Lecture 4 - Authentication and Access

CSE497b - Spring 2007
Introduction Computer and Network Security
Professor Jaeger
www.cse.psu.edu/~tjaeger/cse497b-s07/
Why authenticate?

• Why do we want to verify the identity of a user?
Control Access

• An identity permits access to resources
• In computer security this is called
  – *Access control*
  – *Authorization*

• In authorization, we talk about:
  – Subjects (for whom an action is performed)
  – Objects (upon what an action is performed)
  – Operations (the type of action performed)

• Authorization limits a *subject’s* access perform an *operation* on an *object*
  – The combination of object and operations allowed are called a *permission*
“Project” 1

- Login to Playpen VM
  - We will send you your username, password, IP

- Change your password
  - Do *not* change the root password

- Need to do some minor Linux administration

- Customize your VM
  - You have sudo privilege
  - You are the administrator

- Posted on the calendar (due next Th, Feb 1)
  - If it’s good enough for the President...
A Brief History

• Early computing systems had no isolation
  – Shared memory space
  – Shared file space

• Some physical limitations made this OK
  – Batch processing
  – Load the tape/disk for the application
  – Network? What network?

• In the mid-60s people started to work on ‘multiuser’ or ‘time-sharing’ systems
  – What about a bug?
  – What about my data?

• Mostly about protection
Multiprogrammed Systems

- **Multics project**
  - AT&T, MIT, Honeywell, etc.
  - General purpose, multi-user system
  - Comprehensive security
    - Hardware protection
    - Subject labeling
    - Permission management

- **UNIX project**
  - Arose from the ashes of Multics
  - A stripped-down multiuser system
Authentication and Access

- Authenticate user
  - E.g., login and ssh
  - Verify password or ...

- Create processes with appropriate identity (subject)
  - E.g., UNIX user id

- Limit access of these processes using subject
  - E.g., Access control of files based on subject

- Protect one user from another

Q: Is that enough for enforcing security?
Security vs. Protection

• Protection
  – Focus on process isolation and user separation

• Security Requires
  – Confidentiality: Don’t leak your secret files
  – Integrity: Don’t overwrite your important data
  – Availability: Don’t prevent an operation

• System Protection Mechanisms are Not Enough!
  – Do NOT ensure security of user’s data against an attacker
  – Functional demands result in system compromise
  – Does not scale beyond a single system

• Current access control mechanisms fail to enforce security goals
Your Programs

• What permissions are available to programs that you run?
  – Email
  – Web browser
  – Game
  – A little program that you downloaded from the web

• What can these programs do with your permissions?
Your Programs

• They can do anything that you can
  – Use any permission that you have
  – Including the owner permission
    • They can give anyone access to your files

• Worse yet, traditional access control is not comprehensive
  – A program can send a file anywhere

• What does this mean to the secrecy of your data?

• And it gets even worse…
Security Model

• Adversaries
  – Who?

• Threats
  – What can they do?

• Vulnerabilities
  – What vulnerabilities can the adversaries leverage?

• Trust model
  – What are you trusting (implicit in the discussion so far)?
Security Model

• Adversaries
  – Other system users
  – Program developers
  – Web responses, emails
  – Remote parties

• Threats
  – Code running on same system
  – Input malicious code

• Vulnerabilities
  – User can be tricked
    • Lots of applications enable the user to run downloaded code
  – Application vulnerabilities
  – Misconfigured policy
Email Clients

• In addition to reading emails,
  – Execute attachments (run with your privileges)
  – May even run a malicious script w/o opening an attachment (run with your privileges)

• What kind of attachments can you open?
  – From Granny: May be a forged address
  – Word or Excel: May contain viruses

• But, I’ve really gotta see it
  – Plain text
  – Signed emails
  – Anti-virus may catch some, but no guarantee
Access Matrix

• Describe all possible accesses
  – Operations of $(S_2, O_2)$
  – E.g., read, write, execute

• Specify which users’ processes can access which files

• Necessary to specify policy to protect users

<table>
<thead>
<tr>
<th></th>
<th>$O_1$</th>
<th>$O_2$</th>
<th>$O_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_1$</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>$S_2$</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>$S_3$</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>
Manage the Access Matrix

• How do you give someone access to your file?
• Access matrix also has management permissions
  – owner permission
• A subject with owner permission can
  – Give another user permissions to an object
  – Even the owner permission itself
• This seems necessary, right?

<table>
<thead>
<tr>
<th></th>
<th>O₁</th>
<th>O₂</th>
<th>O₃</th>
</tr>
</thead>
<tbody>
<tr>
<td>S₁</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>S₂</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>S₃</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>
The Door Is Open

- Suppose that you want to download new software
  - Or a software update
- Typically, users lack the permissions to overwrite system files
  - Why update a system file?
  - “Penetrate and patch”
- For convenience, users run with administrative privileges (e.g., Windows)
  - Now, the downloaded code (and the email attachment) runs with full privilege
Tip of the Iceberg

- Viruses
- Worms
- Spyware
- Keyloggers
- What’s next?
Remote Access

• Suppose you are building a service for remote clients
  – E.g., a web application
• How are you going to authenticate identity?
• What rights are you going to assign to which identity?

• Q: What are your vulnerabilities now?
  – Consider the network and the remote computer

![Diagram]
Client ➔ Name/Password ➔ Your Server ➔ Services

CSE497b Introduction to Computer and Network Security - Spring 2007 - Professor Jaeger
Remote Access

- Client selects a name and password
  - How does the client protect the password?
- Server stores state on client for ease of use (cookies)
  - How do we ensure that attacker can’t use this state?
- What other forms of authentication are used in e-commerce?
Single Signon

- Nice feature for users:
  - Login once, then use any number of remote services

- A centralized service provides authenticated users with tokens
Single Signon

• As a remote service provider
  – What is the basis for trust for the single signon?
  – Can you trust the token?
• Can we run a business-to-business on such trust?
  – Is there a second-factor for authentication?
Take Away

• We have just looked at the most common mechanisms
  – Passwords
  – User-based Access Control

• There are a slew of problems with each

• But, this is what the world uses
  – What can we do?

That Is the Topic of This Course