Demystifying Penetration Testing

Prepared by

Debasis Mohanty

www.hackingspirits.com

E-Mail: debasis_mty@yahoo.com

Goals Of This Presentation

- An overview of how Vulnerability Assessment (VA) & Penetration Testing (PT) is done
- Defining scope of the assessment
- Types of Penetration Testing
- A brief understanding on how Buffer Overflow works
- How vulnerabilities are scanned and exploited
- What are the end results
- What a Penetration Testing Report should contain

Acronyms:

- VA – Vulnerability Assessment
- PT – Penetration Testing
- DOS – Denial of Service
- DDOS – Distributed Denial of Service

Difference Between Vulnerability Assessment and Penetration Testing

Vulnerability Assessment (VA)

In this case the security auditor has to only scan for the vulnerabilities in the server or application and filter out the false positives from the scan output by mapping them with the actual vulnerabilities associated with the target host.

VA Scope Includes:

- The VA test can be done both internally and externally
- No vulnerabilities are exploited
- No dangerous attacks like DOS and Buffer Overflow attacks are used
- Automated vulnerability scanning tools line Nessus, Retina or ISS are used

Penetration Testing (PT)

In this case the security auditor or the penetration tester not only has to scan for the vulnerabilities in the server or application but also has to exploit them to gain access to the remote server.

PT Scope Includes:

- The PT test is done both internally and externally
- Vulnerabilities are exploited
- Dangerous attacks like DOS and Buffer Overflow attacks are used depending upon the customer's willingness to do so
- Automated vulnerability scanning tools and as well as exploits are used

Types Of Penetration Testing

Black Box Penetration Testing

- Pen tester has no previous knowledge of the remote network
- Only the company name or the IP address is known
- Simulation of a real world hacking by a hacker who has no knowledge (E.g. Operating System running, application running, device type and network topology etc..) of the remote network environment

White Box Penetration Testing

- Pen tester provided with significant knowledge of the remote network
- Type of network devices (i.e. Cisco gear, TCP/IP),
- WebServer details (i.e., Apache/*nix or Apache/Win2k),
- Operating System type (i.e., Windows/*nix),
- Database platform (i.e., Oracle or MS SQL),
- Load balancers (i.e. Alteon),
- Firewalls (i.e. Cisco PIX).. etc
- Simulation of an attack by a hacker who is having a detailed knowledge of the remote network environment

Scope Of Penetration Testing

Non-Destructive Test

- Scans the remote hosts for possible vulnerabilities
- Analyze and confirm the findings
- Map the vulnerabilities with proper exploits
- Exploit the remote system with proper care to avoid disruption of service
- No highly critical Denial of Service (DoS) attack is tried

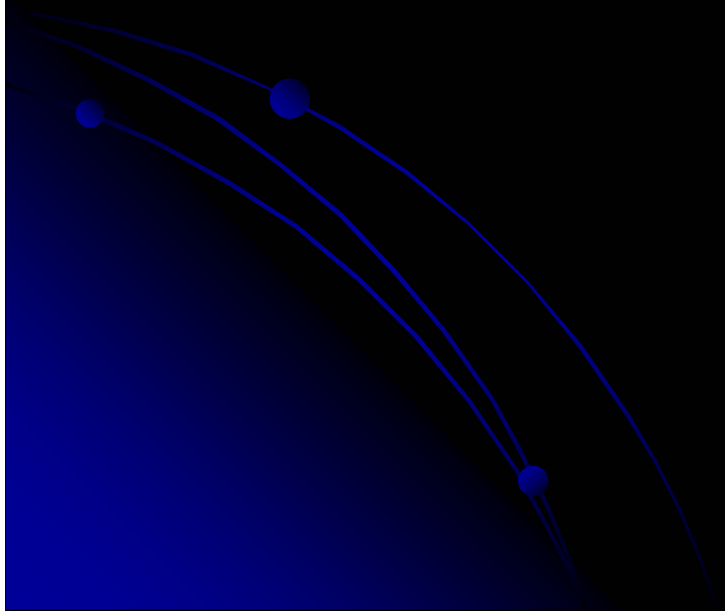
Destructive Test

- Scans the remote hosts for possible vulnerabilities
- Analyze and confirm the findings
- Map the vulnerabilities with proper exploits
- All highly critical Denial of Service (DoS) attacks (e,g like buffer overflows) are tried

Scope Of Penetration Testing (Contd…)

Types of Environment

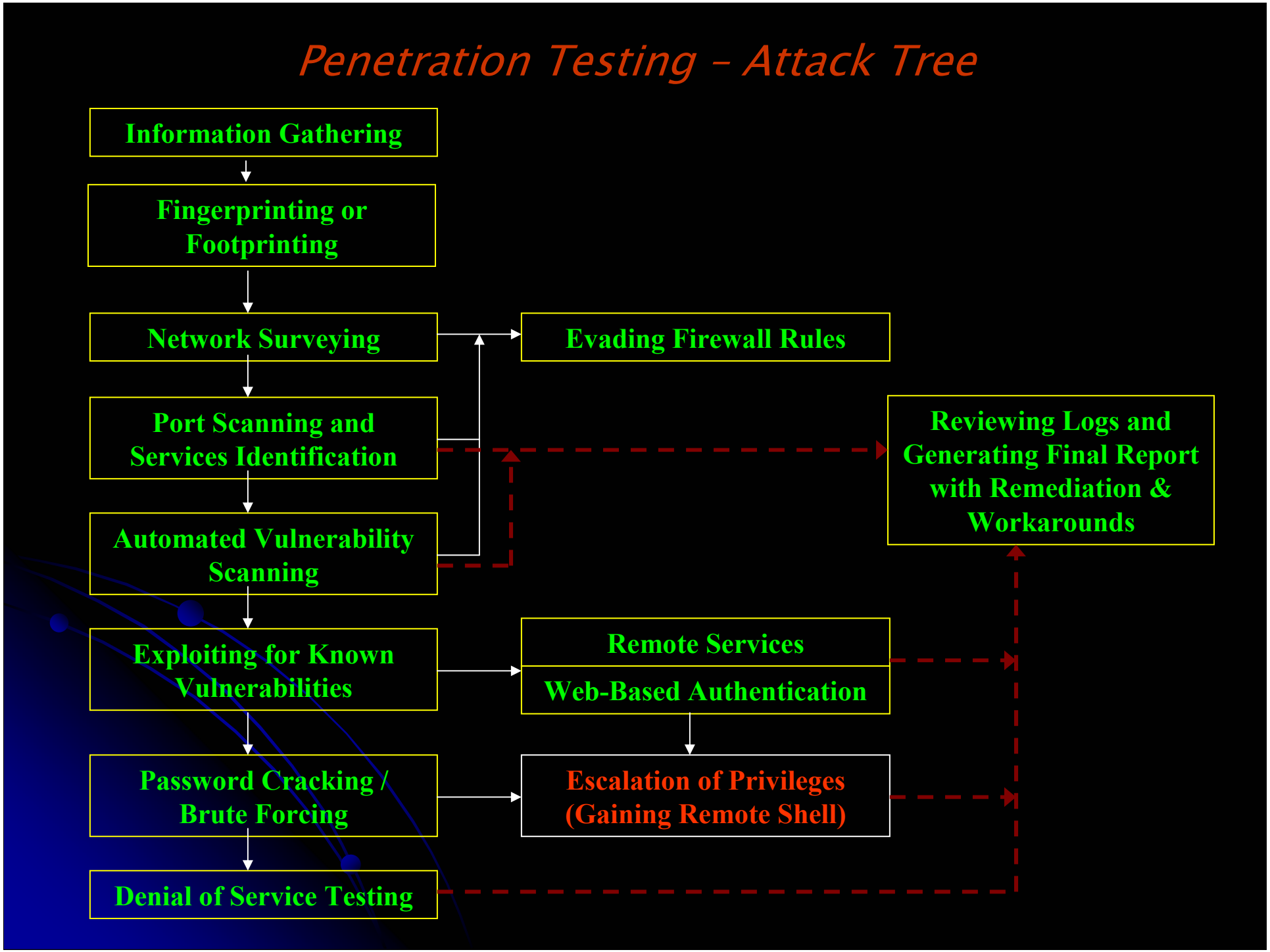
- Wireless Networks
- DMZ environments
- Internet Data Centers (IDC)
- Portal Environment
- Extranet
- VPN Termination points
- Remote Access points
- Dial-In



