BSM 471-AĞ GÜVENLİĞİ

Hafta5: Katman 7 Protokolleri ve Çalışma Yapıları

Dr. Öğr. Üyesi Musa BALTA
Bilgisayar Mühendisliği Bölümü
Bilgisayar ve Bilişim Bilimleri Fakültesi

Konu İçeriği

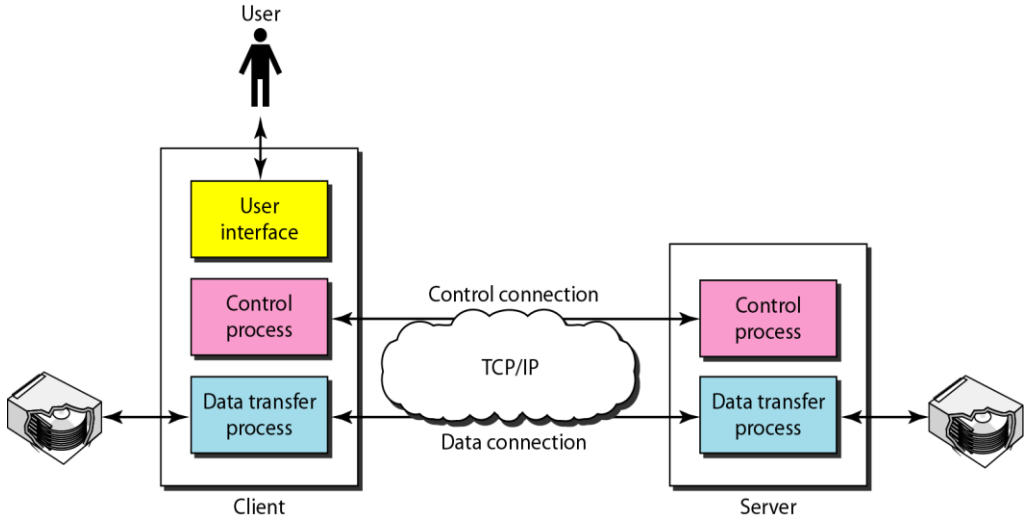
- FTP
- SMTP
- SNMP
- HTTP
- DHCP
- DNS

Dosya Transfer Protokolü (File Transfer Protocol-FTP)

- Dosya Aktarım Protokolü (FTP), TCP / IP tarafından bir dosyayı bir ana bilgisayardan diğerine kopyalamak için sağlanan standart mekanizmadır.
- Her ne kadar bir sistemden başka bir sisteme dosya aktarımı kolay ve anlaşılabilir metodoloji olarak görünse de, aşağıdaki gibi bazı sorunlar oluşabilmektedir.
  - Kaynak ve hedef sistemler ***farklı yollarla verileri tutabilir*** veya temsil edebilirler.
  - Kaynak ve hedef sistemler ***farklı klasör yapılarına*** sahip olabilirler.
  - Bağlantı esnasında ***kurulum parametreleri*** ve veri transferinde kontrol gerekebilir.
- FTP, ana bilgisayarlar arasında iki bağlantı kurması bakımından diğer istemci-sunucu uygulamalarından farklıdır.
- Bir bağlantı veri aktarımı için, diğeri kontrol bilgileri (komutlar ve yanıtlar) için kullanılır.

Dosya Transfer Protokolü (devam)

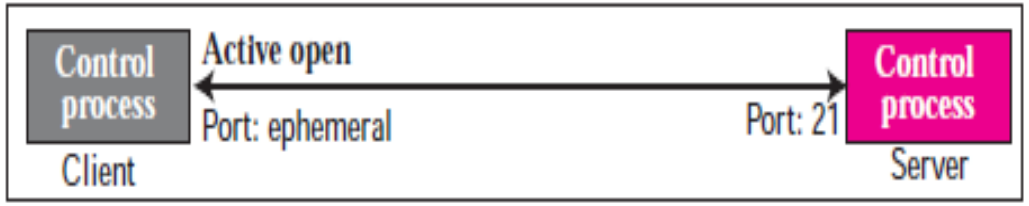
- Komutların ayrılması ve veri aktarımı FTP'yi daha verimli hale getirir. Bir seferde yalnızca bir komut satırı veya bir yanıt satırı aktarmamız gerekir.
- Kontrol bağlantısı çok basit iletişim kuralları kullanır.
- Veri bağlantısı ise, aktarılan veri türlerinin çeşitliliği nedeniyle daha karmaşık kurallara ihtiyaç duyar.
- FTP, iyi bilinen iki TCP bağlantı noktası kullanır:
  - Kontrol bağlantıları için **Port 21**
  - Veri aktarımı için ise **Port 20**



Dosya Transfer Protokolü (Kontrol Bağlantısı)

- Bunun için 2 aşama vardır:
  - Sunucu, iyi bilinen bağlantı noktası 21'de pasif bir açık açar ve bir müşteri bekler.
  - İstemci geçici bir bağlantı noktası kullanır ve etkin bir açık yayınlar.
- Bağlantı tüm süreç boyunca açık kalır.
- IP protokolü tarafından kullanılan hizmet türü gecikmeyi en aza indirir, çünkü bu bir kullanıcı (insan) ile sunucu arasında etkileşimli bir bağlantıdır.
- Kullanıcı komutları yazar ve önemli bir gecikme olmadan yanıt almayı bekler.

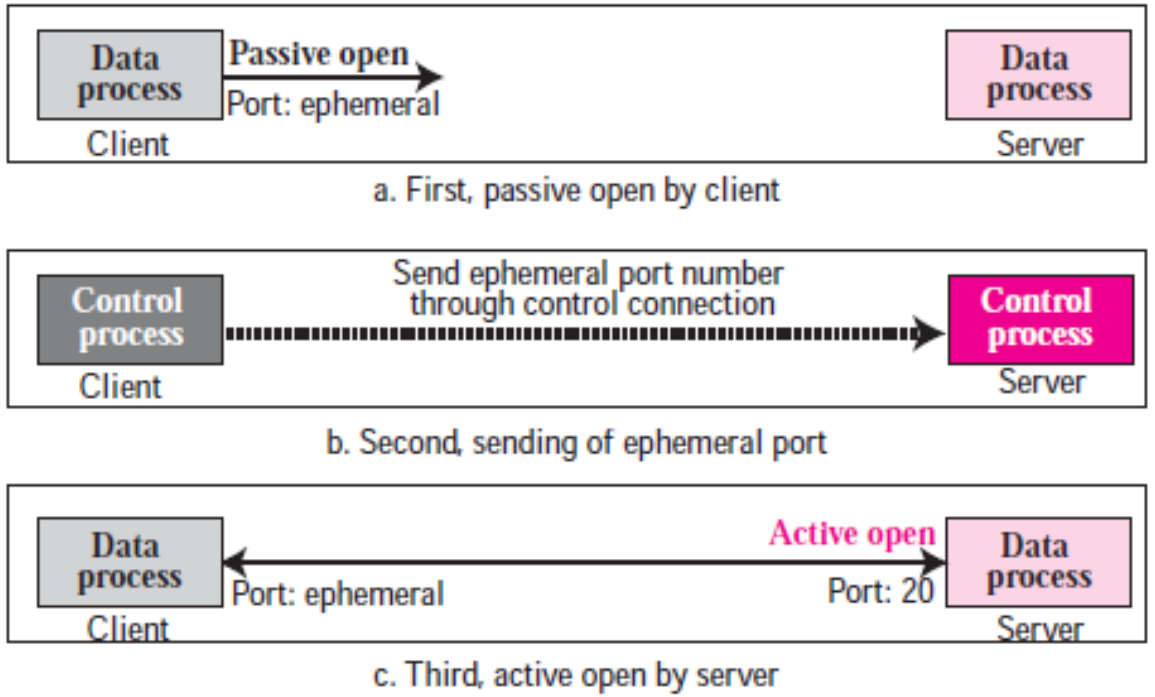
a. First, passive open by server



b. Later, active open by client

Dosya Transfer Protokolü (Veri Bağlantısı)

- Veri bağlantısı, sunucu tarafında iyi bilinen bağlantı noktasını (20) kullanır.
- Aşağıda FTP'nin nasıl veri bağlantısı oluşturduğu gösterilmektedir:
    1. Sunucu değil, istemci geçici bir bağlantı noktası kullanarak pasif bir açık yayınlar.
    2. İstemci, bu bağlantı noktası numarasını PORT komutunu kullanarak sunucuya gönderir.
    3. Sunucu, bağlantı noktası numarasını alır ve iyi bilinen bağlantı noktası 20'yi ve alınan geçici bağlantı noktası numarasını kullanarak etkin bir açık verir.
- Veri bağlantısı açılır ve aktarılan her dosya için kapatılır.
- Dosya aktarımını içeren komutlar her kullanıldığında açılır ve dosya aktarıldığında kapanır.
- Kontrol bağlantısı açıkken, birkaç dosya aktarılırsa veri bağlantısı birkaç kez açılıp kapatılabilir.

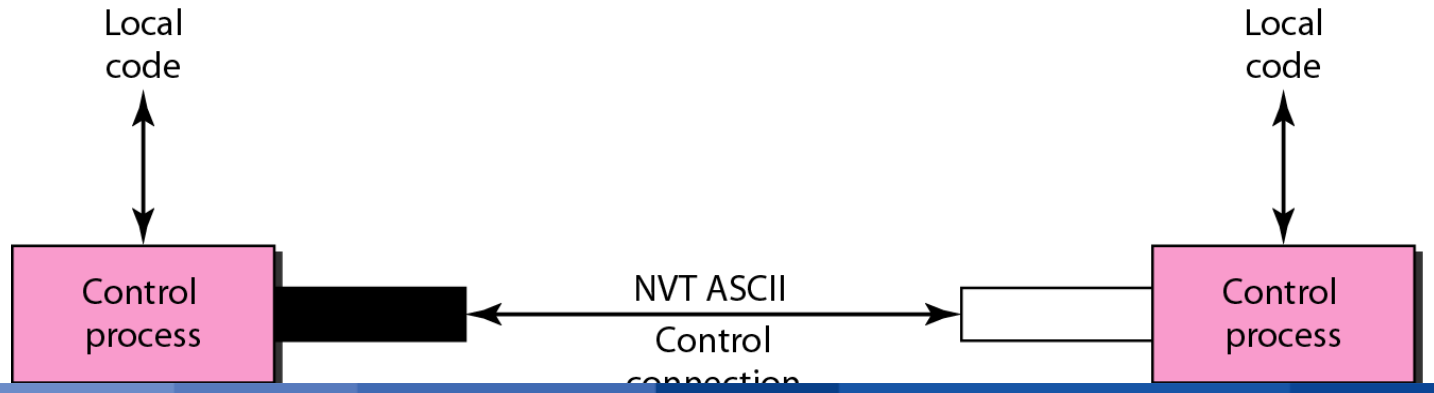


Dosya Transfer Protokolü (Veri İletişimi)

- Farklı bilgisayarlarda çalışan FTP istemcisi ve sunucusu birbirleriyle iletişim kurmalıdır.
- Bu iki sistem:
  - Farklı işletim sistemlerine,
  - Farklı karakter kümelerine,
  - Farklı dosya yapılarına,
  - Farklı dosya formatlarına sahip olabilirler.
- FTP bu heterojenliği uyumlu hale getirmelidir.
- FTP'nin biri kontrol bağlantısı diğeri veri bağlantısı için olmak üzere iki farklı yaklaşımı vardır.

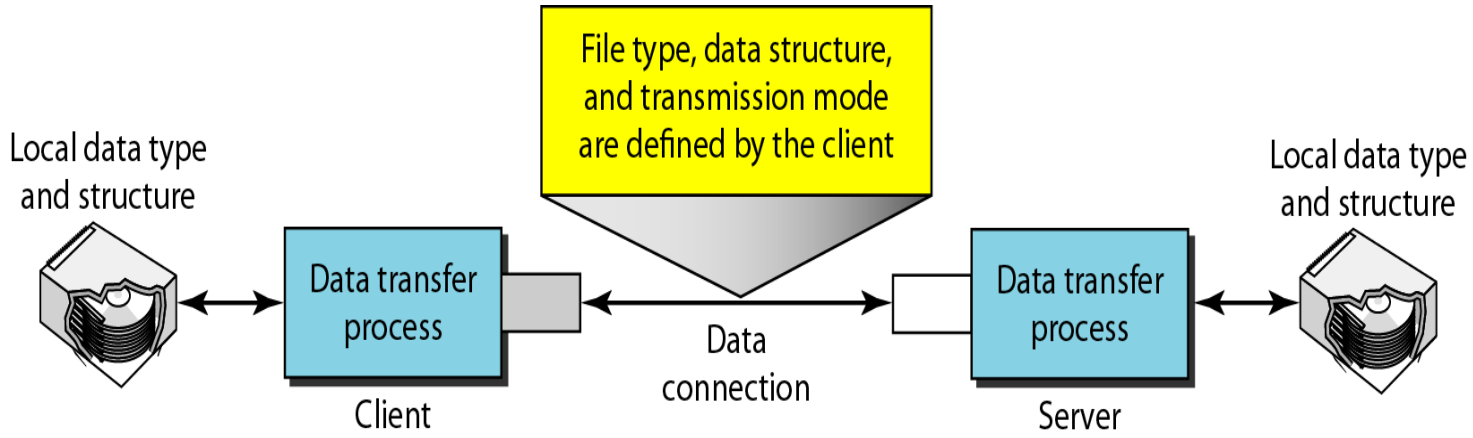
Dosya Transfer Protokolü (Kontrol bağlantısı üzerinden iletişim)

- FTP, kontrol bağlantısı üzerinden iletişim kurmak için SMTP ile aynı yaklaşımı kullanır.
- 7-bitlik ASCII karakter kümesini kullanır.
- İletişim, komutlar ve yanıtlar aracılığıyla sağlanır.
- Her komut veya yanıt yalnızca bir kısa satırdır, bu nedenle dosya biçimi veya dosya yapısı hakkında endişelenmemize gerek yoktur.
- Her satır iki karakterli (satır başı ve satır besleme) satır sonu jetonu ile sonlandırılır.



Dosya Transfer Protokolü (Veri bağlantısı üzerinden iletişim)

- Dosya aktarımı, kontrol bağlantısı üzerinden gönderilen komutların kontrolü altında veri bağlantısı üzerinden gerçekleşir.
- Kontrol bağlantısı üzerinden iletim için hazırlanır.
- Heterojenlik sorunu, dosyayı veri bağlantısı üzerinden göndermeden önce üç iletişim niteliği tanımlanarak çözülür:
  - Dosya tipi
  - Veri yapısı
  - İletim modu



Dosya Transfer Protokolü (Dosya Tipi)

- FTP, veri bağlantısı üzerinden aşağıdaki dosya türlerinden birini aktarabilir:
  - ASCII dosyası
  - EBCDIC dosyası
  - Görüntü dosyası
- **ASCII dosyası**, metin dosyalarını aktarmak için varsayılan biçimdir.
  - Her karakter 7 bitlik ASCII kullanılarak kodlanır.
  - Gönderen dosyayı kendi gösteriminden ASCII karakterlerine ve alıcı ASCII karakterlerini kendi gösterimine dönüştürür.
- **EBCDIC dosya** türü ise, bağlantının bir veya iki tarafı birden EBCDIC kodlama türünü (IBM tarafından kullanılan dosya formatı) kullanılırsa, kullanılır.
- **Görüntü dosyası**, ikili dosyaları aktarmak için varsayılan biçimdir.
  - Dosya, herhangi bir yorum veya kodlama yapılmaksızın sürekli bit akışı olarak gönderilir.
  - Bu çoğunlukla derlenmiş programlar gibi ikili dosyaları aktarmak için kullanılır.

Dosya Transfer Protokolü (Veri Yapıları)

- FTP, verilerin yapısı hakkında aşağıdaki yorumlardan birini kullanarak bir dosyayı veri bağlantısı üzerinden aktarabilir:
- **dosya yapısı:** dosya sürekli bir bayt akışıdır.
- **kayıt yapısı**: dosya kayıtlara ayrılmıştır. Bu yalnızca metin dosyalarıyla kullanılabilir.
- **sayfa yapısı:** dosya sayfalara bölünür ve her sayfada bir sayfa numarası ve bir sayfa başlığı bulunur.
- Sayfalar rastgele veya sıralı olarak saklanabilir ve erişilebilir.

Dosya Transfer Protokolü (İletim Modu)

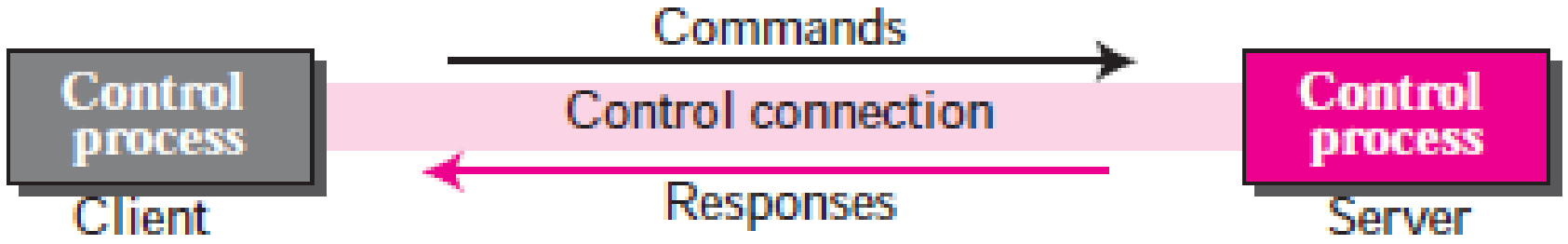
- FTP, aşağıdaki üç iletim modundan birini kullanarak bir dosyayı veri bağlantısı üzerinden aktarabilir:
  - Akış (stream) modu
  - Block modu
  - Sıkıştırılmış mod
- **Stream mod:**
- Bu varsayılan moddur. Veriler sürekli bayt akışı olarak FTP'den TCP'ye gönderilir.
- TCP, verileri uygun boyuttaki bölümlere ayırmaktan sorumludur.
- Veriler sadece bir bayt akışı (dosya yapısı) ise, dosya sonu gerekmez.
- Bu durumda dosya sonu, gönderen tarafından veri bağlantısının kapatılmasıdır.
- Veriler kayıtlara ayrılırsa (kayıt yapısı), her kaydın 1 bayt kayıt sonu (EOR) karakteri olur ve dosyanın sonu 1 bayt dosya sonu (EOF) karakteri içerir .

