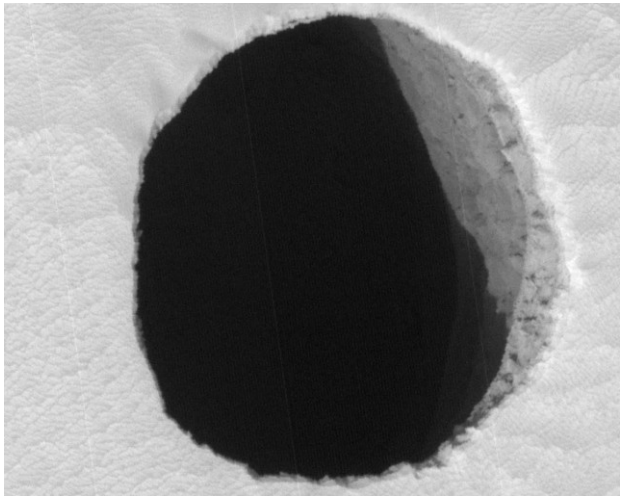
CSCD 303
Essential Computer Security
Fall 2017



Security Hole

Lecture 8 -

Vulnerabilities, Scanning and Vulnerability Data Bases

Reading: References at end of Slides, Chap. 5, CompTIA text

Overview

- Learning Objectives
  - Introduce OS Vulnerabilities
    - What are they
    - Why do they happen
  - Introduce Vulnerability Databases
    - Vulnerability Analysis

Overview Vulnerabilities

- Looked at designing for security in operating systems
    - Turns out if security is designed in from the beginning, likely system is more secure
- Yet, systems will still have problems even even when security has been carefully considered
- Look at OS vulnerabilities, classification and tools for discovery

Desktop Vulnerabilities

- All OS platforms have vulnerabilities
  - Windows, Linux and yes, MAC too!
  - OS drivers and utilities have vulnerabilities
  - Applications that run on OS platforms have vulnerabilities
  - Browsers and Web platforms have vulnerabilities
    - These “holes” into your network and systems are beyond network protocol vulnerabilities
  - Vulnerabilities include users
    - As many users as you have, each one is a walking vulnerability

Vulnerabilities Defined

or

According to Merriam-Webster, Vulnerable Defined

"exposed to possibility of being attacked or harmed, either physically or emotionally: 'we were in a vulnerable position'."

In Computer Security, Vulnerability Defined

Security Vulnerability refers to system flaw that can leave it open to attack

A vulnerability may also refer to any type of weakness in a

1. Computer system itself,
2. Set of procedures, or
3. Anything that leaves information security exposed to a threat

OS Vulnerabilities

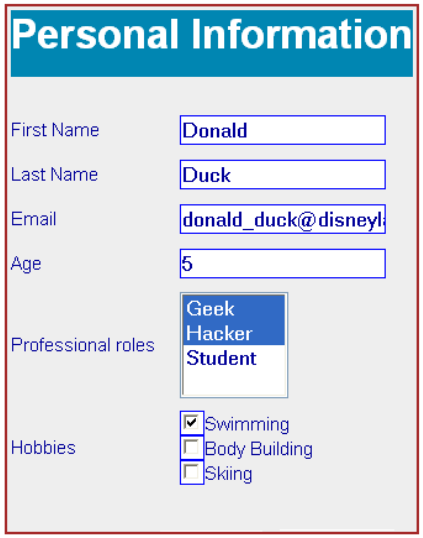
- What are some vulnerabilities common to all OS's?

OS Vulnerabilities

Look Common OS Vulnerabilities

1. Buffer Overflow
2. Unvalidated input
3. Race conditions
4. Access-control problems
5. Weaknesses in authentication

Buffer Overflow



- Every program that allows input
  - Needs to store input in memory

    until it can be used for its intended purpose

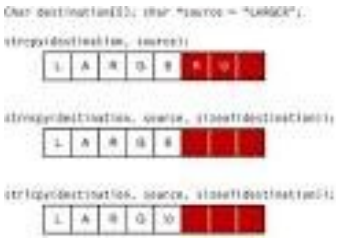
  - Examples: Web form, enter your name

    Saving a file, enter file name,

    Search engine, enter search string

What is the definition of a buffer?

Buffer Defined



- A temporary storage area, usually in RAM
- Purpose is to act as holding area, enabling CPU to manipulate data before transferring it to a device

- Process of reading and writing data to a disk are slow, programs keep track of changes in a buffer, then copy buffer to a disk
- For example, word processors employ a buffer to keep track of changes to files
- Java has concept of buffered reads and writes
  - Only writes or reads from disk in batches

Buffer Overflow

- Program should check user input to make sure it's correct length
  - Programmer does not bother to check length of input Programmer assumes user will not do anything unreasonable … False!
  - Language allows him/her to overwrite buffer, C or C++
  - For example
    - Form asks you to enter your first name
      Has room for 12 characters

      First Name

      [ ]
    - User's first name is really long, 15 characters

      Francessca-Ally

F r a n c e s s c a - A | lly ← Overflow Chars

Buffer Overflows

- How are buffer overflows used to compromise your computer?

See Diagram

  - As part of long data input, attacker will include some of his own code
  - Then, manipulates flow of program in memory to execute his code ...more on this later
  - If process is running with administrator privileges, attacker code has administrator privileges
  - Then, computer is toast !!!

Microsoft Vulnerabilities

- Does anyone know about the vulnerability described in

  Microsoft Security Bulletin MS08-067 ?

Buffer Overflow MS08-067

- Buffer overflow vulnerability in Windows Server Service
  - For systems running Windows 2000, XP, Windows 7 and Server 2003, remote, unauthenticated attacker could exploit this vulnerability
    - In Vista, attacker would need to be authenticated
  - Since Server service runs with **Administrator** privileges, an attacker could take complete control of a vulnerable system
  - This IS the vulnerability that **Conficker worm** exploited!

Details of MS08-067

- Specifically, this vulnerability is a buffer overflow in an unauthenticated Windows SMB file sharing session
  - **SMB = Server Message Block,** protocol for sharing server resources like files and printers
- Malicious client can bind to this service and issue a request with an overly long argument
  - Overflowing a buffer and possibly executing arbitrary code on the vulnerable server
- This is one way malware is getting onto systems

http://asert.arbornetworks.com/2008/10/ms08-067-server-service-vulnerabilities-redux-and-wormability/

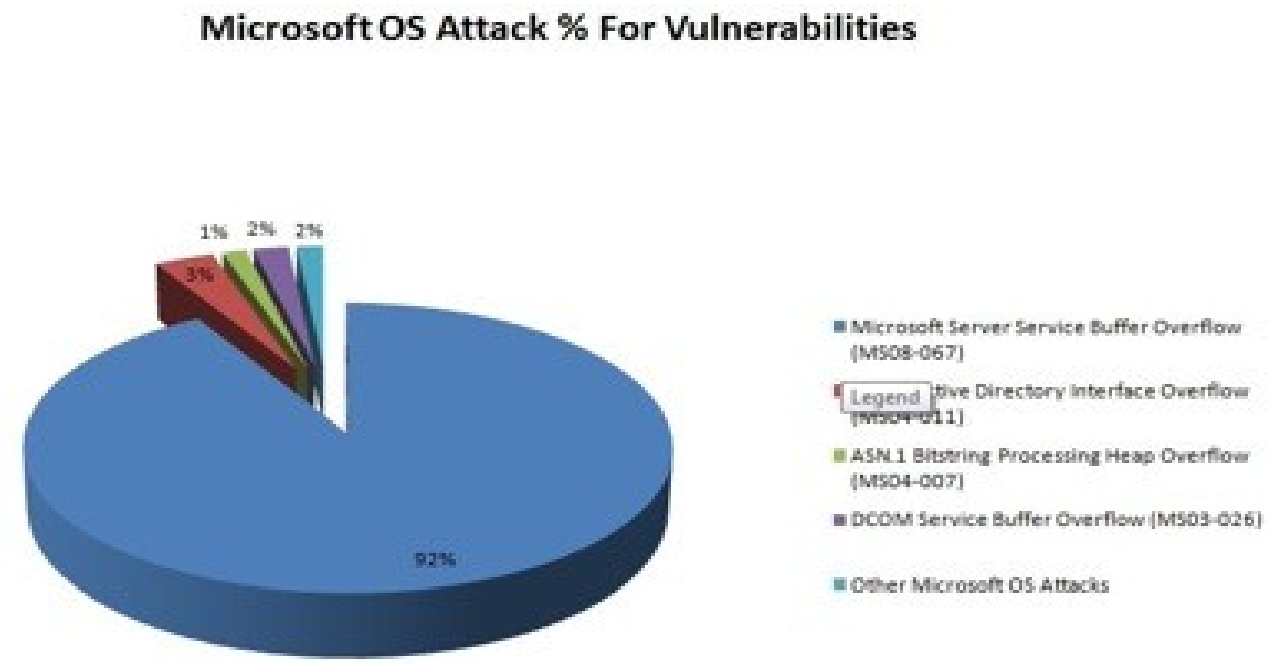
What is the Server Message Block?

- Operates as an **application-layer network protocol**
  - Provides shared access to files, printers, serial ports
  - Also provides an authenticated inter-process communication mechanism
  - Shortcut in the MS world to client/server applications !!!

Microsoft Vulnerabilities

http://www.sans.org/top-cyber-security-risks/#trends

- September 2009
- Over 90% of the attacks recorded for Microsoft targeted the buffer overflow vulnerability described in the Microsoft Security Bulletin MS08-067



Linux Buffer Overflow Vulnerabilities

- Is Linux or Mac OS X immune to buffer overflows?
    - No. They have these too …
- Google search of “buffer overflow vulnerabilities in linux”
    - Came back with 717,000 hits
    - Among the problems
        - Stack based X-Windows vulnerability
            - Affects all linux distributions
        - Adobe flash player – Linux
        - Critical flaw in glibc, GNU C Libraray

Unvalidated Input Attacks

- Any input received by a program from an untrusted source is a potential target for attack
  - Hackers look at every source of input
  - Try to inject their own code or script to be run by the system accepting the input
  - Use automated fuzzing tools …
    - Jams every type of input into forms, SQL queries, or login pages
  - May allow them unauthorized access

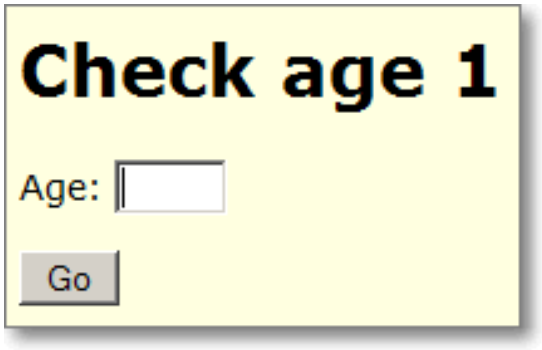
Validating Input

- Input needs to meet programmer expectations

  For whatever input required:

  - HTML, email, userid or valid database request
- Compare input to what is known to be acceptable

  Use regular expressions, patterns of characters describe allowable input
- Bad input is either rejected or altered

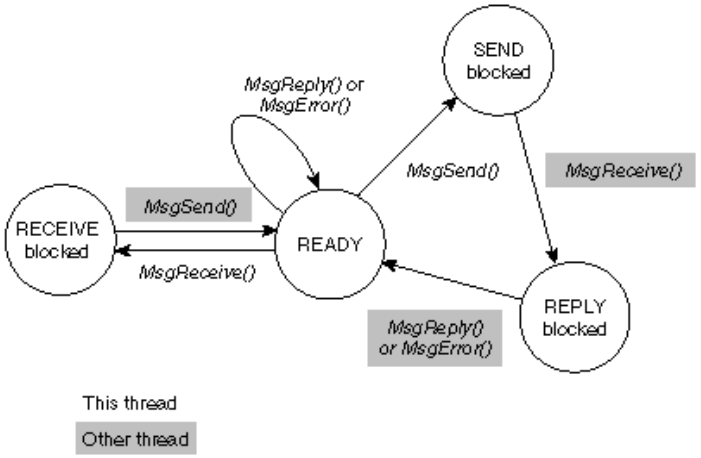


Race Condition

- A race condition exists when two events can occur out of sequence … unexpected
  - If correct sequence is required for proper functioning of program, potential vulnerability can be exploited
  - If attacker can cause correct sequence not to happen and insert malicious code, change a filename, or otherwise interfere with normal operation
  - Race condition is a security vulnerability
- Attackers can sometimes take advantage of small time gaps in processing of code
  - Interfere with sequence of operations
  - Which they then exploit

Race Conditions

- There are two basic types of race condition that can be exploited
  1. Time of check/time of use
  2. Interprocess communication



Race Condition: Time of Check/Time of Use

- Application checks some condition before undertaking an action
- Example, Check if file exists before writing to it

  - **Attacker**, by continuously running program that creates new temporary file can create file in gap between when application checked to make sure temporary file didn't exist and when it opens it for writing
  - Application opens attacker's file and writes to it ...

Race Condition: Interprocess Communication

A writes foo = 2
C writes foo = 3
B reads foo

- Separate processes—either within a single program or in two different programs—sometimes have to share information
  - For example, if two processes share same data,
    - Attacker alters data after one process sets it but before other reads it
  - Solution to this race condition, use some locking mechanism to prevent one process from changing a variable until another is finished

Access Control

- Many OS security vulnerabilities are created by careless or improper use of access controls, or by failure to use them at all
  - Exploits involve an attacker gaining more privileges than they should have
    - Privileges are access rights granted by the operating system
    - Controls who is allowed to read and write files, see directories, execute a programs, backup system etc.

Vulnerabilities, Finding them, Publishing Them

Vulnerabilities

- Who discovers them?

  Humans discover them,
  - Hacker groups
  - Security company or
  - “Researchers”

  – Discovers specific way to violate security of a software product
  – Discovery may be accidental or through directed research
  – Vulnerability, is then released to security community

Release of Vulnerabilities

- Both security researchers and hackers research and publish vulnerabilities
- Publishing vulnerabilities is controversial
- Question ....
- What are pros and cons of alerting the world to vulnerabilities?

Scanning

- Vulnerability Scanning
  - Can automate process of checking system for known vulnerabilities
    - Maybe hundreds of vulnerabilities in a given year, What are the chances they didn't all get patched?

Question

- What does an attacker do with identified vulnerabilities?

