Forensic Computer Investigations

Based on a presentation by Steve Romig (Ohio State U., Office of the CIO)

Definitions and Principles

- What is “Forensic Computer Investigation”?
  - Forensic means “pertaining to the law”
  - We have forensic anthropology, ballistics, genetics, chemistry, liquid splatter analysis, dentistry, …
- Good general introduction: *Criminalistics* by Richard Saferstein (Prentice Hall)

Why Bother? (1)

- Academic misconduct
- Policy/human resources issues
- Criminal incidents
- Civil incidents
- These same techniques are useful for general investigations on computers
  - The system crashed, why?
  - We were compromised, how?

Why Bother? (2)

- Some questions to ask:
  - How did they break in?
  - What damage was done?
  - Who did it?
  - Who else did they hit?
- We do it in a “forensically sound way” to:
  - Meet legal requirements
  - Reduce liability
  - Preserve evidence

The Four Steps (1)

- Good definition:
  - "Process of identifying, preserving, analyzing and presenting digital evidence in a manner that is legally acceptable in any legal proceedings (*i.e.*, a court of law)."
  - Rodney McKennish: "1998 Donald Mackay Churchill Fellowship to Study Overseas Developments in Forensic Computing" (Australia)

The Four Steps (2)

- Identify the evidence
  - Must identify the type of information that is available
  - Determine how to best retrieve it
  - Examples: disk images, memory dumps, process listings, log files, network traffic logs, etc.
  - We may need to prioritize the evidence, based on what questions we're trying to answer or what we expect to find

The Four Steps (3)

- Preserve the evidence
  - With the least amount of change possible
  - You must be able to account for any changes
  - How can you show that what you have now is ***identical*** to what you had way back then?

The Four Steps (4)

- Analyze the evidence
  - Extract, process, interpret
  - *Extract:* evidence collection may produce binary “gunk” that isn't human readable
  - *Process:* make it humanly readable
  - *Interpret:* requires a deeper understanding of how things fit together
- Your analysis should be reproducible

The Four Steps (5)

- Present the evidence
  - To law enforcement, attorneys, in court, etc.
  - Acceptance will depend on
    - Manner of presentation (did you make it understandable, convincing?)
    - The qualifications of the presenter
    - The credibility of the processes used to preserve and analyze the evidence
    - Credibility enhanced if you can duplicate the process
  - This is especially important when presenting evidence in court

Investigation Workflow

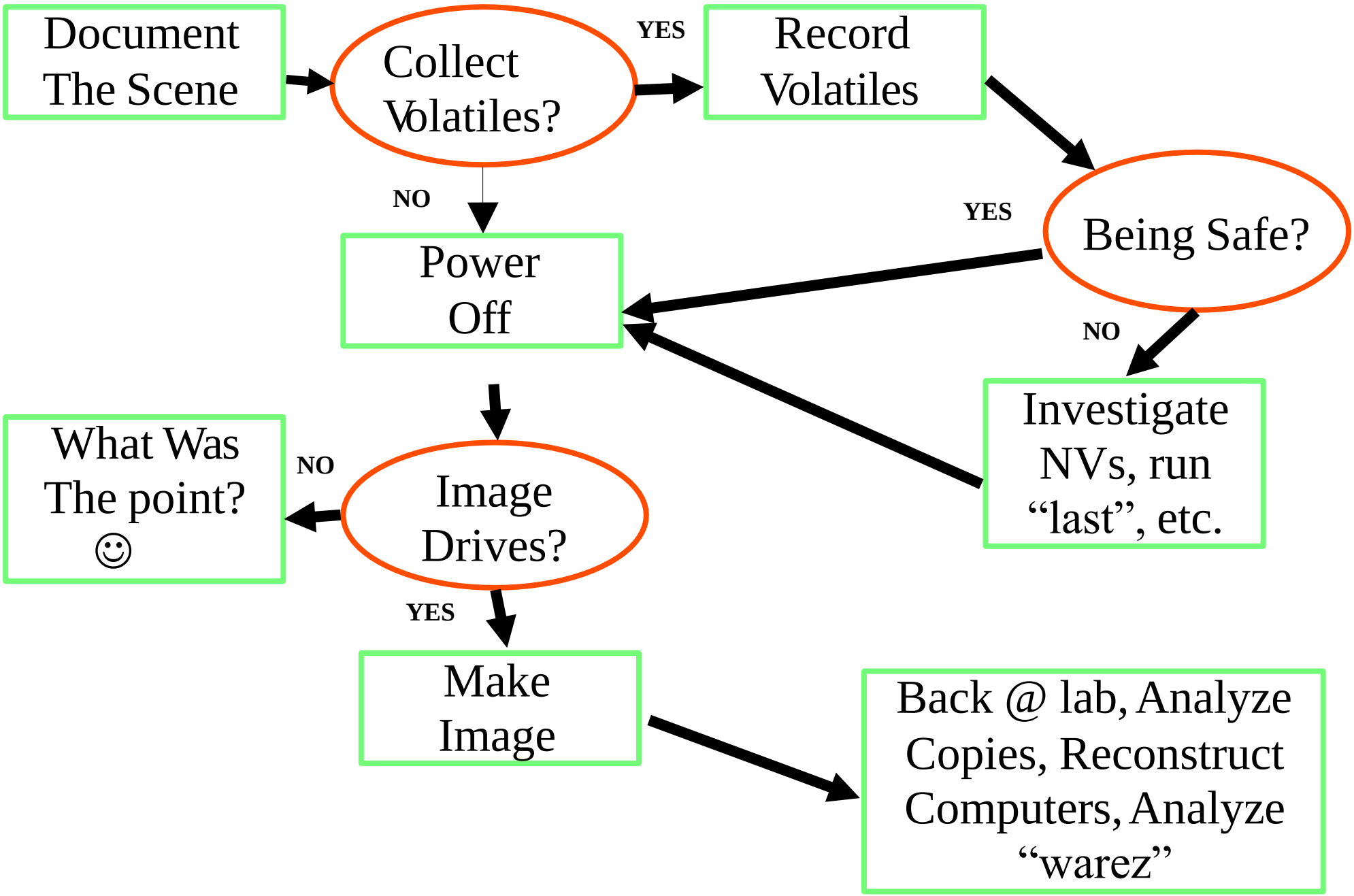
- Collect and analyze evidence to form one or more chronological sequences of events that fit the evidence
- We can't always be conclusive!
  - "The butler did it"
  - "Either the butler did it or he picked up the knife after the murder"
- A feedback loop: analysis leads to more evidence which feeds analysis…

