**Red Team:**

**Summary of Operations**

By Dean Baberowski

## 

## 

## 

## 

## 

## 

## 

**Table of Contents**

[Exposed Services](#_yl5568s91c07) **3**

[Critical Vulnerabilities](#_apze8nm64uib) **5**

[Exploitation](#_nrp2caayeo5q) **6**

## 

### 

### 

### 

### 

### 

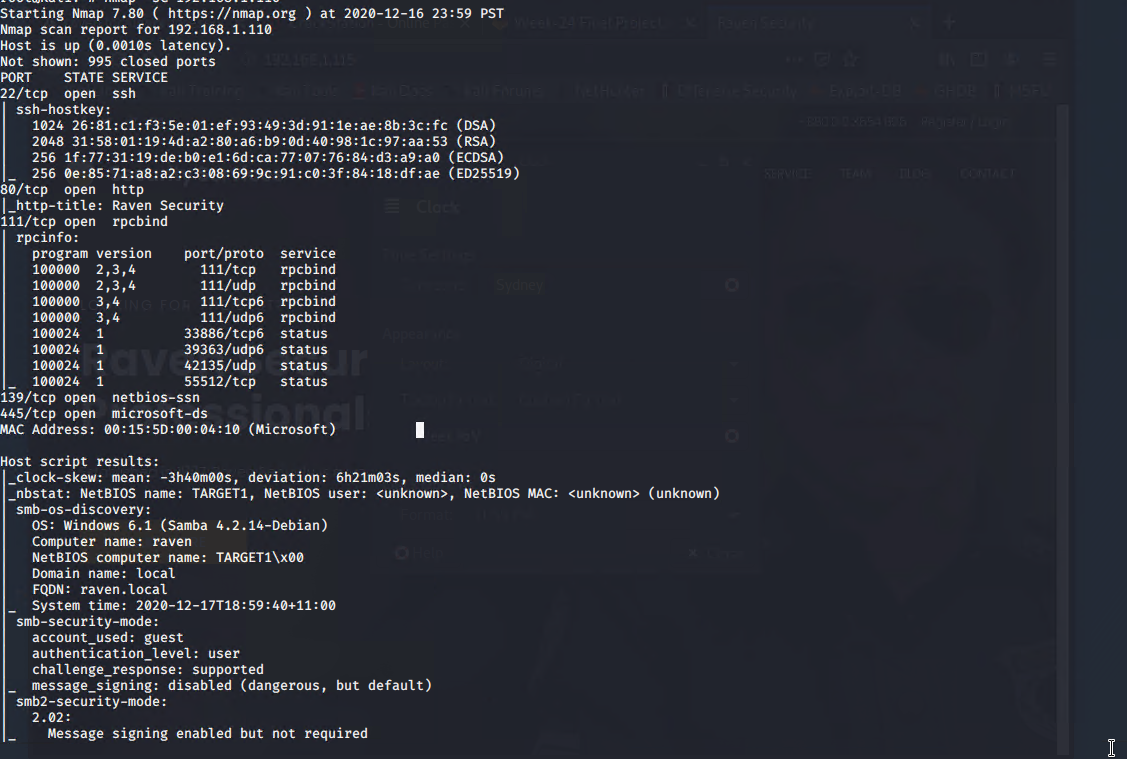
### 

### 

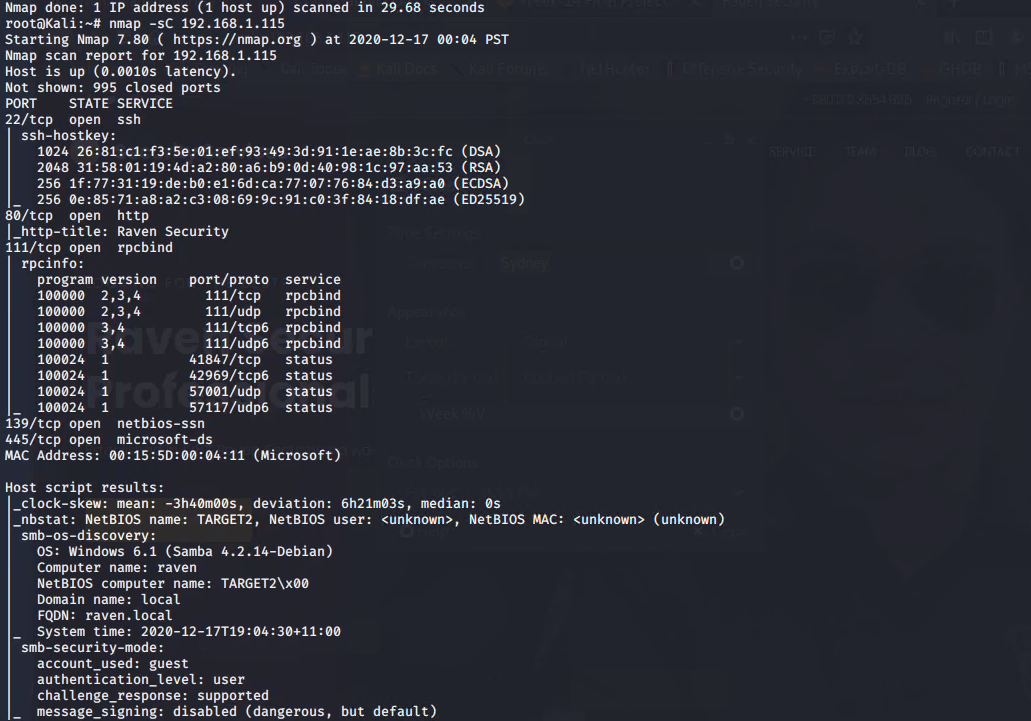
# **Exposed Services**

Nmap scan results for each machine reveal the below services and OS details:

$ nmap -sC 192.168.1.110



$ nmap -sC 192.168.1.115



This scan identifies the services below as potential points of entry:

**Target 1**

1. Port 22/SSH login
2. HTTP port 80
3. RPCbind port 111
4. Port 130 netbios

**Target 2**

1. Multiple SQL Injection wordpress
2. Port 22 SSH
3. HTTP port 80
4. Port 139 netbios-ssn
5. Port 445 microsoft-ds

# **Critical Vulnerabilities**

The following vulnerabilities were identified on each target:

**Target 1**

1. Weak passwords CWE-522 high -transmits or stores authentication credentials, but it uses an insecure method that is susceptible to unauthorized interception and/or retrieval.
2. SSH Vulnerability CVE-1999-0398 - high severity In some instances of SSH 1.2.27 and 2.0.11 on Linux systems, SSH will allow users with expired accounts to login.
3. SQL Injection - CWE 564 -Medium Allows attackers to inject an SQL database.
4. RPC bind #10000 CVE 2017-8779 Medium Allows remote attackers to cause a denial of service through access of the windows RPC protocol
5. Samba smbd vulnerability CVE-2019-3824 Medium- A flaw found in the LDAP server
6. Wordpress Vulnerability CVE-2009-3891 Low

**Target 2**

1. SSH CVE-1999-0398 - high severity In some instances of SSH 1.2.27 and 2.0.11 on Linux systems, SSH will allow users with expired accounts to login.
2. SQL Injection - CWE 564 -Medium Allows attackers to inject an SQL database.
3. Wordpress Vulnerability CVE-2009-3891 Low



Screenshot of scan results

# **Business Risks and Mitigation strategies**

Each of the vulnerabilities listed above can expose a business to a considerable amount of risk if the vulnerabilities are not ratified.

The risk for the business is that it could be financial, a loss of reputation and lack of confidence from customers in the business.

The following lists the mitigations and business risk of each vulnerability.

**Mitigation Strategies for Weak passwords**

1. Set stronger password requirements in the password policy of the business
2. Set a minimum length of the password
3. Make sure the password policy states that it has to be a mix of letters, numbers and special characters
4. Set expirations for the password
5. Deleted inactive users and their passwords.

If a system has a weak password this could expose the business risk from a technical and non technical point of view.

**Mitigation Strategies for SSH Vulnerabilities**

1. Implement a firewall that blocks traffic from ports
2. Use SSH hardening practices to strengthen the ability of an attacker to gain access to a system
3. Patch the operating system with 48 hours don’t leave it unsupported

**Business Risks and Technical View**

If the SSH vulnerability is left unattended, it could expose a business to a security risk as it means that anyone can access the system.

From a technical point of view it means that the system can be left exposed to an attacker that could comprise the system.

**Mitigation Strategies for SQL Injection**

Mitigation strategy for SQL Injection is to sanitize the inputs on the forms by having validations on the websites forms.

The business risk is that if an SQL injection attack happens it could lead to customer data stolen or financial information leaked. This could cause a lack of confidence in the business from the customers. From a technical view the form and website could be compromised and data could be leaked.

**Mitigation Strategy for RPCbind**

1. Update the system to the latest update
2. Update antivirus definitions
3. Disable Port access to RPCbind

**Business Risks and Technical View**

If this port is left open there is a potential for a DDOS ( Distributed Denial of Service) attack to be exploited by an attacker. From a business risk point of view it could lead to systems being slow or down for extended periods of time.

**Mitigation Strategy for Samba smbd vulnerability**

1. Update the Samba service to the latest version as patches have been released for this

**Business Risks and Technical View**

If the samba service is left unpatch an attacker could launch a distributed denial of service attack leading to potential loss of business for a company.

**Mitigation Strategy for Wordpress Vulnerability**

1. Update to the latest version of wordpress

**Business Risk and Technical view**

Although this vulnerability is low it is still important to pay attention to as if it is left unattended the attacker could modify files to a wordpress site . For the business it could mean that parts of the site become inaccessible as there are missing files.

# 

# 

# **Exploitation**

The Red Team was able to penetrate both Target 1 and Target 2 and retrieve the following confidential data:

**Target 1**

* flag1.txt: TODO: Insert flag1.txt hash value.

Flag1.txt 72eace3b06b6871afcf313e371b9ccf5

* Exploit Used: Used the SSH exploit
* flag2.txt



* Exploit Used
  + Used the sql injection
  + SQL map

**Target 2**

* flag1.txt: 6ed61d4b80bb0f81937b32418e98adca
* Exploit Used
  + Shell exploit
  + ./ exploit.sh
* flag2.txt:
* 
* Exploit Used
  + Used a backdoor into the target system
  + Ran the shell script exploit.sh using ./exploit.sh