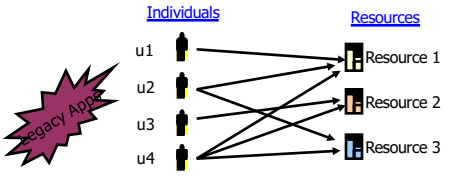
CS458/CS558 Introduction to Computer Security

Access Control

2

Discretionary Access Control

- Restricts access to objects based solely on the identity of users who are trying to access them.
- The owner of a resource decides who can access to the resource and what privileges they have.



http://csrc.nist.gov/rbac/alvarez.ppt

3

Discretionary Access Control Example

- Linux uses discretionary access control
- ls –l

```
-rw-r--r--  1 pyang pyang   17446 Mar  6 11:28 a
drwx------ 21 pyang pyang   4096 Jun 10  2010 b
lrw-r--r--  21 pyang pyang    13 Mar 15  2011 c -> /var/run
```

4

Discretionary Access Control Example

- Linux uses discretionary access control
- ls –l

```
-rw-r--r--  1 pyang pyang   17446 Mar  6 11:28 a
drwx------ 21 pyang pyang   4096 Jun 10  2010 b
lrw-r--r--  21 pyang pyang    13 Mar 15  2011 c -> /var/run
```

- Bit 1:
  - -: normal file, d: directory, l:link pointing to a file
- Bits 2 – 4: r (read), w (write), and x (execution) permissions for owner.
- Bits 5 – 7: r, w, and x permissions for group.
- Bits 8 – 10: r, w, and x permissions for others.

5

Discretionary Access Control Example (Cont.)

```
-rw-r--r--  1 pyang pyang   31399 Mar  6 11:32 t.c
```

```
> chmod 700 t.c
```

6

Discretionary Access Control Example (Cont.)

```
-rw-r--r--  1 pyang pyang   31399 Mar  6 11:32 t.c

> chmod 700 t.c
-rwx------  1 pyang pyang   31399 Mar  6 11:32 t.c
```

7

Discretionary Access Control Example (Cont.)

```
-rw-r--r--  1 pyang pyang   31399 Mar  6 11:32 t.c

> chmod 700 t.c
-rwx------  1 pyang pyang   31399 Mar  6 11:32 t.c

> chmod g+r t.c
```

8

Discretionary Access Control Example (Cont.)

```
-rw-r--r--  1 pyang pyang   31399 Mar  6 11:32 t.c

> chmod 700 t.c
-rwx------  1 pyang pyang   31399 Mar  6 11:32 t.c

> chmod g+r t.c
-rwxr-----  1 pyang pyang   31399 Mar  6 11:32 t.c
```

9

Mandatory Access Control

- Assign a security level to each object
  - Reflects sensitivity of information in that object
- Assign a security clearance to each user
  - Reflects his trustworthiness
- Security level/clearance:
  - Top Secret (TS)
  - Secret (S)
  - Confidential (C)
  - Unclassified (U)

TS > S > C > U

10

Mandatory Access Control

- No Read-up: A user A can read only those objects whose security level <= the security level of A
- No Write-down: A user A can create only objects whose security level >= the security level of A.
- Example:

  Security clearance: (u1, TS), (u2, C),

  Security level: (o1, TS), (o2, S), (o3,C), (o4,U)

11

Mandatory Access Control

- No Read-up: A user A can read only those objects whose security level <= the security level of A
- No Write-down: A user A can create only objects whose security level >= the security level of A.
- Example:

  Security clearance: (u1, TS), (u2, C),

  Security level: (o1, TS), (o2, S), (o3,C), (o4,U)

  u1 can read o1, o2, o3, o4

12

Mandatory Access Control

- No Read-up: A user A can read only those objects whose security level <= the security level of A
- No Write-down: A user A can create only objects whose security level >= the security level of A.
- Example:

  Security clearance: (u1, TS), (u2, C),

  Security level: (o1, TS), (o2, S), (o3,C), (o4,U)

  u1 can read o1, o2, o3, o4

  write o1

13

Mandatory Access Control

- No Read-up: A user A can read only those objects whose security level <= the security level of A
- No Write-down: A user A can create only objects whose security level >= the security level of A.
- Example:

  Security clearance: (u1, TS), (u2, C),

  Security level: (o1, TS), (o2, S), (o3,C), (o4,U)

  u1 can read o1, o2, o3, o4

  write o1

  u2 can read o3, o4

  write o1, o2, o3

14

Role-Based Access Control (RBAC)

- Users are assigned to appropriate roles; Permissions are associated with roles.

