Vulnerability Assessment: Nocturnal.htb - Complete System Compromise

Document Type: Vulnerability Assessment Report
Target: nocturnal.htb
Assessment Date: July 2025
Severity: CRITICAL
Status: Complete System Compromise

Executive Summary

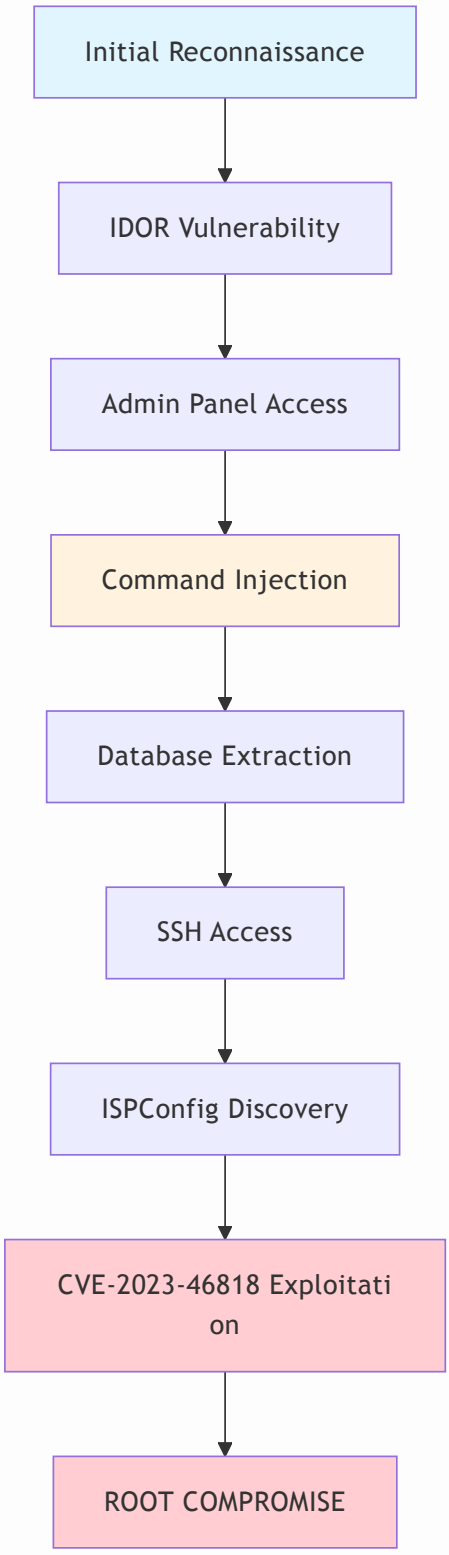
A critical security assessment of nocturnal.htb revealed multiple high-severity vulnerabilities that allowed complete system compromise. The attack chain progressed from an **Insecure Direct Object Reference (IDOR)** vulnerability to **command injection**, **credential compromise**, **privilege escalation**, and ultimately **root access** through an ISPConfig CVE exploitation. This represents a complete security failure across multiple layers of the application and infrastructure.

CRITICAL FINDING: Complete system compromise achieved through vulnerability chaining

Scope and Methodology

- **Target**: nocturnal.htb
- **Testing Approach**: Black-box to White-box penetration testing
- **Tools Used**: Nmap, ffuf, Burp Suite, SecLists, Hydra, SQLite3, CrackStation
- **Attack Vector**: Web application → Database extraction → SSH access → Local privilege escalation

Attack Chain Overview



1. **Initial Access**: IDOR vulnerability in file upload system
2. **Lateral Movement**: Admin panel access through compromised credentials
3. **Command Injection**: Backup functionality exploitation
4. **Data Exfiltration**: Database dump extraction
5. **Credential Compromise**: Password cracking and SSH access

6. **Privilege Escalation**: ISPConfig CVE-2023-46818 exploitation
7. **Root Compromise**: Complete system takeover

Detailed Vulnerability Analysis

1. Insecure Direct Object Reference (IDOR) - CRITICAL

> **CVE Reference**: CWE-639 (Authorization Bypass Through User-Controlled Key)
> **Risk Score**: 9.0/10

Affected Endpoint: `/view.php?username=<USERNAME>&file=<FILENAME>`

Description: The file viewing functionality allows unauthorized access to any user's uploaded files through parameter manipulation.

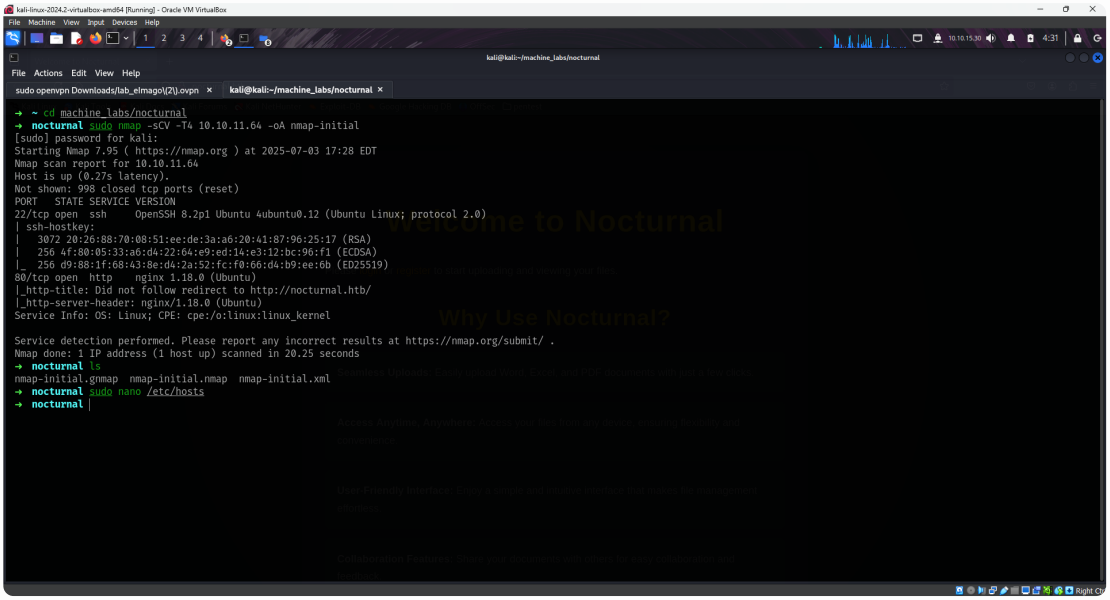
Technical Details:

- No session-based authorization checks
- Predictable file access pattern
- Username enumeration possible
- Direct file system access