Dosya Transfer Protokolü (İletim Modu)

- **Block mod:**
- Veriler FTP'den TCP'ye bloklar halinde iletilebilir.
- Her bloktan önce 3 baytlık bir başlık gelir. İlk bayta blok tanımlayıcı denir; sonraki iki bayt, bloğun boyutunu bayt olarak tanımlar.
- **Sıkıştırılmış mod:**
- Dosya büyükse, veriler sıkıştırılabilir.
- Normalde kullanılan sıkıştırma yöntemi çalışma uzunluğu kodlamasıdır.
- Bir veri biriminin ardışık görünümleri, bir oluşum ve tekrar sayısı ile değiştirilir.
- Bir metin dosyasında bu genellikle boşluklardır (boşluklar).
- İkili dosyada, boş karakterler genellikle sıkıştırılır.

Dosya Transfer Protokolü (Komut İşleme)

- FTP, kontrol bağlantısını istemci kontrol işlemi ile sunucu kontrol işlemi arasında bir iletişim kurmak için kullanır.
- Bu iletişim sırasında komutlar istemciden sunucuya gönderilir ve yanıtlar sunucudan istemciye gönderilir.



Dosya Transfer Protokolü (Komutlar)

- Komutlar FTP istemci kontrol işleminden gönderilir.
- ASCII büyük harf biçimindedir, bunu bir argüman izleyebilir veya izlemeyebilir.
- komutlar altı gruba ayrılır:
  - Erişim komutları
  - Dosya yönetim komutları
  - Veri biçimlendirme komutları
  - Port tanımlama komutları
  - Dosya transfer komutları
  - Çeşitli komutlar

Dosya Transfer Protokolü (Komutlar-devam)

- **Erişim komutları:** Bu komutlar kullanıcının uzak sisteme erişmesini sağlar.

Command	*Argument(s)*	*Description*
USER	User id	User information
PASS	User password	Password
ACCT	Account to be charged	Account information
REIN		Reinitialize
QUIT		Log out of the system
ABOR		Abort the previous command

- **Dosya yönetim komutları:** Bu komutlar, kullanıcının uzak bilgisayardaki dosya sistemine erişmesini sağlar. Kullanıcının dizin yapılarında gezinmesine (yeni dizi yapıları oluşturması, dosya silmesi vb.) izin verir.

Command	*Argument(s)*	*Description*
CWD	Directory name	Change to another directory
CDUP		Change to parent directory
DELE	File name	Delete a file
LIST	Directory name	List subdirectories or files
NLIST	Directory name	List subdirectories or files without attributes
MKD	Directory name	Create a new directory
PWD		Display name of current directory
RMD	Directory name	Delete a directory
RNFR	File name (old)	Identify a file to be renamed
RNTO	File name (new)	Rename the file
SMNT	File system name	Mount a file system

Dosya Transfer Protokolü (Komutlar-devam)

- **Veri biçimlendirme komutları:** Bu komutlar kullanıcının veri yapısını, dosya türünü ve iletim modunu tanımlamasını sağlar. Tanımlanan format daha sonra dosya aktarım komutları tarafından kullanılır.

Command	*Argument(s)*	*Description*
TYPE	A (ASCII), E (EBCDIC), I (Image), N (Nonprint), or T (TELNET)	Define file type
STRU	F (File), R (Record), or P (Page)	Define organization of data
MODE	S (Stream), B (Block), or C (Compressed)	Define transmission mode

- **Port tanımlama komutları:** Bu komutlar, istemci tarafındaki veri bağlantısı için bağlantı noktası numarasını tanımlar.

Command	*Argument(s)*	*Description*
PORT	6-digit identifier	Client chooses a port
PASV		Server chooses a port

Dosya Transfer Protokolü (Komutlar-devam)

- **Dosya transfer komutları:**Bu komutlar aslında kullanıcının dosyaları aktarmasına izin verir.

Command	*Argument(s)*	*Description*
RETR	File name(s)	Retrieve files; file(s) are transferred from server to client
STOR	File name(s)	Store files; file(s) are transferred from client to server
APPE	File name(s)	Similar to STOR, but if file exists, data must be appended to it
STOU	File name(s)	Same as STOR, but file name will be unique in the directory
ALLO	File name(s)	Allocate storage space for files at the server
REST	File name(s)	Position file marker at a specified data point
STAT	File name(s)	Return status of files

- **Çeşitli komutlar:** Bu komutlar istemci sitesindeki FTP kullanıcısına bilgi verir.

Command	*Argument(s)*	*Description*
HELP		Ask information about the server
NOOP		Check if server is alive
SITE	Commands	Specify the site-specific commands
SYST		Ask about operating system used by the server

Dosya Transfer Protokolü (Cevaplar)

- Her FTP komutu en az bir yanıt oluşturur.
- Yanıt iki bölümden oluşur: üç basamaklı bir sayı ve ardından metin.
- Sayısal kısım kodu tanımlar.
  - İlk hane komutun durumunu tanımlar.
  - İkinci basamak ayrıca komutun durumunu tanımlar.
  - Üçüncü basamak ek bilgi sağlar.
- Metin bölümü, gerekli parametreleri veya ek açıklamaları tanımlar.

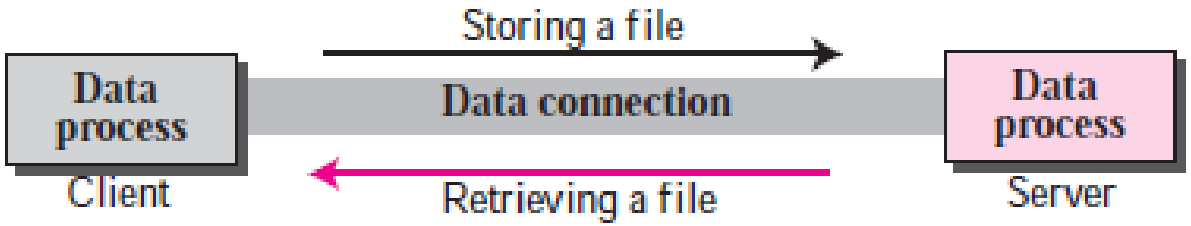
Dosya Transfer Protokolü (Cevaplar-devam)

Code	*Description*
Positive Preliminary Reply	
120	Service will be ready shortly
125	Data connection open; data transfer will start shortly
150	File status is OK; data connection will be open shortly
Positive Completion Reply	
200	Command OK
211	System status or help reply
212	Directory status
213	File status
214	Help message
215	Naming the system type (operating system)
220	Service ready
221	Service closing
225	Data connection open
226	Closing data connection
227	Entering passive mode; server sends its IP address and port number
230	User login OK
250	Request file action OK
Positive Intermediate Reply	
331	User name OK; password is needed
332	Need account for logging

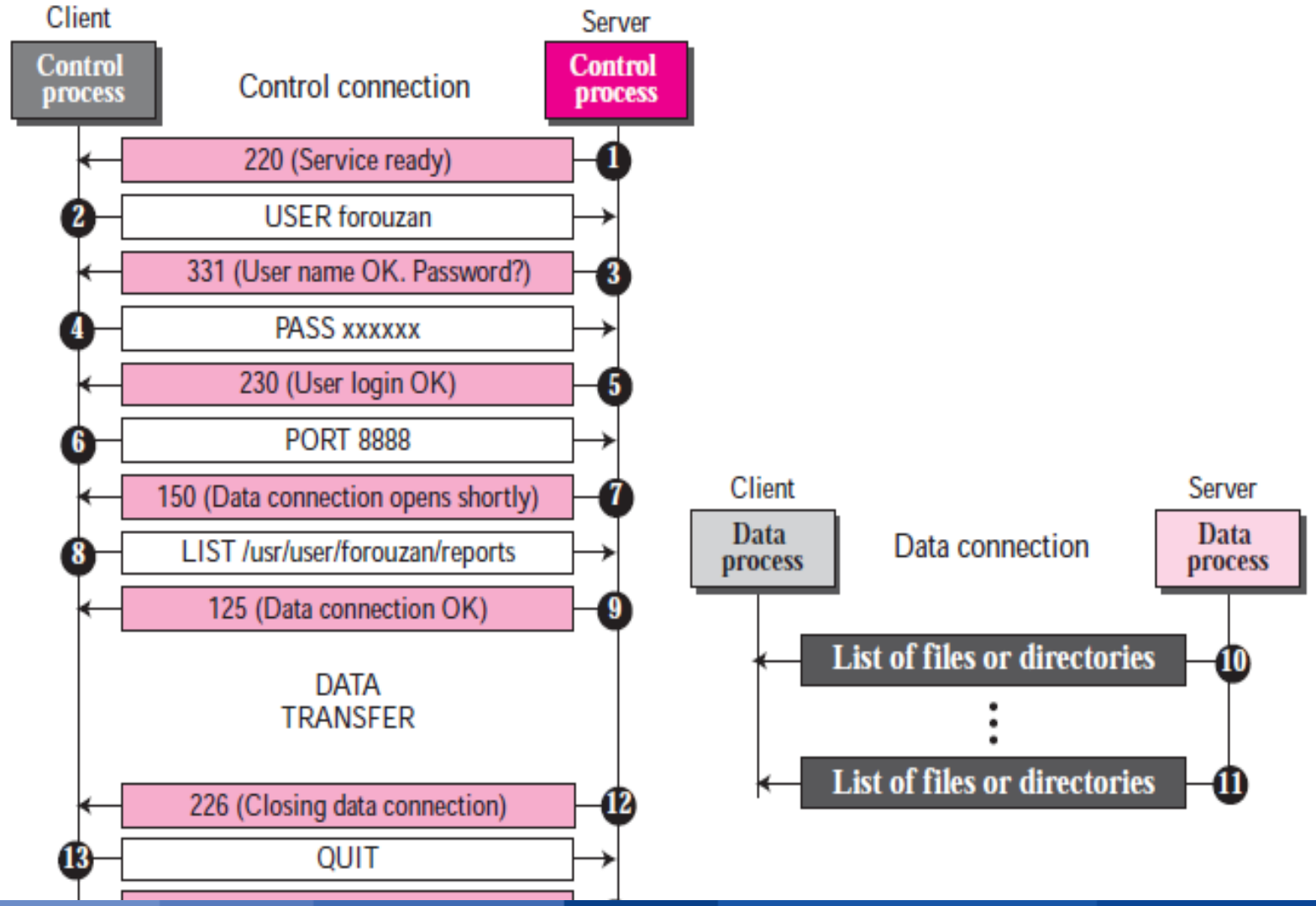
Code	*Description*
Transient Negative Completion Reply	
425	Cannot open data connection
426	Connection closed; transfer aborted
450	File action not taken; file not available
451	Action aborted; local error
452	Action aborted; insufficient storage
Permanent Negative Completion Reply	
500	Syntax error; unrecognized command
501	Syntax error in parameters or arguments
502	Command not implemented
503	Bad sequence of commands
504	Command parameter not implemented
530	User not logged in
532	Need account for storing file
550	Action is not done; file unavailable
552	Requested action aborted; exceeded storage allocation
553	Requested action not taken; file name not allowed

Dosya Transfer Protokolü (Dosya-Transferi)

- FTP'de dosya aktarımı üç şeyden biri anlamına gelir:
- **Alınıyor:** Bir dosya sunucudan istemciye kopyalanacaktır. RETR komutunun denetimi altında yapılır.
- **Depolama:** Bir dosya istemciden sunucuya kopyalanacaktır. STOR komutunun denetimi altında yapılır.
- **Listeleme:** Dizin veya dosya adlarının bir listesi sunucudan istemciye gönderilir. Bu, LIST komutunun denetimi altında yapılır. FTP, dizin veya dosya adlarının listesini dosya olarak değerlendirir.



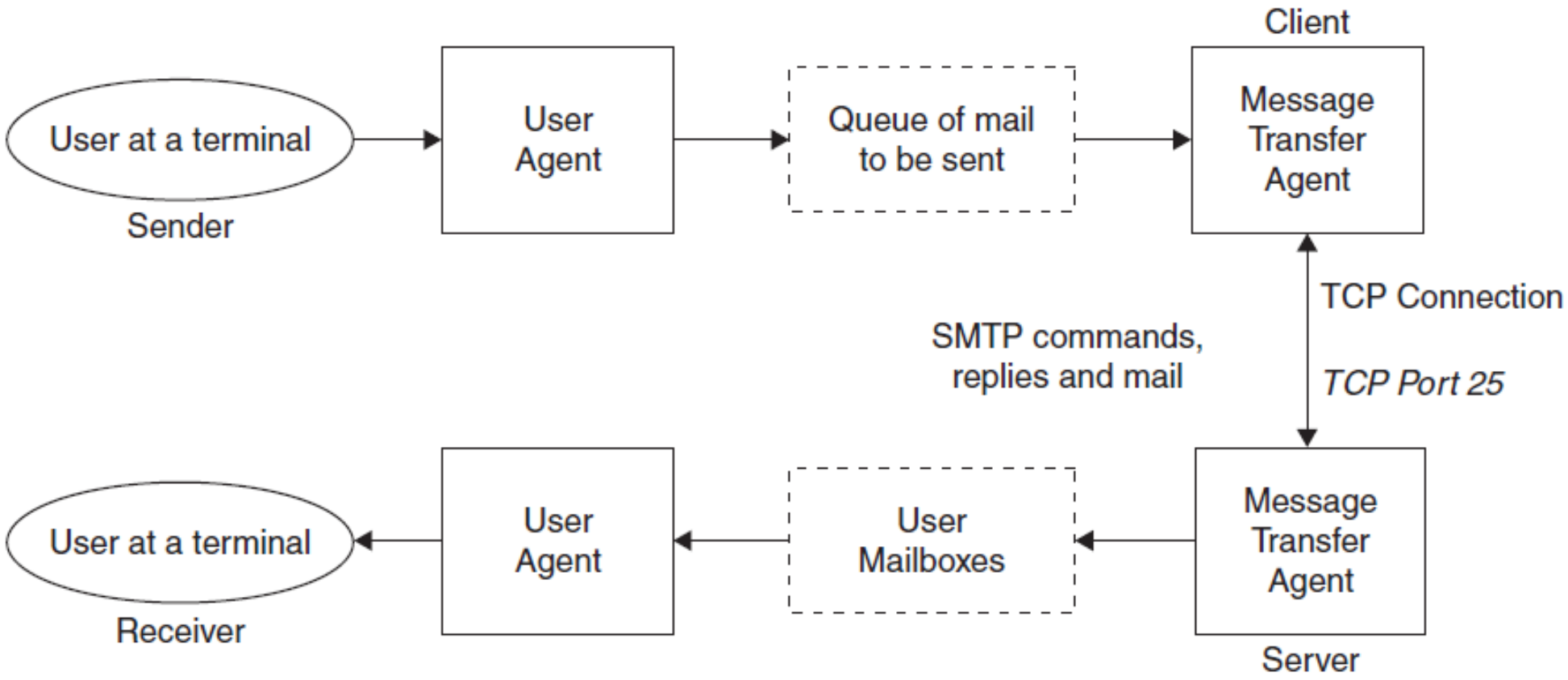
Bir Dizindeki Öğelerin Listesini Almak İçin FTP Kullanma



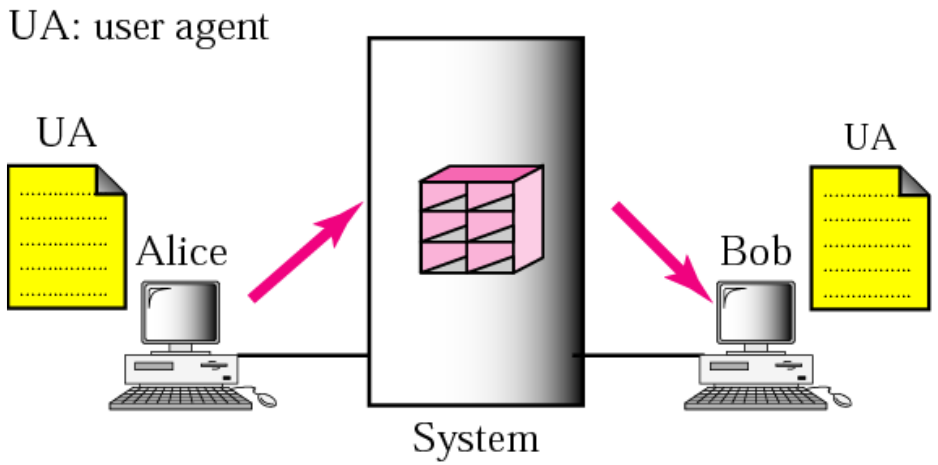
Basit Mail Transfer Protokolü (Simple Mail Transfer Protocol-SMTP)

- Elektronik mail (e-mail) en popüler ağ hizmetlerinden bir tanesidir.
- Bir sunucudan başa bir sunucuya Internet üzerinden mail göndermek için bir çok e-mail sistemi SMTP protokolünü kullanır. Mesajlar daha sonra **POP (post office protocol)** veya **IMAP (Internet message Access protocol)** kullanan bir e-mail istemcisi ile sunucudan alınabilir.
- SMTP bir RFC 821 standardıdır.
- Bir SMTP istemcisi uzak bir düğüm üzerindeki SMTP sunucusuna bir TCP bağlantısı kurar. SMTP sunucusu bu bağlantı için **25 nolu** port numarasını kullanır. Bağlantı kurulduktan sonra basit bir istek-cevap mekanizması çalışır.
- SMTP-RFC 821'e göre gönderilen iletide; Mesaj başlığı ve içeriği olmak zorundadır.

