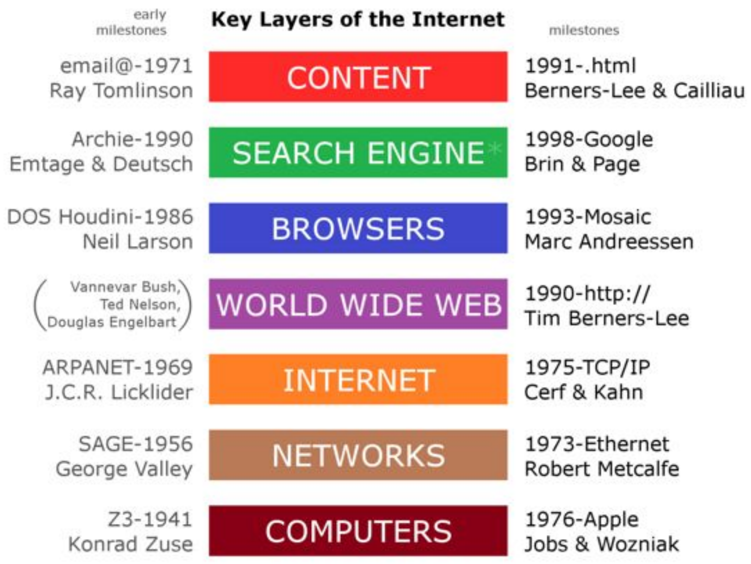
Writeup 7
CS 373
James Tyler Ball

Web Security Lesson 1 (Cedric Cochin):

HTTP enables data communication on the Web. Browsers and technology move very quickly. In a short amount of time attacks or defenses can be adjusted to change what is effective and what isn't.



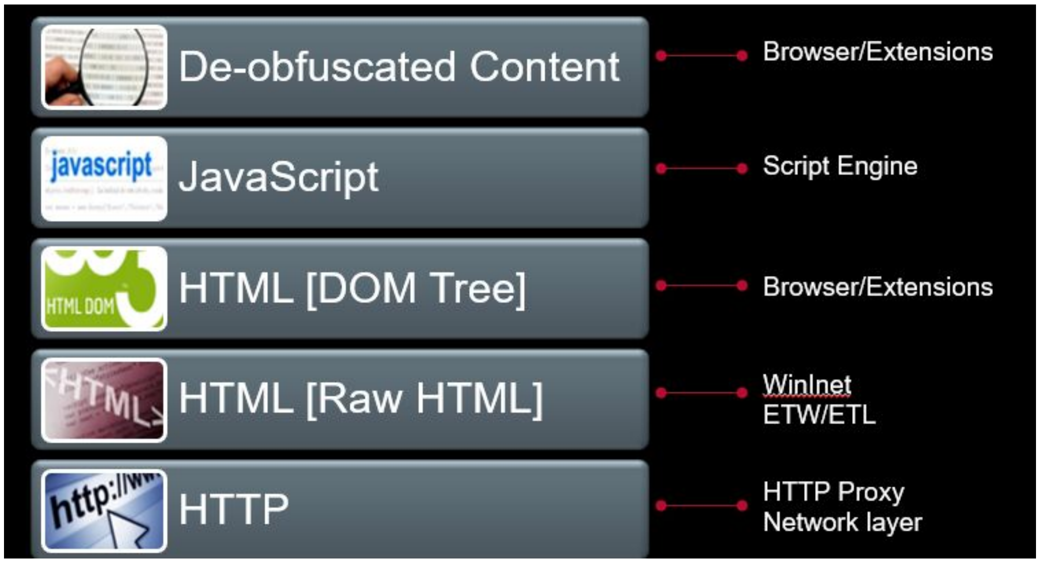
early milestones
Key Layers of the Internet
milestones
email@-1971 Ray Tomlinson
CONTENT
1991-.html Berners-Lee & Cailliau
Archie-1990 Emtage & Deutsch
SEARCH ENGINE
1998-Google Brin & Page
DOS Houdini-1986 Neil Larson
BROWSERS
1993-Mosaic Marc Andreessen
Vannevar Bush, Ted Nelson, Douglas Engelbart
WORLD WIDE WEB
1990-http:// Tim Berners-Lee
ARPANET-1969 J.C.R. Licklider
INTERNET
1975-TCP/IP Cerf & Kahn
SAGE-1956 George Valley
NETWORKS
1973-Ethernet Robert Metcalfe
Z3-1941 Konrad Zuse
COMPUTERS
1976-Apple Jobs & Wozniak

Components of the infrastructure we rely on now was established ~30 years ago.

95% of malware are delivered via the web. The methodology has changed continuously though. Script bombing was popular early on, now MitB is more common. Phishing has been consistent, but the type has changed. Near the turn of the millenium java was especially vulnerable.

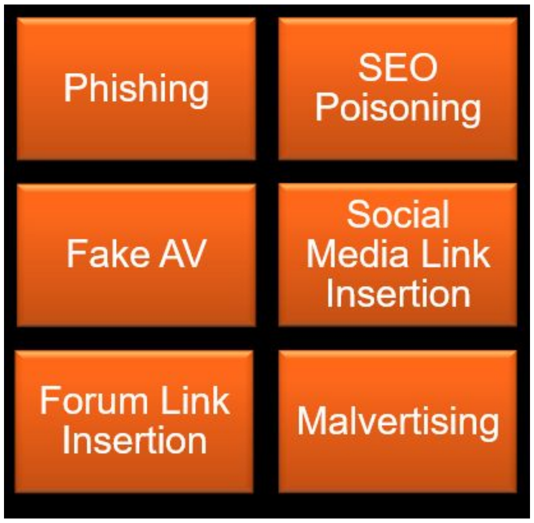
Web Browser 1.0 had a layer of javascript that helped with the render. Web Browsers now use javascript to interact with every layer (the Network/Protocol, DOM, and Graphical Interface/Input Devices). This helps the browser become much more versatile.