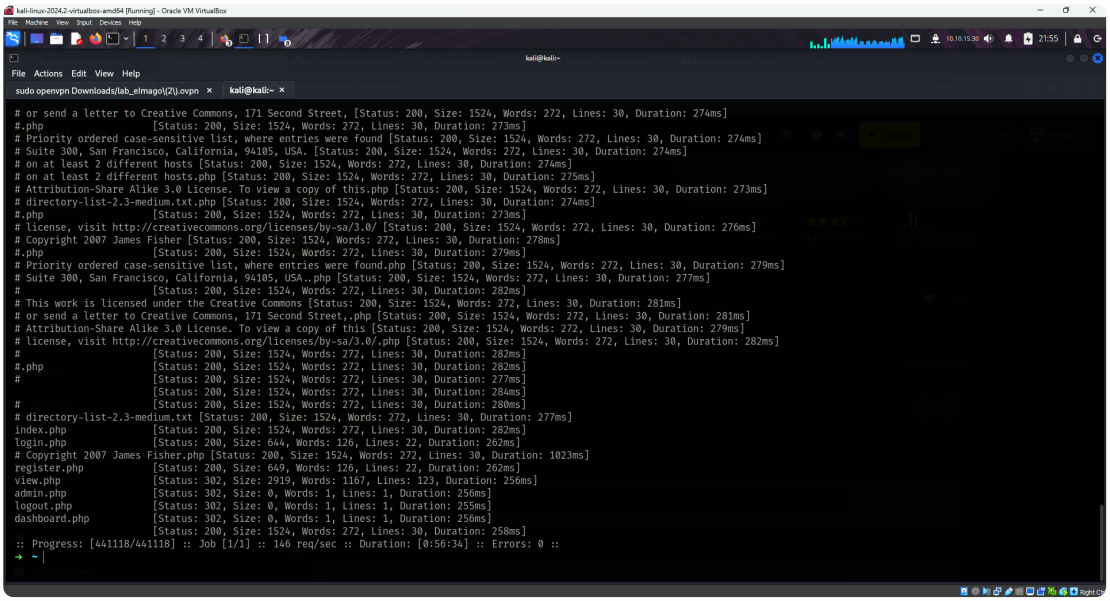
Exploitation Steps:

1. Initial reconnaissance with Nmap revealed SSH (22) and HTTP (80)
2. Directory enumeration with ffuf discovered admin endpoints
3. Legitimate user registration to understand application flow
4. Parameter manipulation in file viewing functionality
5. Username enumeration using SecLists wordlist
6. Successful access to amanda's `privacy.odt` file
7. Discovery of temporary password from IT communications

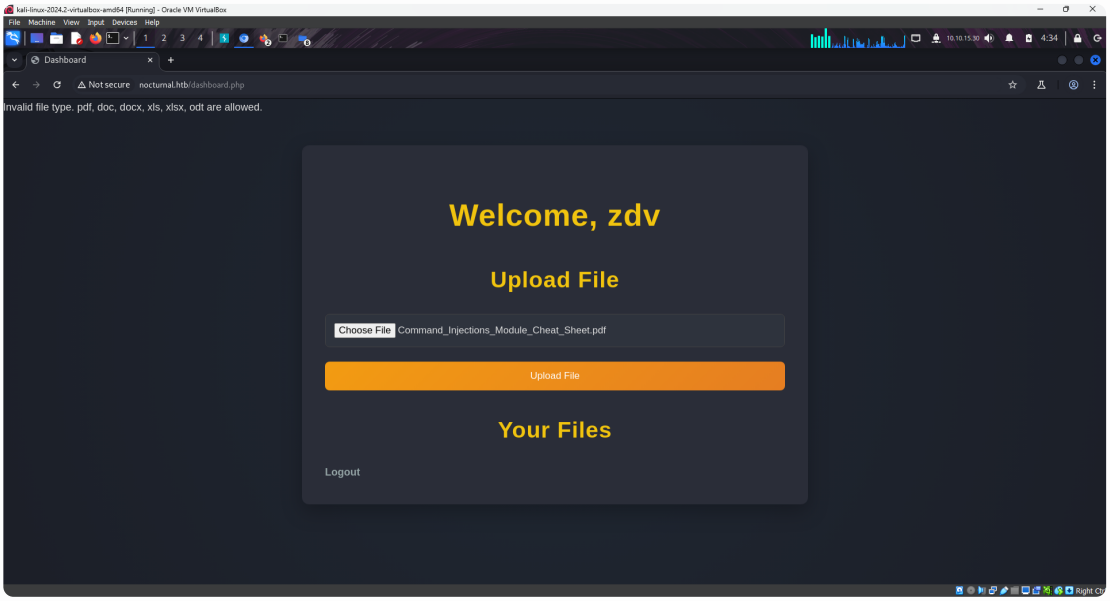
> **Screenshot Requirements:**

- 

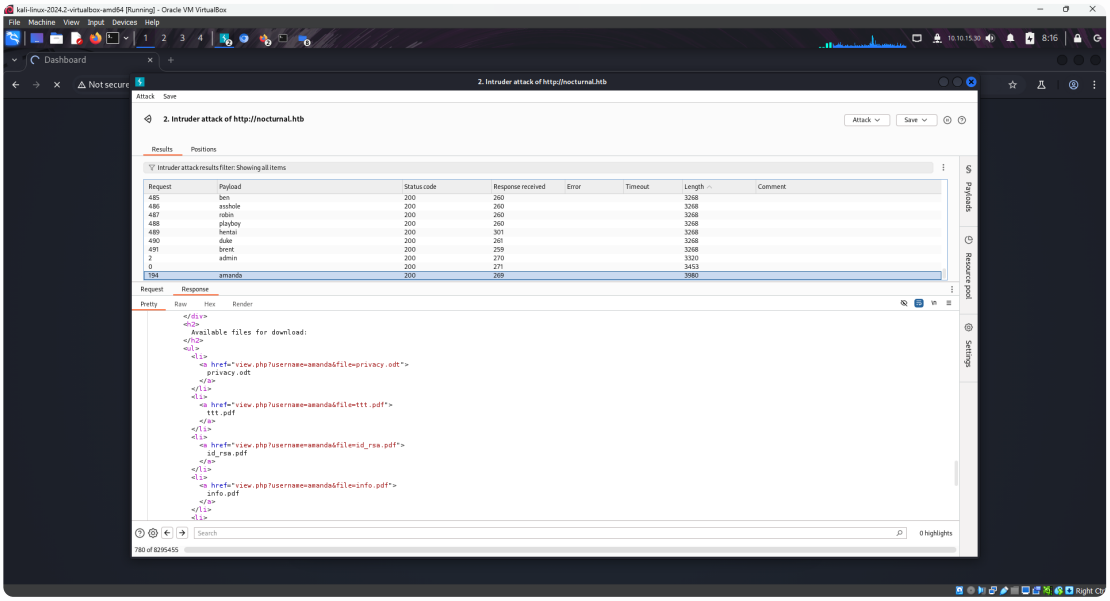
[Screenshot 1: Nmap scan showing open ports]

- 

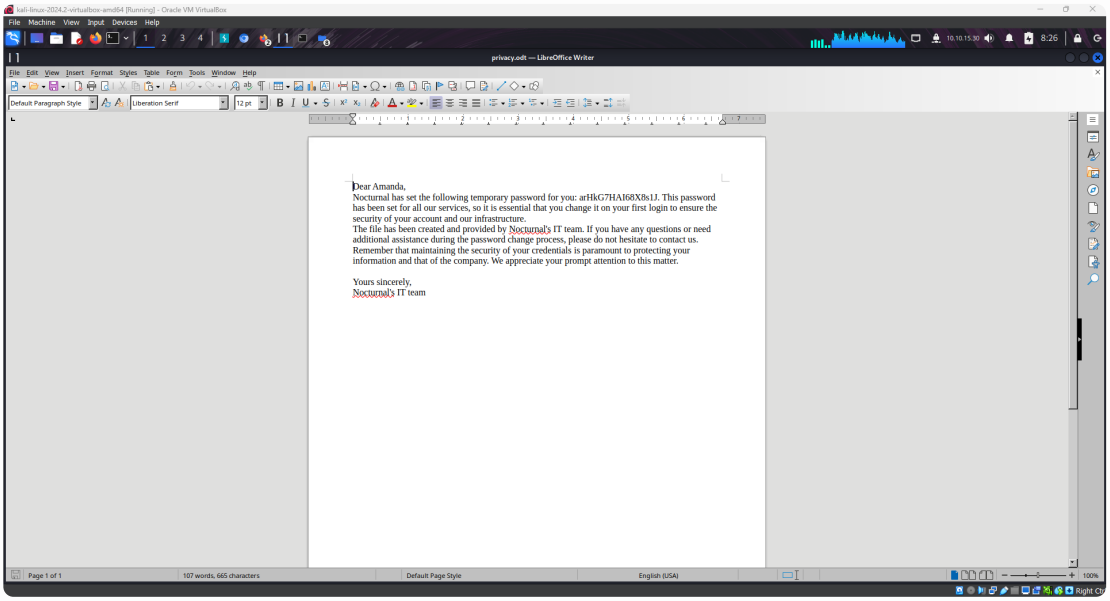
[Screenshot 2: ffuf directory enumeration results]



[Screenshot 3: File upload interface and normal functionality]



[Screenshot 4: IDOR exploitation showing username parameter manipulation]



[Screenshot 5: Successful access to amanda's privacy.odt file]

---

2. Broken Access Control - HIGH

> **Risk Score**: 8.0/10

Description: Administrative functions accessible with regular user credentials.

