System Security Notes
Sept 24th 2010

Topics covered in previous lecture
- Stack overflows.
- Buffer overflow.
- Integer overflows.
- SQL injection.
- Shell command injection.

Cross-site Scripting
- Involves stealing user’s cookies to gain access to its account.
- Cookies are files stored by web-browsers containing user’s authentication information. This information is passed on whenever user makes a request to the web server.
- The cookie forwarding request follows same-origin-policy.
- Contains key value pairs.
- In most of the cases, this attack is carried out using Javascript.

Steps Involved:-
- Client sends authentication information to the Web Server.
- Web Server sends the client a cookie.
- The client’s browser stores the cookie.
- For any requests made to this web server, the client sends the cookie with the request for authentication.
Line of attack:-
- The attacker binds the cookie stealing code in the pages coming from the webserver.
- The attacker tricks the client to execute Javascript code which steals cookie.
- A malicious user can craft amazon.com review to include his Javascript which might be executed by the client.
- The stolen cookie, again, can be posted as a review and retrieved by the attacker.
- It can then be used to gain access to user’s account.

Defenses: -
- Appropriate place of putting defenses. Client or Web Server?
- Web Server is provides both code and data channels to the attacker.
- Code channel needs to be plugged.
- Validate input using a whitelist (failsafe defaults).
- Using CAPTCHA might prevent sending cookie back to the attacker.
- IP addresses in the cookie? Breaks availability.

Cross-site Request Forgery (CSRF)
- A malicious webpage provides a hyperlink.
- On clicking, that hyperlink opens in the same browsing window and thus can use cookie information stored in the browser process.
- For example, an attacker wants to transfer some money from victim’s account to his/her account. The money transfer request looks something like GET transfer.html?src=..&dst=...&amount=....
- The user is tricked to click a hyperlink which generates this malicious request.
- If the victim happens to have an active session to bank website, this request will use those cookies and the request will be successful.

Defenses: -
- Add random numbers to GET requests.
- For critical operations like money transfer, a confirmation might be useful. Though, it requires involvement of user.
- Encrypted URLs.
- Using POST instead of GET.
Critical information disclosure in Error Messages
- Printing SQL queries on failure.
- Divulges information on database structure.
- Address information provided in a BSOD message.

Defenses:
- Generic error messages should be posted by any webserver.
- Failsafe defaults.

Tractor Beaming Attack
- Observed in UNIX.
- A process runs as with two uids, real uid and effective uid.
- FTP servers runs with ruid=0.
- Changes user to euuid=1001 (say) on receiving request from user.
- SITE EXEC allows running command.
- If an exception occurs, the effective uid will be 0.
- Does not reset to 1001 on exception.
- Comefrom instruction. Opposite of goto. JOKE!!

Do Special Command (...)
{
    Seteuid(0);
    If (...)            
        Throw exception.
    Seteuid(1001);
}
Eventloop {...}
{
    Try
    {
        Do Special Command ();
    }
    Catch ( e ){}
}

Defenses
- Transactions. If exception occurs, control returns to OS.
- Run FTP server with reduced privileges.
- Reset privileges in catch or just before throwing the exception.
- Include privileges as a part of static scope of a block.