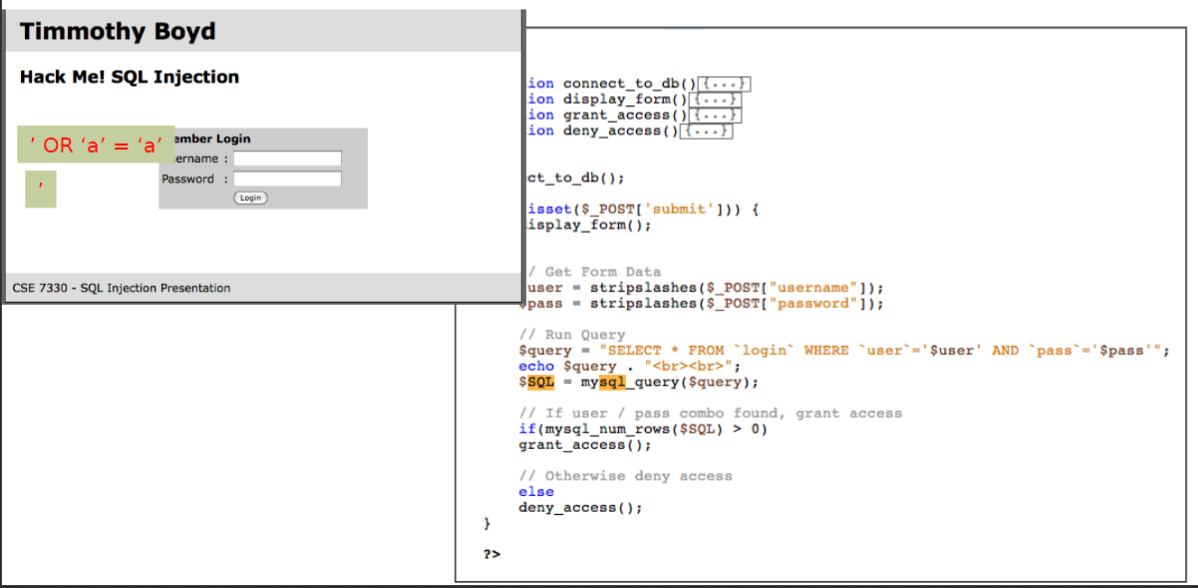
CEG 4430 Midterm Review

If IP packets with spoofed source IP addresses can be detected and discarded when they attempt to enter the Internet, then RST injection attack is impossible.

TRUE/FALSE?

**True**

Design one attack to gain access to the web service without knowing the username and password by exploiting the following PHP script.



**DROP TABLE ‘login’;--**

If a censor only sends out RST packets to clients, how can clients bypass RST injections?