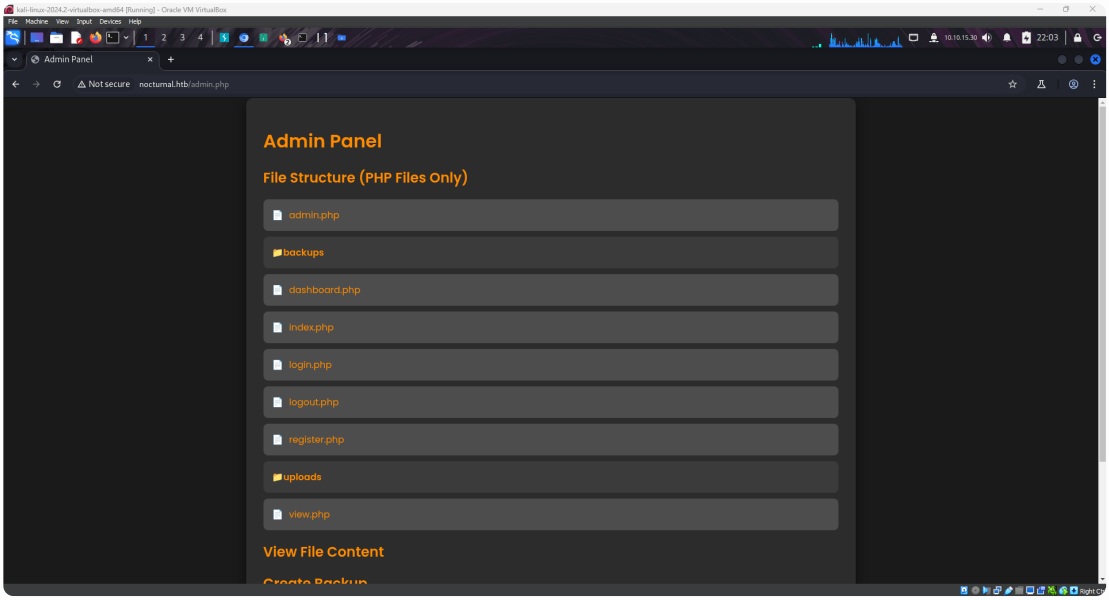
Technical Details:

- Admin panel accessible at `/admin.php`
- No proper role-based access control
- Source code disclosure functionality
- Backup creation accessible to non-admin users

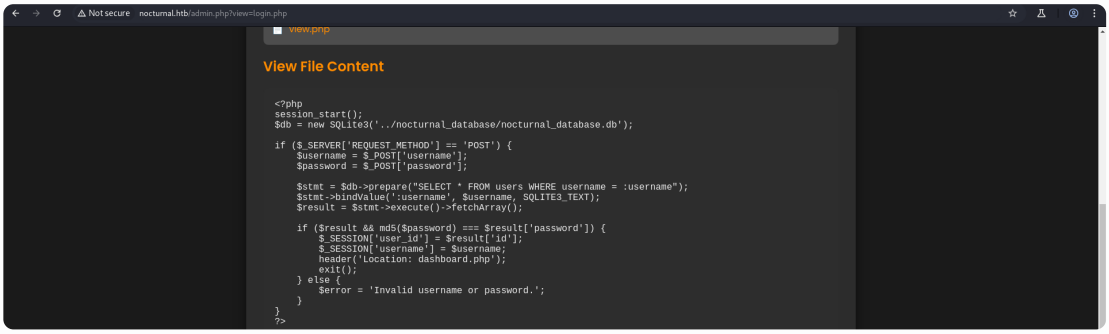
Exploitation:

- Used amanda's credentials (obtained from IDOR) to access admin panel
- Gained access to source code and backup functionality

> **Screenshot Requirements:**

- 

[Screenshot 6: Admin panel access with amanda credentials]

- 

[Screenshot 7: Source code disclosure functionality]

---

3. Command Injection - CRITICAL

> **CVE Reference**: CWE-78 (OS Command Injection)
> **Risk Score**: 9.5/10

Affected Component: Backup functionality in `admin.php`

Vulnerable Code Analysis:

```
function cleanEntry($entry) {
    $blacklist_chars = [';', '&', '|', '$', ' ', '`', '{', '}', '&&'];

    foreach ($blacklist_chars as $char) {
        if (strpos($entry, $char) !== false) {
            return false; // Malicious input detected
        }
    }

    return htmlspecialchars($entry, ENT_QUOTES, 'UTF-8');
}
```

```
$password = cleanEntry($_POST['password']);
$command = "zip -x './backups/*' -r -P " . $password . " " . $backupFile . " .
```

Vulnerability Analysis:

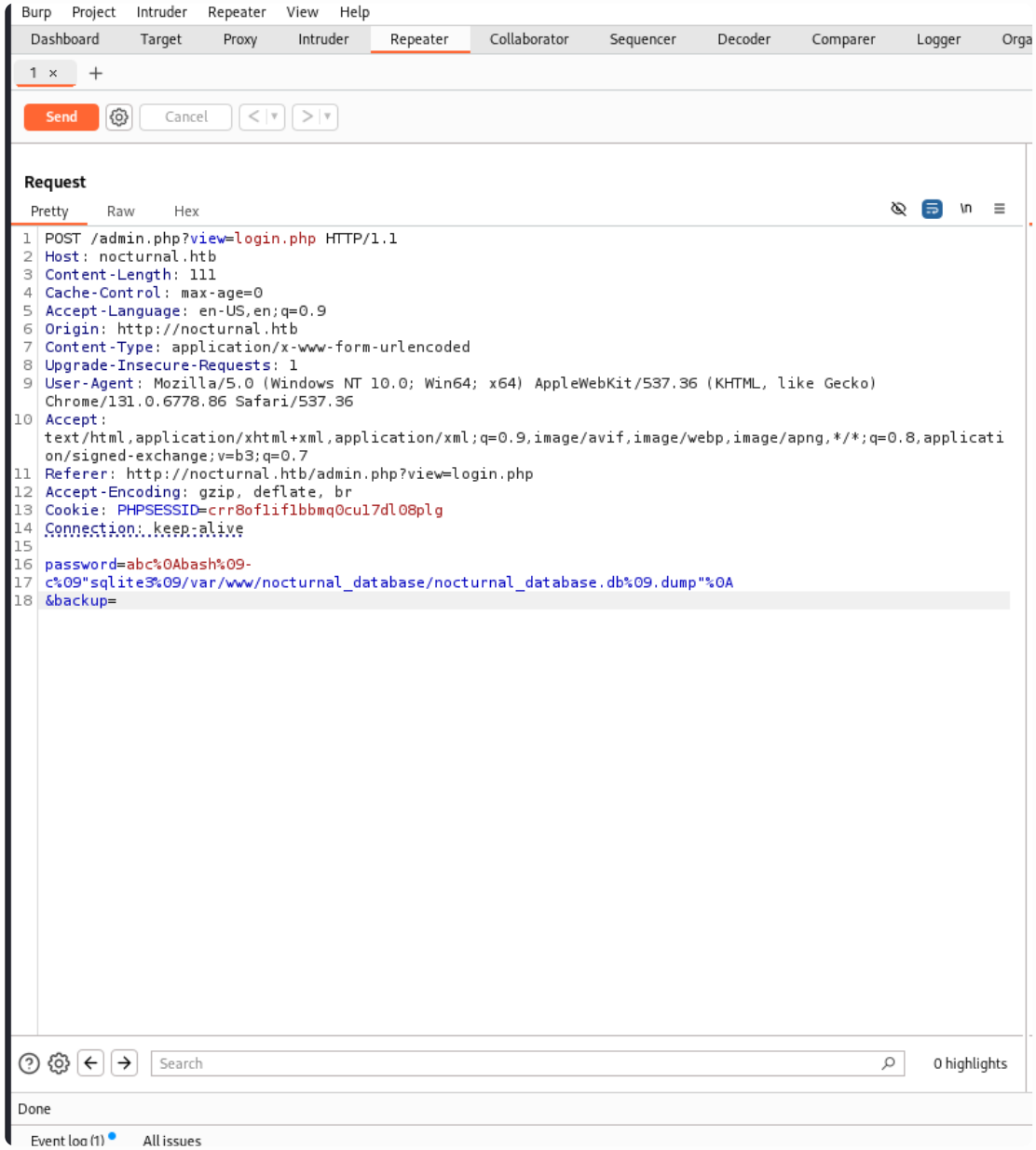
- **Insufficient Input Validation**: Blacklist approach with incomplete character set
- **Command Injection**: Direct concatenation of user input into shell command
- **Bypass Technique**: Using URL-encoded newlines (`%0A`) and tabs (`%09`) to bypass blacklist

Exploitation Payload:

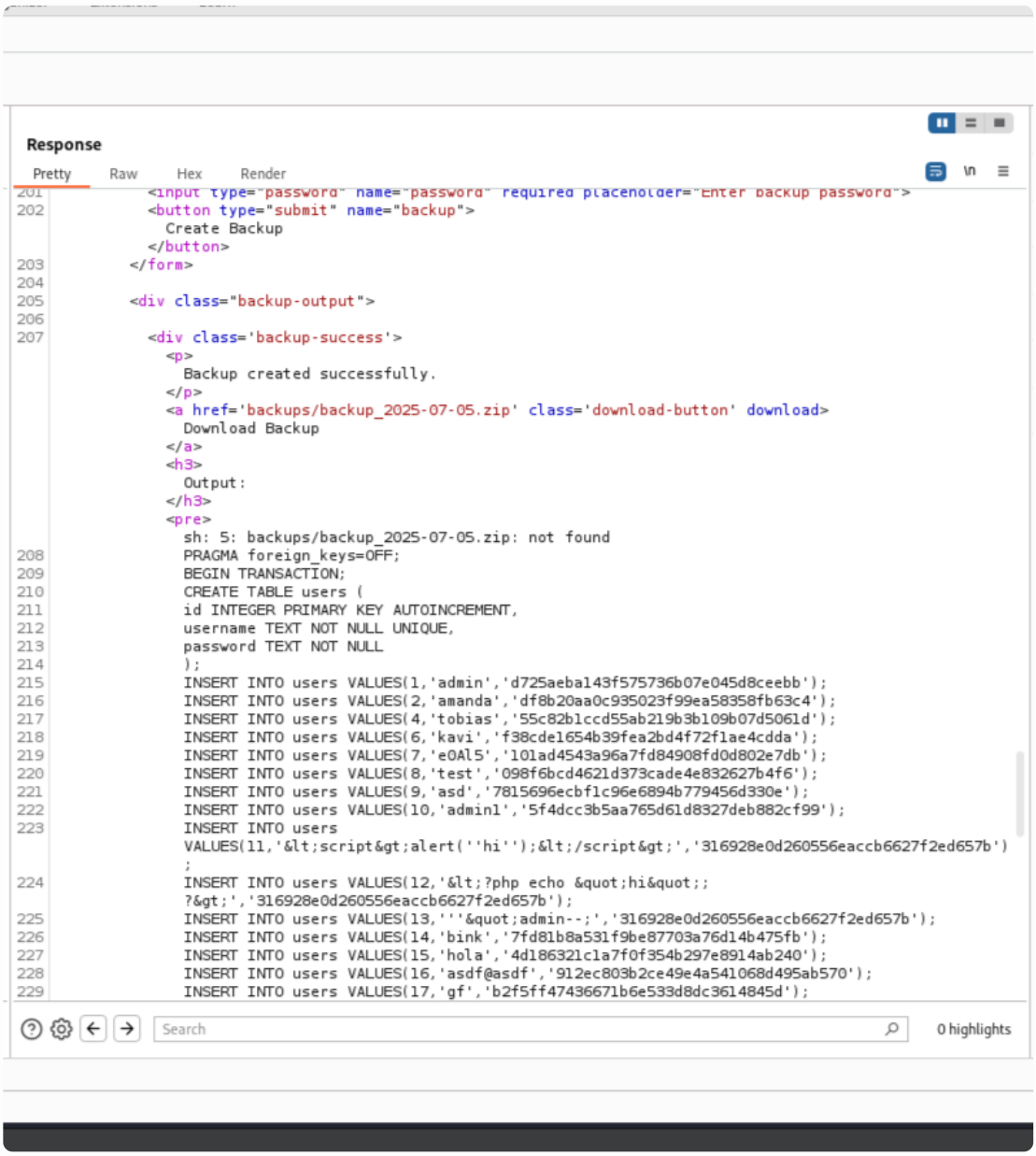
```
POST /admin.php?view=dashboard.php HTTP/1.1
Host: nocturnal.htb
Content-Type: application/x-www-form-urlencoded

password=%0Abash%09-c%09"sqlite3%09/var/www/nocturnal_database/nocturnal_databas
```

Impact: Complete database extraction including all user credentials and sensitive data.

> **Screenshot Requirements:**



[Screenshot 8: Burp Suite showing command injection payload]



[Screenshot 9: Successful database dump extraction]

