Lecture 23: System & Network Security

Adam Hawley

April 10, 2019

Contents

1 System Security & Protection

A computer consists of a collection of objects, hardware or software. Each object has a unique name and can be accessed through a well-defined set of operations. A **security policy** defines what it means to be secure for a particular system. The **protection problem** is to ensure that each object is accessed correctly and only by those processes that are allowed to do so.

2 Principles of Protection

- A **privilege** is the right to execute a particular operation on a given object.

One guiding principle is named the **principle of least privilege**, where:

- Programs, users and systems should be given just enough **privileges** to perform their tasks (this limits damage if the entity has a bug or gets abused).

Privileges can be one of:

- **Static** (During life of system, during life of process)
- **Dynamic** (Changed by process as needed): **Domain switching privilege escalation**

"Need to know" is a similar concept regarding access to data.

It is important to consider the **grain** aspect.

- **Rough-Grained**: Management is easier,simpler but least privilege is now done in large chunks.
  - For example, traditional Unix processes either have abilities of the associated user or of the root.
- **Fine-Grained**: More complex, more overhead but more protective.
  - For examlpe, ACL (Access Control List) or RBAC (Role Based Access Control).

Privilege management is commonly supported by the notion of domains which can be user, process, procedure etc.

3 Domain Structure

- Access-Right = `<object-name, rights-set>`
  - `rights-set` is a subset of all valid operations that can be performed on the object.
- Domain = set of access-rights.

A process, at any point in time, is associated with one domain but can switch domains in a controlled way. Domains can overlap.

See slide 7 for Unix example.

4 Security

A system is said to be secure if resources are used and accessed as intended under all circumstances.

Intruders Attempt to breach security

Threat Potential security violations

Attack Attempt to breach security (can be accidental or malicious but easier to protect against accidental).

4.1 Security Violation Categories

Breach of Confidentiality Unauthorised reading of data.

Breach of Integrity Unauthorised modification of data.

Breach of Availability Unauthorised destruction of data.

Theft of Service Unauthorised use of resources.

Denial of Service (DoS) Prevention of legitimate use.

4.2 Security Violation Methods

Masquerading (breach authentication) Pretending to be an authorised user to escalate privileges.

Replay Attack As is or with **message modification**.

Man-in-the-middle Attack Intruder sits in data flow, masquerading as sender to receiver and vice versa.

Interception Intercept an already-established session to bypass authentication (e.g. sniffing, hijacking, covert channel).

4.3 Security Measure Levels

It is impossible too have absolute security, but make cost to perpetrator sufficiently high to deter most intruders. Security must occur at four levels to be effective:

Physical Control access to data ceters, servers, connected terminals

Human Avoid social engineering, phishing, dumpster diving

Operating System Protection mechanisms

Network Encryption, firewalls, blacklisting

Security is as weak as the weakest link in the chain.

5 Examples

See lecture for examples on **Meltdown** and **DNS Spoofing**.