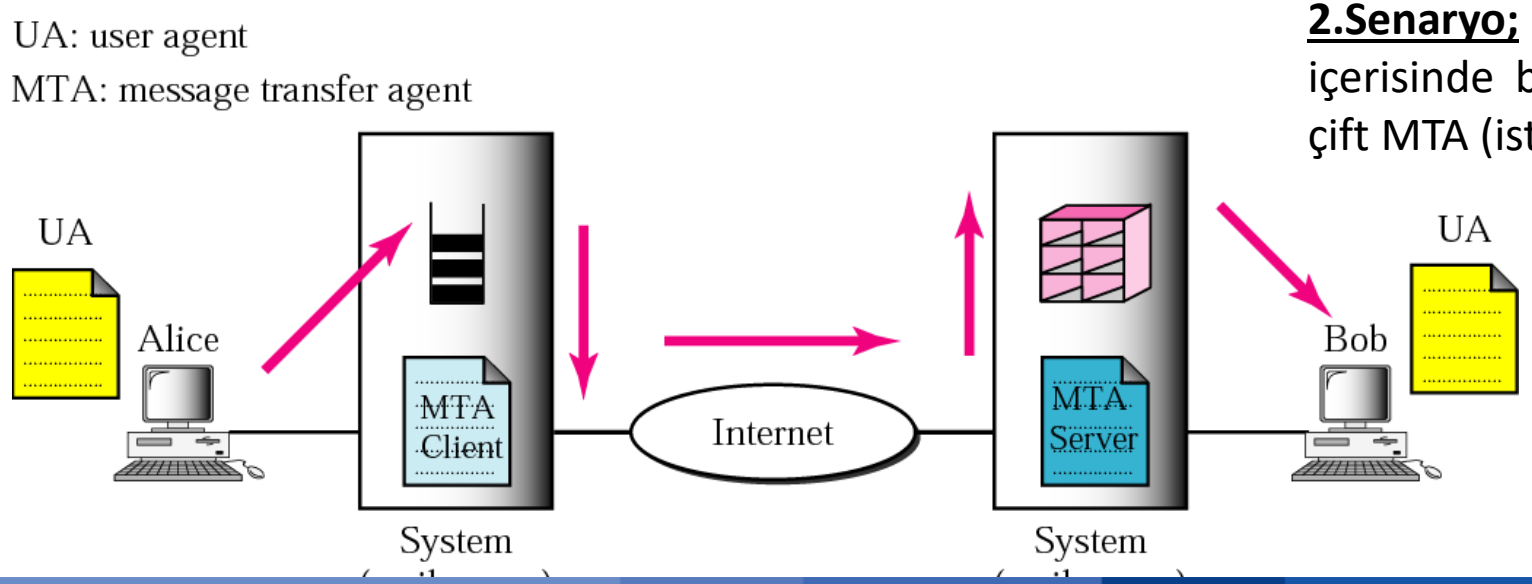
Basit Mail Transfer Protokolü (SMTP Modeli)



Basit Mail Transfer Protokolü (SMTP Senaryolar)

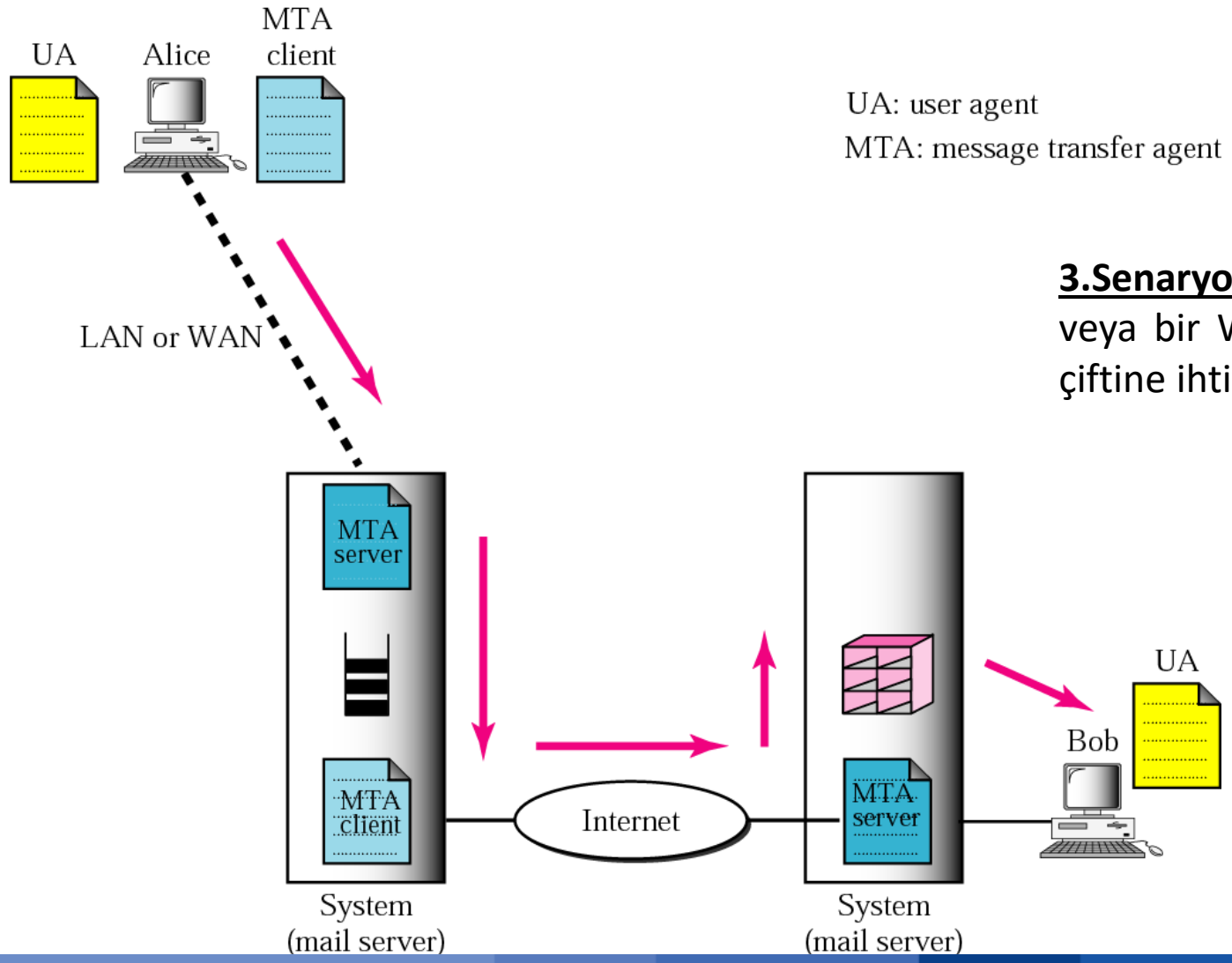


1.Senaryo; E-mailin gönderici ve alıcısı aynı sistem içerisinde bulunuyorlarsa, sadece iki adet kullanıcı ajanı yeterlidir.



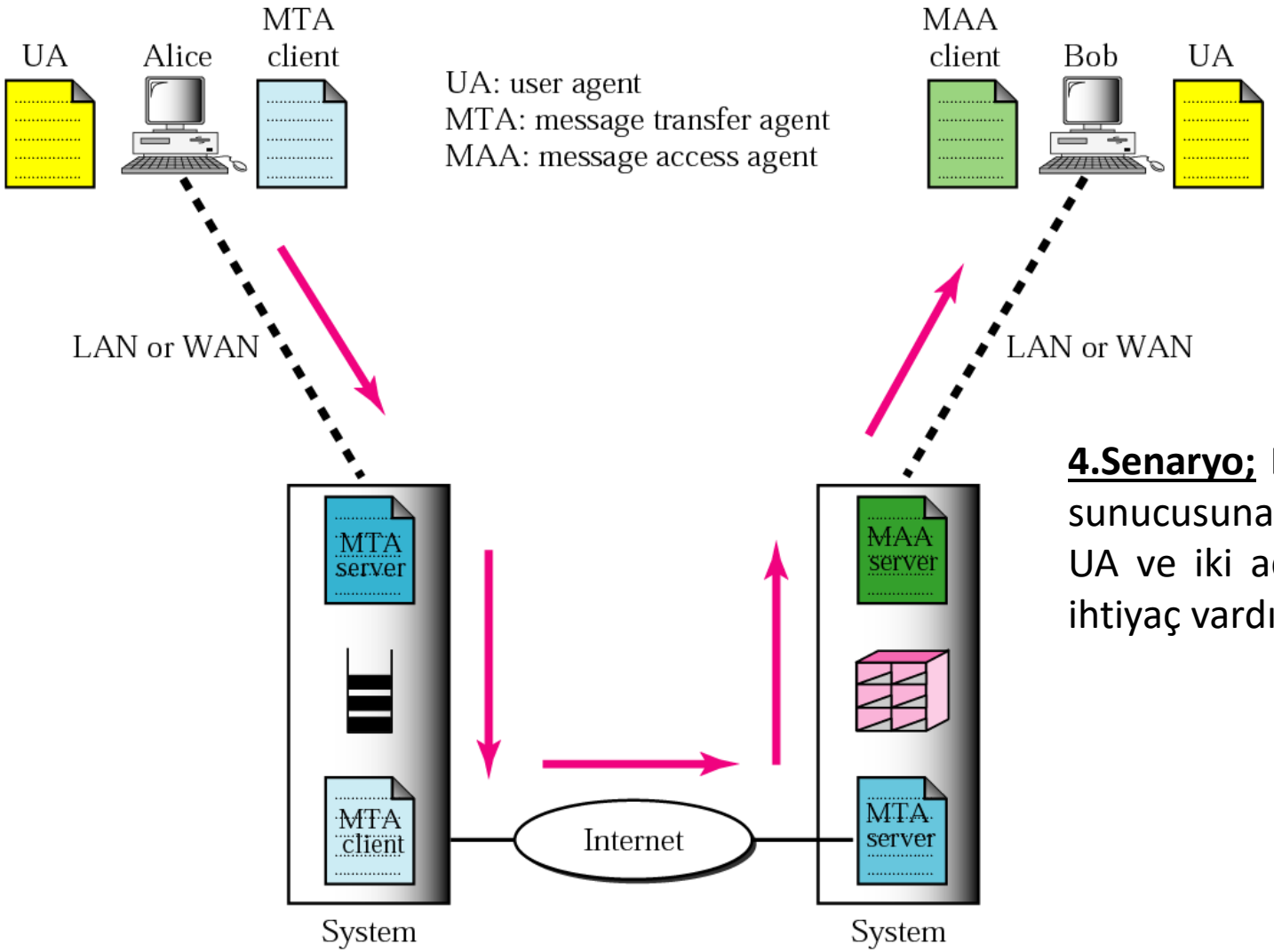
2.Senaryo; E-mailin gönderici ve alıcısı farklı sistem içerisinde bulunuyorlarsa, iki adet kullanıcı ajanı ve bir çift MTA (istemci ve sunucu için) gereklidir.

Basit Mail Transfer Protokolü (SMTP Senaryolar-Devam)



3.Senaryo; E-mailin göndericisi mail sunucusuna bir LAN veya bir WAN ile bağlıysa, iki adet UA ve iki adet MTA çiftine ihtiyaç vardır.

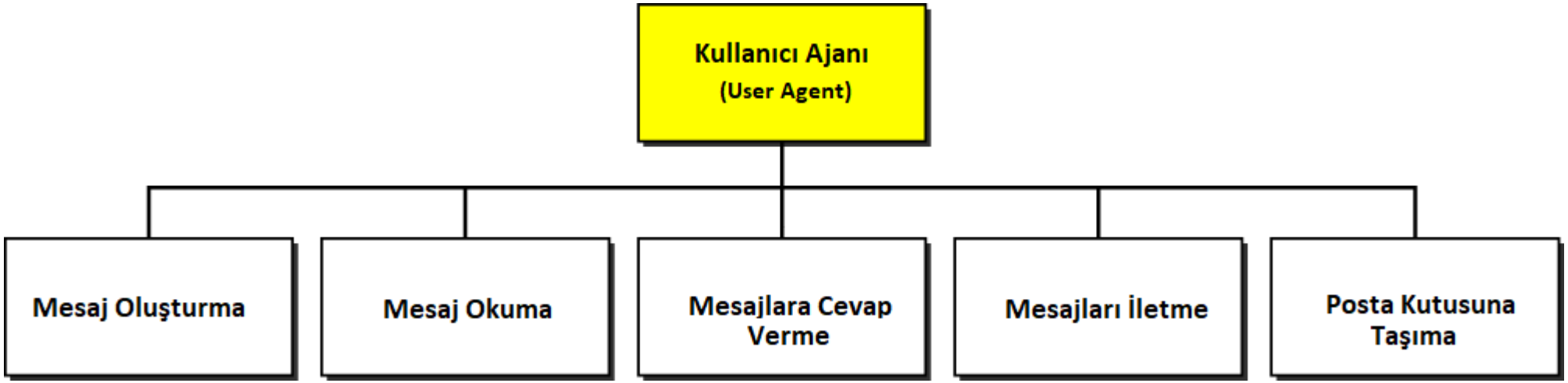
Basit Mail Transfer Protokolü (SMTP Senaryolar-Devam)



4.Senaryo; E-mailin hem göndericisi hem de alıcısı, mail sunucusuna bir LAN veya bir WAN ile bağlıysa, iki adet UA ve iki adet MTA çiftine ve de bir adet MAA çiftine ihtiyaç vardır. ***Günümüzde kullanılan yapı bu şekildedir.***

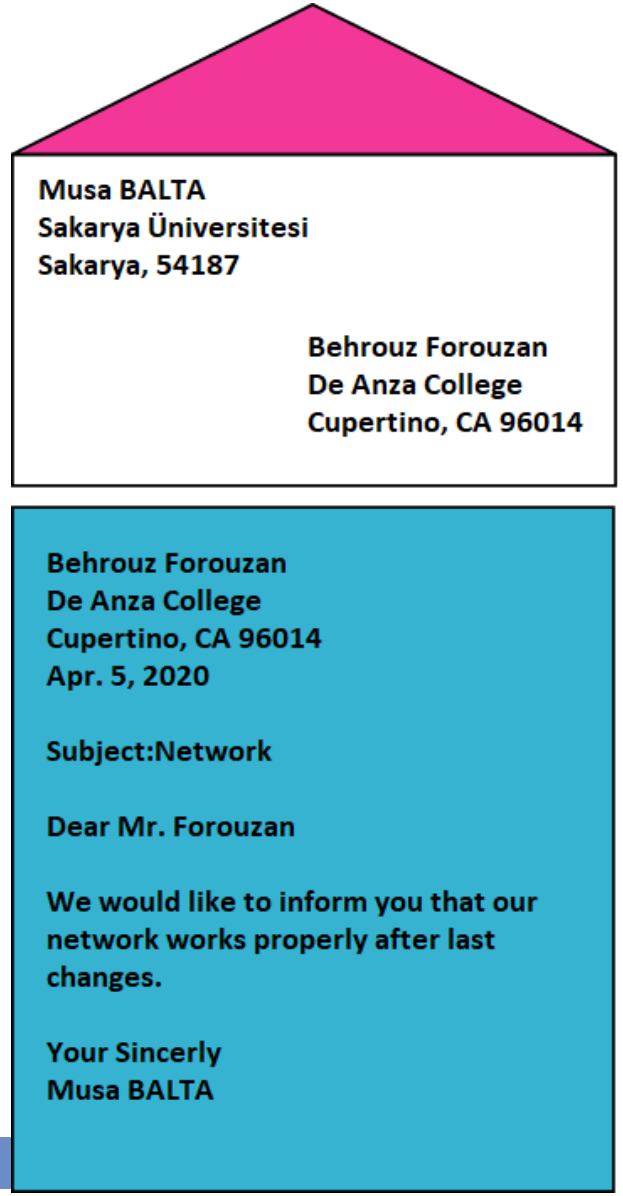
Basit Mail Transfer Protokolü (SMTP Modeli)