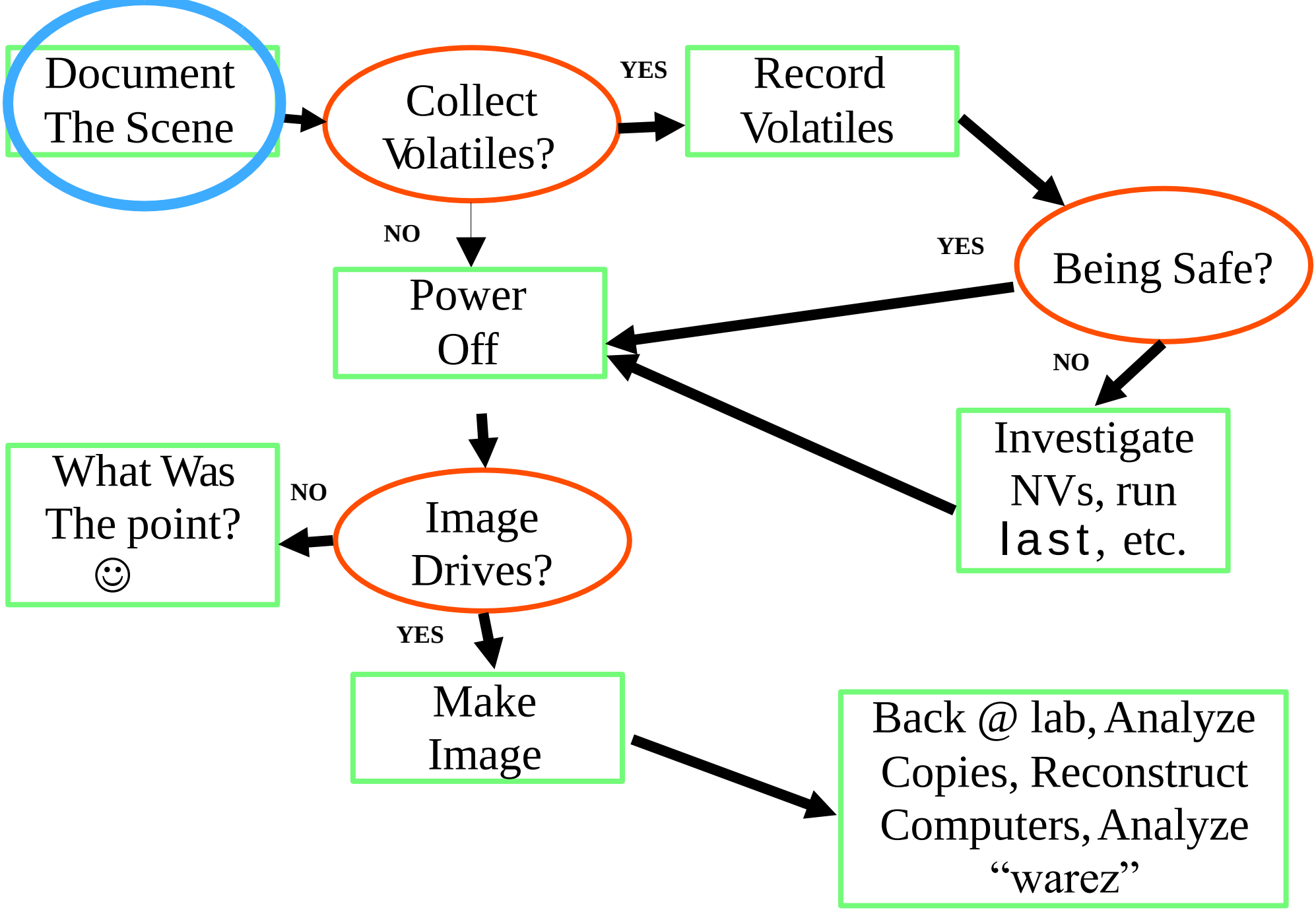
Five Points to Consider

Point	Description
Admissibility	Conforms to legal requirements (“rules of evidence”)
Authenticity	Relevant to the case at hand
Completeness	Complete logs are better than extracts from logs
Reliability	Evidence collected, handled appropriately
Believability	Understandable and convincing

