Security & Privacy

Stephen Farrell

stephen.farrell@cs.tcd.ie

Course materials:

https://down.dsg.cs.tcd.ie/cs7053/

https://github.com/sftcd/cs7053

Slideware + some papers

Computer Security Concepts

Introduction of some basic concepts and terminology in computer security

Next hour(s)...

- A bit of history...
- Application layer security
- Security evaluation
- Network security
- Identification & Authentication

Computer Security Goals

Traditional Security Goals: "CIA"

- Confidentiality
  - Keeping secrets
- Integrity
  - Preventing unauthorised modifications
  - Keeping data consistent
- Authentication/Assurance/Availability
  - The meaning of the “A” depends upon who you ask!

Services and mechanisms

- From the old “OSI” world, we've inherited two (sometimes confusing) concepts:
  - Security service: provides a security function to the system, based on the use of security mechanisms, e.g. confidentiality
  - Security mechanism is a technique, or protocol etc. which can be used to provide a service, e.g. encryption, access control

Services/Mechanisms

- Confidentiality
  - Access control
  - Cryptography
  - Secure communication channels
- Integrity
  - Checksums
  - Digital signatures
  - Secure communication channels
- Availability
  - Designing secure software and communication protocols
  - Load balancing
  - Anti-DDoS services

Structure of Military Security

- Information is classified according to national security
- The classification is clearly labeled on the binder
- All classified information is stored in a safe
- All users are “cleared” to see information up to a certain level
- Users have to prove their clearance to withdraw the binder from the safe
- Additional compartments enforce the *need-to-know* principle

Creating New Information

- New files are labelled with the classification of the most secret component
- Aggregation of unclassified information may generate a “top secret” file
- Sanitization downgrades the label of existing information