AN Approach To Penetration Testing

- Information Gathering
- Fingerprinting or Footprinting
- Network Surveying / Network Mapping
- Ports Scanning and Services Identification
- Evading Firewall Rules
- Automated Vulnerability Scanning
- Exploiting Services for Known Vulnerabilities
- Exploiting Web-Based Authorization
- Password Cracking / Brute Forcing
- Denial of Services (DoS) Testing
- Escalation of Privileges

Penetration Testing – Attack Tree



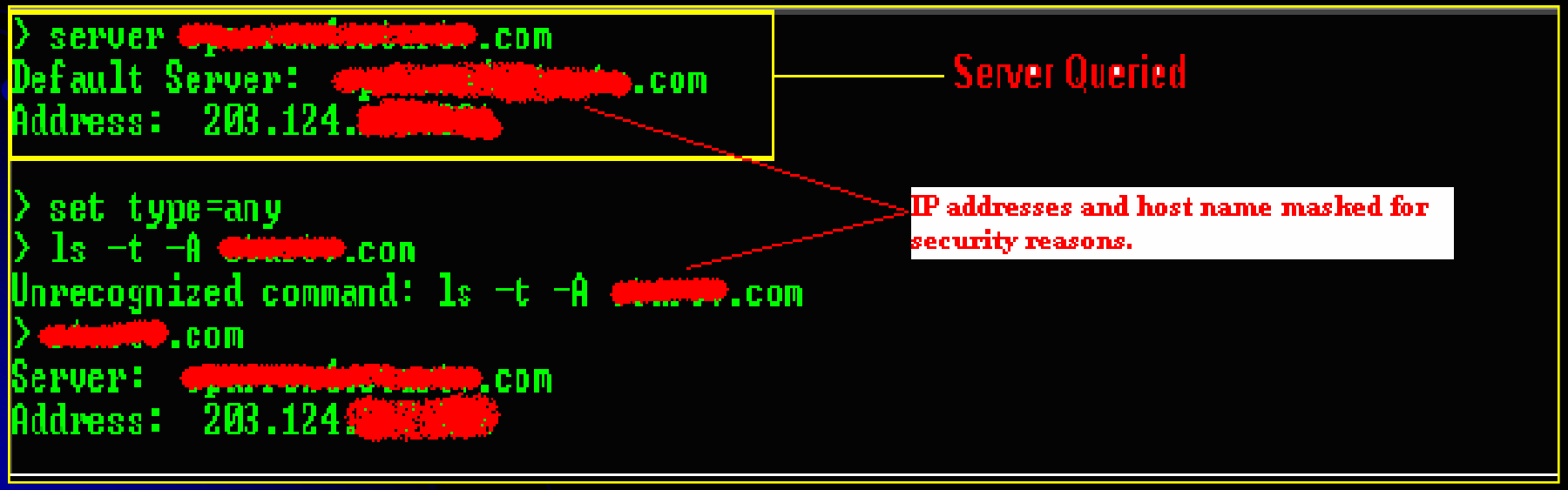
1. Information Gathering

This is the first step for any remote host Penetration Testing. Here the pen-tester try to gather maximum information on the remote host to precise the attack.

Expected Results:

- ❑ Zone Transfer Information
- ❑ Domain Registration Information
- ❑ Email IDs
- ❑ IP Addresses Range

Sample Screenshot (Server queried for Zone-Transfer Info):



1. Information Gathering (Contd…)

Sample Screenshot: (Information Gathered from Zone-Transfer Info)

2. Footprinting / Fingerprinting

In this step, information like WebServer and OS type running on remote host are gathered to further precise the attack.

Expected Results:

- Remote server OS type
- Remote server web-server type
- Applications running on remote server

Sample Screenshot (Banner displaying OS, application & WebServer details):

```
$ ./ap_scalp 6 203.124.[redacted]:80

[*] Connecting.. connected!
[*] Currently using retaddr 0x932ae, length 29896, localport 48684
HTTP/1.1 302 Found
Date: Sat, 28 Feb 2004 18:03:03 GMT
Server: Apache/2.0.45 (Unix) mod_ssl/2.0.45 OpenSSL/0.9.7c PHP/4.3.4
X-Powered-By: PHP/4.3.4
X-Accelerated-By: PHPA/1.3.3r2
Location: ./redirect.php
Content-Length: 0
Content-Type: text/html; charset=ISO-8859-1
```

The last two octet has been hidden for security reasons.

The default banner exposes OS and application details

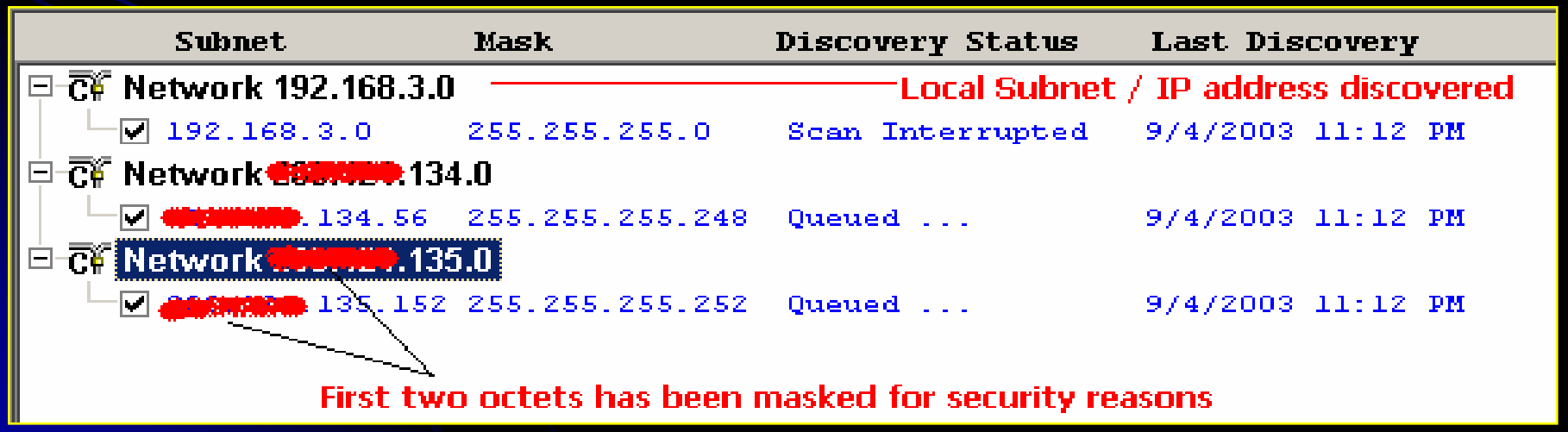
3. Network Surveying / Network Mapping

A network survey serves often as an introduction to the systems to be tested. It is best defined as a combination of data collection, information gathering, and policy control.

Expected Results:

- ❑ Firewall / Routers / IDS Discovery
- ❑ Possible Local Network / Subnet Discovery
- ❑ IP Addresses Range
- ❑ Network Topology Mapping
- ❑ ISP information

Sample Screenshot (Local address of the remote network discovered):



4. Port Scanning & Services Identification

Port scanning is the invasive probing of system ports on the transport and network level. This module is to enumerate live or accessible Internet services as well as penetrating the firewall to find additional live systems.

