Welcome to

11. Incident Response / Computer Forensics

KEA Kompetence Computer Systems Security 2023

Henrik Kramselund he/him han/ham xhek@kea.dk @kramse

Slides are available as PDF, kramse@Github

11-incident-response.tex in the repo security-courses

Subjects

- Attack and Response
- Attack graphs
- Attack surfaces, and reducing them
- Intrusion Handling, phases
- Incident Response
- Digital Forensics / Computer Forensics
- Honeypots

Exercises

- Clean or rebuild a server, use example Debian with your cron job vuln as example
- Cloud environments influence on incident response

DSH chapter 6: Incident Response

DSH chapter 7: Disaster Recovery

Skim: Incident Handler's Handbook by Patrick Kral - ca 18 pages

Definition 27-1 *Attack* is a sequence of actions that create a violation of a security policy.
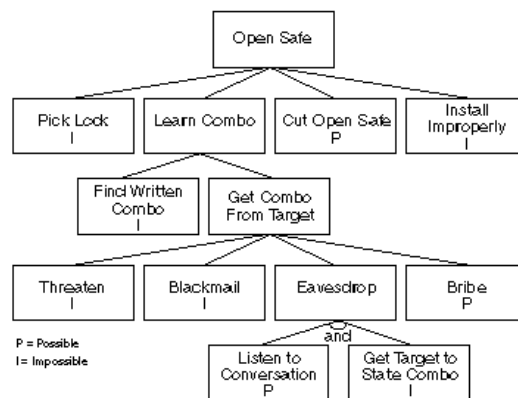
Definition 27-2 A *goal* is that which the attacker hopes to achieve.

Definition 27-3 A *target* of an attack is the entity that the attacker wishes to affect.

Definition 27-4 A *multistage attack* is an attack that requires several steps to achieve it's goal.

Most attacks are multistage.

Example goals: access to systems for learning, stealing, for spamming, for embarrassment

Attack trees

The attack tree diagram shows:

- **Open Safe** (root)
  - **Pick Lock** (I)
  - **Learn Combo**
    - **Find Written Combo** (I)
    - **Get Combo From Target**
      - **Threaten** (I)
      - **Blackmail** (I)
      - **Eavesdrop** (and)
        - **Listen to Conversation** (P)
        - **Get Target to State Combo** (I)
      - **Bribe** (P)
  - **Cut Open Safe** (P)
  - **Install Improperly** (I)

P = Possible
I = Impossible

- Attacks can be said to be based on a chain of dependencies, or graphs
- To achieve goal, need to achieve sub goal x, y, and z – Break the chain and the attack fails!
- Simple example, installing updates remove a dependency for a vulnerability
- Attack trees, picture from Bruce Schneier Attack Trees article December 1999:
  https://www.schneier.com/academic/archives/1999/12/attack_trees.html

Attack surfaces, and reducing them

- Incident prevention
- Real-time intrusion detection systems (IDS/IPS)
- **Definition 27-7** An *attack surface* is the set of entry points and data that attackers can use to compromise a system.
- Reducing the chance of success also helps, randomization
- Address space layout randomization (ASLR) is a host-level moving target defense.
- OpenBSD even randomizes the kernel on install – kernel address randomized link (KARL)

Remember the MITRE ATT&CK framework

MITRE ATT&CK™ is a globally-accessible knowledge base of adversary tactics and techniques based on real-world observations. The ATT&CK knowledge base is used as a foundation for the development of specific threat models and methodologies in the private sector, in government, and in the cybersecurity product and service community.

With the creation of ATT&CK, MITRE is fulfilling its mission to solve problems for a safer world — by bringing communities together to develop more effective cybersecurity. ATT&CK is open and available to any person or organization for use at no charge.

ATT&CK™

https://attack.mitre.org/

Penetration testing

Verification of the system in place

Examines procedural and operational controls

Is the system in fact installed and operated as expected

Example, is the firewall even enabled?

Penetration testing methodologies
https://www.owasp.org/index.php/Penetration_testing_methodologies

Security Assessment Frameworks

- Structured approach to testing, finding and eliminating security flaws
- Information Systems Security Assessment Framework ISSAF
- Penetration Testing Execution Standard (PTES)
- PCI Penetration testing guide, Payment Card Industry Data Security Standard (PCI DSS)
- Technical Guide to Information Security Testing and Assessment (NIST800-115) (GISTA)
- Open Source Security Testing Methodology Manual (OSSTMM)
- CREST Penetration Testing Guide

Which one to choose?