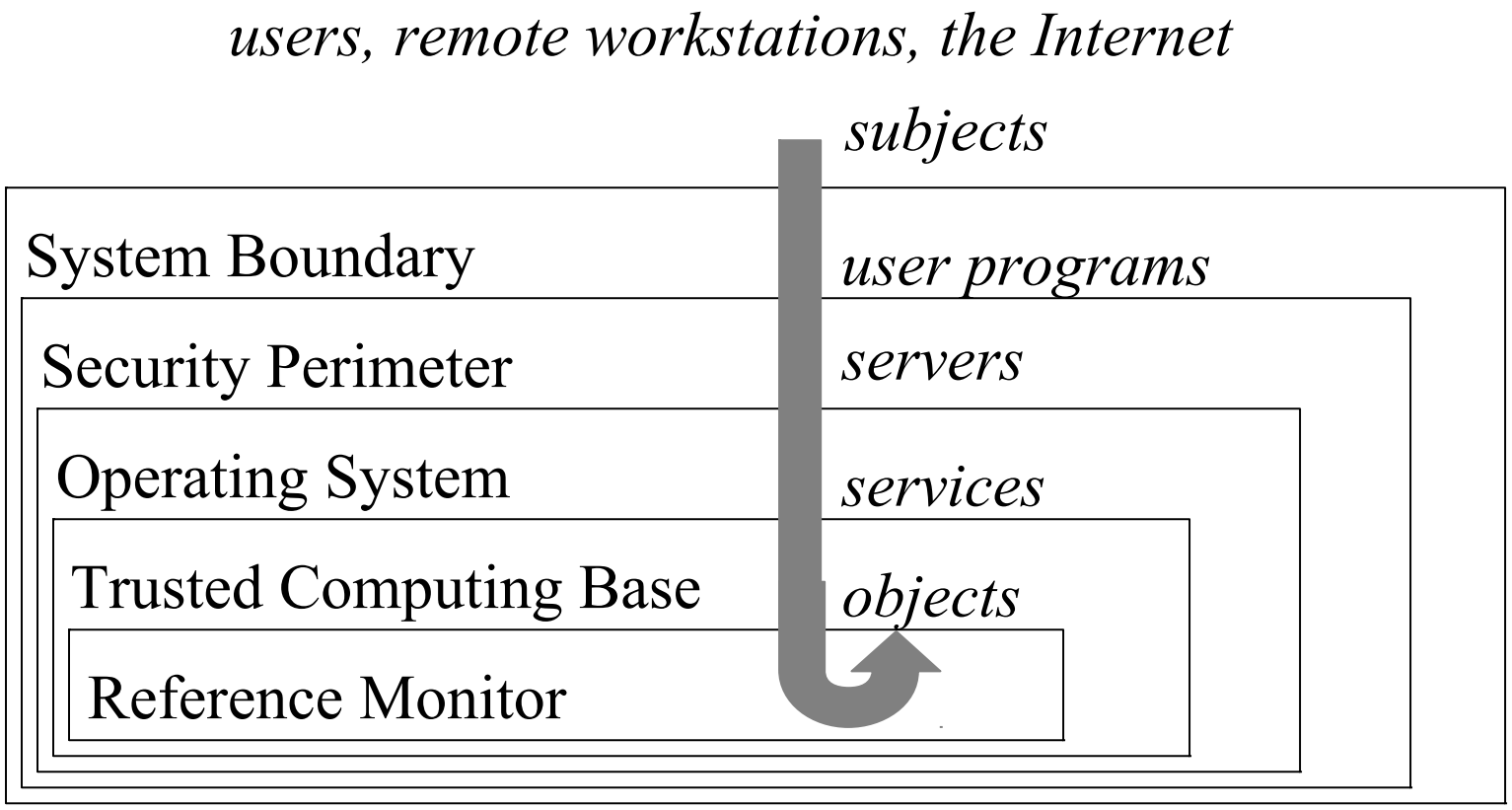
Old Problems Aggravated

- Aggregation
  - it is easier to aggregate information from a vast set of information
- Authentication
  - it is more difficult for a computer to identify a person (partly solved)
- Browsing
  - it is easier to read all files in a file system than all files in a safe
- Integrity
  - modification is harder to detect (mostly solved)
- Copying
  - digital copies are indistinguishable from the originals (really there are no originals at all!)
- Denial of Service
  - denial of service is a notorious characteristic of computer systems

Covert Channels

- Hidden means of communication, that allows information to be leaked to third parties
- Some types:
  - Timing Channels
    - observable differences in system utilization
  - Inference Channels
    - intersection of non-classified information
  - Fabrication Channels
    - aggregation of non-classified information

Security Perimeters



Sidebar: Operating systems security

- Won't cover this since its very system dependent,
  - But it will come up occasionally
- Basic idea is to separate processes (users are actually always represented as processes) and privileges
- OSes: UNIXes, incl. GNU/Linux, Apple iOS/OS X, Android, flavours of busybox, Windows and Cisco IOS, ...

Commercial Security

- Historically the military paid more for security, were more accepting of the inconvenience (which almost all security mechanisms introduce), and had a chain-of-command upon which to fall back
- Co-operating commercial enterprises have none of these

The 1988 Internet worm

- Propagated (mainly) via SMTP
  - using the DEBUG build of sendmail (then the prevalent mail server)
  - Also exploited fingerd problems
- It killed the Internet for a whole day!
- Spafford: “The Internet worm program: An analysis” well worth a read

https://down.dsg.cs.tcd.ie/cs7053/materials/spafford88internet.pdf

Typical Enterprise Security Model (circa 1995)

- System and network administrators setup and manage users and applications
- DCE exemplifies this approach
  - http://www.opengroup.org/dce/
    - Interesting that URL still works
  - Kerberos (RFC4120) – still persists as part of MSFT windows and a few other systems

Problems with the enterprise security model

- Generally assumed a homogeneous network and set of applications
  - Which was false
- Assumption that all users and applications are centrally managed
- Deployment showed up performance and usability issues

Security APIs

- Periodically someone tries to develop an API to “hide” security
- Has worked fine for cryptographic primitives
  - MS-CAPI, PKCS#11, JCE
  - Javascript WebCrypto API
- Not so successful for higher layer functions
  - GSS-API, SPKM

Sidebar: Crypto makes interop hard

- It is **much** easier to get a system to work in “insecure” mode, compared to “secure” mode
  - And **much, much** easier if secure mode involves cryptography
- This is a general problem which leads many people to turn off security
- Bear it in mind as you design things

Along came the web

- The number of connected hosts rose exponentially for a while
- Highlights security issues with:
  - Proxies (various bad things can happen at a proxy)
  - Tunnelling (“protocol-X” over HTTP)
  - Having a ubiquitous tunnel end-point on many machines
  - Browser security models

Security Evaluation and Network security

Customers need some confidence that the system or network that they are about to purchase is “secure”.

