Here is a **professionally formatted, sentence-by-sentence study breakdown** of the document **“Password Cracking Notes.docx”**, based on your latest upload. This version is structured for **direct pasting into Word**—minimal spacing, no reformatting needed and aligned with **CompTIA A+ 220-1102 (Core 2)** exam objectives (especially Objective 2.5).

**🔐 Password Cracking – Detailed Study Notes**

**CompTIA A+ 220-1102 | Domain 2.0: Security | Objective 2.5: Threats, vulnerabilities, and attacks**

**✅ 1. What Is a Password Analysis Tool?**

* A **password analysis tool** is used to **test password strength** and ensure that password policies are being followed.
* Another name for these tools is a **password cracker**.
  + Password cracker is **used for comparative analysis** to guess or break passwords systematically until the correct one is identified.
* Popular password crackers include:
  + **Cain & Abel**
  + **John the Ripper**

**🧪 2. Demo Setup Context**

* The instructor uses **John the Ripper** in a lab to crack the root password of a **Kali Linux** machine.
* Kali’s default root password is often **“toor”**, making it a common demonstration target.

**🔓 3. Four Main Password Cracking Methods**

**1. Password Guessing**

* This is a **manual approach**, often based on personal information.
* Example: If someone’s dog is named **Fluffy**, attackers might try fluffy0167 or fluffy17.
* Not a technical method, but can be **surprisingly effective**.
* Story example: A 9-year-old guessed the four-digit PIN to unlock a parent’s smartphone.
* Prevention: Use **stronger passwords or biometric authentication**.

**2. Dictionary Attack**

* Uses a **list of known passwords or words** to guess the correct one.
* These lists go beyond real dictionary words and include:
  + Common passwords
  + Leetspeak or modified words (e.g., p@ssw0rd)
* **John the Ripper** can use dictionary attacks.

**3. Brute-Force Attack**

* Tries **every possible combination** until the correct password is found.
* Very slow, but always effective **given enough time**.
* Example: A 4-digit PIN (like 8157) could be cracked in minutes by a computer.
* Efficiency increases with password complexity:
  + Adding uppercase, lowercase, symbols, and numbers dramatically increases difficulty.
* Prevention: Use **long, complex passwords**.

**4. Cryptanalysis Attack (Rainbow Table)**

* Uses a **precomputed lookup table** of password hashes.
* Example: Instead of calculating each hash, the attacker compares the target’s hash to entries in a **rainbow table**.
* If a match is found, the plain text password is revealed from the table.
* Example resource: **crackstation.net** hosts a rainbow table with:
  + **15 billion entries**
  + File size: **190 GB**
* Prevention: **Salting** passwords before hashing makes rainbow tables ineffective.

Absolutely — let’s break down **Cryptanalysis Attack** in a way that’s simple, practical, and exam-ready for **CompTIA A+ 220-1102**.

**🔓 What Is a Cryptanalysis Attack? (Easy Definition)**

A **cryptanalysis attack** is when a hacker tries to figure out a **password** by analyzing its **hashed version** (the scrambled version stored by systems) — often using **precomputed lookup tables** like **rainbow tables** to speed things up.

**✅ Even Simpler:**

It’s like having a **cheat sheet** that matches every possible password to its hashed version, so the attacker can look up the right password **instead of guessing it**.

**🧠 Real-Life Example:**

Let’s say someone’s password is **“P@ssw0rd”**. When it’s stored, it’s **hashed** into something like:

fa7d89...2be1

The attacker checks their **rainbow table** (a huge list of password–hash pairs).

They find that this hash **matches “P@ssw0rd”** — now they know the password **without guessing**.

**🔐 Why It Works:**

* Systems store **hashed** passwords for security.
* But if they don’t add a **salt** (random data), those hashes are predictable.
* Attackers build huge tables that link common passwords to their hash.

**🚫 How to Stop It:**

* Use **salting**: Add random text to the password **before hashing**.
  + Example: "P@ssw0rd" + "1xZ9" → hashed
* This makes each hash unique, so the **cheat sheet no longer works**.

**🧾 In Summary:**

| **Term** | **Simple Meaning** |
| --- | --- |
| Cryptanalysis Attack | Using data and math (like rainbow tables) to reverse-engineer passwords from their hashes |
| Rainbow Table | A pre-made list of passwords and their hashes |
| Salt | Random data added to make each hash unique |

**👀 Bonus: Fifth (Legacy) Method – Rubber Hose Attack**

* Not part of current exam objectives but mentioned for interest.
* The **rubber hose method** involves **coercion or physical threats** to force someone to reveal their password.
* Common in fiction (e.g., spy or military interrogation scenes).
* Not technical or ethical but historically referenced in textbooks for its **shock value and humor**.

**🛠️ 4. Demonstration: Using John, the Ripper**

**Step-by-Step Password Cracking on Kali Linux:**

* Target machine: Kali Linux (password: **toor**).
* **Passwords are stored in**:
  + /etc/passwd – holds user accounts
  + /etc/shadow – holds password hashes

**Extraction Process:**

1. Combine /etc/passwd and /etc/shadow into one file using:

unshadow /etc/passwd /etc/shadow > password.txt

1. View the new file using:

more password.txt

**Cracking the Password:**

1. Run John the Ripper:

john password.txt

1. Display the cracked password:

john --show password.txt

* Output shows:
  + Username: **root**
  + Password: **toor**
* Demonstrates how **quickly and effectively** John the Ripper can crack weak hashes.

