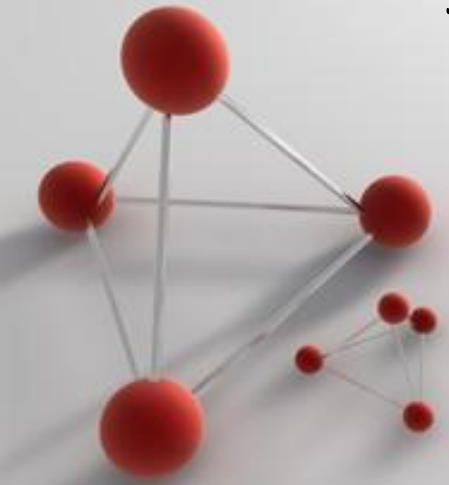
Security Assessment
-
Web Application

Summary

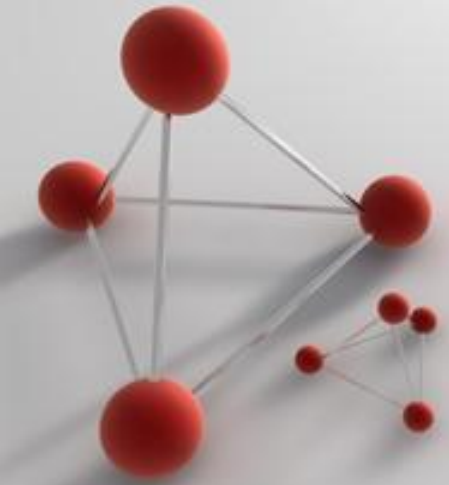
1. Introduction
    a. Types of security assessment
    b. Mission
2. Methodology
3. Tools
4. Pwn Me if You Can
5. Security report



1. Introduction

The web application is manipulated through a web navigator.
It’s stored on the servers side and is an interface (middleware) between the client and the database server.

It’s a 3-tier architecture application.



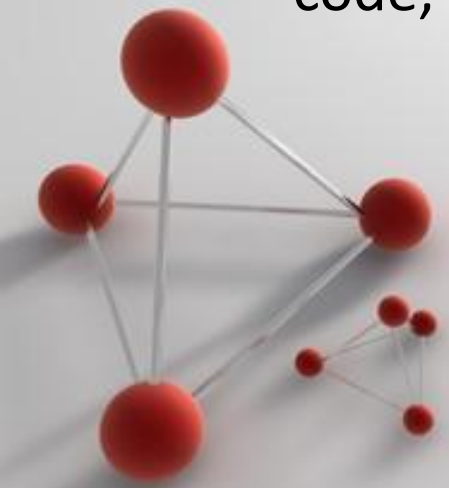
1-a. Types of Assessments 1/3

Important axes are :

A- Black box: No information is given, except the starting point.

B- Grey box: Limited information (database scheme, application's design…).

C- White box: Detailed information (application's source code, design and database scheme).



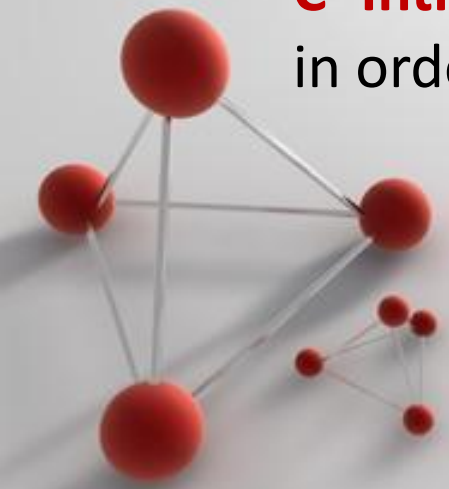
1-a. Types of Assessments 2/3

Assessments can be categorized in three types:

A- Verification: based on the experience. It is oriented to the organization, architecture, protocols and configuration assessment.

B- Validation: validate a system's security at a given time according to a referential

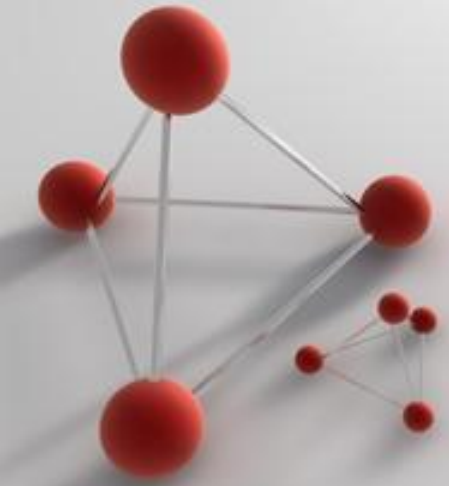
C- Intrusive: active search of vulnerabilities and weaknesses in order to take the control of a maximum of components.



