**Penetration Test Report – Ice Host**

**Client:** TryHackMe (Lab Environment)  
**Date of Engagement:** 8/28/2028  
**Consultant:** Justin Alexander  
**Scope:** One Windows host ("Ice") within the TryHackMe lab network

**1. Executive Summary**

A penetration test was performed against the **Ice host** to identify vulnerabilities, assess potential impact, and demonstrate possible exploitation. The test revealed a **critical remote code execution vulnerability** in the Icecast service running on TCP port 8000. Exploitation of this flaw allowed unauthenticated remote attackers to gain full **SYSTEM-level control** of the host.

This compromise enabled retrieval of local password hashes and the ability to create persistent administrative accounts, effectively granting long-term unauthorized access.

The vulnerability is **critical** in severity and should be remediated immediately by upgrading or removing the vulnerable Icecast service.

**2. Assessment Goals**

* Identify vulnerabilities in exposed services
* Attempt exploitation to validate findings
* Determine post-exploitation impact (privilege escalation, credential access)
* Provide actionable remediation recommendations

**3. Methodology**

Testing followed the **SANS Penetration Testing Execution Standard (PTES)** and included:

1. **Reconnaissance:** Full-port network scanning and service identification
2. **Enumeration:** Banner grabbing, web service inspection, and vulnerability research
3. **Exploitation:** Leveraging known exploits against vulnerable services
4. **Post-Exploitation:** Privilege escalation, credential dumping, and persistence testing
5. **Reporting:** Documentation of findings and recommendations

**4. Findings**

**4.1 Critical Vulnerability – Icecast Remote Buffer Overflow**

* **Service:** Icecast 2.0.1 (TCP 8000)
* **Vulnerability:** Remote Buffer Overflow (CVE-2004-1561)
* **Impact:** Remote, unauthenticated attackers can execute arbitrary code with SYSTEM privileges.
* **Evidence:** Successful exploitation via Metasploit module exploit/windows/http/icecast\_header resulted in a Meterpreter session as **NT AUTHORITY\SYSTEM**.

**Risk Rating:** Critical (CVSS 9.8)

**4.2 Credential Exposure – Local Account Hashes**

* **Service Impacted:** Windows Local Security Authority (LSA)
* **Vulnerability:** Dumped password hashes from the SAM database using Meterpreter hashdump.
* **Impact:** Adversaries can crack hashes offline, recover plaintext passwords, and leverage credential reuse across systems.

**Risk Rating:** High

**4.3 Insecure Service Exposure – Remote Desktop Protocol (RDP) Open to All**

* **Service:** TCP 3389 (RDP)
* **Impact:** With SYSTEM compromise, attackers could enable RDP and maintain persistent GUI access. Exposing RDP directly to the internet increases attack surface.

**Risk Rating:** Medium

**5. Exploitation Path**

1. Performed Nmap scan → Identified Icecast service on TCP 8000.
2. Verified service and located public exploit (Metasploit).
3. Executed buffer overflow exploit → Gained SYSTEM-level shell.
4. Dumped local password hashes (hashdump).
5. Demonstrated ability to create persistence (net user backdoor /add).

**6. Recommendations**

**6.1 Patch and Upgrade**

* Immediately remove or upgrade vulnerable **Icecast 2.0.1** to a supported, patched version.
* Regularly update third-party software to mitigate known exploits.

**6.2 Credential Hardening**

* Invalidate compromised accounts and reset passwords.
* Implement **password complexity and rotation policies**.
* Deploy **LSA protection** to prevent direct hash dumping.

**6.3 Network Hardening**

* Restrict **RDP access** to internal subnets or VPN only.
* Apply firewall rules to limit exposure of non-essential services.
* Monitor authentication logs for unusual activity.

**7. Conclusion**

The Ice host is vulnerable to a critical remote code execution flaw, enabling full system compromise. This issue underscores the need for proactive patch management, credential security, and network service hardening. Implementing the recommendations above will significantly reduce the risk of similar compromises.