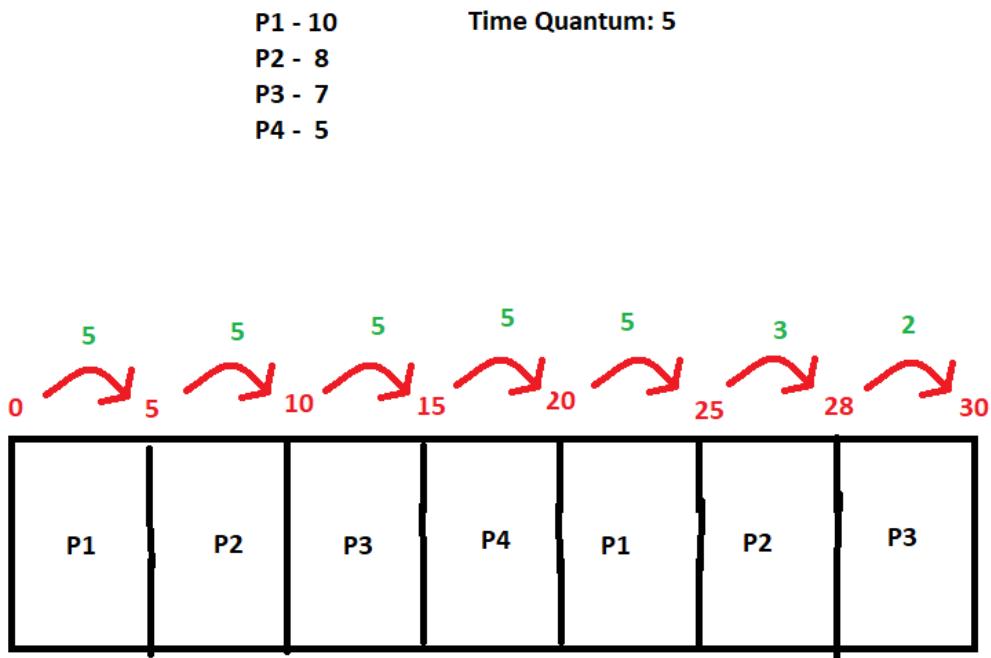
## Round-Robin Nedir

Round-Robin, çeşitli bilgisayar bilimleri ve ağ teknolojileri alanlarında kullanılan bir yük dengeleme ve zamanlama algoritmasıdır. Bu yöntemin temel prensibi, her bir görevin veya isteğin sırayla, belirli bir döngü içinde eşit zaman dilimlerinde işlenmesidir. Round-Robin algoritması, adil ve dengeli kaynak dağıtımını sağlamayı amaçlar.

### DNS Yük Dengeleme:

Round-Robin, DNS (Domain Name System) sunucularında da kullanılır. Bir alan adı için birden fazla IP adresi varsa, DNS sunucusu bu IP adreslerini Round-Robin yöntemiyle döndürebilir. Bu, web trafiğini birden fazla sunucu arasında dağıtarak, tek bir sunucunun aşırı yüklenmesini önler.

### Örnek (Round Robin Algoritması):



Yukarıdaki örnekte bir işlem için ayrılan maksimum süre 5 birim olarak verilmiştir. P1 10 birim, P2 8 birim, P3 7 birim ve p4 5 birimidir. Sırasıyla p1, p2, p3, p4 işlemleri gerçekleşecektir.

1. P1 için 5 birim (P1 Kalan: 5 [10-5])
2. P2 için 5 birim (P2 Kalan: 3 [8-5])
3. P3 için 5 birim (P3 Kalan: 2 [7-5])
4. P4 için 5 birim (P4 Kalan: 0 [5-5])
5. P1 için 5 birim (P1 Kalan: 0 [5-5])
6. P2 için 3 birim (P2 Kalan: 0 [3-3])
7. P2 için 2 birim (P3 Kalan: 0 [2-2])

İşlemlerin gerçekleşme sırası Round Robin algoritmasına göre yukarıdaki şekildedir. Round Robin işlemcilerde zaman paylaşımında kullanılabilir. Bunun yanında DNS üzerinde olduğu gibi başka örneklerde de bulunur. DNS üzerinde inceleyelim.

