pasja-informatyki.pl

Sieci komputerowe

Konfiguracja przełącznika CISCO

Szyfrowanie haseł, AAA, SSH,

Port Monitor, EtherChannel

Damian Stelmach

Spis treści

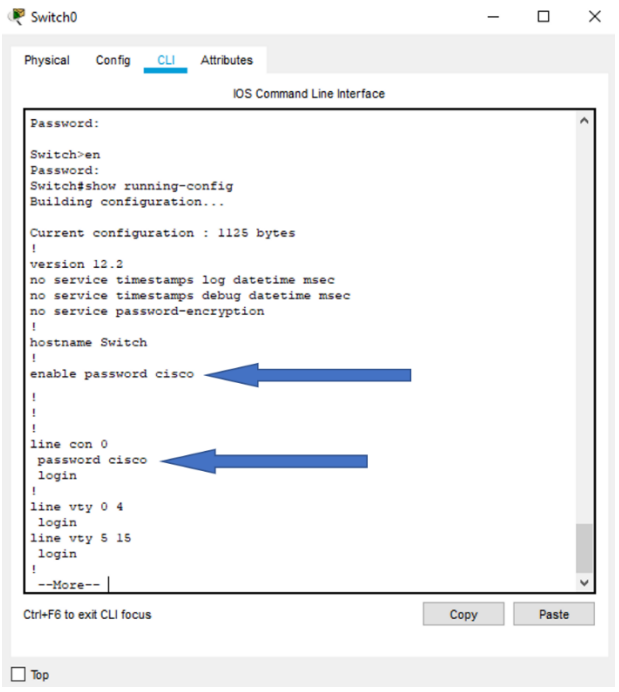
Pierwszą czynnością jaką powinien wykonać administrator przed rozpoczęciem konfiguracji każdego urządzenia sieciowego jest zabezpieczenie dostępu do tejże konfiguracji poprzez hasło. Na przełącznikach CISCO możemy skonfigurować hasło wymagane do zalogowania się do panelu konfiguracyjnego (konsoli CLI), także hasło wymagane w celu przejścia do trybu uprzywilejowanego. Przypomnijmy:

```
Switch> enable
Switch# conf t
Switch(config)# line console 0
Switch(config-line)# password cisco
Switch(config-line)# login
```

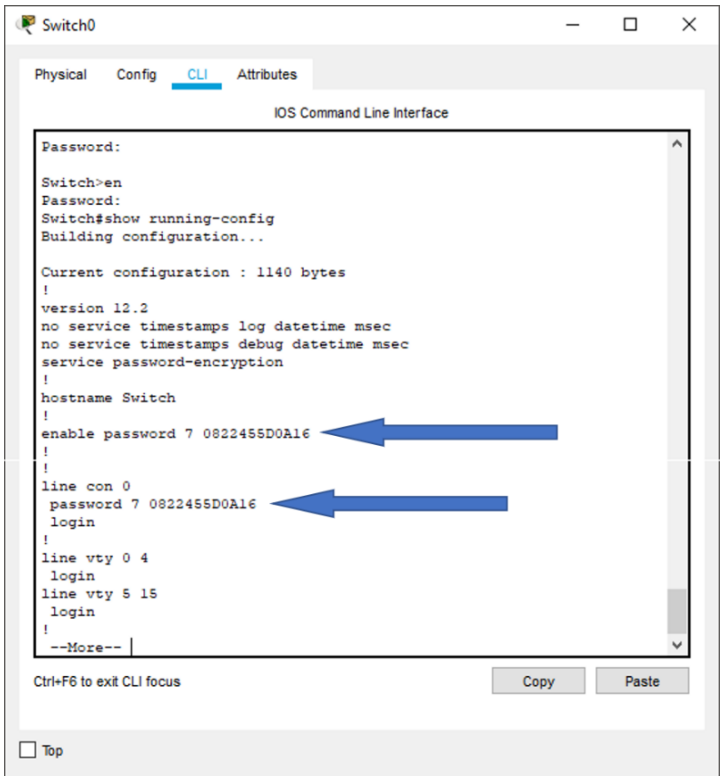
Wykonanie tych poleceń utworzy nam hasło (***cisco***) wymagane do połączenia konsolowego. Z kolei wydanie tych poleceń:

```
Switch> enable
Switch# conf t
Switch(config)# enable password cisco
```