Security Evaluation

- How do we compare the security of different computer systems?
  - Different authentication mechanisms
    - Kerberos, passwords, smart cards, …
  - Different access control mechanisms
    - Access Control Lists, Role-Based Access Control, …
  - Different cryptographic algorithms
    - DES, AES, RSA, ECC, D-H, Curve25519…

Security Evaluation Criteria

- **[TCSEC]** Department of Defence Trusted Computer Systems Evaluation Criteria (Orange Book)
  - http://csrc.nist.gov/publications/secpubs/rainbow/
- **[ITSEC]** IT Security (UK, NL, FR, DE)
- **[Others] CTCPEC (Canada) JCSEC (Japan)**
- **[CC]** Common Criteria (ISO IS 15408)
  - http://www.cesg.gov.uk/publications/Documents/criteria.pdf

“Orange Book” Classification

Class	Title	Key Features
A1	Verified Design	Formal top-level specification and verification, formal covert channel analysis, informal code correspondence demonstration
B3	Security Domains	Reference monitor (security kernel), “highly resistant to penetration
B2	Structured Protection	Formal model, covert channels constrained, security oriented architecture, “relatively resistant to penetration”
B1	Labeled Security Protection	Mandatory access controls, security labeling, removal of security related flaws
C2	Controlled Access Protection	Individual accountability (authentication) and extensive auditing
C1	Discretionary Security Protection	Discretionary access controls, protection against accidents among cooperating users
D	Minimal Protection	Not Classified

ITSEC

- France, Germany, Holland and the UK
- Separation of functionality and assurance
- Functionality-classes (F-C1 – F-B3) corresponds to the Orange Book
- Assurance-classes (E0 – E6)
- Target of Evaluation
  - Set of evaluated components: security policy, security related functions, definition of the required security mechanisms for the target level of evaluation

Common Criteria

- Combines previous criteria
  - Separation of functionality and assurance
  - Target of Evaluation (TOE)
  - Security Target (ST) = desired level of evaluation
- Protection Profiles (PP)
  - Security Requirements
  - Security objectives
  - Independent of Implementation

Other aspects of assurance

- Installation and use must be considered
  - What are the defined security policies?
  - How is the software installed?
  - How are the machines administered?
- Development tools & environments
  - SNMP exploit due to buggy 3rd party ASN.1 handling library

Assurance != "Works-as-advertised”

- Assurance only shows that the product/system matches a specification
  - NOT that it does what the customer wants!
- The product owner pays the evaluator (lots!!)
  - so although the evaluator is Goverment licensed, they have reasons to be nice to the developer.
- Caveat emptor rules ok.

Network security

- Different from application layer security
  - Usually no real interest in APIs here
  - Normally trying to secure either a (sub)network, or network node from network based attacks
  - Frequently network nodes don't use standard operating systems
    - E.g. Cisco routers, 802.11 access points

Network Security

- First, read about networks, you need to have some level of understanding of IP (both), DNS, TCP, BGP, HTTP and how e.g. Javascript/PHP works on the web
  - Go find your own URLs:-) Or a book.
- Then, general network security:
  - https://www.sans.org/
  - https://isc.sans.edu/
  - https://first.org/library/
  - https://www.owasp.org/

Network security view

- Original Internet “architecture” assumed end-to-end connectivity
- Hence end-to-end security was the main consideration for those developing the Internet
  - But they were **very** slow developing IPsec (about 10 years!)
  - And meanwhile NAT and firewalls arrived

The End-to-End Argument

Saltzer, Jerome H., David P. Reed, and David D. Clark. "End-to-end arguments in system design." ACM Transactions on Computer Systems (TOCS) 2.4 (1984): 277-288.

http://dl.acm.org/citation.cfm?id=357402

READ THAT!!!

And then RFCs 1958 & 3439

And RFC 7258 (why not:-)

- caution, author present

E2E Argument has current consequences...

HTTP/2 and QUIC are protocols that attempt to have e2e encryption as a mandatory mechanism (ish) not in order to achieve a confidentiality service but primarily to mitigate ossification

Some, but not all, of those who know what's going on are happy with that

FWIW, (which is not much) I am happy with that.