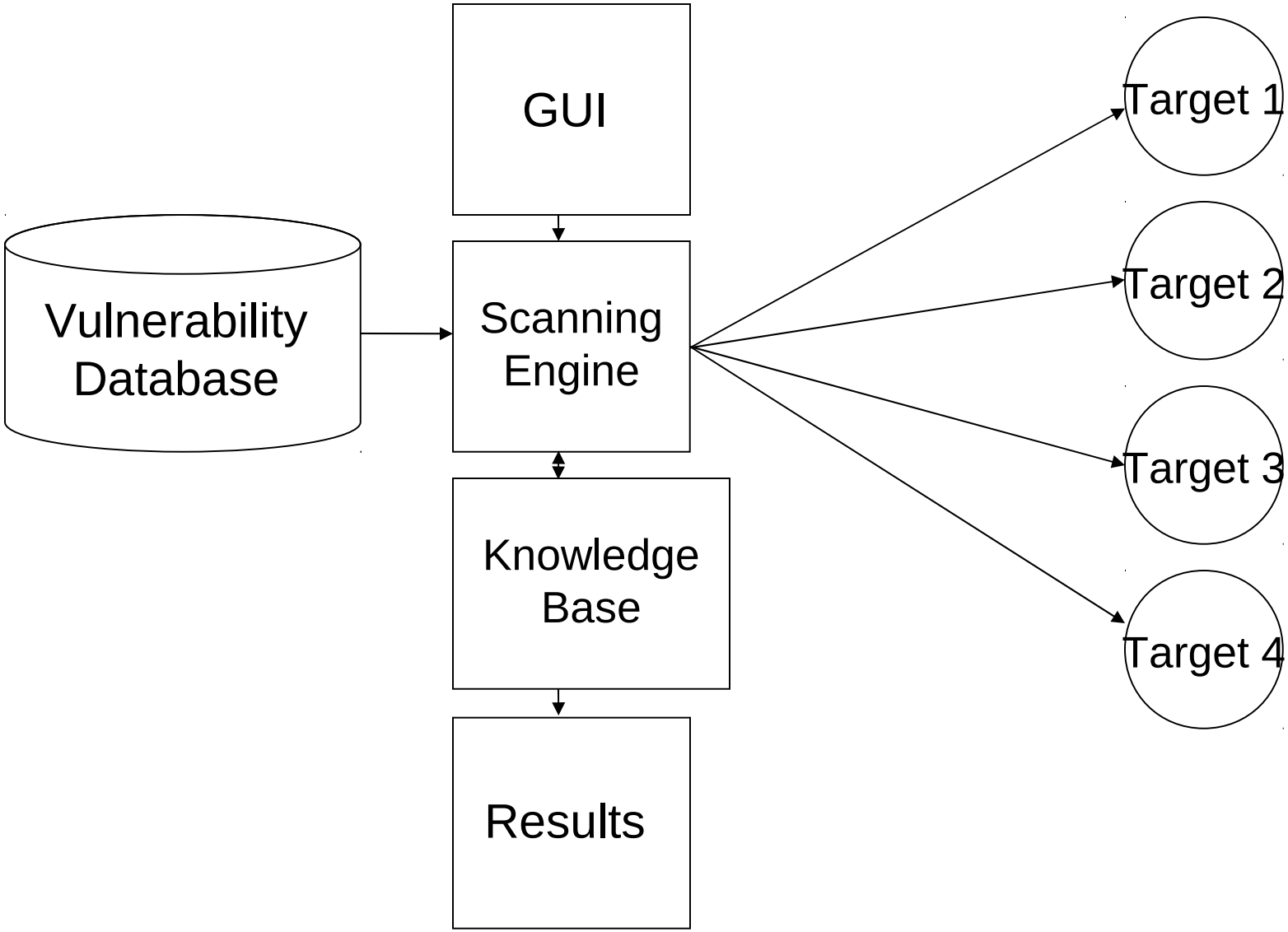
Vulnerability Scanners

- 1992 - First one
  - Internet Security Scanner (ISS)
- 1995
  - SATAN - Security Admin Tool for Analyzing Networks
  - Dan Farmer and Wietse Venema
  - Wider checks
- 1998
  - Nessus - Was Open Source, built on their ideas
  - Still one of most popular, home use still free
  - Now, charge for its use!
- 2008
  - OpenVAS was initially named GNessUs as a fork of the Nessus security scanner

Scanning

- Vulnerability Scanning
  - Looks for several types of vulnerabilities
    - Configuration errors
    - Default configuration weaknesses
    - Well-known system vulnerabilities
  - Number of scanners available
    - Some are free
    - Some cost a lot of money
    - Some of the most popular vulnerability scanners are free

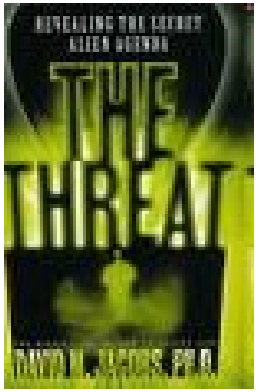
Vulnerability Scanning



Vulnerability Scanning

- Similar to virus scanning software:
  - Contain a database of vulnerability signatures that the tool searches for on a target system
  - Cannot find vulnerabilities not in the database
    - New vulnerabilities are discovered often
    - Vulnerability database must be updated regularly

Scanning



- Vulnerability Scanners

Retina http://www.eeye.com

IBM ISS Internet Scanner http://www.iss.net

Nessus http://www.nessus.org/

Open Vas http://www.openvas.org/

GFI LANguard Network Security Scanner
http://www.gfi.com/lannetscan

Scanning Nessus

- Nessus
  - Flexible – can write your own vulnerability checks
    - Called plugins, has own scripting language
    - Source code supplied
    - Lots of developers – to enhance functionality
    - Free for home use, corporate use - now costs money
  - Uses

    Common Vulnerabilities and Exposures database
  - Allows Nessus to cross reference with other tools that are CVE compliant

Scanning Nessus

- Nessus
  - Runs on Linux and Windows
  - Nessus doesn't use large Database of vulnerabilities that gets updated
  - It uses Nessus Attack Scripting Language (NASL)
  - Allows people to write their own scripts, plug-ins
    - Provides plug-in interface
  - Many free plug-ins are available from

    http://www.nessus.org/plugins/index.php?view=all

  Plug-ins specific to detecting a common virus or vulnerability

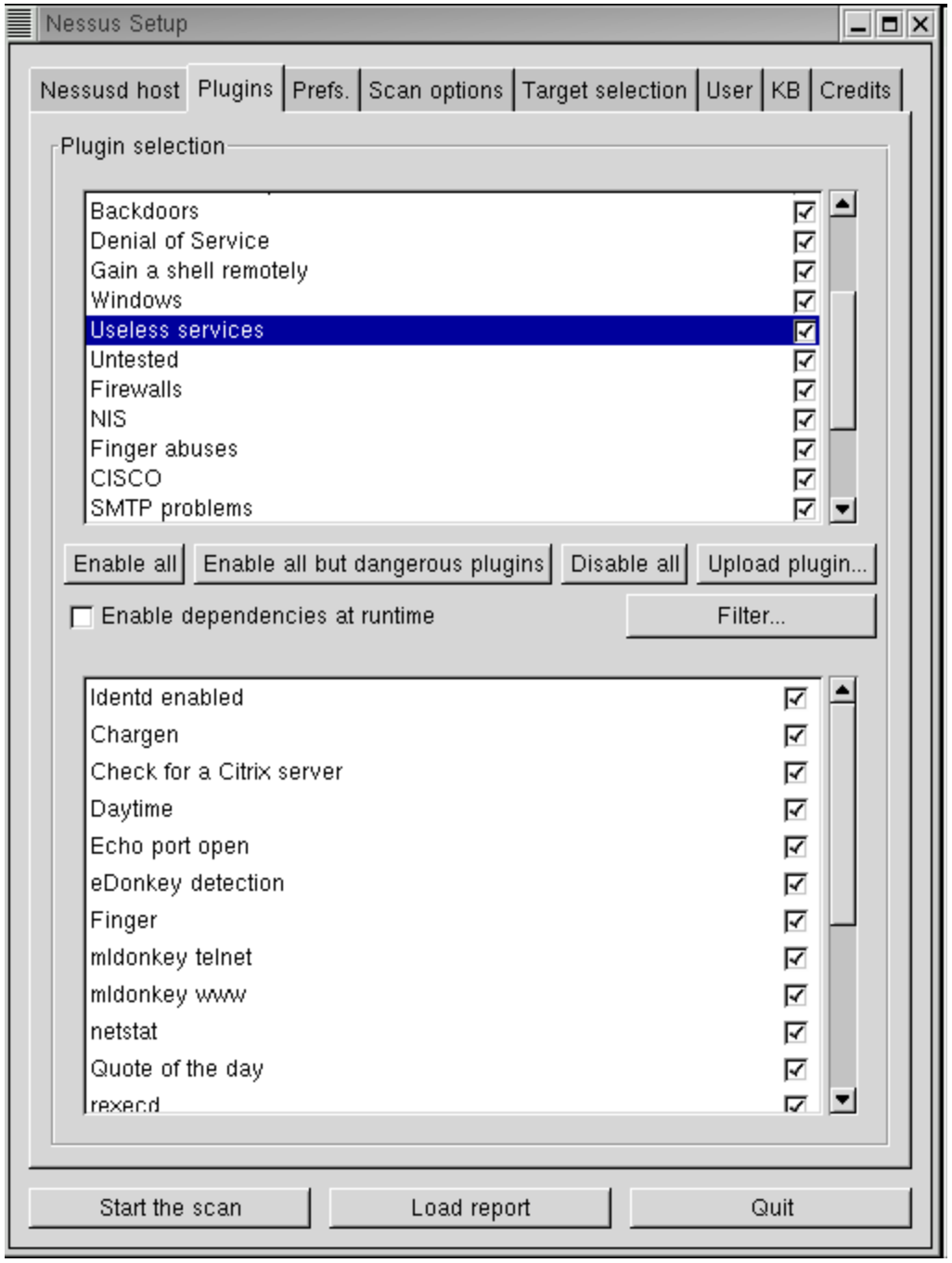
  Like a virus signature

Scanning With Nessus

- Nessus
  - What vulnerabilities can it discover?
    - **A few of the common ones include**
      - Finger – often misconfigured
      - Windows Vulnerabilities – many of them
      - CGI Problems –
        Scripts often have vulnerabilties
      - RPC – remote procedure call program
      - Firewalls – mis-configured
      - FTP – has had a lot of vulnerabilities
        » Looks for unpatched FTP implementation
      - Can just look at the plug-ins list for sample

Nessus

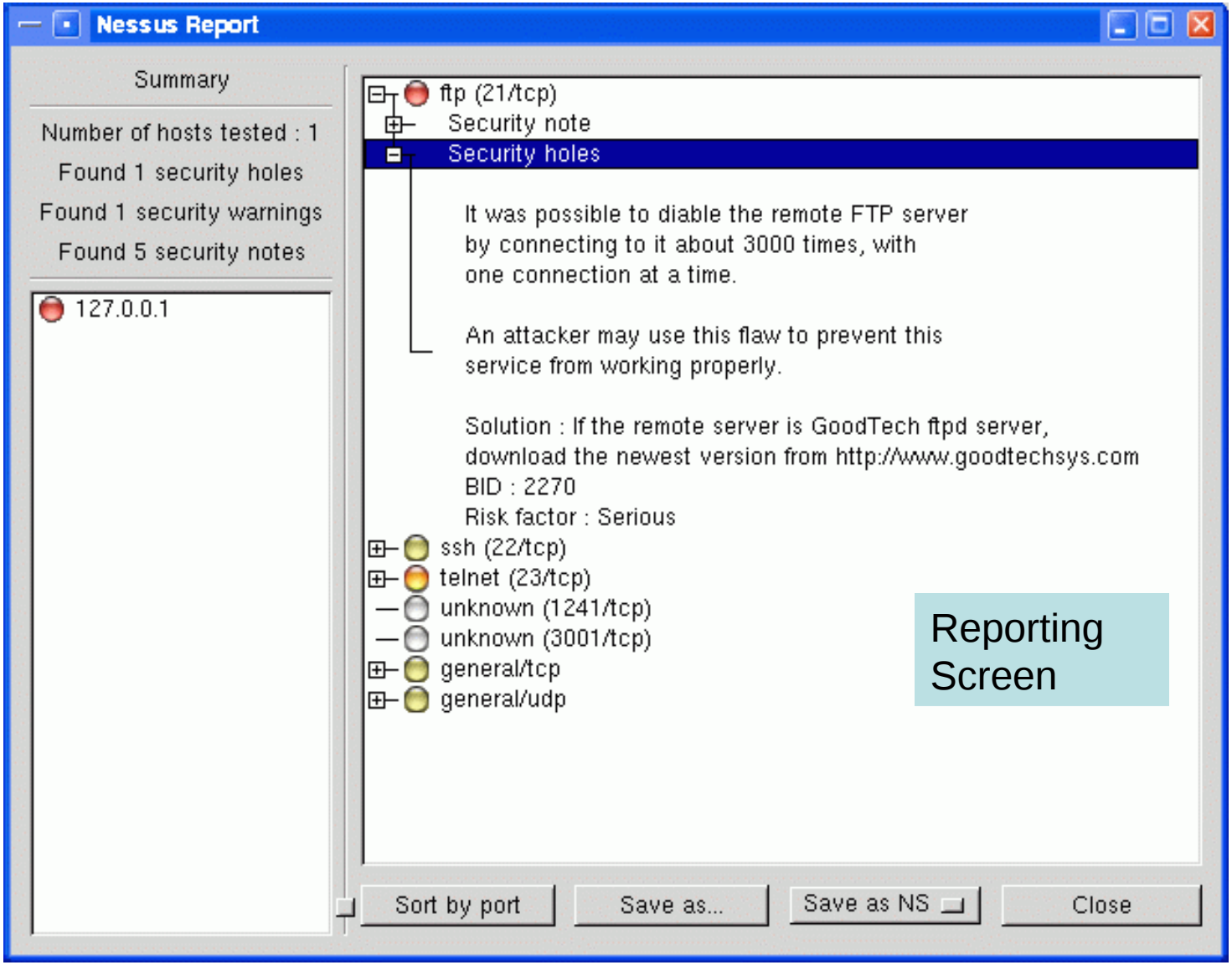
Configure with Respect to Plugin



Scanning With Nessus

- Nessus
  - Each vulnerability is ranked with respect to risk
    - Low, medium and high
    - Should interpret the risk results only in view of your own system
    - Same vulnerability may not be high risk for you
  - Recommendations are made for fixing vulnerability

Nessus Reports



OpenVAS vs. Nessus

- As Nessus became commercialized, OpenVAS became open source version
- OpenVAS was initially named GNessUs as a fork of the Nessus security scanner to allow future free development of the now-proprietary tool
- OpenVAS was originally proposed by pentesters at Portcullis Computer Security ... around 2005
- OpenVAS is actively being developed and supported

  http://www.openvas.org/

Vulnerability Databases and Information

Purposes of a VDB

Provide accurate information on security vulnerabilities

Provide historic reference on software bugs

Provide information on solutions

Provide innovations to help organizations deal with vulnerabilities

Major Players

Comprehensive VDBs

NatVulDataBase - http://nvd.nist.gov/
BID - http://www.securityfocus.com/bid
CVE - http://www.cve.mitre.org/
ISS X-Force - http://xforce.iss.net/
OSVDB – http://www.osvdb.org/
Secunia - http://www.secunia.com/
Security Tracker - http://www.securitytracker.com/

Vulnerability Notification Services

CERT - http://www.cert.org/
CIAC Advisory - http://www.ciac.org/ciac/index.html

Common Vulnerabilities and Exposures (CVE)

- A list of standardized names for vulnerabilities and other information security exposures (CVE)
  - **CVE** standardizes names for all publicly known vulnerabilities and security exposures and is a community wide effort
  - Content of CVE is collaborative effort of CVE Editorial Board
    - Includes representatives from over 20 security related organizations
    - Security tool vendors, academic institutions, and government
  - MITRE Corporation maintains CVE and moderates Editorial Board discussions.
    - CVE, http://cve.mitre.org

National Vulnerability Database

- NVD, comprehensive cyber security vulnerability database
  - Integrates all publicly available U.S. Government vulnerability resources and provides references to industry resources
  - Based on and synchronized with the CVE vulnerability naming standard
    - NVD is the CVE standard, augmented with additional analysis, a database, and a fine grained search engine. NVD is a superset of CVE
    - NVD is synchronized with CVE such that any updates to CVE appear immediately on NVD

http://nvd.nist.gov/

National Vulnerability Database

- National Vulnerability Database

  http://web.nvd.nist.gov/view/vuln/search?execution=e2s1

  How would you use this resource?