There are many opportunities for attacks to reach the browser:



HTTP attack can be done on the host. Raw HTMLcan use WinINet to view traffic. Different browsers have different script engines.

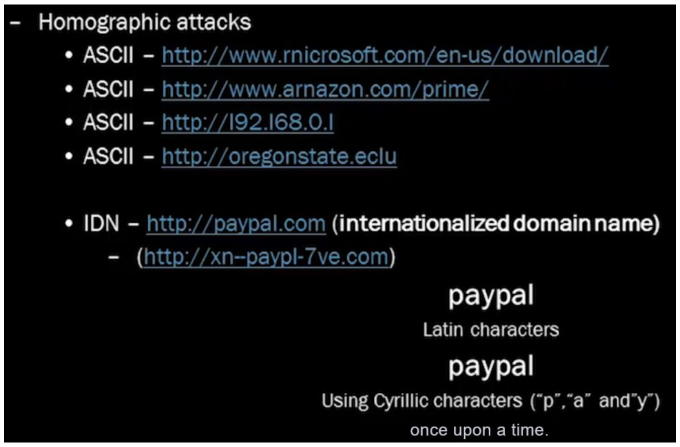
Many attacks use social engineering to persuade victims to willingly execute malicious code.



No matter how resilient the browser, it is impossible to account for all the user behavior - ‘clickaholic’. This can lead to the user installing malware, launching executables, activating browser exploits, or filling out malicious forms. Malicious sites can work with each other to try and promote their priority on google searches.

- Phishing sites can copy another page’s appearance, have a similar URL, get certified, all to trick users.
- Malicious sites also use google trends to lure victims to get infected. Google’s image search is not as solid as their text search.

- Fake update prompts that appear legitimate can trick users into installing malware.
    - This is especially hard to detect as malware because the user agrees to each step.
- using rn for m, I for l, cl for d, can all be used to trick users



- Homographic attacks
    - ASCII – http://www.rnicrosoft.com/en-us/download/
    - ASCII – http://www.arnazon.com/prime/
    - ASCII – http://l92.l68.0.l
    - ASCII – http://oregonstate.eclu
    - IDN – http://paypal.com **(internationalized domain name)**
        - (http://xn--paypl-7ve.com)

paypal

Latin characters

paypal

Using Cyrillic characters ("p","a" and"y")

once upon a time.

    - this is less effective now

Fake social network accounts can lend credibility to malicious attackers. It can also help to gather information on victims. Catfishing can also be used to gather info on victims.

It is important to note that not all attacks either have a specific victim in mind or target anyone on windows. Malvertising and waterhole attacks can tailor attacks knowing that only certain potential victims will encounter their software. Think of putting a malicious ad or injecting code on a specific website.

How do we defend against these attacks?

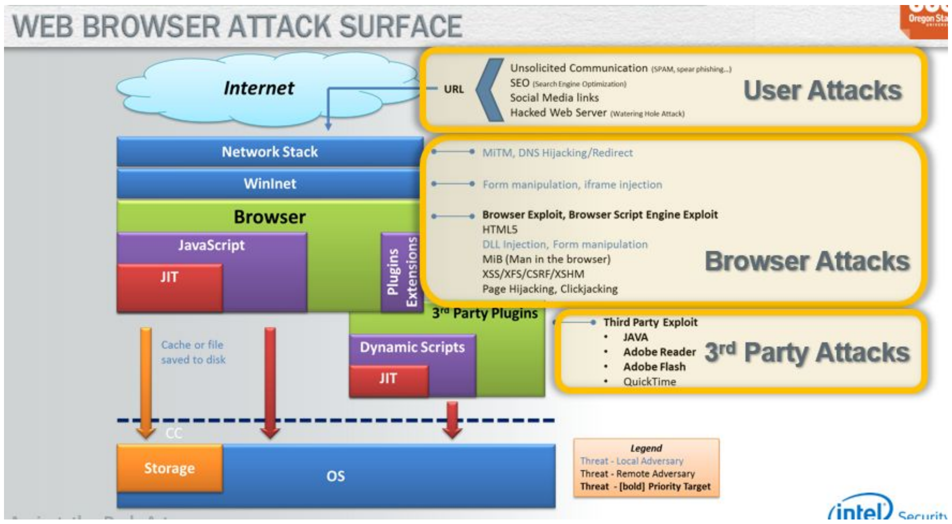
- have browsers look at a URL's reputation
    - URL annotation works well for older people
- look for certification
    - PCI
    - who signed the website?
- gateways are still effective
- URL shorteners need to be maintained and governed
    - adds a layer of authenticity
- train users
    - this is hard, but it works with banking
- note that with the increasing mobile traffic, users just want to click links and not type

There is one website that is always ahead of the banking industry in terms of authentification: pornography

Note that many attackers time their actions around when they expect the least resistance (Friday evening). West coast security is the last line of defense.

Browser-level attacks: get the user to click on the next step.