Legal Issues

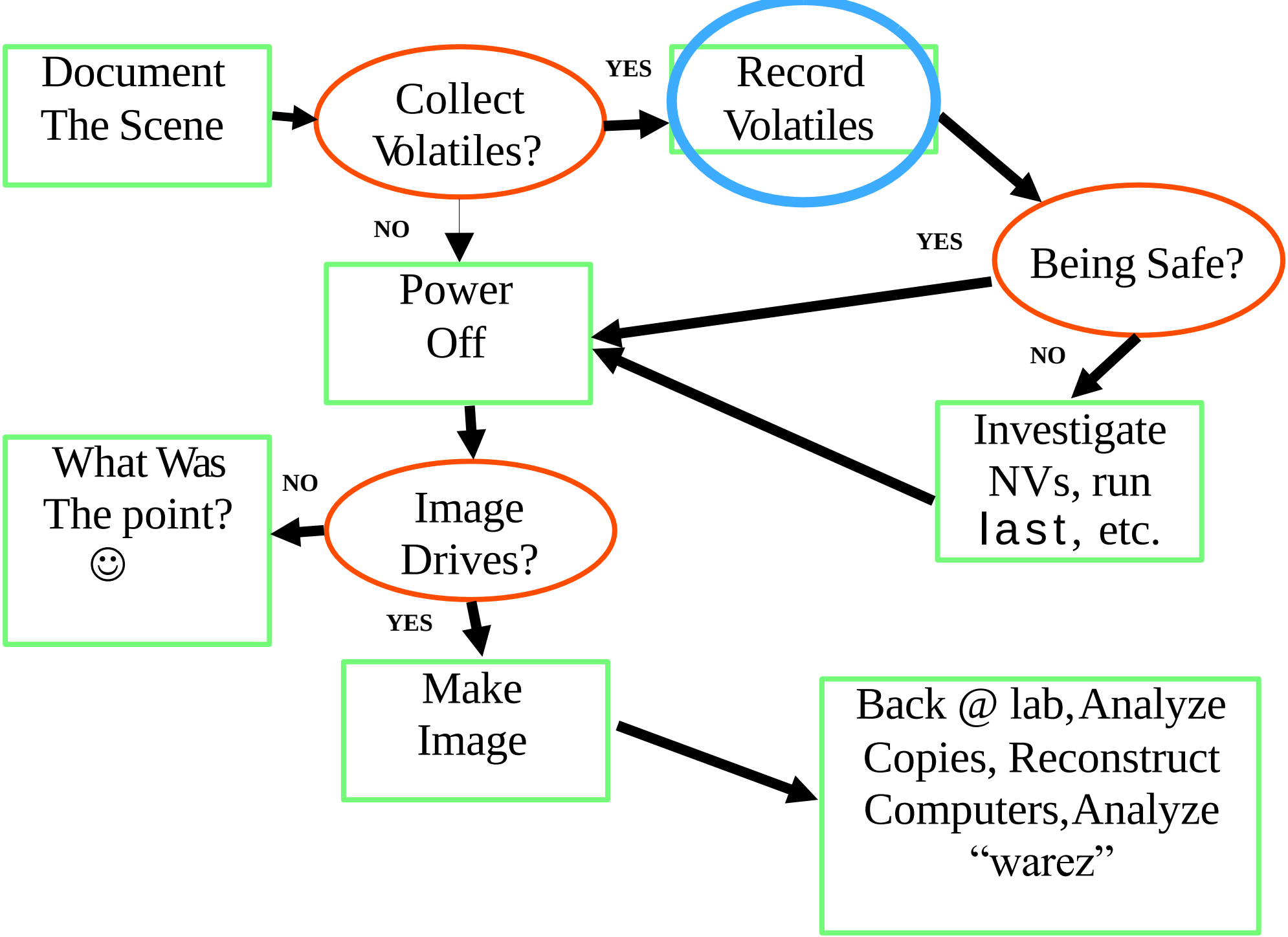
- Best Evidence
- Hearsay
- The Frye and Daulbert Tests
- Chain of Custody
- Exculpatory Evidence
- Fruit of the Poisonous Tree
- Acting Under Color of Law


Document The Scene
Collect Volatiles?
YES
Record Volatiles
NO
Power Off
YES
Being Safe?
NO
Investigate NVs, run "last", etc.
What Was The point?
☺
NO
Image Drives?
YES
Make Image
Back @ lab, Analyze Copies, Reconstruct Computers, Analyze "warez"


Document The Scene
Collect Volatiles?
YES
Record Volatiles
NO
Power Off
YES
Being Safe?
NO
Investigate NVs, run last, etc.
What Was The point?
☺
NO
Image Drives?
YES
Make Image
Back @ lab, Analyze Copies, Reconstruct Computers, Analyze "warez"

Document the Scene

- Map the room(s)
- Take pictures
- Label everything
  - Permanent, or removable sticky notes (***not*** Post-It® notes – they fall off)
  - Unique “tag” (e.g, 315-1-2 means room 315, computer 1, disk 2)
- Catalog everything


Document The Scene
Collect Volatiles?
YES
Record Volatiles
NO
Being Safe?
YES
Power Off
NO
Investigate NVs, run last, etc.
What Was The point?
☺
NO
Image Drives?
YES
Make Image
Back @ lab, Analyze Copies, Reconstruct Computers, Analyze "warez"

Collect Volatile Evidence (1)

- *Volatile evidence:* evidence that will disappear soon, such as information about active network connections, or the current contents of volatile memory.
- Contrast this with persistent storage (e.g., the contents of a disk drive)

Collect Volatile Evidence (2)