Expected Results:

- Open, closed or filtered ports
- Services Identification

Sample Screenshot (NMAP port scan output):

```
(The 1649 ports scanned but not shown below are in state: closed)
PORT      STATE SERVICE
21/tcp    open  ftp
23/tcp    open  telnet
25/tcp    open  smtp
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
443/tcp   open  https
445/tcp   open  microsoft-ds
1025/tcp  open  NFS-or-IIS
1026/tcp  open  LSA-or-nterm
Device type: general purpose
Running: Microsoft Windows 95/98/ME|NT/2K/XP
OS details: Microsoft Windows Millennium Edition (Me), Windows 200
 or Advanced Server, or Windows XP
```

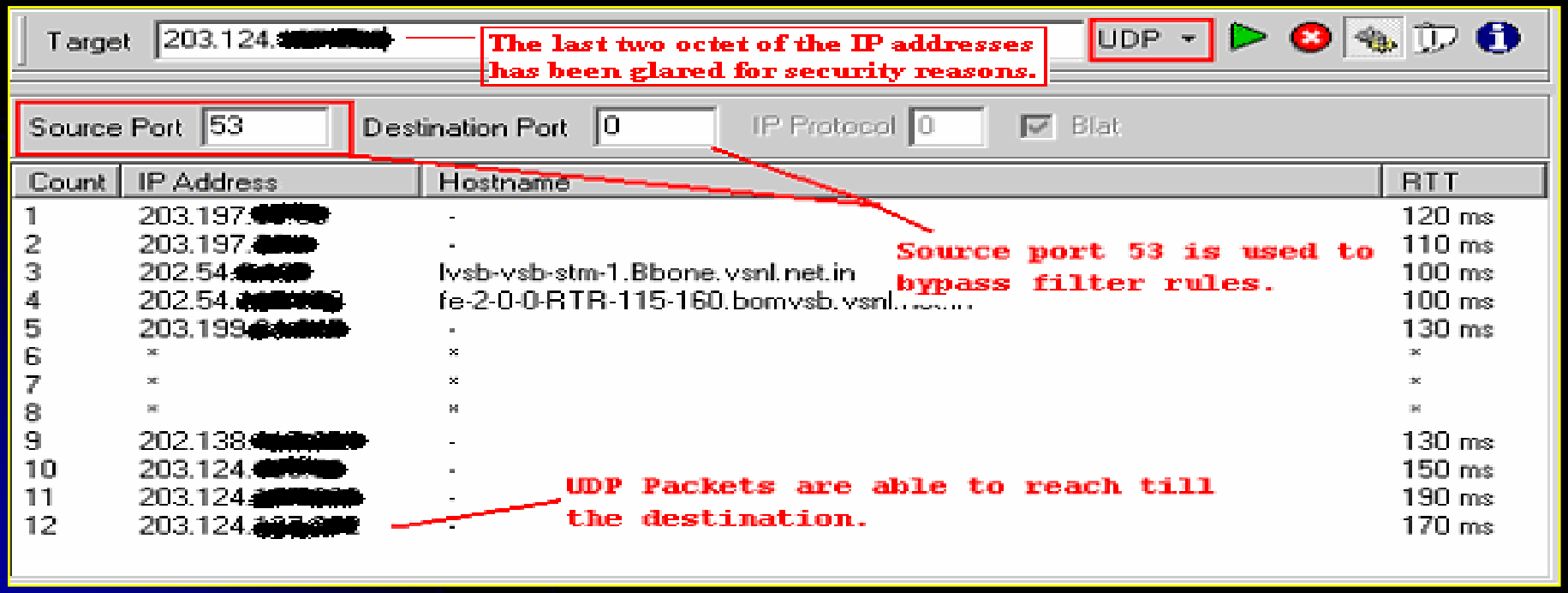
5. Evading Firewall Rules

In this phase, firewall evasion techniques are used to bypass firewall rules. This can further help in port scanning, remote host detection and remote network discovery.

Expected Results:

- Mapping of firewall configuration rules
- Partial Access to devices behind the firewall

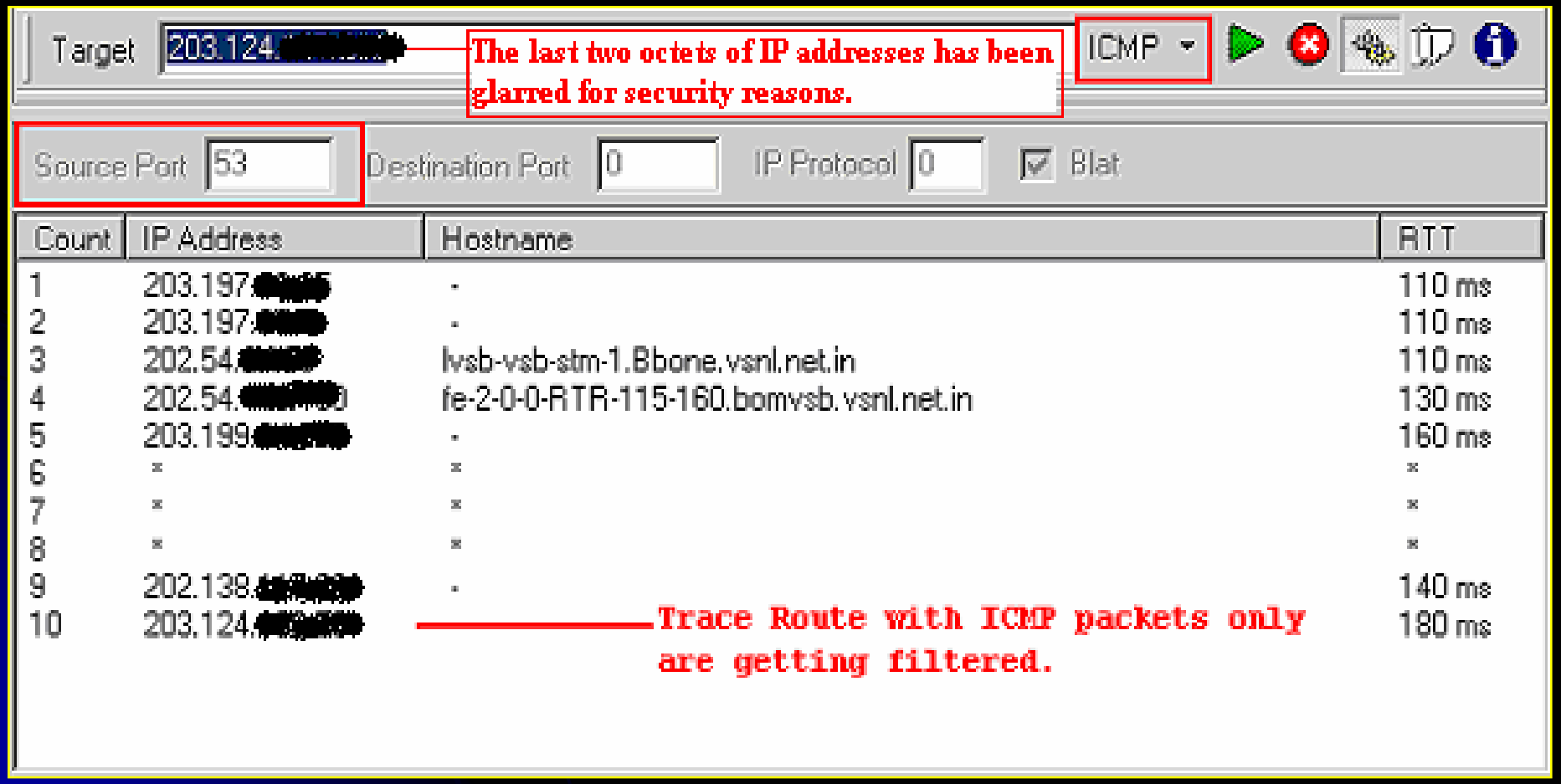
Sample Screenshot 5.a: (Trace Route using UDP packets)



5. Evading Firewall Rules (Contd…)

It is clear for the two screenshots (Screenshot 5.a & 5.b) that the packet filtering device (i.e. Firewall / Router) is not configured to block UDP packets.