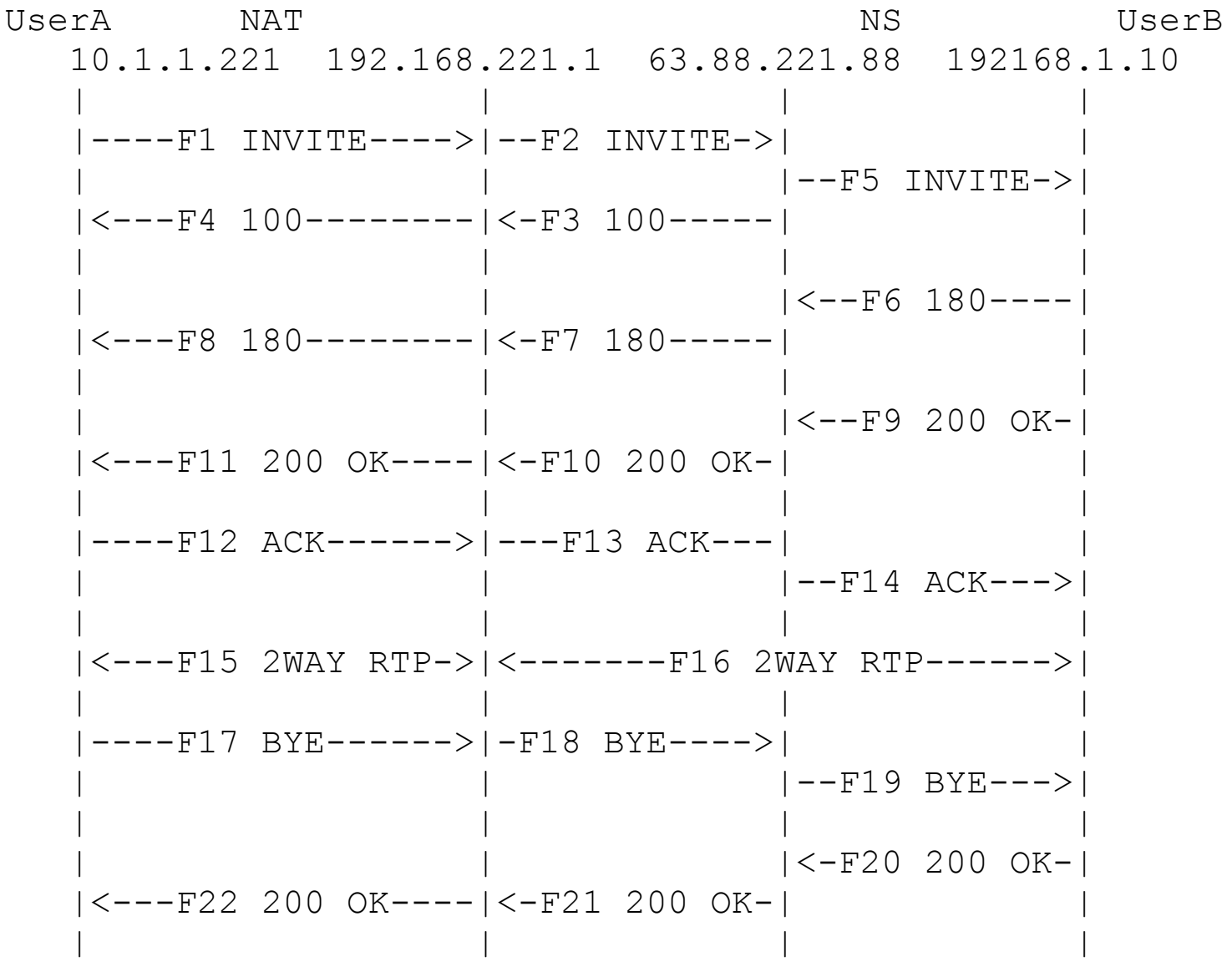
Network Address Translation (NAT)

- NAT is today mainly used to hide local addresses from the Internet for (mostly) provisioning reasons
  - NAT means that the higher layer end-points “see” different addresses for one another
  - Breaks many end-to-end assumptions
- “Carrier Grade” NAT (CGN) is coming to an ISP near you soon (or has already)

