Part 1: Find Information about Email Breaches

Objective: Investigate email breaches using online services and tools like EmailHarvester and Spiderfoot to identify compromised email addresses and related information.

Step 1: Investigate Your Email Status

Possible Outcomes:

- **Personal Email Breaches**: Using services like haveibeenpwned.com, you might discover that your email address appears in breaches such as the Adobe breach (2013, ~153M accounts), LinkedIn breach (2016, ~164M accounts), or others listed on these platforms. Each breach might reveal compromised data like passwords, usernames, or associated accounts.
- **Domain Breaches**: For a company domain (e.g., example.com), you may find breaches affecting employee emails, potentially exposing sensitive information like credentials or personal details. For instance, a recent breach might indicate unpatched vulnerabilities or poor security practices at the organization.
- **Actionable Insights**: If a client's domain is found in a breach, this could indicate weak password policies or unawareness of compromised accounts, which can be exploited during social engineering or credential-stuffing attacks.

Additional Insights:

- Services like haveibeenpwned.com aggregate data from public breach dumps, often including plaintext or hashed passwords. Ethical hackers can use this to assess the risk level of a client's email infrastructure.
- Breaches may reveal patterns, such as employees reusing work emails on insecure third-party platforms (e.g., social media or forums), increasing the attack surface.
- **Privacy Note**: Be cautious when searching personal emails, as some services log queries, which could raise ethical or legal concerns in a penetration testing context.

Example Process:

1. Visit haveibeenpwned.com and enter an email (e.g., test@example.com).
2. Review the results, which might list breaches like:
    a. **Collection #1** (2019, ~773M records): Email and password combinations.
    b. **LinkedIn** (2016): Email addresses and hashed passwords.
3. Check breachdirectory.com for additional details, such as whether passwords were exposed in plaintext.

Suggestions:

- Cross-reference findings with password reuse analysis. If breached passwords are found, test them (with client permission) against the target's systems to check for reuse.
- Use pwndb (a dark web breach search tool, available on Kali) to search for credentials in underground databases, but ensure legal compliance and client authorization.

Step 2: Use EmailHarvester to Find Email Addresses for a Domain

Possible Outcomes:

- **Email Discovery**: Running emailharvester -d example.com -s google might yield emails like john.doe@example.com or support@example.com scraped from public web pages indexed by Google.
- **Organizational Insights**: Discovered emails might reveal department names (e.g., hr@example.com, it@example.com), indicating the company's structure or key personnel.
- **Limitations**: The tool may return outdated or irrelevant emails if the domain's web presence is sparse or if pages are not indexed.

Code Example: To install and use EmailHarvester in Kali Linux:

bash
CollapseWrapRun
Copy
 # Install EmailHarvester
sudo apt update
sudo apt install emailharvester
 # Check available options
emailharvester -h
 # Run a scan for a domain using Google as the search engine
emailharvester -d example.com -s google -f output.txt

- **Explanation**: The -d option specifies the target domain, -s selects the search engine (e.g., Google, Bing), and -f saves results to a file.
- **Output Example** (in output.txt):

text
CollapseWrap
Copy
john.doe@example.com
jane.smith@example.com

support@example.com

Additional Insights:

- EmailHarvester relies on search engine scraping, so results depend on the domain's public footprint. Combine with other tools like theHarvester for broader coverage (e.g., LinkedIn, Twitter, or DNS records).
- **Enhancement**: Use the -e option to expand the search to subdomains (e.g., sub.example.com) for a more comprehensive email list.
- **Legal Note**: Ensure compliance with search engine terms of service and client agreements, as automated scraping may violate policies.

Suggestions:

- Validate discovered emails using tools like SimplyEmail or Hunter.io to confirm their activity (e.g., via SMTP verification).
- Use the harvested emails as input for phishing simulations (with client permission) or to identify high-value targets like executives.

Step 3: Use Spiderfoot to Research Email Addresses

Possible Outcomes:

- **Breach Data**: Spiderfoot modules like sfp_haveibeenpwned or sfp_leaklookup might confirm an email's presence in breaches, providing details like breach dates or exposed data types.
- **Social Media Links**: Modules like sfp_accounts could reveal accounts linked to the email on platforms like GitHub, Twitter, or LinkedIn, exposing usernames or repositories that might contain sensitive code or information.
- **Technical Footprint**: Modules like sfp_email or sfp_emailcrawlr might find email mentions on forums or paste sites (e.g., Pastebin), indicating potential leaks or public exposure.
- **Example Findings**:
    - Email john.doe@example.com is linked to a GitHub account with public repositories containing configuration files.
    - The email appears on a forum discussing a company project, revealing internal details.