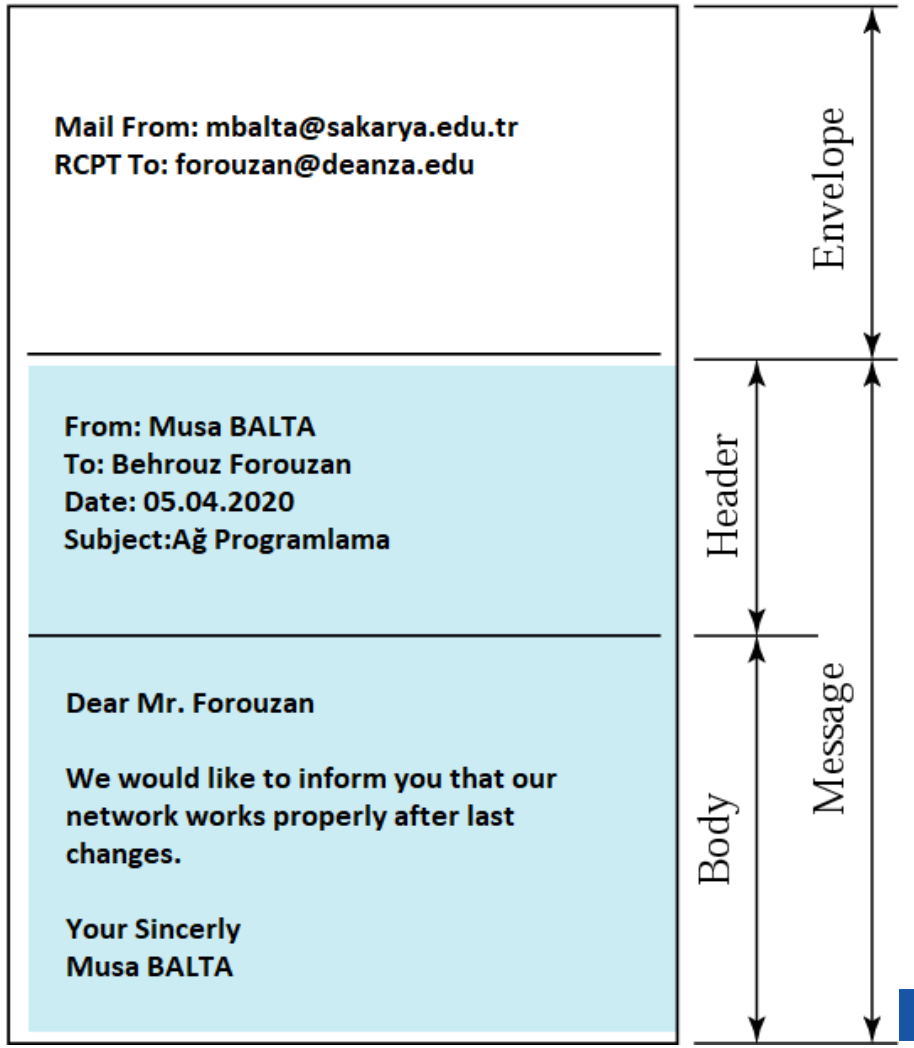
- ***User Agent-Kullanıcı Ajanı;*** Kullanıcı ajanı bir mesajı daha kolay almak veya göndermek için kullanıcılara servis sağlamaktadır.
- Kullanıcı ajan tipleri şu şekildedir:



- Kullanıcı ajanı olarak en çok kullanılan örnekler **mail, pine** ve **elm**'dir.

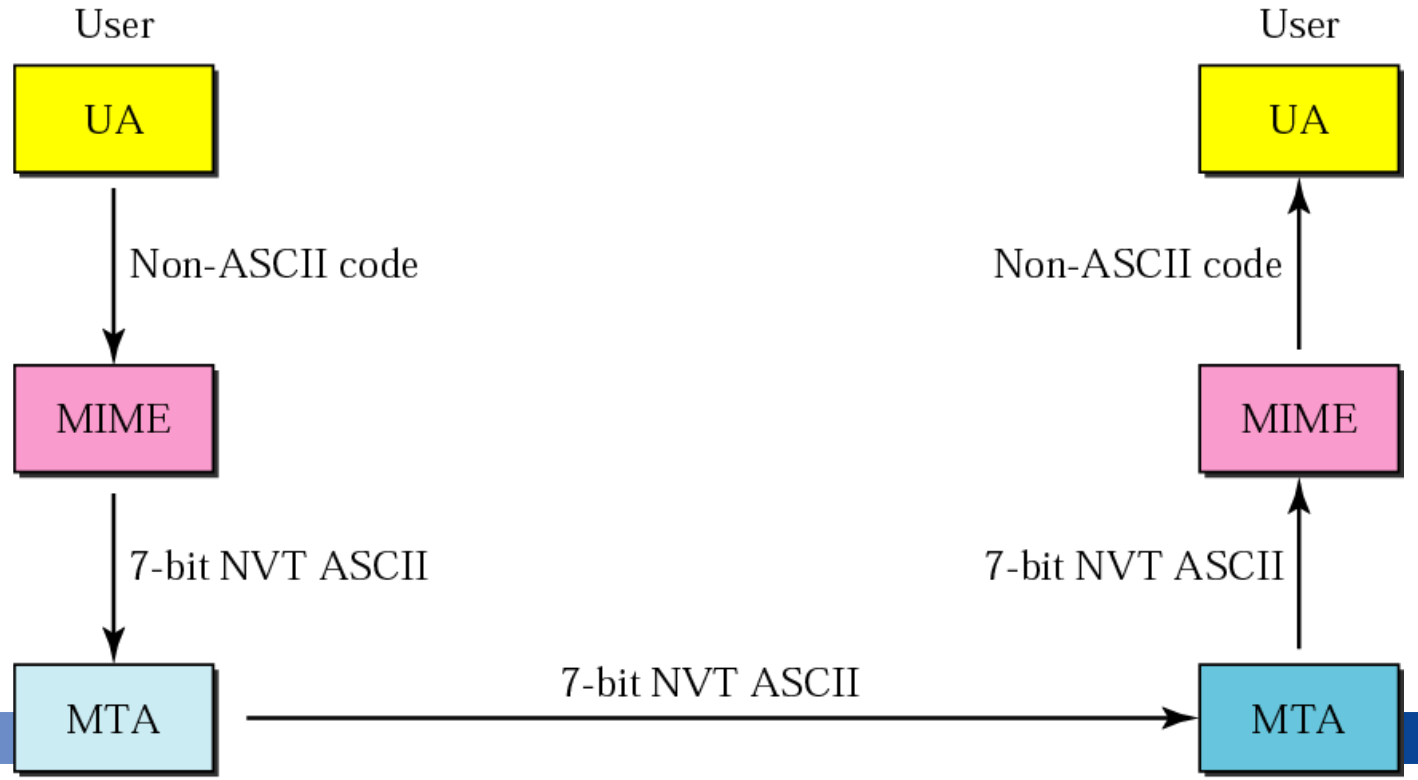
Basit Mail Transfer Protokolü (Mail Taslağı)





Basit Mail Transfer Protokolü (MIME)

- ***Multipurpose Internet Mail Extensions;*** E-posta uygulamaları aracılığıyla gönderilecek olan iletiye çeşitli türdeki içeriği eklemek için kullanılan bir İnternet standartıdır. MIME Basit Posta Aktarım Protokolü'nü hem metin hem de metin içerikli olmayan birden çok içerik eklenebilecek şekilde genişletir.



Basit Mail Transfer Protokolü (MIME)

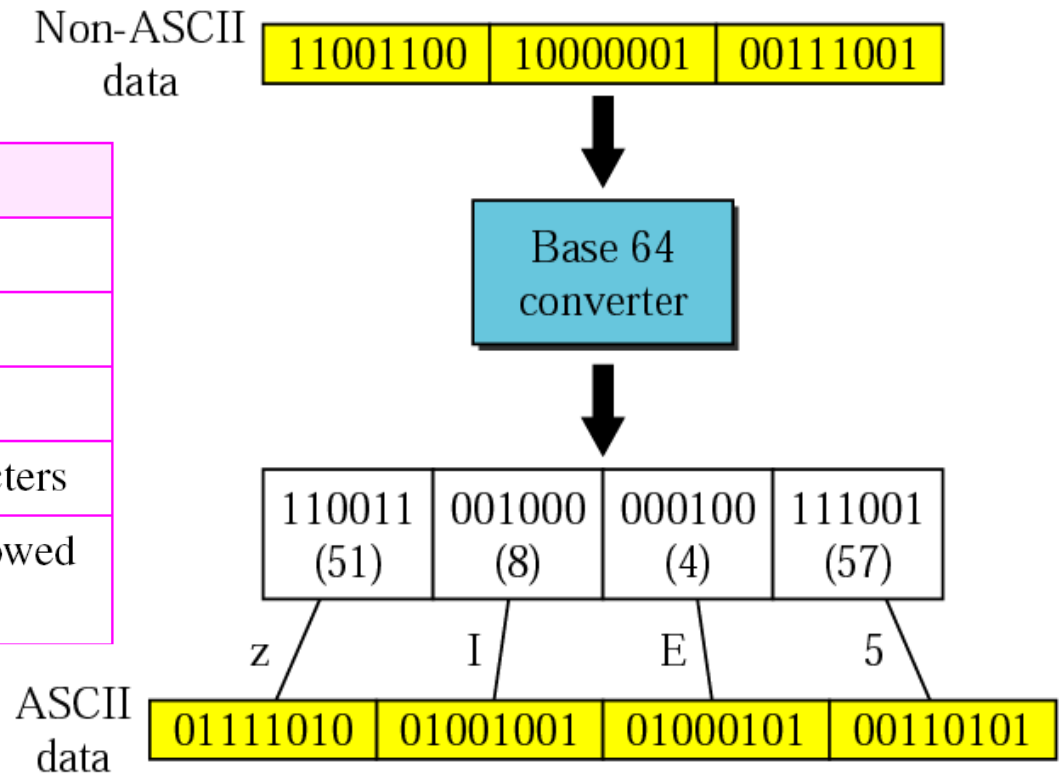
- **Veri Tipleri**

Type	*Subtype*	*Description*
Text	Plain	Unformatted
	HTML	HTML format
Multipart	Mixed	Body contains ordered parts of different data types
	Parallel	Same as above, but no order
	Digest	Similar to Mixed, but the default is message/RFC822
	Alternative	Parts are different versions of the same message
Message	RFC822	Body is an encapsulated message
	Partial	Body is a fragment of a bigger message
	External-Body	Body is a reference to another message
Image	JPEG	Image is in JPEG format
	GIF	Image is in GIF format
Video	MPEG	Video is in MPEG format
Audio	Basic	Single channel encoding of voice at 8 KHz
Application	PostScript	Adobe PostScript
	Octet-stream	General binary data (eight-bit bytes)

Basit Mail Transfer Protokolü (MIME)

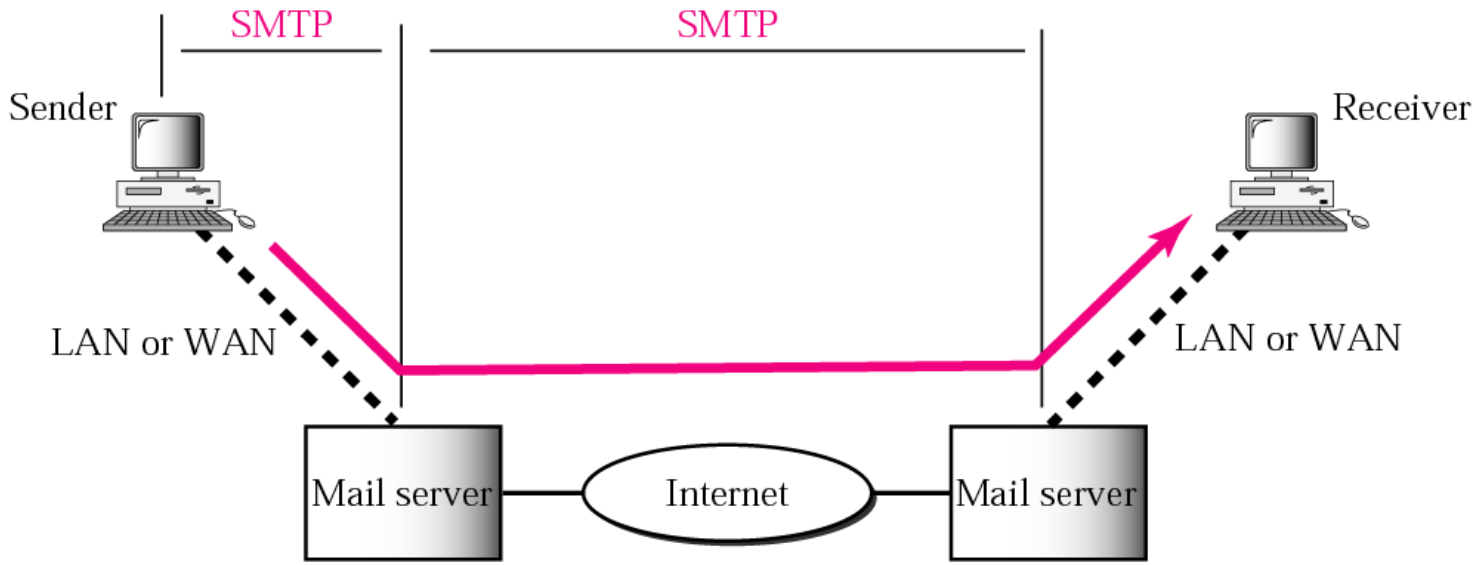
- **İçerik Transfer Kodlama**

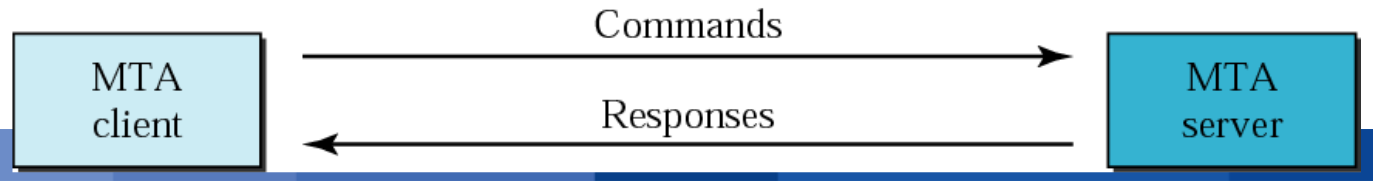
Type	*Description*
7bit	NVT ASCII characters and short lines
8bit	Non-ASCII characters and short lines
Binary	Non-ASCII characters with unlimited-length lines
Base64	6-bit blocks of data are encoded into 8-bit ASCII characters
Quoted-printable	Non-ASCII characters are encoded as an equal sign followed by an ASCII code



Basit Mail Transfer Protokolü (MTA)

- ***Mesaj Transfer Ajanı;*** Gerçek posta aktarımı için ileti aktarım aracıları (MTA'lar) gerekir. İnternet'teki MTA istemcisi ve sunucusunu tanımlayan protokole Basit Posta Aktarım Protokolü (SMTP) denir.





Basit Mail Transfer Protokolü (MTA-Komutlar)

Komutlar	Argümanlar
HELO	Gönderen Adı
MAIL FROM	Mesajın Göndereni
RCPT TO	Mesajın Alıcısı
DATA	Mail Gövdesi
QUIT	
RSET	
VRFY	Doğrulama İçin Alıcı İsmi
NOOP	
TURN	
EXPN	Genişletilecek Posta Listesi
HELP	Komut İsmi
SEND FROM	Mesajın Alıcısı
SMOL FROM	Mesajın Alıcısı
SMAL FROM	Mesajın Alıcısı

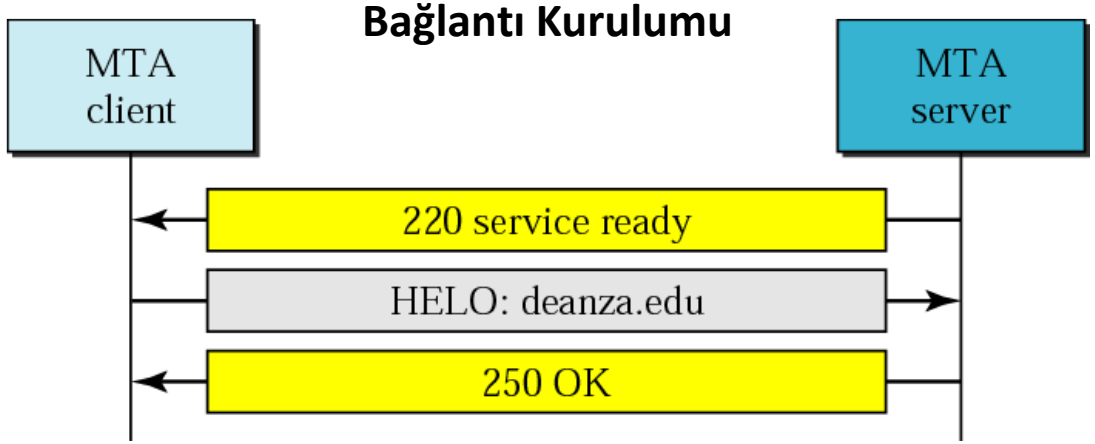
Basit Mail Transfer Protokolü (MTA-Cevap)

Kodlar	Tanımlar
Olumlu Cevaplar	
211	Sistem durumları veya yardım cevapları
214	Yardım mesajı
220	Servis hazır
221	İletişim kanalı sonlandırılıyor
250	İstek komutu tamamlandı
251	Kullanıcı yerel değil, mesaj iletilecek
Olumsuz Cevaplar	
421	Servis çalışmıyor
450	Mail kutusu çalışmıyor
451	Komut iptal edildi; lokal hata
452	Komut iptal edildi; yetersiz depolama alanı