  - Role: typically associated with a function or position in an organization, e.g., doctor, nurse, patient.
  - Permission = (operation, resource).
  - A user has a permission if he is a member of some role with that permission.

http://csrc.nist.gov/rbac/sandhu-ferraiolo-kuhn-00.ps

15

Role-Based Access Control (RBAC)

- Benefits
  - Greatly reduces redundancy: permissions are associated with roles, not users. E.g.

Discretionary (10000 rules):

$(u_1, f_1), \ldots, (u_1,\ldots,f_{10})$

……

$(u_{1000},f1), \ldots, (u_{1000}, f_{10})$

Role-based (1000+10 rules):

$(u_1$, Student), …, $(u_{1000}$, Student)

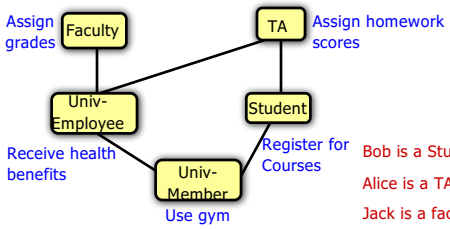
(Student, f_1), …, (Student, f_{10})

  - Easy to administer: roles reflect organizational structure and change less frequently.

16

RBAC: Role Hierarchy

- $r_1 \geq r_2$: r_1 inherits from r_2, r_1 is senior to r_2.
  - Permission flows up, Membership flows down.
    - Every member of r_1 is a member of r_2.
    - Members of r_1 have all the permissions that members of r_2 have.



17

RBAC: Separation of Duty

- Prevent a single user from garnering too much authority.
- Two types
  - Static Separation of Duty (SSD)
  - Dynamic Separation of Duty (DSD)

18

Static Separation of Duty

- Constraints on assignment of user to role.
- If user is member of one role, it gets no membership for other role.

Example:

(DeptChair, Dean) ∈ SSD

A user cannot be in both Deptchair and Dean roles

19

Dynamic separation of Duty

- User cannot act simultaneously in both roles,

Example:

(Supervisor, Cashier) ∈ DSD

A user cannot invoke both Supervisor and Cashier roles at the same time

Cashier

Closes Cashier Role session

Opens Supv Role session

Supervisor

Accounting Error

Correct Error

20

Database Access Control

http://www.fsl.cs.sunysb.edu/~swam/cse590/papers/db-sec-concepts.pdf

21

System R Authorization Model

- Most commercial database management systems (DBMS) adopt DAC
- System R model: one of the first authorization models for relational database management system.
- Database table (relation)

FirstName	LastName	Email	Phone
John	Smith	John.Smith@yahoo.com	626 222-2222
Steven	Goldfish	goldfish@fishhere.net	323 455-4545
Paula	Brown	pb@herowndomain.org	416 323-3232
James	Smith	jim@supergig.co.uk	416 323-8888

22

System R Authorization Model

- Current discretionary authorization models for relational DBMS are based on the System R authorization model.
- Access privilege:
  - SELECT: retrieve data from a database
  - INSERT: insert new records in a database
  - DELETE: delete records from a database
  - UPDATE: update records in a database

23

System R Authorization Model

- Based on ownership administration with administration delegation
- When a user creates a table, becomes the owner of that table.
  - Authorized to exercise all access modes on the table
- Owner can delegate privileges to other subjects using GRANT option.

  Bob: GRANT select, insert ON Employee TO Ann
- If a privilege is granted with the grant option, the user receiving it cannot only exercise the privilege, but can also grant it to other users

24

System R Authorization Model

- Subject, to whom the administration right on a given table has been granted and then revoked, may have granted to another subject an authorization to access the table.