1-a. Types of Assessments 3/3

In a simple way:

1. Thick client
2. Web application
3. Surface of exposition
4. Physical security
5. Intrusive security assessment (internal or external)



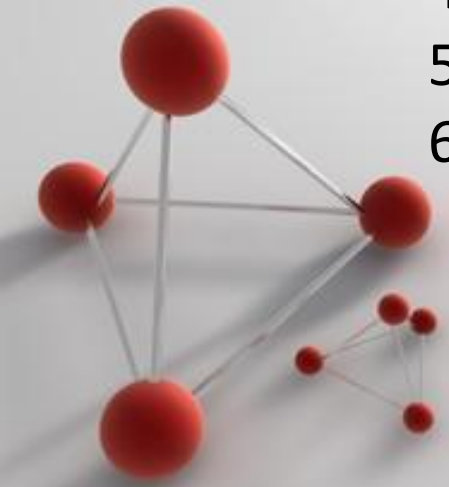
1.b. Mission

A mission is divided into several types:

Remotely / On site

Timeline:

1. **First contact:** 15mn, 1-3 day before starting
2. **Mission's start:** starting at 9h30, lunch 12h00, pause 16h30, end of the day 18h30.
3. **Regular checkups:** 1 every 2 days
4. **Final checkup:** duration of 30 to 45mn
5. **Report writing:** 30 to 40% of the mission's duration
6. **Delivery:** approx 3 hours

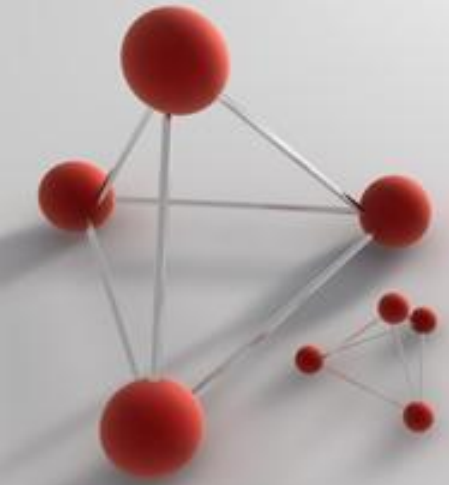


2. Methodology

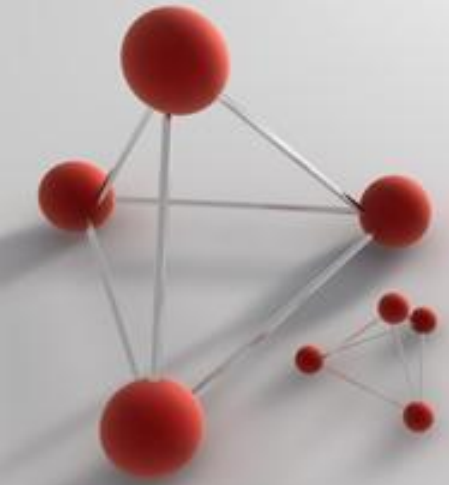
The security assessment moves across those points:

1. Discovery
2. Vulnerability research
3. Exploitation
4. Privileges escalation

Again…, do never cross the scope !



Scope’s discovery

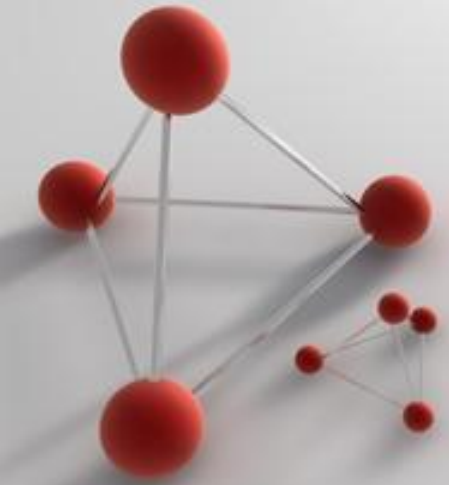


2. Passive Discovery

1/ Website crawling, HTML code and structure analysis

2/ Keyword based searches on different search engines:

- The number of results allows the evaluation of the size of the targeted company
- Unexpected results when using more sophisticated search queries



2. Google Dorking / Hacking

Google's search operators:

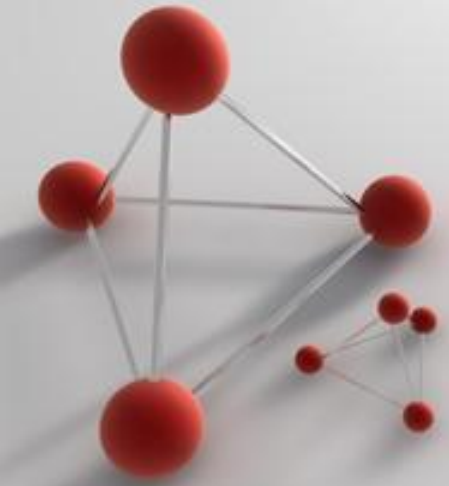
- site: filters the output based on the domain name (eg. site:www.microsoft.com)
- filetype: filter based on a defined file extension (eg. filetype:pdf)
- intitle: keyword contained in the page's title (eg. intitle:administration)
- intext: keyword contained in the body of the page (eg. intext:"index of /")
- inurl: keyword present in the URL (eg. inurl:backup)

2. Google Dorking / Hacking

Examples:

- Looking for a DVDrip film in a directory listing:
Intitle:"index of /" intext:"index of /" intext:dvdrip intext:movie_name

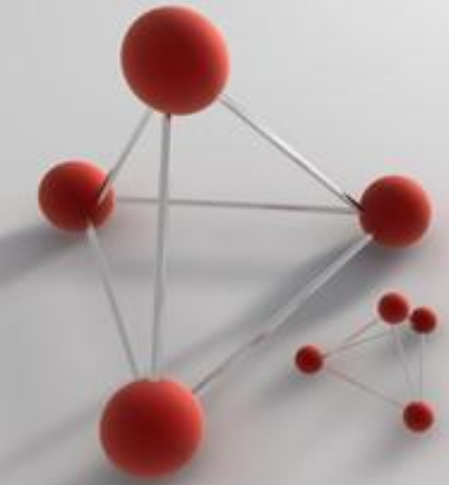
- Looking for the backup of some databses:
inurl:backup inurl:wp-content filetype:sql



2. Active Discovery

Directly requesting the target or one of its components:

- DNS (DNS server, sub-domains, IP addresses, …)
- Network discovery (ping, specific TCP/UDP ports, ...)
- TCP/UDP port scan
- Specific protocols: SMB, SNMP, SMTP, FTP, HTTP, ...



2. Active Discovery - DNS

DNS: service used for domain name translation from FQDN string to an IP address and vice versa

Tools: host, dig, nslookup

- Get the authoritative DNS servers:
$ host -t ns google.com

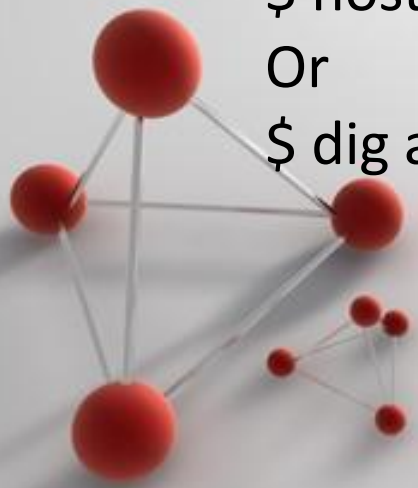
- Get the list of mail servers:
$ host -t mx google.com

2. Active Discovery - DNS

- Get the IP addresses associated with the domain name:
$ host www.google.com

- Get the domain names associated with an IP :
$ host 216.58.208.196

- Get all the recorded information in the Authoritative DNS server (Zone Transfer) :
$ host -l ns1.google.com
Or
$ dig axfr @ns1.google.com google.com



2. Active Discovery - DNS

-More sub-domains can be obtained through Google Hacking:
site:google.com -site:www.google.com

-More can also be obtained by bruteforcing:

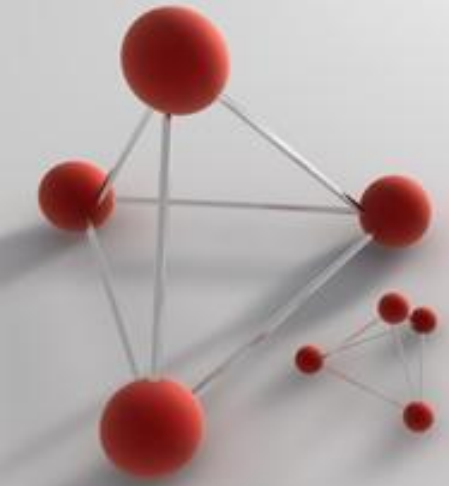
```
woody@tank ~ $ for subdomain in $(cat ~/tools/wordlists/john.txt); do \
> host $subdomain.google.com 2>&1 | grep -v 'not found:'; \
> done
upload.google.com is an alias for large-uploads.l.google.com.
large-uploads.l.google.com has address 216.58.198.239
large-uploads.l.google.com has IPv6 address 2a00:1450:4007:812::200f
help.google.com is an alias for www3.l.google.com.
www3.l.google.com has address 216.58.204.238
www3.l.google.com has IPv6 address 2a00:1450:4007:813::200e
```