CGN & Security

- Many more endpoints behind each public IPv4 address; put a NAT box in the ISP network
  - NAT444
  - There are other, maybe better, IPv6 transition mechanisms
  - Some think this is an IPv6 avoidance mechanism
- Good way to freak out “legal intercept” (LI) fans
  - LI+CGN: Good way to freak out ISPs who have to log
- Good way to freak out anyone who wants to offer a public service
  - PITA for DNSSEC
- Bad way to get to an IPv6 world?
  - But may be needed for some time

NAT problems with SIP



Firewalls

- Until the mid-90's most sites didn't bother filtering traffic
- It became clear that exposing your internal network topology could cause problems
  - E.g. If I know that router brand “X” has an open default configuration and I can see that you've got one of those, then my attack point is clear

Filtering routers

- Initially people put in filtering rules in border routers
  - E.g. “no packets with a destination on my inner network are allowed out”
- But, IP spoofing attacks meant that this wasn't sufficient
  - So firewall products developed
  - Note: has been biggest security products market, I've not checked recently

Denial-of-Service (DoS)

- Consume resources to make a service unavailable
  - Has been known as a vulnerability for many years
  - Began to be exploited in last decade on major Internet sites
    - DDoS attacks against the DNS roots
- DDoS = Distributed Denial of Service

TCP SYN flooding attack

- SYN packets with spoofed IP addresses cause the server to maintain state
- Flooding the server that way is your basic DoS attack
  - CERT Advisory CA-96.21
    - https://www.cert.org/historical/advisories/CA-1996-21.cfm

Network access protocols

- Issue is how to decide when to allow a node to join/use the network
  - IEEE 802.1x networks (eduroam)
  - Corporate networks
  - Mobile operator networks
- RADIUS protocol (or Diameter)
  - RFC2865 (or RFC6733)

Identification and Authentication

Establishing identity and verifying credentials

Identification

- Establish the identity of principals by means of:
  - something known (*password, PIN*)
  - something possessed (*smart card, Java ring*)
  - something personal (*fingerprint, retina scan*)
  - something to do (*signature*)
- Combinations of above (*smart card* + *PIN*)

Passwords

- Most systems rely exclusively on passwords

login: *username*

password: ********

- Password scheme problems:
  - users choose “bad” passwords (*password, secret*)
  - compromised via sniffing
- Good passwords won't be in a dictionary
- There are better password protocols
  - but first we need some more crypto

2008 Count of some of my passwords

Category	Count	Known	Examples
Logins	10	6	Laptops, host systems (incl. root accounts)
Devices	5	0	DSL router, home print/file servers, sensor nodes
Network access	4	0	Work n/w, WLAN, ISP, etc
Protocol	14	1	Outbound HTTP proxy, IMAP, Jabber, skype, etc.
Service	21	4	Mainly web sites with password stored outside browser

Kerberos Authentication

- Project Athena at MIT (mid to late 1980s)
- Hundreds of diskless workstations
  - open terminal access, no physical security
  - insecure network
- Few servers (files, print, mail, …)
  - physically secure

Simple Authentication

- A password per service is infeasible
- New authentication service (AS) introduced
- Both users and services have passwords
- AS identifies user by password
- AS/TGS issues a “ticket” to the user
  - ticket includes identity encrypted with the service’s password
  - if the ticket decrypts properly, access to the service is granted
  - TGS = ticket granting service

Insecure Workstations

- What happens to tickets after a user has logged out?
  - an opponent could log on to the workstation and use the tickets
  - could be explicitly destroyed when user logs out
  - sniffer could be used to capture tickets, hacker may then login to the same workstation and use the tickets (*replay session*)

Biometrics

- Many people (apparently) think that replacing username & password with a biometric is a great idea.
- It might be:
  - If secure biometrics exist
  - If the re-use implications are acceptable
  - If the context supports the full life-cycle

Biometric methods

- Fingerprint
- Retina scan
- Face recognition
- Gait (walking)
- Toe-smell

Fingerprint

- Prof. Tsutomu Matsumota's well publicisied 2002 attack against most common fingerprint recognition engines
  - https://cryptome.org/gummy.htm
  - Information here is directly from his paper: “Impact of Artificial "Gummy" Fingers on Fingerprint Systems”