Sample Screenshot 5.b: (Trace Route using ICMP packets)



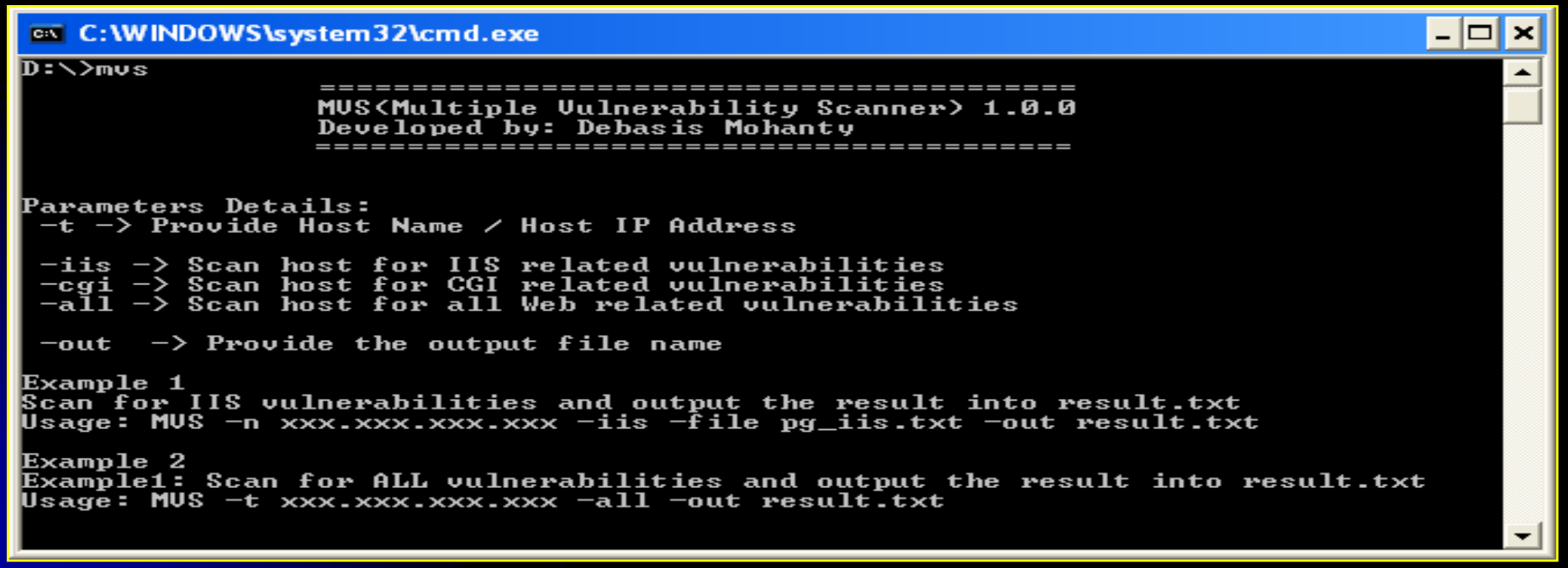
6. Automated Vulnerability Scanning (Contd…)

The focus of this module is identifying, understanding, and verifying the weaknesses, misconfigurations and vulnerabilities associated with remote host. The scanning is done using automated tools or scripts to make the process faster.

Expected Results:

- List of vulnerabilities associated with each remote services
- List of possible denial of service vulnerabilities
- Possible misconfiguration on the remote server

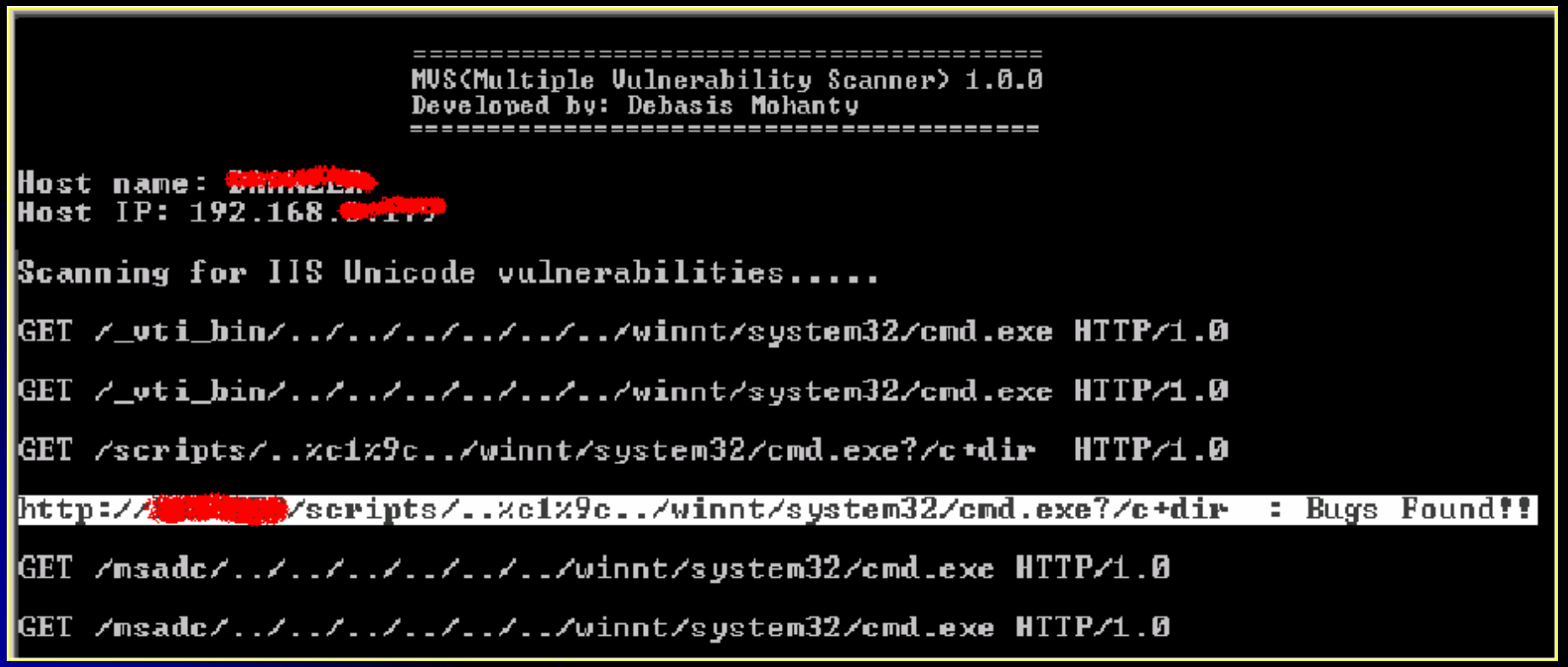
Sample Screenshot 6.a:



6. Automated Vulnerability Scanning (Contd…)

MVS is an automated Internet Vulnerability Scanner (view Screenshot) which can scans for web based vulnerabilities (Ex: CGI/IIS Unicode) associated with a remote host running a web server. The scanner displayed, shows that the target host is vulnerable to IIS Unicode. The vulnerable string has been highlighted in the screenshot 6.b.

Sample Screenshot 6.b:



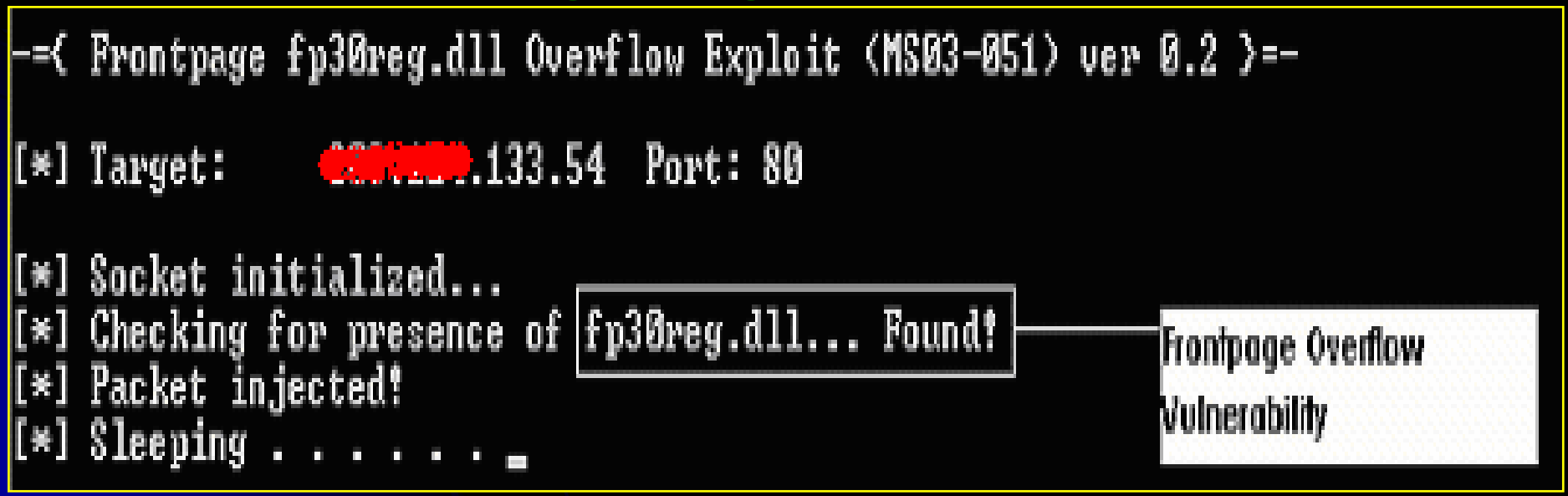
7. Exploiting Services For Known Vulnerabilities

This is the most important phase of penetration testing. Here the weaknesses found in the remote services are exploited using openly available exploits or self developed or customized exploits.