2. Active Discovery – Network Hosts

ICMP requests allows the identification of active hosts.
Outils: ping, nmap, ...

- Send ICMP Echo requests and receive a Reply :
$ ping www.google.com

- Same with nmap :
$ nmap -sP www.google.com



2. Active Discovery – Port Scanning

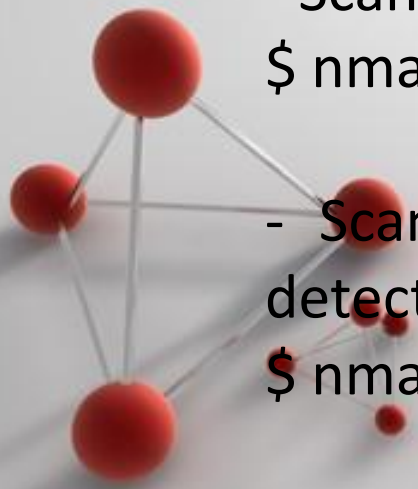
Un port scan allows the discovery of all open and not filtered ports on a target.

Outils : nmap, ...

- Scan the most commonly used TCP ports:
$ nmap www.google.com

- Scan the 80/TCP port without performing a Ping:
$ nmap -Pn –p80 www.google.com

- Scan TCP ports from 20 to 200 with OR and version detection:
$ nmap -p20-200 -A www.google.com

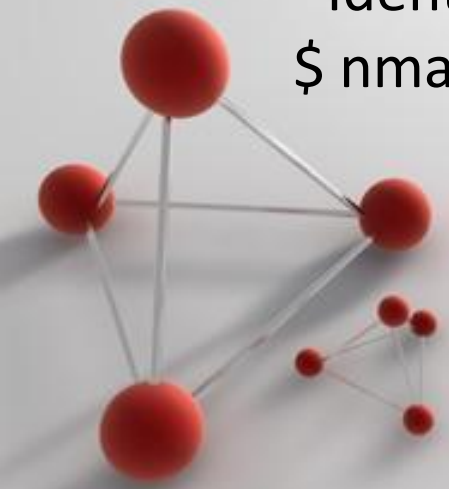


2. Active Discovery – Port Scanning

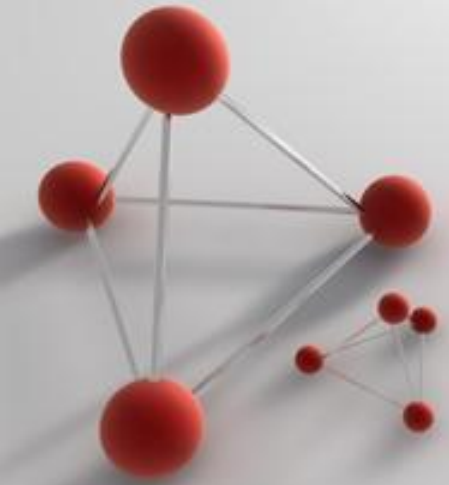
The nmap scripts (NSE – Nmap Scripting Engine) allow execution of automatic tasks (DNS Zone Transfer, FTP bruteforce, SNMP discovery, …).
Available in: /usr/share/nmap/scripts

- Execute all NSE scripts:
$ nmap --script all www.google.com

- Identification of the MS08-067 vulnerability:
$ nmap --script smb-vuln-ms08-067 www.google.com



