

CCT395, Week 10

Database Security

Catalin Bidian

University of Toronto

November 10, 2010

Database Security – Main Objectives

1. Confidentiality (aka Secrecy):

- a. Data must be kept private
- b. Information should NOT be disclosed to unauthorized users

2. Integrity: data are accurate – protected from unauthorized modification and/or destruction

3. Availability:

- a. Data are accessible when needed
- b. Authorized users are not denied access
- c. Protecting the network from events that would render data unavailable (including power outages)

To Achieve the Main Objectives

1. Security policies (the 3-C's):

- a. Must be clear
- b. Must be consistent
- c. Must be concise

2. Security mechanisms:

- a. Internal (e.g. Operating System, DBMS, firewalls, etc)
- b. External (e.g. restrict physical access, outside-the-network attacks, social engineering attacks, etc.)

Sources of Threats

1. Internal (aka insiders):

- a. Employee attacks (deliberate or accidental)
- b. Accidents and security oversights

2. External:

- a. Physical attacks
- b. Software attacks



Who are Insiders?

An *insider* is any entity that has authorized access to the organization's network and data/information resources

1. Employees:

- a. Full/Part-time employees and their families
- b. Former employees
- c. Contractors, co-op students

2. Network users:

- a. Partners (recent mergers and acquisitions)
- b. Clients, customers

Who are Insiders? (cont'd)



3. **IT product/service suppliers** – software development, hardware maintenance, remote support

4. **Automated systems and processes** (e.g. CIBC faxes to US Allstar and Wade Peer, Quebec and who knows where else... 😊) - http://www.priv.gc.ca/incidents/2005/050418_02_e.cfm and <http://www.theglobeandmail.com/report-on-business/article959327.ece>



The Insider Advantage

- Knowledge of asset value
- Access to assets
- Knowledge of business operations and procedures
- Knowledge of protective controls **and how to bypass them**
- Knowledge of corporate culture
- Trusted by management and “dog watchers”



What do Insiders Do?

- Information leakage (knowingly or not)
- Inappropriate activity
 - Inappropriate use of corporate resources
 - Access to internal information
- Malicious activity
 - Inappropriate or illegal access to accounts/resources
 - Fraud and/or identity theft
 - Sabotage



Information Leakage

- Job postings
- Newsgroups and blogs
- Social networking sites
- Instant messaging services
- Newspapers
- Legal investigations and court trials



An Insider's Profile

- Sense of entitlement (“I’ve been here 20 years and you wouldn’t dare restrict my access” or “I work 17 hours a day and never got a bonus”)
 - Any challenge on “entitlement” leads to more resistance and frustration → prerequisite for revenge!
- Frequently frustrated in the workplace (may also be personally and/or socially frustrated)
- May possess strong computer skills (or think they do 😊)
- Tend to plan their revenge (watch out for the early signs)
- Financial gain is emerging as a significant motivating factor

The Insider's MO

1. Employee attacks:

- a. Hacking techniques
- b. Take advantage of legitimate access
- c. Break into computer rooms
- d. Social engineering

2. Accidents and security oversights:

- a. Victims of social engineering
- b. Accidents causing physical damage
- c. Misuse of system(s)
- d. Installing personal hardware/software on the network

Lessons Learned from Internal Attacks*

- Negative impact on corporate finances
- Negative impact on corporate reputation
- Internal threats **ARE** a corporate problem
 - Sometimes corporations refuse to acknowledge this... 😊
- Internal threats cannot be solved by technology alone
- Growing lack of reporting and information sharing

External Threats

1. Physical attacks:

- a. Physical access to computer rooms
- b. Leaving Admin accounts logged-in

2. Software attacks:

- a. White-hat hackers
- b. Black-hat hackers
- c. Script kiddies
- d. Cyber-terrorists

Cyberterrorists: Cyberterrorists are hackers who are motivated by a political, religious, or philosophical agenda. They may propagate their beliefs by defacing Web sites that support opposing positions.