- A browser is tasked with protecting itself from outside threats as well as threats from the local machine seeking to use the browser
- Multi-stage authentication is a great counter to this and other trojan attacks, but asks more from the user
- On mobile, adding additional confirmation checks was helpful in preventing users from installing malware
- User attacks can work with browser attacks (especially on mobile and with the help of botnets) to install malware

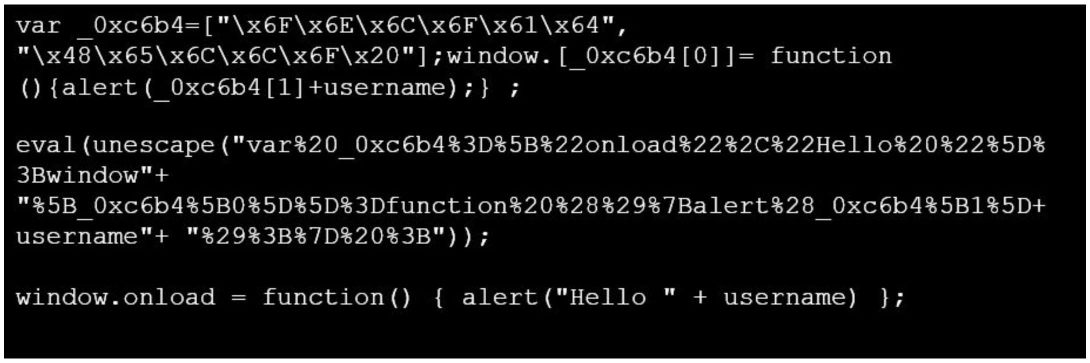

WEB BROWSER ATTACK SURFACE
Internet
URL
Unsolicited Communication (SPAM, spear phishing...)
SEO (Search Engine Optimization)
Social Media links
Hacked Web Server (Watering Hole Attack)
User Attacks
Network Stack
WinInet
Browser
JavaScript
JIT
Plugins Extensions
MiTM, DNS Hijacking/Redirect
Form manipulation, iframe injection
Browser Exploit, Browser Script Engine Exploit
HTML5
DLL Injection, Form manipulation
MiB (Man in the browser)
XSS/XFS/CSRF/XSHM
Page Hijacking, Clickjacking
Browser Attacks
3rd Party Plugins
Dynamic Scripts
JIT
Third Party Exploit
JAVA
Adobe Reader
Adobe Flash
QuickTime
3rd Party Attacks
Cache or file saved to disk
CC
Storage
OS
Legend
Threat - Local Adversary
Threat - Remote Adversary
Threat - [bold] Priority Target

Browser security features

- modern browsers like chrome and firefox are like small OSs
- browsers try to isolate themselves from the rest of the system for security

browser exploits are now multi step processes as mentioned above. Often malware requires the user to so something to install malware on the system.

javascript is all over http, and is an interpreted language with a number of potential security issues.



```
var _0xc6b4=["\x6F\x6E\x6C\x6F\x61\x64",
"\x48\x65\x6C\x6C\x6F\x20"];window.[_0xc6b4[0]]= function
(){alert(_0xc6b4[1]+username);} ;

eval(unescape("var%20_0xc6b4%3D%5B%22onload%22%2C%22Hello%20%22%5D%
3Bwindow"+
"%5B_0xc6b4%5B0%5D%5D%3Dfunction%20%28%29%7Balert%28_0xc6b4%5B1%5D+
username"+ "%29%3B%7D%20%3B"));

window.onload = function() { alert("Hello " + username) };
```

javascript is a great language to obscure purpose and make detection difficult for observers. McAfee has found malicious packers on more than one “good” site.

eBay sellers will often obfuscate code intentionally to prevent their posting from being stolen. Javascript can also trick the user into seeing a different bidding deadline.

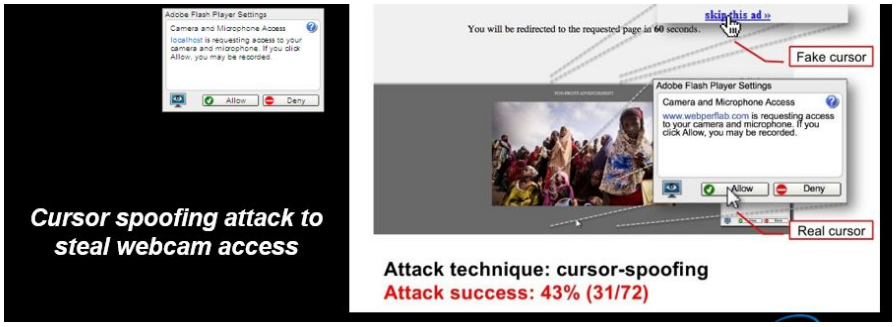
Offering free WiFi is a great way to launch a MitM attack. A lot of things that should be encrypted are not (credentials), so this is an attractive option for attackers. It is also possible to inject packets this way, as mentioned in a previous lecture. MitM can also be used on home networks.

MitB (man in the browser) is related to MitM. This can intercept traffic from within the browser.

DNS spoofing - another method to perform phishing, exploits. USB sticks can disguise themselves as keyboards, network adaptors, or any number of things.

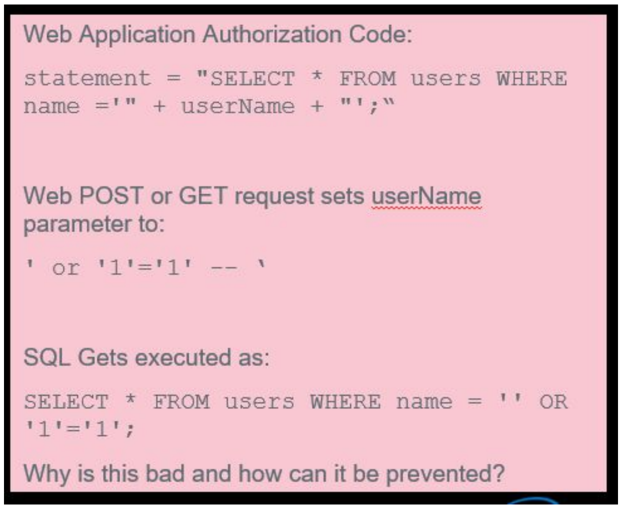
- a network connection could cause DHCP (dynamic host configuration protocol) to turn on automatically and move sensitive data

Clickjacking - waits on the user to click something, but fools the user into clicking something else