From the book Bishop and https://www.owasp.org/index.php/Penetration_testing_methodologies

Coordinating Response

- **Definition 27-8** A *computer security incident response team* (CSIRT) is a team established to assist and coordinate responses to a security incident among a defined constituency
- Constituency may be a company, an organization, a sector (academic institutions), or even broader
- Morris internet worm lead to the formation of the Computer Emergency Response Team (CERT/CC) coordination center at Carnegie Mellon University
  https://en.wikipedia.org/wiki/CERT_Coordination_Center
- The main danish CERT/CSIRT is https://www.cert.dk/en unfortunately it only covers forskningsnettet the National Research and Educational Networks (NREN) in Denmark!
- In Denmark we also had GovCERT, which is now part of Center for Cyber Security (CfCS)
- There is an internet standard document about Incident Response
  Expectations for Computer Security Incident Response https://www.ietf.org/rfc/rfc2350.txt

Method in Practice

- 1. Capture and preserve the current state of the system and network data
- 2. Extract information about that state and about prior states
- 3. Analyze the data gathered to determine the sequence of actions, which objects they affected, and how
- 4. Prepare and report the results of the analysis to the intended audience

- Example steps:
- Create list of all files
- Create timeline of all changes
- Find all deleted files
- Check free space, previously deleted files, use file-carving tools
- Check other sources, system logs, intrusion detection systems etc.

Example incident response procedures

5.4 Handling an Incident

Certain steps are necessary to take during the handling of an incident. In all security related activities, the most important point to be made is that all sites should have policies in place. Without defined policies and goals, activities undertaken will remain without focus. The goals should be defined by management and legal counsel in advance.

- *Incident Handler's Handbook* by Patrick Kral, SANS Information Security Reading Room
  https://www.sans.org/reading-room/whitepapers/incident/paper/33901
- *Computer Security Incident Handling Guide*, NIST Paul Cichonski, Tom Millar, TimGrance, Karen Scarfone
  https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-61r2.pdf
- Quote from RFC2196 *Site Security Handbook* September 1997, IETF
  https://tools.ietf.org/html/rfc2196#section-5.4
- https://cloud.google.com/security/incident-response/
- Microsoft Azure https://medium.com/@cloudyforensics/azure-forensics-and-incident-response-c13098a14d8d

Intrusion Handling, phases

- *Preparation* for an attack, establish procedures and mechanisms for detecting and responding to attacks
- *Identification* of an attack, notice the attack is ongoing
- *Containment* (confinement) of the attack, limit effects of the attack as much as possible
- *Eradication* of the attack, stop attacker, block further similar attacks
- *Recovery* from the attack, restore system to a secure state
- *Follow-up* to the attack, include lessons learned – improve environment

This method list is from the International Workshop on Incident Response at the Software Engineering Institute in Pittsburgh, Pennsylvania in July 1989.

These are very high-level. Multiple books and courses exist on this subject alone. A short example for today was the *Incident Handler's Handbook* by Patrick Kral

Table 3-5. Incident Handling Checklist

	Action	Completed
	Detection and Analysis	
1.	Determine whether an incident has occurred	
1.1	Analyze the precursors and indicators	
1.2	Look for correlating information	
1.3	Perform research (e.g., search engines, knowledge base)	
1.4	As soon as the handler believes an incident has occurred, begin documenting the investigation and gathering evidence	
2.	Prioritize handling the incident based on the relevant factors (functional impact, information impact, recoverability effort, etc.)	
3.	Report the incident to the appropriate internal personnel and external organizations	
	Containment, Eradication, and Recovery	
4.	Acquire, preserve, secure, and document evidence	
5.	Contain the incident	
6.	Eradicate the incident	
6.1	Identify and mitigate all vulnerabilities that were exploited	
6.2	Remove malware, inappropriate materials, and other components	
6.3	If more affected hosts are discovered (e.g., new malware infections), repeat the Detection and Analysis steps (1.1, 1.2) to identify all other affected hosts, then contain (5) and eradicate (6) the incident for them	
7.	Recover from the incident	
7.1	Return affected systems to an operationally ready state	
7.2	Confirm that the affected systems are functioning normally	
7.3	If necessary, implement additional monitoring to look for future related activity	
	Post-Incident Activity	
8.	Create a follow-up report	
9.	Hold a lessons learned meeting (mandatory for major incidents, optional otherwise)	

Now lets do the exercise

ⓘ Clean or rebuild a server 20min

which is number **43** in the exercise PDF.