**Don’t accept RST packets**

**Introduction:**

**Assets are accessed only by authorized parties.**

**- Often referred to as privacy or secrecy.**

**Assets are accessible to authorized parties at appropriate times.**

**Vulnerability**

**-** **A weakness in the security system that might be exploited to cause loss or harm.**

**Threat**

**- A set of circumstances that has the potential to cause loss or harm.**

**Attack**

**-** **The execution of a plan to carry out a threat by exploiting a vulnerability.**

**Network**

**- A network is a set of protocols, software, and hardware used to exchange information among different systems.**

**The Internet is a global-scale network**

**- Offers general-purpose service**

**- Has billions of users worldwide**

**- Consists of millions of "smaller" networks**

**- Supports numerous**

**types of services (Web, Email, VoIP...)**

**The Internet mainly uses the standard Internet protocol suite (a.k.a., TCP/IP).**

**The Internet Protocol (IP) address**

**- Logical address**

**- Manually or automatically assigned to a network interface controller (NIC)**

**- 32 bits for IPv4**

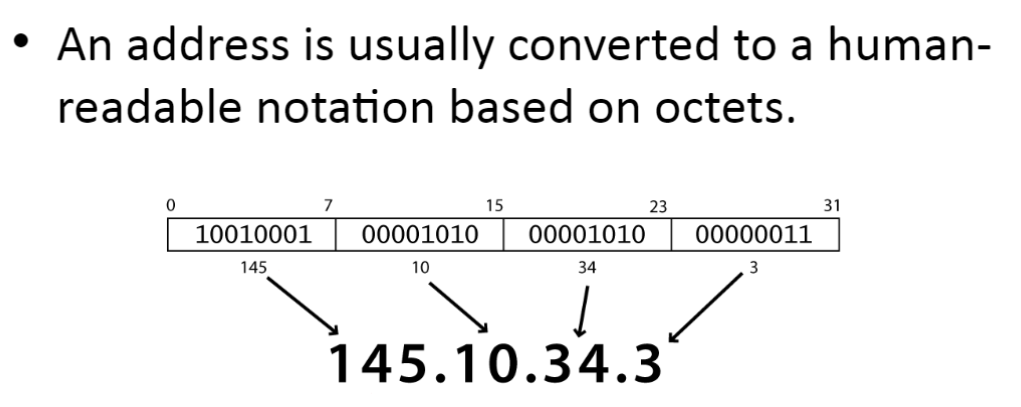
**Media Access Control (MAC)**

**address**

**- Physical address**

**- Assigned to a NIC by the manufacturer and are stored in hardware or firmware**

**- 48 bits**



**UDP offers best-effort, unreliable service — E.g., if a packet is dropped, UDP does not have mechanism to inform networked nodes.**

