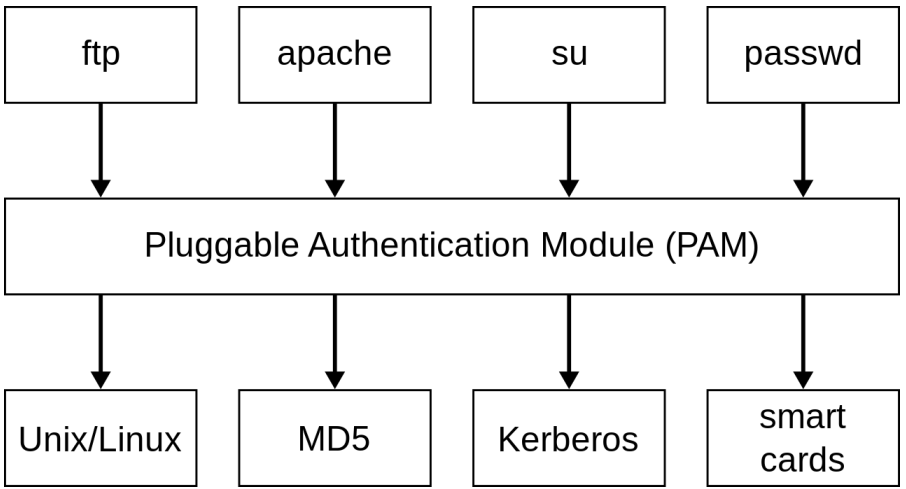
Pluggable Authentication Modules (PAM)

Linux System Administration
Spring 2026

What is PAM?

- Common authentication module used by many services to authenticate users
  - Allows fine grained control for user access
  - Single service to simplify authentication



Important Password Attributes

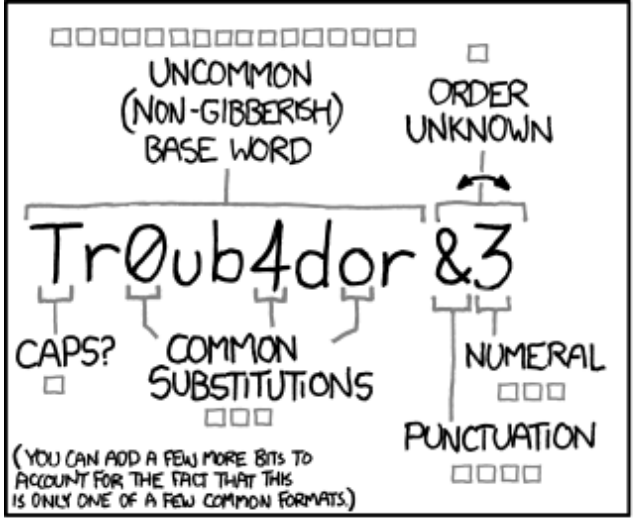
- Username
- Password hash (Method + Salt + Hash)
- Last date changed
- Maximum age
  - Warning time
  - Inactive time
  - Expire

Password Threats

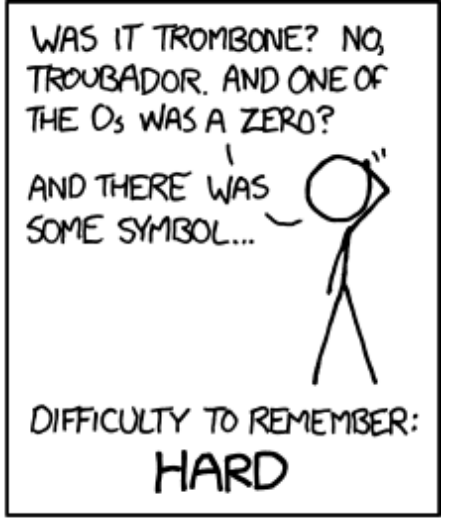
- Hashes from /etc/shadow or elsewhere
- Plain-text passwords in scripts
- Sniffing of insecure protocols
- Shoulder surfing
- Keyboard loggers (hardware or software)
- Brute force guessing or dictionary attacks