### Round Robin DNS (DNS üzerinde)

Round-Robin DNS, birden fazla web sunucusuna sahip web siteleri için yük dengeleme ve hata toleransı sağlayan bir tekniktir. DNS sunucuları, alan adı sorgularına yanıt verirken, aynı alan adına ait birden fazla IP adresi varsa, Round-Robin algoritmasını kullanarak her seferinde farklı bir IP adresi döndürür.

### İşleyiş Adımları

- Alan Adı Sorgusu: Kullanıcı, bir web sitesinin alan adını tarayıcısına girer. Bu, DNS sunucusuna bir alan adı sorgusu gönderir.
- Sorgu Devamı: Sorgu root server ve tld serverda gidip gelir.

- Yetkili İsim Sunucusu (authoritative name server): Sorguda belirtilen alan adına ait kayıtları bulmak için yetkili isim sunucusu kullanılır. Yetkili isim sunucusu, aynı alan adına ait birden fazla IP adresi içeriyorsa, Round-Robin algoritmasını kullanır.
- Round-Robin Algoritması: Round-Robin algoritması, IP adreslerini bir kuyrukta tutar ve her seferinde kuyruğun en başındaki IP adresini döndürür. Kuyruğun sonuna geldiğinde ise tekrar başa döner.  
Örneğin bir web site düşünelim. Bu web site birden fazla IP adrese sahip olsun. Bu adreslerin her birisi için sırasıyla Round-Robin algoritması kullanılır.
- Web Sunucusuna Bağlantı: Kullanıcı tarayıcısı, yetkili isim sunucusu tarafından döndürülen IP adresine sahip web sunucusuna bağlantı kurar.

#### **Round Robin Avantajları:**

- Adil Dağıtım: Her görevin eşit zaman dilimi olması sağlanır.
- Kolay Uygulama: Algoritma basit ve anlaşılır.
- Öngörülebilir Performans: İş yükleri dengeli bir şekilde dağıtılmıştır.

#### **Round Robin Dezavantajları:**

- Önceliklendirme Eksikliği: Görevlerin önceliği yoktur, bu da kritik görevlerin gecikmesine yol açabilir.
- Sabit Zaman Dilimi: Zaman dilimi uygun seçilmezse, performans sorunları ortaya çıkabilir.

## **Netmask Ordering**

Bir ağda, netmask, IP adresinin hangi kısmının ağ adresini ve hangi kısmının ana bilgisayar adresini belirlediğini tanımlar. Netmasklar, ikilik noktalı gösterimle (örneğin 255.255.255.0) veya ondalık gösterimle (örneğin /24) ifade edilir.

Netmask ordering, DNS sunucularının IP adreslerini bir DNS yanıtında belirli bir düzende sunması yöntemidir. Bu yöntem, istemcinin IP adresine göre en yakın olanı öne çıkararak ağ performansını ve erişim hızını artırmayı amaçlar. Netmask ordering, özellikle büyük ağlarda ve coğrafi olarak dağılmış sunuculara sahip sistemlerde oldukça faydalıdır.

Netmasklar genellikle üç sınıfa ayrılır:

- **A Sınıfı Netmasklar:** Büyük ağlar için kullanılırlar. Örneğin, 255.0.0.0 netmaskı, 16 milyon host barındırabilen bir ağ oluşturur.
- **B Sınıfı Netmasklar:** Orta ölçekli ağlar için kullanılırlar. Örneğin, 255.255.0.0 netmaskı, 65.536 host barındırabilen bir ağ oluşturur.
- **C Sınıfı Netmasklar:** Küçük ağlar için kullanılırlar. Örneğin, 255.255.255.0 netmaskı, 254 host barındırabilen bir ağ oluşturur.