**TCP–Reliable–Relatively expensive**

**UDP–Best-effort–Inexpensive**

**Protocols that support functions based on the services provided by the transport layer**

**- DNS**

**- FTP -HTTP -IMAP -IRC**

**Tools**

**— TCPDUMP/WireShark**

**— Ngrep**

**Programming**

**— Writing your own packet sniffer**

**Libpcap**

**A collection of APIs for a program to capture packets from a network interface**

**Programming with Pcap**

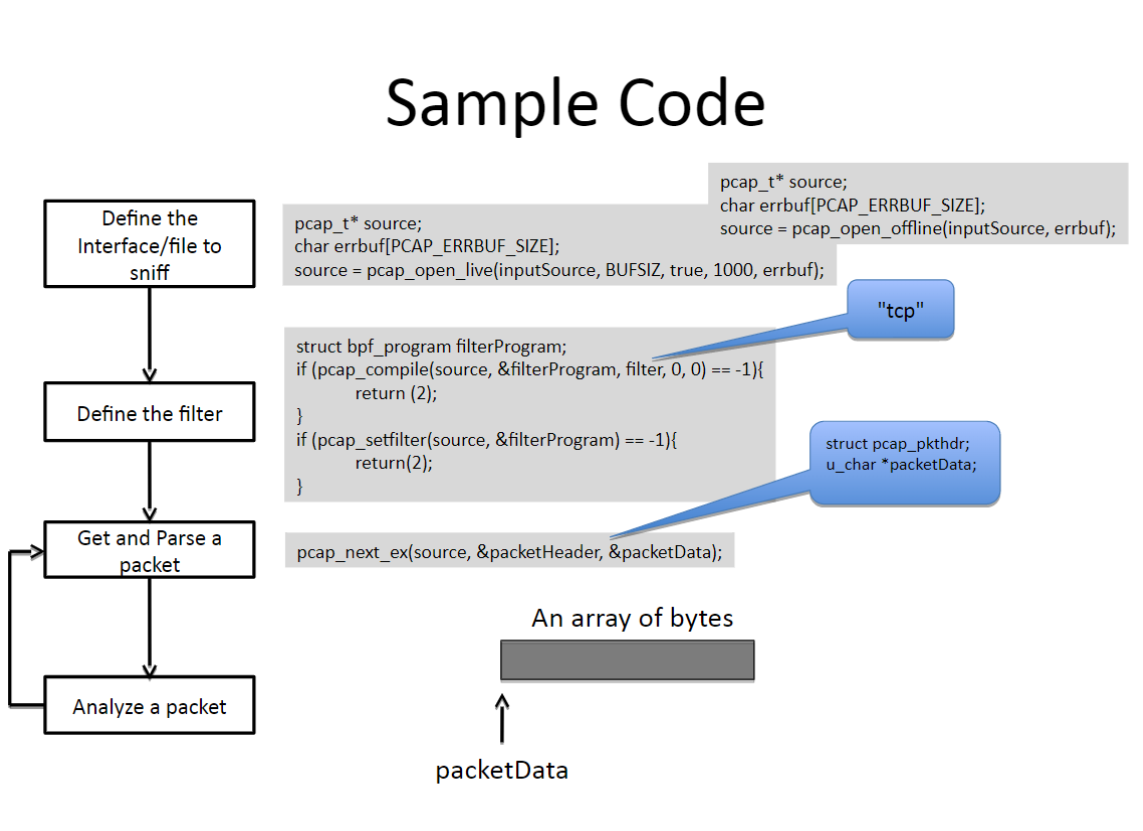
**Existing tools are excellent**

**But we need more flexibility to develop customized analysis capabilities**

**— Intrusion detection based on statistical analysis**

**- Rich semantics in traffic analysis**

**• A packet that has never been seen for the last 7 days**



**Protocol/Vulnerability**

**The Data Link Layer**

**- Above the physical-layer**

**— Ethernet is one of the most popular ways for the data link layer**

**Ethernet:**

**Two typical implementations**

**- 10Base2**

**• Each computer is attached to the cable using a T- Connector**

**— Ethernet hub**

**• Each computer is attached to the hub through an input/output port.**

**An Ethernet switch forwards a packet to the appropriate destination instead of broadcasting it.**

**What is ARP?**

**Host 1 wants to send a packet to host 3.**

**- It knows the IP address of host 3**

**- It does not know the MAC address of host 3**

**ARP Protocol**

**The Address Resolution Protocol (ARP)**

**— A link-layer protocol that provides services to the network layer**

**— Used to find a host's MAC address given its IP address**

**Solutions to ARP Spoofing**

**Checking for multiple occurrences of the MAC address**

**Static ARP tables**

**Maximum Transmission Unit**

**A network link is constrained by the size of the packet is can deliver Maximum Transmission Unit (MTU)**

**- The largest IP packet a link will accept**

**IP Fragmentation**

**If an IP packet is longer than the MTU of a link, the router breaks packet into smaller packets**

**— Each smaller packet is a fragment**

**— A fragment is still in the format of an IP packet**

**Fragmentation/Defragmentation**

**Fragmentation**

**- The router usually performs the fragmentation**

**- The sender/host can deliberately perform fragmentation**

**Defragmentation**

**- The destination host usually defragments the fragmented packets**

**- Use a timer to handle any lost fragment (i.e., discard the whole packet)**

**IP Fragmentation Attack**

**Denial-of-Service (DoS) Attack**

**• An attack used to cause a machine/service unable to perform its essential functionality**

**IP-Fragmentation-based DoS**

**— 1st fragment: flag = 001, offset = 0;**

**— 2nd fragment: flag = 001, offset = 64800**

**Consequence**

**— the machine will allocate 64800 bytes memory, and hold it for some time (e.g., 15 to 255 seconds)**

**IP Spoofing**

**Sending an IP packet with forged source IP address**

**To impersonate another host with respect its IP address**

**Applications**

**- Denial of Service Attacks**

**— Access the IP-based authentication services such as the R services suite (rlogin, rsh, etc.)**

**SYN Flood Attack**

**Denial-of-Service (DoS) Attack**

**- An attack used to cause a machine/service u to perform its essential functionality**

**SYN flood**

**- A specific type of denial-of-service attack**

**- Targeting at the implementation of the TCP protocol**

**A TCP reset basically kills a TCP connection instantly.**

**— If the RST bit is set to 1, it indicates to the receiving computer that the computer should immediately stop using the TCP connection; it should not send any more packets using the connection's identifying numbers, called ports, and discard any further packets it receives with headers indicating they belong to that connection.**

**Discussion**

**Why does GFW send RST packets to both client and server?**

**— GFW can send RST to only client. It should be enough to terminate the connection.**

**Race Condition**

**The sender can send a few (decided by both congestion control and flow control algorithms) packets with out receiving acknowledgments.**

**-Seq number < Min(window-size, rwnd)**

**• Window-size for congestion control, decided by the sender of the packet**

**• rwnd for flow control, decider by the receiver of the packet.**

**Detection Rules**

**RST\_SEQ\_DATA:**

**- The RST packet is "out of sequence", with the receiver observing a sequence number less than the preceding data packet would suggest.**