  You can search this database for all the vulnerabilities associated with a system

Common Vulnerabilities and Exposures

- Example CVE Entries
  - CVE-1999-0002 Buffer overflow in NFS mountd gives root access to remote attackers, mostly in Linux systems.
  - CVE-1999-0003 Execute commands as root via buffer overflow in Tooltalk database server (rpc.ttdbserverd)
  - CVE-1999-0005 Arbitrary command execution via IMAP buffer overflow in authenticate command.

Summary

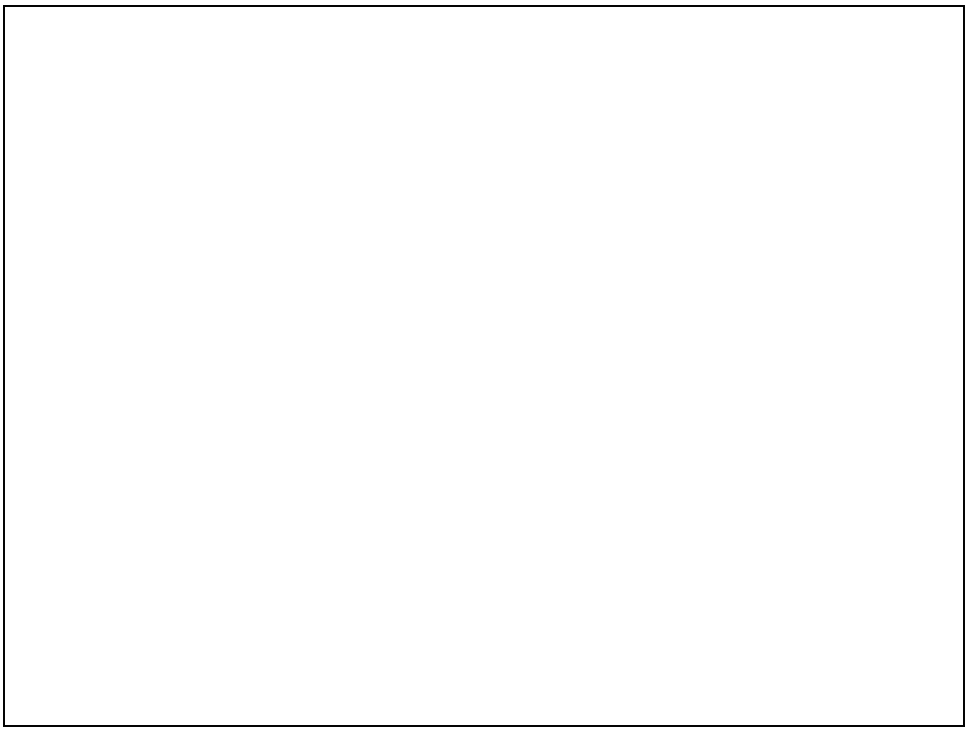
- Vulnerabilities are in ALL current popular OS's
  - Hard to go beyond the “hype” to understand how vulnerable you are given a certain OS
  - Try to discover for yourself how secure OS is that you are using
  - Read bulletins, seek opinions of people you trust and try to protect yourself
  - Buy add-on security products, disable OS features, run with reduced privilege

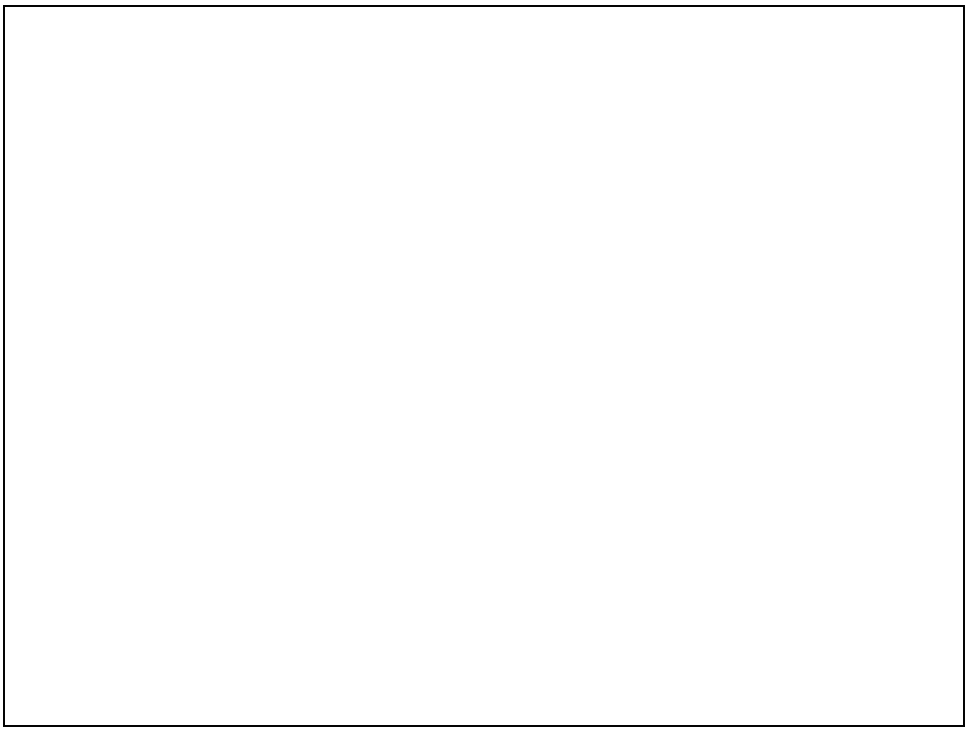
The End

Windows

Linux

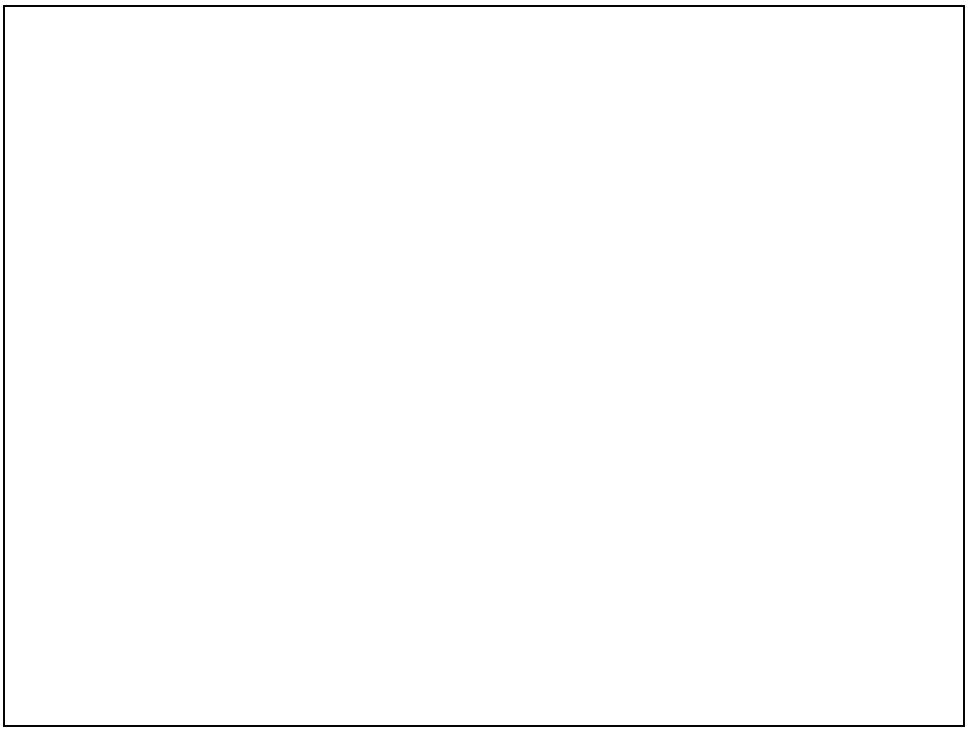
Cheese?





Overview Vulnerabilities

- Looked at designing for security in operating systems
  - Turns out if security is designed in from the beginning, likely system is more secure
- Yet, systems will still have problems even even when security has been carefully considered
- Look at OS vulnerabilities, classification and tools for discovery



Vulnerabilities Defined

 or 

According to Merriam-Webster, Vulnerable Defined

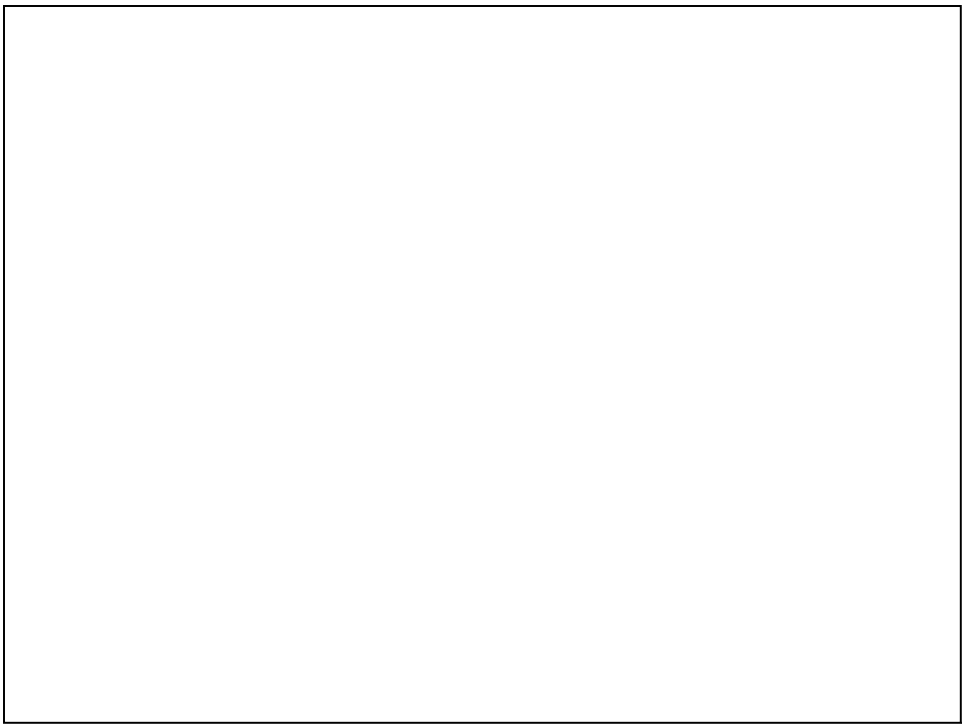
"exposed to possibility of being attacked or harmed, either physically or emotionally: 'we were in a vulnerable position'."

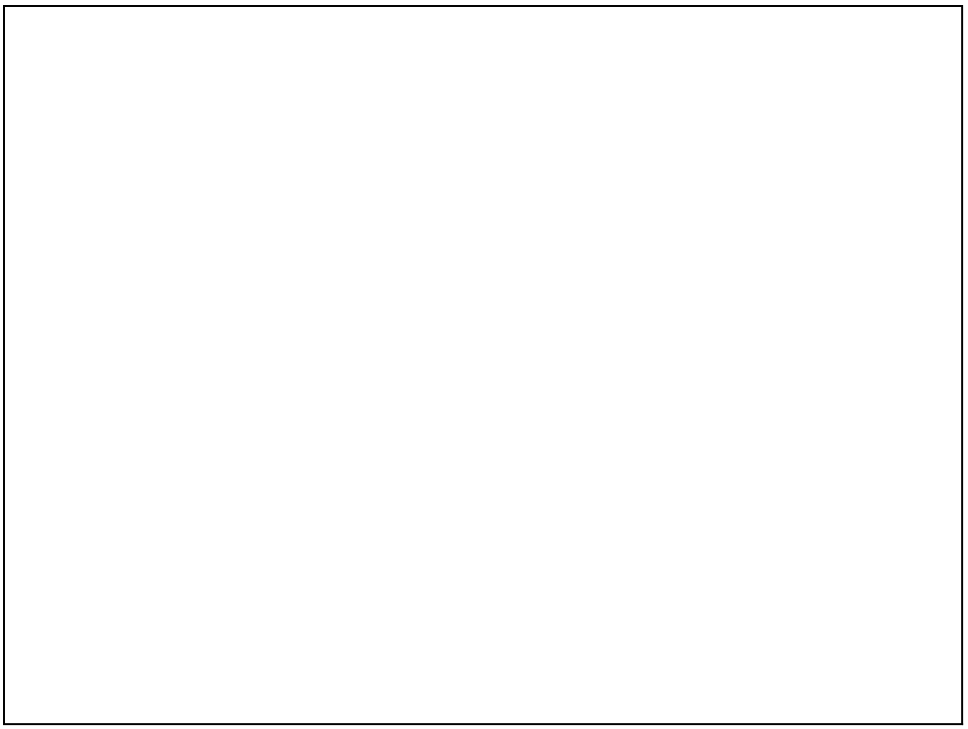
In Computer Security, Vulnerability Defined