  Bob: GRANT select, insert ON Employee TO Ann

  Ann: GRANT select ON Employee TO Jim

25

System R Model: Revocation

- Revoke: take back to the authorizations from a subject
- Recursive revocation: removes all authorizations for the table from the revokee
- Non-cascading revocation
  - Authorization granted by user from whom the authorization is being revoked are not revoked
- Periodic Authorization
  - Users given access authorization to data only for the time periods in which they need data

26

Content-Based Access Control

- Access control decisions based on data contents
- Example
  - Table employee: records information about employees of a company
  - Access control Requirement: A manager can only access the information of employees who work in the project that he manages.
  - When a manager issues a query, the system returns only the tuples related to the employees who are working in the project managed by this manager.

27

View based Access Control

- Suppose we want to authorize user Ann to access only the employees whose salary is lower than 20000:

```
CREATE VIEW AnnView AS
SELECT * FROM Employee
WHERE Salary < 20000;
GRANT Select ON AnnView TO Ann;
```

Ann: `SELECT * FROM AnnView`

28

Query Modification

- Example: suppose we want authorize user Ann to access only the employees whose salary is lower than 20000:

Ann: `SELECT * FROM Employee`

After query modification:

```
SELECT * FROM Employee
        WHERE Salary < 20000
```

29

Chapter 18
Intrusion Detection

Intrusion

- Unauthorized intrusion into a computer system/network is one of the most serious threats to computer security.
- Intrusion detection systems (IDS): detect unusual patterns of activity or patterns of activity that are known to correlate with intrusions.
  - Host based: monitors systems calls or logs
  - Network based: monitors the flow of network packets
  - Provide early warning of an intrusion so that defensive action can be taken to prevent or minimize damage.

Intruders

- A significant issue for networked systems is hostile or unwanted access by users or software
  - User trespass:
    - Unauthorized logon to a machine
    - Acquisition of privileges or performance of actions beyond those that have been authorized
  - Software trespass: Virus, worm, or Trojan horse
  - These attacks relate to network security.
  - However, these attacks are not confined to network-based attack.
    - E.g. virus may be introduced into a system by means of a diskette.

Three Classes of Intruders

- Masquerader (outsider): an individual who is not authorized to use the computer and who penetrates a system's access controls to exploit a legitimate user's account.
- Misfeasor (insider): a legitimate user who accesses data, programs, or resources for which such access is not authorized, or is authorized but misuses his/her privileges
- Clandestine user (outsider/insider): an individual who seizes supervisory control of the system and uses this control to evade auditing and access controls or to suppress audit collection.

Intruders

- May seem benign, but still cost resources – may slow performance for legitimate users
- However, there is no way in advance to know whether an intruder will be benign or malign.
- May use compromised system to launch other attacks
- Example: attack occurred at Texas A&M, 1992
  - The computing center was notified that one of its machines was being used to attack computers at other location
  - Several outside intruders involved – obtained hundreds of passwords, including some of the major and supposedly security servers.
  - One machine was set up as a hacker bulletin board.

Two Levels of Hackers

- High level: sophisticated users with thorough knowledge of the technology
- Low level: foot soldiers who merely used the supplied cracking programs with little understanding of how they worked.
- Team work combined these two: sophisticated knowledge of how to intrude and a willingness to spend countless hours "turning doorknobs" to probe for weaknesses.

Intrusion Techniques

- Aim to gain access and/or increase privileges on a system - key goal often is to acquire passwords
- Typically, a system must maintain a file that associates a password with each authorized user. If such a file is stored with no protection, then it is easy to access it and learn password.

Intrusion Techniques

- Aim to gain access and/or increase privileges on a system - key goal often is to acquire passwords
- Typically, a system must maintain a file that associates a password with each authorized user.
- Protecting password
  - One-way function: the system stores only the value of a function based on the user's password.
    - When the user presents a password, the system transforms that password and compares it with the stored value.
  - Access control: access to the password file is limited to one or a very few accounts.