**- What doesn't it happen for normal cases?**

**DATA\_SEQ\_RST:**

**- The receiver will see further data packets from the sender after it has already received the RST.**

**- What doesn't it happen for normal cases?**

**Detection Rules**

**RST\_SEQ\_CHANGE:**

**- Back-to-pack pairs of RSTs in which the second RST has a sequence number higher than the first, and that exceeds the current maximum sequence number.**

**— What doesn't it happen for normal cases?**

**Detection Rules**

**RST\_ACK\_CHANGE:**

**- Nonsensical ACK numbers**

**SYN\_RST**

**- SYNs immediately followed by RST**

**SYN\_ACK\_RST**

**- SYN/ACKs immediately followed by RST**

**Domain Name System (DNS)**

**Indispensable component for the Internet**

**-** [**www.google.com**](http://www.google.com) **-> 173.194.75.106**

**Used by a huge percentage of Internet applications**

**- Browsers -FTP**

**- Instant Messengers**

**Honey-Query**

**What do you expect after you send out the honey-query?**

**- Not censored?**

**• No DNS response at all.**

**— Censored?**

**• A DNS response**

**• Is the IP address in the response trustable?**

**Do not forget**

**All honey queries are sent from real IP addresses controlled by authors**

**These IP addresses are unlikely to reside in the censored networks**

**Censors should actually not censor these IP addresses**

**But they inject fake DNS responses anyway - "the DNS injector does not consider packet origin when injecting packets".**