- D. Farmer and W. Venema, Coroner's Toolkit (http://www.porcupine.org)
  - Registers, peripheral memory, cache values...
  - Memory (virtual, physical)
  - Network state
  - Running processes/services
  - Loaded kernel modules/DLLs/drivers
  - Network shares
  - Mounted file systems
- Sleuthkit is more recent (http://www.sleuthkit.org/)

Collect Volatile Evidence (3)

- Your actions on the system will affect remaining evidence
  - Running `ps` will overwrite parts of memory
  - Your shell may overwrite its history file
  - You may affect file access times
  - There's always the risk of trojans! (e.g. running programs via `gcore`)

Collect Volatile Evidence (4)

- Rootkits
  - Everything you know about a system is given to you through the software you use (the applications, the libraries, the operating system)
  - A rootkit is software that subverts the system to hide processes, files, network connections and so on
  - These often contain back doors, which give the intruder easy return access
- Examples:
  - Hacker Defender (Windows)
  - 2005 Sony BMG CDs' copy protection (later recalled)
  - Anti-cheating software packaged with some games

Collect Volatile Evidence (5)

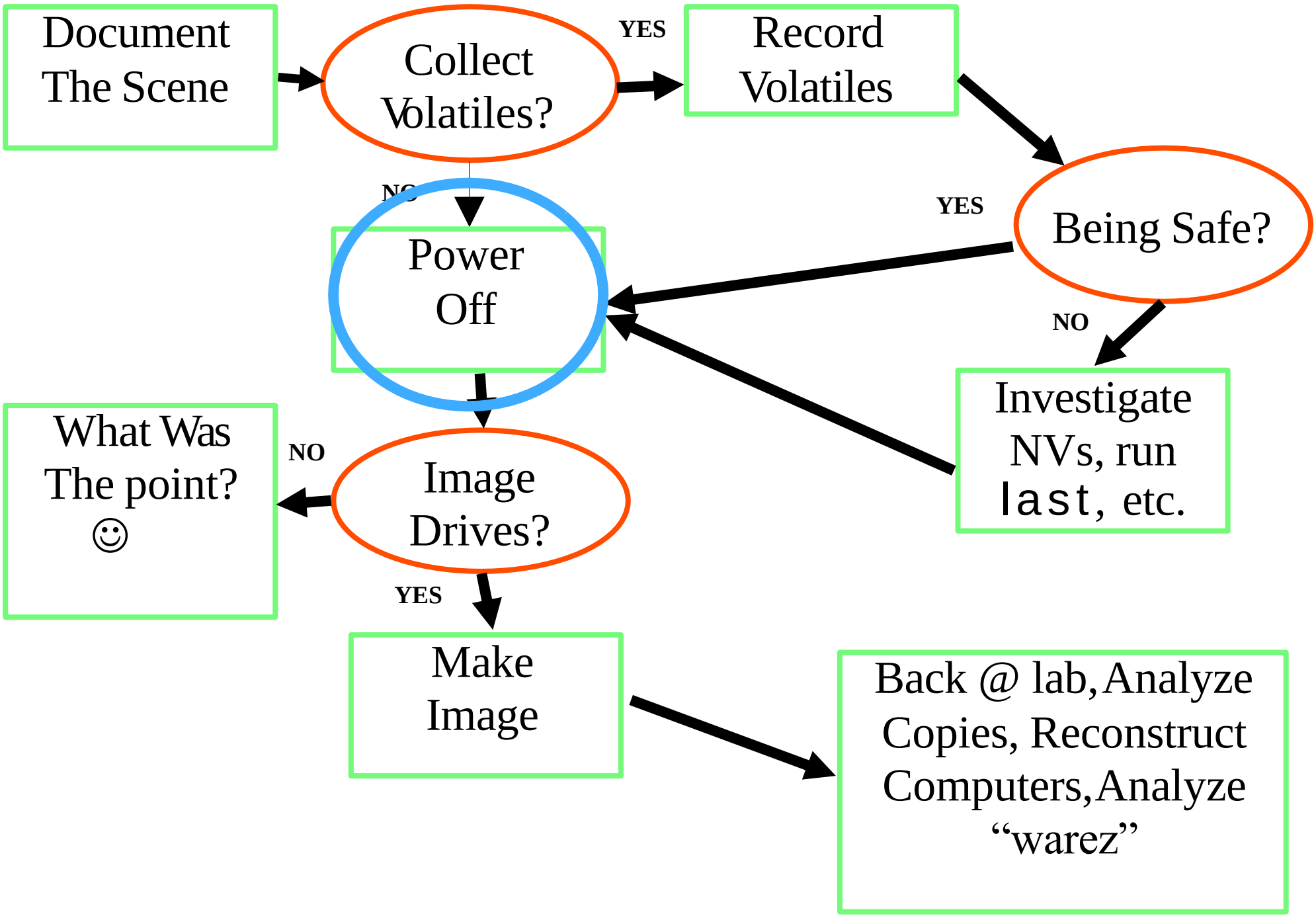
- You need to use known, safe tools to examine a system
  - Statically linked
  - Or include your own libraries
  - Mount from floppy or CD, through net, or download through net
- Won't help with kernel rootkits

Collect Volatile Evidence (6)

- Toolkit might include:
  - Microsoft Sysinternals' FileMon, RegMon, Process Explorer, TCPView, Autoruns, RootkitRevealer, Dumpevt, Dumpreg...
  - F-Secure's Blacklight
  - IceSword
  - Microsoft's Windows Defender
- Live distros such as KNOPPIX (Linux), Windows "rescue" DVDs/USB drives