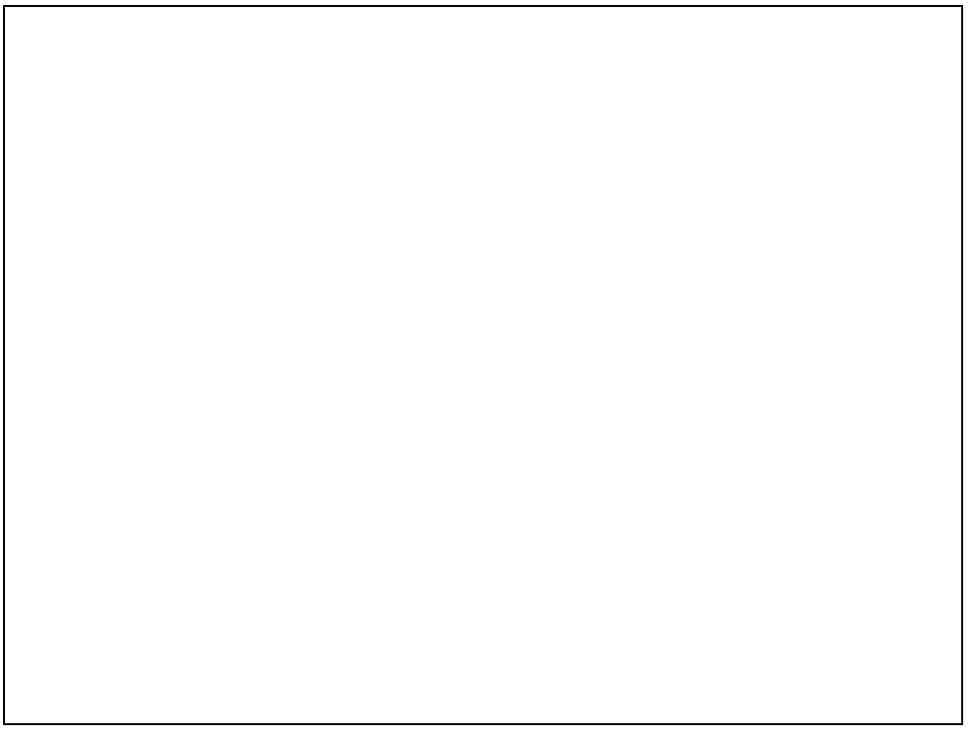
Security Vulnerability refers to system flaw that can leave it open to attack

A vulnerability may also refer to any type of weakness in a

1. Computer system itself,
2. Set of procedures, or
3. Anything that leaves information security exposed to a threat



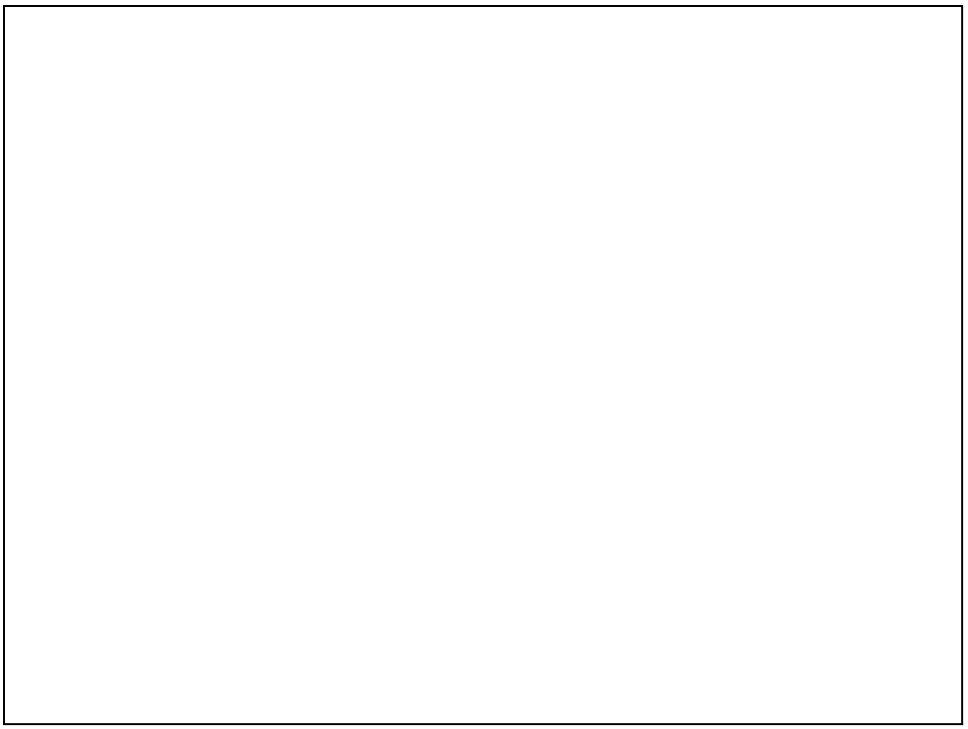


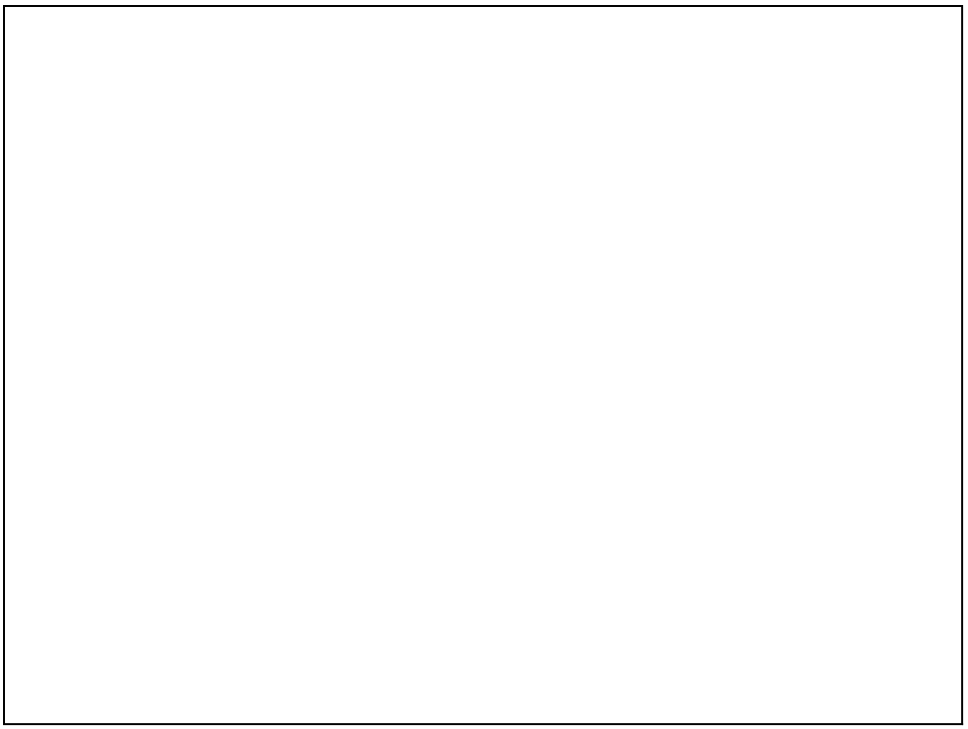


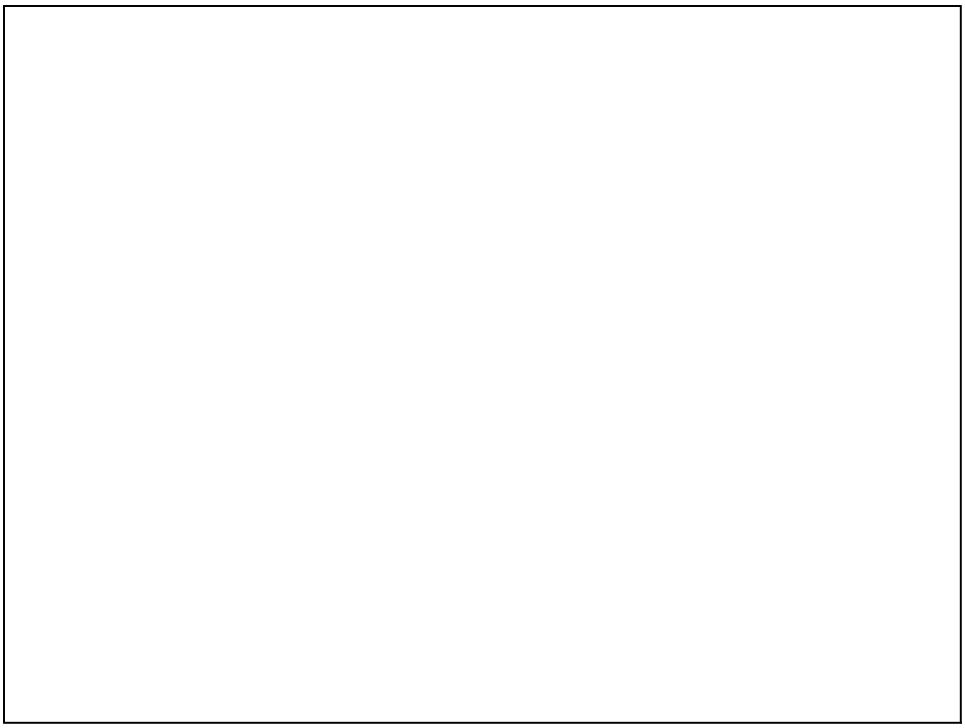
Buffer Defined

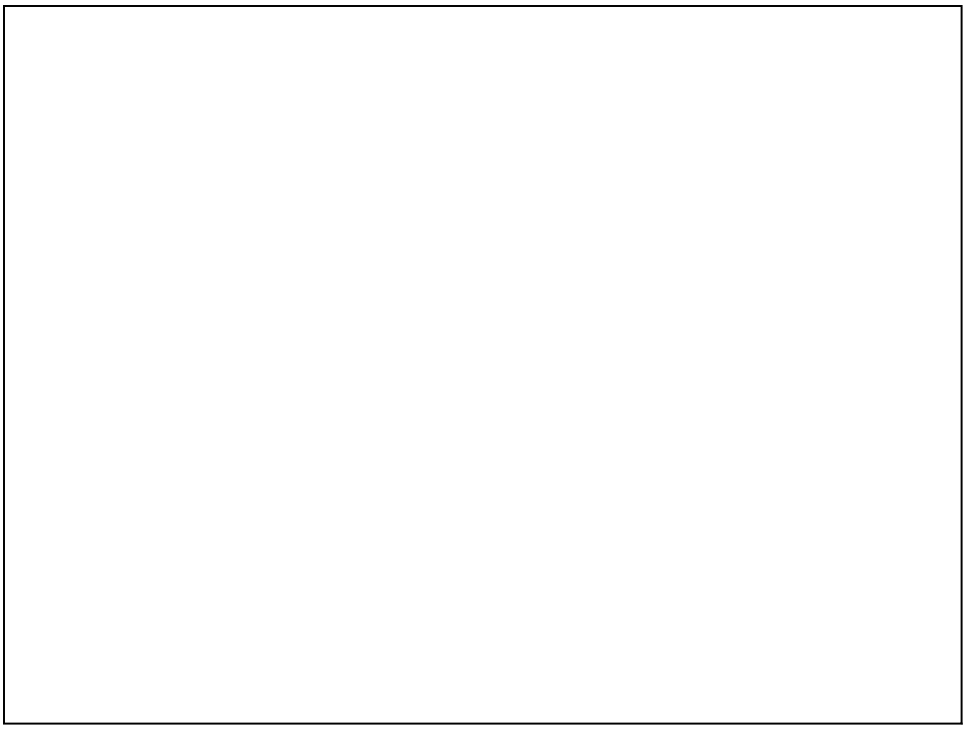
- A temporary storage area, usually in RAM
- Purpose is to act as holding area, enabling CPU to manipulate data before transferring it to a device

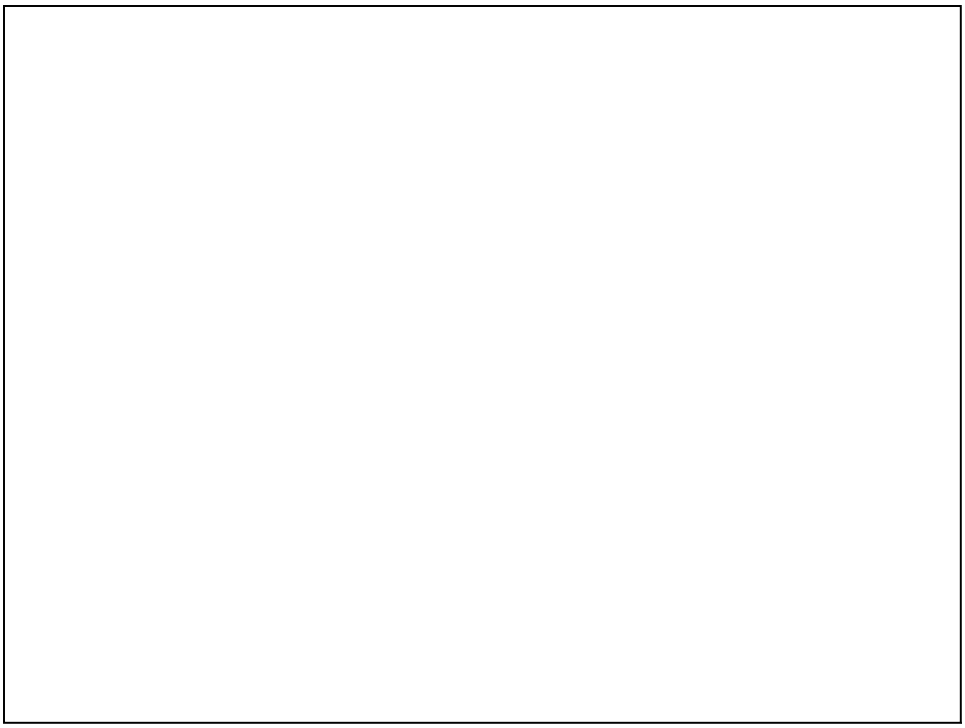
- Process of reading and writing data to a disk are slow, programs keep track of changes in a buffer, then copy buffer to a disk
- For example, word processors employ a buffer to keep track of changes to files
- Java has concept of buffered reads and writes
  - Only writes or reads from disk in batches





