

## Korumanın amacı

- İşletim sistemindeki **process'ler**, diğer process'lerin erişimine karşı **korunmalıdır**.
- **Protection (koruma)**, programların, process'lerin ve kullanıcıların kaynaklara erişimlerinin denetlenmesini ifade eder.
- Her programın **belirlenmiş kurallara göre sistem kaynaklarına erişmesi gereklidir**.
- **Protection-oriented sistemler yetkili ve yetkisiz kullanıcıları ayırt eder**.
- Kaynaklara **erişim kuralları** process'ler için ayrı ayrı olabilir ve **zamana göre değişebilir**.
- İşletim sistemlerinin yanı sıra, **kullanıcı uygulamaları da kendi koruma mekanizmalarını geliştirebilir**.

3

## Konular

- Korumanın amacı
- **Korumanın temelleri**
- Koruma alanı
- Erişim matrisi
- Erişim haklarının geri alınması
- Güvenlik problemi
- Program tehditleri
- Sistem ve ağ tehditleri
- Kriptolojinin güvenlik aracı olarak kullanımı
- Kullanıcı kimlik doğrulama
- Sistemler ve ağlar için firewall kullanımı

4

## Korumanın temelleri

- Protection için temel prensip, kullanıcılar, programlar ve sistemler **görevlerini yerine getirmelerini sağlayacak kadar hakka sahip olmalıdır (principle of least privilege).**
- Bir process'te ortaya çıkan hata, sadece kendi çalışmasını etkileyecektir.
- Her kullanıcı için ayrı **hesap (account) oluşturulması** ve ihtiyaçlarına göre erişim yetkisi verilmesi gereklidir.
- Bazı sistemler **rol tabanlı erişim denetimi** yapmaktadırlar.
- Sistemler, erişim kontrol listesi ile her bir servisi, **kullanıcı veya process'ler için enable/disable yaparlar.**

5

## Konular

- Korumanın amacı
- Korumanın temelleri
- **Koruma alanı**
- Erişim matrisi
- Erişim haklarının geri alınması
- Güvenlik problemi
- Program tehditleri
- Sistem ve ağ tehditleri
- Kriptolojinin güvenlik aracı olarak kullanımı
- Kullanıcı kimlik doğrulama
- Sistemler ve ağlar için firewall kullanımı

6

## Koruma alanı

- Bir bilgisayar sistemi process'ler ve nesneler topluluğudur.
- Nesneler, donanımsal (CPU, hafıza, yazıcılar, ...) ve yazılımsal (file, programlar, ...) olabilir.
- Her nesne ile yapılabilecek işlem türü farklıdır (CPU için execute, hafıza için yazma/okuma, dosya için açma/kapama, yazma/okuma, oluşturma/silme, ...).
- **Bir process kaynaklara kendisine verilen yetkiye göre erişmelidir.**
- **Bir process bir dosya kümesine erişim için kendisine verilmiş kurallara uymak zorundadır.**

7

## Koruma alanı

### Domain yapısı

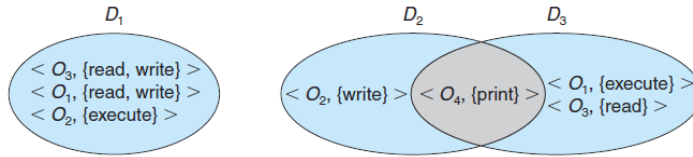
- **Protection domain, bir process'in erişebileceği kaynakları belirler.**
- Her **domain, bir grup nesneyi ve bu nesnelere erişim türünü** belirler.
- Bir nesne üzerinde yapılabilecek işlem kabiliyetine **erişim hakkı (access right)** denir.
- Bir domain sıralı bir ikili topluluğundan oluşur.  
<nesne adı, erişim kümesi>
- Eğer bir **D domain'i** <file F, {read, write}> erişim hakkına sahipse, D domain'inde çalışan bir process **F dosyasına okuma ve yazma amaçlı erişebilir.**
- Farklı domain'ler ortak erişim haklarına sahip olabilirler.

8

## Koruma alanı

### Domain yapısı

- Şekilde 3 farklı domain görülmektedir.