Netmask Ordering Çalışma Prensibi:

- İstemci, DNS sunucusuna bir alan adı sorgusu gönderir.
- DNS sunucusu, alan adına bağlı tüm IP adreslerini arar.
- Eğer birden fazla IP adresi varsa, DNS sunucusu istemcinin IP adresini ve netmaskını kullanarak en uygun IP adresini seçer.

- DNS sunucusu, seçilen IP adresini istemciye gönderir.

Subnetmask'ten daha önce bahsetmiştik. Çalışma prensibinde subnet mask yatıyor. Bir örnek üzerinde bakalım:

- İstemci IP adresi: 192.168.1.10
- DNS yanıtındaki IP adresleri: 192.168.1.20, 192.168.2.30, 192.168.3.40
- Subnetmask: 255.255.255.0 (C tipi)

Karşılaştırma yaparken IP adres ile subnetmask and işlemine tutulur. Eğer karşılaştırılan iki IP adresin subnet'leri aynı ise (Subnet=IP adres **AND** subnet mask) önceliklendirilir. Örnek üzerinden bakalım:

- 192.168.1.10 ve 192.168.1.20 -> Aynı alt ağ (192.168.1.0)
- 192.168.1.10 ve 192.168.2.30 -> Farklı alt ağlar
- 192.168.1.10 ve 192.168.3.40 -> Farklı alt ağlar

Bu karşılaştırma sonucunda, 192.168.1.20 IP adresi istemciye en yakın olan IP adresidir ve listenin başına yerleştirilir.

### **Netmask Ordering ve Round Robin Arasındaki İlişki**

Netmask ordering ve round-robin DNS, her ikisi de DNS sunucularının istemcilere IP adresleri sunma yöntemleridir, ancak farklı amaçlara hizmet ederler ve farklı şekillerde çalışırlar. İkisi birlikte çalışabilir (ikisi birbirinden bağımsızdır). Böylece ağ performansı, güvenlik ve yük dengeleme konusunda katkı sağlayabilir. Örnek üzerinde görelim:

#### **DNS Sunucusu Konfigürasyonu:**

- Alan adı: example.com
- IP adresleri: 192.168.1.10, 192.168.2.20, 192.168.3.30
- Hem netmask ordering hem de round-robin etkin

#### **İstemciler:**

- İstemci A IP adresi: 192.168.1.5
- İstemci B IP adresi: 192.168.2.15

#### **İlk Sorgu:**

- İstemci A example.com için sorgu gönderir.
- Round-Robin sırası: 192.168.1.10, 192.168.2.20, 192.168.3.30
- Netmask Ordering uygulanır: İstemci A'nın IP adresi 192.168.1.5 olduğu için 192.168.1.10 en yakın IP adresi olarak öne çıkar.
- Yanıtı Sırası: **192.168.1.10, 192.168.2.20, 192.168.3.30**

#### **İkinci Sorgu:**

- İstemci B example.com için sorgu gönderir.
- Round-Robin sırası: 192.168.2.20, 192.168.3.30, 192.168.1.10 (Round-robin sırasıyla değiştirilmiştir.)
- Netmask Ordering uygulanır: İstemci B'nin IP adresi 192.168.2.15 olduğu için 192.168.2.20 en yakın IP adresi olarak öne çıkar.
- Yanıtı Sırası: **192.168.2.20, 192.168.3.30, 192.168.1.10**

### Üçüncü Soru:

- İstemci A example.com için tekrar soru gönderir.
- Round-Robin sırası: 192.168.3.30, 192.168.1.10, 192.168.2.20 (Round-robin sırasıyla değiştirilmiştir.)
- Netmask Ordering uygulanır: İstemci A'nın IP adresi 192.168.1.5 olduğu için 192.168.1.10 en yakın IP adresi olarak öne çıkar.
- Yanıt Sırası: **192.168.1.10, 192.168.3.30, 192.168.2.20**