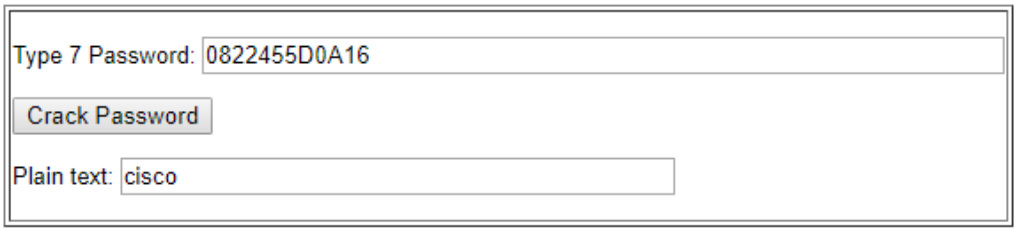
Uruchomi nam hasło wymagane podczas przejścia do tryby uprzywilejowanego. Utworzenie hasał za pomocą takiej metody nie gwarantuje pełnego bezpieczeństwa ponieważ hasła w pamięci urządzenia przechowywane są jawnym tekstem. Można je odczytać uruchamiając w trybie uprzywilejowanym, poleceniem ***show running-config*** plik konfiguracyjny, część jego zawartości, razem z hasłami widoczna jest poniżej:



Istnieje kilka metod, które pozwalają nam zabezpieczyć hasła zapisane w pliku konfiguracyjnym. Pierwsza z nich to zaszyfrowanie haseł metodą Vigenere (poziom 7 szyfrowania). Aby takie szyfrowanie utworzonych już haseł zrealizować, wystarczy w trybie konfiguracji wykonać polecenie ***service password-encryption***. Od tego momentu, wszystkie zapisane hasła na urządzeniu będą szyfrowane właśnie metodą Vigenere:



Niestety szyfrowanie czegokolwiek na poziomie 7 nie daje żadnych gwarancji bezpieczeństwa, ponieważ jest to bardzo łatwa do odwrócenie metoda. Z łatwością można znaleźć w sieci wiele stron, które po podstawie skopiowanego ciągu znaków potrafią odwrócić to szyfrowanie:

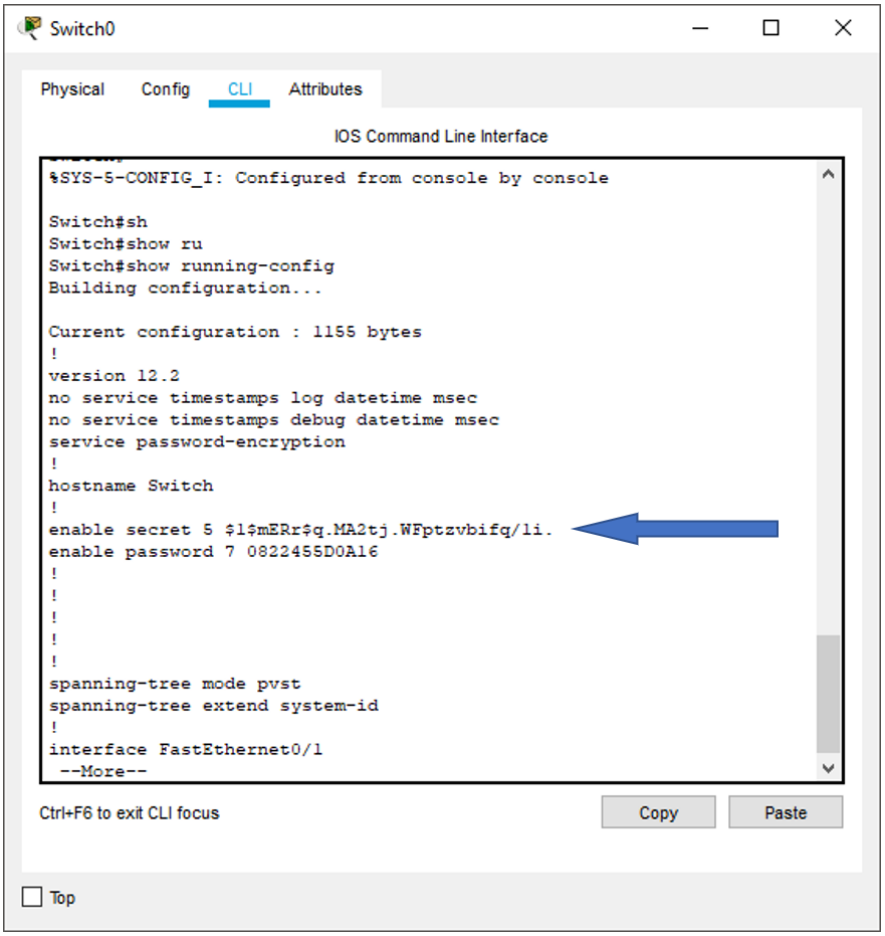


Jak widać i to rozwiązanie nie daje nam żadnych gwarancji bezpieczeństwa, dlatego warto przyjrzeć się innej metodzie. Dużo bezpieczniejsza i bardziej skuteczna będzie metoda szyfrowania wykorzystująca algorytm **MD5crypt** (nie mylić ze „zwykłym" MD5, który jest dużo słabszym algorytmem haszowania). Jest to metoda, z której nie da się odczytać hasła z przechwyconego ciągu znakowego, dlatego jest

dużo bardziej skuteczna. Aby wykorzystać algorytm MD5crypt do zaszyfrowania hasła przejścia do trybu uprzywilejowanego należy wykonać następujące polecenia:

```
Switch> enable
Switch# conf t
Switch(config)# enable secret cisco
```

Kluczowym jest tutaj polecenie ***secret***, które odpowiada właśnie za szyfrowanie algorytmem MD5crypt. Od momentu wykonania tego polecenia hasło przejścia do trybu uprzywilejowanego zapisane będzie na urządzeniu w postaci zaszyfrowanego ciągu znaków:

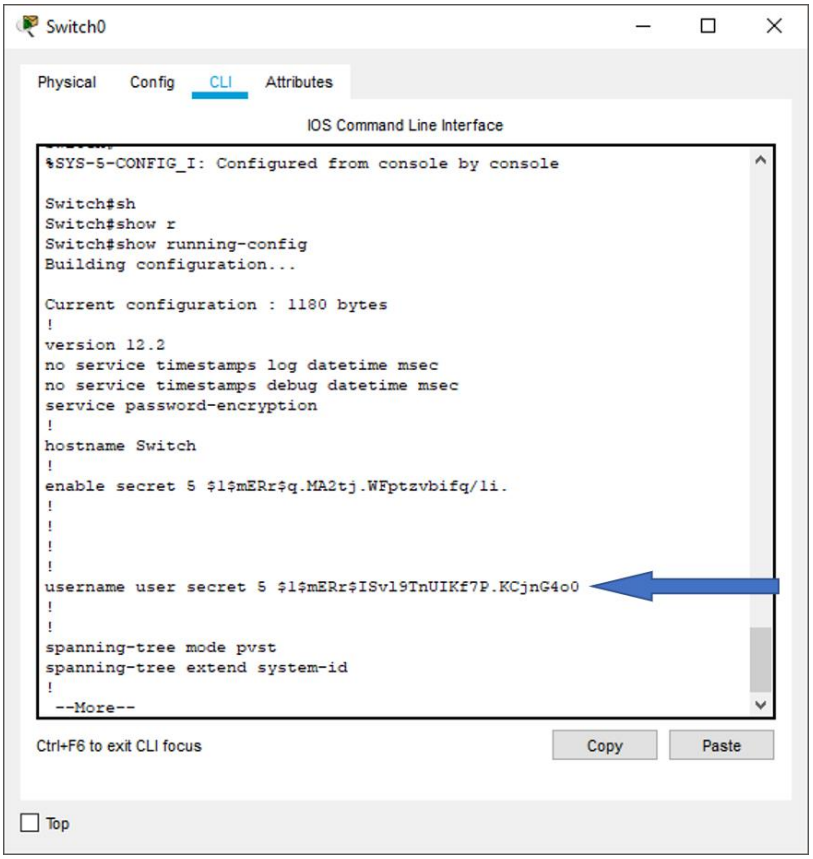


Dla zwiększenia bezpieczeństwa powinniśmy jeszcze usunąć hasło zaszyfrowane na poziomie 7. Do realizacji tej czynności należy wykonać w trybie konfiguracji polecenie ***no enable password***.

