Password Cracker
Password Cracker

• Recover a user’s password
  – Compute password hashes by a hash function
  – The hash function is one way.
• Recover a password by doing an exhaustive search
  – Given a target hashed password, compare it to the hash of every possible password.
• Use MD5 as an *insecure* hash function
• Speed up password cracking
Part A: Cracking a Single Password

• Password has
  ▪ Six-character length.
  ▪ Lowercase alphabetic characters (a-z).

• Generate a candidate password
• Compute hash of the candidate password with MD5
• Compare the hashed password to a given hash value

• Do not generate passwords randomly
  ▪ $26^6 > 300,000,000$ passwords
Part B: Cracking a List of Passwords

• More efficient
• Compare the hash of a candidate password to a list of hashed passwords
  – List contains usernames and hashed passwords.
• Program outputs usernames and cracked passwords.
Part C: Cracking Passwords in Parallel

- Even faster
- Use POSIX threads
- Each thread generates a subset of candidate passwords.
  - Divide candidate passwords among four threads.
- You should not need to synchronize among threads.