### Dördüncü Soru:

- İstemci B example.com için tekrar soru gönderir.
- Round-Robin sırası: 192.168.1.10, 192.168.2.20, 192.168.3.30 (Round-robin sırasıyla değiştirilmiştir.)
- Netmask Ordering uygulanır: İstemci B'nin IP adresi 192.168.2.15 olduğu için 192.168.2.20 en yakın IP adresi olarak öne çıkar.
- Yanıt Sırası: **192.168.2.20, 192.168.1.10, 192.168.3.30**

Özet olarak bu örnekte, round-robin DNS her sorguda IP adreslerinin sırasını değiştirir. Ancak, netmask ordering istemcinin IP adresine en yakın IP adresini listenin başına yerleştirir. Böylece, her iki yöntem de birlikte çalışarak hem yük dengelemesi hem de istemciye yakınlık sağlanır.

- Round-robin DNS, IP adreslerini döngüsel olarak sıralar ve yük dengesi sağlar.
- Netmask ordering, istemciye en yakın IP adresini öne çıkararak ağ performansını optimize eder.

## Fail on Load Nedir

"Fail on Load", "Yüklemeye Hata Verirse Başarısız Ol" anlamına gelir. Bu terim, genellikle yazılım ve bilgisayar sistemlerinde kullanılır ve bir programın veya sistemin başlatılmadan önce belirli dosyaları veya konfigürasyonları yüklemesi gerektiğini ve bu yükleme işleminde bir hata oluşması durumunda programın veya sistemin çalışmayı durduracağının ifade eder.

### Kullanım Alanları

- Yazılım Uygulamaları: Bazı yazılım uygulamaları, başlatma sırasında gerekli dosyaları yüklerken hata oluşursa çalışmayı durduracak şekilde tasarlanabilir.
- Bilgisayar Sistemleri: İşletim sistemleri de, önyükleme işlemi sırasında gerekli sürücüler ve konfigürasyon dosyalarını yüklerken "Fail on Load" özelliğini kullanabilir.
- DNS Sunucuları: DNS sunucularında, bölge dosyalarında (zone files) hata olup olmadığını kontrol etmek için kullanılır. Hata varsa, sunucu başlamaz ve hata loglanır. Biz DNS sunucu üzerinde duracağız.

### DNS Üzerinde Fail on Load

#### 1. DNS Yük Dengeleme:

- DNS Round Robin:** Bu yöntemde, birden fazla IP adresi tek bir alan adı için döngüsel olarak döndürülür. Eğer bir IP adresine yapılan istek başarısız olursa (örneğin, sunucu

yanıt vermiyorsa), "Fail on Load" politikası bu IP adresini geçici olarak yük dengeleme havuzundan çıkarabilir ve başka bir IP adresine yönlendirme yapabilir.

- b. **GeoDNS:** Coğrafi konuma dayalı DNS çözümlemesinde, kullanıcı en yakın sunucuya yönlendirilir. Eğer yakın sunucu yanıt vermezse, "Fail on Load" prensibi uygulanarak kullanıcı bir sonraki en yakın sunucuya yönlendirilebilir.

## 2. DNS Sunucu Güvenilirliği:

- a. **Primary ve Secondary DNS:** Birincil DNS sunucusu yanıt vermezse, istemci otomatik olarak ikincil (veya yedek) DNS sunucusuna geçer. Bu, "Fail on Load" prensibinin bir örneğidir çünkü bir DNS sunucusunun yüklenmemesi durumunda başka bir sunucu devreye girer.

## 3. DNSSEC (DNS Security Extensions):

- a. **DNSSEC:** DNSSEC, DNS yanıtlarının doğruluğunu ve güvenilirliğini sağlamak için kullanılan bir güvenlik protokolüdür. Eğer bir DNSSEC imzalı yanıt doğrulanamazsa, DNS çözümleyici yanıtı reddeder ve "Fail on Load" prensibine uygun olarak hatayı bildirir.