- $\langle O_4, \{print\} \rangle$  erişim hakkı,  $D_2$  ve  $D_3$  tarafından paylaşılmaktadır.
- Bir process ile domain ilişkilendirmesi **kalıcı (static)** veya **dinamik** olabilir.
- Dinamik domain ilişkilendirmesinde process, zamana göre domain değiştirebilir (**domain switching**).

9

## Konular

- Korumanın amacı
- Korumanın temelleri
- Koruma alanı
- Erişim matrisi
- Erişim haklarının geri alınması
- Güvenlik problemi
- Program tehditleri
- Sistem ve ağ tehditleri
- Kriptolojinin güvenlik aracı olarak kullanımı
- Kullanıcı kimlik doğrulama
- Sistemler ve ağlar için firewall kullanımı

10

## Erişim matrisi

- Protection için erişim matrisi (**access matrix**) oluşturulabilir.
- Matristeki **satırlar domain'leri**, **sütunlar ise erişim haklarını** gösterir.

object domain	$F_1$	$F_2$	$F_3$	printer
$D_1$	read		read	
$D_2$				print
$D_3$		read	execute	
$D_4$	read write		read write	

- Şekilde, üç dosya ve bir yazıcının 4 farklı domain tarafından erişimi düzenlenmiştir.
- Erişim matrisi **statik** veya **dinamik** yapıda olabilir.

11

## Erişim matrisi

- Erişim matrisinde **bir domain'den hangi domain'lere dinamik olarak geçiş yapılabileceği** de belirtilebilir.

object domain	$F_1$	$F_2$	$F_3$	laser printer	$D_1$	$D_2$	$D_3$	$D_4$
$D_1$	read		read			switch		
$D_2$				print			switch	switch
$D_3$		read	execute					
$D_4$	read write		read write		switch			

- Bir domain'deki process'lerin tüm yaşam döngüleri boyunca hangi domain'lere geçiş yapabileceği sınırlandırılabilir.

12

## Erişim matrisi

- Erişim matrisinde, **copy**, **owner** ve **control** işlemleri ile değişiklikler tanımlanabilir.
- copy**, bir domain'in kendisine ait erişim yetkisini başka domain'e kopyalayabileceğini gösterir.
- Bir domain'in erişim yetkisini aktarabileceği \* ile gösterilir.

object \ domain	$F_1$	$F_2$	$F_3$
$D_1$	execute		write*
$D_2$	execute	read*	execute
$D_3$	execute		

object \ domain	$F_1$	$F_2$	$F_3$
$D_1$	execute		write*
$D_2$	execute	read*	execute
$D_3$	execute	read	

- Şekilde,  $D_2$  sahip olduğu **read** yetkisini  $D_3$ 'e kopyalamıştır.

13

## Erişim matrisi

- owner**, bir domain'in erişim yetkisi ekleyip çıkarabileceğini gösterir.
- Bir domain **owner** olduğu sütunda, başka domain'e erişim yetkisi ekleyip çıkartabilir.

object \ domain	$F_1$	$F_2$	$F_3$
$D_1$	owner execute		write
$D_2$		read* owner	read* owner write
$D_3$	execute		

object \ domain	$F_1$	$F_2$	$F_3$
$D_1$	owner execute		write
$D_2$		owner read* write*	read* owner write
$D_3$		write	write

- Şekilde,  $D_2$  owner olduğu için **write** yetkisini eklemiştir.

14

## Erişim matrisi

- **control**, bir domain'in **başka bir domain'in erişim yetkisini değiştirebileceğini gösterir.**

object domain	$F_1$	$F_2$	$F_3$	laser printer	$D_1$	$D_2$	$D_3$	$D_4$
$D_1$	read		read			switch		
$D_2$				print			switch	switch control
$D_3$		read	execute					
$D_4$	write		write		switch			

- Şekilde,  $D_2$  control yetkisine sahip olduğundan  $D_4$  domain'inin yetkilerini değiştirebilir.

