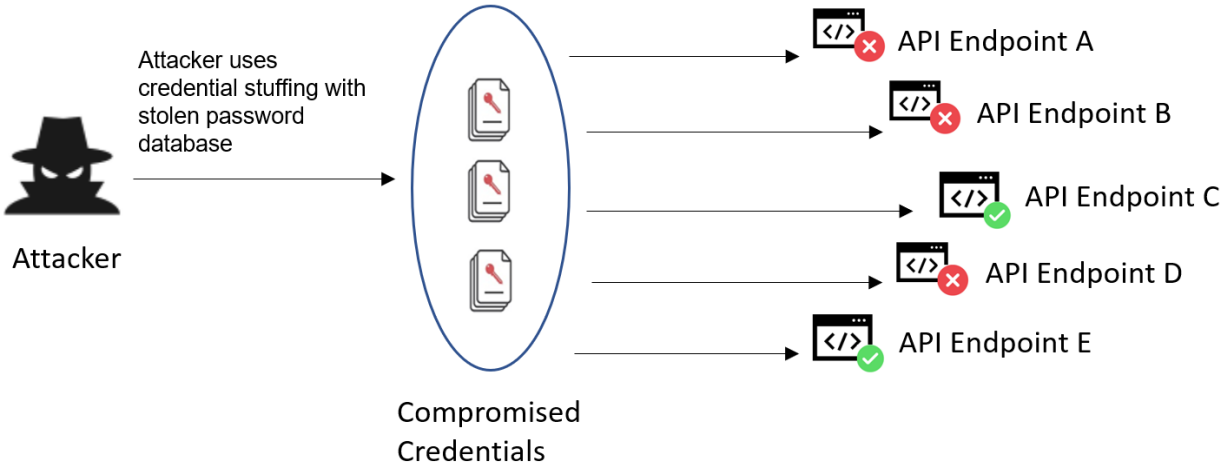
API PENTESTING



BROKEN AUTHENTICATION

BY HARSH PATEL

WHAT IS AUTHENTICATION?

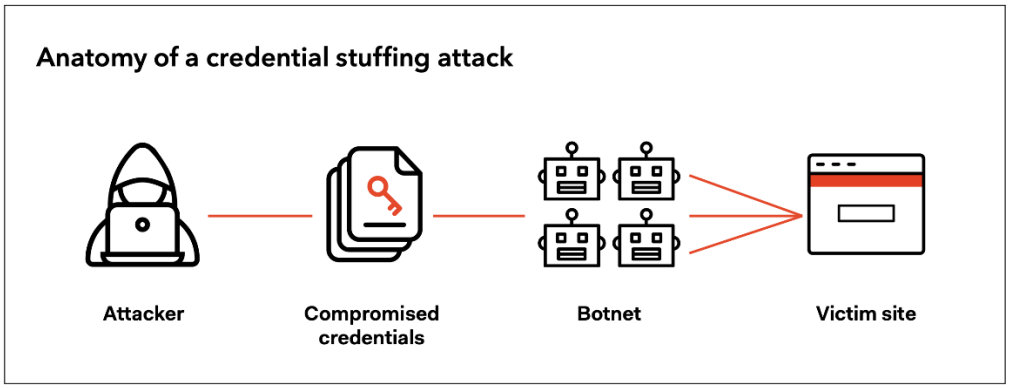
- Authentication is the process of determining whether someone or something is who or what they say they are. Authentication technology provides access control for systems by checking to see if a user's credentials match the credentials in a database of authorized users or a data authentication server. In doing this, authentication ensures that systems, processes and enterprise information are secure.

- There are several authentication types. For user identity, users are typically identified with a user ID; authentication occurs when the user provides credentials, such as a password, that match their user ID.

- The practice of requiring a user ID and password is known as single-factor authentication (SFA). In recent years, organizations have strengthened authentication by asking for additional authentication factors. These can be a unique code provided to a user over a mobile device when a sign-on is attempted or a biometric signature, such as a facial scan or thumbprint. This is known as two-factor authentication (2FA).

- Authentication protocols can go further than 2FA and use multiple factors to authenticate a person or system. Authentication methods that use two or more factors are called multifactor authentication (MFA).

WHAT IS BROKEN AUTHENTICATION?