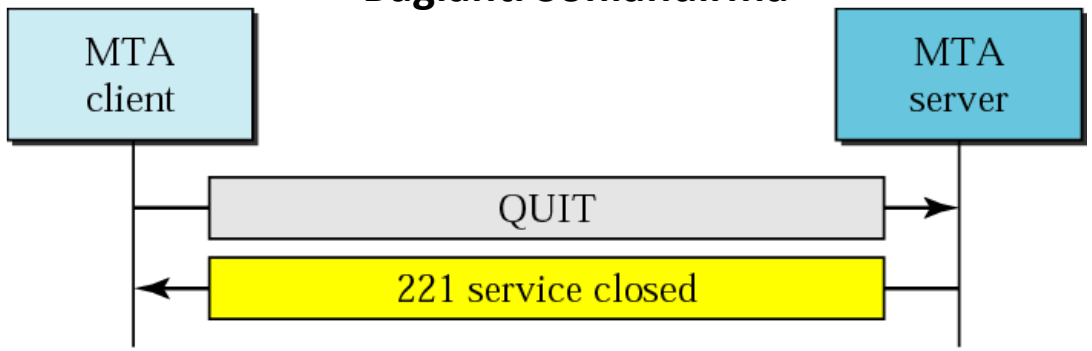
Kalıcı Olumsuz Cevaplar	
500	Syntax hatası; tanımlanamayan komut
501	Parametre veya argümanda syntax hatası
502	Komut uygulanmadı
503	Kötü komut dizisi
504	Komut geçici olarak uygulanmadı
550	Komut uygulanamıyor; mail kutusu servis dışı
551	Kullanıcı lokal değil
552	İstek engellendi; depolama aşımı
553	İstek işleme alınmadı
554	İşlem başarısız

Basit Mail Transfer Protokolü (Bağlantı Aşamaları)

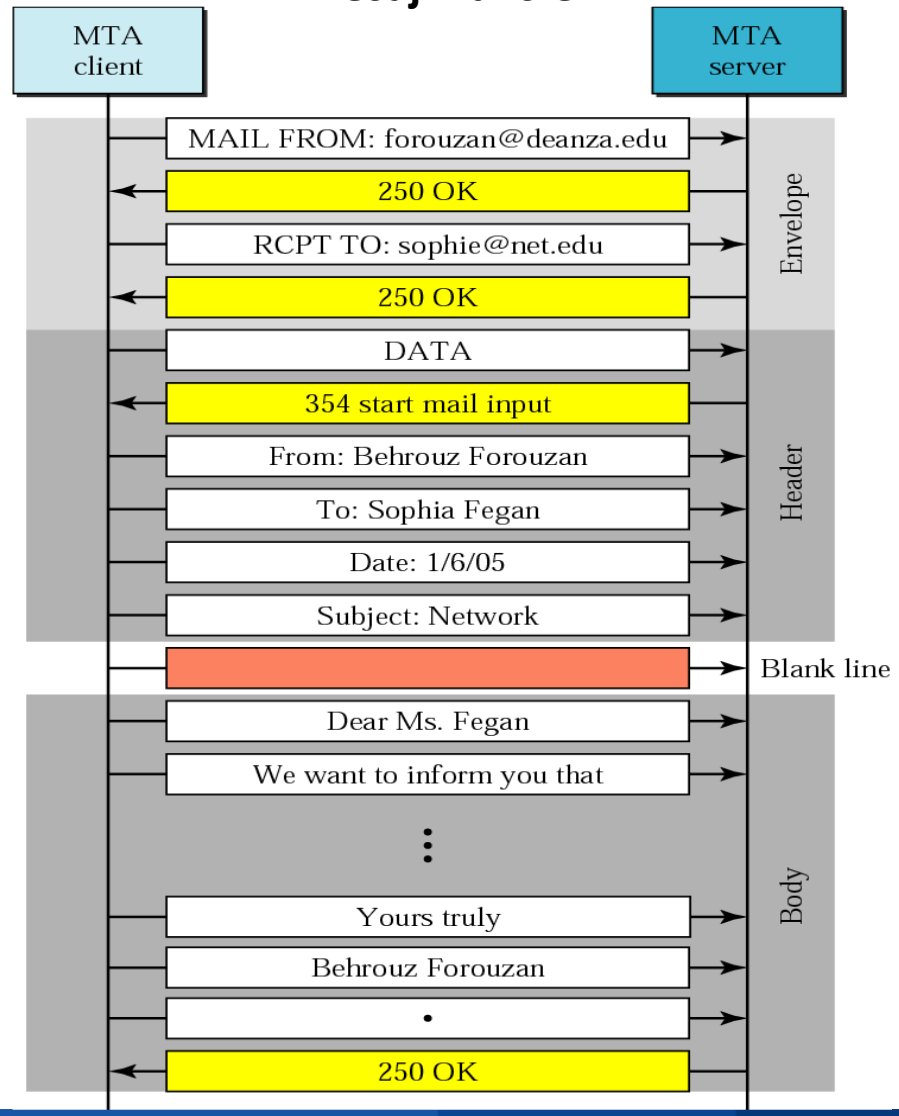
Bağlantı Kurulumu



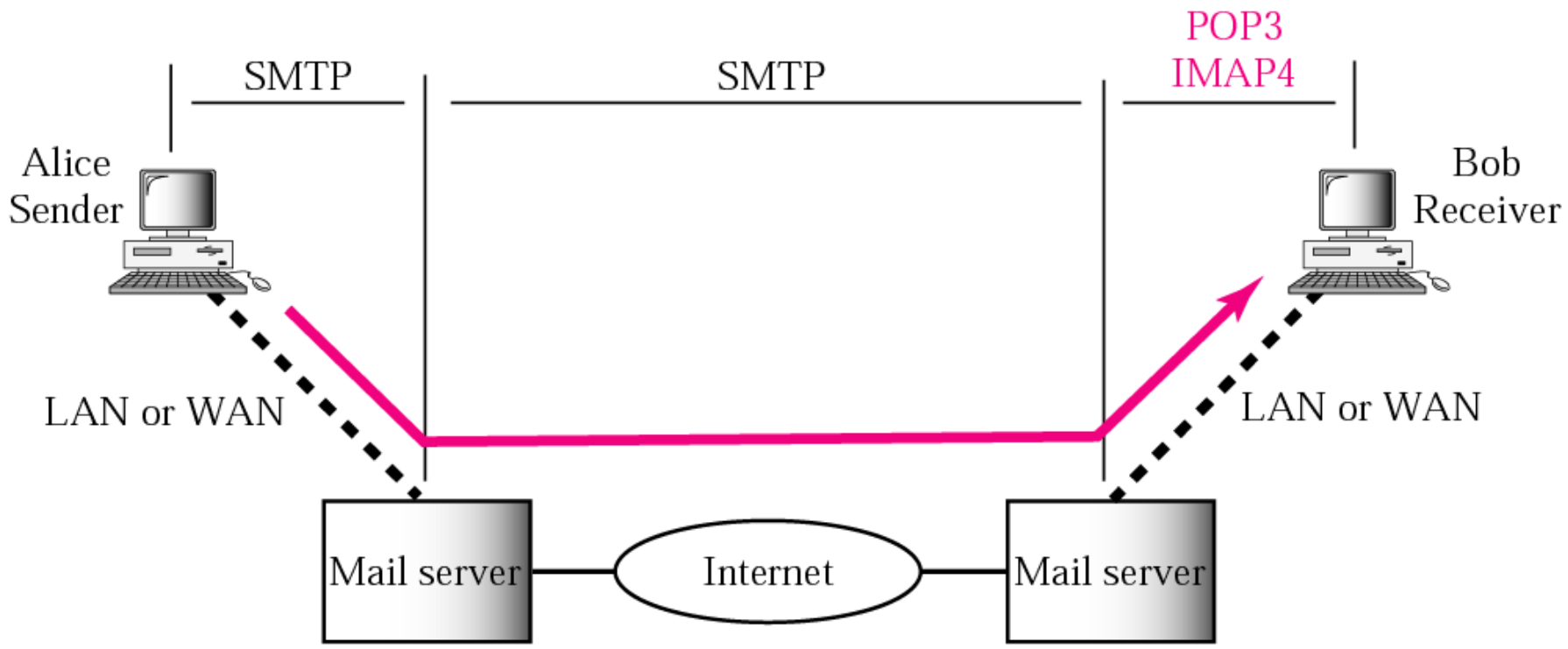
Bağlantı Sonlandırma



Mesaj Transferi

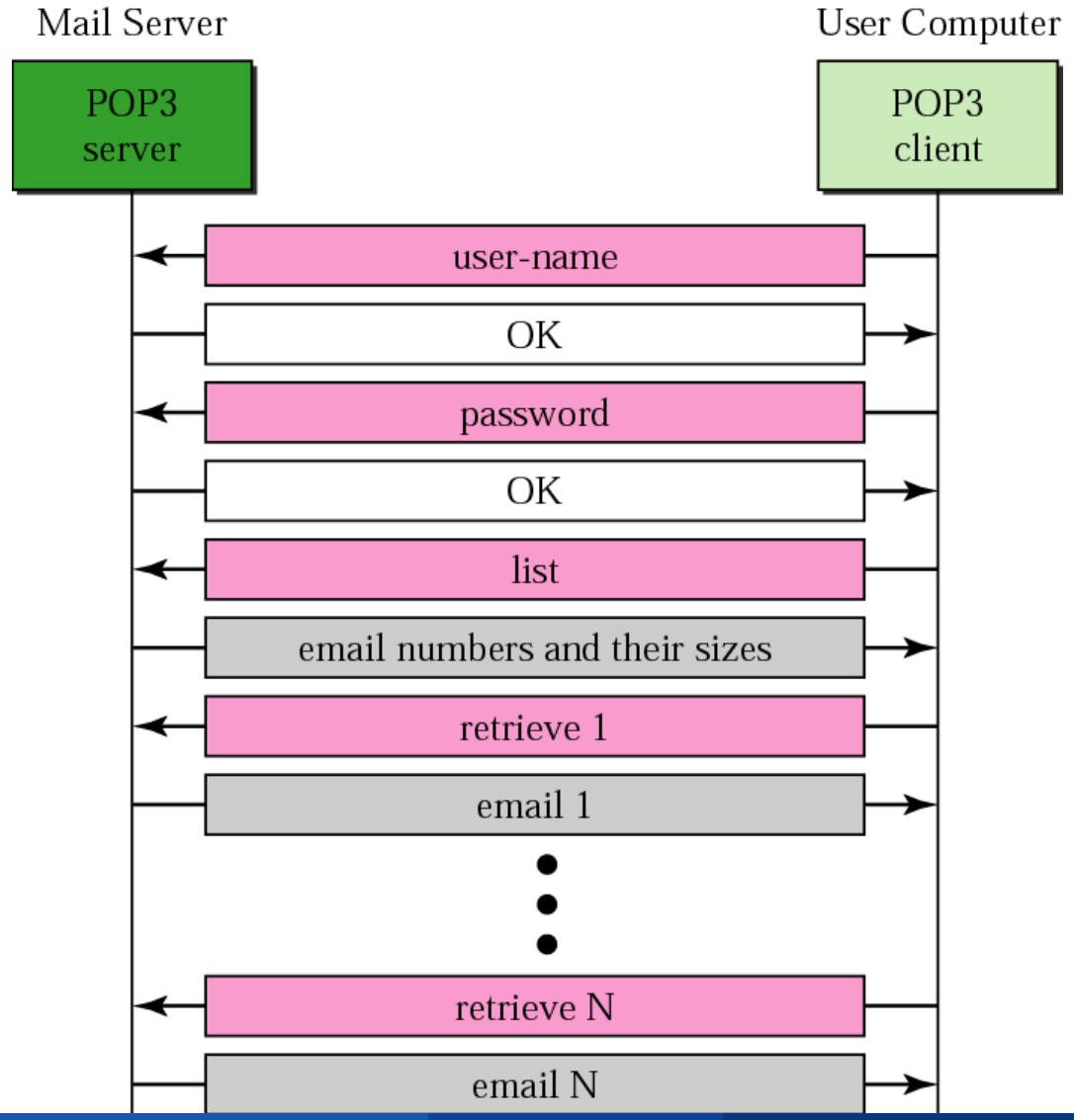


Basit Mail Transfer Protokolü (MAA: POP ve IMAP)



Basit Mail Transfer Protokolü (MAA: POP3)

- Postane Protokolü, e-posta istemcileri tarafından bir e-posta sunucusundan e-posta almak için kullanılan uygulama katmanı Internet standart protokolüdür.
- POP sürüm 3, yaygın olarak kullanılan sürümdür.
- POP, e-posta hizmetinizle iletişim kurarak ve tüm yeni iletilerinizi oradan indirerek çalışır. Kişisel bilgisayarınıza indirildikten sonra e-posta hizmetinden silinir.
- Gönderilmiş posta, e-posta sunucusunda değil PC veya Mac bilgisayarınızda yerel olarak depolanır.
-



Basit Mail Transfer Protokolü (MAA: IMAP)

- IMAP, bir e-posta iletişim protokolüdür. 1986 yılında Stanford Üniversitesi'nde geliştirilmiştir.
- IMAP4 olarak de bilinen IMAP, yerel kullanıcıların uzaktaki bir e-posta sunucusuna erişmesini sağlayan bir uygulama katmanı protokolüdür.
- En son sürümü IMAP sürüm 4 Revizyon 1 olup, RFC 3501'de tanımlanmıştır.
- IMAP kullanarak e-posta iletisi okurken, gerçekten indirme veya bilgisayarınızda depolama işlemi yapmazsınız.
- IMAP yalnızca üzerine tıkladığınız iletiyi indirir ve ekler otomatik olarak indirilmez. Bu şekilde, iletilerinizi POP yönteminden çok daha hızlı bir şekilde denetleyebilirsiniz.

Basit Ağ Yönetim Protokolü (Simple Network Management Protocol-SNMP)

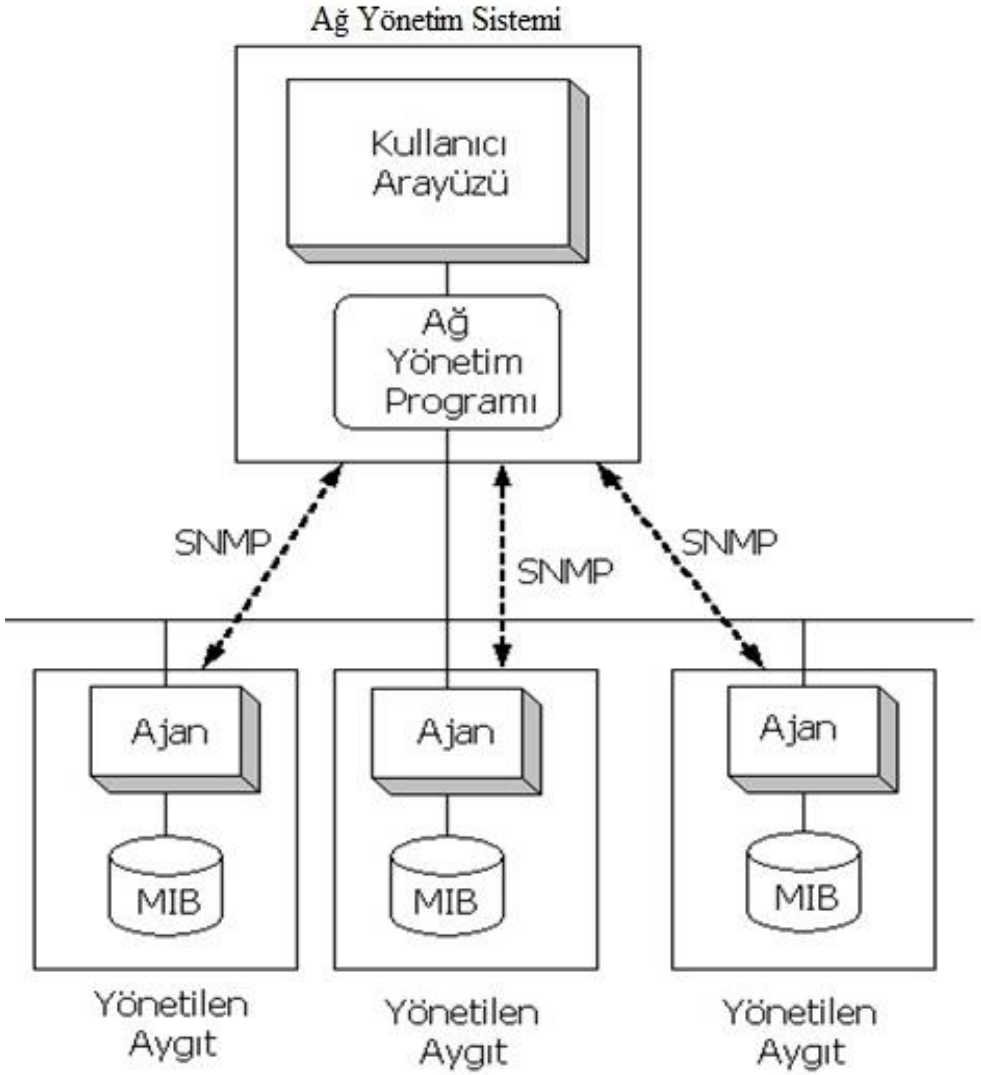
- Günümüz ağ topolojileri gerek büyüklükleri gerekse karmaşıklıklarından dolayı ağ yönetimini gerekli hale getirmişlerdir.
- Büyük ağlar, kendi kurumları için maliyet, zaman, performans, güvenilirlik ve güvenlik konularında çok önemli yere sahiptirler. Ancak yine aynı ağlar iyi yönetilemedikleri takdirde aynı başlıklar altında kurumlarına sıkıntı çıkarabilirler.
- Ağ yönetiminin 3 ana amacı vardır:
  - Ağın devamlılığı (çalışır halde tutmak)
  - Ağın performansını yönetmek
  - Maliyeti düşürme

Basit Ağ Yönetim Protokolü (SNMP)