Password Guessing

- Attacker knows a login (from email/web page etc)
- Then attempts to guess password for it
- Weak passwords

Password Guessing

- Attacker knows a login (from email/web page etc)
- Then attempts to guess password for it
- Weak passwords
  - Short password (eg. sdgw@)
  - Contains only small-case letters (e.g. absdrsgs)
  - Contains only capital letters (e.g. SIRLSOFN)
  - Contains only digits (e.g. 12389506)
  - Contains dictionary words (e.g. helloworld)
  - Contains only adjacent keys (e.g. sdfghjkl)

Password Guessing

- Attacker knows a login (from email/web page etc)
- Then attempts to guess password for it

Password Guessing

- Attacker knows a login (from email/web page etc)
- Then attempts to guess password for it
  - Exhaustively try all short passwords (1-3 characters)
    - The system can simply reject any login after 3 password attempts
  - The default passwords used with standard accounts
  - Common word searches: try words in the system's online dictionary or a list of likely passwords
  - Collect information about users: variations on names, birthday, phone, common words/interests

Password Guessing

- Guessing attacks are highly effective when a large number of guesses can be attempted automatically and each guess verified without the guessing process being detected
- Use a Trojan horse to bypass restrictions on access
  - A low-privilege user produces a game program and invite the system administrator to use the program.
  - The program indeed plays the game, but in the background it also copies the password file (unencrypted, but access protected).
  - Because the game was running under the administrator's high-privilege mode, it was able to gain access to the password file.
  - Trojan horse can be very difficult to counter

Password Capture

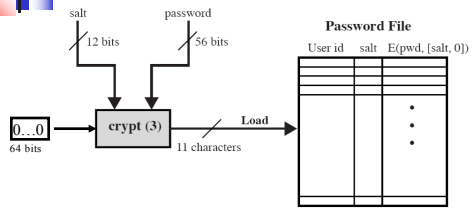
- Watching over shoulder as password is entered
- Using a trojan horse program to collect password
- Monitoring an insecure network login
  - eg. telnet, FTP, web, email

Password Management

Unix Password Management

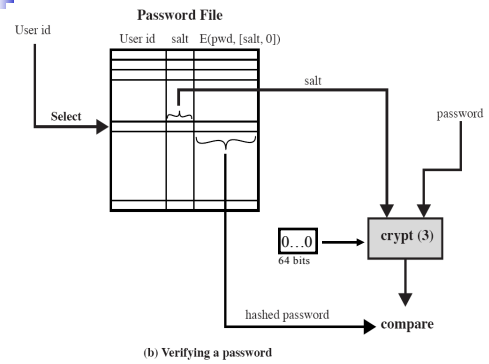
- The user selects a password
- The password is converted into a 56-bit value that serves as the key.
- The encryption routine is known as crypt(3), which is based on DES
- http://en.wikipedia.org/wiki/Shadow_password

Loading a New Password



- 12-bit salt: the time at which the password is assigned to the user
- Encrypt a 64-bit block of zeros 25 times
- The resulting in 64-bit output is translated into 11 character sequence

Verifying a password



(b) Verifying a password

Managing Passwords - Education

- Studies have shown that many people use short passwords or guessable passwords
- Need some approach to counter this
- Educate on importance of good passwords
  - Give guidelines for good passwords
    - Minimum length (>6)
    - Require a mix of upper & lower case letters, numbers, punctuation
    - Not dictionary words

Managing Passwords - Computer Generated

- Let computer create passwords

Managing Passwords - Computer Generated

- Let computer create passwords
  - If the passwords are quite random in nature, users will not be able to remember them.
  - Even if the password is pronounceable, the user may have difficulty remembering it.
  - Have history of poor user acceptance

Managing Passwords - Computer Generated

Forgot Password

- Forgot password
  - If the user has provided an alternative email address, then
    - email the password to the user (not secure)
    - email a link to the user which enables the user to change the password
      - The link should not contain ID of the user
      - The link should expire shortly, e.g. within 2 days

Forgot Password

- Forgot password
  - If the user has provided a phone number, then
    - send the password to the phone (not secure)
    - Send a token to the phone; the user needs to enter the token in order to change the password
      - The token should be randomly generated
      - The token should expire after short time e.g. within 2 days

Forgot Password

- Forgot password
  - If the user has provided security questions and answers, then prompt the user security questions
  - The answer to a good security question is:
    - Safe: cannot be guessed or researched
    - Stable: does not change over time
    - Memorable: can remember
    - Simple: is precise, easy, consistent
    - Many answers: has many possible answers

Forgot Password

- Forgot password
  - If the user has provided security questions and answers, then prompt the user security questions
    - It is difficult to find questions that meet all five criteria
      - People share so much personal information on social media, blogs, and websites, that it is hard to find questions that meet the criteria above.
      - Many questions are not applicable to some people; for example, what is your oldest child's nickname – but you don't have a child.