- Broken authentication vulnerability arises when there is weakness in authentication and session management function that allows attackers to compromise passwords, keys, session tokens, and user account information, enabling then to assume user identities. The root cause of this vulnerability is poorly implemented authentication and session management functions. Attackers can use broken authentication to gain unauthorized access to one or more accounts, granting then the same privileges as legitimate user. Some common flaws which are actively exploited by attackers are:

    - **Brute force attacks:** If a web application uses usernames and passwords, an attacker is able to launch brute force attacks that allow them to guess the username and passwords using multiple authentication attempts.

    - **Use of weak credentials:** web applications should set strong password policies. If applications allow users to set passwords such as ‘password1’ or common passwords, then an attacker is able to easily guess them and access user accounts. They can do this without brute forcing and without multiple attempts.

    - **Weak Session Cookies:** Session cookies are how the server keeps track of users. If session cookies contain predictable values, an attacker can set their own session cookies and access users’ accounts.

TYPES OF BROKEN AUTHENTICATON

1. **Credential Stuffing :**
   ⇒ When attackers access a database filled with unencrypted emails and passwords, they frequently sell or give away the list for other attackers to use. These attackers then use botnets for brute-force attacks that test credentials stolen from one site on different accounts. This tactic often works because people frequently use the same password across applications.
   ⇒ There are currently billions of compromised credentials available to attackers. Most of the time, users don't even know that the password they've been using since high school just became a skeleton key for all their accounts.

2. **Password Spraying :**
   ⇒ Password spraying is a little like credential stuffing, but instead of working off a database of stolen passwords, it uses a set of weak or common passwords to break into a user's account. A 2019 survey by the UK's National Cyber Security Centre (NCSC) found that 23.2 million accounts used "123456" as their password, while millions more used sports names, curse words, and the ever-popular "password".
   ⇒ Password spraying is a type of brute-force attack, but it often slips by automatic lockouts that block IP addresses after too many failed login attempts. It does this by trying the same password, one user at a time, rather than trying password after password on a single user.

3. **Phising Attacks :**
   ⇒ Attackers typically phish by sending users an email pretending to be from a trusted source and then tricking users into sharing their credentials or other related information. It can be a broad-based attempt that hits everyone at an organization with the same phony email, or it can take the form of a "spear phishing" attack tailored to a specific target.

⇒ Spear phishing can be particularly useful to attackers. They can use that attack technique to manipulate someone's emotions based on their personal information. For example, an email with the subject line "pictures of your sister" is much more effective if it mentions your sister's name.

⇒ The 2020 CrowdStrike Services Report found that 35% of successful network breaches started with a spear phishing attack in 2019. Attackers had different mechanism to lure their victims through spear phishing: 19% used attachments, 15% included a malicious link, and 1% employed spear phishing via a service.

⇒ By now, most organizations are familiar with phishing attacks and warn their customers not to open suspicious emails. Despite that, the 2017 Phishing Resiliency and Defense Report by Cofense found that "organizational susceptibility" to phishing was still around 5%. If you have 20 employees, one of them might still click a phishing email.

EXAMPLE OF BROKEN AUTHENTICATION

1. **Use of Passwords as the Only Authentication Factor**
   - ⇒ Relying solely on passwords for user authentication is a significant vulnerability in web application security. Passwords, while being a traditional and widely used method for securing accounts, are often weak due to poor user practices such as using easy-to-guess passwords or reusing the same password across multiple sites. This vulnerability becomes more critical when additional layers of security, like multi-factor authentication (MFA), are not in place.
   - ⇒ Attackers exploit weak or reused passwords through various methods like phishing attacks, credential stuffing, or brute force attacks. Phishing attacks trick users into revealing their passwords, while credential stuffing uses previously leaked credentials to gain unauthorized access. Brute force attacks involve systematically checking all possible passwords until the correct one is found. When passwords are the only line of defense, any of these methods can lead to broken authentication, granting attackers access to user accounts and sensitive data.

2. **Application Session Timeouts Aren't Set Properly**
   - ⇒ Another common source of broken authentication vulnerabilities is improperly set application session timeouts. When a user logs into a web application, a session is established. This session should expire after a period of inactivity to prevent unauthorized access in case the user leaves their device unattended. If the session timeout is not properly set, it could allow an attacker to hijack the session and gain access to the user's account.
   - ⇒ Inadequate session timeouts can also lead to session fixation attacks, where an attacker induces a user to use a specific session ID, and then uses that same session ID to gain unauthorized access to the user's account.