---

4. Weak Password Storage - HIGH

> **Risk Score**: 7.5/10

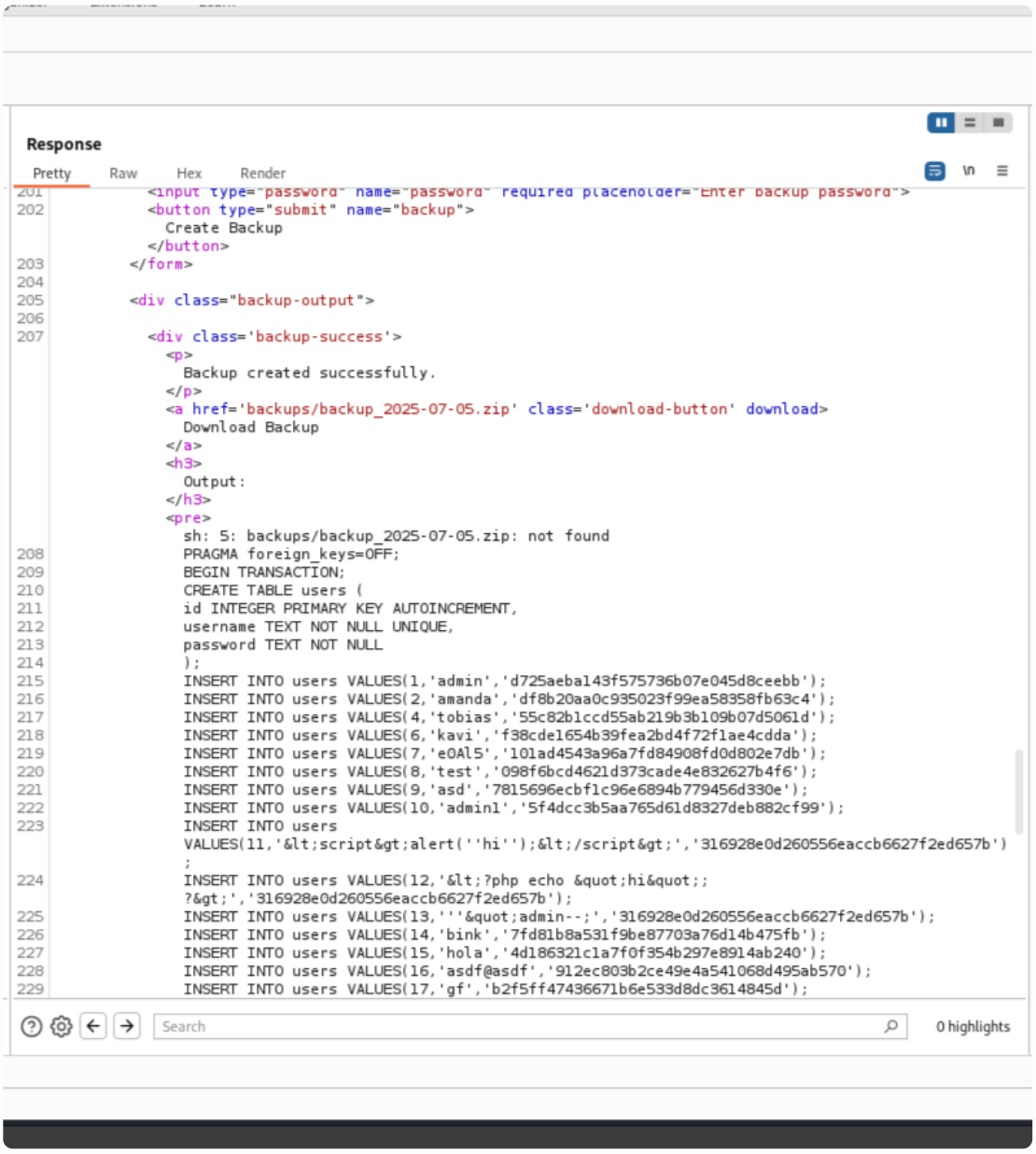
Description: Database contained user credentials with weak hashing or plaintext storage.

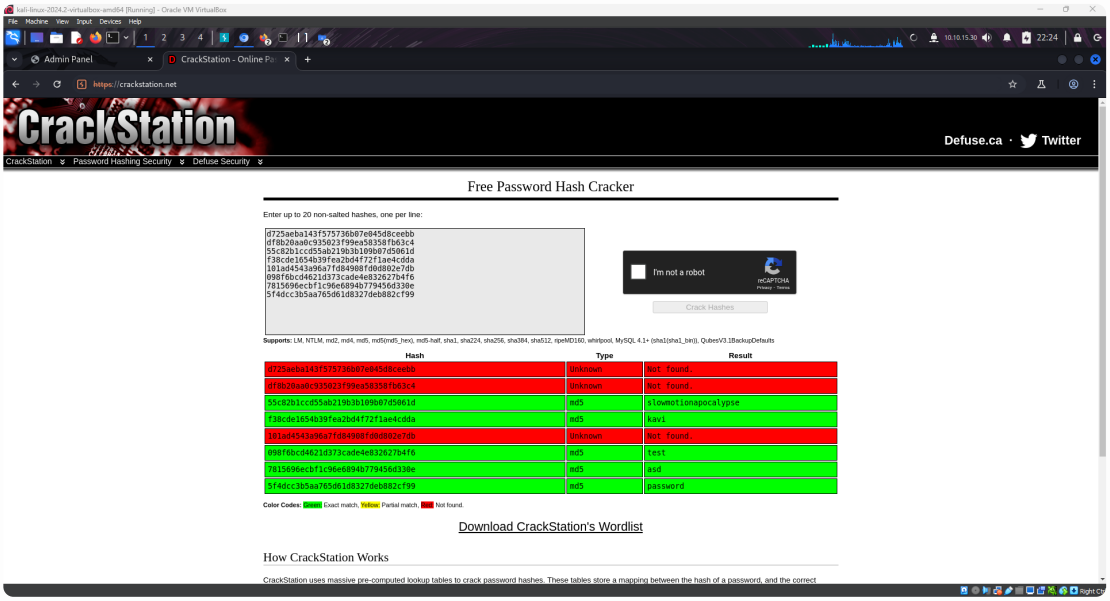
Technical Details:

- Database location: `/var/www/nocturnal_database/nocturnal_database.db`
- Multiple user accounts with weak passwords
- Successful password cracking using CrackStation

Compromised Credentials:

- `tobias:slowmotionapocalypse`
- `admin:slowmotionapocalypse`
- Additional user accounts with cracked passwords

> **Screenshot Requirements:**
>
> - 
>
>
> [Screenshot 10: Database dump showing user credentials]



[Screenshot 11: CrackStation password cracking results]

---

5. SSH Access and Lateral Movement - MEDIUM

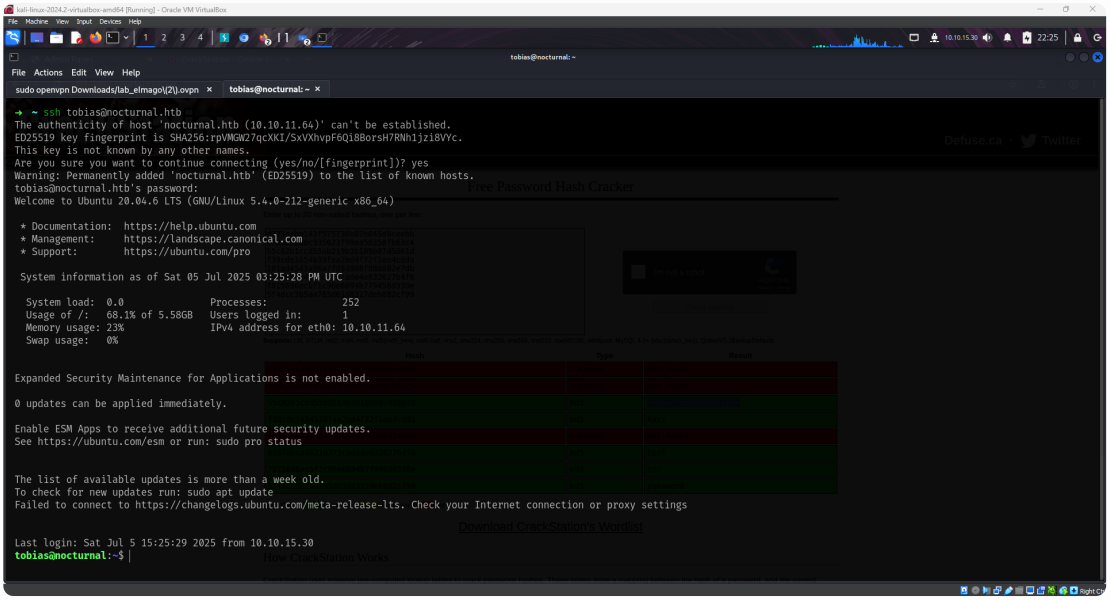
> **Risk Score**: 6.5/10

Description: Compromised credentials provided SSH access to the target system.