Forgot Password

- Forgot password
  - If the user provides neither email, phone, nor security questions
    - Ask user to provide e.g.
      - Name, birthday etc
      - When was the account created
      - The last date when the user logs onto the account
      - Three contacts
      - Three email subjects
      - IP address

Intrusion Detection

Intrusion Detection

- Intrusion detection system
  - Block if detected quickly. The sooner the intrusion is detected, the less the amount of damage and the more quickly that recovery can be achieved.
  - Act as deterrent
  - Collect info to strengthen the intrusion prevention facility

Intrusion Detection

- Assume intruder will behave differently from a legitimate user
  - But there is an overlap in these behaviors.
  - A loose interpretation of intruder behavior will lead to a number of false positives
  - A tight interpretation of intruder behavior will lead to an increase in false negatives or intruders not identified as intruders.
  - An element of compromise in the practice of intrusion detection

Statistical Anomaly Detection

- Collect of data relating to the behavior of legitimate users over a period of time.
- Statistical tests are applied to observed behavior to determine with a high level of confidence whether the behavior is not legitimate user behavior.

Statistical Anomaly Detection

- Threshold detection
  - Defines thresholds for the frequency of occurrence of various events - independent of users
    - Count occurrences of specific event over time
    - If exceed reasonable value assume intrusion
    - Alone is an ineffective detector: may generate a lot of false positive or a lot of false negative

Statistical Anomaly Detection

- Profile based
  - A profile of the activity of each user is developed and is used to detect changes in the behavior of individual accounts.
    - Characterize past behavior of users
    - Detect significant deviations from this

Rule-Based Detection

- Rule-based detection: attempt to define a set of rules that can be used to decide that a given behavior is that of an intruder.
  - Rules identify known penetration, weakness patterns or suspicious behavior
  - Compare audit records or states against rules
  - Quality depends on how well this is done

Example: Rule-Based Detection

- Uses heuristic rules to assign degrees of suspicion to activities
  - Users should not read files in other users' personal directories
  - Users must not write other users' files
- Audit records are examined, and they are matched against the rule base.
- If a match is found, then the user's suspicion rating is increased.
- If enough rules are matched, then the rating will pass a threshold that results in the reporting of an anomaly.

Audit Records

- Fundamental tool for intrusion detection
- Native audit records
  - Part of all common multi-user OS
  - Advantage: no additional collection software is needed
  - Disadvantage: may not contain the needed information or may not contain it in a convenient form.
- Detection-specific audit records
  - Created specifically to collect wanted info
  - At cost of additional overhead on system

Audit Records

- The audit records developed by Dorothy Denning
  - Subject: initiators of actions – a terminal user or a process acting on behalf of users or groups of users.
  - Object: e.g. files, programs, etc.
  - Action: operation performed by the subject on or with an object, e.g. read, login, execute.
  - Exception-condition: which exception condition is raised on return
  - Resource-usage: e.g. number of records read or written, processor time, etc.
  - Time-stamp: identify when an action took place.

Audit Records

- Most user operations are made up of a number of elementary actions
  - E.g. file copy
    - Access validation
    - Read from one file
    - Write to another file
- Consider the command copy game.exe to <Library> game.exe issued by Smith.

smith	execute	<Library>copy.exe	0	CPU=00002	11058721678
smith	read	<smith>game.exe	0	CPU=00020	11058721679
smith	write	<Library>game.exe	write-vio	CPU=00005	11058721680

Base-Rate Fallacy

- Practically an intrusion detection system needs to detect a substantial percentage of intrusions with few false alarms
  - If too few intrusions detected ➔ false security
  - If too many false alarms ➔ waste time
- This is very hard to do
- Existing systems seem not to have a good record