3. Passwords Not Properly Hashed and Salted

⇒ Proper handling of user passwords is a crucial aspect of web application security. When passwords are not properly hashed and salted, it can lead to broken authentication. Hashing is a process that transforms a password into a unique, fixed-size string of characters, which is then stored in the system. Salting involves adding an additional, random string of characters to the password before it's hashed.

⇒ If an attacker manages to breach the system and gain access to the password data, and if the passwords are not properly hashed and salted, they could potentially crack the passwords using various methods such as brute force attacks, dictionary attacks, or rainbow table attacks. Once the attacker has the user's password, they can easily gain unauthorized access to their account, leading to broken authentication.

IMPACT OF BROKEN AUTHENTICATION

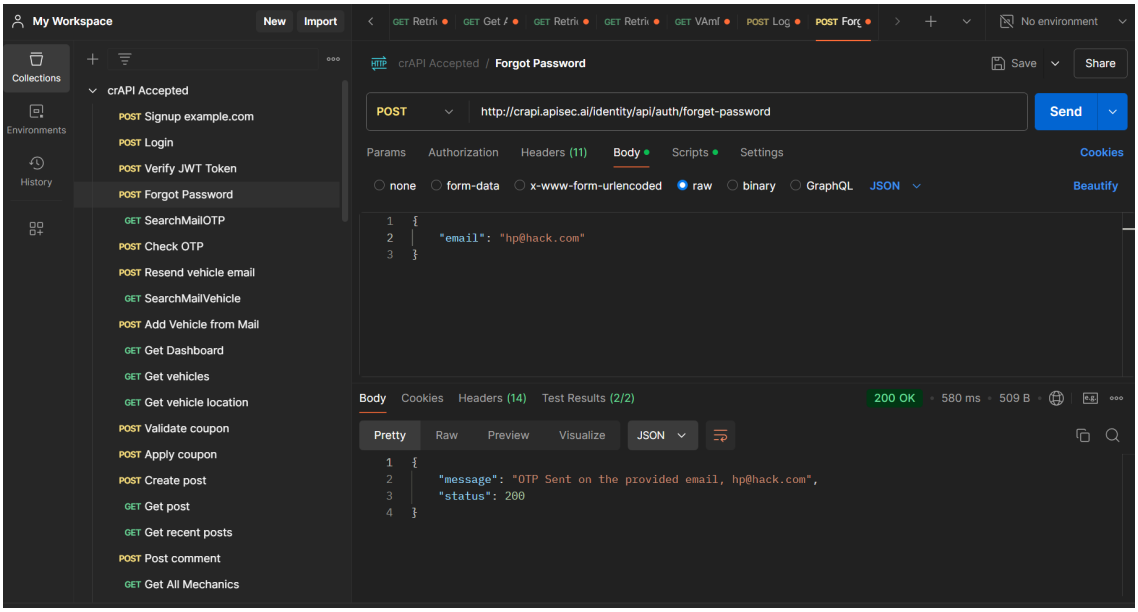
- The impact of broken authentication attacks can be devastating for both an organization and its customers. When attackers exploit these vulnerabilities, they gain unauthorized access to user accounts, personal data, sensitive business information, and more. This not only leads to a breach of privacy and potential financial losses but can also severely tarnish the reputation of the impacted organization.