**🧾 5. Summary Table – Password Cracking Methods**

| **Method** | **How It Works** | **Example** | **Prevention** |
| --- | --- | --- | --- |
| Password Guessing | Based on known info or behavior | Guessing pet name + birthdate | Avoid predictable info, use biometrics |
| Dictionary Attack | Uses list of known/modified passwords | Using wordlists like 123456, etc | Complex, unique passwords |
| Brute-Force Attack | Attempts all possible combinations | Trying all 4-digit PINs | Long, complex passwords |
| Cryptanalysis Attack | Matches password hash to known hash in rainbow table | CrackStation’s 190GB table | Hash + Salt, modern hashing methods |
| Rubber Hose (Legacy) | Coercion through physical intimidation | “Tell me or I’ll hit you!” | Not technical; mentioned for context |

**📘 6. CompTIA A+ 220-1102 Exam Relevance**

✅ **Yes – Password cracking is a required topic under Objective 2.5: Security threats and attacks.**

You must be able to:

* Identify types of password cracking
* Understand their methods and risks
* Recommend appropriate prevention techniques (e.g., salting, complex passwords, MFA)

Based on the detailed content from **“Password Cracking Notes.docx”**, here is a **10-question multiple-choice quiz** aligned with **CompTIA A+ 220-1102 (Core 2)** — specifically **Objective 2.5: Security threats, vulnerabilities, and attacks**.

This quiz is **professionally formatted** for direct pasting into Microsoft Word — clean alignment and minimal spacing.

**🔐 Password Cracking Quiz – CompTIA A+ 220-1102**

**Instructions:** Select the most correct answer for each question.

**1. Which of the following best defines a password analysis tool?**

a) A utility that creates default login credentials for network devices

b) A forensic tool used for system memory dumps

c) A tool that tests password strength or attempts to recover passwords

d) A browser extension that blocks weak passwords

**2. Which password cracking method involves trying every possible character combination until success?**

a) Dictionary attack

b) Password guessing

c) Brute-force attack

d) Hybrid attack

**3. What is a rainbow table used for in a cryptanalysis attack?**

a) Encrypting password data using salt

b) Storing login tokens for browser sessions

c) Matching hashed passwords to known plaintext passwords

d) Generating randomized passwords for user accounts

**4. What is the primary defense against rainbow table attacks?**

a) Strong antivirus software

b) Biometric authentication

c) Password complexity rules

d) Salting before hashing passwords

**5. In which attack does the attacker manually guess a user’s password using known personal information?**

a) Hybrid attack

b) Brute-force attack

c) Password guessing

d) Dictionary attack

**6. What makes brute-force attacks more difficult to execute?**

a) Reusing passwords across accounts

b) Using common dictionary words

c) Increasing password complexity and length

d) Disabling two-factor authentication

**7. Which of the following is true about dictionary attacks?**

a) They test random strings of letters and numbers

b) They only work if the password is stored in plaintext

c) They use lists of common passwords and words

d) They are slower than brute-force attacks

**8. What command is used to combine the /etc/passwd and /etc/shadow files for cracking in Kali Linux?**

a) john –shadow merge

b) cat /etc/shadow /etc/passwd > users.txt

c) unshadow /etc/passwd /etc/shadow > password.txt

d) shadowtool combine > password.db

**9. Which tool was used in the demonstration to crack the password “toor”?**

a) Cain & Abel

b) CrackStation

c) Hydra

d) John the Ripper

**10. What is the best reason to avoid using common passwords like “123456”?**

a) They take up too much system memory

b) They are immune to salting and hashing

c) They are easily found in dictionary or rainbow table attacks

d) They cannot be detected by password analysis tools

Thanks for submitting your answers! Here’s your score, a breakdown of the correct answers, and clear explanations to help you master the topic. This format is fully aligned for pasting into Microsoft Word — professional, minimal spacing, and well-organized.

**✅ Graded Answer Sheet – Password Cracking Quiz (CompTIA A+ 220-1102)**

| **#** | **Your Answer** | **Correct?** | **Correct Answer** | **Explanation** |
| --- | --- | --- | --- | --- |
| 1 | c | ✅ | c | A password analysis tool (aka password cracker) tests strength or attempts to recover passwords. |
| 2 | a | ❌ | **c** | Brute-force attacks try every possible combination. Dictionary attacks use known word lists. |
| 3 | c | ✅ | c | Rainbow tables contain precomputed hashes to match against stolen password hashes. |
| 4 | a | ❌ | **d** | Salting passwords before hashing ensures unique hashes, making rainbow tables useless. |
| 5 | c | ✅ | c | Password guessing involves trying passwords based on what the attacker knows about the user. |
| 6 | c | ✅ | c | Long, complex passwords increase the time needed for brute-force attacks, making them impractical. |
| 7 | c | ✅ | c | Dictionary attacks use precompiled lists of common passwords or modified word variations. |
| 8 | c | ✅ | c | The correct command is: unshadow /etc/passwd /etc/shadow > password.txt to prep for cracking. |
| 9 | d | ✅ | d | John the Ripper was used in the demo to crack the password “toor” on a Kali Linux machine. |
| 10 | c | ✅ | c | Common passwords are easily guessed or found in rainbow/dictionary lists — a major risk factor. |

**🎯 Final Score:**

**8 out of 10 – Strong performance!**

**❌ Missed Questions Review:**

**Question 2**

* **Your answer:** a
* **Correct answer:** **c**
* **Why:** Brute-force attacks try **all possible character combinations**. Dictionary attacks are faster but limited to known passwords.

**Question 4**

* **Your answer:** a
* **Correct answer:** **d**
* **Why:** Antivirus doesn’t protect password hashes. **Salting + hashing** makes rainbow tables ineffective by ensuring the same password generates a **unique hash per user**.

Would you like a **follow-up quiz**, **scenario-based questions**, or a **visual cheat sheet** on password attack types? Let me know how you’d like to reinforce your learning!