Collect Volatile Evidence (7)

- If you are collecting volatiles
  - Download/mount your tools (net, floppy, cd, flash)
  - Copy memory, swap, `/tmp`, `pagefile.sys`…
  - Get info about network state (connections, promiscuous interfaces)
  - Get info about running processes
  - Write results to flash drive or across the network: never to the local hard drive


Document The Scene
Collect Volatiles?
YES
Record Volatiles
NO
Power Off
YES
Being Safe?
NO
Investigate NVs, run last, etc.
What Was The point?
☺
NO
Image Drives?
YES
Make Image
Back @ lab, Analyze Copies, Reconstruct Computers, Analyze "warez"

Turning a Computer Off

- When you examine a computer, should you:
  - Turn it off?Use the switch vs. battery/cord?
  - CTRL-ALT-DELETE?
  - Reboot?
  - Unplug it from the net?
  - Filter it at the router?
  - Leave it running and examine it quickly?

Three-Fingered Salute

- Ctrl-Alt-Delete, L1-A (Suns), etc.
  - Can be caught, redirected to destruct routines
  - No real advantage to doing this (that I can think of; you might as well just power off).

Shutdown

- Shutdown/halt/sync would leave file systems clean
  - But these routines might be rigged for destruction
- Don't reboot!
  - Worse than doing a shutdown!
  - Wiping `/tmp` on reboot (if it isn't a RAM-disk)
  - Is it rigged to restart "bad stuff" (backdoors, destructive things) at reboot? Or later, through `cron`?

Unplug from Network

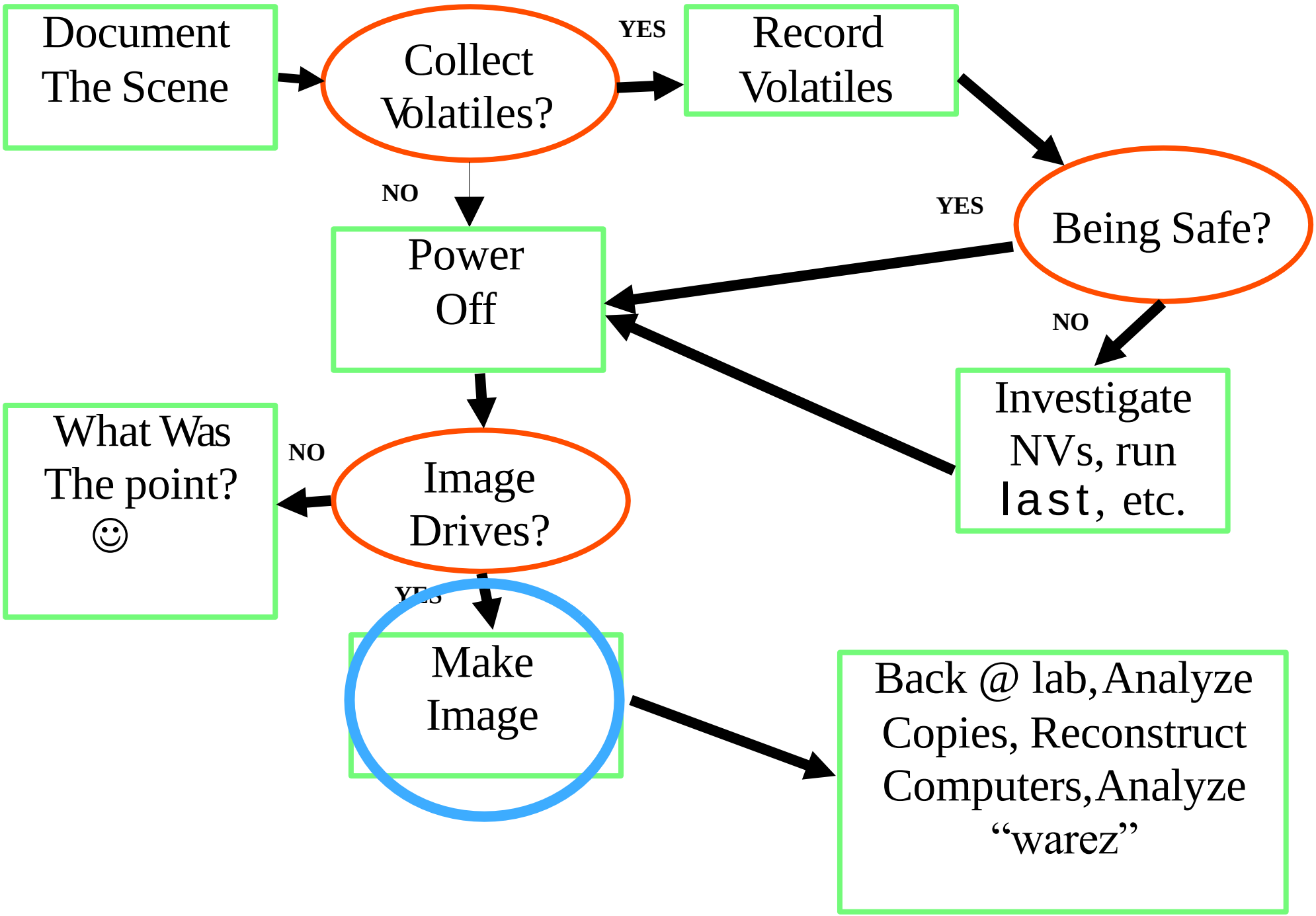
- If you unplug from the network or filter it...
  - What about “dead man switches” that detect when they're off the net and wipe evidence?
  - Marcus Ranum wrote about this in the CSI Alert, September 1999, #198

Leave it Running?

