This week (Mon + Wed)

- Security (Lesson 11)
  - ⇒ * Terminologies
- Security implementation in AFS

24 %

24 %

Exam week 100 %

Lesson outline

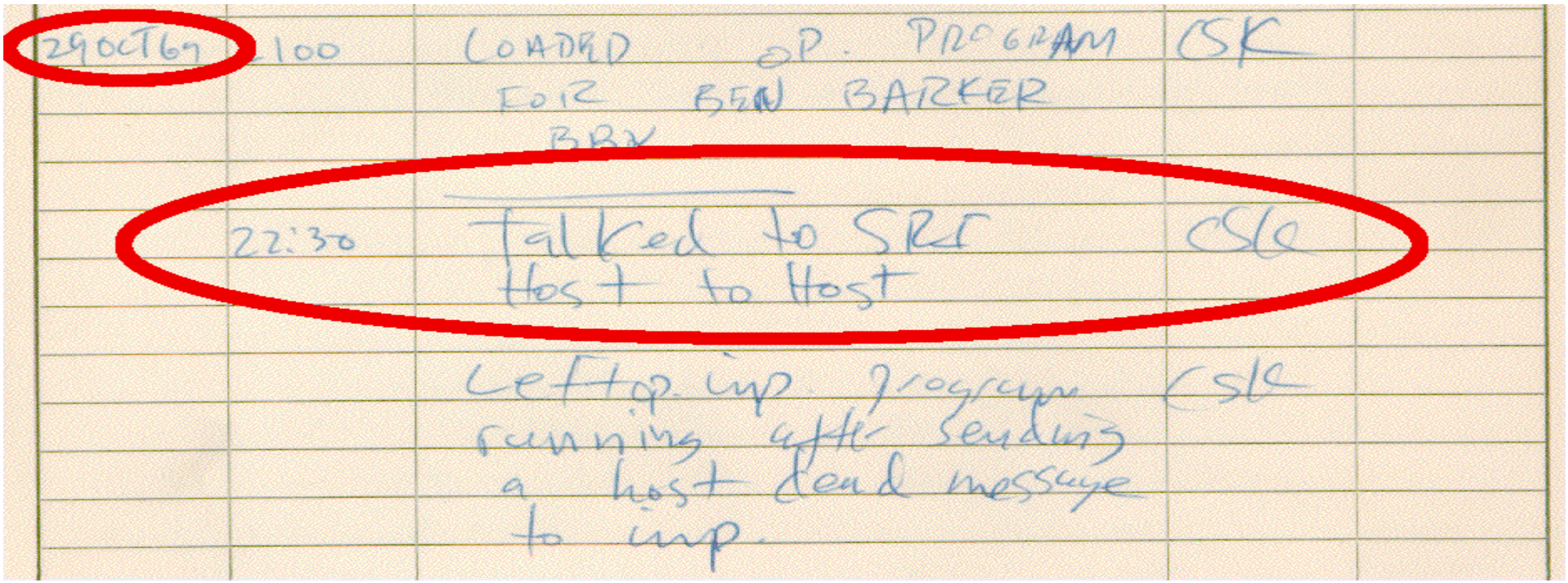
→ Terminologies

Security and authentication using a Private Key infrastructure

"Firsts" from Computing Pioneers

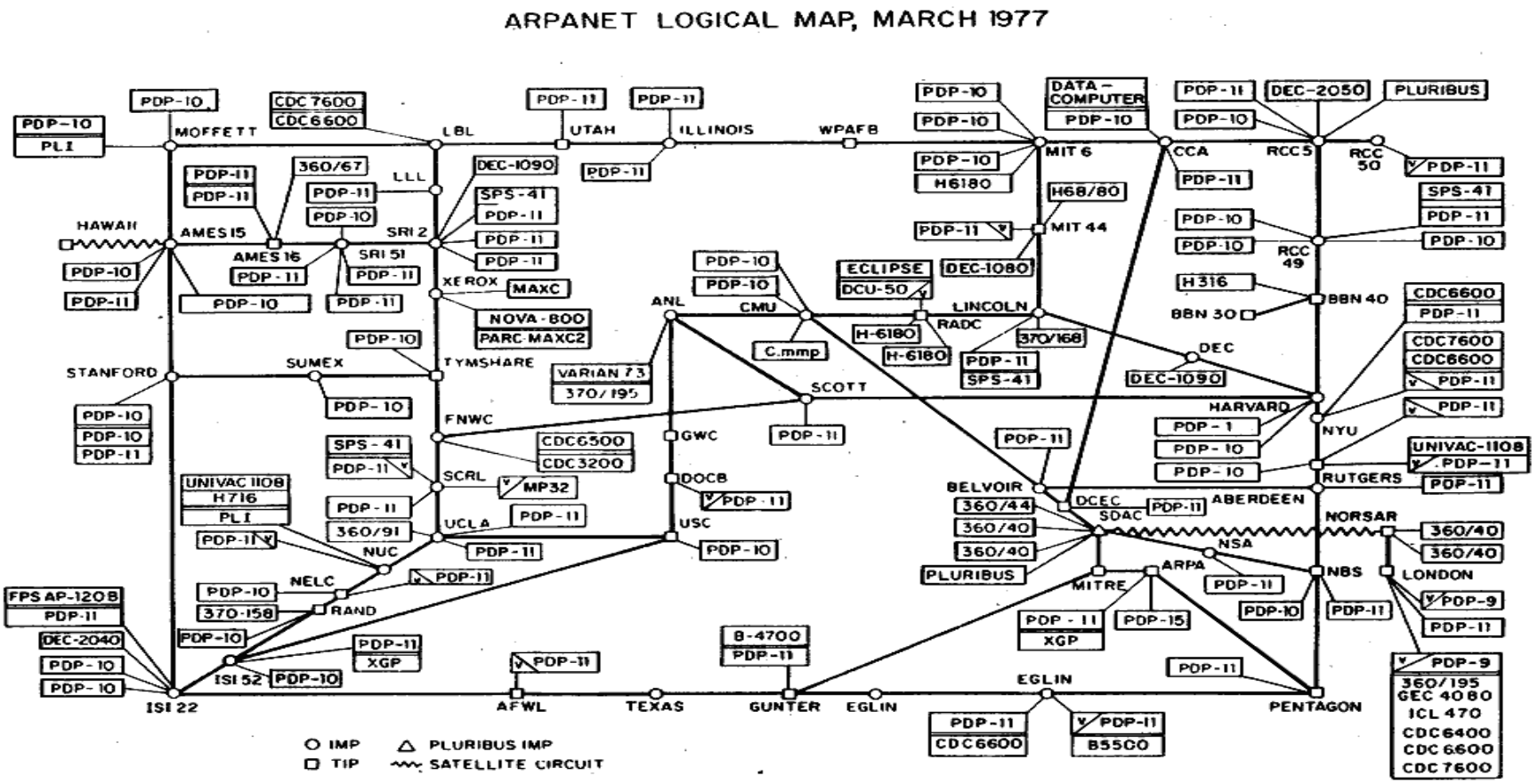
- First vision for a network of computers
  - Memorandum For Members and Affiliates of the Intergalactic Computer Network, by J. C. R. Licklider, April 23, 1963
- First computer-computer communication
  - The Day the Infant Internet Uttered its First Words, Oct 29, 1969
- First e-mail
  - Ray Tomlinson, BBN, 1971