- SNMP’inin ağ yönetimide sağlamış olduğu ***avantajlar şu şekildedir:***
  - Basit dizaynı kolay entegrasyon sağlıyor.
  - Kullanım alanı çok geniş
  - Güncelleme işlemi kolay
  - Artan gereksinimlere kolay adaptasyon
  - Bir standart olması
  - Genişletilebilir ve taşınabilir olması
  - Dağıtık ve merkezi yönetimi desteklemesi
- ***Dezavantaj olarak;***
  - UDP’yi kullandığından güvenilirlik düşük (son sürüm hariç)

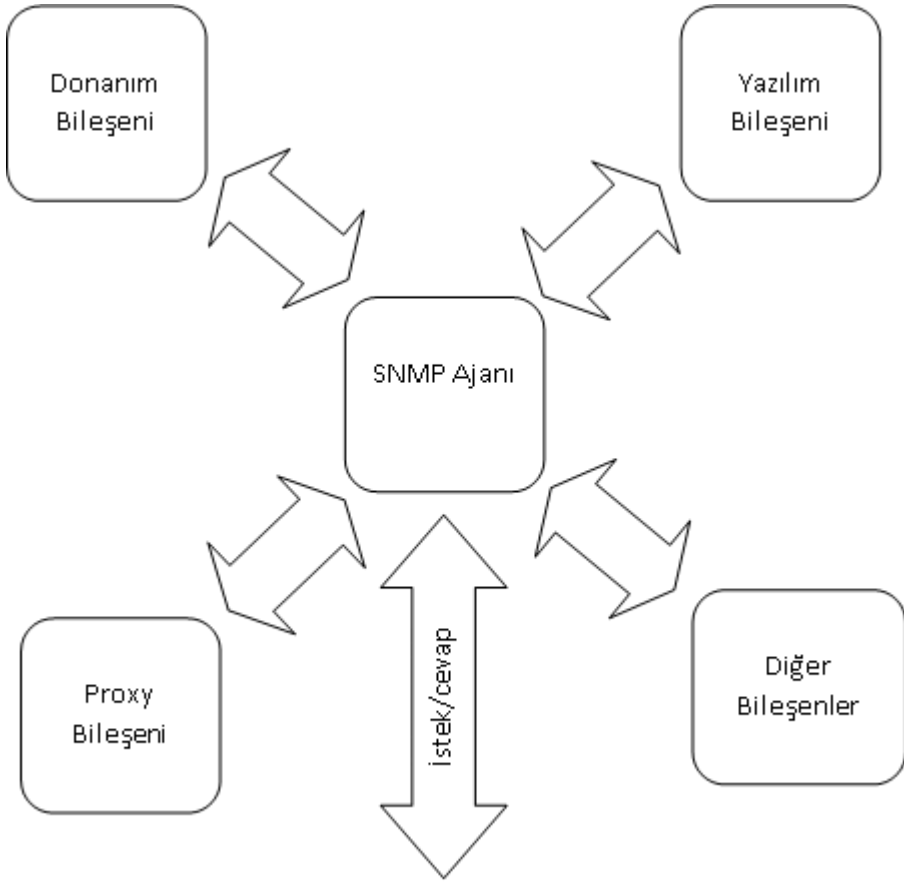
Basit Ağ Yönetim Protokolü (Mimari Yapı)

- SNMP mimarisi ağ yönetim sistemi, ajan ve yönetici kısmından oluşur;
- Üç temel bileşenden oluşan bu mimaride, mimarinin en alt seviyesinde cihazdan istenilen veriyi çekmeyi sağlayan ajan yazılımı bulunur.
- Orta seviyede ise ağ yönetim sistemi ile ajan arasındaki iletişimi kuran yönetici kısmı bulunur.
- Mimarinin en tepesinde ise tüm yönetim işlemlerinin yapıldığı ağ yönetim sistemi bulunur.



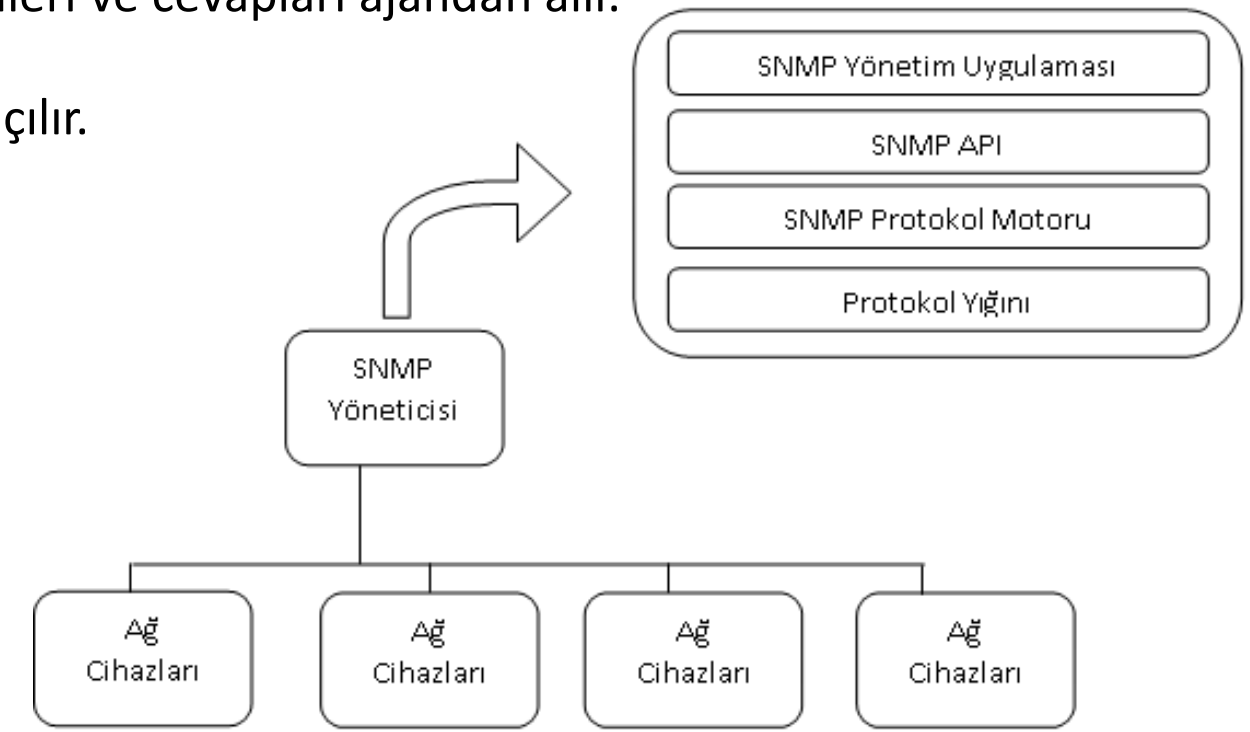
Basit Ağ Yönetim Protokolü (SNMP Ajanı)

- SNMP ajanı, kontrol veya takip edilen sistem düğümlerinden her birinde aktif edilen bir yazılımdır.
- Bu yazılım şekillendirmiş yapı içinde öğelerin her birine bir arayüz sağlar.
- Bu öğeler de yönetim bilgi tabanı dediğimiz MIB (Management Information Base, Yönetim Bilgi Tabanı)'lerde depolanır.
- Cihaz, üzerindeki tüm SNMP iletişimini kontrol eder.
- SNMP ajanı aktif edilmeden önce sistemin ne tarz bir ajana ihtiyacı olduğu tespit edilir. Bu da SNMP sürümü ile alakalıdır.



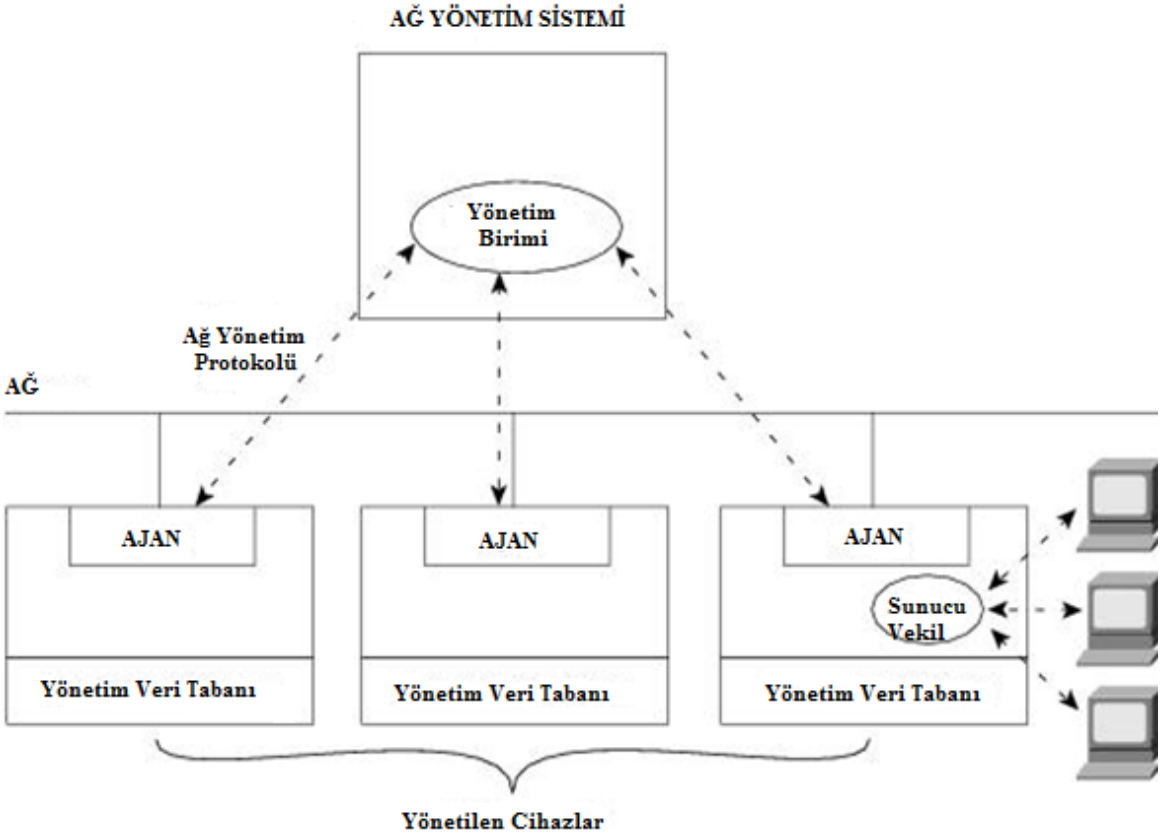
Basit Ağ Yönetim Protokolü (SNMP Yöneticisi)

- Ajan uygulamadan ihtiyaç duyulan bilgileri alıp kullanıcıya gösteren ve kullanıcının değiştirmek istediği değerleri cihaza gönderen yazılımdır.
- SNMP ajanına istek gönderir ve gerekli bildirimleri ve cevapları ajandan alır.
- SNMP yöneticisi istek gönderirken oturumlar açılır.



Basit Ağ Yönetim Protokolü (Ağ Yönetim Sistemi)

- Yönetici birimde çalışan ve bir ağa bağlı tüm cihazların izlenmesini ve yönetimini sağlayan uygulamaya verilen isimdir.
- SNMP ajanı ve SNMP yönetici arasındaki bilgi akışından iletişime kadar her türlü işlemi gerçekleştirir.



Basit Ağ Yönetim Protokolü (SNMP Sürümleri-SNMPv1)

- İlk SNMP sürümüdür. UDP, IP ve IPX protokolleri üzerinde çalışabilir.
- Çalışma mantığında ise SNMP, özetle bir sorgu-cevap protokolü olduğu için bu işlem, Get, GetNext, Set ve Trap komutları aracılığıyla olmaktadır.
- ***Get***, Ağ yönetim sistemi tarafından bir ya da daha fazla nesne bilgisi almak için kullanılır. Eğer yönetilen aygıt üzerinde çalışan ajan, istenen verilerin hepsini cevaplayamıyor ise ağ yönetim sistemine bir cevap yollamaz.
- ***Getnext işlemi*** tabloda yada ajan listesindeki bir sonraki değeri almak için kullanılır.
- ***Set işlemi*** ile yönetilen aygıtın MIB içerisindeki değerleri değiştirilebilir.
- ***Trap işlemi*** ise ağ yönetim sistemine, yönetilen aygıt tarafından oluşan değişiklikleri bildirmek için kullanılır.

Basit Ağ Yönetim Protokolü (SNMP Sürümleri-SNMPv2c)

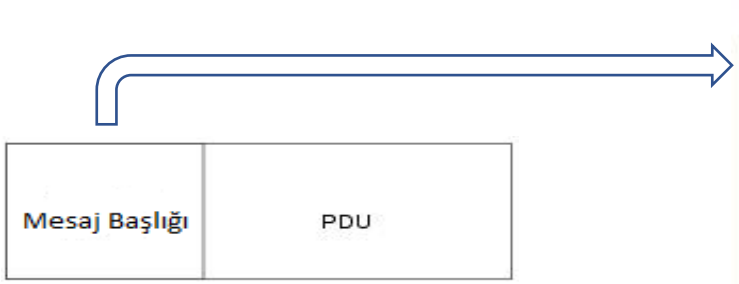
- SNMPv2'ye, SNMPv1'in evrimleştirilmiş hali diyebiliriz.
- Get, GetNext ve Set işlemleri SNMPv1 ile aynı olmasına rağmen SNMPv2'de trap işlemi biraz daha farklıdır. SNMPv2, v1'e göre iki yeni protokol işlemi daha içermektedir.
- ***GetBulk işlemi*** ile ağ yönetim sistemine büyük miktarda veri yollamak mümkündür. Eğer istenen veri bir paket boyutundan daha fazla ise ajan tarafından ard arda birkaç paket yollanır.
- ***Inform işlemi*** ise bir ağ yönetim sisteminin trap mesajlarını ağdaki başka bir ağ yönetim sistemine yollayabilmesi için kullanılır.
- SNMPv1'den farklı olarak eğer ajan yazılımı istenen değerlerin hepsini karşılayamıyorsa sisteme geri cevap döndürmemek yerine sadece sağlayabildiği mesajları gönderir.

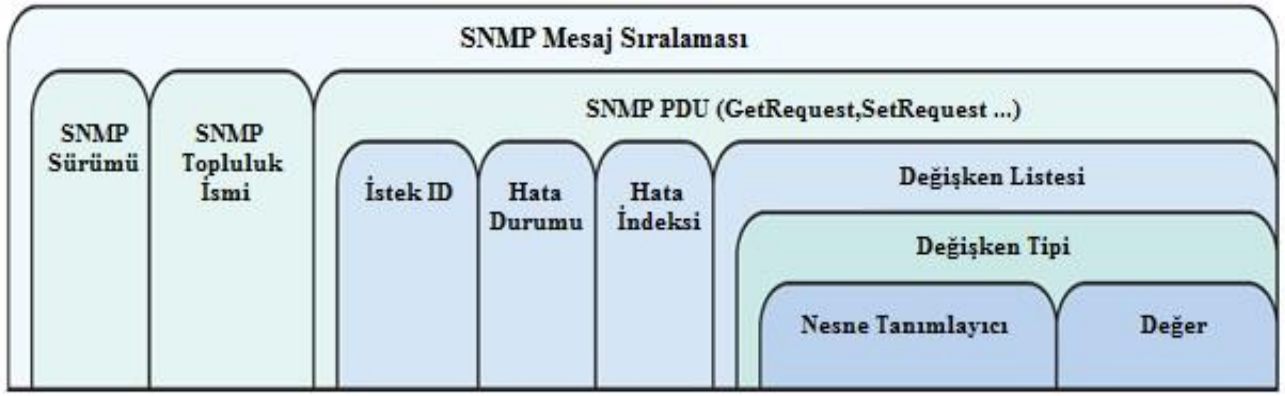
Basit Ağ Yönetim Protokolü (SNMP Sürümleri-SNMPv3)

- SNMPv3 önceki sürümlere göre güvenlik açısından daha gelişmiş olan bir sürümüdür.
- SNMPv3'te güvenlik düzeyi kavramı ortaya çıkmıştır. Bu düzeyler;
  - **noAuthNoPriv (Kimlik denetimi ve şifreleme yok)**; v1 ve v2c'ye karşılık gelen güvenlik düzeyidir. Sadece kullanıcı adı bazlı şifreleme işlemleri yapar. Bu yapısından dolayı güvenli değildir.
  - **authNoPriv (Kimlik denetimi var, şifreleme yok)**; kimlik denetimini kullanıcı adı ve şifre bazlı yapmasının yanı sıra MD5 veya SHA algoritmalarını kullanarak veri bütünlüğü de sağlar.
  - **authPriv (Kimlik denetimi ve şifreleme var)**; DES, 3DES ya da AES algoritmasını kullanarak sadece aynı anahtara sahip alıcıların çözebileceği bir şekilde veriyi şifreler.

Basit Ağ Yönetim Protokolü (SNMP Paket Yapıları)

- SNMP, paket yapısı açısından genel itibariyle iki yapıdan oluşur: Mesaj başlığı ve PDU
- **SNMP Mesaj Başlığı;**
- SNMP mesaj başlığı da 2 alan içerir: Sürüm numarası ve topluluk (community) ismi
- **Sürüm numarası**: Kullanılan SNMP sürümünü tanımlar**.**
- **Topluluk ismi:** Ağ yönetim sistemleri için erişim alanı tanımlar. Topluluk isimleri doğrulama (authentication) mekanizması gibi çalışır.