**Application Vulnerabilities**

**Two General Approaches**

**Identify the credential to break in**

**- Weak password**

**- Brute force**

**- Social engineering**

**Compromise the program vulnerabilities**

**- Incomplete mediation**

**- Timing attacks**

**- Buffer overflow**

**Buffer Overflow**

**A Buffer Overflow is an anomaly where data is stored beyond the boundary of a fixed-length buffer.**

**Program V.S. Process**

**Program**

**— A file**

**• Static instructions**

**• Static data**

**Process**

**- An instance of a program**

**• Loaded into the memory**

**• Executing and dynamic**

**HTTP Response**

**Data**

**-HTML**

**Executable Scripts**

**— Javascript**

**- VBScript**

**How Does a Server Generate Responses**

**Static Data**

**- E.g., an pure HTML file (with javascript)**

**Scripts: dynamically generate content**

**— PHP -ASP.NET -Java**

**SQL Injection Attack**

**Exploit a security vulnerability in an SQL application (e.g., web application)**

**- A specific case of "incomplete mediation"**

**Considered as one of the top 10 web application vulnerabilities of 07 and 10**

**A large number of incidents**

**- 2014: Biomedical Engineering Servers, Johns Hopkins University**

**— 2013: 71 Chinese government databases are compromised using SQL injection**

**Same-Origin Policy**

**One webpage can only read properties of another webpage if they share the same server, protocol, and port**

**If the same server hosts unrelated sites, scripts from one site can access document properties on the other**

**Two Types of Attacks**

**Persistent XXS Attacks**

**- Saved persistently in the database**

**- Damage massive users**

**Non-Persistent XXS Attacks**

**- Targeted victims**

**TaintTracker**

**Tracks each instruction that manipulates data in order to determine whether the result is tainted.**

**— Data movement instructions**

**• LOAD, STORE, MOVE, PUSH, POP, and etc.**

**• Rule: if any byte of data at the source location is tainted, the data at the destination will be tainted**

**- Arithmetic instructions**

**• ADD, SUB, XOR, and etc.**

**• Rule: the result will be tainted if any byte of the operands is tainted**

**Randomization**

**Identify keywords in an SQL statement**

**Rewrite all keywords with the random key appended.**

**De-randomization**

**Identify terms/keywords (e.g., starting from the original keywords)**

**— E.g., selectl23 or select456 or select Evaluate the format**

**— Keyword + random number**

**— Detect mal-formed keywords Stripping away the random number Send it to the DBMS**

**Parse Tree Validation**

**Objective**

**- Detect SQL injection attacks at runtime.**

**Observation**

**- All SQL injections alter the structure of the query intended by the programmer.**

**Web Security**

**Complexity**

**- Protocols**

**• Client**

**• Server**

**- Browsers**

**• Data**

**• Scripts**

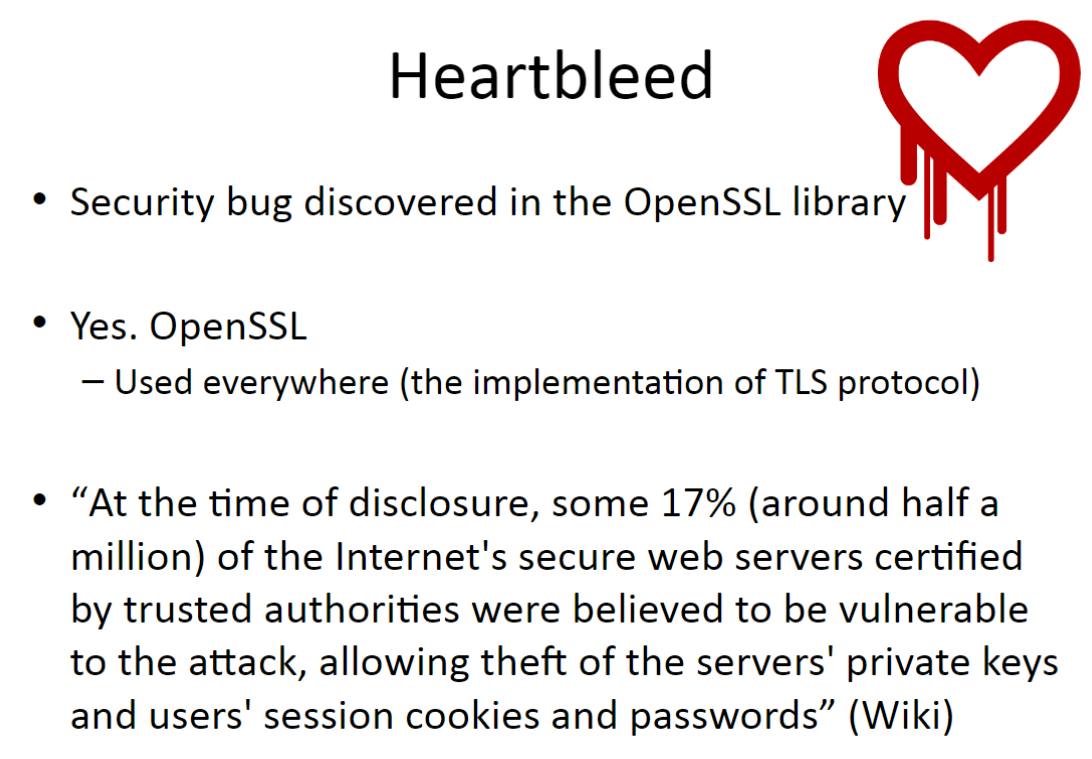
**- Servers**

**• Data**

**• Scripts**

**Example**

**Each file on the web server (under the web directory) can be invoked directly from the network by giving its name as part of the URL - -> Each file represents an unintended entry point**