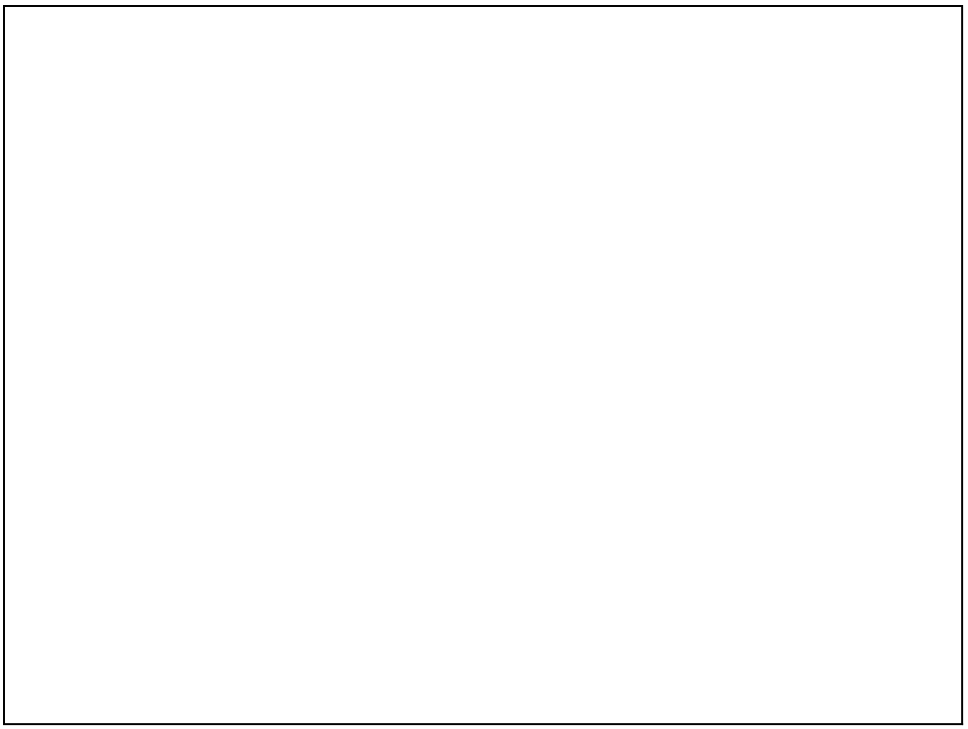






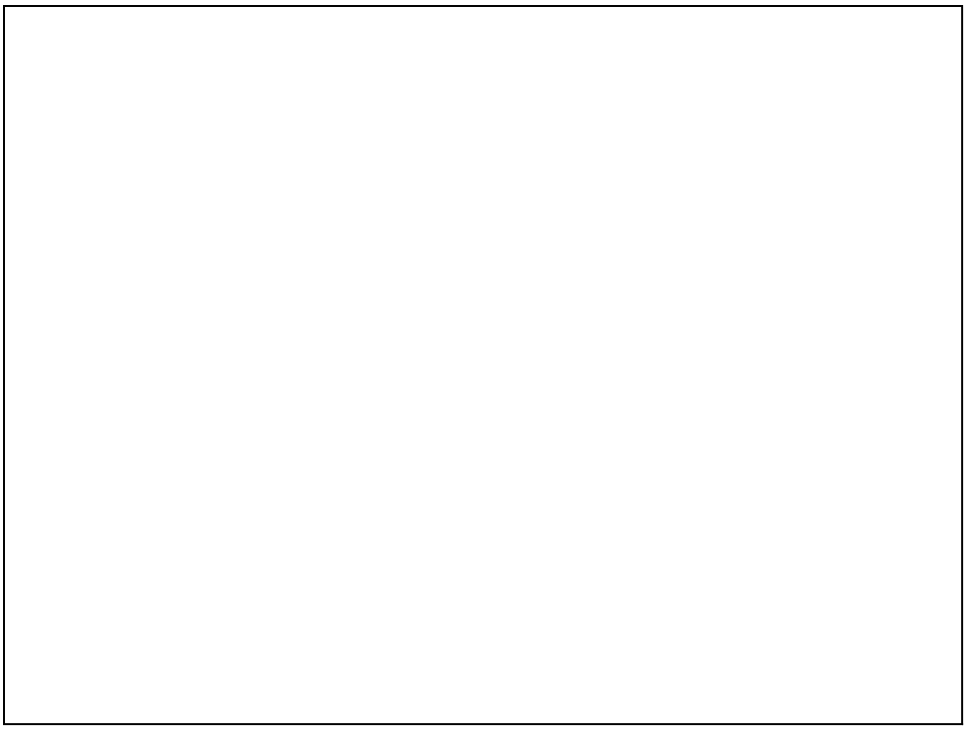
What is the Server Message Block?

- Operates as an **application-layer network protocol**
  - Provides shared access to files, printers, serial ports
  - Also provides an authenticated inter-process communication mechanism
  - Shortcut in the MS world to client/server applications !!!



Linux Buffer Overflow Vulnerabilities

- Is Linux or Mac OS X immune to buffer overflows?
  - No. They have these too …
- Google search of “buffer overflow vulnerabilities in linux”
  - Came back with 717,000 hits
  - Among the problems
    - Stack based X-Windows vulnerability
      - Affects all linux distributions
    - Adobe flash player – Linux
    - Critical flaw in glibc, GNU C Libraray



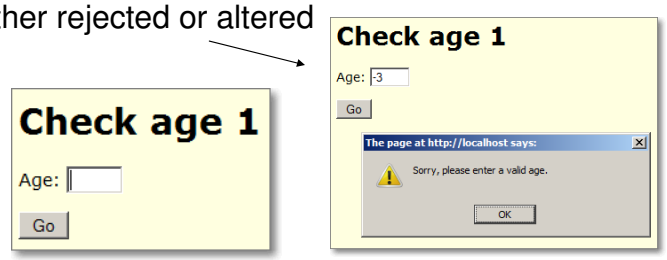
Validating Input

- Input needs to meet programmer expectations

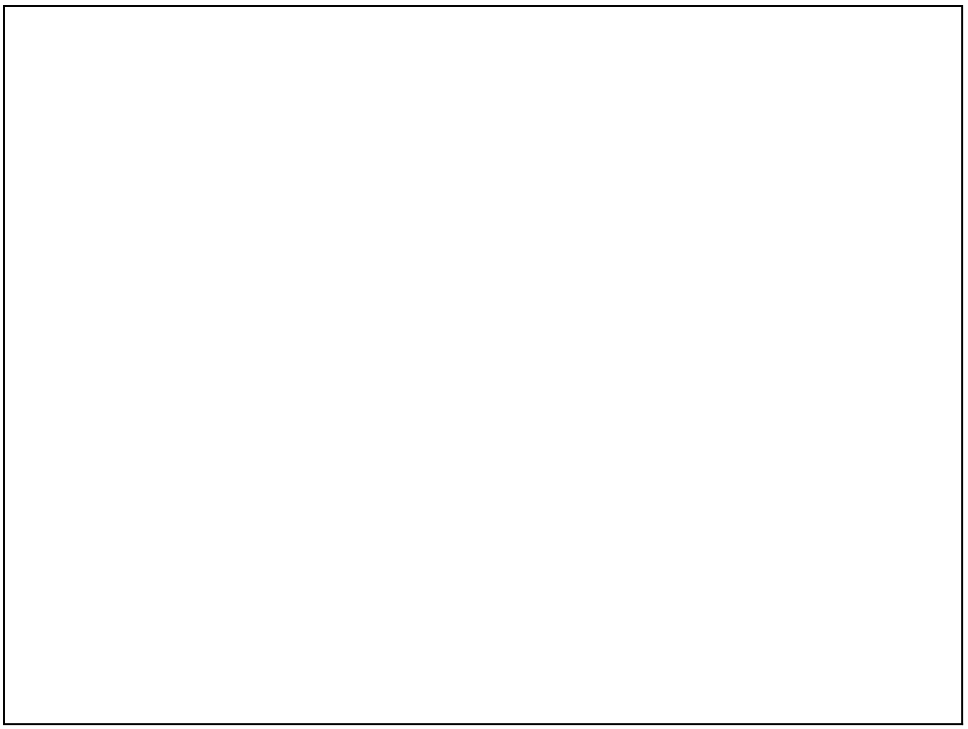
  For whatever input required:

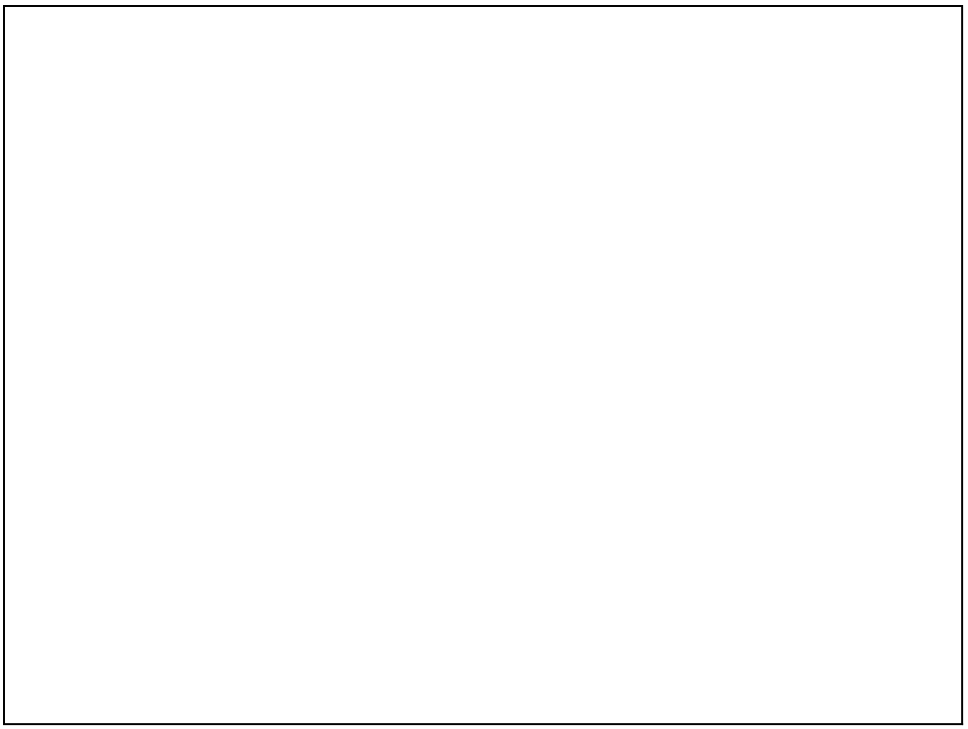
  - HTML, email, userid or valid database request
- Compare input to what is known to be acceptable

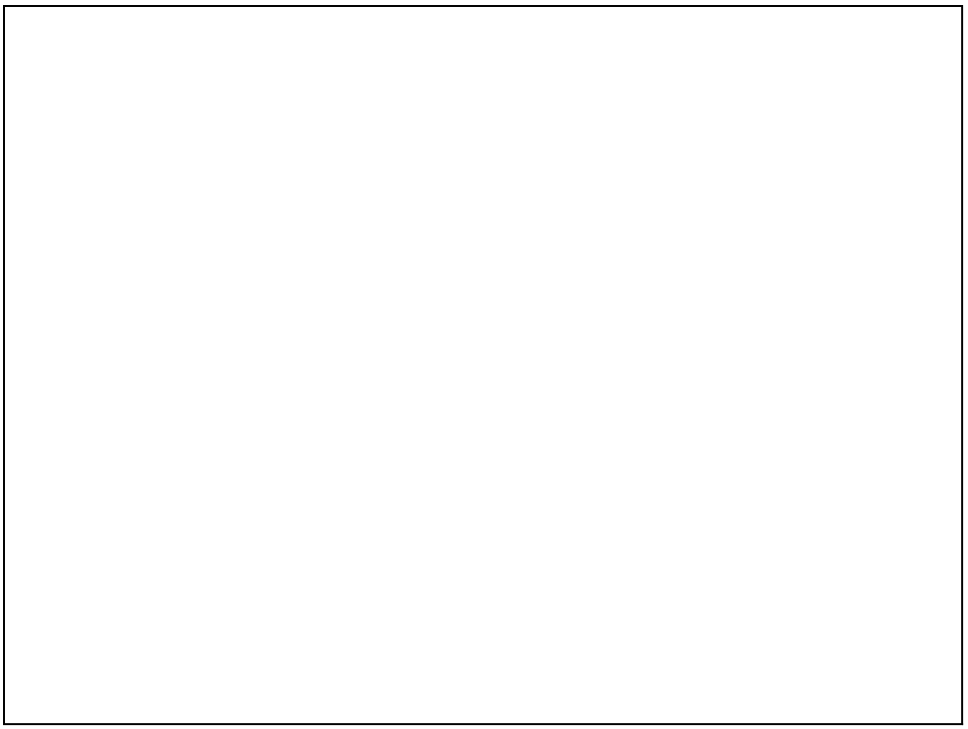
  Use regular expressions, patterns of characters describe allowable input
- Bad input is either rejected or altered

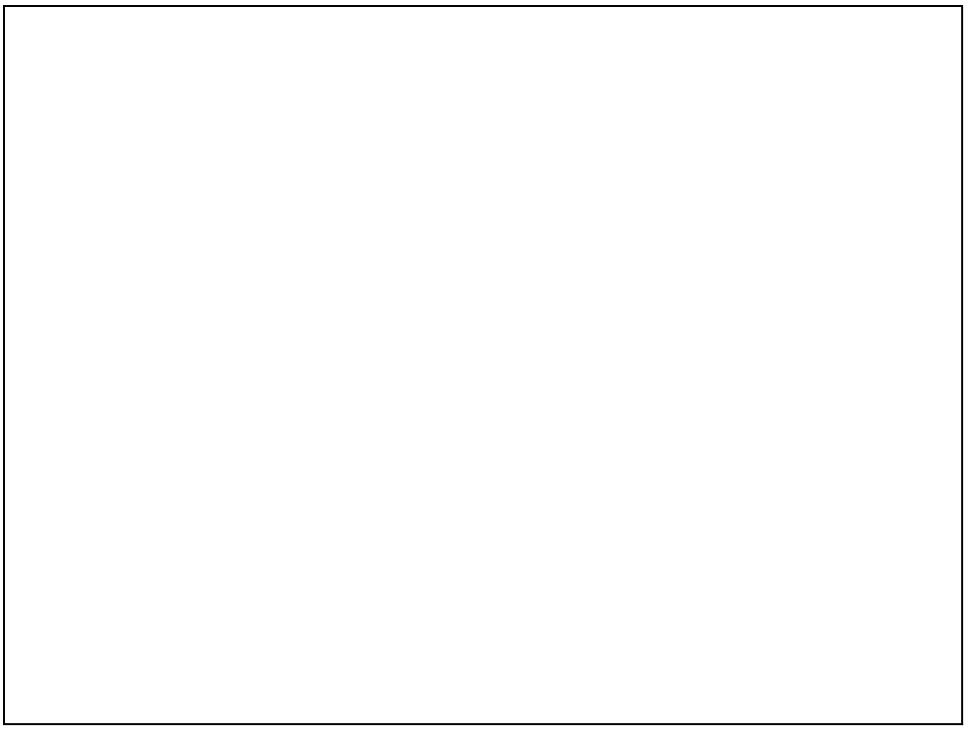


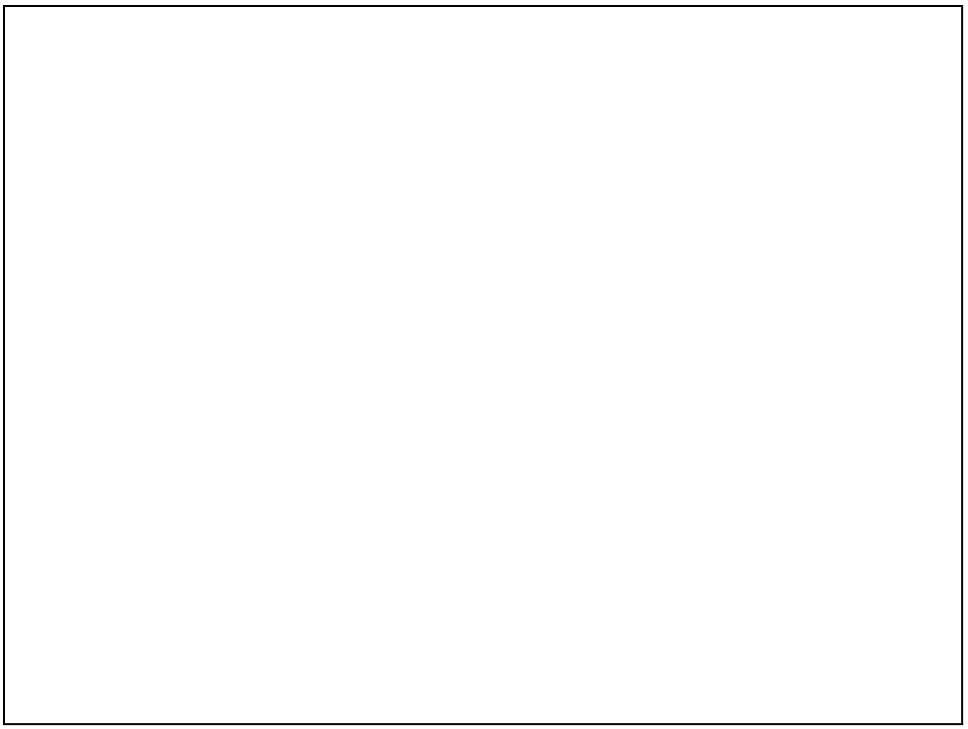
Given the programmer cannot control the content of input data, it is necessary to ensure that such data conforms with any assumptions made about it before subsequent use. If the data is textual, these assumptions may be that it contains only printable characters, has certain HTML markup, is the name of a person, a userid, an email address, a filename or is a URL. A program using such input should confirm that it meets these assumptions. An important principle is that input data should be compared against what is wanted, accepting only valid input. The alternative is to compare the input data with known dangerous values. The problem with this approach is that new problems, and methods of bypassing existing checks continue to be discovered. By trying to block known dangerous input data, an attacker using a new encoding may succeed. By only accepting known safe data, the program is more likely to remain secure. This type of comparison is commonly done using regular expressions. It may be explicitly coded by the programmer, or may be implicitly included in a supplied input processing routine. A regular expression is a pattern composed of a sequence of characters that describe allowable input variants. Some characters in a regular expression are treated literally, and the input compared to them must contain those characters at that point. Other characters have special meanings, allowing the specification of alternative sets of characters, classes of characters, and repeated characters. Details of regular expression content and usage vary from language to language. An appropriate reference should be consulted for the language in use. If the input data fails the comparison, it could be rejected. Alternatively, it may be altered so that it does conform. This generally involves “escaping” metacharacters to remove any special interpretation, thus rendering the input safe.