Basit Ağ Yönetim Protokolü (SNMP Paket Yapıları-devam)

Alan	Tanımı	Boyutu
Snmp Mesajı Sıralaması	Snmp sürümünü, topluluk ismini ve Snmp PDU'sunu belirten Snmp mesaj sırasını belirtir.	2 byte
Snmp Sürümü	Hangi sürümün kullanıldığını belirtir. Şekil 2.11 ve 2.12'de gösterilmiştir.	3 byte
Snmp Topluluk İsmi	Snmp cihazlarına güvenlik ekleyebilmek ve onlara kolay erişebilmek için tanımlanan "octet string"dir. Şekil 2.11 ve 2.12'de gösterilmiştir.	8 byte
Snmp PDU	Snmp mesajının ana bölümünü oluşturur. Farklı protokol veri birimlerini (PDU) tanımlar. Aşağıda PDU çeşitleri ve nasıl bir çerçeve yapısı kullanıldığı ve nasıl bir işlev sunduğu ayrıca anlatılacaktır	2 byte
İstek ID	Belirli Snmp isteklerini tanımlar. Bu index, Snmp yöneticisine uygun isteğe dönen cevapı eşletirme izni verir, Snmp ajan yazılımdan dönen cevabın yansıması gibidir.	3 byte
Hata Durumu	Snmp yöneticisinden gönderilen isteğe 0x00 değeri atanır. Sistemde bir hata varsa Snmp ajanı bu alanı değiştirir. 0x00—Hata yok 0x01—Dönen cevap aktarım için büyük. 0x02—İstenen nesne bulunamadı. 0x03—istekteki veri tipi, Snmp ajanındaki veri tipiyle eşleşmiyor. 0x04—Snmp yöneticisi sadece okuma parametresi atadı. 0x05—Genel Hata	3 byte
Hata İndeksi	Hata olursa, hataya neden olan nesne işaretlenir, diğer taraftan hata indeksi 0x00	3 byte
Değişken Listesi	Bu alanda SNMP PDU çeşidine göre veya uygulama alanına göre farklı değişkenler alabilir.	2 byte
Değişken Tipi	İki alandan oluşur. OID ve OID'nin değeri	2 byte
Nesne Tanımlayıcı (OID)	Snmp ajanındaki parametreleri ifade eder.	12 byte
Değer	SetRequest PDU – Değer, Snmp ajanındaki belirtilen OID'ye atanır. GetRequestPDU – Değeri bostur, dönen verinin izi gibi davranır.	2 byte

Basit Ağ Yönetim Protokolü (SNMP Çalışma Mekanizması)

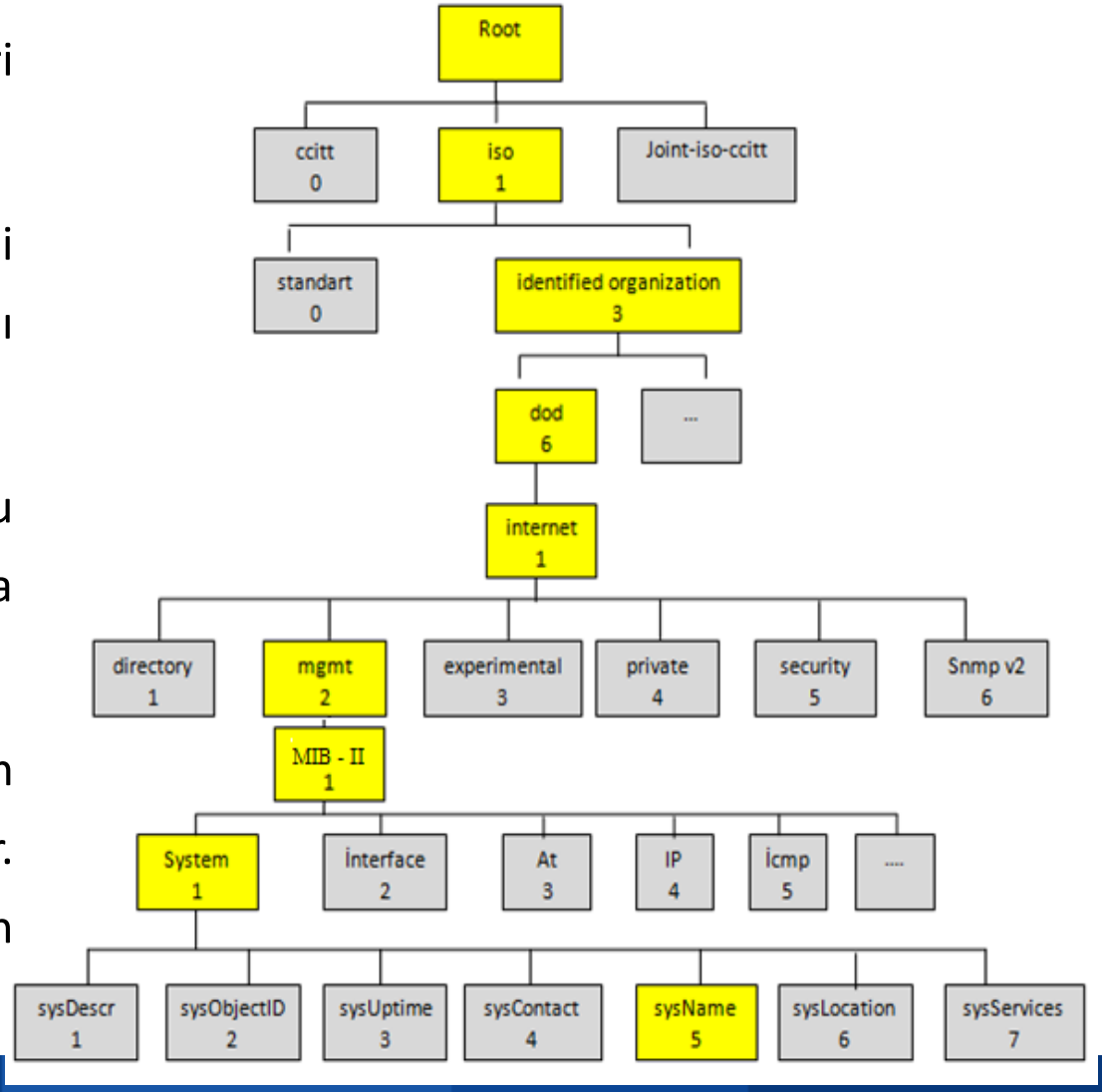
- SNMP'nin çalışma mekanizması istek gönderme ve isteğe cevap alma şeklindedir ve bunun için taşıma katmanında kullandığı protokol UDP'dir.
- Ağ yönetim sistemi, istekleri herhangi bir portundan, ajanın 161. portuna gönderir. Ajan geri bildirim için kendisine gelen istekleri 162. portundan gönderdiği cevaplarla sağlar.
- SNMP ajan yazılımı, cihazda herhangi fiziksel bir sorun oluştuğunda (cihaz üzerindeki fiziksel değerlere atanan değerlerin üzerine çıkıldığı zaman veya periyodik olarak veri gönderimi yapmak için ayarlandığı zaman) iletişimini kendi başlatır.
- SNMP sayesinde bir cihazdan bilgi alınabileceği gibi, cihazdaki bilgi değiştirilebilir ve cihazda yeni bir yapılandırma uygulanabilir. Örneğin cihaz baştan başlatılabilir, cihaza bir yapılandırma dosyası gönderilebilir ya da cihazdan alınabilir.

Basit Ağ Yönetim Protokolü (MIB Kavramı)

- MIB kavramı bir ağaç yapısına benzetilebilir. Ulaşılmak istenen değeri tutan değişkene OID (Object IDentifier, Nesne Tanımlayıcısı) adı verilir. MIB yapısındaki sıralamaya göre değer alır.
- Her kuruluşun, "**Internet Engineering Task Force *(*IETF*)*"** tarafından atanan bir değeri vardır, yani belirli bir yere kadar ağaç yapısı evrenseldir, ancak kurumların kendi kullanacakları yönetim nesneleri için bu kodu her kurum kendi tanımlar.
- Bu değişkenler ağacın dallarının en uç noktasında olup bir cihazla ilgili tek bir değeri tutabileceği gibi kendisinden sonra gelen bütün alt dalları ifade etmek için de kullanılabilir. Kökten ağaç dalına uzanan bu hiyerarşi birbirlerinden nokta ile ayrılmış sayı dizileriyle ifade edilir.

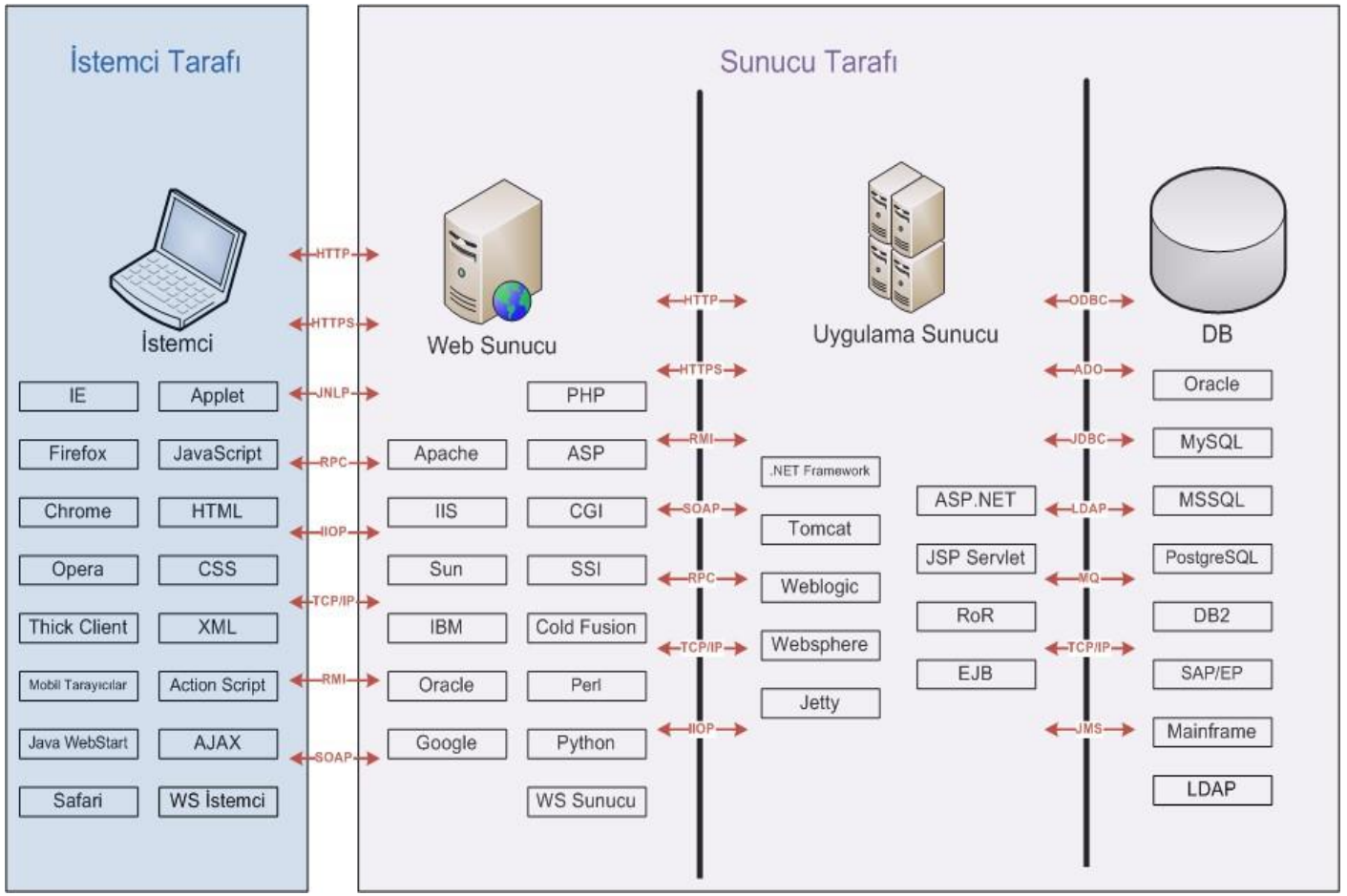
Basit Ağ Yönetim Protokolü (MIB Kavramı-devam)

- Şekil'de, OID değeri "1.3.6.1.2.1.1.5" olan "sysName" değeri ağaç yapısında gösterilmiştir.
- Buradaki ilk girdi de sysName.0 olarak adlandırılır. Yani komutta 1.3.6.1.2.1.1.5.0 yerine sysName.0 yazılırsa da aynı işlevi görür.
- Değişkenin başındaki ilk dört sayı, yani 1.3.6.1 standarttır. Bu noktadan sonra ulaşmak istediğimiz bilgiye göre alt dallara ilerlenir.
- Örneğin 1.3.6.1.2.1.1 dalı sistemle ilgili sistem adı, sistem tanımı, sistemin ayakta olduğu süre gibi değerleri tutar. Bunun alt dalı olan 1.3.6.1.2.1.1.5.0 değişkeni bunlardan biridir (sistem adı).



Hypertext Transfer Protocol (HTTP)

- HTTP, web üzerinde veriye erişebilmek ve onu yönetebilmek için kullanılan bir uygulama katmanı protokolüdür.
- HTTP taşıma katmanında TCP protokolünü kullanır ve port numarası 80'dir.
- HTTP, gösterildiği gibi ***asimetrik bir istek yanıtı istemci-sunucu*** protokolüdür.

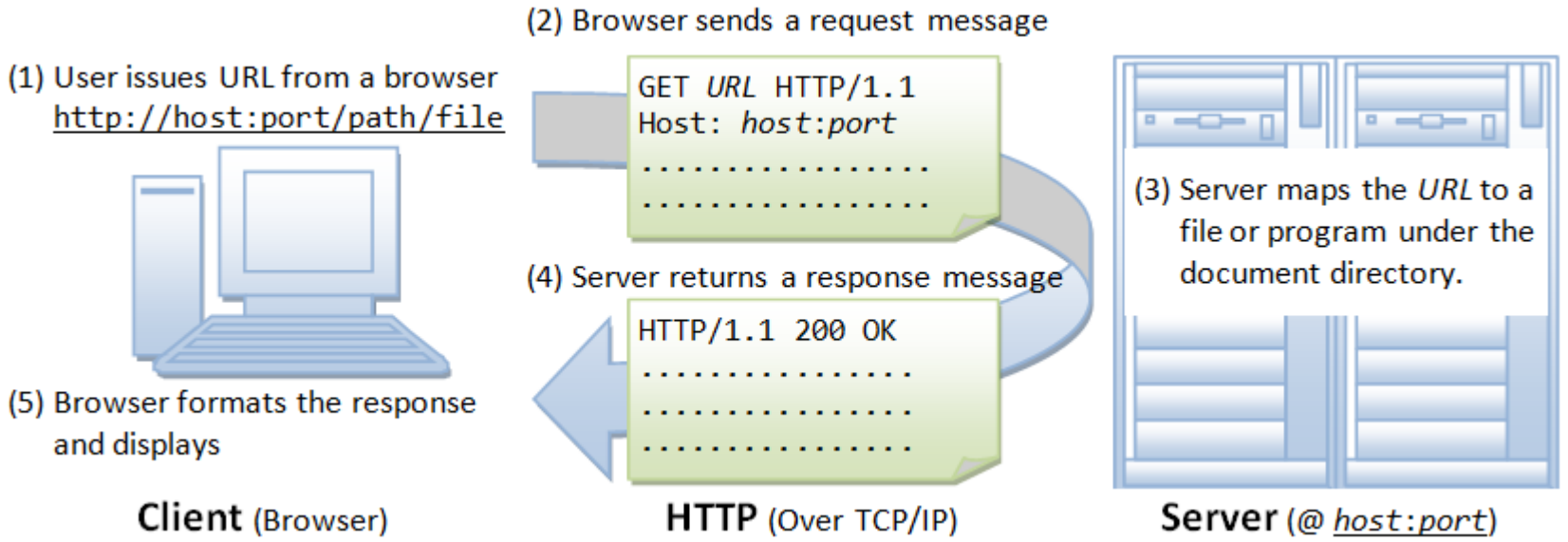


Hypertext Transfer Protocol (HTTP)

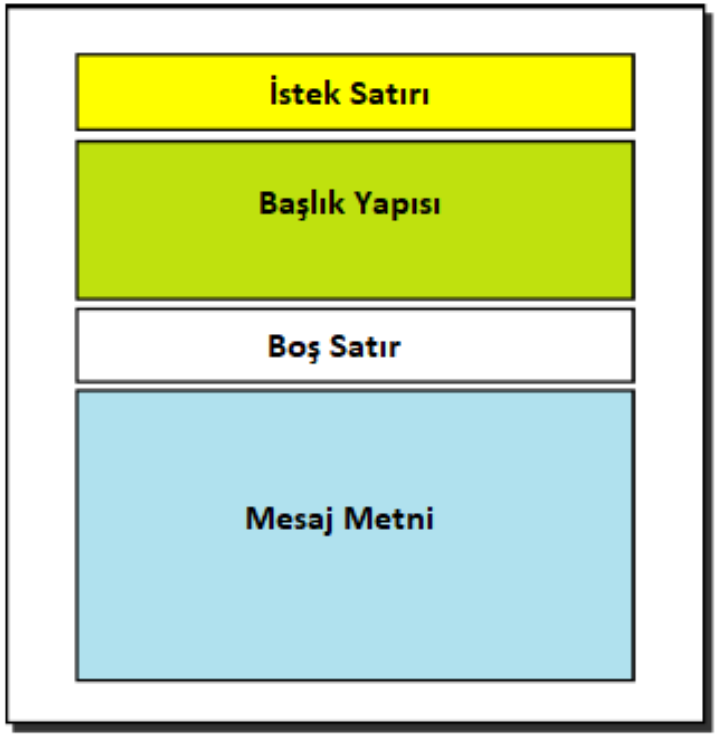
- Bir HTTP istemcisi bir HTTP sunucusuna bir istek iletisi gönderir. Sunucu sırayla bir yanıt mesajı döndürür. Başka bir deyişle, HTTP bir çekme protokolüdür, istemci bilgileri sunucudan alır (sunucu yerine bilgileri istemciye iletir).
- HTTP durum bilgisi olmayan bir protokoldür. Başka bir deyişle, mevcut istek önceki isteklerde ne yapıldığını bilmiyor.
- HTTP, sistemlerin aktarılan verilerden bağımsız olarak oluşturulmasına izin vermek için veri türü ve temsilinin görüşülmesine izin verir.
- ***RFC2616'dan alıntı:*** "Köprü Metni Aktarım Protokolü (HTTP), dağıtılmış, işbirlikçi, hiper ortam bilgi sistemleri için uygulama düzeyinde bir protokoldür. Köprü metin için kullanımının ötesinde birçok görev için kullanılabilen genel, durumsuz, ad sunucuları (dns) ve dağıtılmış nesne yönetim sistemleri, istek yöntemlerini, hata kodlarını ve başlıkları genişleten bir protokoldür.

