1. **Zabezpieczanie dostępu do komputera:**

- Ustawiłem statyczny adres IP dla Ubuntu na Adapterze 2, modyfikując plik konfiguracyjny netplan (**/etc/netplan/01-netcfg.yaml**) i zastosowując zmiany poleceniem **netplan apply**.

- Następnie zainstalowałem serwer SSH (**apt install openssh-server -y**) i DenyHosts (**apt-get install denyhosts**).

2. **Blokowanie i odblokowywanie dostępu:**

- Blokowanie dostępu do serwera dla wszystkich komputerów wykonano poprzez edycję pliku **/etc/hosts.deny**, dodając **ALL: ALL**.

- Następnie odblokowano dostęp dla wybranego komputera i podsieci, dodając odpowiednie wpisy do pliku **/etc/hosts.allow**.

- Przetestowałem połączenie z serwerem używając usługi Telnet.

3. **Testowanie zabezpieczeń z DenyHosts:**

- Zmieniłem adres IP na komputerze klienta, aby symulować różne próby logowania.

- Wykonałem próby logowania jako nieistniejący użytkownik (baduser) kilkukrotnie, co skutkowało zablokowaniem adresu IP przez DenyHosts.

- Przeczytałem zawartość pliku **/etc/hosts.deny** w celu potwierdzenia blokady.

4. **Konfiguracja DenyHosts:**

- Przeanalizowałem i opisałem różne opcje konfiguracyjne w pliku **/etc/denyhosts.conf**, takie jak **DENY_THRESHOLD_INVALID**, **DENY_THRESHOLD_VALID**, **DENY_THRESHOLD_ROOT**, **HOSTNAME_LOOKUP**, **AGE_RESET_VALID**, **AGE_RESET_ROOT**, **AGE_RESET_INVALID**.

5. **Wnioski:**

- Zabezpieczanie dostępu do komputera wymaga uwzględnienia różnych aspektów, takich jak statyczny adres IP, serwer SSH, DenyHosts i konfiguracja plików hosts.deny i hosts.allow.

- Mechanizm DenyHosts jest skutecznym narzędziem do blokowania dostępu dla potencjalnie niebezpiecznych adresów IP po wielokrotnych nieudanych próbach logowania.

- Odpowiednia konfiguracja pliku DenyHosts (/etc/denyhosts.conf) umożliwia dostosowanie parametrów blokowania i resetowania dla lepszej kontroli dostępu.