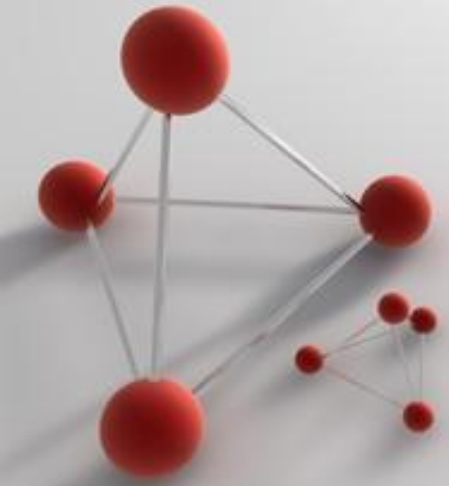
Web application’s assessment



3. Minimal Toolbox

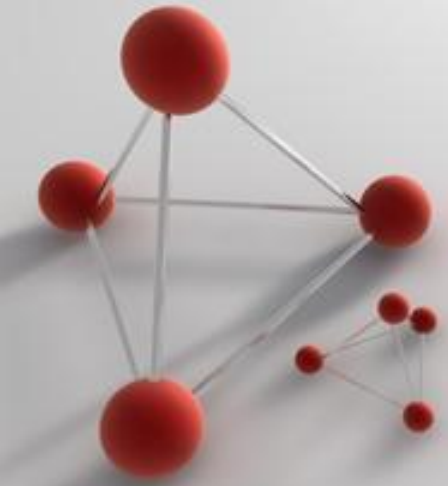
1. **Web navigator:** Firefox, Chrome, Internet Explorer, …
2. **Proxy:** Burp, WebScarab, Paros, …
3. **Data manipulation:** FireBug (FireCookie), Live HTTP Headers, …
4. **Network stream:** Wireshark.

All this in a virtual machine!



3. Vulnerabilities

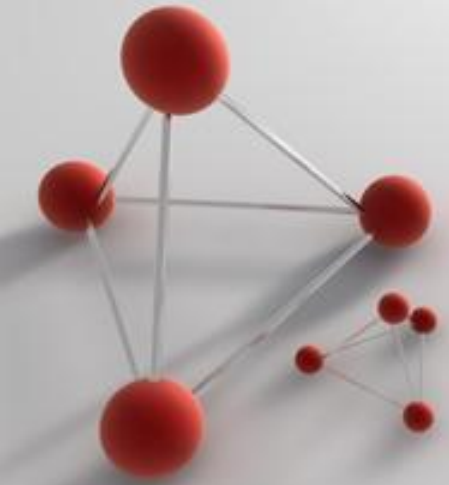
1. Cross-Site Scripting (XSS)
2. Cross-Site Request Forgery (CSRF)
3. Arbitrary code execution
4. Arbitrary command execution
5. Path traversal
6. Local file inclusion (LFI)
7. Remote file inclusion (RFI)
8. Arbitrary file upload
9. SQL injection (SQLI)
10. Bad segregation
11. Full path disclosure (FPD)
12. Technical information disclosure
13. Cleartext communication
14. Bad configuration of the Cookie
15. Directory listing
16. …



E-COMMUNE

…

Pwn Me if You Can



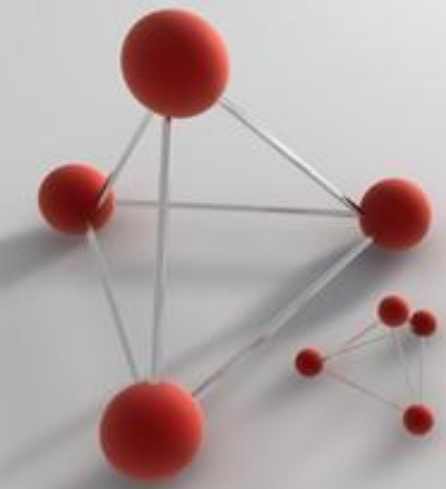
5. Report

Synthesis:

a. Scope reminding
b. Most critical vulnerabilities
c. Business risk
d. Recommendations

Vulnerability sheet:

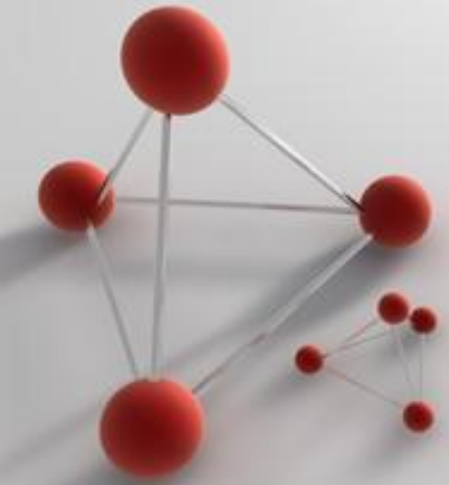
a. Short and clear title
b. Indicators of risk (risk, exploitation and remediation)
c. Description
d. Exploitation (the more detailed possible)
e. Remediation (the more detailed possible) + External URL reference that details more the remediation.



I’m a report

…

Write Me if You Can



End.

Questions ?