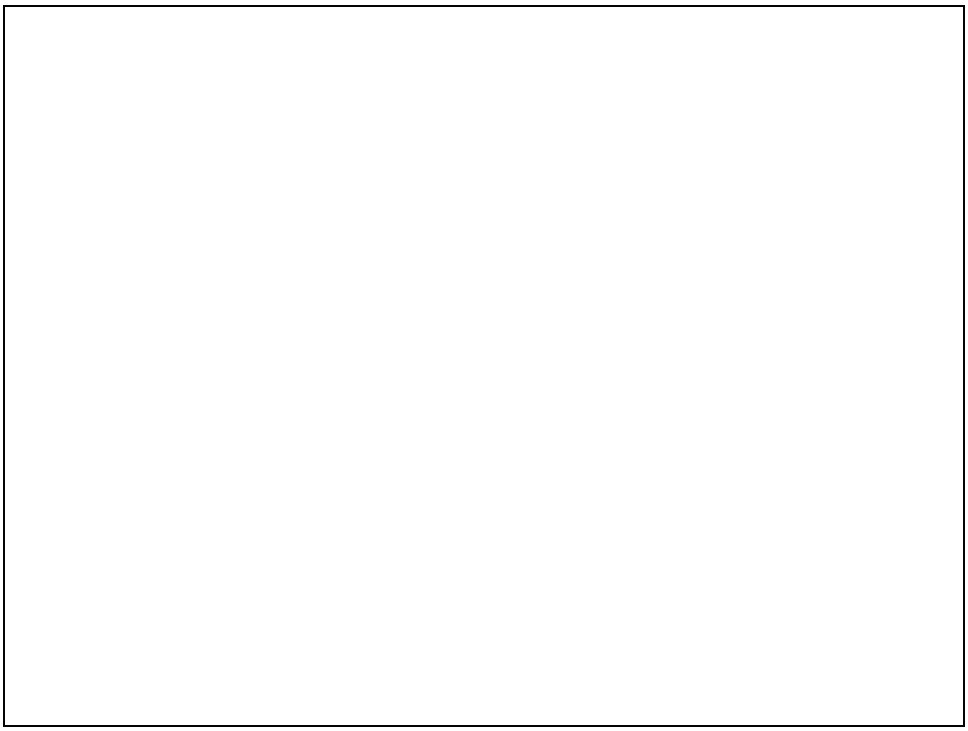


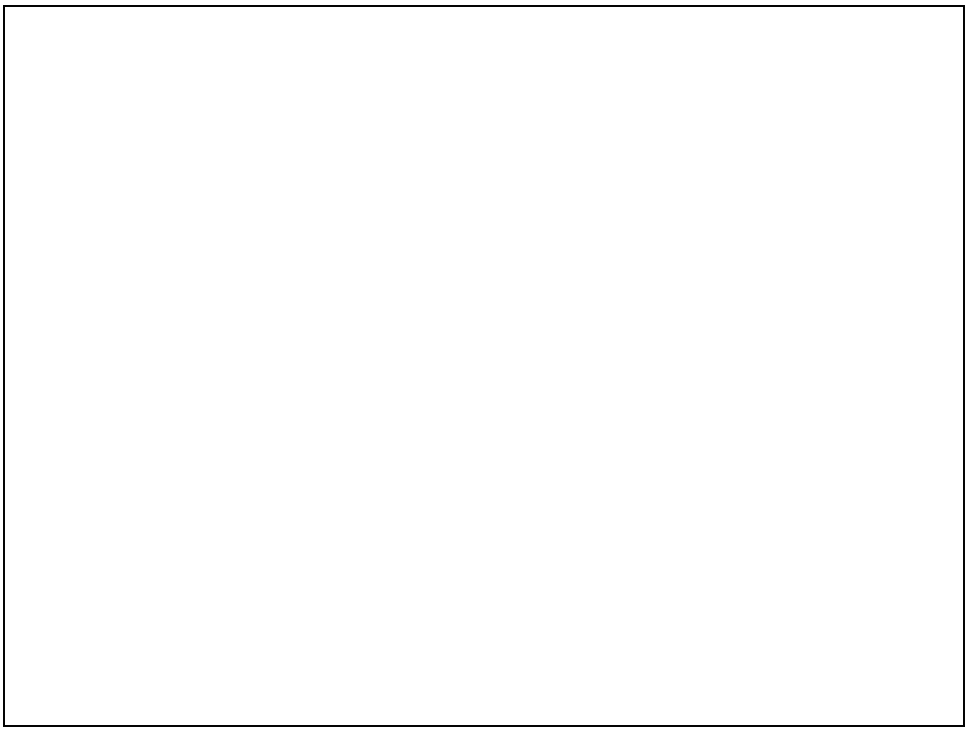


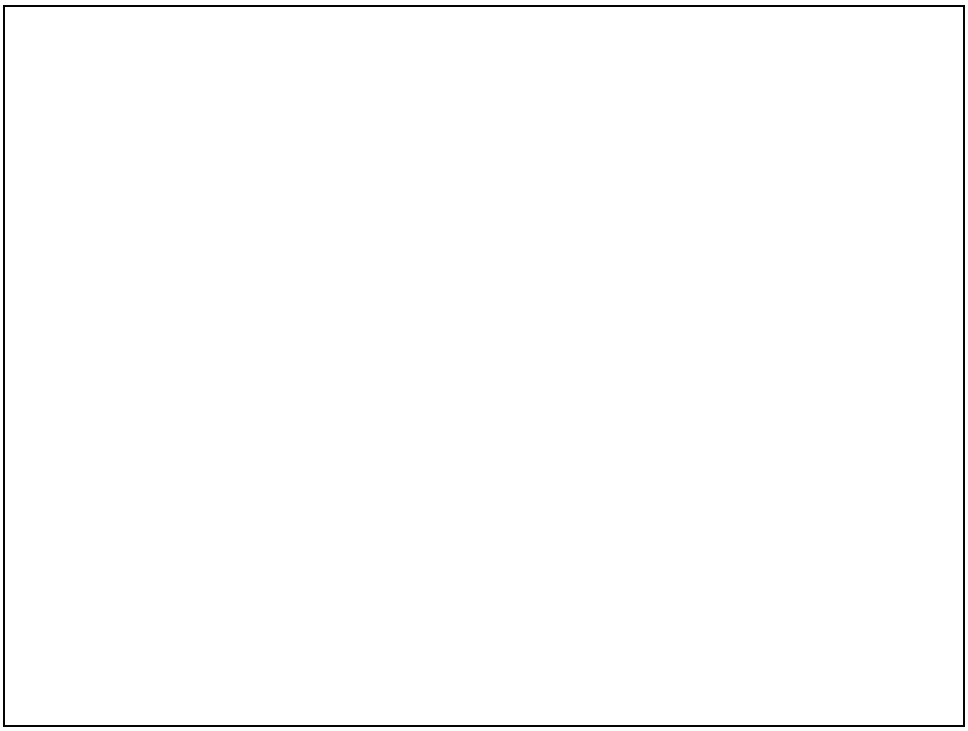


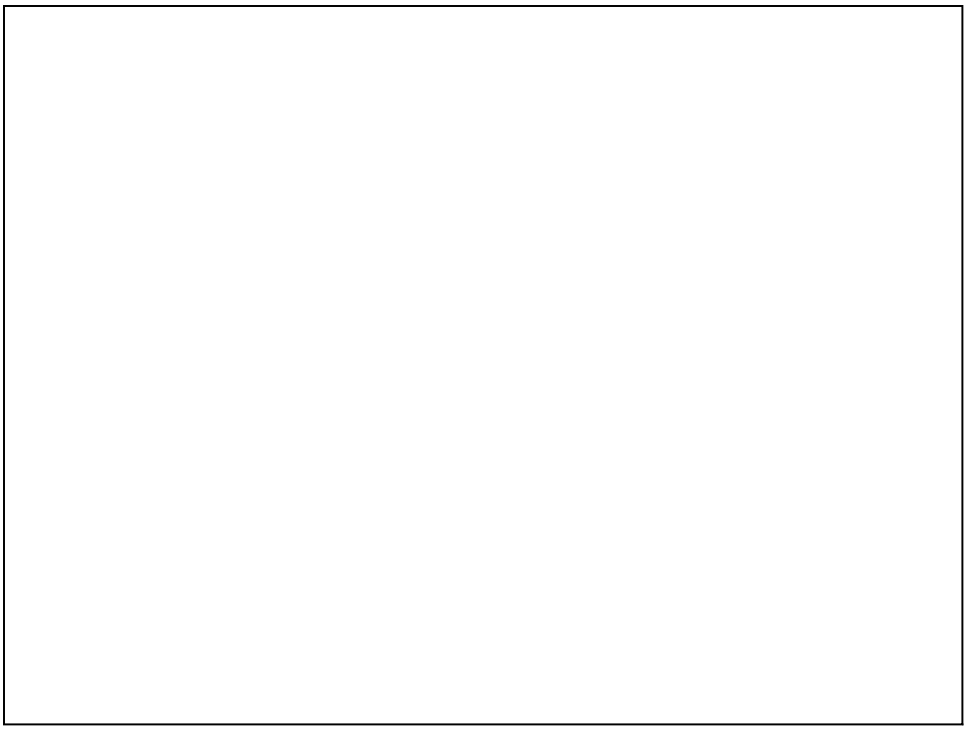


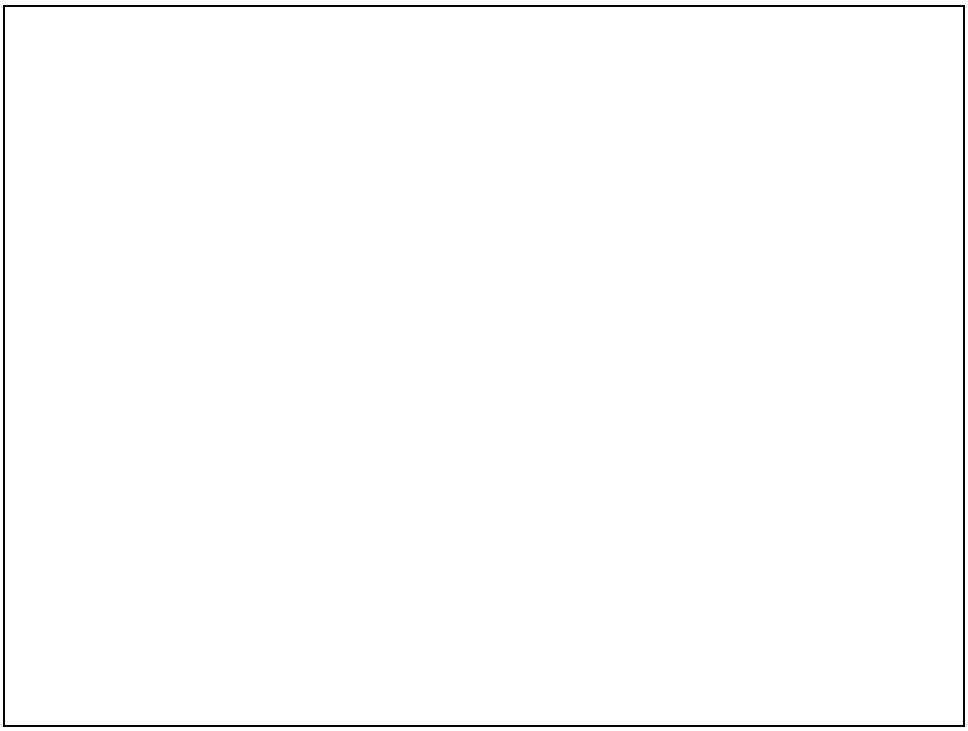
Vulnerabilities, Finding them, Publishing Them