- put a button over another button
- 
Adobe Flash Player Settings
Camera and Microphone Access
localhost is requesting access to your camera and microphone. If you click Allow, you may be recorded.
Allow
Deny
Cursor spoofing attack to steal webcam access
You will be redirected to the requested page in 60 seconds.
skip this ad »
Fake cursor
Adobe Flash Player Settings
Camera and Microphone Access
www.webperflab.com is requesting access to your camera and microphone. If you click Allow, you may be recorded.
Allow
Deny
Real cursor
Attack technique: cursor-spoofing
Attack success: 43% (31/72)

- can also be used for text input

SQL injection - most things on the web are driven by some kind of database and many actions interact with that database in some way

- there may by exploits in the design that allow for user input to inject SQL code that interacts directly with the database
- developers often carry the mentality that user input doesn't include the URL
- this can be used to get sensitive data (passwords) from the database, give privileges to the user
- 

  Web Application Authorization Code:

  ```
  statement = "SELECT * FROM users WHERE
  name ='" + userName + "';"
  ```

  Web POST or GET request sets userName parameter to:

  ```
  ' or '1'='1' -- '
  ```

  SQL Gets executed as:

  ```
  SELECT * FROM users WHERE name = '' OR
  '1'='1';
  ```

  Why is this bad and how can it be prevented?
- usually try a quote first
- error driven SQL injection - MS-SQL will tell the user it can't convert a string (*question interrupted this section*)
- blind SQL injection - all you need is "error" or "not error"

Same origin policy attacks

- browsers are not standardized
- I think the big takeaway from this segment was that browsers are often the main way a local and personal system interacts with the outside world, and as such need to balance a series of increasingly diverse responsibilities while keeping themselves isolated enough so as to protect the local system

Cross-site scripting

- stored cross-site scripting can be kept in a link or a URL
  - new website may ask user to do something on a new website (cookies or credentials)
- SQL injections

- goal is to put attackers scripts into the victims browsers

Non-persistent XSS example - look for javascript in the URL

Cross-site request forgery (XSRF) - exploit the server trust in the browser.

- bypass SOP
- instruct someone to perform some action on the server on your behalf
    - could be a GET request
    - create a POST request
    - victim's requests have authentication due to cookies
- could be done by clicking an image or a link

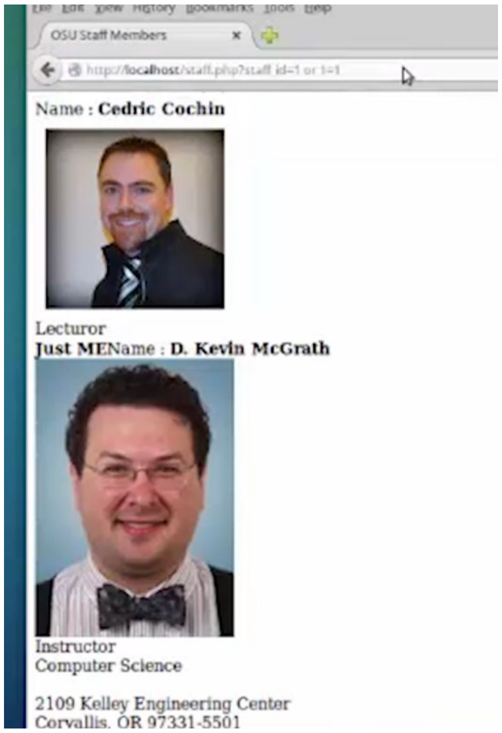
WEB GOAT - free for all attack target that has multiple lessons, fake target so it's not a perfect replica, but a good series of exercises

- remember that one big limitation of VM research is that you may not have internet access when you need it to look for malicious activity

Web Security Lesson 2 (Cedric Cochin):

The lecture opens with an activity in which Cedric goes to a local website he created, and alters the id to get to different individuals. He asks what might happen if he replaces an integer with an algebraic expression. The result depends on how the website was authored. If the algebraic expression is evaluated, then it might mean that the message is interacting with the database directly.

You can also find out if a website is supposed to fetch all rows that match the criteria to better understand the behavior of the website:



```
http://localhost/staff.php?staff id=1 or 1=1
```