Expected Results:

- Gaining Access to the system
- Retrieving hidden information
- Domain Hijacking
- Spamming Mail Servers

Sample Screenshot (FrontPage fp30reg.dll Overflow Exploit):



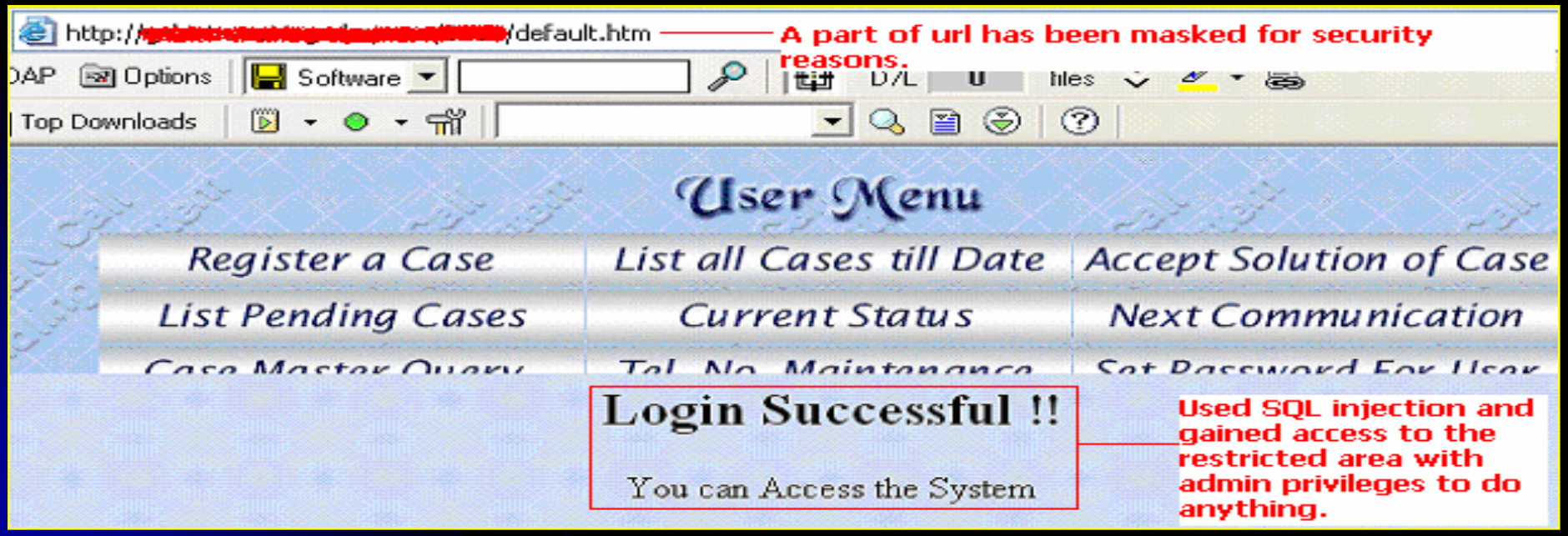
7. Exploiting Services For Known Vulnerabilities (Contd...)

Here the web application flaws are exploited to gain access to restricted information. The Web-Based authentication is exploited by using XSS (Cross-Site Scripting) or SQL injection or MITM (Man-in-the-middle) attacks etc...

Expected Results:

- Access to restricted / confidential information
- Control over web configuration
- Can also leads to gaining access over other servers

Sample Screenshot (SQL injection used for gaining access to admin page):



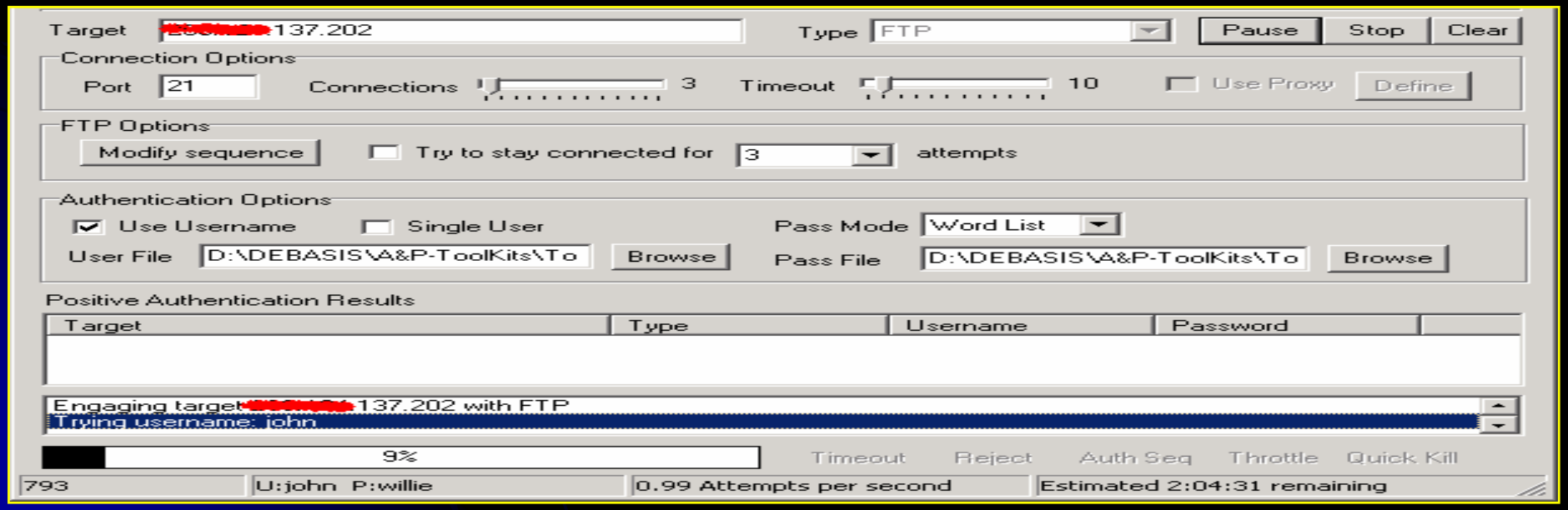
8. Password Cracking or Brute Forcing

Password cracking is the process of validating password strength through the use of automated password recovery tools that expose either the application of weak cryptographic algorithms, incorrect implementation of cryptographic algorithms, or weak passwords due to human factors.