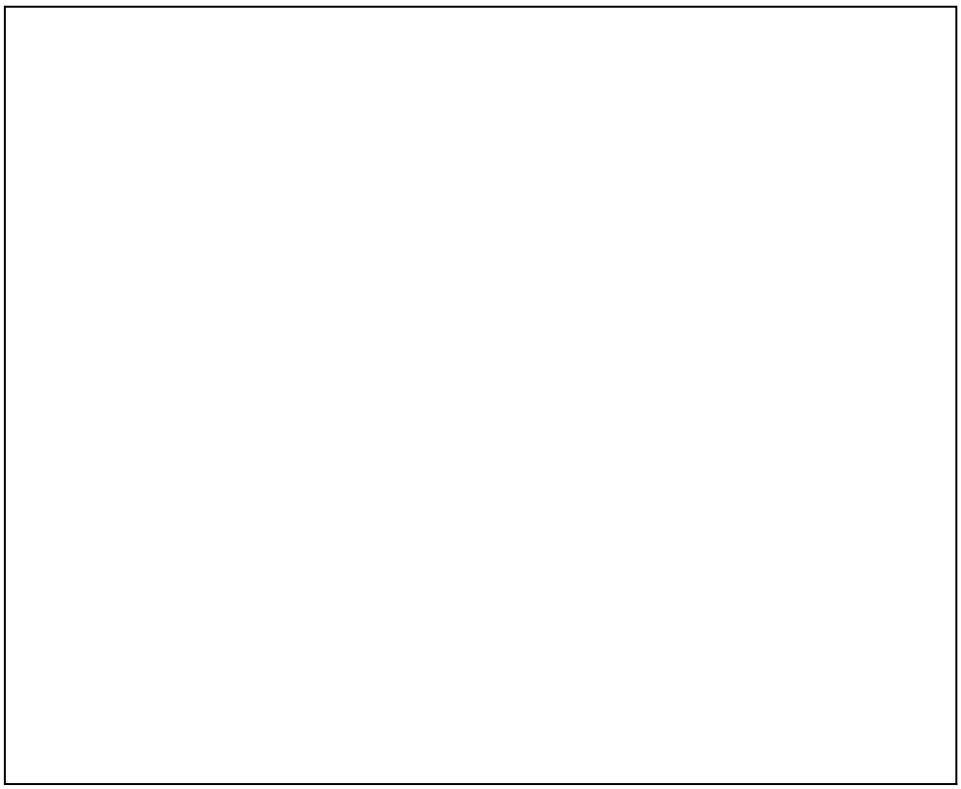


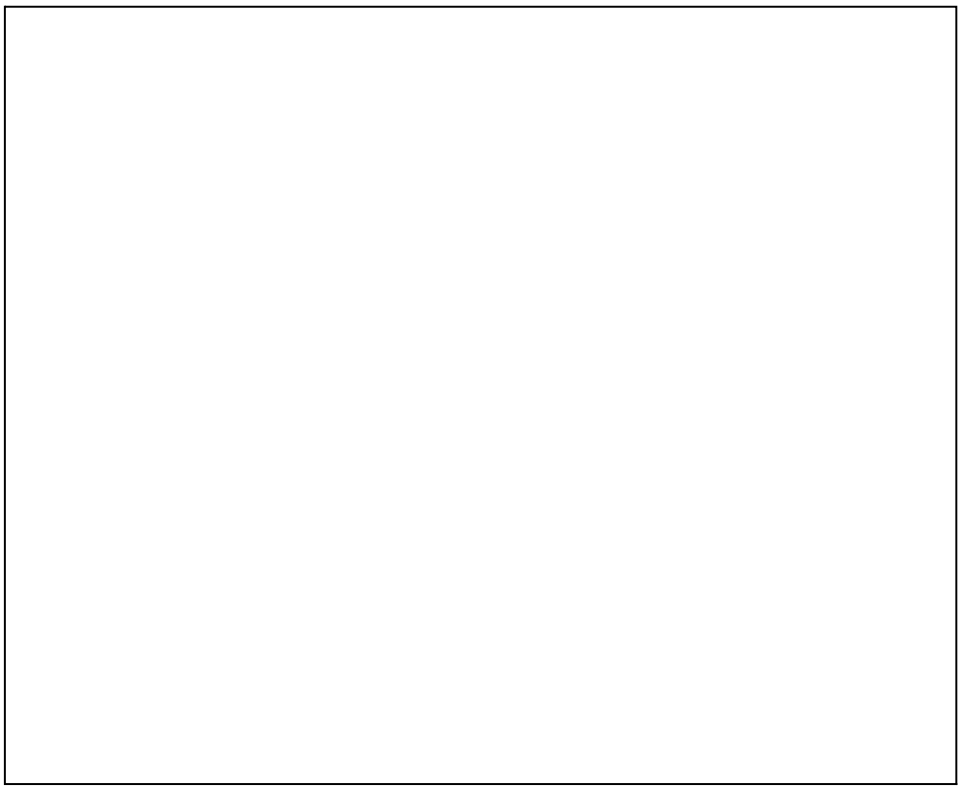


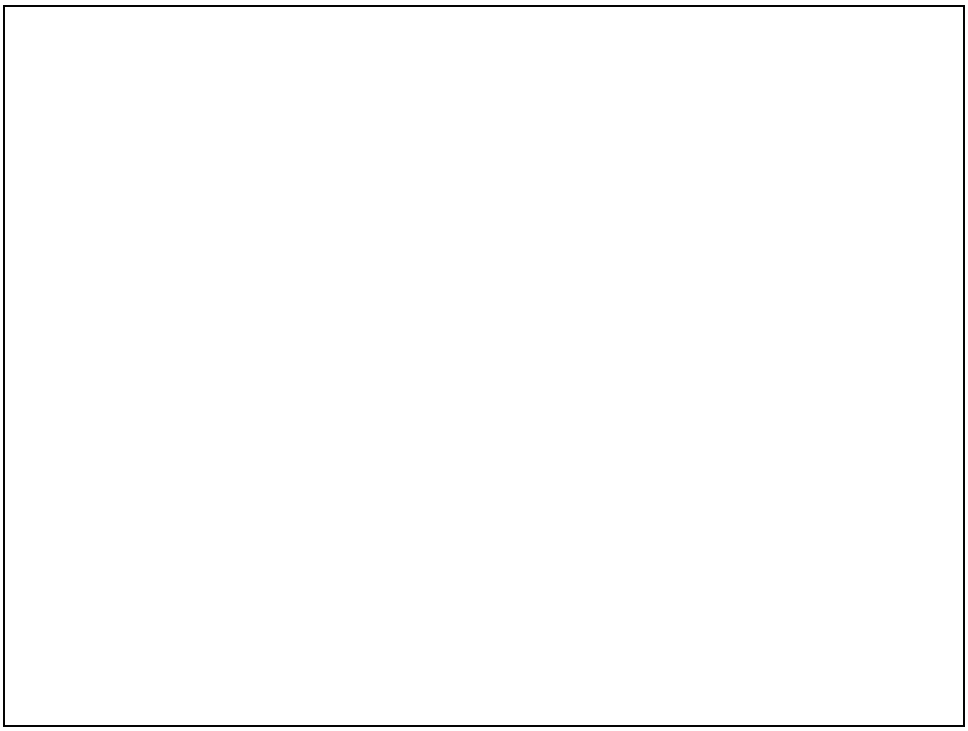


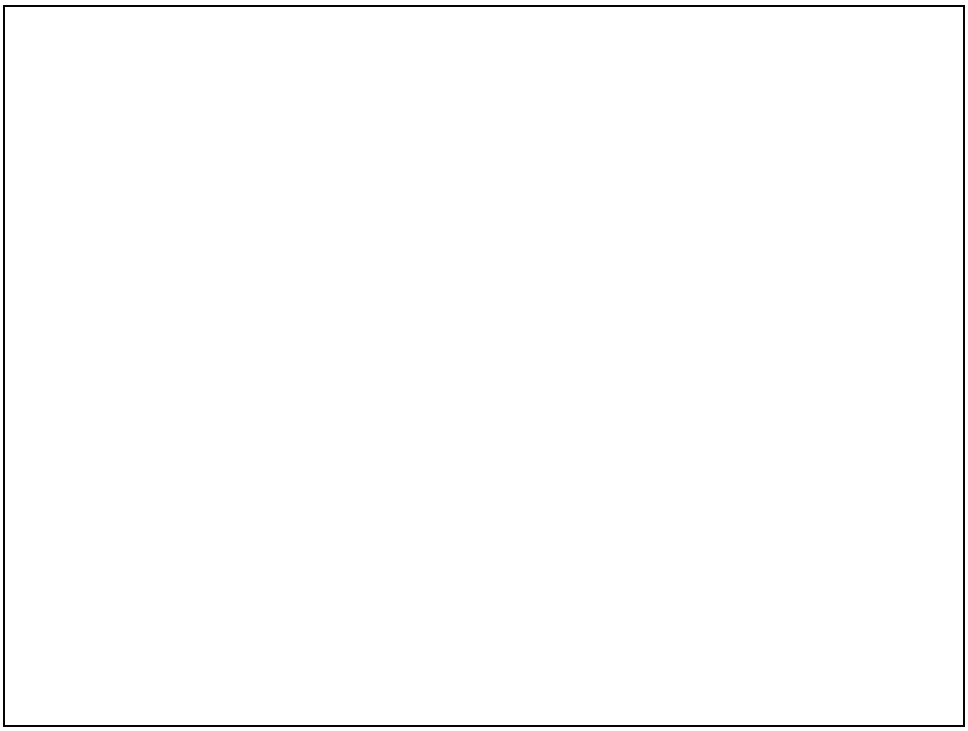


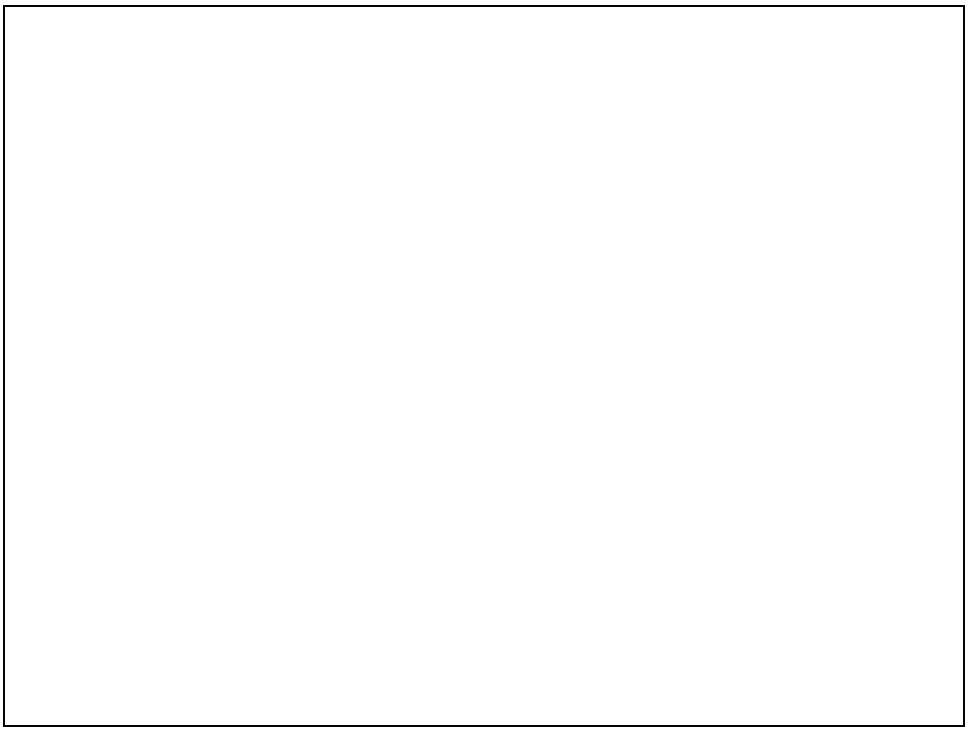


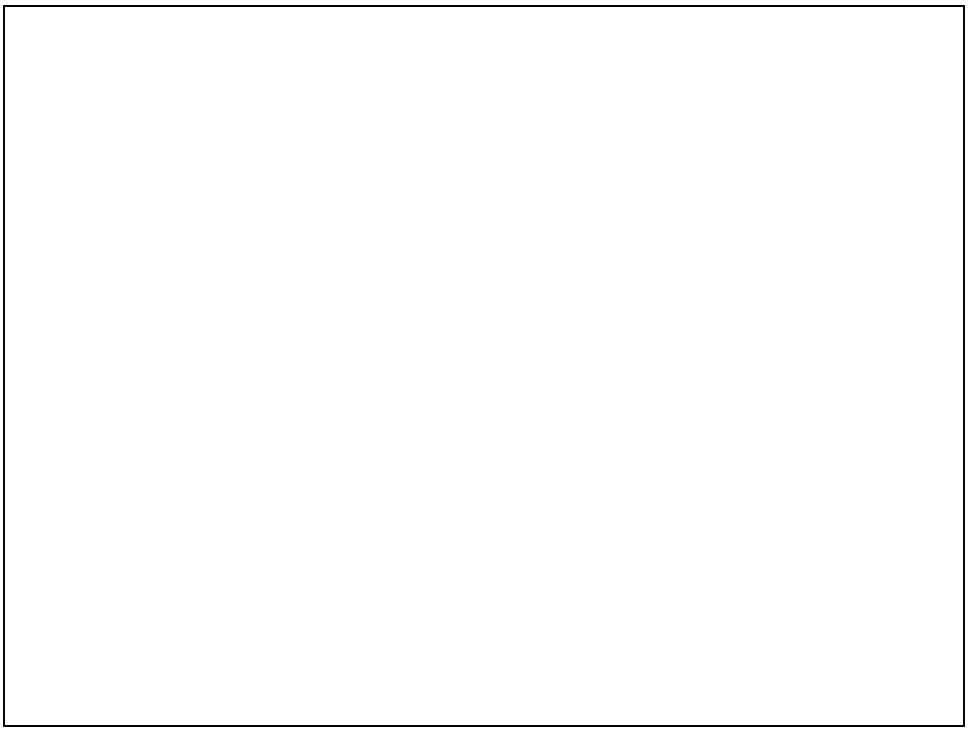


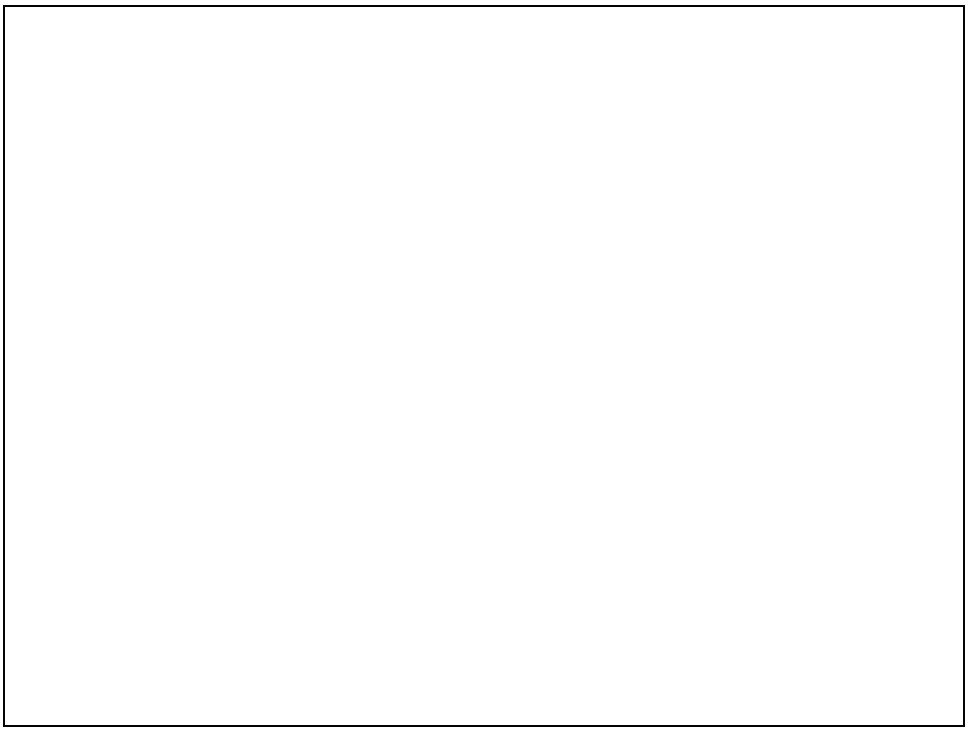


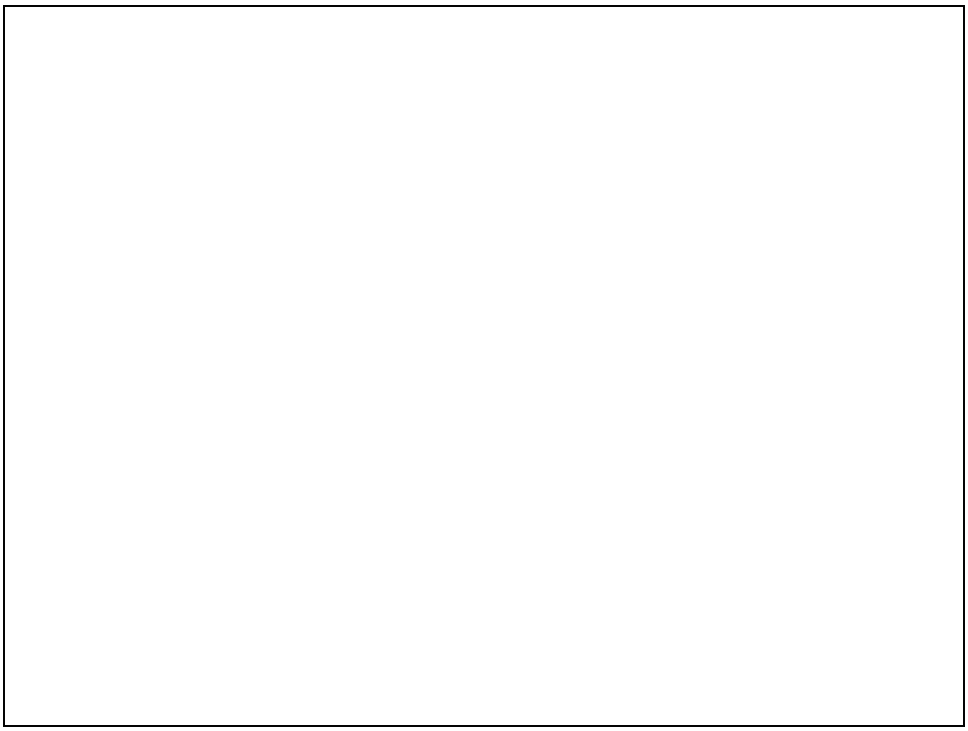


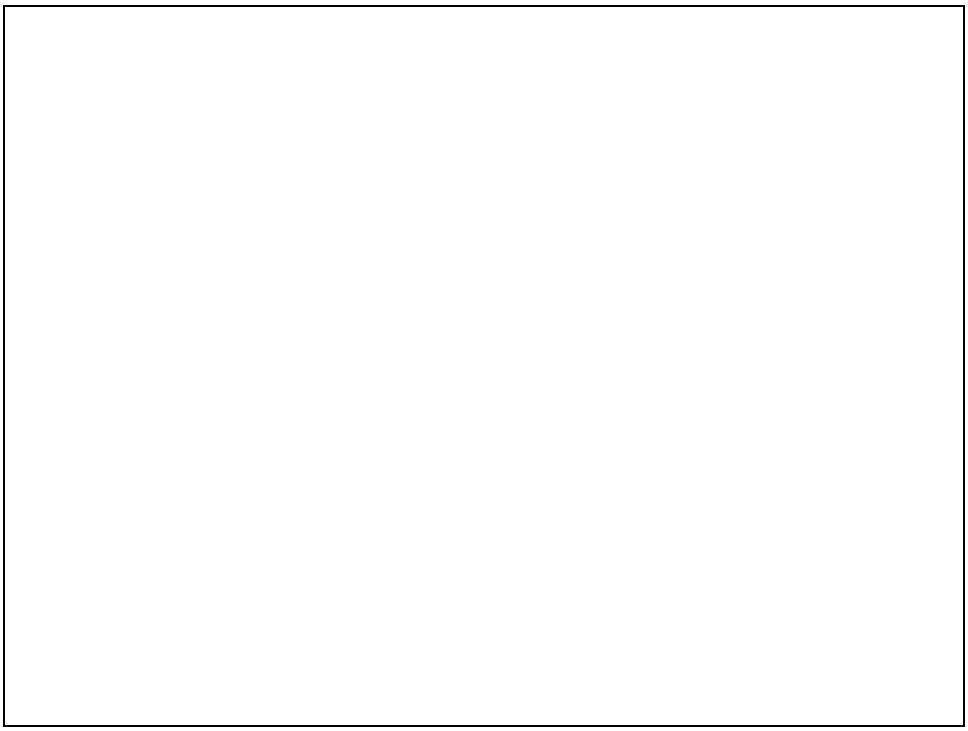


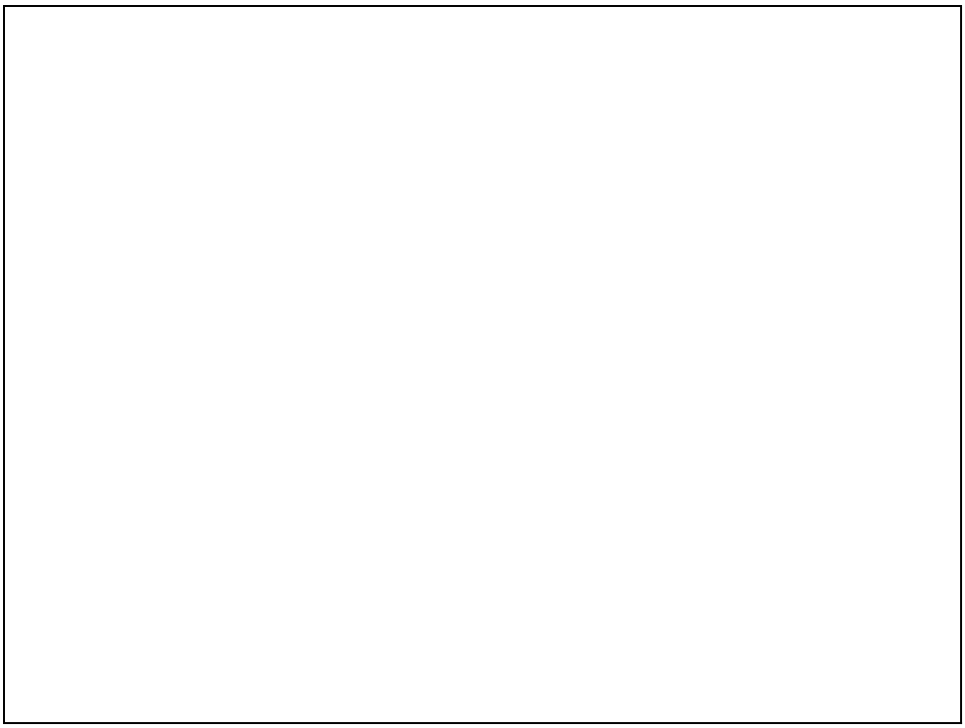


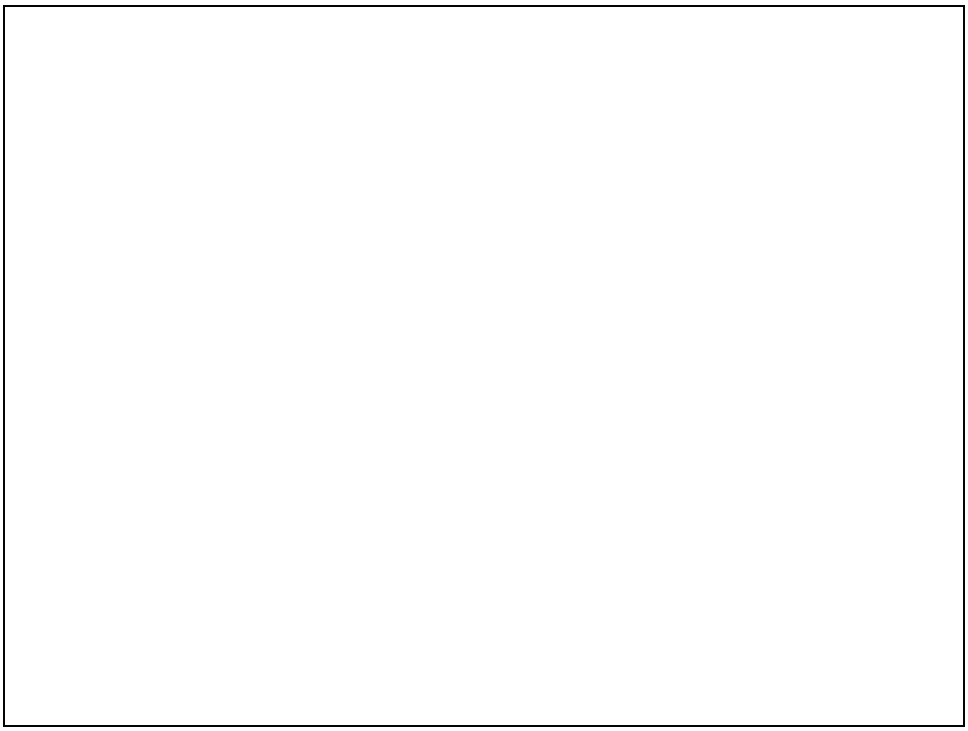












Purposes of a VDB

Provide accurate information on security vulnerabilities

Provide historic reference on software bugs

Provide information on solutions

Provide innovations to help organizations deal with vulnerabilities