Exploitation:

- Used `tobias:slowmotionapocalypse` for SSH authentication
- Successful system access with user-level privileges
- Discovery of additional services running locally

> **Screenshot Requirements:**

- 

  [Screenshot 12: Successful SSH login as tobias]

---

6. ISPConfig Service Discovery - MEDIUM

> **Risk Score**: 6.0/10

Description: Undocumented ISPConfig service running on localhost:8080.

Technical Details:

- Service accessible only from localhost
- ISPConfig web management interface
- Required port forwarding for external access

Discovery Process:

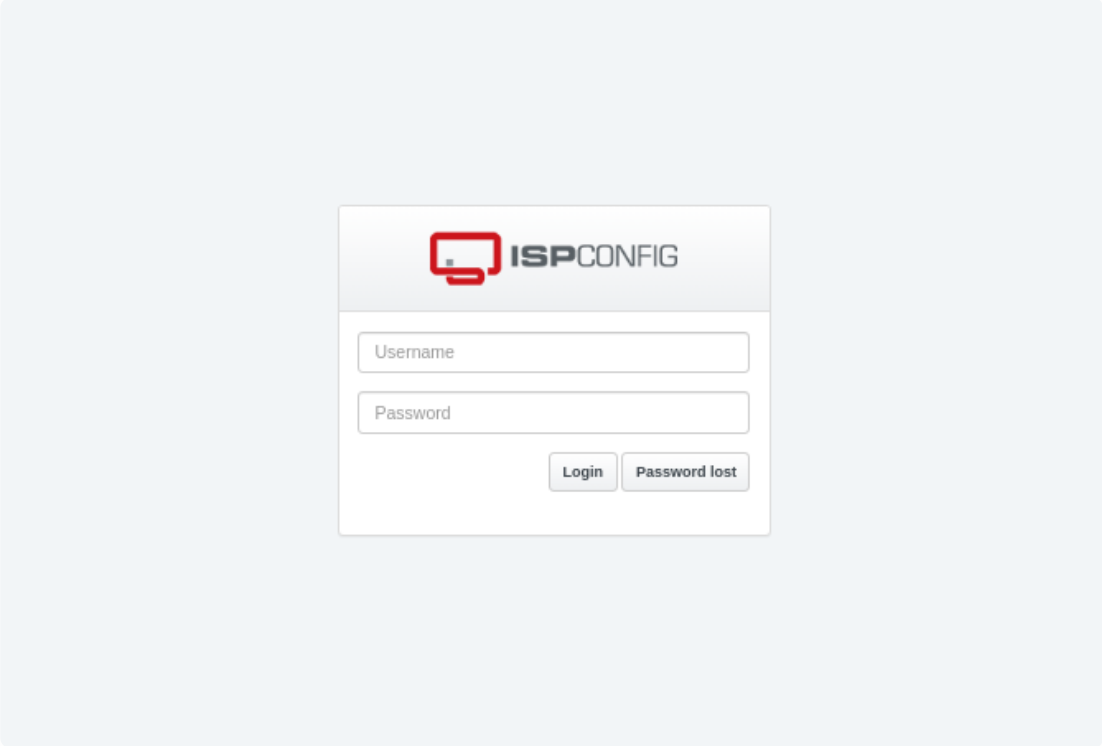
1. Enumeration of `/var/www/` directory revealed ISPConfig installation
2. Network statistics analysis showed localhost:8080 binding
3. Port forwarding setup for external access
4. Service identification as ISPConfig web panel

> **Screenshot Requirements:**
>
> - 

>
>   [Screenshot 14: Directory enumeration showing ISPConfig]

```
tobias@nocturnal:/var/www$ netstat -tupln
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:8080          0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:8000            0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:33060         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:587           0.0.0.0:*               LISTEN      -
tcp6       0      0 :::22                   :::*                    LISTEN      -
udp        0      0 127.0.0.53:53           0.0.0.0:*                           -
tobias@nocturnal:/var/www$
```

[Screenshot 15: Network statistics showing localhost:8080]



[Screenshot 16: ISPConfig login interface after port forwarding]

---

7. ISPConfig Authentication Bypass - CRITICAL

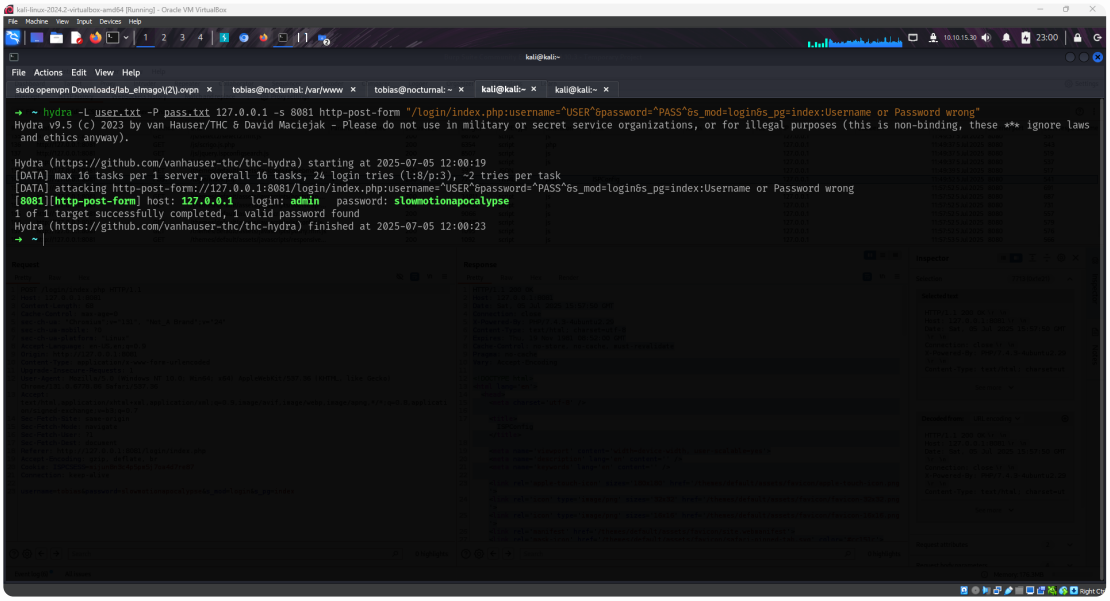
> **Risk Score**: 8.5/10

Description: Weak credential reuse allowed access to ISPConfig management panel.