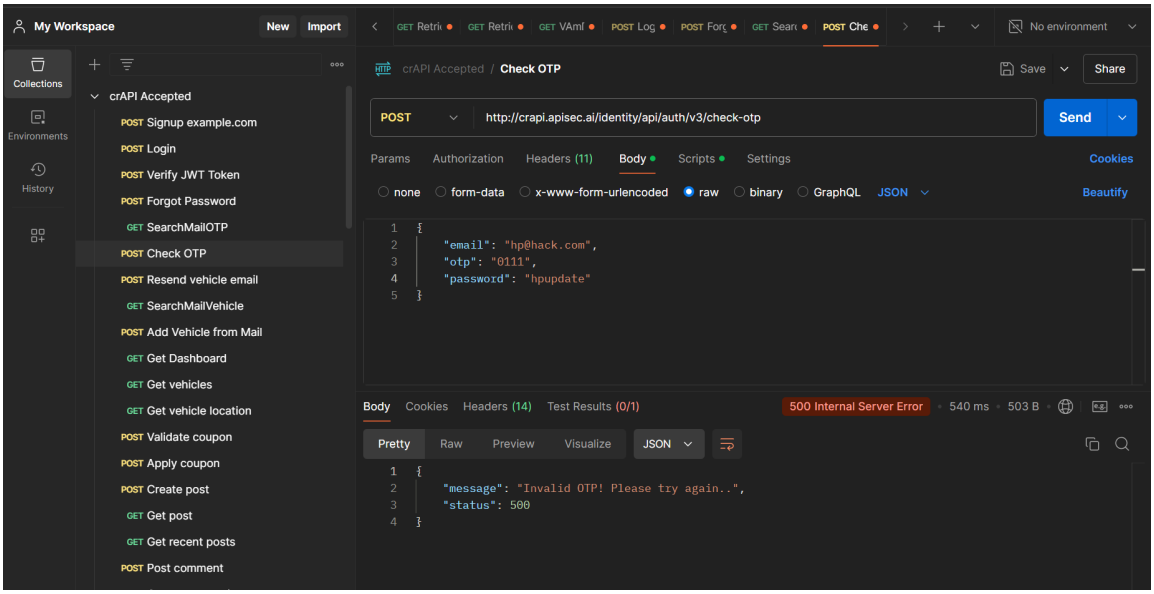
- For an end-user, a broken authentication attack could mean unauthorized access to their account, leading to the theft of sensitive personal data such as credit card information, social security numbers, and more. This could further result in identity theft, unauthorized transactions, and other forms of personal harm.

- For businesses, the consequences can be even more severe. A successful attack could potentially give cybercriminals access to privileged accounts, allowing them to manipulate data, perform malicious actions, or even take control of the entire system. This could lead to substantial financial losses, damage to the organization’s reputation, loss of customer trust, and potential legal implications.
    - Unauthorized access: Attackers can gain access to user accounts and sensitive information, such as financial details, personal data, or intellectual property.
    - Identity theft: Users can be exposed to identity theft and other cybercrimes.
    - Financial losses: Organizations can suffer financial losses.
    - Reputation damage: An organization's reputation can be tarnished.
    - Compromised systems: If the compromised account belongs to an administrator or privileged user, the entire system or network could be compromised.