Counter Attacks

- General rule, never *hack back*
- Usually not the real source of the attacks
- End up attacking random innocent victims
- Unintended consequences – hacking back medical equipment? SCADA or ICS?
- Ethically not sound

- **Definition 27-9** *Digital forensics* is the science of identifying and anlyzing entities, states, and state transitions of events that have occurred or are occurring.
- 1. Consider the entire system – access to at least the information that the intruder had before and during attack
- 2. Assumptions should not control what is logged
- 3. Consider the effects of actions as well as the actions
- 4. Context assists in understanding meaning
- 5. Information must be processed and presented in an understandable way

Computer Forensics involves the preservation, identification, extraction, documentation and interpretation of computer data.

Computer Forensics: Incident Response Essentials, Warren G. Kruse II og Jay G. Heiser, Addison-Wesley, 2002

Inspired by The Coroners Toolkit (TCT) by Dan Farmer and Wietse Venema

Created by Brian Carrier

Official home TASK and autopsy www.sleuthkit.org

TASK are the command line tools, replace TCT

Autopsy is a Forensic Browser – interface to TASK

kea

Chaosreader Report

Created at: Sun Nov 16 21:04:18 2003, Type: snoop

Image Report - Click here for a report on captured images.
GET/POST Report (Empty) - Click here for a report on HTTP GETs and POSTs.
HTTP Proxy Log - Click here for a generated proxy style HTTP log.

TCP/UDP/... Sessions

1.	Sun Nov 16 20:38:22 2003	30 s	192.168.1.3:1368 <-> 192.77.84.99:80	web	383 bytes	• as_html
2.	Sun Nov 16 20:38:22 2003	29 s	192.168.1.3:1366 <-> 192.77.84.99:80	web	381 bytes	• as_html

Older tool chaosreader, can open network dump and provide HTML output

HTML show sessions, pictures etc.

http://chaosreader.sourceforge.net/

Example output http://chaosreader.sourceforge.net/Chaos01/index.html

Extremely simple, but a good example - quick to demo.

Case: Maltrail

Maltrail is a malicious traffic detection system, utilizing publicly available (black)lists containing malicious and/or generally suspicious trails, along with static trails compiled from various AV reports and custom user defined lists, where trail can be anything from domain name (e.g. `zvpprsensinaix.com` for Banjori malware), URL (e.g. `http://109.162.38.120/harsh02.exe` for known malicious executable), IP address (e.g. `185.130.5.231` for known attacker) or HTTP User-Agent header value (e.g. `sqlmap` for automatic SQL injection and database takeover tool). Also, it uses (optional) advanced heuristic mechanisms that can help in discovery of unknown threats (e.g. new malware).

https://github.com/stamparm/maltrail
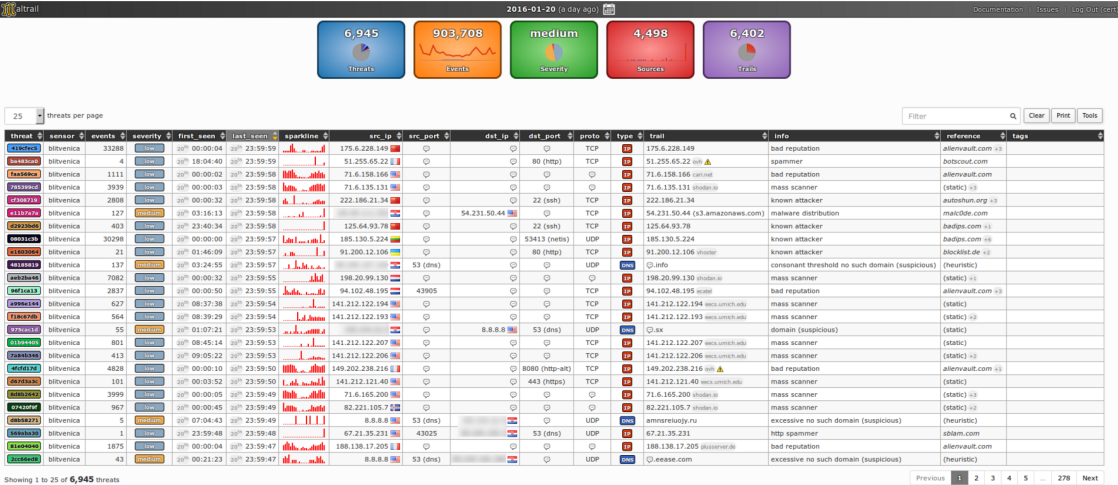
Packet sniffing tools