(p. 326)

Types of External Attacks

1. (Distributed) Denial of Service (D/DoS):

- Easy to detect ******; difficult to defend against
- Can be in fact both internal and external
- First incidents – 2001 – Register.com, Dept. of Finance
- 2002 and 2007 attempts to bring down the Internet (DDoS attack against the DNS Backbone)
- Facebook, Twitter, Livejournal, Amazon, Google, etc, etc, etc...
- **Prevention Tools:**
 - Firewalls → not very efficient on port 80 (Internet) for DDoS
 - Switches and routers (ACL capability) to limit and shape traffic
 - Intelligent hardware – bandwidth management, deep packet inspection
 - Intrusion Prevention Systems (IPS)

Types of External Attacks (cont'd)

2. Buffer Overflow:

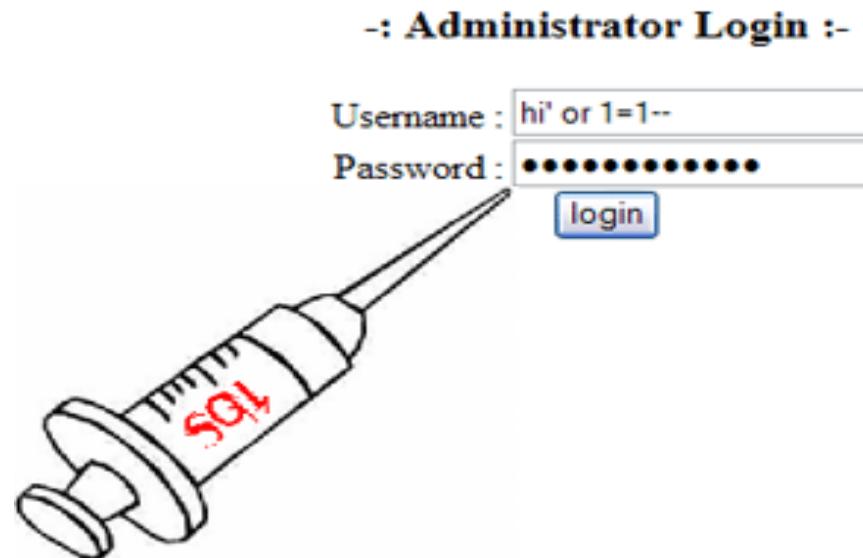
- Almost impossible to detect by network/software engineers
- Hackers can insert their own code into a program and take control of the system **
- **Variations:** stack-based and heap-based
- Attacks are not very common – 2001 “Code Red” worm (MS IIS), 2003 Xbox exploit and “SQL Slammer”
- **Prevention Tools:**
 - NOP slide (no-operation instructions)
 - “Jump to Address” technique
 - Choice of programming language (C/C++ not a very good choice if used without the C++ libraries)

Types of External Attacks (cont'd)

Web customers cannot issue ad hoc queries using a query language; they can only use the browser-based application provided for them. Therefore, there is little that the typical Web user can do to compromise the security of the database.

3. SQL Injections:

- Exploit openings in SQL statements to insert and execute code, altering the database and taking control of the system **
- Variations:
 - Incorrectly filtered escape characters (application layer)





SQL Injections (cont'd)

3. SQL Injection Variations:

- **Incorrectly filtered escape characters (application layer)**

SELECT authorization_level FROM Users WHERE user_name = '\$email';

Normal user input: catalin.bidian@utoronto.ca → SELECT authorization_level FROM Users WHERE user_name = 'catalin.bidian@utoronto.ca' → Auth Level: Admin

SQL Injection attack: test@test.com' OR '1' = '1' → SELECT authorization_level FROM Users WHERE (user_name = 'test@test.com' OR '1' = '1') → Auth Level : ??? (LIST ALL)

SQL Injections (cont'd)

3. SQL Injection Variations:

- **Incorrect type handling (application layer)**

```
SELECT * FROM Users WHERE user_id = "" + $variable + "";
```