Fingerprint scanner

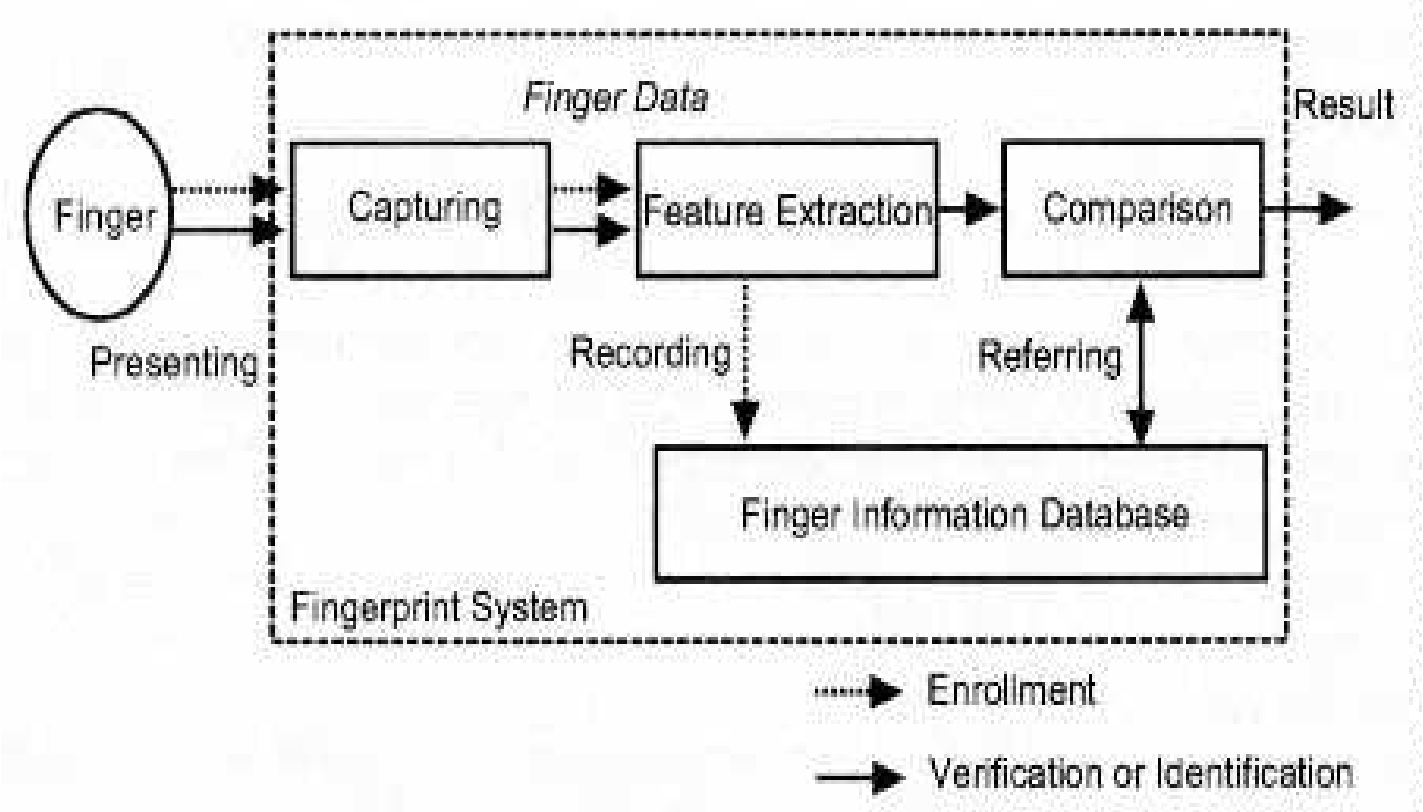
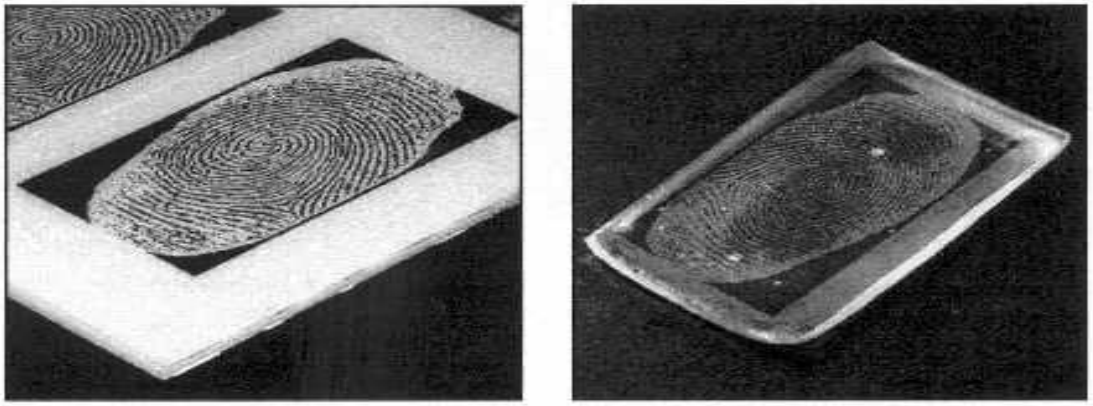