15

## Konular

- Korumanın amacı
- Korumanın temelleri
- Koruma alanı
- Erişim matrisi
- Erişim haklarının geri alınması
- Güvenlik problemi
- Program tehditleri
- Sistem ve ağ tehditleri
- Kriptolojinin güvenlik aracı olarak kullanımı
- Kullanıcı kimlik doğrulama
- Sistemler ve ağlar için firewall kullanımı

16

## Erişim haklarının geri alınması

- Dinamik koruma sistemlerinde, bir nesneye **erişim haklarının başka kullanıcılardan alınması gerekebilir.**
- Erişim haklarının alınması farklı şekillerde olabilir:
  - **Immediate / delayed:** Haklar **hemen geri alınabilir** veya **belirli bir süre sonunda alınabilir.**
  - **Selective / general:** Bir nesneden erişim hakkı alındığında bu nesneye erişen **tüm kullanıcılardan alınabilir** veya **bir grup kullanıcıdan alınabilir.**
  - **Partial / total:** Bir nesneden **bazı erişim hakları alınabilir** veya **tüm erişim hakları alınabilir.**
  - **Temporary / permanent:** Bir nesneden **erişim hakları geçici olarak** veya **kalıcı olarak alınabilir.**

17

## Konular

- Korumanın amacı
- Korumanın temelleri
- Koruma alanı
- Erişim matrisi
- Erişim haklarının geri alınması
- **Güvenlik problemi**
- Program tehditleri
- Sistem ve ağ tehditleri
- Kriptolojinin güvenlik aracı olarak kullanımı
- Kullanıcı kimlik doğrulama
- Sistemler ve ağlar için firewall kullanımı

18



## Güvenlik problemi

- **Protection**, bilgisayar sistemindeki veri ve programlara denetimli erişimi sağlar.
- **Security**, yeterli düzeyde protection'ın yanı sıra sistemin çalıştığı dış ortamı da göz önüne alır.
- Protection sistemi, bir programın yetkisiz kişi tarafından çalıştırılması halinde etkisiz kalır.
- Bilgisayar kaynakları, yetkisiz erişimlere, kötücül işlemlere, hataların oluşmasına karşı korunmalıdır (CPU, hafıza, diskler, ağ, ...).
- Bir sistemin kaynaklarına tüm şartlar altında planlanan/amaçlanan erişim ve kullanım sağlanıyorsa bu sistem güvenlidir (secure) denir.
- Bir sistemdeki güvenlik ihlali, kasıtlı (kötücül) olarak veya kasıtsız (bilmeyerek) ortaya çıkabilir.

19

## Güvenlik problemi

- Bir sistemi, kasıtsız ortaya çıkan kötü kullanımlara karşı korumak, kasıtlı olanlara göre daha kolaydır.
- Protection yöntemlerinin çoğu kasıtsız kullanımlara karşı geliştirilmiştir.
- Bir sistemdeki güvenliği kırmaya yönelik girişimde bulunanlara intruder veya cracker (saldırgan) denir.
- **Threat**, güvenlik ihlali olma potansiyelini gösterir.
- **Attack**, güvenlik kırmaya yönelik girişimi ifade eder.