Temel mantık ilk aşamada hata alırsak diğer aşamaya geçiyoruz. Ama çalışma mantığında her bir aşama için farklı çözümler olabilir.

## Cache Pollution Nedir

Cache kirliliği, bir bilgisayarın önbelleğinde yanlış veya güncel olmayan verilerin depolanması durumudur. Bu, genellikle DNS (Alan Adı Sistemi) önbellekleri gibi sistem önbelleklerinde gerçekleşir.

Cache kirliliğinin birkaç sebebi olabilir:

- **Yanlış veya güncel olmayan bilgiler:** Bir sunucu yanlış veya güncel olmayan bilgiler sağlayabilir.
- **DNS saldırıcıları:** Kötü amaçlı aktörler, bir sunucunun DNS önbelleğini yanlış bilgilerle kirletmek için saldırular düzenleyebilir.
- **Yanlış konfigürasyon:** Bir sunucu yanlış şekilde yapılandırılmışsa, önbelleğinde hatalı bilgiler saklayabilir.

Cache kirliliği, çeşitli sorunlara yol açabilir:

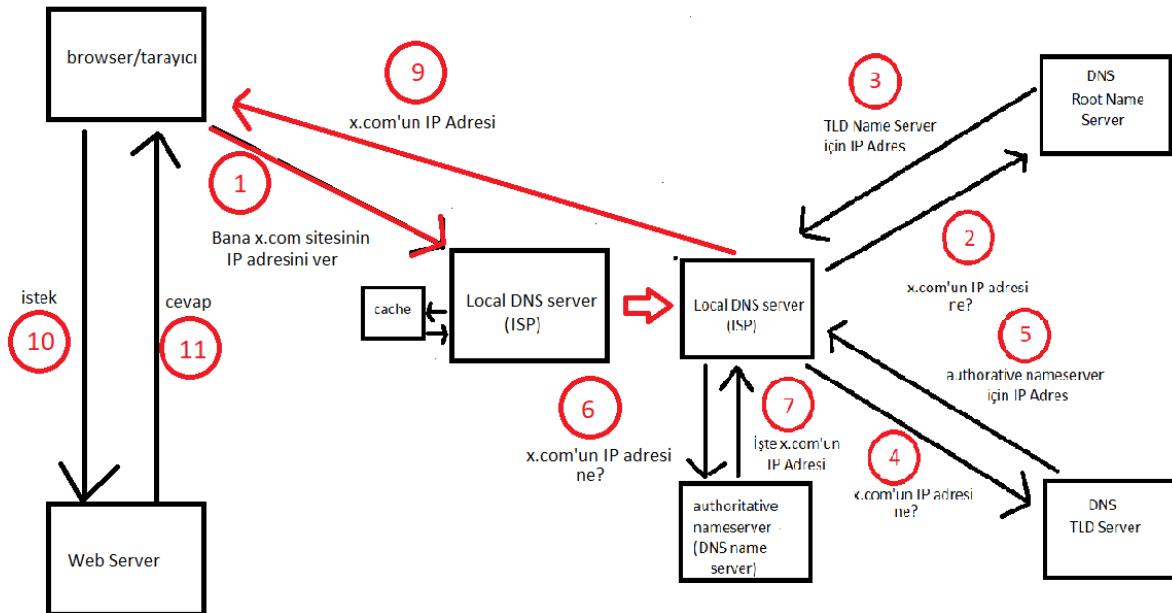
- **Yanlış yönlendirme:** Kullanıcılar, yanlış IP adreslerine yönlendirilebilir ve web sitelerine veya diğer çevrimiçi kaynaklara erişemeyebilir.
- **Güvenlik riskleri:** Kirli bir önbellek, kötü amaçlı aktörlerin bilgisayarlara ve ağlara sızmasına olanak sağlayabilir.
- **Performans sorunları:** Önbellekte hatalı bilgiler olması, sunucu performansını düşürebilir.