Tcpdump for capturing packets

Wireshark for dissecting packets manually with GUI

Zeek Network Security Monitor

Snort, old timer Intrusion Detection Engine (IDS)

Suricata, modern robust capable of IDS and IPS (prevention)

ntopng High-speed web-based traffic analysis

Maltrail Malicious traffic detection system https://github.com/stamparm/MalTrail

Often a combination of tools and methods used in practice

Full packet capture big data tools also exist

NetFlow is getting more important, more data share the same links

Accounting is important

Detecting DoS/DDoS and problems is essential

NetFlow sampling is vital information - 123Mbit, but what kind of traffic

Currently also investigating sFlow - hopefully more fine grained

In computer terminology, a **honeypot** is a computer security mechanism set to detect, deflect, or, in some manner, counteract attempts at unauthorized use of information systems. Generally, a honeypot consists of data (for example, in a network site) that appears to be a legitimate part of the site, but is actually isolated and monitored, and that seems to contain information or a resource of value to attackers, who are then blocked.

Source: https://en.wikipedia.org/wiki/Honeypot_(computing)
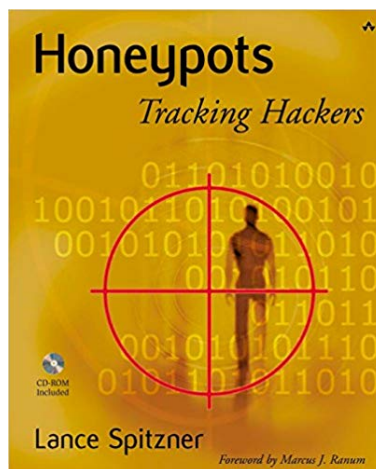
also used as Honeynet - monitored network infrastructure

En honeypot består typisk af:

- Et eller flere sårbare systemer
- Et eller flere systemer der logger traffik til og fra honeypot systemerne

Meningen med en honeypot er at den bliver angrebet og brudt ind i, se også Canary Tokens

History of honeypots – An Evening with Berferd

Artikel om en hacker der lokkes, vurderes, overvåges

Et tidligt eksempel på en honeypot

Senere kom The Honeynet Project http://www.honeynet.org

Billede er: *Honeypots: Tracking Hackers* af Lance Spitzner, 2003

Honeypot High interaction and low interaction

High-interaction honeypots imitate the activities of the production systems that host a variety of services and, therefore, an attacker may be allowed a lot of services to waste their time. By employing virtual machines, multiple honeypots can be hosted on a single physical machine. Therefore, even if the honeypot is compromised, it can be restored more quickly. In general, high-interaction honeypots provide more security by being difficult to detect, but they are expensive to maintain. If virtual machines are not available, one physical computer must be maintained for each honeypot, which can be exorbitantly expensive. Example: Honeynet.

Low-interaction honeypots simulate only the services frequently requested by attackers. Since they consume relatively few resources, multiple virtual machines can easily be hosted on one physical system, the virtual systems have a short response time, and less code is required, reducing the complexity of the virtual system's security. Example: Honeyd.

Source: https://en.wikipedia.org/wiki/Honeypot_(computing)
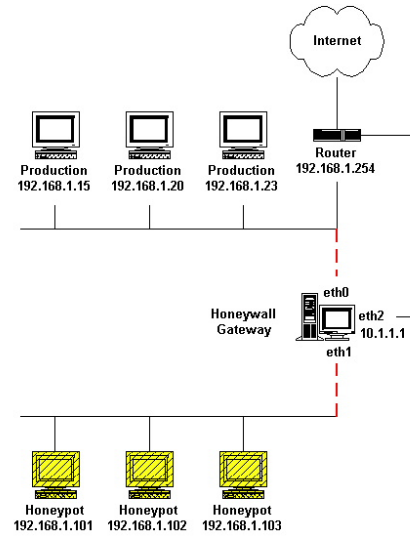
Creating a network architecture with multiple systems become a honeynet.