Code Example: To start Spiderfoot and run a scan:

bash
CollapseWrapRun
Copy
 # Start Spiderfoot server
spiderfoot -l 127.0.0.1:5001

- Open a browser and navigate to http://127.0.0.1:5001.
- Configure modules (e.g., sfp_haveibeenpwned, sfp_accounts, sfp_emailcrawlr) with free API keys from respective services if required.
- Run a scan with:
    - **Scan Name**: EmailRecon
    - **Target**: john.doe@example.com
    - **Modules**: Select sfp_haveibeenpwned, sfp_accounts, sfp_leaklookup, etc.
    - **Output**: A report listing breaches, associated accounts, and web mentions.

Additional Insights:

- **Useful Modules**:
    - **Ahmia**: Searches dark web sites for email mentions, potentially uncovering leaked credentials.
    - **Leak-Lookup**: Checks breach databases for email-related data.
    - **EmailCrawlr**: Finds email mentions on public websites or forums.
- **API Keys**: Many modules (e.g., sfp_dehashed, sfp_haveibeenpwned) require free or paid API keys. Registering for these enhances scan accuracy but may involve costs for heavy usage.
- **Performance**: Spiderfoot scans can be slow for comprehensive module selections. Prioritize modules based on the target's profile (e.g., focus on sfp_accounts for social media-heavy targets).

Suggestions:

- Combine Spiderfoot with manual OSINT techniques, such as searching LinkedIn for employee profiles matching discovered emails.
- Use findings to map organizational hierarchies (e.g., identifying C-level executives) or identify weak points like public repositories.
- **Security Tip**: Ensure Spiderfoot is run locally (127.0.0.1) to avoid exposing scan data over the internet.

Part 2: View File Metadata

Objective: Use ExifTool to analyze file metadata and extract information useful for penetration testing.

Step 1: Install ExifTool and Review Supported Formats

Possible Outcomes:

- **Supported File Types**: ExifTool can read metadata from:
  - **Documents**: PDF, TXT, DOC, DOCM, DOCX, HTML
  - **Audio**: FLAC, MP3, WAV, AIFF, RA, WMA
  - **Video**: AVI, DV, FLV, MOV, QT, MP4, MPEG, RM, WEBM, WMV
  - **Graphics**: BMP, EXIF, GIF, JPEG, JPG, PNG, SVG, TIFF
  - **Archives**: GZ, GZIP, RAR, ZIP
- **Metadata Insights**: Common tags include:
  - **Author/Creator**: Reveals usernames or real names (e.g., Creator: John Doe).
  - **Software**: Indicates tools used (e.g., Software: Microsoft Word 2016).
  - **GPS Coordinates**: For images, may show where the photo was taken.
  - **Device Info**: Camera model or OS version (e.g., Camera Model Name: iPhone 12).

Code Example: To install ExifTool and list supported formats:

bash
CollapseWrapRun
Copy
```
# Install ExifTool
sudo apt update
sudo apt install libimage-exiftool-perl
# List all tags ExifTool can process
exiftool -list
# List supported file formats
exiftool -listf
```

Additional Insights:

- ExifTool is versatile, supporting over 100 file formats, making it ideal for analyzing diverse file types found during reconnaissance.
- Metadata can reveal sensitive information, such as internal network paths in DOCX files (e.g., \\server\share\file.docx) or software versions vulnerable to known exploits.

Suggestions:

- Use the -listg option to view tag groups (e.g., EXIF, IPTC) for targeted metadata extraction.
- Automate metadata extraction for multiple files using a script:

bash
CollapseWrapRun
Copy

```
# Extract metadata for all files in a directory to CSV
exiftool -csv -r /path/to/folder > metadata.csv
```

Step 2: Use ExifTool and GHDB to Find and Analyze Files

Possible Outcomes:

- **GHDB Dorks**: Using Google Hacker Database dorks like filetype:pdf site:*.example.com might locate public PDF files containing metadata like:
  - **Author**: Employee names or usernames (e.g., John Doe).
  - **Company**: Organization name (e.g., Example Corp).
  - **Software**: Outdated software versions (e.g., PDF Producer: Acrobat 9.0, potentially vulnerable).
- **Image Metadata**: JPEG files might reveal:
  - **GPS Coordinates**: Indicating office locations or employee travel patterns.
  - **Camera Details**: Device vulnerabilities (e.g., outdated firmware on a camera model).
- **Document Metadata**: DOCX or PDF files might expose:
  - **Internal Paths**: Network share names or server details.
  - **Comments**: Unintended notes or revision history.
- **Example Finding**: A PDF file's metadata shows Creator: gd-jpeg v1.0, indicating use of an old PHP GD library. Researching PHP GD vulnerability might reveal exploits like CVE-2006-1015 (buffer overflow in older versions).

