Continued from last class

- Idle Scan
  - A portscan is noticeable by the victim
  - Idle scan lets you scan without the victim being aware

An idle scan works by creating a valid connection to the victim (i.e. web server) and sending a spoofed SYN (source address of the victim) to a 3\textsuperscript{rd} party. The 3\textsuperscript{rd} party will then send a SYN-ACK to the victim and then if the victim has a service running on that port then it will respond to the 3\textsuperscript{rd} party with a RESET (because the victim did not send out the SYN). This will allow the attacker to see a hole in the IP-ID (16bit) value of the packets being sent out by the victim – most OSs sequentially increment the IP-ID. The IP-ID and OFFSET values of the packet are used to manage the fragmentation of large data packets; assembly is done at the receiving end host - packets with the same IPID are combined using the OFFSET.

To counteract this, you can disable the RESET response from an unsolicited SYN-ACK.

**Web Security**

- Phishing/Pharming Attacks
Phishing is the act of tricking someone into giving up confidential information or tricking them into doing something that they normally wouldn’t do or shouldn’t do. For example: sending an e-mail to a user falsely claiming to be an established legitimate enterprise in an attempt to scam the user into surrendering private information that will be used for identity theft. Phishing requires the active participation of the victim. Phishing can be done using doctored urls. The general form of a url is Protocol://userid:passwd@domain-name/path e.g. ftp://guest:anonymous@download.com/foo.html. However the userid and passwd is ignored when the protocol is http and can be used to confuse a user, e.g. http://www.bankofamerica.com@phisher.com When the user clicks on this url they are taken to phisher.com. The casual user will only “see” www.boa.com and not pay attention to the rest assuming it is the path but the browser ignores everything to the left of the @ and points the user to phisher.com.

Pharming is much more insidious and involves diverting the user’s request. It requires little or no active participation by the user and even sophisticated users can get taken in. Using DNS cache poisoning or by poisoning BGP routes the attackers divert legitimate traffic to the fake website.

Even with https people can be fooled. In normal usage when the user visits a website if the domain name the user typed in is different from the CN (common name) in the certificate then the browser pops up a warning box. Many people ignore this warning. But even if the user is sophisticated enough to read the warning, the user can sometimes be misled by a popup that covers the users address bar with a url whose domain-name matched that in the warning box and is suspiciously close to the actual name.

HTTP Headers

When your browser views a website it sends a request to the server, and the server sends a response. Headers are also sent back and forth.

From:
The From header has information about the user: eg; name, email, address. It has been deprecated since its creation and is now mainly used by “polite” spiders.

Cookie:
HTTP is a stateless protocol. The cookie header allows for a site to save state. A cookie can be 4KB max. It consists of name and value pairs. Additionally a cookie contains a TTL (time to live) and domain (at least 2 dots “.yahoo.com”) value. The domain is the domain to which the browser is expected to release the cookie. Good security settings only allow own-domain cookie placement. You should not place sensitive information in a cookie, unless it is adequately encrypted and authenticated because otherwise attackers may craft fake cookies.

**Authorization:**

The Authorization header is used to get a userid and password. The authorization header has 2 modes, basic and digest.

1. Basic sends userid/ password in clear text
2. Digest sends userid,hash(password@server,none:nonce-count) along with integrity check
   a. Hash(password@server) is used to make a unique password that is not in clear text, even on the server; the addition of “server” prevents replay attacks on other servers.
   b. Nonce prevents replay attacks
   c. Nonce-count saves round trip time cost by using the original header multiple times and increasing the count by one each time.

**Referer:**

The referer header contains the URL of the site used to travel to the current site. This can be a violation of the user’s privacy if sensitive information is stored in the URL, especially when moving between HTTPS and HTTP sites.

**Tracking**

**Web bug:**

A Web bug is a 1x1 pixel gif that appears invisible to the user. The link for the web bug is on a different domain name – the advertising or tracking company’s domain name. It is used to monitor and track the activity of users across different domains. Without web bugs website operators would have had to exchange logs to track users across multiple domains.

**Active Content**

Java – runs in a sandbox so it can not harm your computer
Javascript – very limited functionality, can do limited monitoring (can view your clipboard)
ActiveX – Digitally signed (dangerous if not from a trusted site) but can do anything once installed on your machine.
Cross scripting attack:

Client goes to A. In A there is an embedded link to B that has a query string with JavaScript in it. Suppose B sends the query string back to the end-user in html then this Java Script would run on the client and perform operations on B, potentially malicious operations engineered by A. Though this example is a little contrived it demonstrates the potential for mischief that exists – even though Java Script is limited in its capabilities on the client nevertheless it can execute malicious operations on the server.

Public Computers:
Forms & Passwords
Public computers are dangerous because any passwords stored will be available to the next person who uses the computer. Also a key logger or camera could be recording your every keystroke.

Google
Google has 12 data centers around the world (Educated guess, they keep it a secret)

How google works:
DoS Google

- Google bomb: Harvested websites all have the same keywords and point to one website making the targeted website seem like a authority on that topic and is boosted in its Google ranking.
- Many terms, repeatedly: currently Google limits their queries to 10 terms, but finding the intersection of all 10 terms is expensive. One attack involves a screensaver that looks normal, but in the background it sends a 10-term query to Google constantly. When many people download this hacked screen saver they can DoS Google. Google “solved” this problem by implementing a quota on queries per IP in a certain time period.

BGP Anycast

There are 13 root DNS servers \{a – m\}.root-servers.net

They are protected by BGP anycast.

If one IP gets flooded then others are still open.