- Lessons Learned from http://old.honeynet.org/papers/edu/
- Out of all of this were a variety of lessons learned things to do and NOT to do. Hopefully this short list can help you avoid some common mistakes.

Honeynets – Why use them research, production

- Start Small - If you are going to install a honeynet within your enterprise, start small. Begin initially with two machines (in order to detect sweep scans of your honeynet) with operating systems that you are familiar with installed behind the reverse firewall.

- Maintain good relations with your enterprise administrators. THIS IS CRITICAL! Inform your network administrators of the types of exploits that you are seeing. In some cases, they will already be aware of these exploits, but in other cases, you will have been the first person to notice them.

Honeynets - Why use them research, production

- Focus on attacks and exploits originating from within your enterprise network. Theses are the attacks that can do the most damage to your enterprise. Inform your enterprise administrators immediately of these types of attacks since they indicate machines that have already been compromised within the enterprise.

- Don't publish the IP address range of the honeynet. There is no need to do this. Hackers and worms are constantly scanning across the Internet for machines to exploit. You honeynet will be found and attacked.

- Don't underestimate the amount of time required to analyze the data collected from the honeynet. This data must be analyzed every day. You will be collecting lots of information and it must be analyzed to provide any benefit.

- Powerful machines are not necessary to establish the honeynet. The Georgia Tech Honeynet did not use state of the art machines and it functioned as intended. Everything we needed to establish our honeynet was already available on campus.

Source: *Know Your Enemy: Honeynets in Universities Deploying a Honeynet at an Academic Institution*

NIDS

- \+ See all traffic
- − see and need to process ALL TRAFFIC
- \+ Known and understood by management

Honeypot

- \+ See only attack traffic
- \+ Few false positives
- \+ Require less ressources

Myth: laptop passwords protect data

Now that we are talking about reading disk and file systems.

Myth:

Some still believe passwords on laptops protect data

Truth:

They are only a minor nuisance

The login prompt does not protect the data

It is possible to boot from another disk, or remove disk

Are your data secure - data at rest

Lorem ipsum dolor sit amet, consectetur adipiscing elit, set eiusmod tempor incidunt et labore et dolore magna aliquam. Ut enim ad minim veniam, q   nostrud exerc. Irure dolor in reprehend incididunt ut labore et dolore magna aliqua. Ut enim ad minim v   nostrud exercitation ullamco la   nisi ut aliquip ex ea commodo consequa Duis aute irure dolo   nderit in voluptate velit esse   cillum. Tia non ob ea soluad inco   quae egen ium imp   end. Officia deserunt mollit a   orum Et harumd dereud fac   er expedit distinct. Gothica quam nunc putamus parum   eposuerit litterarum formas humanitatis per seacula quarta; modo typ   is videntur   clari fiant sollemnes in futurum; litterarum f   humanitatis per sea   cima et quinta decima, modo typi qui nu   tur parun   llemnes in futuru   rit ! Nam liber te conscient to factor tum p   ioque civi   eque pecun mod   honor et imper r   et, conse   ing elit, sec   t dolore magna aliquam is nostrud exercitatio   lo conse   e in voluptate vem esse cillum dolore eu fugiat nulla pariatur. At vver e   am dignissum qui blandit est praesent.

Stolen laptop, tablet, phone - can anybody read your data?

Do you trust "remote wipe"

How do you in fact wipe data securely off devices, and SSDs?

Encrypt disk and storage devices before using them in the first place!

