Day 4 — SOCMINT Runbook (commands + results)

Target username placeholder: `xyz` (replace with [redacted] or any username)

---

1. Prepare tools folder & clone Sherlock

Create tools folder (if missing) and clone Sherlock:

```
cd C:\
mkdir C:\tools
cd C:\tools
git clone https://github.com/sherlock-project/sherlock.git
```

What this does: downloads Sherlock into `C:\tools\sherlock` (or `sherlock-master` depending on ZIP/clone).
Verify:

```
dir C:\tools
dir "C:\tools\sherlock"   # or dir "C:\tools\sherlock-master"
```

---

2. Install required Python libs for Sherlock

Install minimal libs Sherlock needs (if you didn't use `requirements.txt`):

```
pip install requests beautifulsoup4 lxml
```

Verify: `python -c "import requests, bs4, lxml; print('OK')"`

---

3. Run Sherlock (single username)

Run Sherlock to check where the handle exists and print to console:

```
python "C:\tools\sherlock-master\sherlock_project\sherlock.py" xyz
```

Save to file (raw output):

```
mkdir "C:\Users\Nidhi [redacted]xyz\raw\sherlock" -Force
```

```
python "C:\tools\sherlock-master\sherlock_project\sherlock.py" xyz >
"C:\Users\Nidhi [redacted]erlock\sherlock
_raw.txt"
```

Optional JSON (if supported / version dependent):

```
python "C:\tools\sherlock-master\sherlock_project\sherlock.py" xyz
--json > "C:\Users\Nidhi [redacted]aw\sherlock\sherlock
_raw.json"
```

Expected output files:

- `...\raw\sherlock\sherlock_raw.txt`
- optionally `...\raw\sherlock\sherlock_raw.json`

Quick verify (print found lines):

```
Select-String -Pattern "Found" "C:\Users\Nidhi [redacted]\raw\sherlock\sherlock
_raw.txt"
```

---

4. Run Sherlock for a list (targets.txt) and save all results

If you have a `targets.txt` with usernames:

```
mkdir "C:\Users\Nidhi [redacted]OneDrive\[redacted]
CSF notes\sherlock_output" -Force
```

```
python "C:\tools\sherlock-master\sherlock_project\sherlock.py" -d
"C:\Users\Nidhi [redacted]r CSF
notes\targets.txt" > "C:\Users\Nidhi [redacted]
[redacted]
notes\sherlock_output\all_results.txt"
```

Verify:

```
dir "C:\Users\Nidhi [redacted]
CSF notes\sherlock_output"
Get-Content "...\all_results.txt" -Tail 50
```

---

5. Parse Sherlock output (keep only found accounts)

Create `parsed` folder and extract “Found” lines:

```
mkdir "C:\Users\Nidhi [redacted]
[redacted]xyz\parsed" -Force

Select-String -Pattern "Found" "C:\Users\Nidhi [redacted]
[redacted]ck
_raw.txt" |
Out-File "C:\Users\Nidhi [redacted]
[redacted]
d.txt"
```

Result: `parsed\sherlock_parsed.txt` — ready evidence (platform + URL lines).

---

6. TheHarvester — domain oriented

If the target has a domain (replace with real domain), run TheHarvester. If no domain, skip and use manual search.

Install via pip (if not):

```
pip install theharvester
```

Run (use `.txt` output for reliability on Windows):

```
mkdir "C:\Users\Nidhi [redacted]
-Force

python -m theharvester -d example.com -b google -l 200 -f
"C:\Users\Nidhi [redacted]harvester\theh
arvester.txt"
```

If no domain: do manual Google searches for username and save results to `manual_results.txt`.

---

7. Manual Google / site: searches for social presence (Step 4)

Use these queries in the browser (replace username):

```
"xyz"
"xyz" with variants (e.g., "xyz_10", "xyz10")
site:instagram.com "xyz"
site:linkedin.com "xyz"
site:github.com "xyz" OR "xyzname"
site:reddit.com "xyz"
"xyz" email OR "@gmail.com" OR "@yahoo.com"
```

Save evidence: screenshots into respective folders:

```
...\day4_socmint\xyz\linkedin-screens
...\day4_socmint\xyz\instagram-screens
...\day4_socmint\xyz\twitter-screens
```

And paste URLs/snippets into:

```
...\day4_socmint\xyz\raw\theharvester\manual_results.txt
```

---

8. Reddit check (multi-method)

- Sherlock: already checks Reddit in its sweep.
- Google Dork:

```
site:reddit.com "xyz"
```

- Reddit search bar:

```
author:xyz
```

Save results to `raw\reddit_manual.txt` and parsed entries to `parsed\reddit_parsed.txt`.

---

9. Image EXIF check (exiftool)

If you downloaded images:

1. Install ExifTool on Windows and place `exiftool.exe` somewhere convenient (e.g., `C:\tools\exiftool\exiftool.exe`).
2. Run for each image:

```
mkdir "C:\Users\Nidhi[redacted]xyz\parsed\images" -Force

cd "C:\tools\exiftool"
.\exiftool.exe "C:\path\to\raw\images\instagram_post.jpg" >
"C:\Users\[redacted]mages\instagram_post_exif.txt"
```

What to look for: `GPSLatitude`, `GPSLongitude`, `CreateDate`, `ModifyDate`, `Device Model`, `Software`.
If empty or stripped, note in `sensitive_finds.md`.