Cache kirliliğini önlemek için çeşitli yöntemler kullanılabilir:

- **DNS önbelleklerini temizlemek:** Önbellekleri düzenli olarak temizlemek, hatalı bilgilerin birikmesini önerler.
- **Güvenlik önlemleri almak:** DNS sunucularını korumak için güvenlik duvarları ve diğer güvenlik araçları kullanılabilir.
- **DNS önbellek ayarlarını optimize etmek:** Önbellek boyutunu ve süresini optimize etmek, kirliliği önlemeye yardımcı olabilir.

## Conditional Forwarder Nedir

Conditional forwarder, DNS (Domain Name System) yapısında kullanılan bir kavramdır. Bu özellik, bir DNS sunucusunun belirli bir DNS bölgesi için sorguları başka bir DNS sunucusuna yönlendirmesini sağlar. Bu, genellikle farklı organizasyonlar arasında veya birden fazla alt ağ (subnet) içeren büyük ağlarda kullanılır.



Yukarıdaki resimde Conditional Forwarder resmedilmiştir. Bu aşama bildiğimiz DNS sorgusudur. Tek fark ilk DNS sunucuda DNS çözümlemesi yapılmaz, bunun yerine birinci DNS sunucudan ikinci DNS sunucuya soru iletilir. İkinci DNS sunucuda DNS çözümlemesi yapılır.

Conditional forwarder'in kullanışlı olduğu bazı senaryolar şunlardır:

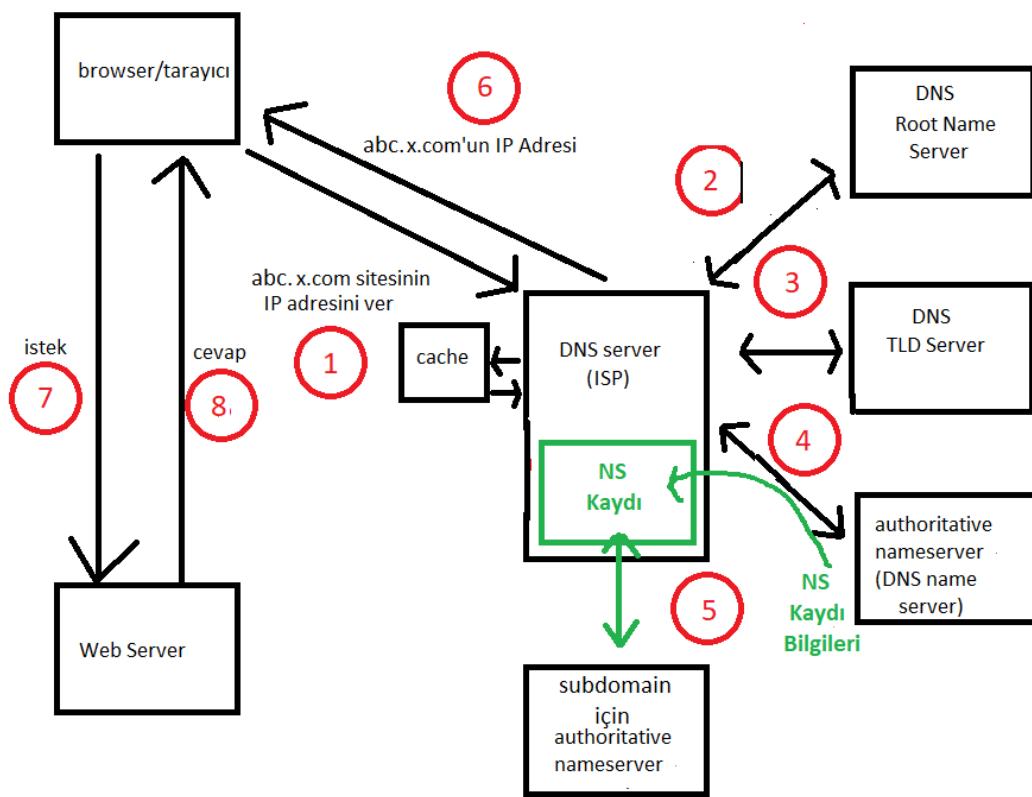
- Şirketilerin alt alanlarının (subdomain) çözümlenmesi için farklı bir DNS sunucusuna yönlendirme.
- Yerel ağınız dışındaki bir domain için yetkili bir DNS sunucusuna yönlendirme.