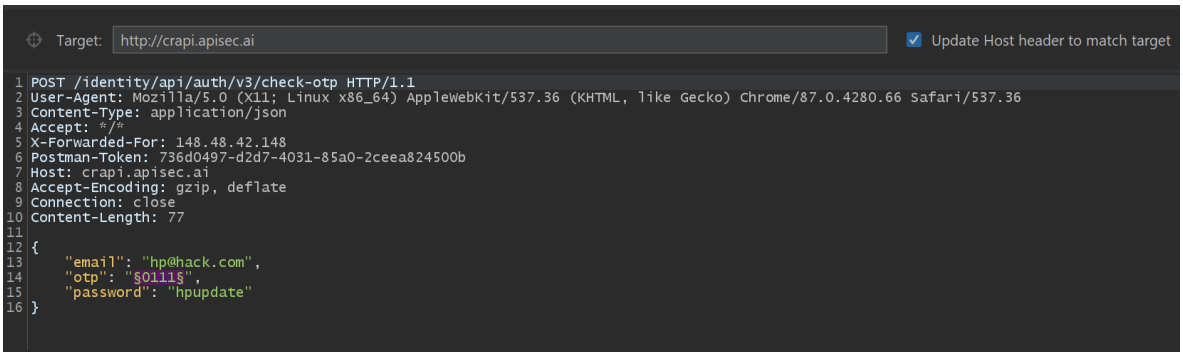
PRACTICAL LAB crAPI

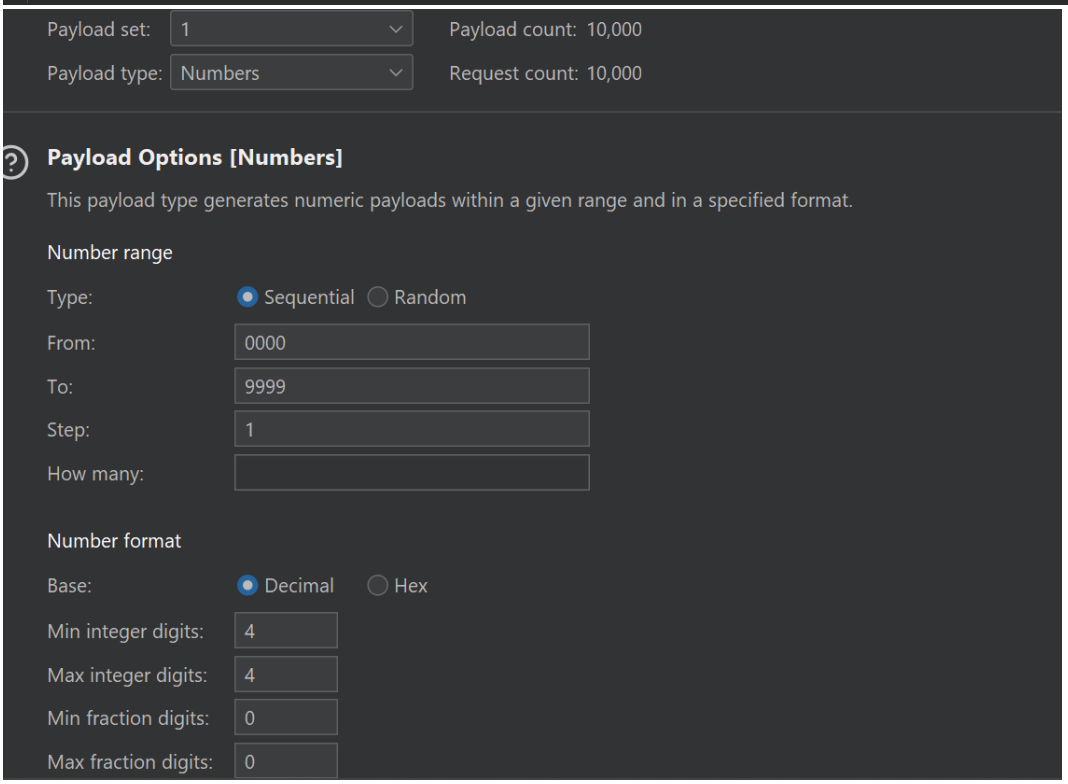
1. **Let's say we have one user victim that we had enumerated through the BOPLA where victim we get here is hp@hack.com and attacker here is pp@hack.com , so let's change the victims password …**