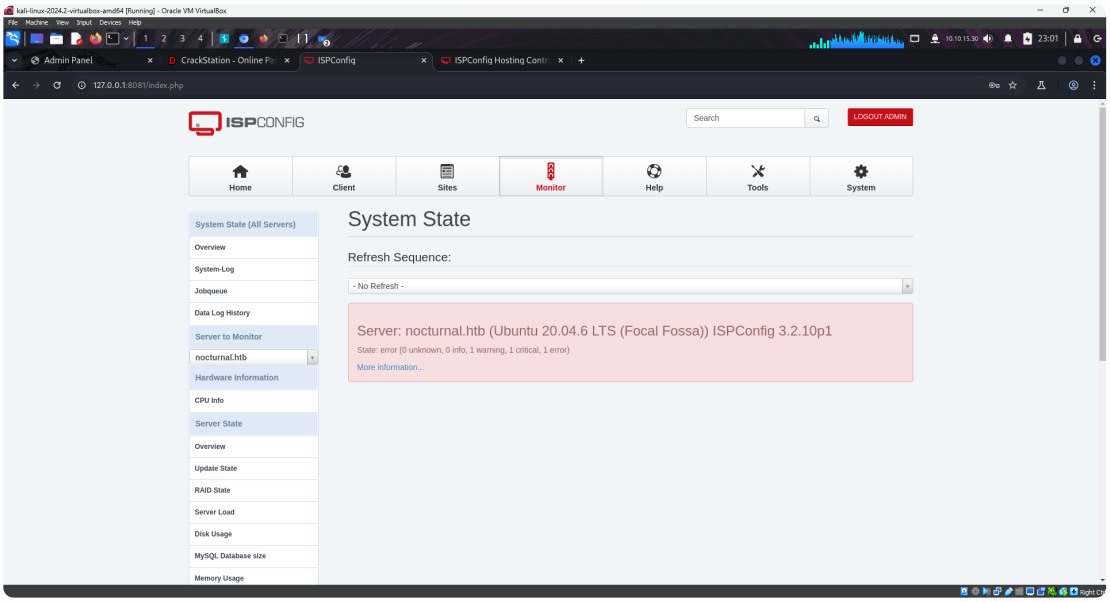
Exploitation:

- Hydra brute force attack using previously discovered credentials
- Successful authentication with `admin:slowmotionapocalypse`
- Access to ISPConfig administrative functions

Screenshot Requirements:

- 

[Screenshot 17: Hydra brute force results]

- 

[Screenshot 18: Successful ISPConfig authentication]

---

8. CVE-2023-46818 Exploitation - CRITICAL

> **CVE Details**: ISPConfig Remote Code Execution vulnerability
> **Risk Score**: 10.0/10

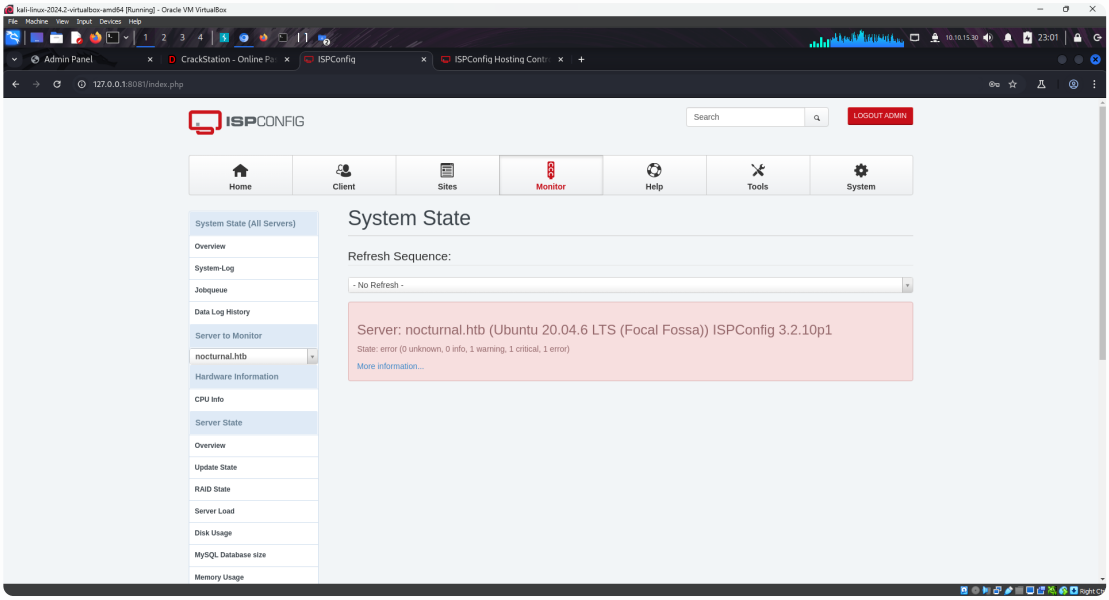
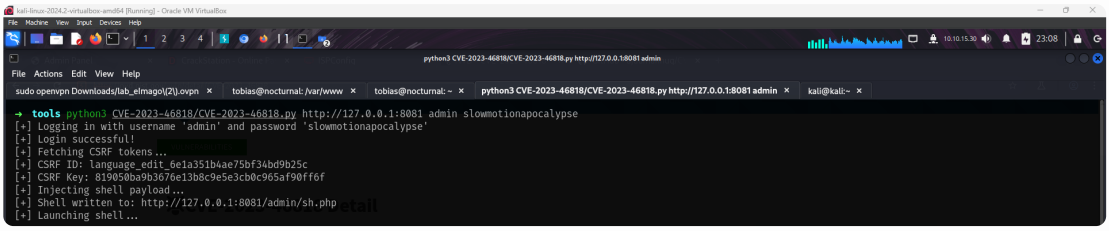
Description: Known vulnerability in ISPConfig allowing authenticated remote code execution.

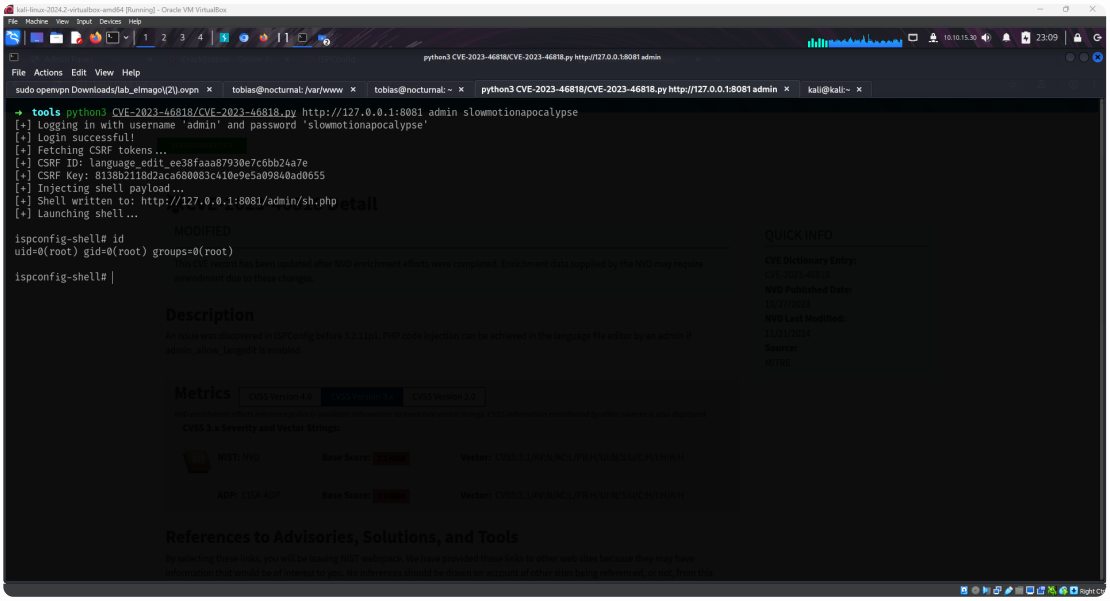
Technical Details:

- Affects specific ISPConfig versions
- Requires authenticated access (already obtained)
- Allows arbitrary code execution with root privileges

Exploitation:

- Identified vulnerable ISPConfig version
- Located and executed CVE-2023-46818 exploit
- Achieved root-level system compromise

> **Screenshot Requirements:**
>
> - 
>
> [Screenshot 19: ISPConfig version identification]
> - 
>
> [Screenshot 20: CVE-2023-46818 exploit execution]