Conditional forwarder'lar, DNS sorgularının yönlendirilmesi konusunda daha fazla esneklik sağlar ve ağın DNS çözümleme yeteneklerini arttırmır.

## DNS Delegation Nedir

DNS (Domain Name System) Delegation, bir DNS alan adının sorumluluğunun bir DNS sunucusundan diğerine devredilmesi işlemidir. Bu, alan adının alt alan adlarıyla (subdomain) ilgilidir ve bu alt alan adlarının farklı DNS sunucularına yönlendirilmesini sağlar.

Örneğin, "example.com" alan adınızın sahibi olduğunuzu düşünelim. "example.com" alan adını yöneten DNS sunucunuz olabilir, ancak "sub.example.com" alt alan adını yönetmek için farklı bir DNS sunucusuna ihtiyaç duyabilirsiniz. DNS delegasyonu, "sub.example.com" alt alan adının sorgularının bu farklı DNS sunucusuna yönlendirilmesini sağlar.



### DNS Subdomain Sorgusu

Subdomain için DNS sorgusunda ilk aşamada istemci DNS sunucuya soru gönderir. Daha sonra DNS sunucu sırasıyla root, tld ve authoritative name serverda ip adresi arar. Bu işlemler x.com adresini aramak içindir. Son olarak subdomain için auhtoritative name servera gidilir ve buradan subdomainin IP adresi alınır. Böylece abc.x.com alan adının IP adresi elde edilir.

NS kaydı için gerekli bilgiler x.com'un elde edilmesinde kullanılan sorgudan gelir. Bu sorgudaki (root, tld ve authoritative serverı gezen soru) son aşama olan authoritative name server aşaması ile, auhtoritative name server üzerinde bulunan bilgiler (NS kaydı bilgileri) DNS sunucuya döner.

DNS sunucu içerisinde NS kaydı bilgileri geldikten sonra subdomain için işlemler yapılabilir. Subdomain için authoritative name server, DNS sunucuda bulunan NS kaydı ile oluşur. Yani NS kaydı sayesinde subdomain çözümlemesi yapılır. (NS kaydı dışında A, MX, CNAME gibi diğer DNS kayıtları da iletilir.)

Subdomain sorgusuna özet olarak adımlar üzerinden bakacak olursak:

1. abc.x.com subdomaini için IP adres istenir.
2. DNS sunucu ilk önce cache içerisinde bakar. Eğer IP adres yoksa soru başlar.
3. DNS root name servera IP sorulur. Cevap olarak sadece “.” döner.
4. Daha sonra TLD servera eldeki bilgilerle (“.”) IP adres sorulur. TLD server sadece “.com” cevabını döndürür.
5. Bu aşamadan sonra ise authoritative name servera IP adres sorulur (“.com.” bilgileriyle). Bu sunucu ise “x.com.” alan adının IP adresini döndürür. Bunun yanında NS kaydı bilgileri de dns sunucuya döner. Normalde biz x.com'un IP adresini öğrendik. Ancak bize x.com'un subdomain adresi lazım. (NS kaydı dışında A, MX, CNAME gibi diğer DNS kayıtları da iletilir.)
6. Son olarak subdomain için bulunan authoritative name servera başvurulur. NS kaydı sayesinde bu sunucudan, abc.x.com alan adının IP adresi döner.
7. Böylece abc.x.com alan adının IP adresi DNS sunucudan istemciye (tarayıcı) döner. Bu sayede bu IP adres ile web sunucuya istek gönderili ve cevap alınır. abc.x.com arayüzü kullanıcıya gösterilir.

**SON**