Single user mode boot

Unix systems often allow single user mode boot
Previously holding command-s on Mac OS X gave single user boot

Laptops can often be booted using PXE network or CD boot

Macbooks boot from CDROM press c

Macbooks can also be turned into *firewire* hard discs
Press t during boot, firewire target mode - same with Thunderbolt now

Unrestricted access to un-encrypted data

Moving hard drive to another computer is also easy

Physical access to a system – **game over**

Full Disk Encryption (FDE)

Full Disk Encryption protect data from physical access

Available in the most popular client operating systems

- Microsoft Windows Bitlocker
- Apple Mac OS X - FileVault
- FreeBSD GEOM og GBDE - encryption framework
- Linux LUKS distributions like Ubuntu ask to encrypt home dir during installation
- Some vendors have BIOS passwords, or disk passwords

34

Attacks on disk encryption

- Firewire, DMA & Windows, Winlockpwn via FireWire
  Hit by a Bus: Physical Access Attacks with Firewire Ruxcon 2006
- Removing memory from live system - data is not immediately lost, and can be read under some circumstances
  Lest We Remember: Cold Boot Attacks on Encryption Keys
  http://citp.princeton.edu/memory/
- This is very CSI or Hollywoord like - but a real threat
- VileFault decrypts encrypted Mac OS X disk image files
  https://code.google.com/p/vilefault/
- FileVault Drive Encryption (FVDE) (or FileVault2) encrypted volumes
  https://code.google.com/p/libfvde/

So perhaps use both hard drive encryption AND turn off computer after use?

kea

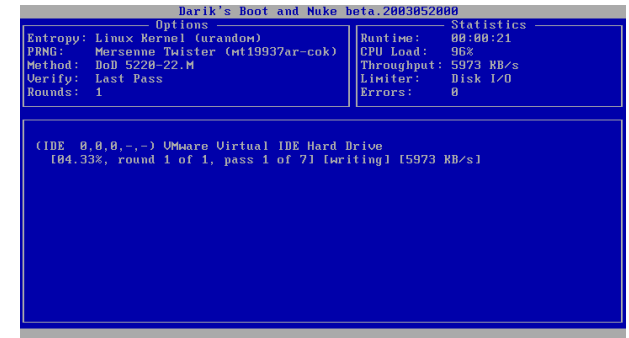Getting rid of data from old devices is a pain

Some tools will not overwrite data, leaving it vulnerable to recovery

Even secure erase programs might not work on SSD
- due to reallocation of blocks

I have used Darik's Boot and Nuke ("DBAN")
http://www.dban.org/

```
                Darik's Boot and Nuke beta.2003052000
 ─ Options ─────────────────────────  ─ Statistics ───────
 Entropy: Linux Kernel (urandom)      Runtime:     00:00:21
 PRNG:    Mersenne Twister (mt19937ar-cok)  CPU Load:    96%
 Method:  DoD 5220-22.M               Throughput:  5973 KB/s
 Verify:  Last Pass                   Limiter:     Disk I/O
 Rounds:  1                           Errors:      0

 (IDE  0,0,0,-,-) VMware Virtual IDE Hard Drive
   [04.33%, round 1 of 1, pass 1 of 7] [writing] [5973 KB/s]
```

Drive	1	2	3	4	5	6	7	8	9	Impact
Crucial MX100 (all form factors)	✗	✗	✗							✗ Compromised
Crucial MX200 (all form factors)	✗	✗	✗							✗ Compromised
Crucial MX300 (all form factors)	✓	✓	✓		✗	✓	✓	✓	✓	✗ Compromised
Samsung 840 EVO (SATA)	✗	✓	✓		✓	✓	✓	✗	✓	~ Depends
Samsung 850 EVO (SATA)	✗	✓	✓		✓	✓	✓	✓	✓	~ Depends
Samsung T3 (USB)				✗						✗ Compromised
Samsung T5 (USB)				✗						✗ Compromised

1 Cryptographic binding in ATA Security (High mode)
2 Cryptographic binding in ATA Security (Max mode)
3 Cryptographic binding in TCG Opal
4 Cryptographic binding in proprietary standard
5 No single key for entire disk
6 Randomized DEK on sanitize
7 Sufficient random entropy
8 No wear leveling related issues
9 No DEVSLP related issues

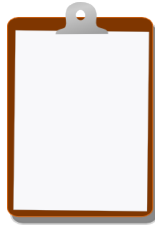*self-encrypting deception: weakness in the encryption of solid state drives (SSDs)*
https://www.ru.nl/publish/pages/909282/draft-paper.pdf

Now lets do the exercise

ⓘ Cloud environments influence on incident response 20min

which is number **44** in the exercise PDF.

For Next Time

Think about the subjects from this time, write down questions

Check the plan for chapters to read in the books

Visit web sites and download papers if needed

Retry the exercises to get more confident using the tools