I tre codici proposti differiscono a livello di sicurezza e presentano varie vulnerabilità che possono essere sfruttate da potenziali attaccanti. Di seguito, evidenzio le differenze a livello di sicurezza tra i tre codici e le relative vulnerabilità e soluzioni:

Codice 1: Livello di Sicurezza - Basso

Vulnerabilità:

1. **SQL Injection**: Il codice incorpora direttamente i valori dell'input utente nella query SQL senza alcuna sanitizzazione o controllo, rendendo l'applicazione vulnerabile agli attacchi di SQL injection.

2..**Uso di Hashing MD5**: L'hashing delle password utilizzando l'algoritmo MD5 è considerato debole e vulnerabile agli attacchi di hashing rainbow table.

3. Mancanza di controllo sul numero di tentativi di accesso consentiti.

Soluzioni:

1. **Utilizzo di Parametrizzazione delle Query o ORM**: Utilizzare istruzioni SQL parametrizzate o un Object-Relational Mapping (ORM) per evitare gli attacchi di SQL injection.

2. **Utilizzo di Algoritmi di Hash Sicuri:** Sostituire l'hashing MD5 con algoritmi di hash più sicuri come bcrypt o Argon2 per proteggere le password dagli attacchi di hash rainbow table.

3. Implementare un meccanismo di blocco dell'account dopo un certo numero di tentativi falliti.

Codice 2: Livello di Sicurezza - Medio

Vulnerabilità:

1. **Utilizzo di funzioni deprecate:** Utilizza la funzione `mysql_real_escape_string`, che è deprecata e non offre una protezione completa contro gli attacchi di SQL injection.

2. **Uso di Hashing MD5:** Utilizza ancora l'hashing MD5 per proteggere le password degli utenti, che è vulnerabile agli attacchi di hash rainbow table.

Soluzioni:

1. **Utilizzo di Funzioni di Sanitizzazione Moderne:** Utilizzare funzioni di sanitizzazione moderne come `mysqli_real_escape_string` o parametrizzazione delle query per prevenire gli attacchi di SQL injection.

2. **Utilizzo di Algoritmi di Hash Sicuri:** Sostituire l'hashing MD5 con algoritmi di hash più sicuri come bcrypt o Argon2 per proteggere le password dagli attacchi di hash rainbow table.

Codice 3: Livello di Sicurezza - Medio-Alto

Vulnerabilità:

1. **Utilizzo di funzioni deprecate:** Utilizza ancora la funzione `mysql_real_escape_string`, che è deprecata e non offre una protezione completa contro gli attacchi di SQL injection.

2. **Uso di Hashing MD5:** Continua a utilizzare l'hashing MD5 per proteggere le password degli utenti, vulnerabile agli attacchi di hash rainbow table.

3. **Sleep Time per Login Fallito**: Aggiunge un ritardo di 3 secondi in caso di login fallito, che potrebbe essere sfruttato per eseguire attacchi di forza bruta.

Soluzioni:

1. **Utilizzo di Funzioni di Sanitizzazione Moderne:** Utilizzare funzioni di sanitizzazione moderne come `mysqli_real_escape_string` o parametrizzazione delle query per prevenire gli attacchi di SQL injection.

2. **Utilizzo di Algoritmi di Hash Sicuri:** Sostituire l'hashing MD5 con algoritmi di hash più sicuri come bcrypt o Argon2 per proteggere le password dagli attacchi di hash rainbow table.

3. **Implementazione di Controlli Anti-Brute Force:** Implementare meccanismi di rilevamento e mitigazione degli attacchi di forza bruta, come limiti di tentativi di accesso, captcha o ritardi crescenti.

Nota Finale sull'Attacco di Brute Force e le Falle nei Tre Script:

L'attacco di brute force è una tecnica utilizzata dagli attaccanti per ottenere l'accesso non autorizzato ad un sistema o un'applicazione provando ripetutamente diverse combinazioni di username e password fino a trovare quelle corrette. È importante comprendere e mitigare le vulnerabilità che rendono possibile questo tipo di attacco.

In generale, è fondamentale mantenere i sistemi e le applicazioni costantemente aggiornati, utilizzare pratiche di sviluppo sicuro e implementare misure di sicurezza robuste per prevenire gli attacchi di brute force e proteggere i dati sensibili degli utenti.

Oltre a queste note di natura tecnica è necessario e fondamentale che si adotti una formazione ed un processo che educhi i dipendenti alla cura delle loro password in un percorso che vada dalla scelta al cambio periodico di queste ultime. Si dovrebbe inoltre spiegare ai dipendenti l'importanza della scelta di una passphrase lunga ed articolata.

Si sottolinea comunque l'importanza della ricerca di un equilibrio tra sicurezza ed usabilità.