20

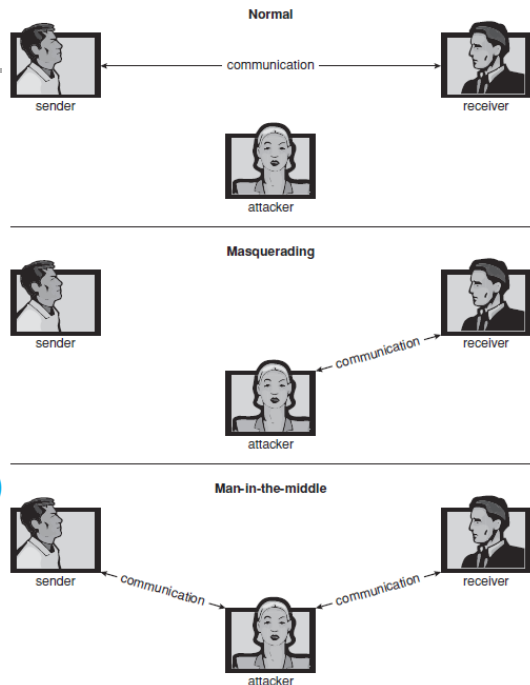
## Güvenlik problemi

- Güvenlik ihlalleri farklı şekillerde olabilir:
  - **Gizlilik ihlali:** Bir **verinin yetkisiz kişi tarafından okunmasıdır** (kimlik bilgileri, kredi kartı bilgileri, ...).
  - **Bütünlüğün bozulması:** Bir **veri veya programda yetkisiz değişiklik yapılmasıdır** (veri içeriğinin değiştirilmesi, program kodunun değiştirilmesi).
  - **Kullanılabilirliğin engellenmesi:** Bir **verinin içeriğinin değiştirilerek kullanılamaz hale getirilmesidir** (Web sayfalarının içeriğinin değiştirilmesi).
  - **Servis çalınması:** **Kaynakların yetkisiz kullanılmasıdır** (Saldırgan bilgisayara bir program yükler ve dosya sunucusu yapabilir).
  - **Servis engellenmesi:** **Sistemin servis sağlayamayacak hale getirilmesidir** (Denial-of-Service (DOS) saldırıları).

21

## Güvenlik problemi

- Saldırganlar genellikle başka birisi gibi davranarak (**masquerading**) sistemlere saldırırlar.
- Saldırganlar kimlik doğrulamayı kırarak (**authentication**) saldırırlar ve yetkileri olmayan işlemleri gerçekleştirirler.
- Saldırganlar gönderici ve alıcı arasındadır (**man-in-the-middle**) ve tüm veriyi ele geçirir.



## Güvenlik problemi

- Bir sistemin korunması için 4 seviyeli güvenlik önlemi alınmalıdır:
  - **Physical:** Bilgisayar sistemi fiziksel olarak güvenli bir yerde bulunmalıdır ve yetkisiz erişimler engellenmelidir.
  - **Human:** Uygun kullanıcılara sisteme erişim yetkisi verilmelidir. Bu kullanıcılar sosyal mühendislik ile aldatılabilirler (phishing – şifre avcılığı).
  - **Operating system:** Sistem kendisini kasıtlı veya kasıtsız güvenlik ihlallerinden korumalıdır (DOS saldırısı, şifre çalınması).
  - **Network:** Ağ üzerinde işlem yapan sistemlere yetkisiz erişim yapılabilir veya başka kullanıcılara servis veremeyecek hale getirilir.
- Fiziksel ve insan düzeyinde güvenlik zaafiyeti varsa, alt boyutlardaki güvenlik önlemleri etkisiz kalır.
- Bir sistemin güvenliği, en zayıf noktanın güvenliği kadardır.

23

## Konular

- Korumanın amacı
- Korumanın temelleri
- Koruma alanı
- Erişim matrisi
- Erişim haklarının geri alınması
- Güvenlik problemi
- **Program tehditleri**
- Sistem ve ağ tehditleri
- Kriptolojinin güvenlik aracı olarak kullanımı
- Kullanıcı kimlik doğrulama
- Sistemler ve ağlar için firewall kullanımı

24

## Program tehditleri

### *Trojan horse*

- Saldırganların yaygın amacı, güvenlik ihlali oluşturmak veya bir process'in normal yaptığı işi değiştirmektir.
- Trojan horse, sistem üzerinde kötücül amaçlı çalışan kodu ifade eder.
- Spyware (casus yazılımı), Trojan atının bir varyasyonudur.
- Spyware, genellikle freeware veya shareware yazılımla yüklenir. Bazen ticari yazılımlarda spyware içerebilir.
- Spyware, kullanıcıya reklam gösterme, kullanıcı bilgilerini elde etme ve başka noktaya gönderme gibi işlevleri yapabilir.

25

## Program tehditleri

### *Trap door*

- Program tasarımcısı, kendisinin kullanabileceği bir boşluk (trap door) bırakır.
- Örneğin, belirli kullanıcı ID ve şifre bilgisi için farklı işlemler gerçekleştirilebilir.
- Compiler programı derleme sırasında executable dosya içerisine trap door oluşturabilir.
- Bu durumda, programın kaynak kodunun incelenmesi ile trap door bulunamaz.
- Compiler'ın source kodu incelenirse bulunabilir!