Normal user input: catalin → SELECT * FROM Users WHERE user_id = 'catalin'

SQL Injection attack:

- a) test'; DROP TABLE Users → SELECT * FROM Users WHERE user_id = 'test'; DROP TABLE Users
- b) test'; INSERT INTO Users (user_id, password, auth_level) VALUES ('catalin', 'cct395', 'Admin') → SELECT * FROM Users WHERE user_id = 'test'; INSERT INTO....
- c) test'; UPDATE Users SET authorization_level = 'Admin' WHERE user_id = 'catalin'

SQL Injections (cont'd)



3. SQL Injection Variations:

- **Brute force attacks (application layer)**

```
SELECT * FROM Users WHERE user_name = '$email' AND password = '$user_password';
```

SQL Injection attack:

- The attacker tries countless values for *user_password* until he/she succeeds
- Assumes knowing at least one user name
- Time consuming
- Not generally feasible

SQL Injections (cont'd)



3. SQL Injection Variations:

- Blind SQL injection
 - Conditional responses
 - Conditional errors
 - Time delays
- Schema field mapping (sequential queries)
 - WHERE *field* = 'x' AND user_email IS NULL;
 - WHERE user_email = 'x' AND user_id IS NULL;
- Routine data base design (e.g. *user_id*)
- Exploiting vulnerabilities in SQL/mySQL server



SQL Injections (cont'd)

Mitigation Tools:

- Cleanup the user input
 - Limit input boxes to a certain number of characters
 - Validate input programmatically (e.g. phone numbers, SIN, etc) - some numbers have check digit logic embedded
 - Quote-safe the input (e.g. John O'Connell)
- Use bound parameters
 - myQuery = "SELECT *...WHERE user_id = \$email;"
 - Sth→execute(\$email);
- Use xp_cmdshell, xp_startmail, xp_sendmail, sp_makewebtask
- Limit permissions on the database
- Use stored procedures
- Hide URL address in web-browser
- Configure error reporting, monitor logs, trigger alerts, etc

Some Examples

Monkeys with mood

```
"<?php
    echo $_GET [ "mood" ] ;
?>" :
```

http://.../.../monkeys_3.php?mood=H



http://.../.../monkeys_3.php?mood=A



Some Examples (cont'd)

Using *mysql_real_escape_string*

Not very good:

```
$owner = $_GET['owner'];  
$query = "select name, species from pet where owner=" . $owner . "";
```

Better:

```
$owner = mysql_real_escape_string($_GET['owner']);  
$query = "select name, species from pet where owner=" . $owner . "";
```



Some Examples (cont'd)

Using *mysql_real_escape_string*

However:

```
$result = "SELECT salary FROM Employees WHERE id = "  
.mysql_real_escape_string($_POST['id']);
```

if `$_POST['id']` is injected with **45005 OR 1=1** then the resulting query becomes:

```
SELECT salary FROM Employees WHERE id = 45005 OR 1=1
```




Some Examples (cont'd)

Using *mysql_real_escape_string*

Another one:

```
$result=mysql_query('SELECT * FROM users WHERE  
username="'.$_GET['username'].'"');
```

```
$result=mysql_query('SELECT * FROM users WHERE  
username="'.mysql_real_escape_string($_GET['username']).'");
```

This way, if the user tried to inject another statement such as a DELETE, it would harmlessly be interpreted as part of the WHERE clause parameter

```
SELECT * FROM users WHERE username = \';DELETE FROM  
comments WHERE title != \'
```

Types of External Attacks (cont'd)

4. Malware:

- Malicious software (viruses, trojans, spyware, worms, adware, etc) used by attackers to gain control over the system
- John von Neumann's postulate: a machine (aka program) can reproduce itself → nanotechnology
- **Mitigating Tools:**
 - Anti-virus and anti-spyware programs
 - Firewalls
 - Log and file monitoring software
 - Intrusion prevention systems
 - Patch management → difficult to maintain and decide which to install