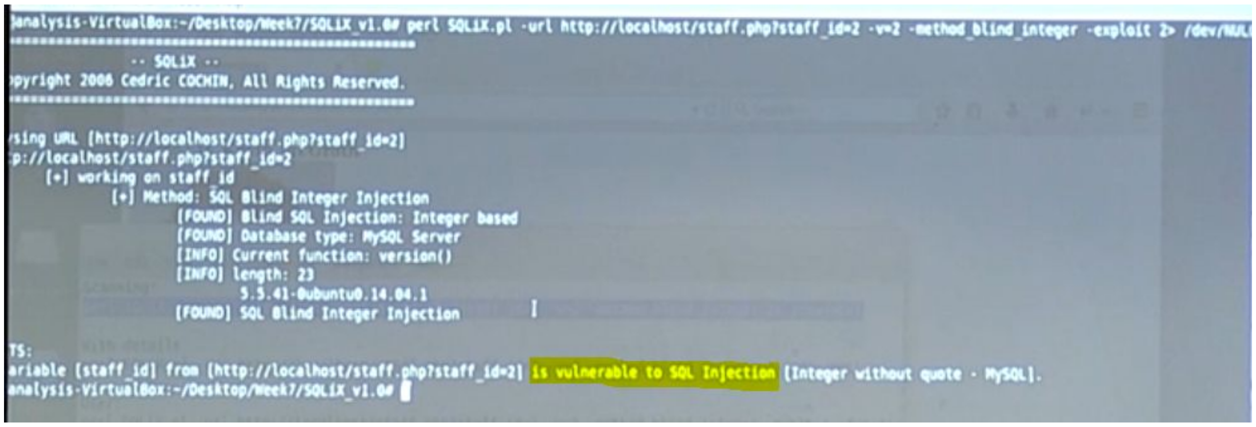
Name : **Cedric Cochin**

Lecturor
Just ME Name : **D. Kevin McGrath**

Instructor
Computer Science

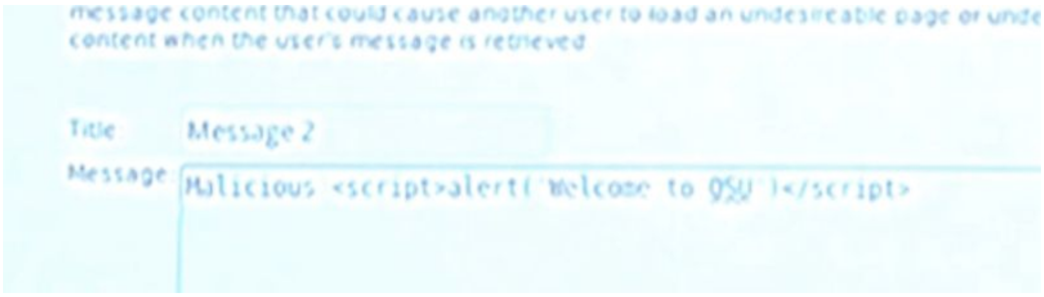
2109 Kelley Engineering Center
Corvallis, OR 97331-5501

Next Cedric introduced us to a PERL script he wrote to help find exploits. It looked at a URL and was able to describe how it was vulnerable. I don't know PERL, but this would be an interesting script to explore.



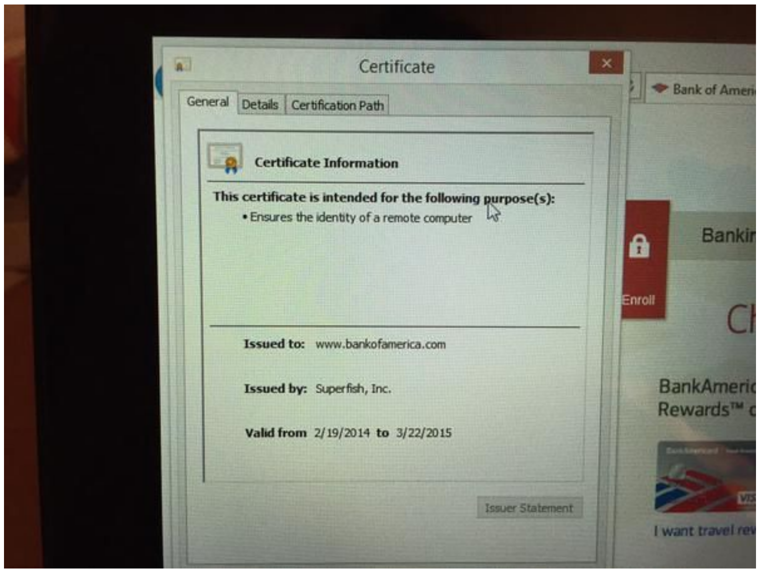
```
analysis-VirtualBox:~/Desktop/Week7/SQLiX_v1.0# perl SQLiX.pl -url http://localhost/staff.php?staff_id=2 -v=2 -method blind_integer -exploit 2> /dev/NUL
==============================================
               -- SQLiX --
pyright 2006 Cedric COCHIN, All Rights Reserved.
==============================================

sing URL [http://localhost/staff.php?staff_id=2]
p://localhost/staff.php?staff_id=2
     [+] working on staff_id
            [+] Method: SQL Blind Integer Injection
                    [FOUND] Blind SQL Injection: Integer based
                    [FOUND] Database type: MySQL Server
                    [INFO] Current function: version()
                    [INFO] length: 23
                            5.5.41-0ubuntu0.14.04.1
                    [FOUND] SQL Blind Integer Injection

TS:
ariable [staff_id] from [http://localhost/staff.php?staff_id=2] is vulnerable to SQL Injection [Integer without quote - MySQL].
analysis-VirtualBox:~/Desktop/Week7/SQLiX_v1.0#
```

Cedric next outlined the XSS labs for his local website. Stored XSS is "when you want a value in a database and someone else could see what's inside" (student answer). Cedric was able to insert a script into a text input and execute it by clicking a link:


message content that could cause another user to load an undesireable page or unde
content when the user's message is retrieved

Title	Message 2
Message	Malicious <script>alert('Welcome to OSU')</script>

Cross-site scripting vulnerabilities are extremely common. He also mentioned that Lenovo shipped some products with MitM software pre-installed. I use a lenovo laptop, so I paused the lecture and found that the software in question is ‘Superfish’:


Certificate

General | Details | Certification Path

Certificate Information

This certificate is intended for the following purpose(s):
- Ensures the identity of a remote computer

Issued to: www.bankofamerica.com

Issued by: Superfish, Inc.

Valid from 2/19/2014 to 3/22/2015

Issuer Statement

The software hijacks web sessions. Luckily for me my laptop is too old for Superfish.

Getting back to XSS, note that developers may put restrictions on user input to add security. However, if this is done via javascript, then it is on the user’s side and can be modified.

We covered FUZZING before - trying many different variants of variables or extensions until something breaks.

Alexa (not amazon product) - not URL based, a mechanism for learning how prevalent a domain is. Doesn’t indicate clean or not clean. Usually if a domain is “bad” it will become prevalent quickly and then die - could be a phishing company.

Archive.org - frequently used by researchers to look at websites at the time they were hacked.

