CHIPPY'S CORNER RED TEAM REPORT

AGENDA

EXECUTIVE SUMMARY

RULES OF ENGAGEMENT

ATTACK NARRATIVE

OBSERVATIONS

RECOMMENDATIONS

EXECUTIVE SUMMARY

PURPOSE

To test the security posture of Chippy's Corner staff subnet, the organization hired Bah Red Labs to perform a red team engagement.

The red team was tasked with simulating a full-scale adversarial attack on the staff network. The red team will utilize the MITRE ATT&CK Framework and its tactics, techniques, and procedures

MAIN GOALS

Gain administrator access on the vulnerable web server with remote code execution.

Compromise the workstation belonging to the boss to exfiltrate sensitive employee information.

Maintain persistence between both compromised endpoints.

RULES OF ENGAGEMENT

SCOPE AND SCENARIO

This simulated attack will occur on the staff subnet with the IP range of 192.168.110.100 – 192.168.110.200.

This attack follows an assumed breach model. The team has been provided access in the subnet masking as an “old endpoint” for simulation's sake.

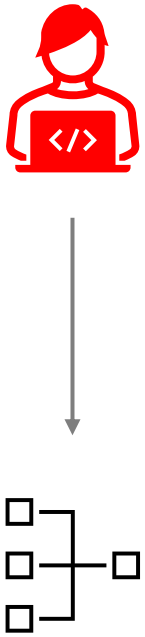
ENGAGEMENT RULES

- Denial of Service attacks are not allowed.
- No persistence outside of engagement window.
- No other subnet should be touched outside of the staff subnet.

ATTACK NARRATIVE

Phase 1: Reconnaissance

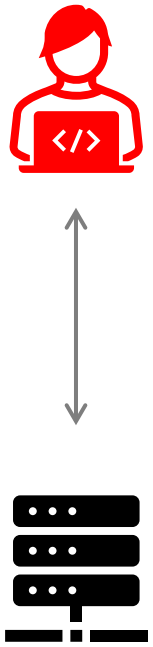
Attacker sends nmap service scan to identify the services ran by the hosts on the network



Notable results included an HTTP Server used to host a vulnerable web application used by the organizations developers and SSH running on an endpoint named “boss”.

Phase 2: Initial Access

After discovering the vulnerable web application, the attacker used command injection to achieve remote command execution with administrator privileges on the web server’s host.

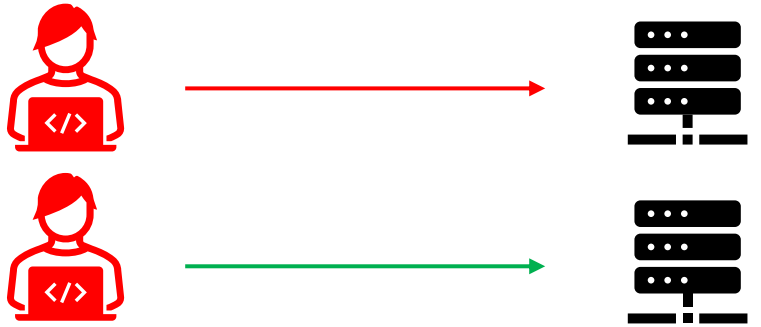


After RCE is established, the attacker uploads a crafted PowerShell TCP reverse shell onto the host and established an inbound connection from the web server’s host to the attacker

Phase 3: Foothold

Attacker uses temporary TCP reverse shell to drop the C2 payload.

Attacker repackages payload as .pyw to avoid EDR and establish persistence by adding payload path to startup registry



Attacker successfully establishes full RCE

Phase 4: Lateral Movement

Attacker utilizes Hydra to brute force identify potential weak credentials in SSH on the target machine.

Brute force found the password of the target machine and remote SSH access has been established

Phase 5: Post-Exploitation

Attacker identifies classified documents regarding employee information on the target machine and exfiltrates the artifacts successfully to their machine

OBSERVATIONS

OBSERVATIONS

Web app developer tool “ping” utility accepted unvalidated user input.

Injected 127.0.0.1; whoami /groups to chain commands.

Provided attacker with initial Remote Code Execution (RCE) foothold.

Allowed the red team to place and run malicious PowerShell scripts that created reverse shell connections.

index.php - Notepad

File Edit Format View Help

```
<!DOCTYPE html>
<html>
<head>

        <title>Ping Tool</title>

</head>

<body>

        <h1>Ping Tool For Developers</h1>

        <form method="GET">
                <input type="text" name="ip" placeholder="Enter IP">
                <input type="submit" value="Ping">
        </form>

        <pre>

        <?php
        if (isset($_GET['ip'])){
                $ip = $_GET['ip'];
                $output = shell_exec("powershell -Command \"ping -n 3 . $ip\"");
                echo $output;
                }

        ?>
        </pre>

</body>


</html>
```

OBSERVATIONS (CONT.)

Defender blocked .exe payload, but Python interpreter was present.

Payload repackaged as .pyw, executed silently (no console window).

Added entry under HKCU\Software\Microsoft\Windows\CurrentVersion\Run to establish startup persistence.

Stealthy persistence across reboots without detection.

OBSERVATIONS (CONT.)

Workstation exposed SSH service unnecessarily.

Hydra brute force cracked weak credentials.

Sensitive business data identified and exfiltrated to attacker system.

RECOMMENDATIONS

SECURE INPUT

Sanitize and validate all user input while block special shell characters such as ;, |, and &.

Restrict PowerShell & Python to signed/trusted scripts only.

Enable input monitoring and logging to alert of any suspicious inputs.

REDUCING THE ATTACK SURFACE

Disable any unnecessary services and software unless they are needed for day-to-day operations.

Run all applications and servers under low-privileged service accounts.

PERSISTENCE AVOIDANCE

Monitor any unauthorized changes to registry run keys.

Apply an application allowlist to block any type of unauthorized binaries and scripts from executing on startup.

AUTHENTICATION RECOMMENDATIONS

Enforce strong, complex password policies that require multifactor authentication.

Apply account lockout after repeated failures on login.