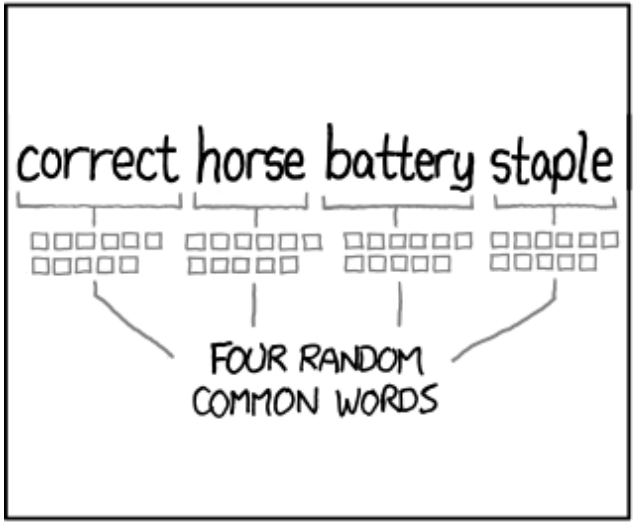
Mitigating password threats

- Exclude nonessential users from the machine
- Enforce passwords that are randomly generated, 12 characters or longer and from a set of 94 characters, symbols and numbers
- Using a key fob or application to generate key
- Two factor authentication
- Biometrics

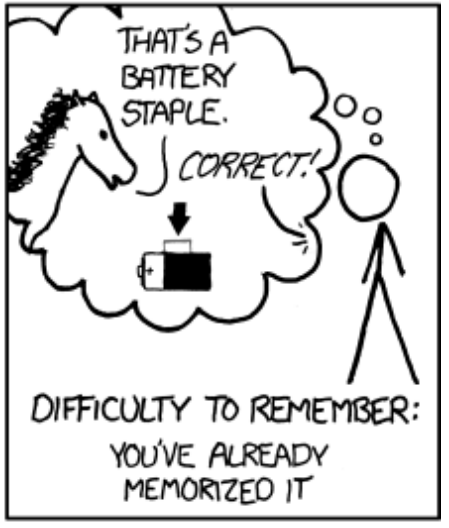












THROUGH 20 YEARS OF EFFORT, WE'VE SUCCESSFULLY TRAINED EVERYONE TO USE PASSWORDS THAT ARE HARD FOR HUMANS TO REMEMBER, BUT EASY FOR COMPUTERS TO GUESS.

Common sense password policy

- Enforce passwords that are strong enough, and educate them on selecting a good memorable password and force them to change it periodically.
- Use two factor authentication for critical systems

Password Expiration Example

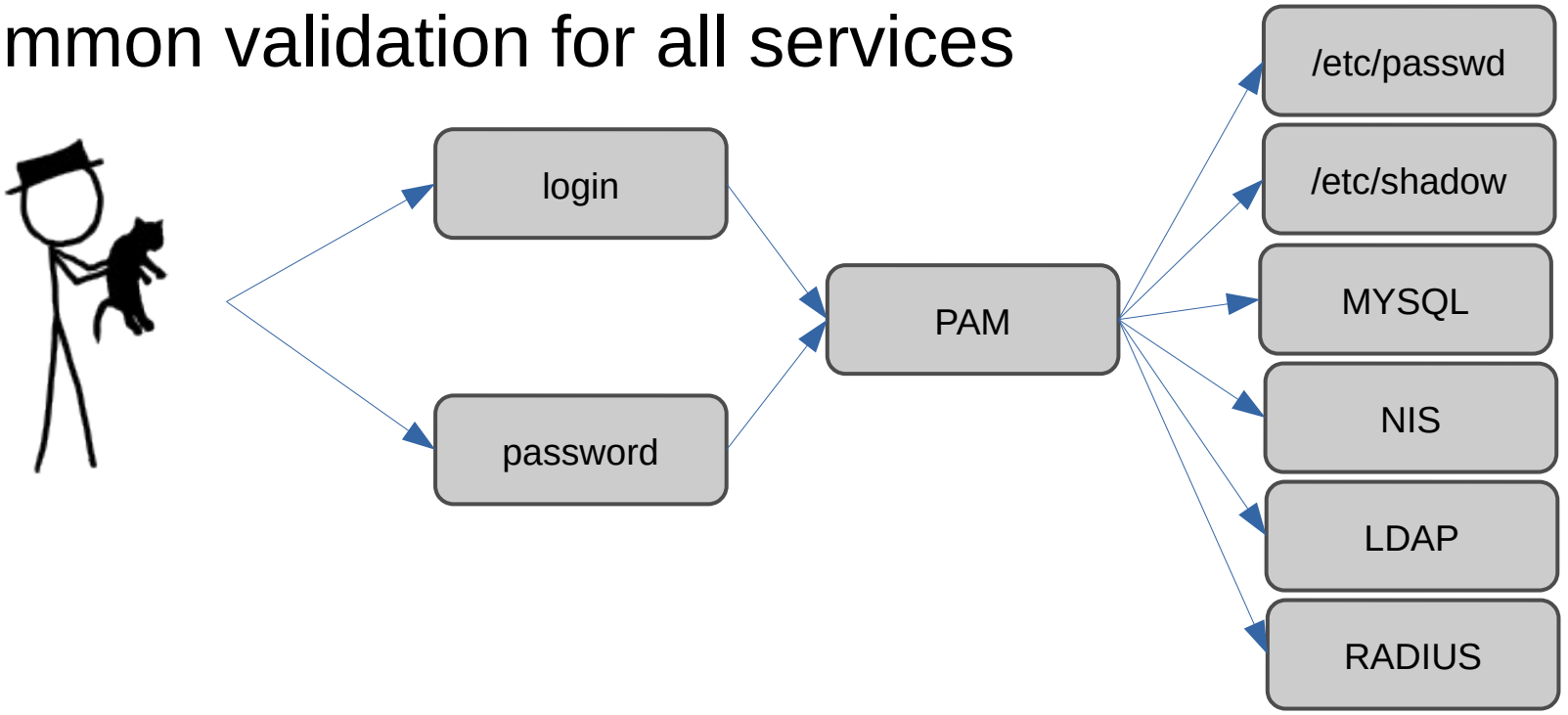
- `mscott:$6$88afba…:19245:0:90:10:7::`
  - `19245` = Sep 10, 2022
  - `90` = Maximum age (Dec 9, 2022)
  - `10` = Warning period (Nov 29, 2022)
  - `7` = Inactivity period (Dec 15, 2022)
  - account expiration period (never)