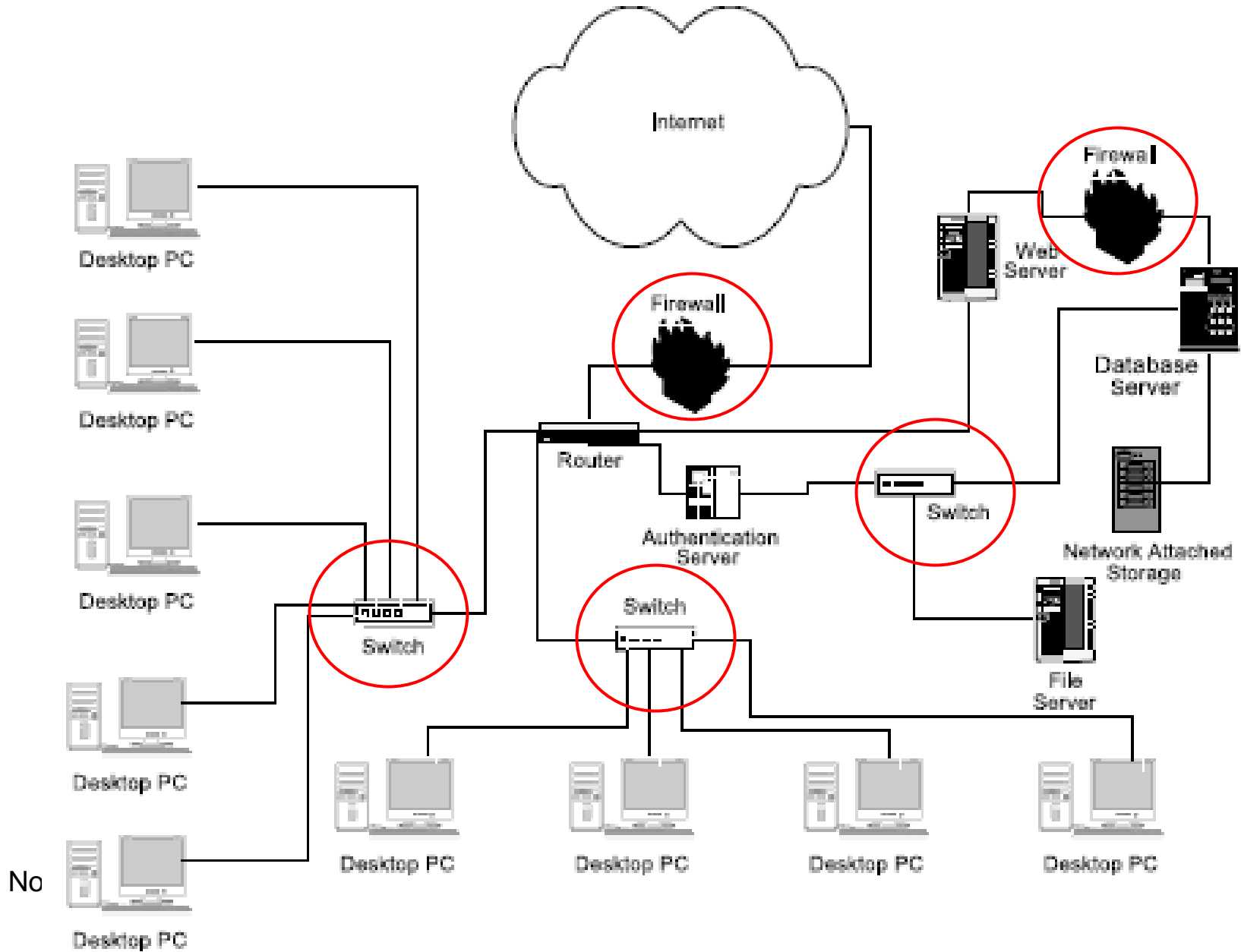
See more at http://www.us-cert.gov/reading_room/malware-threats-mitigation.pdf