Hypertext Transfer Protocol (HTTP-URL)

- HTTP kullanarak bir web kaynağı almak için tarayıcınızdan bir URL yayınladığınızda, ör. http://www.nowhere123.com/index.html, tarayıcı URL'yi bir istek iletisine dönüştürür ve HTTP sunucusuna gönderir. HTTP sunucusu istek mesajını yorumlar ve size istediğiniz kaynak veya bir hata mesajı olan uygun bir yanıt mesajı döndürür. Bu işlem aşağıda gösterilmiştir:



Hypertext Transfer Protocol (İstek-Cevap Mesajları)



İstek Mesajı

Cevap Mesajı

```
GET /doc/test.html HTTP/1.1
Host: www.test101.com
Accept: image/gif, image/jpeg, */*
Accept-Language: en-us
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0
Content-Length: 35

bookId=12345&author=Tan+Ah+Teck
```

İstek Satırı
İstek Başlığı
İstek Mesaj Başlığı
Ayırıcı Boş Satır
İstek Mesaj Metni

```
HTTP/1.1 200 OK
Date: Sun, 08 Feb xxxx 01:11:12 GMT
Server: Apache/1.3.29 (Win32)
Last-Modified: Sat, 07 Feb xxxx
ETag: "0-23-4024c3a5"
Accept-Ranges: bytes
Content-Length: 35
Connection: close
Content-Type: text/html

<h1>My Home page</h1>
```

Durum Satırı
Cevap Başlığı
Cevap Mesaj Başlığı
Ayırıcı Boş Satır
Cevap Mesaj Metni

HTTP İstek Methodları ve Durum Kodları

Method	Açıklama
GET	Sunucudan bir döküman istemek
HEAD	Döküman hakkında bilgi istemek (kendisini değil)
POST	İstemciden sunucuya bilgi göndermek
PUT	Sunucudan istemciye bilgi/döküman göndermek
TRACE	Gelen istekleri okuma
CONNECT	Saklı
OPTION	Mevcut seçenekler hakkında bilgi

Code	*Phrase*	*Description*
	Informational	
100	Continue	The initial part of the request has been received, and the client may continue with its request.
101	Switching	The server is complying with a client request to switch protocols defined in the upgrade header.
	Success	
200	OK	The request is successful.
201	Created	A new URL is created.
202	Accepted	The request is accepted, but it is not immediately acted upon.
204	No content	There is no content in the body.
	Redirection	
301	Moved permanently	The requested URL is no longer used by the server.
302	Moved temporarily	The requested URL has moved temporarily.
304	Not modified	The document has not been modified.
	Client Error	
400	Bad request	There is a syntax error in the request.
401	Unauthorized	The request lacks proper authorization.
403	Forbidden	Service is denied.
404	Not found	The document is not found.
405	Method not allowed	The method is not supported in this URL.
406	Not acceptable	The format requested is not acceptable.
	Server Error	
500	Internal server error	There is an error, such as a crash, at the server site.
501	Not implemented	The action requested cannot be performed.
503	Service unavailable	The service is temporarily unavailable, but may be requested in the future.

HTTP Mesaj Başlıkları

İstek Başlıkları

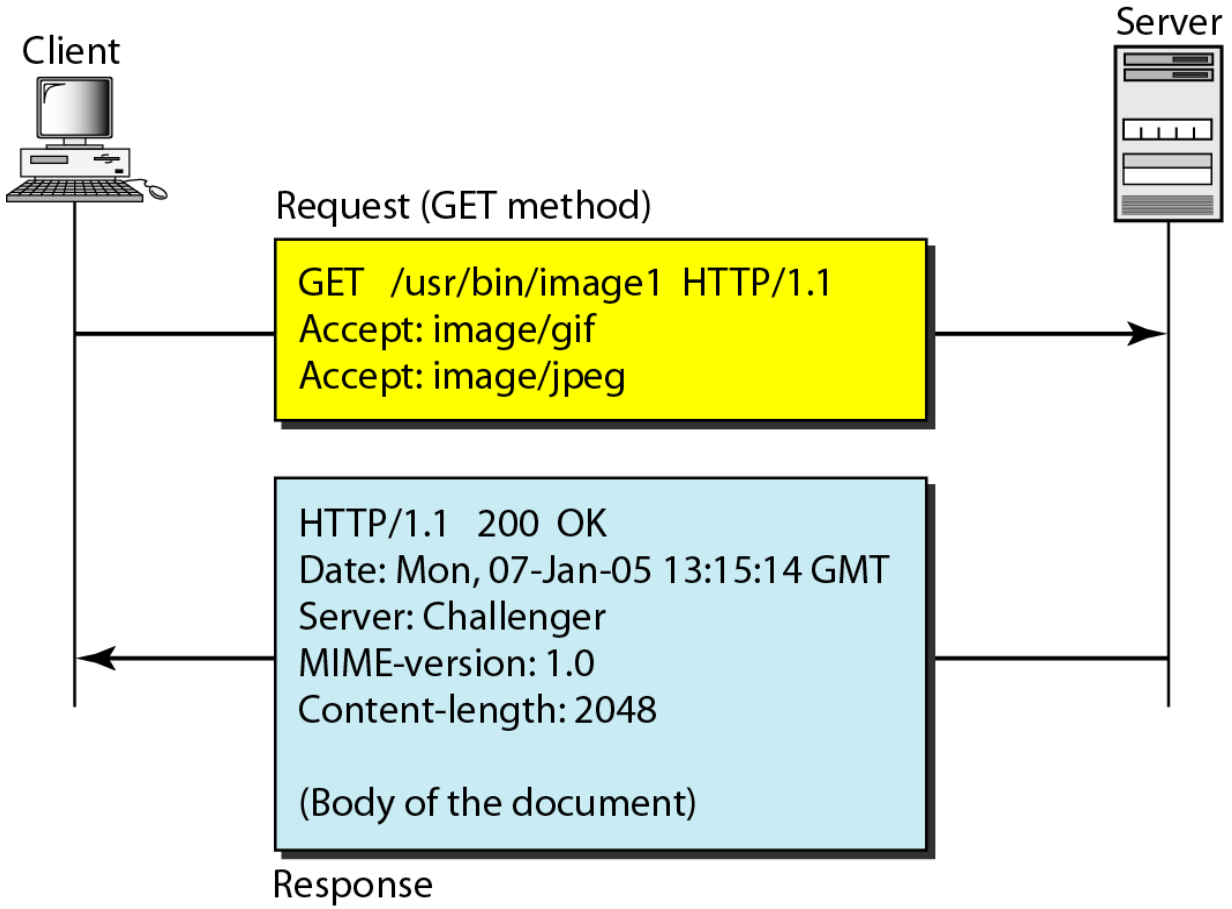
Header	*Description*
Accept	Shows the medium format the client can accept
Accept-charset	Shows the character set the client can handle
Accept-encoding	Shows the encoding scheme the client can handle
Accept-language	Shows the language the client can accept
Authorization	Shows what permissions the client has
From	Shows the e-mail address of the user
Host	Shows the host and port number of the server
If-modified-since	Sends the document if newer than specified date
If-match	Sends the document only if it matches given tag
If-non-match	Sends the document only if it does not match given tag
If-range	Sends only the portion of the document that is missing
If-unmodified-since	Sends the document if not changed since specified date
Referrer	Specifies the URL of the linked document
User-agent	Identifies the client program

Cevap Başlıkları

Header	*Description*
Accept-range	Shows if server accepts the range requested by client
Age	Shows the age of the document
Public	Shows the supported list of methods
Retry-after	Specifies the date after which the server is available
Server	Shows the server name and version number

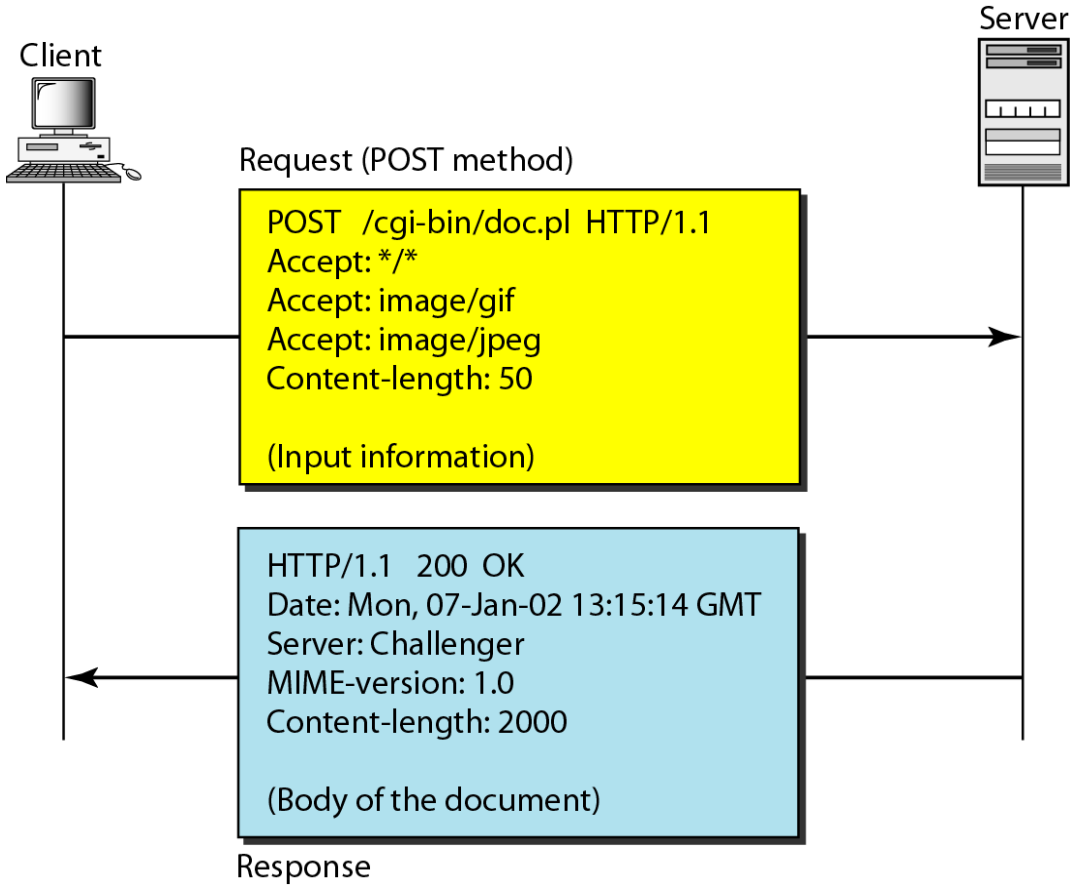
HTTP Örnek-1

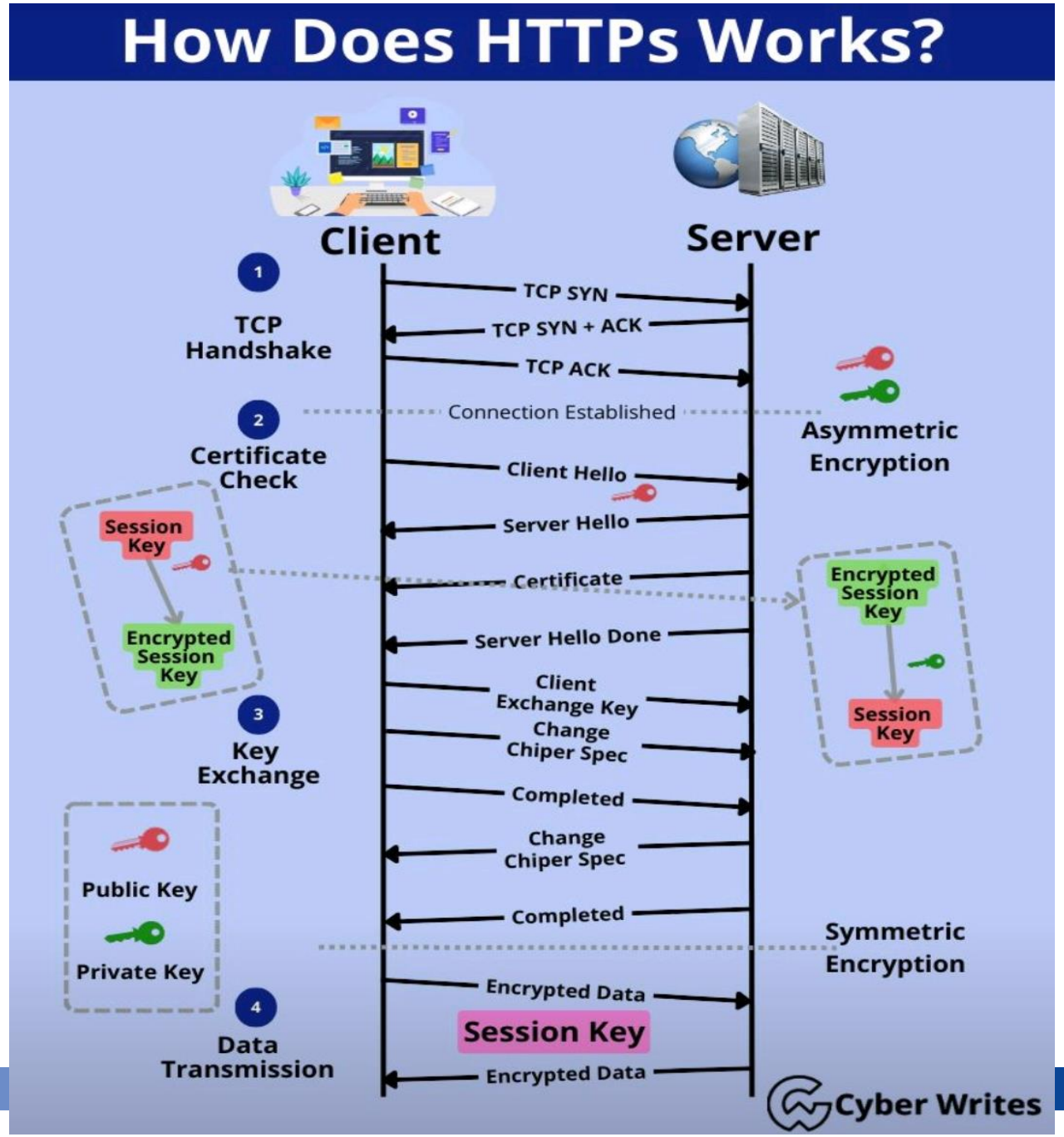
- ***Bu örnek bir belgeyi almaktadır.***
- / Usr / bin / image1 yoluna sahip bir görüntüyü almak için GET yöntemini kullanmaktadır.
- İstek satırı yöntemi (GET), URL'yi ve HTTP sürümünü (1.1) gösterir.
- Başlıkta, istemcinin resimleri GIF veya JPEG biçiminde kabul edebileceğini gösteren iki satır vardır.
- İsteğin bir gövdesi yoktur.
- Yanıt iletisi, durum satırını ve dört başlık satırını içerir.
- Başlık satırları belgenin tarihini, sunucusunu, MIME sürümünü ve uzunluğunu tanımlar.
- Belgenin gövdesi, başlığı izler.



HTTP Örnek-2

- ***Bu örnekte, istemci sunucuya veri göndermek istemektedir.***
- POST yöntemi kullanılmaktadır.
- İstek satırı yöntemi (POST), URL ve HTTP sürümünü (1.1) gösterir.
- Dört başlık satırı vardır.
- İstek gövdesi girdi bilgilerini içerir.
- Yanıt iletisi, durum satırını ve dört başlık satırını içerir.
- Bir CGI belgesi olan oluşturulan belge, gövde olarak eklenir.




How Does HTTPs Works?
Client
Server
1
TCP Handshake
TCP SYN
TCP SYN + ACK
TCP ACK
Connection Established
2
Certificate Check
Client Hello
Server Hello
Certificate
Server Hello Done
Session Key
Encrypted Session Key
Asymmetric Encryption
Encrypted Session Key
Session Key
3
Key Exchange
Client Exchange Key
Change Chiper Spec
Completed
Change Chiper Spec
Completed
Public Key
Private Key
Symmetric Encryption
4
Data Transmission
Encrypted Data
Session Key
Encrypted Data
Cyber Writes