26

## Program tehditleri

### *Logic bomb*

- Bir program **sadece bazı şartlar ortaya çıktığında güvenlik ihlali (logic bomb) gerçekleştirebilir.**
- Bu sorunun belirlenmesi program normal şartlar altında çalışırken oldukça zordur.
- Örneğin, bazı parametreler belirlenmiş değerlere eşit olduğunda ağa uzaktan erişim açılabilir.

27

## Program tehditleri

### *Virüsler*

- Bir virüs, **normal bir programın içine gizlenmiş kod parçasıdır.**
- Virüsler **kendilerini kopyalayabilirler** ve **diğer bilgisayarlara bulaşabilirler.**
- Bir sistemdeki **dosyaları değiştirme, bozma, kullanılamaz hale getirme, programda hataya yol açma** gibi işlemleri yapabilir.
- Virüsler **genellikle e-posta yoluyla bulaşmaktadır.**
- Diğer bir bulaşma yolu ofis yazılımlarına ait dosyalardır.
- Bu **dokümanlar makrolara sahiptir** ve **otomatik olarak execute edilirler.**
- Aşağıda, bir Visual Basic makrosu ile hard disk formatlanmaktadır.

```
Sub AutoOpen()  
Dim oFS  
Set oFS = CreateObject('Scripting.FileSystemObject')  
vs = Shell('c: command.com /k format c:', vbHide)  
End Sub
```

28

## Program tehditleri

### Virüsler

- Virüsler çok farklı kategorilerde olabilir:
  - **File:** Standart bir file virüsü bir dosyaya kendini ekler. Host program hala çalışır durumdadır.
  - **Boot:** Bir boot virüsü sistemin boot sektörüne bulaşır ve sistem her boot edildiğinde çalışır.
  - **Macro:** Çoğu virüs düşük seviyeli dil ile yazılır (Assembly veya C). Macro virüsleri yüksek seviyeli dil ile yazılır (Visual Basic).
  - **Source code:** Kaynak kod virüsü programın kaynak kodunu virüs ekleyerek değiştirir ve kendisini yayar.
  - **Polymorphic:** Polymorphic virüsler kendilerini sürekli değiştirirler ve antivirüs yazılımlarından kurtulurlar.

29

## Program tehditleri

### Virüsler

- Virüsler çok farklı kategorilerde olabilir:
  - **Encrypted:** Şifrelenmiş virüsler kendileriyle birlikte şifre çözme kodunu da bulundurur. Antivirüs yazılımlardan kurtulabilirler.
  - **Tunneling:** Bir virüs antivirüs yazılımlarından kurtulmak için kendisini interrupt-handler içerisine veya cihaz sürücülerinin içerisine gizleyebilir.
  - **Multipartite:** Sistemin birden fazla kısmına (boot sektör, hafıza, dosyalar) aynı anda bulaşabilir.

30

## Konular

- Korumanın amacı
- Korumanın temelleri
- Koruma alanı
- Erişim matrisi
- Erişim haklarının geri alınması
- Güvenlik problemi
- Program tehditleri
- **Sistem ve ağ tehditleri**
- Kriptolojinin güvenlik aracı olarak kullanımı
- Kullanıcı kimlik doğrulama
- Sistemler ve ağlar için firewall kullanımı

31

## Sistem ve ağ tehditleri

- **Program tehditleri, sistemdeki koruma mekanizmasındaki bir hatayı kullanırlar.**
- **Sistem ve ağ tehditleri, servis ve ağ bağlantılarının kötü kullanımını da içerir.**
- Sistem ve ağ tehditleri, **işletim sistemi kaynaklarını ve kullanıcı dosyalarını kötüye kullanabilecek şartları oluşturur.**
- İşletim sistemleri, **başlangıçta maksimum güvenlik şartları ile kurulurlar (secure by default)**, daha sonra administrator tarafından gerekli servisler (FTP, telnet, ...) aktif yapılabilir.

32

## Sistem ve ağ tehditleri

### ***Worms***

- **Worm** bir **process**'tir ve kendisini çoğaltabilir.
- **Kullandığı sistem kaynaklarını sürekli artırır** ve bir süre sonra sistem diğer process'lere servis veremez hale gelir.