**Detection**

**Signature-based detection**

**-"DROP Table"**

**-"Insert into"**

**Anomaly Detection**

**- Query syntax analysis**

**• Injected content change the parse tree of a query**

**Prevention**

**Mediate Input**

**- Use pattern matching**

**Hide error messages**

**READ ONLY Database Access**

**Prevention**

**Whitelisting**

**- Define what should be accepted**

**Blacklisting (NO)**

**— It can never be complete**

**Malicious Applications**

**Malware**

**Insider Attacks**

**- Backdoors/trapdoors**

**- Logic Bombs**

**Virus**

**- Polymorphic and Metamorphic Virus Trojan Horses Privacy-Invasive Software**

**- Adware/Spyware**

**Worm**

**Botnet**

**Backdoor/Trapdoor**

**Secret entry point into a system**

**- Specific user identifier or password that circumvents normal security procedures**

**Commonly used by developers**

**— Could be included in a compiler**

**Logic Bomb**

**A logic bomb is a class of malicious code that activated when a specific condition occurs.**

**— Certain file is created**

**— Certain time is reached**

**— Certain user logins into the system**

**A logic bomb could represent**

**— an "insider" threat**

**- inherently malicious code**

**Virus**

**Self-replicating code**

**- Malicious functions + self-replicating (with users' involvement)**

**- "Normal Code" => "Normal Code + Malicious Code" Stealthy**

**- Attempts to evade detection Operates when infected code executed**

**- Propagates**

**- Performs malicious actions**

**- Redirect to the normal code**

**Virus Infection Vectors**

**Boot Sector**

**- Loaded when the system is booted**

**Executable**

**- Activated when an executable file is activated**

**Macro files**

**- Triggered when a document is loaded**

**Virus Characteristics**

**Terminate and Stay Resident**

**- Stays active in memory after application completes**

**- Allow infection of previously unknown files**

**• Traps calls that execute a program**

**Stealthy**

**- Conceal Infection**

**• Trap read and disinfect**

**• Let executable call infected file**

**- Polymorphic/Metamorphic Virus**

**• Change virus code to circumvent misuse detection**

**Trojan Horse**

**A Trojan Horse is a malicious program that is disguised as legitimate software**

**The gift horse left outside the gate of Troy by the Greeks**

**- Appear to be interesting and innocent**

**- Actually harmful**

**Privacy-Invasive Software**

**Malware with specific malicious objectives**

**- Adware**

**• Pop up advertisements**

**- Spyware**

**• Key logging, screen capturing**

**Worms**

**Definition: Programs that self-propagate across the internet by exploiting security flaws in widely used services**

**Virus v.s. Worms**

**rus**

**A malicious program Propagates depending on the user intervention**

**• File sharing**

**• Execute an infected file**

**• Worm**

**- A malicious program**

**- Propagates automatously**

**• Self-replicate**

**• No user intervention is required**

**A Brief History of Worms**

**Worms that affect the operation of the whole Internet**

**- Morris Worm (1988)**

**- Code Red (2001)**

**- Nimda (2001)**

**- Blaster (2003)**

**- SQL Slammer (2003)**

**Morris Worm**

**The positive impacts**

**- Computer Emergency Response Team (CERT)**

**• Reacted to the damage and disruption caused by Morris worm**

**• Becomes a leading center on information sharing with respect to software vulnerability and malware**

**— Raise attentions to cyber-security**

**The Code Red Worm**

**Rapid propagation**

**- > 2,000 hosts/min**

**- Code Red II Took about 14 hours to fully propagate**

**359,000 hosts are infected**

**Exploits**

**- BOF in Microsoft IIS web server (enabled by default)**

**The Nimda Worm**

**Rapid Propagation**

**- Became the Internet's most widespread worm within 22 minutes**

**Exploits**

**- MS IIS vulnerability (CVE-2000-0884)**

**- Email itself as attachment based on email addresses harvested from the infected machine**

**- Copy itself across open network shares**

**- Add exploit code to web pages on compromised servers in order to infect clients that browse the pages**

**- Leverage the backdoors left behind by Code Red II.**

**Blaster Worm**

**Exploits**

**- BOFin MS OS**

**- The patch had been released one month earlier than the incident**

**• "The original Blaster was created after a Chinese cracking collective called Xfocus reverse engineered the original Microsoft patch that allowed for execution of the attack" from wiki**

**Attack**

**- Launched DDoS attacks against windowsupdate.com**

**The Code Red I**

**Its Scanning Strategy**

**— Launch 99 threads and each thread generated random IP addresses in the whole IP space (2A32)**

**— Probe each IP to determine whether the vulnerable service is available on the host.**

**- If so, infect the host.**

**Hit-list scanning**

**How to collect hit-list in practice?**

**— Stealthy scans**

**— Distributed scanning**

**— DNS searches**

**— Spiders**

**• 33% of automated search engine queries are looking for vulnerable Internet services.**

**— Public surveys -Just listen**

**Permutation Scanning**

**Random scanning**

**- The Code Red I is a salient example to use random scanning**

**— Disadvantage: many addresses are probed multiple times**

**Permutation Scanning**

**- Objective: provide a self-coordinated, comprehensive scan while maintaining the benefits of random probing**

**— Assumption: a worm can detect that a particular target is already infected**

**Permutation Scanning**

**All worms share a common pseudo random permutation of the IP address space**

**Any machine infected during the hit-list phase (or local subnet scanning) starts scanning after its point in the permutation**

**Whenever the worm sees an already infected machine, it chooses a new, random start point and proceeds from there**

**Worms infected by permutation scanning would start at a random point**