"First" Computer – computer Communication



Log Book of Prof. Kleinrock, UCLA, Oct 29, 1969

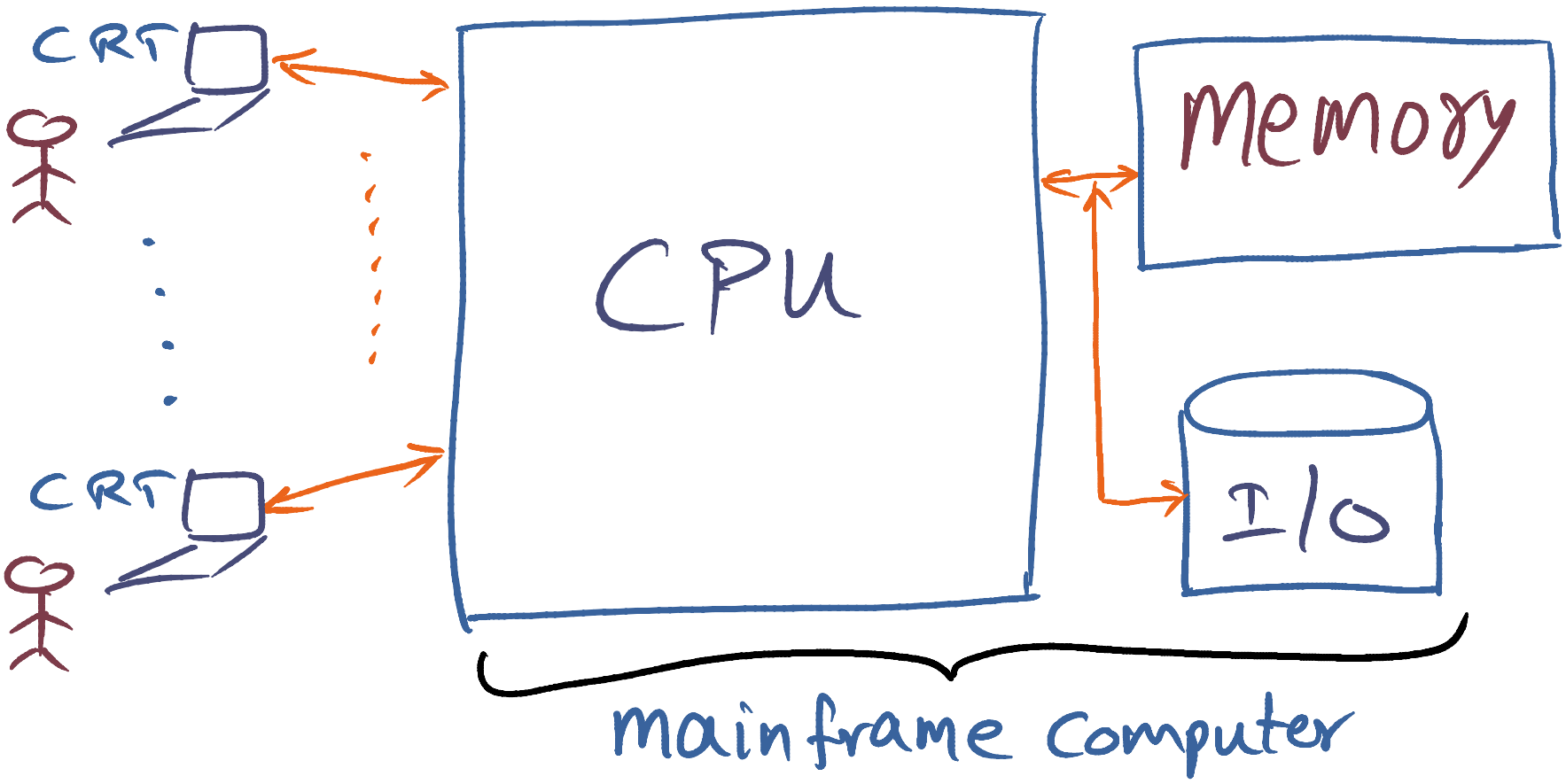
All the Computers in the World in one chart!!



(PLEASE NOTE THAT WHILE THIS MAP SHOWS THE HOST POPULATION OF THE NETWORK ACCORDING TO THE BEST INFORMATION OBTAINABLE, NO CLAIM CAN BE MADE FOR ITS ACCURACY)

NAMES SHOWN ARE IMP NAMES, NOT (NECESSARILY) HOST NAMES

Computer system model, 1975



Terminologies

When to release info?

- privacy vs. security
  - Protection
  - authentication

Terminologies

When to release info?

- privacy vs. security
  - Protection
  - authentication

Comprehensive set of security Concerns

- Unauthorized information release
- unauthorized information modification

Terminologies

When to release info?

- privacy vs. security
  - Protection
  - authentication

Comprehensive set of security Concerns

- Unauthorized information release
- unauthorized information modification
- unauthorized denial of use

Terminologies

When to release info?

- privacy vs. security
  - Protection
  - authentication

Comprehensive set of security Concerns

- Unauthorized information release
- unauthorized information modification
- unauthorized denial of use

⇒ First mention of DOS attacks

Terminologies

When to release info?
- privacy vs. security
  - Protection
  - authentication

Comprehensive set of security Concerns
- Unauthorized information release
- unauthorized information modification
- unauthorized denial of use

⇒ First mention of DOS attacks

Goal of secure System
- Prevent all violations

Terminologies

When to release info?
- privacy vs. security
  - Protection
  - authentication

Comprehensive set of security Concerns
- Unauthorized information release
- unauthorized information modification
- unauthorized denial of use

⇒ First mention of DOS attacks

Goal of secure System
- Prevent all violations ⇒ negative statement !