### ***Port scanning***

- Port scanning bir saldırı değildir, ancak **saldırgan için sistemin zayıf noktalarının bulunmasında yardımcı olur.**
- Saldırgan açık portları belirledikten sonra, port ile ilişkili servisi kullanarak saldırı yapabilir.

33

## Sistem ve ağ tehditleri

### ***Denial of Service***

- **Denial of service (DOS)** saldırısı hedef **sistemin normal servislerini sağlayamaz hale gelmesini amaçlar.**
- Sistemin normal işlemlerini kullandığı için DOS saldırısını önlemek çok zordur.
- **Distributed denial-of-service (DDOS)** saldırısı **farklı noktalardan aynı anda yapılan DOS saldırılarıdır.**
- Farklı bir DOS saldırısında ise, sistemin **birkaç yanlış authentication denemesinden sonra kullanıcıyı bloklamasını kullanır.**
- **Saldırgan tüm kullanıcı hesaplarının bloklanmasına neden olur.**

34



## Konular

- Korumanın amacı
- Korumanın temelleri
- Koruma alanı
- Erişim matrisi
- Erişim haklarının geri alınması
- Güvenlik problemi
- Program tehditleri
- Sistem ve ağ tehditleri
- Kriptolojinin güvenlik aracı olarak kullanımı
- Kullanıcı kimlik doğrulama
- Sistemler ve ağlar için firewall kullanımı

35

## Kriptolojinin güvenlik aracı olarak kullanımı

### Şifreleme

- Kriptoloji, bir **mesajın belirli bir bilgisayar ve process tarafından oluşturulduğunu doğrular.**
- **Şifreleme (encryption), haberleşmede güvenliği sağlamak amacıyla yaygın kullanılmaktadır.**
- Şifrelemede mesajı alan kişi belirlenmiş anahtarı kullanarak orijinal mesajı elde edebilir.
- Gönderici ve alıcı arasında şifrelenmiş mesaj iletilir ve **saldırganlar orijinal mesajı elde edemezler.**
- Şifreleme eski çağlardan beri haberleşmede kullanılan yöntemdir.

36

## Kriptolojinin güvenlik aracı olarak kullanımı

### Şifreleme

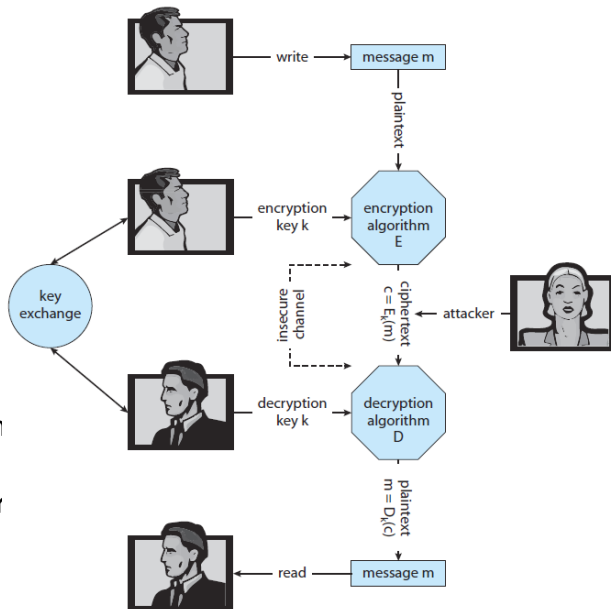
- Bir şifreleme algoritması aşağıdaki bileşenlerden oluşur:
  - Anahtar kümesi (K)
  - Mesaj kümesi (M)
  - Şifreli metin (ciphertext) kümesi (C)
  - Şifreleme fonksiyonu ( $E : K \rightarrow (M \rightarrow C)$ )
  - Şifre çözme fonksiyonu ( $D : K \rightarrow (C \rightarrow M)$ )
- Şifreleme algoritmasında, **c** şifreli metninden **m** mesajını sadece anahtara (**k**) sahip olan elde edebilmelidir.
- **Simetrik** ve **asimetrik** olarak iki tür şifreleme algoritması vardır.