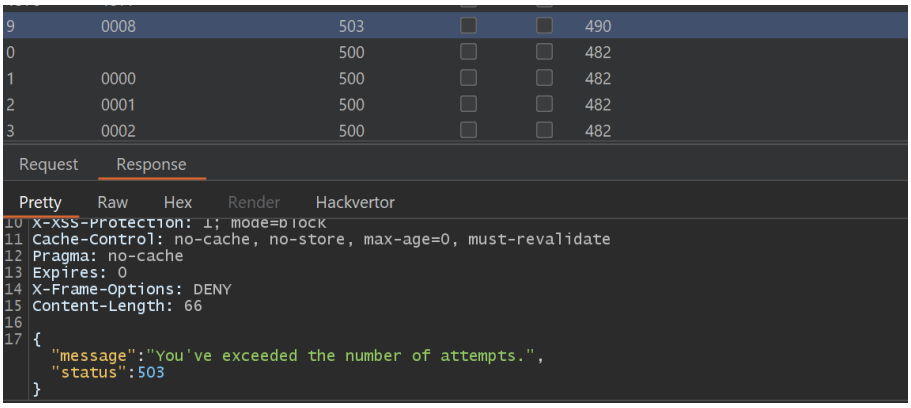


2. Brute Forcing the OTP.. with burp proxy,

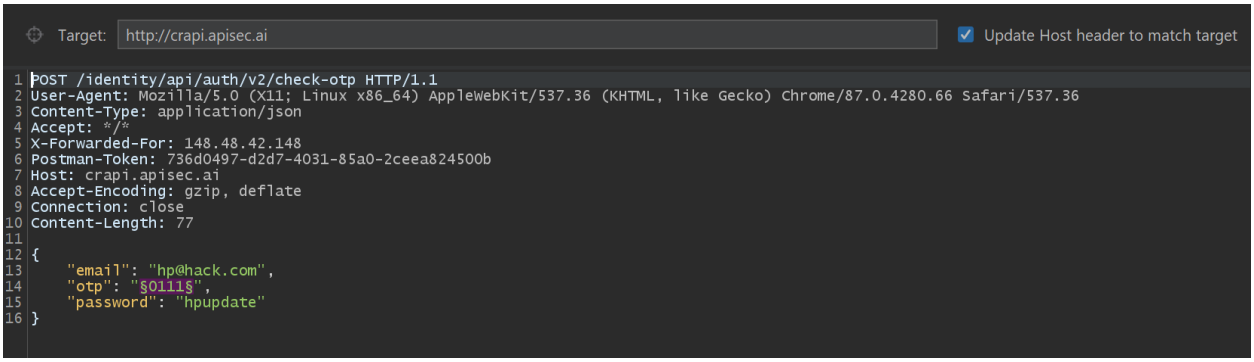




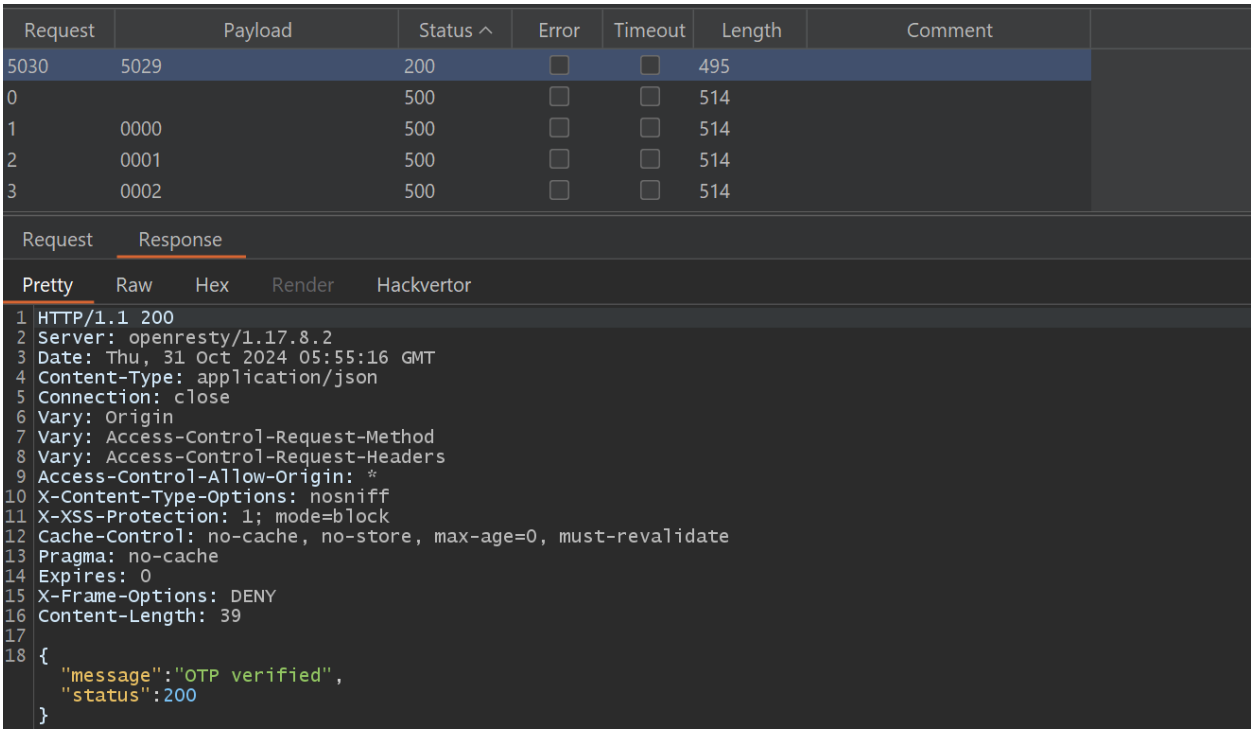
As over here we get rate limit of Exceeding the number of attempts while brute forcing the otp ..



If I look back at the Intruder configuration, in the Payload positions, I can see that its using v3 of the check-otp API endpoint, I had change it to v2..



Boom! We get the OTP…….



Now you can able to change the password by applying the otp you got after brute forcing …

REFERENCES

https://atharvvvsharma.medium.com/owasp-top-10-broken-authentication-a2a70a990dbb

https://auth0.com/blog/what-is-broken-authentication

https://brightsec.com/blog/broken-authentication-impact-examples-and-how-to-fix-it/