Terminologies

When to release info?
- privacy vs. security
  - Protection
  - authentication

Comprehensive set of security Concerns
- Unauthorized information release
- unauthorized information modification
- unauthorized denial of use

⇒ First mention of DOS attacks

Goal of secure System
- Prevent all violations ⇒ negative statement !
  - ↓ hard to achieve

Terminologies

When to release info?
- privacy vs. security
  - Protection
  - authentication

Comprehensive set of security Concerns
- Unauthorized information release
- unauthorized information modification
- unauthorized denial of use

⇒ First mention of DOS attacks

Goal of secure System
- Prevent all violations ⇒ negative statement !
  - * false sense of security
  - ↓ hard to achieve

Levels of Protection

Unprotected

e.g. MS-DOS ... Chooks for mistake prevention
not same as security

Levels of Protection

Unprotected

e.g. MS-DOS ... (hooks for mistake prevention not same as security)

All or nothing

e.g. VM-370 + most time-sharing systems

Levels of Protection

Unprotected

e.g. MS-DOS ... Chooks for mistake prevention
not same as security

All or nothing

e.g. VM-370 + most time-sharing systems

Controlled sharing

e.g. access lists for files

Levels of Protection

Unprotected
- e.g. MS-DOS ... (hooks for mistake prevention not same as security)

All or nothing
- e.g. VM-370 + most time-sharing systems

Controlled sharing
- e.g. access lists for files

User programmed sharing controls
- e.g. Unix like semantics for files

Levels of Protection

Unprotected
- e.g. MS-DOS ... Chooks for mistake prevention (not same as security)

All or nothing
- e.g. VM-370 + most time-sharing systems

Controlled sharing
- e.g. access lists for files

User programmed sharing controls
- e.g. Unix like semantics for files

Strings on into
- e.g. "TOP Secret"

Need to deal with dynamics of use

Design Principles

Economy of mechanisms $\Rightarrow$ easy to verify

Design Principles

Economy of mechanisms ⇒ easy to verify

Fail safe defaults

- explicitly allow access ⇒ default should not be no access

Design Principles

Economy of mechanisms ⇒ easy to verify

Fail safe defaults
- explicitly allow access ⇒ default should not be no access

Complete mediation ⇒ caching passwords.... bad idea

Design Principles

Economy of mechanisms ⇒ easy to verify

Fail safe defaults
- explicitly allow access ⇒ default should not be no access

Complete mediation ⇒ caching passwords.... bad idea

Open Design ⇒ protect keys publish design

Design Principles

Economy of mechanisms ⇒ easy to verify

Fail safe defaults
- explicitly allow access ⇒ default should not be no access

Complete mediation ⇒ caching passwords.... bad idea

Open Design ⇒ protect keys publish design

Separation of privilege: e.g. two keys to open vault

Design Principles

Economy of mechanisms ⇒ easy to verify

Fail safe defaults
- explicitly allow access ⇒ default should not be no access

Complete mediation ⇒ caching passwords.... bad idea

Open Design ⇒ protect keys publish design

Separation of privilege: e.g. two keys to open vault

Least Privilege ⇒ "need to know" based controls

Design Principles

Economy of mechanisms ⇒ easy to verify

Fail safe defaults
- explicitly allow access ⇒ default should not be no access

Complete mediation ⇒ caching passwords.... bad idea

Open Design ⇒ protect keys publish design

Separation of privilege: e.g. two keys to open vault

Least Privilege ⇒ "need to know" based controls

Least common mechanism: e.g. library vs in-kernel

Design Principles

Economy of mechanisms ⇒ easy to verify

Fail safe defaults
- explicitly allow access ⇒ default should not be no access

Complete mediation ⇒ caching passwords.... bad idea

Open Design ⇒ protect keys publish design

Separation of privilege: e.g. two keys to open vault

Least Privilege ⇒ "need to know" based controls

Least common mechanism: e.g. library vs in-kernel

Pyschological acceptability ⇒ good UI

Design Principles

Economy of mechanisms ⇒ easy to verify

Fail safe defaults
- explicitly allow access ⇒ default should not be no access

Complete mediation ⇒ caching passwords.... bad idea

Open Design ⇒ protect keys publish design

Separation of privilege: e.g. two keys to open vault

Least Privilege ⇒ "need to know" based controls

Least common mechanism: e.g. library vs in-kernel

Pyschological acceptability ⇒ good UI

Takeaways:
- Difficult to crack protection boundary
- Detection rather than prevention

Key takeaways

- classic information security issues relevant to this day.
- visionary work by computer pioneers