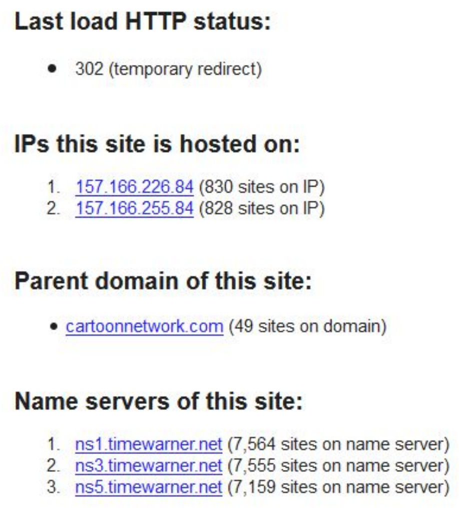
IPVOID - bad guys will use the same IPs over and over. Look for suspicious IPs on a blacklist.

Fast flux is the behavior of malware to potentially use one name but constantly change IP

Good question by student - should a botnet author try to get a phone home to a domain name or an IP address? An IP would die relatively quickly. A domain name would be susceptible to sinkholing and redirection. Usually an author will use an algorithm to create random names over time.

CHECKSHORTURL - many instances of malware hide behind shortened URLs, and it's hard to know where a shortened URL will go. Use a tool to see where the URL points to

SIte Dossier - used to gain a bigger picture view of a site, includeing HTTP status, IP hosts, etc:



Last load HTTP status:

- 302 (temporary redirect)

IPs this site is hosted on:

1. 157.166.226.84 (830 sites on IP)
2. 157.166.255.84 (828 sites on IP)

Parent domain of this site:

- cartoonnetwork.com (49 sites on domain)

Name servers of this site:

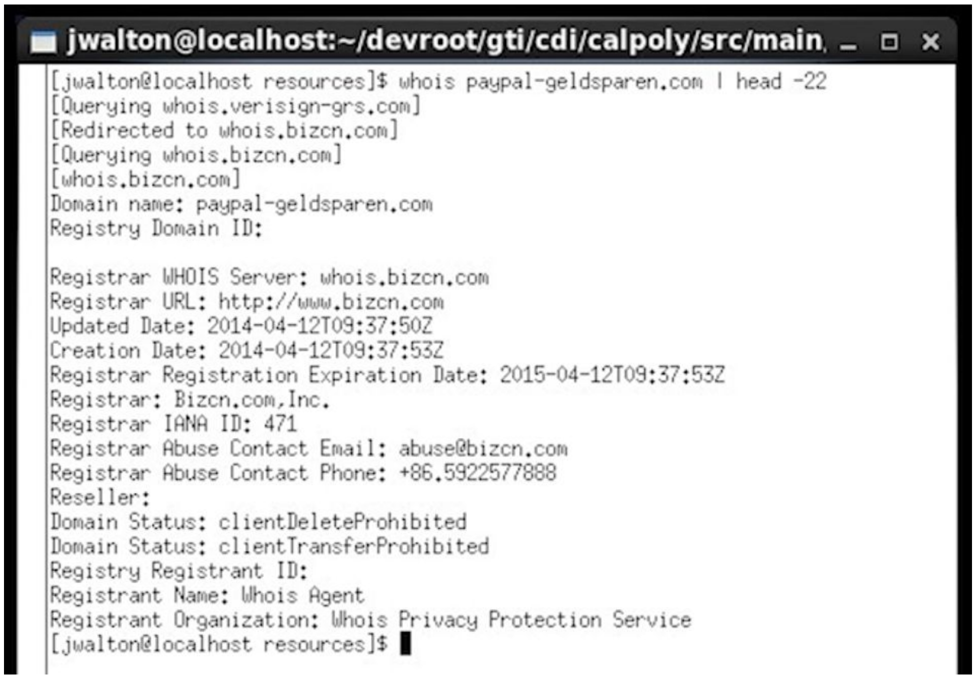
1. ns1.timewarner.net (7,564 sites on name server)
2. ns3.timewarner.net (7,555 sites on name server)
3. ns5.timewarner.net (7,159 sites on name server)

WEBUTATION - check the recent reputation of a site, note that many URLs need to be removed when they become invalid

WEB INSPECTOR - input a URL, this tool does testing to check for scripts, blacklisting, AV scanning, crawling, etc. This sort of tool is very helpful for doing security work when you don't have your own environment to work in

VIRUS TOTAL - Some tools will send a site/file to virus total to get a score. This tool can also get a hash of a file.

LINUX JWHOIS - analyze a new domain to see if the registration data is suspect.



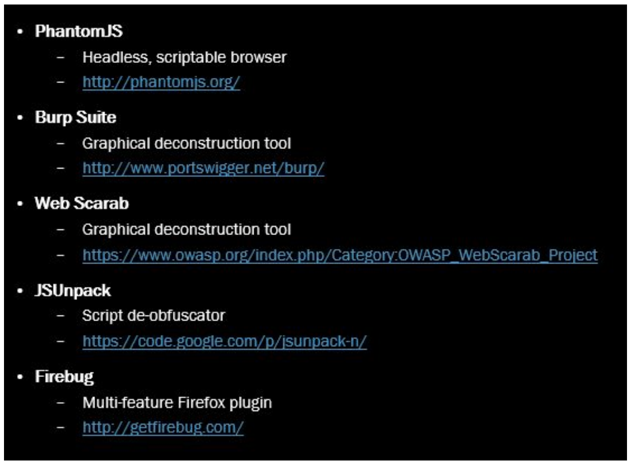
jwalton@localhost:~/devroot/gti/cdi/calpoly/src/main,

```
[jwalton@localhost resources]$ whois paypal-geldsparen.com | head -22
[Querying whois.verisign-grs.com]
[Redirected to whois.bizcn.com]
[Querying whois.bizcn.com]
[whois.bizcn.com]
Domain name: paypal-geldsparen.com
Registry Domain ID:

Registrar WHOIS Server: whois.bizcn.com
Registrar URL: http://www.bizcn.com
Updated Date: 2014-04-12T09:37:50Z
Creation Date: 2014-04-12T09:37:53Z
Registrar Registration Expiration Date: 2015-04-12T09:37:53Z
Registrar: Bizcn.com,Inc.
Registrar IANA ID: 471
Registrar Abuse Contact Email: abuse@bizcn.com
Registrar Abuse Contact Phone: +86.5922577888
Reseller:
Domain Status: clientDeleteProhibited
Domain Status: clientTransferProhibited
Registry Registrant ID:
Registrant Name: Whois Agent
Registrant Organization: Whois Privacy Protection Service
[jwalton@localhost resources]$
```