Types of External Attacks (cont'd)

5. Brute Force Attacks:


- Stems from cryptography
- Continuously run programs that try to break into the system
 - List of email addresses, passwords, phone numbers, etc – aka *Dictionary Attacks*
- **Mitigating Tools:**
 - Anti-virus and anti-spyware programs
 - Firewalls
 - Log and file monitoring software
 - Intrusion prevention systems

Firewalls



Firewalls (cont'd)

FortGuard Firewall V2.2 Build 90212, Advanced (Registered)

 **FortGuard Firewall**
Professional Anti-DDoS System


(C)2003-2009 FortGuard Software Ltd.
<http://www.fortguard.com>
Email: support@fortguard.com

- Monitors
- Ports to Block
- IP Filters
- TCP Flow Control
- Intrusions**
- Logs

[Anti-ArpSpoof](#)
[Register](#)
[Minimize](#)

Intruder Addr	Time	Information
68.221.224.150:...	10/29/08-17:50:17	http_decode: overlong character.
22.73.95.17:54...	10/29/08-17:47:39	spp_stream4: Evasive retransmiti.
22.59.142.136:...	10/29/08-16:50:24	spp_stream4: Evasive retransmiti.
22.59.142.136:...	10/29/08-16:50:24	spp_stream4: Evasive retransmiti.
21.227.171.207...	10/29/08-16:48:37	http_decode: overlong character.
21.227.171.207...	10/29/08-16:48:37	SQL Injection attempt
11.139.116.198...	10/29/08-16:46:17	http_decode: missing uri
11.139.116.198...	10/29/08-16:46:16	http_decode: missing uri
11.139.116.198...	10/29/08-16:46:16	spp_stream4: NMAP Fingerprint 5
68.211.47.118:3...	10/29/08-16:45:22	http_decode: overlong character.
18.91.109.120:...	10/27/08-20:41:32	http_decode: overlong character.
21.226.40.3:27...	10/27/08-20:38:58	http_decode: overlong character.
21.234.85.214:...	10/27/08-20:34:00	spp_stream4: Evasive retransmiti.
21.234.85.214:...	10/27/08-20:34:00	spp_stream4: Evasive retransmiti.
11.139.116.166...	10/27/08-20:32:24	spp_stream4: NMAP Fingerprint 5
11.139.116.166...	10/27/08-20:32:24	http_decode: missing uri
17.88.142.82:1...	10/27/08-18:14:41	http_decode: overlong character.

[Setting](#) [Refresh](#) [Delete](#) [Empty](#) [Export as Html](#)



Types of External Attacks (cont'd)

6. Social Engineering:

- It is in fact both internal and external type of attack based on psychological manipulation
- Kevin Mitnick – security consultant and convicted criminal
- MO's:
 - Pretexting (including an induced sense of crisis)
 - Diversion
 - Phishing (which includes over-the-phone or IVR)
 - Baiting
 - Quid pro quo
 - Confidence tricks
 - Eavesdropping, shoulder surfing, intimidation,

Social Engineering (cont'd) *

Mitigating Tools:

Require employees to take two consecutive weeks of vacation at least once every two years. If an employee is hacking the organiza-

- No technology can fully mitigate SE attacks
- Employee education and raising awareness
- Develop and enforce policies and procedures **
 - Change management
 - Password policies - **DO STRONG PASSWORDS INCREASE SECURITY?** (Password management survey - http://www.roboform.com/enterprise/whitepapers/RoboForm_Enterprise-Password_Management_Survey.pdf)
 - Information classification and access
- Top-down corporate security culture
- Building a human firewall
- Use it as a technique to perform security audits



General Mitigation Techniques

- Securing the perimeter – security cameras, smart locks, removal of explicit signs
- Restrict physical access
 - One-way traffic
 - Access key-cards
 - Environmental design
- Firewalls
 - Stateful packet inspection
 - Circuit-level gateways (CLGs)
 - Application proxies (aka application-level gateways – ALGs)
 - Personal firewalls