Expected Results:

- List of user login IDs or passwords
- List of authentication PINs or Password

Sample Screenshot (Brute Forcing using Brutus):



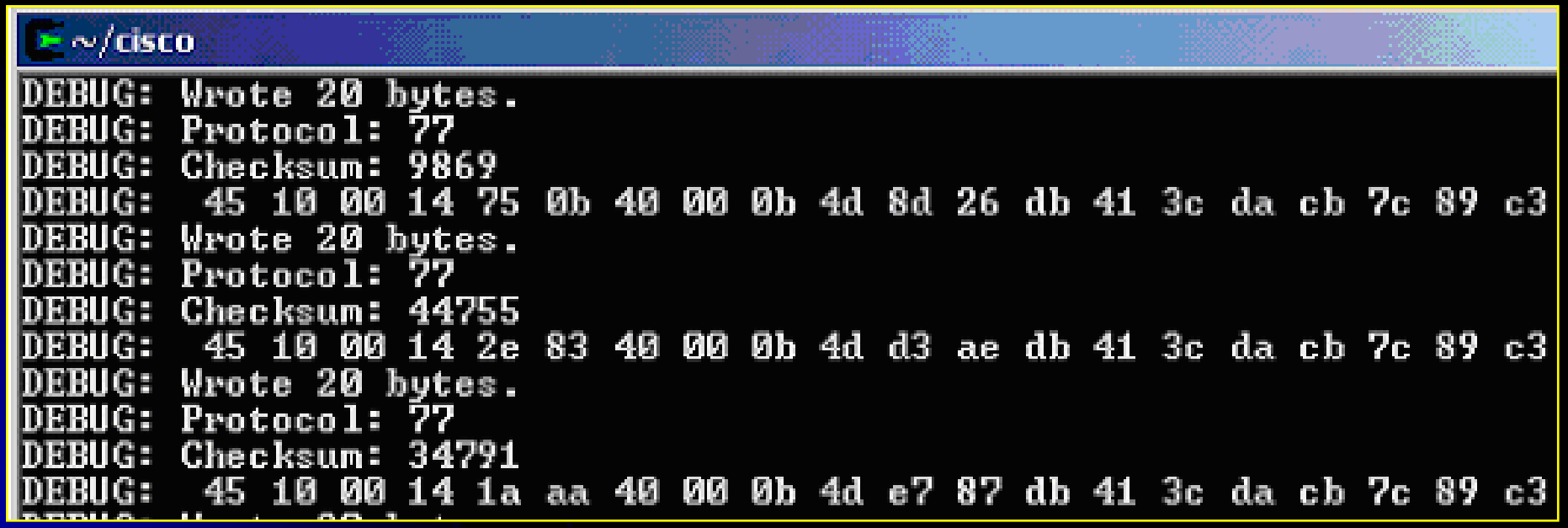
9. Denial of Service (DoS) Testing

Denial of Service (DoS) is a situation where the applications or services running over the remote system stops functioning and prevents authenticated network users or devices to access it.

Expected Results:

- Disruption of Services
- List of other possible DoS vulnerable associated with the systems
- Sabotage of remote network

Sample Screenshot (DOS attack for CISCO):



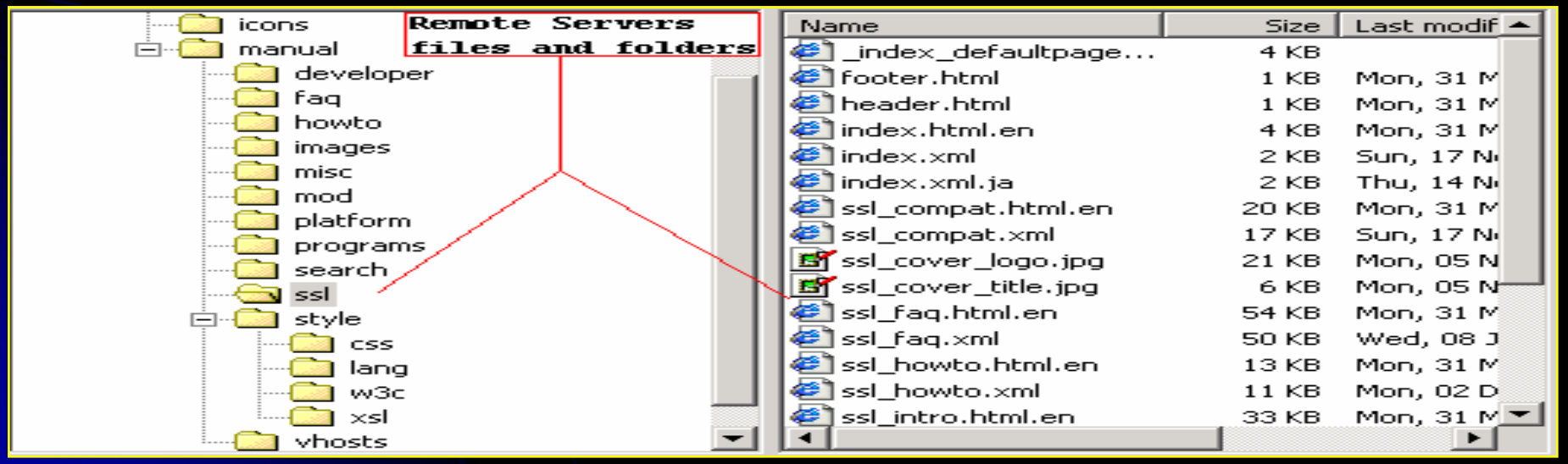
10. Escalation of Privileges

Elevation of Privileges is the type of rights the attacker gains over the remote system. It is the final stage of the remote host hacking where the attacker gains complete control over the remote system.

Expected Results:

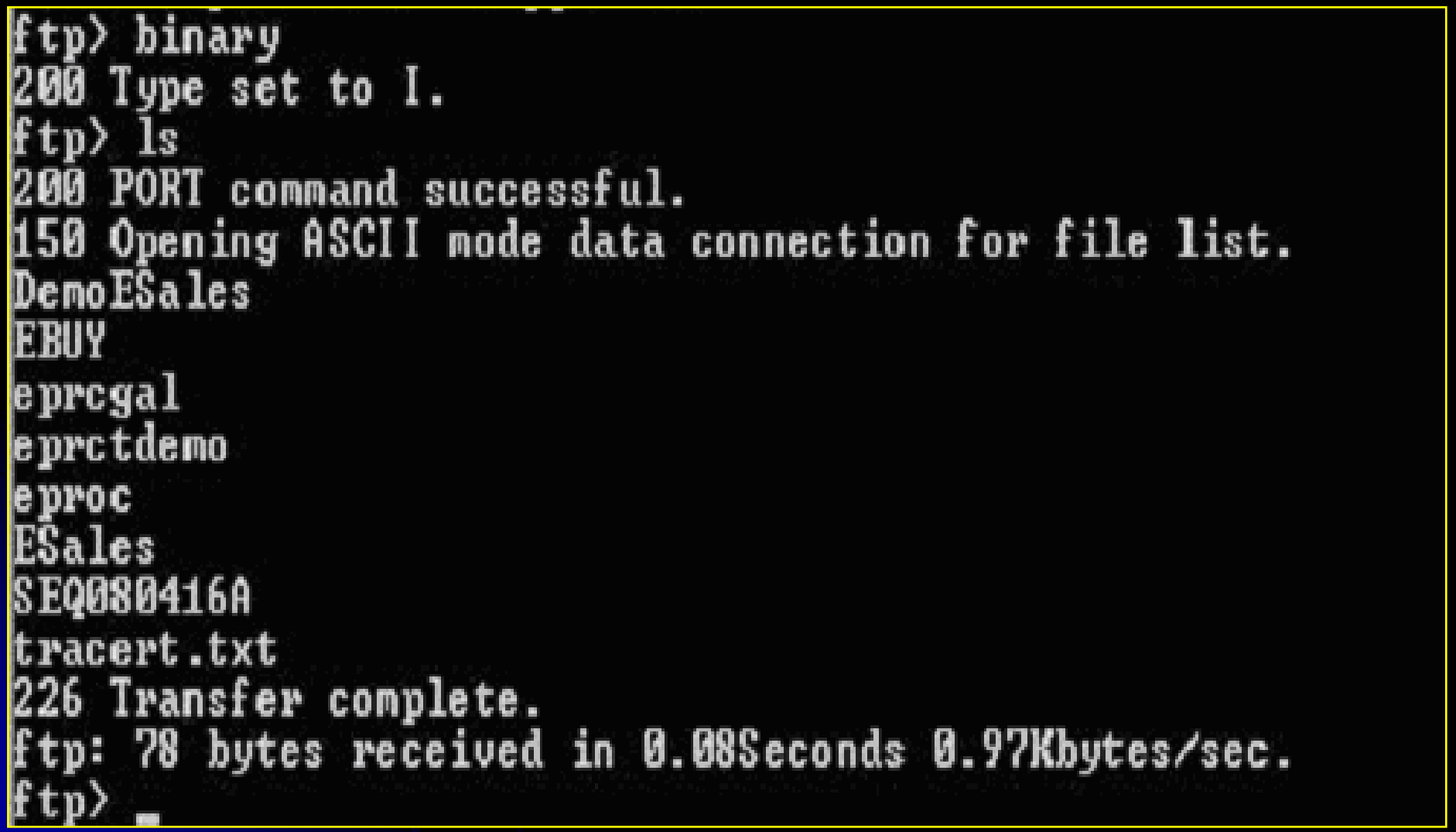
- Gain administrator / super user rights
- Gain privilege to retrieve or modify confidential data
- Gain control over server configuration
- Gain Control over other servers attached to it