LINUX DIG - when DNS is set up, certain signs or patterns during set up can indicate potential nefarious intentions

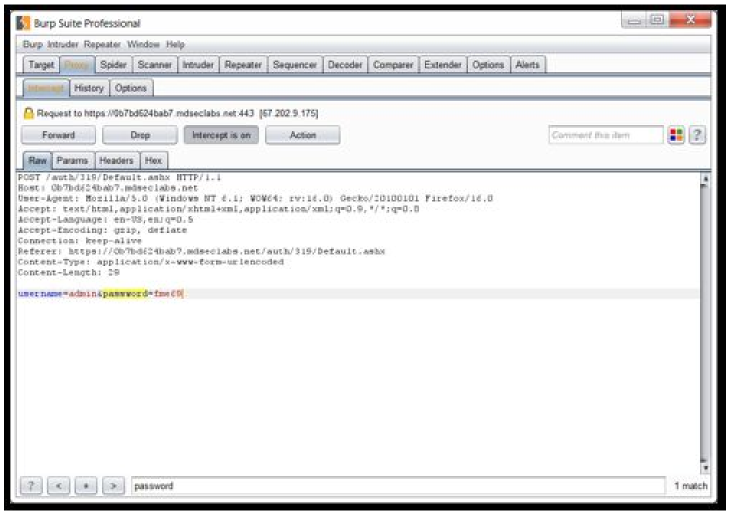
INDICATORS OF COMPROMISE - every time something happens, there are a series of forensic clues left behind. Hunting for patterns can reveal malware. many websites have IOC feeds

Research Tools -- these are good tools even if you don't follow cyber security. They allow for thorough testing and fuzzing of web development products.



- **PhantomJS**
  - Headless, scriptable browser
  - http://phantomjs.org/
- **Burp Suite**
  - Graphical deconstruction tool
  - http://www.portswigger.net/burp/
- **Web Scarab**
  - Graphical deconstruction tool
  - https://www.owasp.org/index.php/Category:OWASP_WebScarab_Project
- **JSUnpack**
  - Script de-obfuscator
  - https://code.google.com/p/jsunpack-n/
- **Firebug**
  - Multi-feature Firefox plugin
  - http://getfirebug.com/

- PhantomJS - A webkit completely replaces a browser
    - there are not enough humans in cyber security to look at all suspect URLs
- JSUNPACK
    - easy way to understand scripts
    - more static (faster) than PhantomJS
- BURP SUITE
    - can be used as a proxy
    - spidering - take one URL and follow the links through the web


Burp Suite Professional
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Options Alerts
Intercept History Options
Request to https://0b7bd624bab7.mdseclabs.net:443 [67.202.9.175]
Forward Drop Intercept is on Action
Comment this item
Raw Params Headers Hex
password
1 match

```
POST /auth/319/Default.ashx HTTP/1.1
Host: 0b7bd624bab7.mdseclabs.net
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:16.0) Gecko/20100101 Firefox/16.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: https://0b7bd624bab7.mdseclabs.net/auth/319/Default.ashx
Content-Type: application/x-www-form-urlencoded
Content-Length: 29

username=admin&password=fme69
```

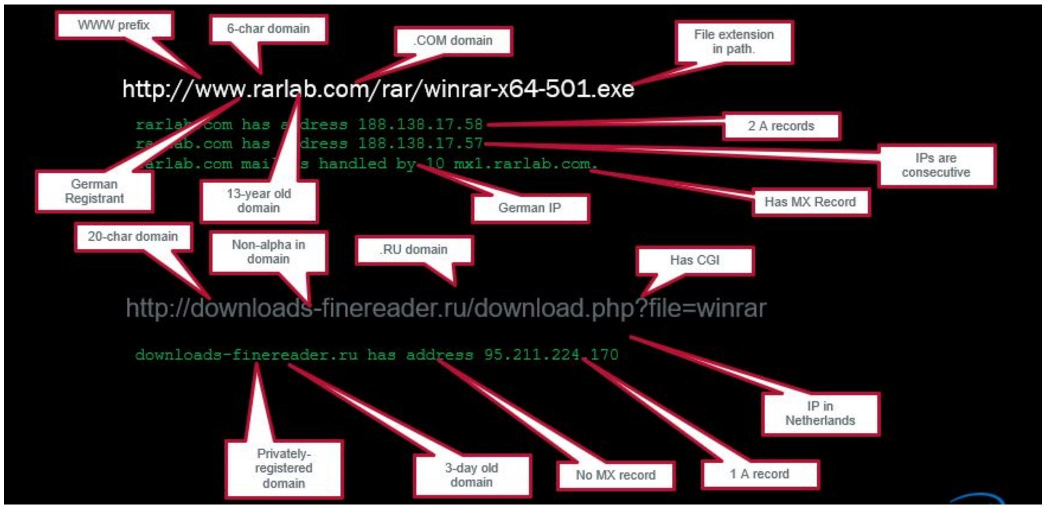
- WEBSCARAB
    - Similar to burp suite
- FIREBUG
    - Similar to Tamper Data
    - plugin extension

Classifying URLs

Content Agnostic - don't even look at the content, just the URL itself (IP, what does it look like?)