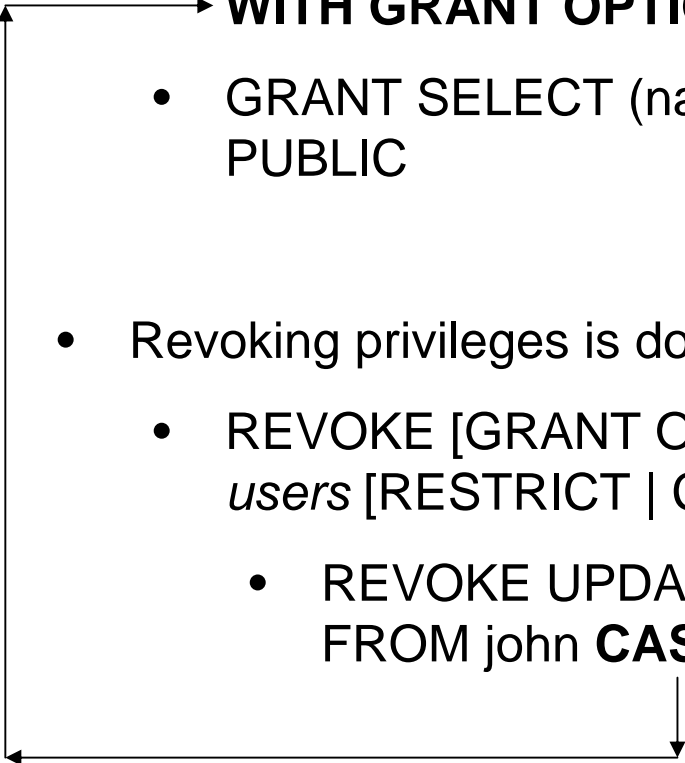
General Mitigation Techniques (cont'd)*

- Virtual Private Networks (VPNs)
- Subverting authentication
- Implement and enforce access controls (e.g. Bell – LaPadula – no read-up / no write-down)
- Limit disk usage
- Enhanced user authentication
 - What you know
 - What you have
 - What you are
- Database authorization matrices (access control)

Database Authorization

- DBMS offers two main approaches to access control:
 - Discretionary access control (DAC)
 - Mandatory access control (MAC)
- SQL supports DAC through GRANT and REVOKE
 - GRANT *privileges* [ON *table*] TO *user*
 - GRANT CONNECT TO john
 - GRANT INSERT, DELETE ON Payroll TO john
 - Additional clause – WITH GRANT OPTION
 - GRANT UPDATE ON Payroll TO john WITH GRANT OPTION

Discretionary Access Control (cont'd)

- Granularity in GRANT – specify the table fields
 - GRANT UPDATE (phone_num, address) ON Employees TO john
→ **WITH GRANT OPTION**
 - GRANT SELECT (name, phone_number) ON Employees TO PUBLIC
 - Revoking privileges is done through REVOKE
 - REVOKE [GRANT OPTION FOR] *privileges* ON *table* FROM *users* [RESTRICT | CASCADE]
 - REVOKE UPDATE (phone_num, address) ON Employees FROM john **CASCADE**
- 

Discretionary Access Control (cont'd)

- DACs have weaknesses
 - Susceptible to Trojan attacks – example:
 - Attacker has no rights to the table containing sensitive information (e.g. Payroll)
 - Attacker creates new table in the database (e.g. MyTable)
 - Attacker provides INSERT privileges to victim ON MyTable
 - Attacker modifies the application (i.e. website) so that when victim executes a SELECT FROM Payroll, the results get automatically inserted into MyTable

DACs must be combined with MACs for good results



Mandatory Access Control

- **Most popular model is Bell – LaPadula**
 - Simple Security Property – no read-up
 - *-Property (aka “star-property”) – no write-down
 - Discretionary Security Property – use of access matrix
- **Multi-level Relations and Polyinstantiation**
 - Security class assigned to each table (or even each row) → the concept of **multi-level table**