Figure 2.1 Typical structure of a fingerprint system

Make a gummy finger



(a) The mold for gummy fingers (b) Gummy finger

Figure 4.5 Photographs of the outside appearance of the mold and a *gummy* finger. The *gummy* finger was produced from a residual fingerprint on a glass plate, enhancing it with a cyanoacrylate adhesive

Figure 4.6 The Fingerprint image of the *gummy* finger, which was displayed by the system with Device H (equipped with a capacitive sensor).

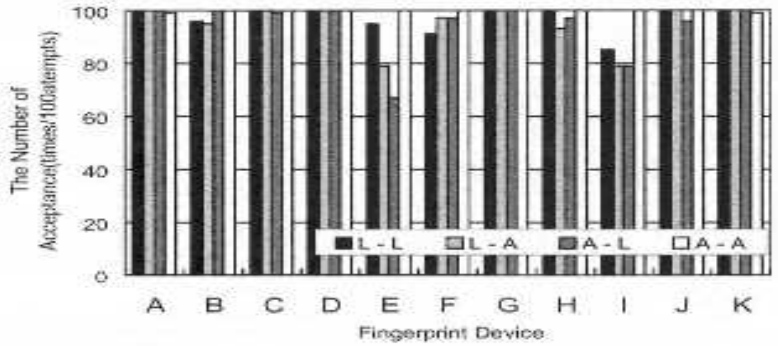


Figure 4.7 Average number of acceptance for each device, in terms of *gummy* fingers which were cloned from residual fingerprints. Here, the subject is one person.

Results

- 11 of 11 scanners tested were broken
  - The list was extended a bit more since
- That's enough of a result!

But there’s more! ...

CCC 2017: 55 minute video of breaking biometrics

https://www.youtube.com/watch?annotation_id=annotation_2684251971&feature=iv&src_vid=pIY6k4gvQsY&v=VVxL9ymiyAU

Fiebig, Tobias, Jan Krissler, and Ronny Hänsch. "Security Impact of High Resolution Smartphone Cameras." WOOT. 2014.

https://www.usenix.org/system/files/conference/woot14/woot14-fiebig.pdf

When are biometrics ok?

- What applications?
  - What type(s) of biometric?
  - What benefits?
  - What costs?
    - Financial and other (e.g. Privacy)

Privacy

- Rad RFC6973 - In addition to “normal” security threats we need to care about
  - Correllation
  - Identification
  - Secondary use
  - Disclosure
  - Exclusion
  - **Re-identification**
- Actually, it's not entirely clear to me that the risk analysis methodology we follow for security works well for privacy
  - There's scope for research there

Privacy Puzzle

- Emails contain a Received header field which can contain the mail user agent IP address
  - What consequences?
  - Overall good or bad from a privacy perspective?

What is your #1 mitigation for all problems?

What is your #1 mitigation for all problems?

Yes, Backup.

Do that. Early and often.

Summary

- Services vs. Mechanisms, CIA
- ~30 year history
- Common Criteria
- N/W security a bit different from system security
- Firewalls etc.
- Passwords (yuk)
- Biometrics (also yuk)
- Privacy (yay!)
- Do backups