- Without unplugging from the network
  - Until you power it off
- This is probably safe in the short term
  - Risk increases with time, though
  - They might use it to do nasty business – liability?
  - They might wipe evidence, especially if they see you poking around

Power Off

- When you turn it off...
  - You lose volatile evidence: processes, network connections, mounted network file systems, contents of memory...
  - This is critical evidence in many cases: crackers increasingly store tools, logs on remotely mounted file systems
  - On the other hand, if you investigate on running system, you risk modifying the system (especially the disk)

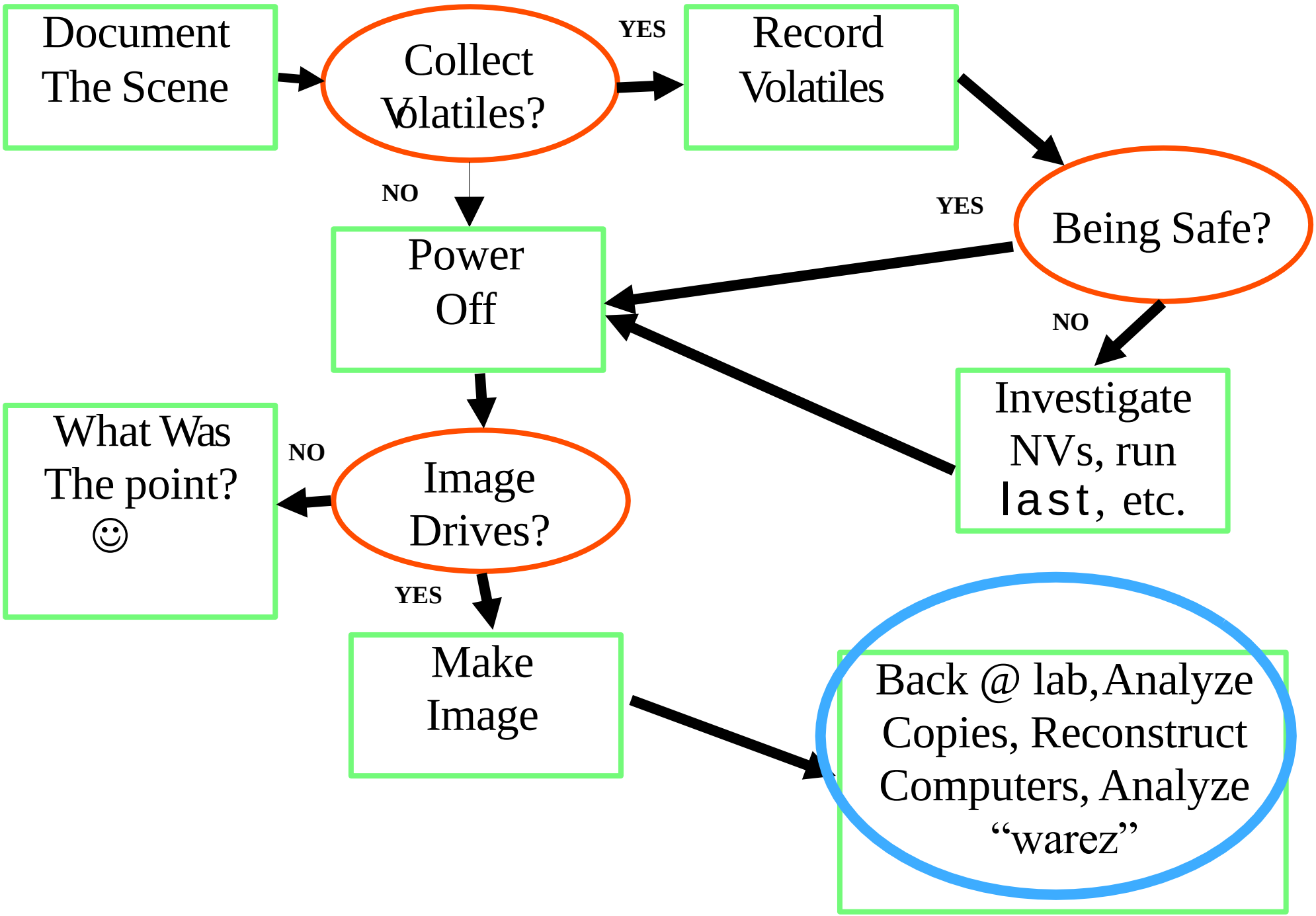

Document The Scene
Collect Volatiles?
YES
Record Volatiles
NO
Power Off
YES
Being Safe?
NO
Investigate NVs, run last, etc.
What Was The point?
☺
NO
Image Drives?
YES
Make Image
Back @ lab, Analyze Copies, Reconstruct Computers, Analyze "warez"

Imaging Disks (1)

- Get partition, RAID, logical volume management configuration
- Make copies of the hard drives (or RAIDs, partitions, ...)
- Calculate and compare hashes (MD5, SHA-1)
- Document and witness copying/verification!
- Reconstruct RAIDs, carve out logical volumes, etc.