Code Example: To analyze a single file and a directory:

bash
CollapseWrapRun
Copy

```
# Analyze a single file
exiftool /path/to/sample.jpg
# Example output
```

 # File Name: sample.jpg
 # Camera Model Name: Canon EOS 5D
 # GPS Latitude: 40 deg 42' 51.00" N
 # GPS Longitude: 74 deg 0' 21.00" W
 # Creator: John Doe
 # Analyze a directory and save to CSV
exiftool -csv -r /path/to/folder > metadata.csv

Additional Insights:

- **GHDB Dorks**: Effective dorks include:
  - filetype:docx site:*.example.com inurl:(report | confidential): Finds sensitive documents.
  - filetype:jpg site:*.example.com exif: Locates images with EXIF data.
- **Vulnerability Research**: If metadata reveals software like Adobe Acrobat 9.0, search for CVEs (e.g., CVE-2010-0188 for Acrobat 9 vulnerabilities) to identify potential attack vectors.
- **Privacy Risks**: Public files with unstripped metadata can leak sensitive data, such as employee names or internal server details, which attackers can use for social engineering or network mapping.

Suggestions:

- Use GHDB creatively to find specific file types. For example:
  - filetype:xlsx site:*.example.com inurl:(budget | financial) to find spreadsheets with financial data.
  - filetype:jpg site:*.example.com inurl:(event | conference) to find event photos with GPS data.
- Strip metadata from sensitive files before sharing:

bash
CollapseWrapRun
Copy
 # Remove all metadata from a file
exiftool -all= /path/to/sample.jpg

- Combine metadata findings with other reconnaissance data (e.g., emails from Part 1) to build a comprehensive profile of the target.

Reflection

Possible Outcomes:

- **Information Yield**: The reconnaissance process typically yields fragmented data (e.g., a few emails, metadata from a PDF, or breach details). While not overwhelming, these pieces can be combined to form a detailed picture of the target.
- **Process Description**: Reconnaissance is methodical, requiring patience to follow leads (e.g., an email leading to a LinkedIn profile, which links to a vulnerable repository). It's a mix of automated tools (Spiderfoot, EmailHarvester) and manual analysis (GHDB searches, metadata review).
- **Corporate/Personnel Reconnaissance**: The process highlights how much sensitive information is publicly available due to poor security practices (e.g., unstripped metadata, exposed emails in breaches). It underscores the need for organizations to implement data sanitization and breach monitoring.

Additional Insights:

- **Real-World Context**: Reconnaissance is the foundation of penetration testing, often accounting for 70-80% of the effort in a successful attack. Tools like those in the lab mimic real attacker workflows but are constrained by free APIs and public data.
- **Challenges**: Free tools may miss data available in paid services (e.g., DeHashed's full breach database). Ethical hackers must balance thoroughness with legal and time constraints.
- **Enhancements**: Integrate findings with other OSINT tools like Maltego for visual relationship mapping or Shodan for device fingerprinting based on metadata (e.g., camera models).

Suggestions:

- Document findings in a structured report, linking emails, metadata, and breach data to potential vulnerabilities.
- Use tools like theHarvester alongside EmailHarvester for broader coverage:

bash
CollapseWrapRun
Copy
theHarvester -d example.com -b google,linkedin -f report.html

- Educate clients on metadata risks and recommend tools like MAT2 (Metadata Anonymization Toolkit) for automatic metadata stripping.

Summary of Key Tools and Commands

Tool	Purpose	Example Command

EmailHarvester	Harvest emails from a domain	emailharvester -d example.com -s google -f output.txt
Spiderfoot	Comprehensive OSINT scanning	spiderfoot -l 127.0.0.1:5001
ExifTool	Extract file metadata	exiftool -csv -r /path/to/folder > metadata.csv
GHDB	Find public files via Google dorks	filetype:pdf site:*.example.com (in Google)

Additional Recommendations

1. **Cross-Tool Integration**: Combine EmailHarvester and Spiderfoot outputs with Maltego to visualize relationships between emails, domains, and metadata.
2. **Automation**: Script repetitive tasks (e.g., running ExifTool on multiple directories) using Bash or Python:

bash
CollapseWrapRun
Copy
 # Bash script to analyze all PDFs in a folder
for file in /path/to/folder/*.pdf; do
exiftool "$file" >> metadata_report.txt
done

3. **Ethical Considerations**: Always obtain client permission before performing reconnaissance, especially when using tools that scrape public data or access breach databases.
4. **Advanced Tools**: Explore paid services like DeHashed or IntelTechniques' OSINT tools for deeper breach data, but redirect to https://x.ai/api for xAI's API if integrating with Grok for automated analysis.

Conclusion

The lab provides a practical introduction to reconnaissance using email breach searches and metadata analysis. By combining tools like EmailHarvester, Spiderfoot, and ExifTool with manual techniques (e.g., GHDB), ethical hackers can uncover actionable insights about a target's vulnerabilities. The process is time-intensive but critical for identifying attack vectors, emphasizing the importance of thoroughness and ethical boundaries in penetration testing.