- Manual
    - a person goes through the URL content
    - record all actions
    - use a squid proxy or fiddler
    - used for top threats
- Static
    - get the page and scan it
    - usually hope to find an exploit
    - quick, used to get through millions of instances in a day
- Low-interaction
    - humans don't interact with it

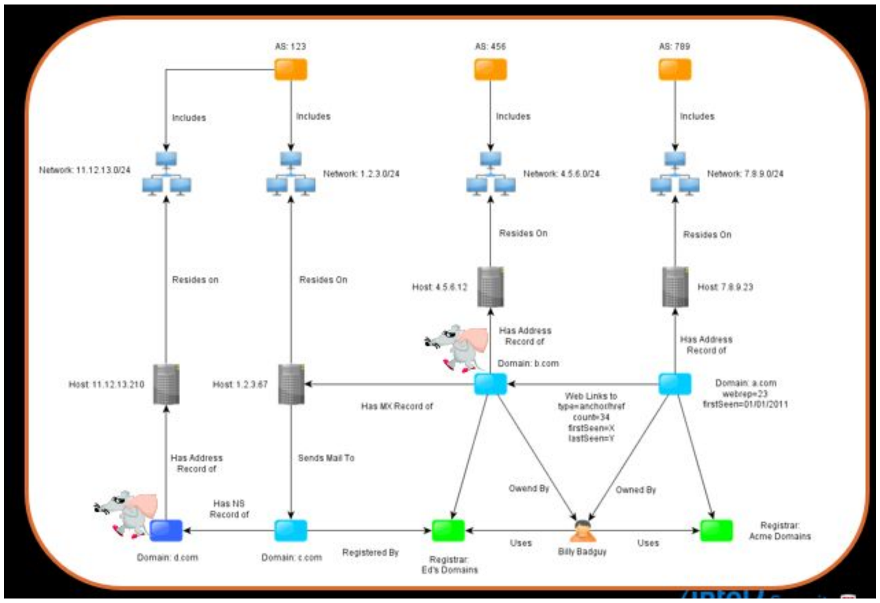
    - only the webkit
- High-interaction
    - bad guys can detect webkit and ‘sleep’
    - malicious designers can delay scripts or attacks knowing that a detection method will look for malware in the first few seconds while an actual user will stay on the page for much longer



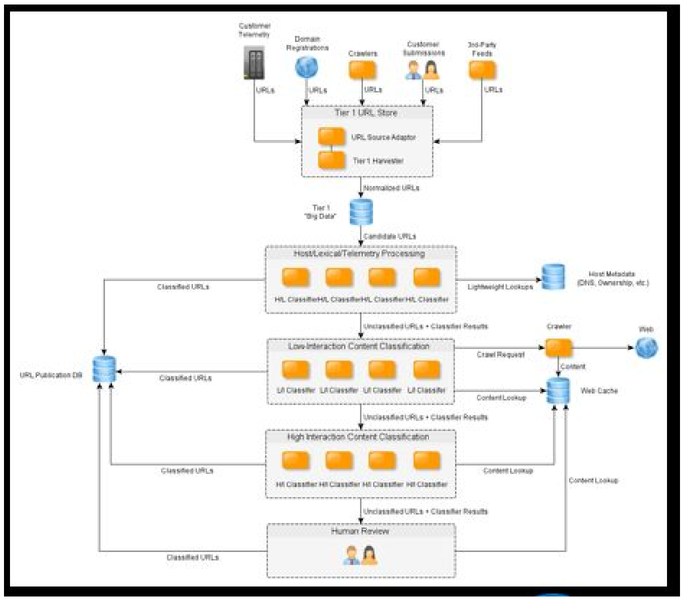
```
http://www.rarlab.com/rar/winrar-x64-501.exe
rarlab.com has address 188.138.17.58
rarlab.com has address 188.138.17.57
rarlab.com mail is handled by 10 mx1.rarlab.com.

http://downloads-finereader.ru/download.php?file=winrar
downloads-finereader.ru has address 95.211.224.170
```

- benevolent URL
    - a properly prefixed URL is less susceptible to be a malware URL
    - shorter URLs are more expensive
    - .com is generally safer than other registers
    - visible extensions are less likely to want to hide
    - german register with a german IP lends credibility
    - two consecutive IPs
    - has mail server
- malicious URL
    - long URL
    - non alpha-numerical characters
    - russian
    - privately registered and three days old
    - only one entry, and the IP doesn’t match the registry

Use an expert system? Graph based classification:



Broad-based analytics helps with the issue that defenders with a relatively small amount of resources are required to deal with millions of threats. “guilt by association”


Customer Telemetry
Domain Registrations
Crawlers
Customer Submissions
3rd-Party Feeds
URLs
URLs
URLs
URLs
URLs
Tier 1 URL Store
URL Source Adaptor
Tier 1 Harvester
Normalized URLs
Tier 1 "Big Data"
Candidate URLs
Host/Lexical/Telemetry Processing
H/L Classifier H/L Classifier H/L Classifier H/L Classifier
Classified URLs
Lightweight Lookups
Host Metadata (DNS, Ownership, etc.)
Unclassified URLs + Classifier Results
Crawler
Web
Low-Interaction Content Classification
Crawl Request
Content
URL Publication DB
Classified URLs
LI Classifier LI Classifier LI Classifier LI Classifier
Content Lookup
Web Cache
Unclassified URLs + Classifier Results
High Interaction Content Classification
Classified URLs
HI Classifier HI Classifier HI Classifier HI Classifier
Content Lookup
Content Lookup
Unclassified URLs + Classifier Results
Human Review
Classified URLs

The expert system will do the bulk of the triage and bring the most likely threats to a human’s attention - then an expert can make the call to investigate further.