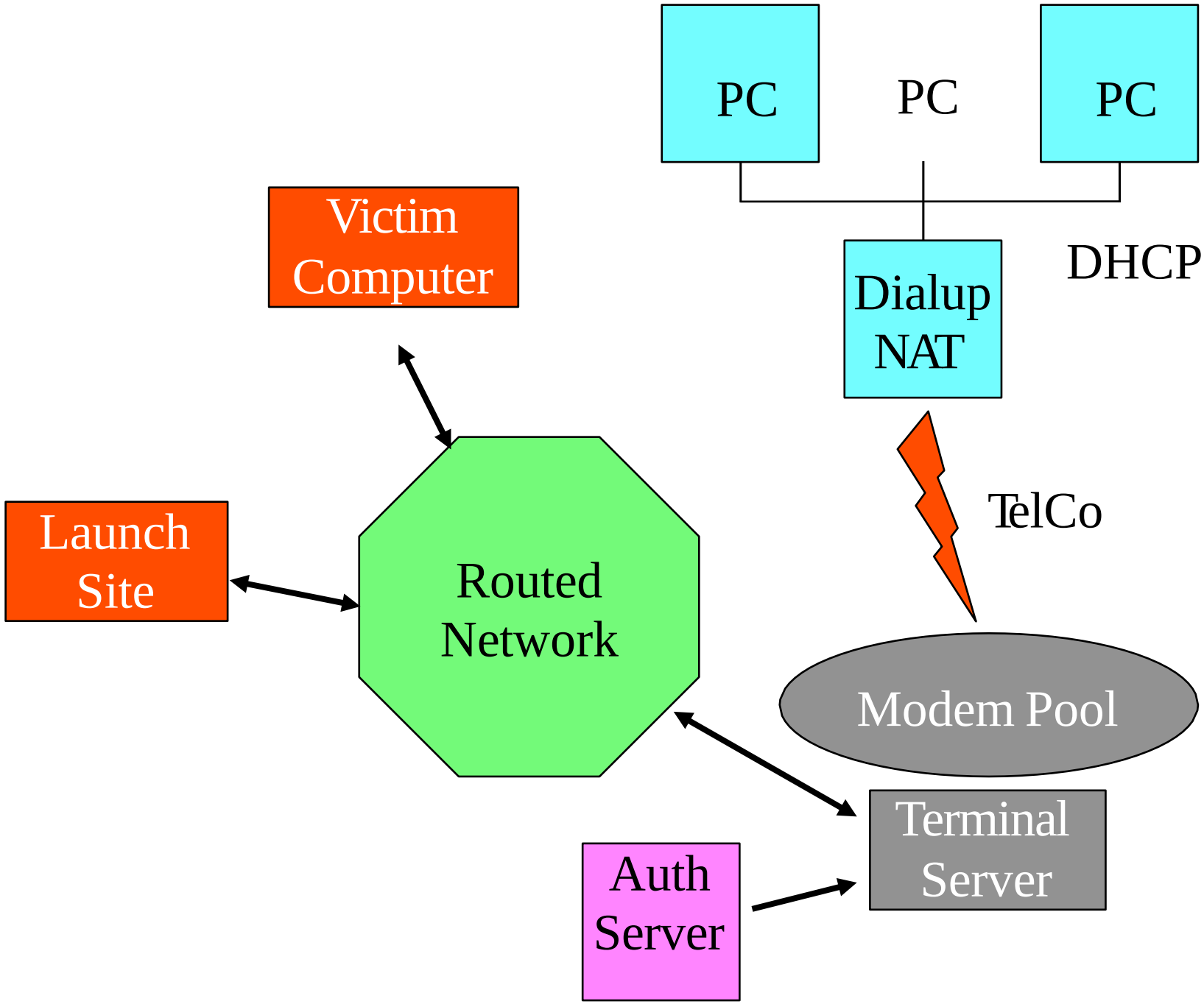
Imaging Disks (2)

- Common tools include:
  - Helix, Knoppix live CDs
  - SMART (Linux live CD) from ASR
  - Forensic ToolKit (FTK) from Access Data
  - EnCase from Guidance Software
  - FTK Imager
  - Raid Reconstructor from Runtime Software
  - Unix dd, md5sum, shasum


Document The Scene
Collect Volatiles?
YES
Record Volatiles
NO
Power Off
YES
Being Safe?
NO
Investigate NVs, run last, etc.
What Was The point?
☺
NO
Image Drives?
YES
Make Image
Back @ lab, Analyze Copies, Reconstruct Computers, Analyze "warez"

We Need to Know:

- Where the evidence is
- What the evidence means
- How to put it together


PC
PC
PC
DHCP
Victim Computer
Dialup NAT
TelCo
Launch Site
Routed Network
Modem Pool
Terminal Server
Auth Server

Where the Evidence Is

- Home system
- Phone system
- Modem pool
- Networks
- Victim computers
- Think about the components
- Ask questions, get expert advice

What the Evidence Means (1)

- This requires a deeper understanding
  - How evidence is created
  - Where it might be missing
  - Or wrong
- Get an expert, ask questions

What the Evidence Means (2)

- A champion.17 login entry in a UNIX wtmp file means…
  - Someone used the champion.17 account to login
  - Or inserted a fake entry
  - ***Not*** necessarily that Adam Champion logged in
- A DHCPlease means...
  - A computer was assigned the lease
  - ***Not*** that that computer was the one using that IP address during the lease time

Importance of Knowing

- Where the logs might be wrong
  - `syslog`, NetFlow exports are sent via UDP
  - Authentication logs from parallel authentication servers
  - NetFlow logs and asymmetric routes
  - Spoofed IP addresses
  - Writable logs (`wtmp`, `utmp` on old UNIX systems)
  - Logs modified by the cracker

Correlating Logs

- You can build stronger case if you can show multiple sources that are in agreement
- Relating log entries to each other
  - Matching log entries by value – e.g. IP address
  - Matching entries by time

Time-Related Issues

- We often use timestamps to correlate entries from different logs on different systems
- Problems include:
  - Time synchronization
  - Time zone
  - Event lag
  - Chronological order of events
  - Event bounding

Time Synchronization

- We can sometimes infer clock offset from the logs
  - Shell history on computer A shows `telnet` B at T1, TCP wrapper on computer B shows `telnet` from A at T2
  - Offset is *probably* T2 – T1
- We can't always do this: not enough info, event lag, etc.