37

## Kriptolojinin güvenlik aracı olarak kullanımı

### Simetrik şifreleme

- Simetrik şifrelemede, **metni şifrelemek için ve şifreli metni çözmek için aynı anahtar** kullanılır.
- Bu yüzden anahtarın korunması zorunludur.
- **DES** (data-encryption standard), **3DES**, **AES** (advanced encryption standard) simetrik **blok şifreleme algoritmalarıdır**
- **RC4 stream şifreleme algoritmasıdır**.



## Kriptolojinin güvenlik aracı olarak kullanımı

### Asimetrik şifreleme

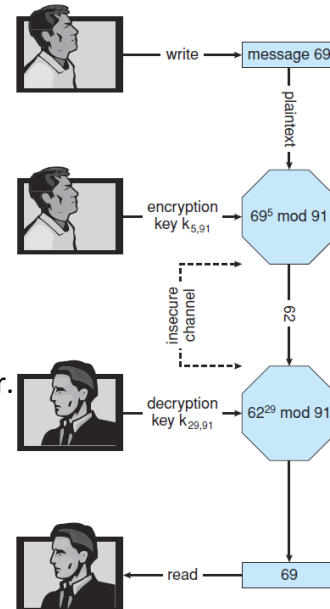
- Asimetrik şifrelemede, **metni şifrelemek için ve şifreli metni çözmek için ayrı anahtarlar** kullanılır.
- Anahtarlardan bir tanesi herkese açıktır (**public key**), diğeri ise sadece sahibi tarafından bilinir (**private key**).
- **RSA (Rivest, Shamir, and Adleman)** en yaygın kullanılan asimetrik şifreleme algoritmasıdır.

39

## Kriptolojinin güvenlik aracı olarak kullanımı

### Asimetrik şifreleme

- RSA için rastgele iki asal sayı seçilir ( $p = 7, q = 13$ ).
- $N = p * q = 7 * 13 = 91$
- $(p-1)(q-1) = 6 * 12 = 72$
- $(p-1)(q-1) = 72$  den küçük ve **aralarında asal sayı** olan  $k_e$  private key seçilir ( $k_e = 5$ ).
- $k_e * k_d \bmod 72 = 1$  olacak  $k_d$  anahtarı hesaplanır.
- $k_d = 29$  hesaplanır.
- Public key  $k_{e, N} = (5, 91)$
- Private key  $k_{d, N} = (29, 91)$



## Kriptolojinin güvenlik aracı olarak kullanımı

### *Authentication*

- Bir mesajı gönderebileceklerin kümesinin belirlenmesine **authentication** denir.
- Bir authentication algoritması aşağıdaki bileşenlere sahiptir:
  - Anahtar kümesi (K)
  - Mesaj kümesi (M)
  - Authenticator kümesi (A)
  - $S : K \rightarrow (M \rightarrow A)$  fonksiyonu
  - $V : K \rightarrow (M \times A \rightarrow \{true, false\})$  fonksiyonu

41

## Kriptolojinin güvenlik aracı olarak kullanımı

### *Key distribution*

- Kriptocu (şifre geliştirici) ile kriptanaliz uzmanı (şifre çözücü) arasındaki en büyük mücadele **anahtarlar** üzerinedir.
- Simetrik algoritmelerde, haberleşen tarafların anahtara sahip olması zorunludur. Ancak, **diğer kişilerin anahtarı ele geçirememesi gerekir.**
- Bir kullanıcı N tane kullanıcı ile haberleşecekse, **N tane gizli anahtarı bildirmesi ve sık sık değiştirmesi gerekir.**
- Asimetrik algoritmelerde, public ve private anahtarlarında diğer kişilerin eline geçmemesi gerekir.