User ID	User Name	e-Mail	Security Class
101	John	john@cct395.org	A
102	Mary	mary@cct395.org	J
103	Catalin	catalin@cct395.org	N

- Someone with “J” wants to INSERT a row... (see next slide)

Mandatory Access Control (cont'd)*

- Someone with “J” wants to INSERT a row

User ID	User Name	e-Mail	Security Class
101	John	john@cct395.org	A
101	Yuri	yuri@cct395.org	J
102	Mary	mary@cct395.org	J
103	Catalin	catalin@cct395.org	N

- If the insertion is allowed → two “101” User IDs
- If the insertion is not allowed (i.e. violation of Primary Key) then we can infer that the Security Class is **higher** than “J”
 - “J” becomes “A”
- Solution: include the Security Class in the Primary Key definition

Mandatory Access Control (cont'd)*

- **Covert Channels (DOD Security Levels)**
 - Two sites with different security classes
 - A = most secure class
 - D = least secure class
 - Both sites have to agree before a transaction is committed
 - Attack:
 - Site D agrees to commit (because of its lower class)
 - Site A agrees only if it transmits 1 bit
 - The attacker will send information from A to D repeatedly in 1-bit packets → tedious but it works!
 - Violation of Bell-LaPadula (no-write down)
 - Solution: most DBMSs have already implemented controls

Other Methods

- Who has access to what
 - Some organizations have solved this problem by appointing a committee to handle the decisions about who has access to what. Users
 - So... what's the problem?
- Establish ROLES
 - CREATE ROLE interns;
 - GRANT interns TO john, catalin;
 - GRANT SELECT, UPDATE (phone_number) ON Employees TO interns;
 - REVOKE interns FROM catalin;
 - DROP ROLE interns;
- Use encryption, SSL, digital signatures, etc

Backup & Disaster Recovery

- **Backup is part of a good security strategy**
 - Ensure the backup is “clean”
 - Ensure there are enough copies and versions
 - Consider “how much” you can afford:
 - To spend on backups
 - To loose should a disaster happen
 - Psychological and technical components
- **Disaster recovery**
 - Always have a disaster recovery plan
 - Where the backups are kept

For small organizations, it's not unheard of for an IT staff member to take backups home for safekeeping.

Disaster Recovery (cont'd)

- Always have a disaster recovery plan
 - Purchase new hardware, O/S, software?
 - How will the data be restored
 - Determine who/what is affected/impacted and to what degree
 - Establish priorities for recovery (immediate, 1-day, 1-week, etc)
 - Test and refine the plan (simulate a disaster)
- Conduct Business Impact Assessments (BIAs)
- Conduct periodical vulnerability assessments
- Implement disaster avoidance and prevention procedures
 - Detective measures
 - Preventive measures
 - Corrective measures

How Much is Too Much?

DEPENDS...

Further Readings

- Illicit Cyber Activity in the Banking and Finance Sector (Technical Report), by the US Secret Service and the CERT Coordination Center of the Carnegie Mellon University - <http://www.sei.cmu.edu/library/abstracts/reports/04tr021.cfm>
- Computer Systems Sabotage in Critical Infrastructure Sectors, by the US Secret Service and the CERT Coordination Center of the Carnegie Mellon University – <http://www.cert.org/archive/pdf/insidercross051105.pdf>
- Association of Certified Fraud Examiners (ACFE) report on Occupational Fraud and Abuse - <http://www.acfe.com/documents/2006-RttN.pdf>
- Annual Computer Security Institute (CSI) & FBI's Computer Crime and Security Survey - <http://gocsi.com/survey>
- Deloitte's annual Global Security Survey - http://www.deloitte.com/view/en_GX/global/industries/technology-media-telecommunications/c4d38a120c9a8210VgnVCM200000bb42f00aRCRD.htm

Q & A



Catalin Bidian

catalin.bidian@utoronto.ca