Setting password aging

- `chage` *`user`*
  - `-l` show current settings
  - `-d 0` force change at next login
  - `-E` set expiration date
  - `-l` lock account
  - `-u` unlock account

PAM login

- Pluggable Authentication Modules
- Common validation for all services



PAM Modules

- Authorization
  - Identifies users, groups and membership
- Account
  - Enforces login restrictions
- Session
  - Takes cares of tasks like creating or mounting home directory
- Password
  - Deals with password changes

PAM Files

- `/etc/pam.d`
  - PAM configuration files
- `/etc/security`
  - Policy files
  - `access.conf` - who can login from where
  - `pwquality.conf` - password requirements

Enable PAM Access

- RedHat
  - Files
    - `/etc/pam.d/system-auth`
    - `/etc/pam.d/password-auth`
  - Add after last account entry
    - `account required pam_access.so`
- Debian
  - Files
    - `/etc/pam.d/login`
    - `/etc/pam.d/sshd`
  - Uncomment

    `account required pam_access.so`

This enables `/etc/security/access.conf`

/etc/security/access.conf

- Format
  - Allow/Deny : user or group : from
- Evaluated in order

`+:root:ALL`	allow root from anywhere
`+:mscott:10.21.32.2`	allow mscott from machinee
`+:(sales):ALL`	allow group sales from anywhere
`-:ALL:ALL`	deny all others

PAM Password Quality Checks

- Is it a dictionary word?
- Is it a palindrome?
- Did only the case change?
- Is it too similar to the old one?
- Is it just a rotated version of the old one?
- Is it too simple?
- Are there more than 5 different characters?
- Does it contain the username or gecos?

/etc/security/pwquality.conf

- `minlen` minimum length
- `difok` difference from old
- `gecoscheck` no gecos
- `dictcheck` no words
- `usercheck` no username
- `maxrepeat` repeat limit
- `minclass` min # classes

- **Classes**
  - Positive is a max credit
  - Negative is a minimum
- `dcredit` digits
- `ucredit` upper case
- `lcredit` lower case
- `ocredit` other

Password quality examples 1

- Required
  - `dcredit = 2`
  - `lcredit = 0`
  - `minlen = 10`
- Proposed
  - `r2wrexac` too short (8+1)
  - `r2wrexa1` OK (8+2)
  - `r2wre32` too short (7+2)

- Required
  - `dcredit = -2`
  - `lcredit = 0`
  - `minlen = 10`
- Proposed
  - `r2wrexacpw` only one digit
  - `r2wrexa1` too short
  - `r2wrexa32r` OK

Password quality examples 2

- Required
  - `dcredit = 2`
  - `ucredit = 1`
  - `lcredit = 0`
  - `minlen = 10`
- Proposed
  - `r2wrexaC` OK (8+1+1)
  - `r2wrexa1` OK (8+2)
  - `r2wRe32` OK (7+2+1)

- Required
  - `dcredit = -2`
  - `lcredit = 0`
  - `minlen = 10`
  - `minclass = 3`
- Proposed
  - `r2wrexacpw` only one digit
  - `r2wrexa32r` only 2 classes
  - `R2wrexa32r` OK