42

## Konular

- Korumanın amacı
- Korumanın temelleri
- Koruma alanı
- Erişim matrisi
- Erişim haklarının geri alınması
- Güvenlik problemi
- Program tehditleri
- Sistem ve ağ tehditleri
- Kriptolojinin güvenlik aracı olarak kullanımı
- **Kullanıcı kimlik doğrulama**
- Sistemler ve ağlar için firewall kullanımı

43

## Kullanıcı kimlik doğrulama

### *Paswords*

- İşletim sistemleri için en büyük problem kullanıcı kimlik doğrulamadır (**user authentication**).
- Kullanıcı kimlik doğrulaması için **password** (şifre) kullanımı en yaygın yöntemdir.
- Her kullanıcı için tekil kullanıcı adı ve şifre ile kimlik doğrulaması yapılır.
- **Şifre** kullanıcı yerine sistemdeki **her nesne veya kaynak için de kullanılabilir**.
- Her şifre için ayrı erişim yetkilendirmesi de yapılabilir.

44

## Kullanıcı kimlik doğrulama

### Paswords

- **Şifre kullanımı kolay yöntemdir**, ancak şifreler tahmin edilebilir, ele geçirilebilir.
- Şifre tahmin etmek için saldırgan kullanıcı hakkında bilgi sahibi olmalıdır.
- Diğer bir yöntemde ise deneme yanılma (**brute force**) ile belirlenebilir.
- Saldırgan kullanıcı sisteme girerken **gözetleyerek şifresini ele geçirebilir (shoulder surfing)**.
- Saldırgan **ağ üzerinden sızarak kullanıcı adı ve şifresini ele geçirebilir (sniff)**. Trojan horse programları ekranı capture ederek ele geçirebilir.
- **Encryption bu problemlerin çözümünde kullanılabilir.**
- Karmaşık ve uzun şifreler daha güvenlidir, ancak hatırlanması zordur.
- Biyometrik veriler de authentication için kullanılabilir.

45

## Konular

- Korumanın amacı
- Korumanın temelleri
- Koruma alanı
- Erişim matrisi
- Erişim haklarının geri alınması
- Güvenlik problemi
- Program tehditleri
- Sistem ve ağ tehditleri
- Kriptolojinin güvenlik aracı olarak kullanımı
- Kullanıcı kimlik doğrulama
- **Sistemler ve ağlar için firewall kullanımı**

46

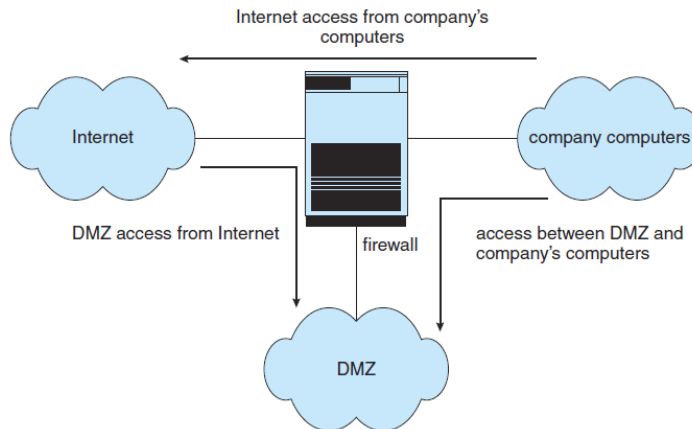
## Sistemler ve ağlar için firewall kullanımı

- Güvenli bir bilgisayar ile güvenli olmayan ağ üzerinden işlem yapılabilir.
- Bir **firewall**, güvenli ve güvenli olmayan sistemleri arasına yer alır.
- Firewall, bir bilgisayar, bir elektronik devre veya router olabilir.
- Bir firewall, ağ üzerinden güvenli alana erişimi izler, denetler ve log kaydı tutar.
- Firewall, erişimi kaynak veya hedef adrese/port numarasına göre sınırlayabilir.
- Firewall, ağı farklı sayıda domain'e böler ve erişimi denetler.
- Genellikle, güvenli olmayan domain olarak İnternet, yarı güvenli ağ (**demilitarized zone - DMZ**) ve firma bilgisayarları olarak domain'ler oluşturulur.

47

## Sistemler ve ağlar için firewall kullanımı

- Şekilde, DMZ bilgisayarlara İnternet ve şirket bilgisayarlarından erişim yapılabilir, şirket bilgisayarlarından İnternet erişimi yapılabilir.
- Firma bilgisayarlarına İnternet ve DMZ bilgisayarlardan erişim yapılamaz.



48