- 

[Screenshot 21: Root shell confirmation]

Impact Assessment

Business Impact: CRITICAL

Impact Area	Severity	Description
Data Confidentiality	Critical	Complete breach of all user files and database
System Integrity	Critical	Ability to modify any system files or data
Service Availability	High	Potential for complete service unavailability
Compliance	Critical	Multiple regulatory framework breaches
Reputation	High	Severe impact on organizational credibility

Technical Impact:

- **Confidentiality**: Complete breach of all system data
- **Integrity**: Ability to modify any system files or data
- **Availability**: Potential for system destruction or ransomware deployment
- **Persistence**: Multiple methods for maintaining access

Affected Systems and Components

1. **Web Application**: nocturnal.htb file upload system
2. **Database**: SQLite database with user credentials

3. **Operating System**: Complete system compromise
4. **ISPConfig**: Management panel exploitation
5. **SSH Service**: Compromised authentication
6. **File System**: Unauthorized access to all files

Remediation Recommendations

Immediate Actions (Priority 1 - 24 Hours)

> **EMERGENCY RESPONSE REQUIRED**

1. **Isolate System**: Immediately disconnect from network
2. **Reset All Passwords**: Change all user and administrative passwords
3. **Revoke Sessions**: Terminate all active user sessions
4. **Patch ISPConfig**: Update to latest version addressing CVE-2023-46818
5. **Audit System**: Check for persistence mechanisms and backdoors

Critical Fixes (Priority 2 - 48 Hours)

> **HIGH PRIORITY FIXES**

1. **Fix IDOR Vulnerability**: Implement proper authorization checks
2. **Secure Admin Panel**: Implement role-based access control
3. **Fix Command Injection**: Use parameterized commands and proper input validation
4. **Strengthen Password Policy**: Implement strong password requirements
5. **Database Security**: Implement proper password hashing (bcrypt/Argon2)

Short-term Improvements (1-2 Weeks)

> **SECURITY HARDENING**

1. **Input Validation**: Implement whitelist-based validation
2. **Access Controls**: Comprehensive review of all access controls
3. **Security Headers**: Implement proper HTTP security headers
4. **Logging and Monitoring**: Enhanced logging for security events
5. **Network Segmentation**: Isolate administrative services

Long-term Security Measures (1-3 Months)

> **STRATEGIC IMPROVEMENTS**

1. **Security Architecture Review**: Complete application security assessment
2. **Penetration Testing**: Regular security testing program
3. **Security Training**: Developer security awareness training
4. **Incident Response Plan**: Develop and test incident response procedures
5. **Compliance Framework**: Implement security compliance program

---

Secure Code Recommendations

IDOR Prevention:

```
// SECURE - Session-based authorization
session_start();
if (!isset($_SESSION['user_id'])) {
    http_response_code(403);
    exit("Unauthorized");
}

$user_id = $_SESSION['user_id'];
$file_id = $_GET['file_id']; // Use file ID instead of filename

// Verify file ownership
if (!verifyFileOwnership($user_id, $file_id)) {
    http_response_code(403);
    exit("Access denied");
}
```

Command Injection Prevention:

```
// SECURE - Use escapeshellarg() and avoid direct concatenation
$password = escapeshellarg($_POST['password']);
$backupFile = escapeshellarg($backupFile);

// Use proc_open with proper argument array
$cmd = ['zip', '-x', './backups/*', '-r', '-P', $password, $backupFile, '.'];
$process = proc_open($cmd, $descriptorSpec, $pipes);
```

Risk Assessment Matrix

Vulnerability	Likelihood	Impact	Risk Score	Priority

IDOR	High (9/10)	High (9/10)	**Critical (81/100)**	P1
Command Injection	High (9/10)	Critical (10/10)	**Critical (90/100)**	P1
Access Control	High (8/10)	High (8/10)	**High (64/100)**	P2
Weak Passwords	High (9/10)	High (8/10)	**High (72/100)**	P2
CVE-2023-46818	Medium (6/10)	Critical (10/10)	**High (60/100)**	P1

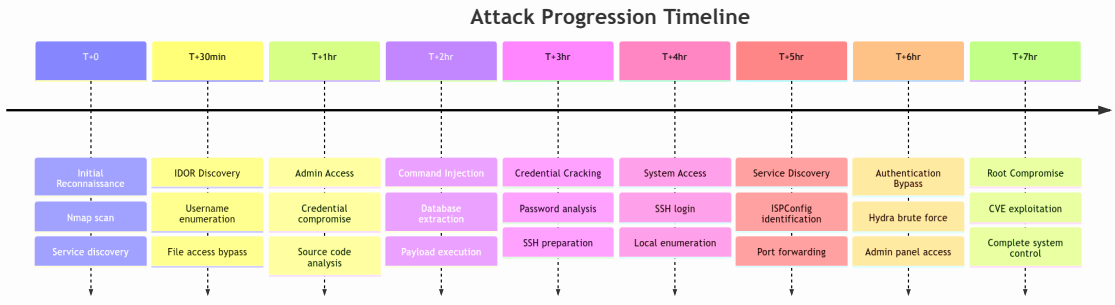
> **Overall Risk Score: Critical (95/100)**

Compliance Impact

This vulnerability chain results in violations of:

Framework	Impact Level	Requirements Violated
GDPR	Critical	Data breach notification, data protection
PCI DSS	Critical	If processing payment data
SOX	High	If publicly traded company
HIPAA	Critical	If processing healthcare data
ISO 27001	Critical	Information security management
NIST	Critical	All five functions compromised

Attack Timeline



Conclusion

> **CRITICAL SECURITY FAILURE**

The nocturnal.htb system suffered from a complete security failure across multiple layers. The vulnerability chain demonstrates how a single IDOR vulnerability can escalate to complete system compromise when combined with poor security practices. The presence of multiple critical vulnerabilities, weak authentication mechanisms, and unpatched services created a perfect storm for exploitation.

Immediate isolation and comprehensive remediation are required to prevent further compromise and data exfiltration.

Visual Evidence Requirements

Critical Screenshots to Include:

#	Screenshot	Purpose
1	Nmap scan results	Initial reconnaissance
2	ffuf directory enumeration	Discovery of admin endpoints
3	IDOR exploitation	Parameter manipulation and file access
4	Admin panel access	Privilege escalation
5	Command injection payload	Burp Suite request
6	Database dump results	Data exfiltration
7	Password cracking	CrackStation results
8	SSH access	System compromise
9	ISPConfig discovery	Service enumeration
10	Port forwarding setup	Local service access
11	Hydra brute force	Authentication bypass
12	CVE-2023-46818 exploit	Root compromise
13	Root shell confirmation	Final system takeover

Appendix

Technical Environment

- **Target**: nocturnal.htb
- **Services**: SSH (22), HTTP (80), ISPConfig (8080)
- **Database**: SQLite at `/var/www/nocturnal_database/nocturnal_database.db`
- **Compromised Users**: amanda, tobias, admin
- **Final Access**: Root-level system compromise

Tools and Techniques

- **Reconnaissance**: Nmap, ffuf
- **Exploitation**: Burp Suite, custom payloads
- **Credential Attacks**: Hydra, CrackStation
- **Database Extraction**: SQLite3 via command injection
- **Privilege Escalation**: CVE-2023-46818 exploit

References

- OWASP Top 10 2021 - Multiple categories affected
- CWE-639: Authorization Bypass Through User-Controlled Key
- CWE-78: OS Command Injection
- CVE-2023-46818: ISPConfig Remote Code Execution
- NIST Cybersecurity Framework