Time Zones

- You can't compare apples to oranges
- Send, request time zone for all logs
- Coordinated Universal Time (UTC) offsets provide a useful reference point
- Make sure you do the math right

Event Lag (1)

- Event lag is the difference in time between related events in different types of logs
  - Connect from computer A to computer B using `telnet` and `login`
  - NetFlow log shows `telnet` starting at 13:05:12
  - TCP wrapper on computer B shows `telnet` at 13:05:12
  - `wtmp` shows actual login at 13:05:58
- Lag can have large variance

Event Lag (2)

- We can use session start time, duration to eliminate some sessions
  - Looking for dialup sessions in phone trace that “match” a login session on the modem pool that started at 2:03:22 and lasted 00:10:05
  - Sessions that start *way* before or after 2:03:22 probably don’t match
  - Sessions that are short than 00:10:05 don’t match
  - Sessions too much longer than 00:10:05 probably don’t match

Event Lab (3)

- Session ending time can sometimes be used to match more accurately than starting time
  - Hang up modem, terminal server terminates login session for you: short lag
  - Logout of UNIX, `telnet` session ends: short lag

Chronological Order of Events (1)

- Some logs are created in chronological order by the ending time of the session
  - Process accounting records on UNIX
  - Cisco NetFlow logs
  - TACACS+ session summary entries

Chronological Order of Events (2)

- This can be very confusing
  - Look through flow log, see traffic from computer, but not `telnet` traffic to computer – might not appear until 30 minutes later in the log
  - Look through process accounting logs, see sub-processes, but not shell process
- We often need to reorder by the starting time of the session

Example Process Accounting Log

ttyp1#user#	12:32:28	00:00:07#	ls
ttyp1#user#	12:33:02	00:00:05#	cat
ttyp1#user#	12:33:45	00:00:03#	egrep
ttyp1#user#	12:33:45	00:00:04#	awk
ttyp1#user#	12:33:45	00:00:04#	sh
. . .	. . .	. . .	. . .
ttyp1#user#	12:30:12#	00:10:02	sh

Event Bounding (1)

- We can use start, end times of one session to “bound” portions of other logs to focus our search for useful information
  - For instance, modem pool auth log shows session from T1 to T2
  - Probably not going to find flow logs for the corresponding IP address of interest outside of that session
  - This is obvious

Event Bounding (2)

- It is not obvious that we can't always do this
  - Easy to leave processes running after your login session on Unix
  - Then there's `at`, `cron`, `procmail` and so on
  - These will leave traces long after the modem pool session

Merging Logs

- Sometimes log entries are spread all over the place
  - Multiple parallel authentication servers
  - Multiple SMTP front ends
  - Multiple routers with asymmetric routing
- Need to merge logs from multiple sources
- Sort into chronological order

Reliability (1)

- Logs vary in reliability
- How are the logs protected?
  - Some wtmp, utmp files are world-writable
  - Shell history files are writable by their owners
- Depends on the integrity of software that creates log entries
  - Crackers replace these with versions that don't log, or which log false entries – rootkit

Reliability (2)

- Is subject to the security of transmission over the network
  - `syslog`, NetFlow both use UDP
  - subject to data loss
  - subject to possible spoofing
- Guard against problems by correlating from as many sources as possible

Reliability (3)

- We will need to adjust theories to account for anomalies
  - See `telnet` session to computer, but there's no login session
  - This might indicate rootkit installation
  - Doesn't call into question validity of the theory that someone broke into the system – supports it

IP Address and Hostname Problems

- IP addresses can be spoofed
  - Need to recognize cases where this is likely/unlikely
  - Common in flooding
  - Uncommon in `telnet`
- Domain stealing, cache poisoning, etc.
  - IP address is "better" than the name it resolves to
  - Really want to log both
  - If you have to choose one, choose the IP address

Recognize What's Missing

- Sometimes the stuff that's missing is what's interesting
  - See long `telnet` in NetFlow to target
  - But there's no login session
  - Raises suspicion that there's a rootkit
- Example 1: We found a _ directory but it doesn't contain anything
  - Might be empty
  - Might be a rootkit
- Example 2: Flow logs shows traffic to TCP/31337
  - But you can't find a process listening on that port
  - There might be a rootkit

Useful Tools (1)

- We use Guidance Software's EnCase, a commercial product (https://guidancesoftware.com)
- Sleuthkit & Autopsy: open source alternatives (http://www.sleuthkit.org)
- Volatility Framework: open source tools for memory forensics (https://www.volatilesystems.com/default/volatility)

Useful Tools (2)

- Microsoft's Sysinternals tools – Autoruns, Rootkit Revealer, Process Monitor/Explorer, TCPView, RegMon, FileMon, etc. (https://docs.microsoft.com/en-us/sysinternals/)

Thank You

Questions?