Sample Screenshot 10.a:



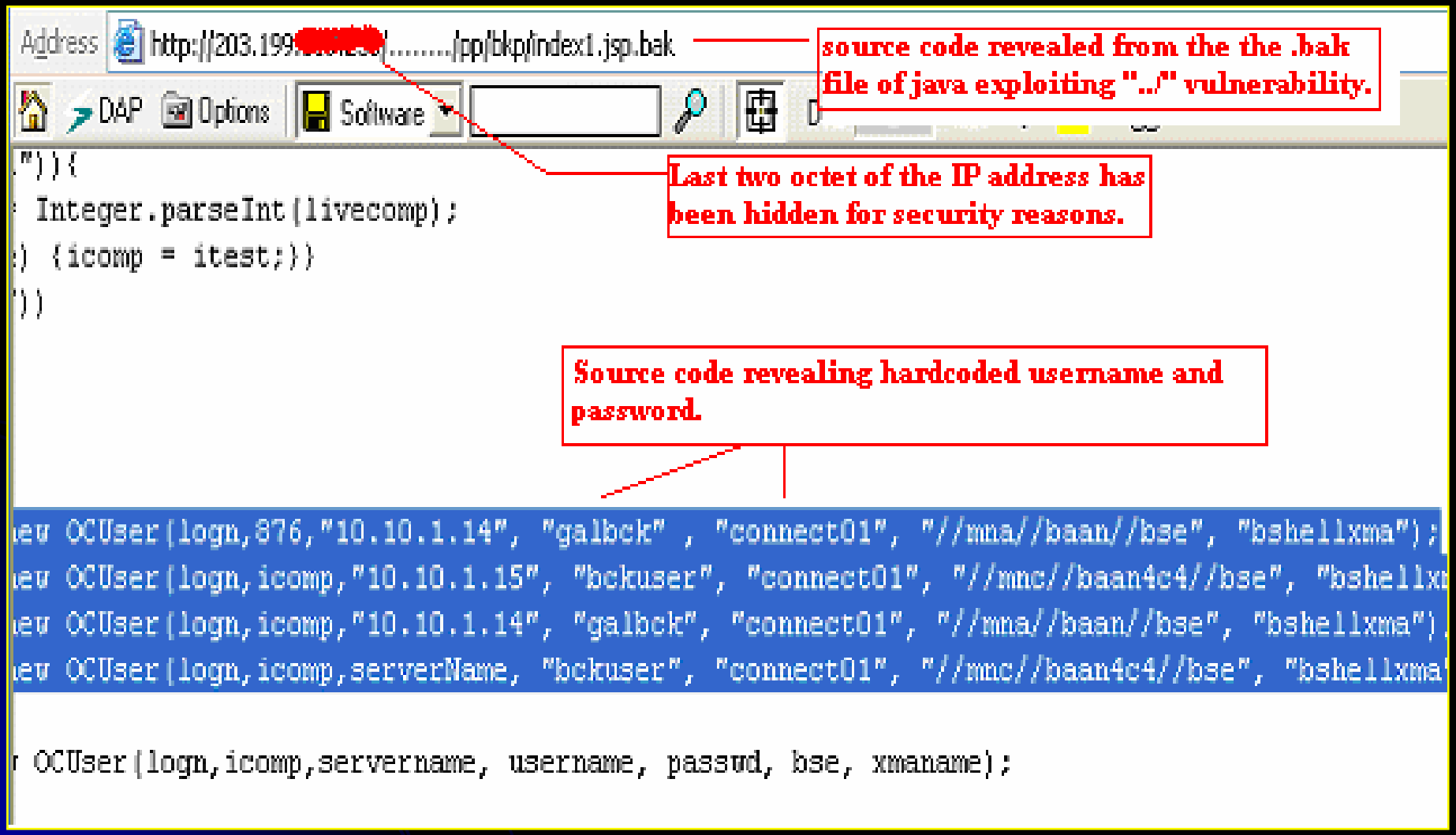
10. Escalation of Privileges (Contd…)

Sample Screenshot 10.b:

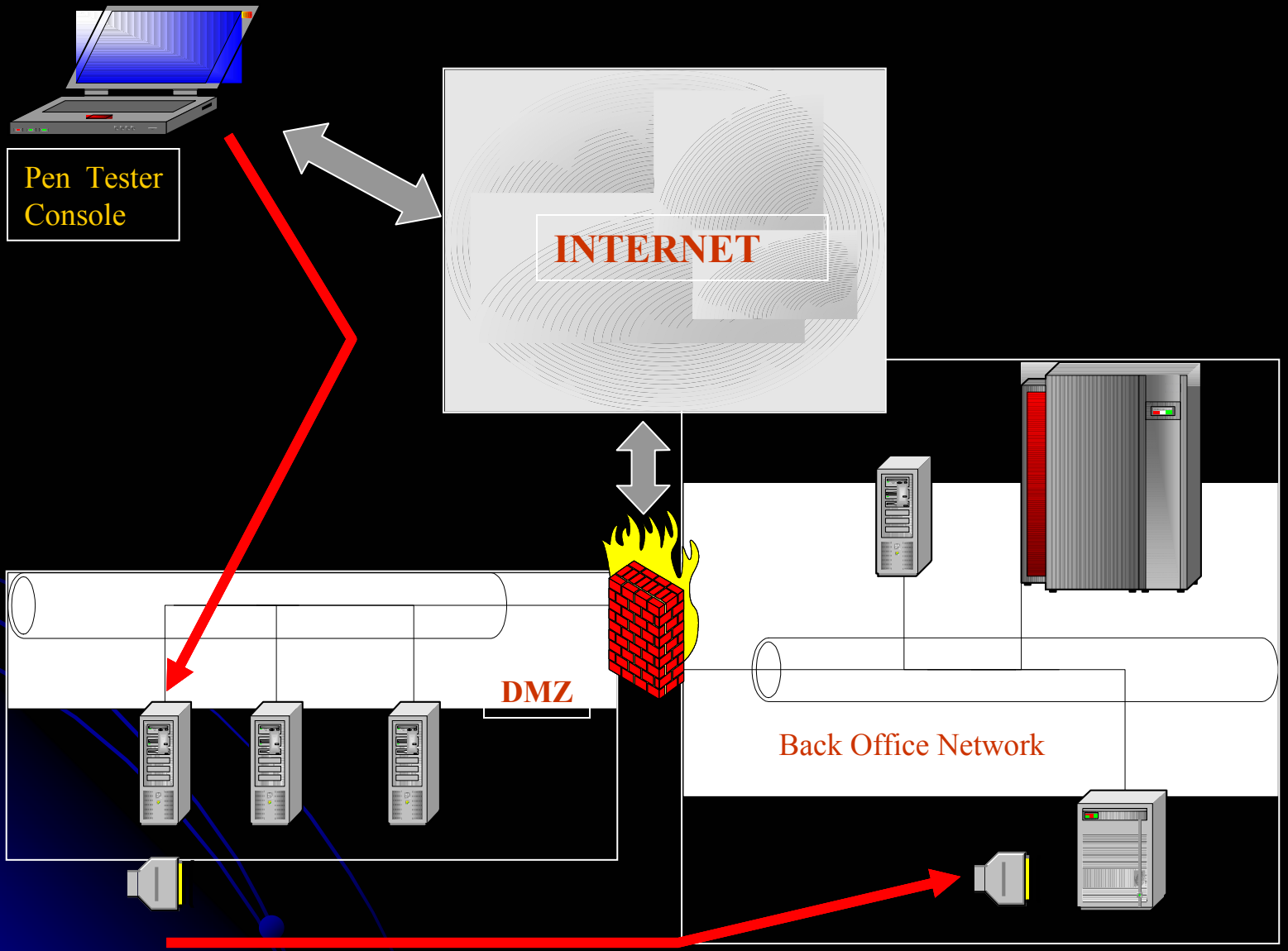


10. Escalation of Privileges (Contd…)

Sample Screenshot 10.c:



11. Final Impact on Successful Escalation of Privileges



Summarized Expected Results

- Domain Registration Information, Email IDs, and IP Addresses Range
- Remote OS Type, Web-Server information
- Firewall / Routers / IDS Discovery
- Mapping Firewall / Network Filters rules by various evasion techniques
- Possible Local Network Discovery / Network Mapping
- Open, closed or filtered ports
- Services Identification
- List of vulnerabilities associated with each remote services
- List of possible denial of service vulnerabilities
- Services Banners and possible misconfiguration on the remote server
- Gaining access to restricted / confidential information
- Domain hijacking and spamming mail servers
- Gaining control over remote system configuration
- Gaining access to other servers attached to main server
- Cracking password files and retrieving list of login IDs with passwords
- Gaining administrator / super user rights
- Retrieve or Modify Confidential data
- Causing unavailability of service (Only for DoS attacks)

Contents of a Penetration Testing Report

Executive Summary

- ✓ Briefing on the type of test performed
- ✓ A pie graph displaying the vulnerabilities in terms of percentage of high, low & medium

Risk Matrix

- ✓ Quantifying the vulnerabilities and showing the high, low & medium in a tabular format
- ✓ Giving a brief of the vulnerabilities found

❑ Proof of Concepts (POC)

- ✓ Giving a detail description with the screenshots and logs of the vulnerabilities found and exploited.

❑ Remedies and Workarounds

- ✓ Providing customised remedies and workarounds for the vulnerabilities found

❑ Best practices

- ✓ Suggesting best practices for the configurations for the device or services

❑ Final Summary

- ✓ Must contain a brief on the overall vulnerability factor found for the remote device

Few List Of Tools Used For Penetration Testing

- **Network Discovery & Information Gathering Tools**
  TraceRoute, MIB Walk, Firewalking, nslookup & dig techniques & Solarwinds Network Discovery, TraceProto, Trout, Sam Spade

- **OS Fingerprinting Tools**
  Nmap, P0F, XProbe2, SuperScan

- **Port Scanning & Services Identification Tools**
  Nmap, MegaPing, MingSweeper, SuperScan, THC-Amap

- **Firewall Bypassing Tools**
  Firewalking, HPING(1/2/3), MPTraceRoute, Firewall Tester, SYN-STEALTH techniques and other open source tools

- **Automated Vulnerability Scanning Tools**
  Nessus, eEye Retina, GFI LanGaurd, ISS Scanner, Shadow Security Scanner, HTTP Scanners (CGI,PHP and ASP etc), SSL Scanners, Nikto, Whisker and Open Source Tools etc.

Few List Of Tools Used For Penetration Testing (Contd…)

- **Automated Exploiting Tools**
  Metasploit Framework, Core Impact, Canvas

- **Password Cracking / Brute Forcing Tools**
  John the ripper, L0phtcrack, MD5 Crack, SQL Bruteforce, CISCO Password decryptor, SolarWinds Network Password Decryptor, Cain & Abel, THC-Hydra, BRUTUS etc.

- **Sniffers**
  Ethereal, Ettercap, Dsniff, Hunt

- **Denial of Service (DoS) Tools**
  HPING & openly available DoS exploits (Zero-Day and Others)

- **Exploits Used**
  Both customized and publicly available exploits (Zero-Days and Others) and sometimes exploits are coded depending upon the requirements

- **Tools Kit**
  Knoppix-STD, PHLAK, Auditor Security Collection etc.

Zero-Days

Zero-Day Exploits:
A zero-day exploit is one that exploits an unknown vulnerability or a know vulnerability on day one when the vulnerability becomes publicly known.

Categories of Exploits:
- Remote Exploit
- Local Exploit

Zero-Day Vulnerability:
A zero-day vulnerability is one which is publicly unknown but only know to the attacker who discovered it.

Understanding Buffer Overflows

Sample C Program (BOTest.c)

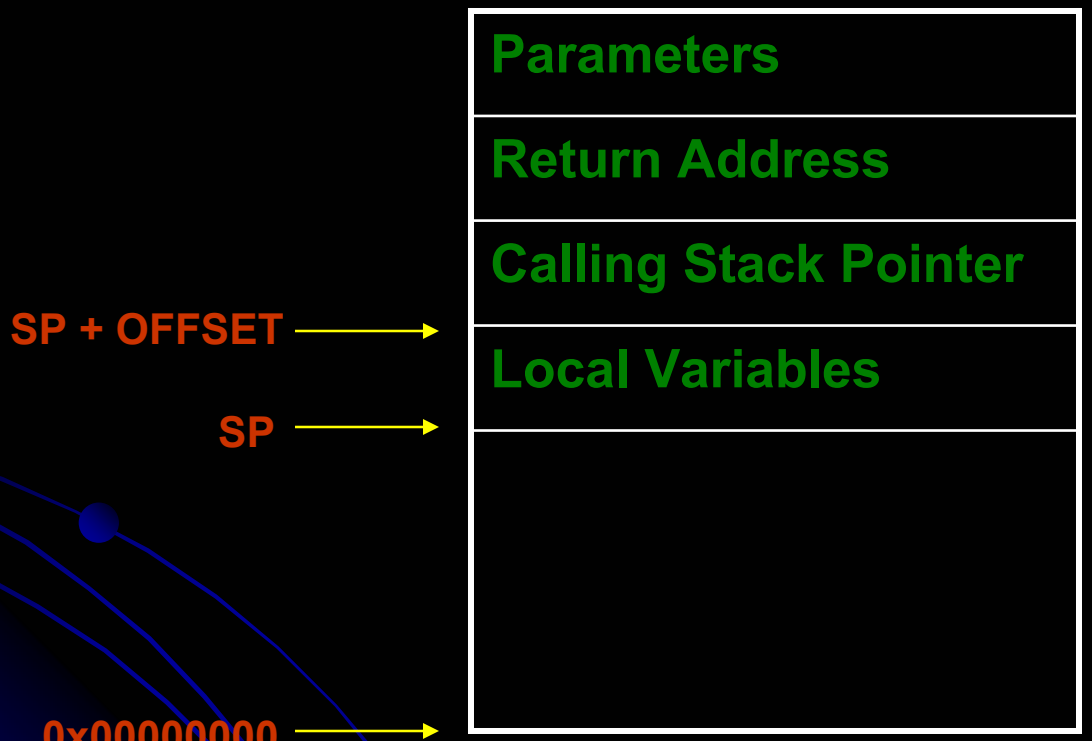
```
#include <stdio.h> // The Sample Vulnerable 'C' Program

void vulnerable_func( char *pszName )
{
  char szBuffer[100];
  strcpy( szBuffer, pszName );
  printf("Name is %s\n", szBuffer);
}

int main(void)
{
  char szBuff[5000];
  read(0, szBuff, 5000);
 vulnerable_func(szBuff);
}
```

Understanding Buffer Overflows (Contd…)

A Stack Frame Details



Understanding Buffer Overflows (Contd…)

Overwriting The Return Address

Original Stack before Buffer Overflow

Stack after Buffer Overflow

Few Good Security Links To Refer

www.securityfocus.com
www.secunia.com
www.infosyssec.com
www.sans.org
www.insecure.org
www.packetstormsecurity.org
www.zone-h.org
www.cnhonker.com
www.phrack.org
www.astalavista.com
www.blackhat.com
www.defcon.org
www.osvdb.org
www.ntbugtraq.com
www.antiserver.it
www.k-otik.com
www.securiteam.com

HAPPY HACKING

THANK YOU

Debasis Mohanty

www.hackingspirits.com

Email Your Comments @

debasis_mty@yahoo.com or
debasis_mohanty@msn.com