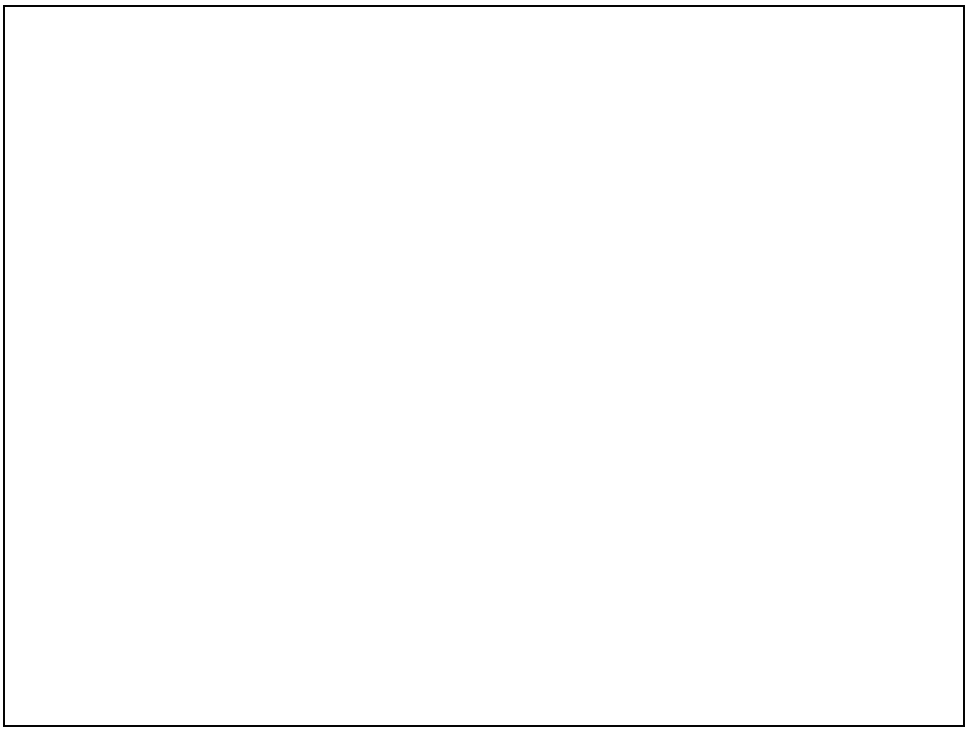
Major Players

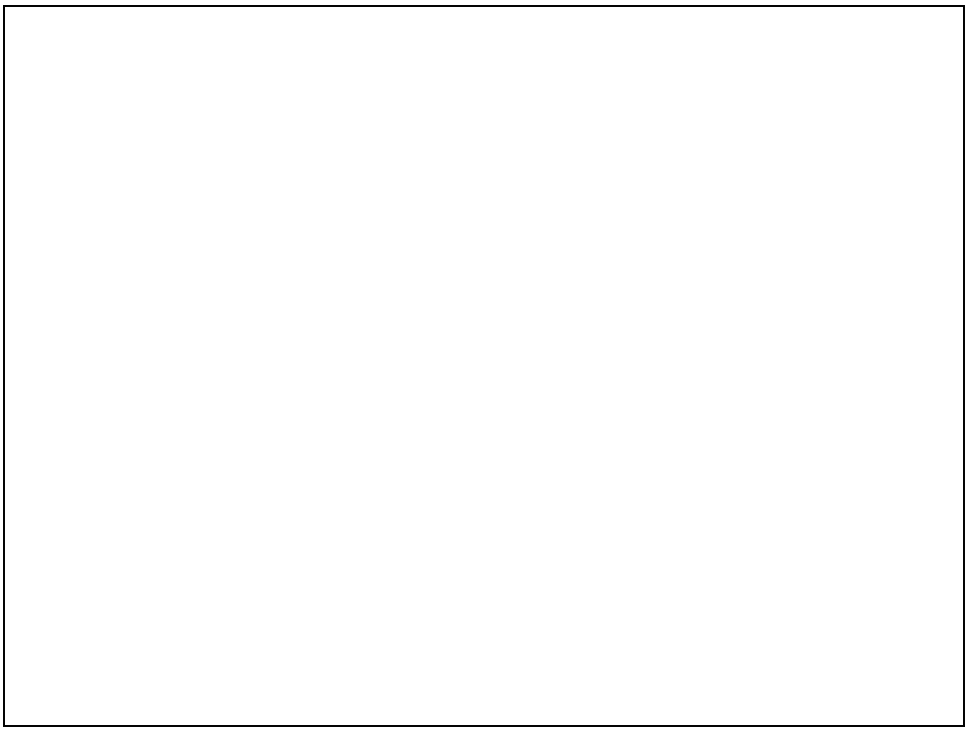
Comprehensive VDBs

NatVulDataBase - http://nvd.nist.gov/
BID - http://www.securityfocus.com/bid
CVE - http://www.cve.mitre.org/
ISS X-Force - http://xforce.iss.net/
OSVDB – http://www.osvdb.org/
Secunia - http://www.secunia.com/
Security Tracker - http://www.securitytracker.com/

Vulnerability Notification Services

CERT - http://www.cert.org/
CIAC Advisory - http://www.ciac.org/ciac/index.html





National Vulnerability Database

- National Vulnerability Database

  http://web.nvd.nist.gov/view/vuln/search?execution=e2s1

  How would you use this resource?

  You can search this database for all the vulnerabilities associated with a system