Idąc dalej, powinniśmy teraz zaszyfrować algorytmem MD5crypt hasło wymagane przy połączeniu konsolowym. Niestety nie jest to możliwe w sytuacji, kiedy do uwierzytelnienia na przełączniku używamy tylko hasła. W pamięci urządzenia hasło wymagane przy połączeniu konsolowym czy też przy połączeniach za pośrednictwem Telnetu czy SSH może być zapisane albo jawnym tekstem, albo w postaci zaszyfrowanego ciągu łatwego do odszyfrowania. Jeśli chcielibyśmy wykorzystać silniejszą metodę szyfrowania należy utworzyć na urządzeniu użytkownika wraz z hasłem. Wówczas możliwa jest do wykorzystania metoda, algorytm MD 5. Użytkownika na urządzeniu tworzymy wydając następujące polecenia:

```
Switch> enable
Switch# conf t
Switch(config)# username user secret hasło
```

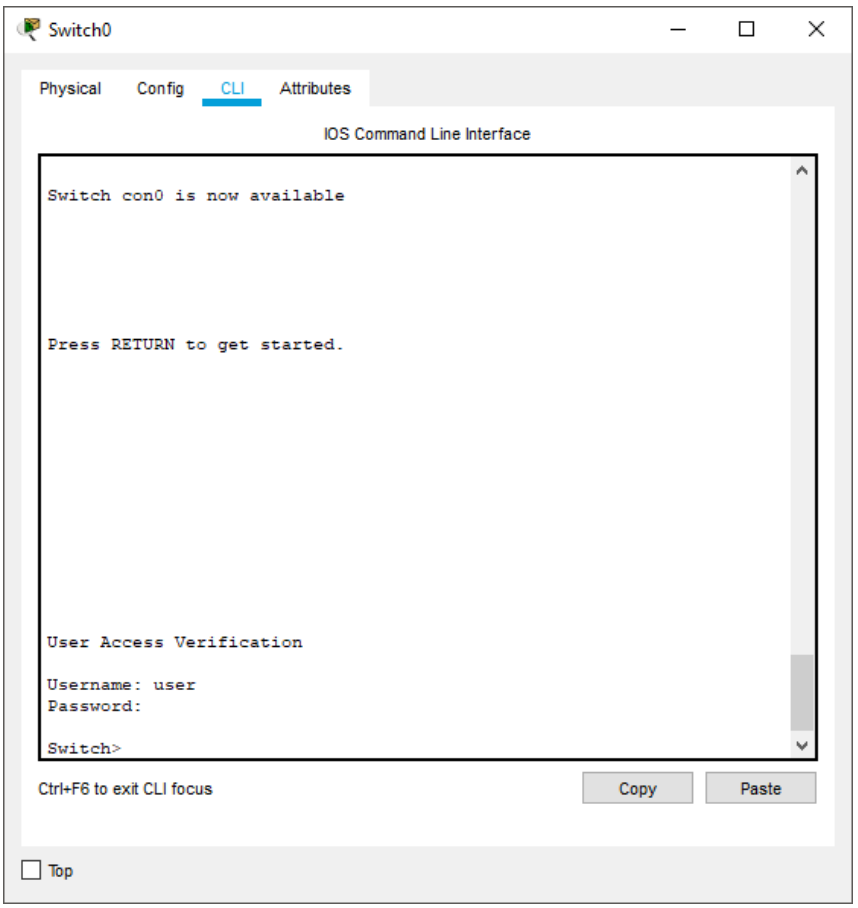
Po ich wykonaniu utworzony zostanie użytkownik o podanej nazwie i haśle, które zaszyfrowane będzie algorytmem MD5crypt:



Teraz należy jeszcze wykonać polecenia, które pozwolą stosować nazwę użytkownika i hasło podczas logowania do przełącznika za pośrednictwem połączenia konsolowego. W trybie konfiguracji wydajemy następujące polecenia:

```
Switch(config)# line console 0
Switch(config-line)# login local
```

Pierwsze polecenie uruchamia nam konfigurację połączenia konsolowego, drugie z kolei uruchamia logowanie za pomocą lokalnej bazy użytkowników. Od teraz, chcąc połączyć się z przełącznikiem za pomocą połączenia konsolowego, należy podać nazwę użytkownika oraz hasło:



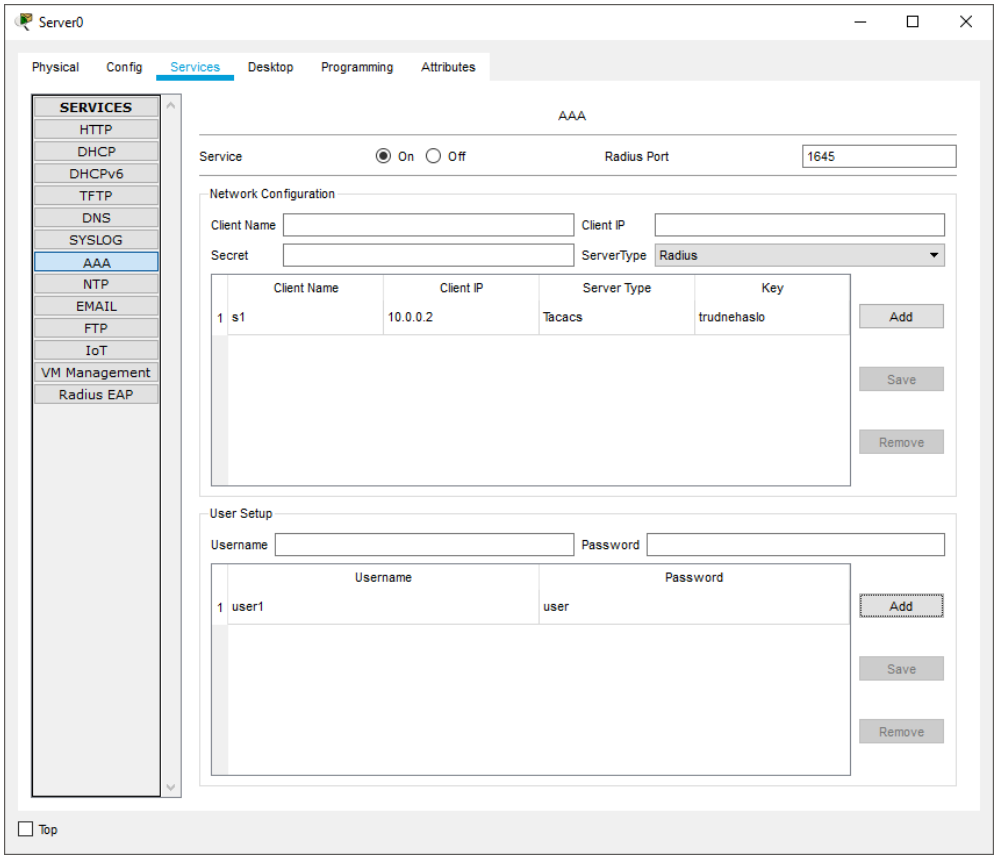
Podsumowując: aby utworzyć użytkownika z hasłem oraz uruchomić uwierzytelnianie wymagane przy połączeniu konsolowym należy wydać następujące polecenia:

```
Switch> enable
Switch# conf t
Switch(config)# username user secret haslo
Switch(config)# line console 0
Switch(config-line)# login local
```

Omówione poprzednio metody weryfikacji czyli metoda oparta na haśle oraz metoda oparta na bazie użytkowników nie zamykają jeszcze całkowicie kwestii związanych z dostępem do konfiguracji urządzeń sieciowych. Jest jeszcze jedna metoda, najbardziej zaawansowana, zwana metodą potrójnego A. Nazwa tej metody wywodzi się od 3 słów – Authentication, Authorization oraz Accounting. Autentykacja to uwierzytelnienie czyli sprawdzenie czy dany użytkownik jest faktycznie tym za kogo się podaje, autoryzacja to nic innego jak weryfikowanie do jakich zasobów konkretny użytkownik ma dostęp, accouting natomiast, zwany po polsku raportowaniem to nic innego jak zbieranie informacji – logów o czynnościach jakie wykonał użytkownik. Mechanizm potrójnego A może wykorzystywać lokalną bazę użytkowników, którą utworzoną mamy na urządzeniu, ale także bazy użytkowników zapisane na serwerach logowania, które stosują protokoły uwierzytelniania takie jak RADIUS lub też TACACS. Oba te protokoły służą do weryfikacji danych o użytkownikach, i można je wykorzystać również do uwierzytelniania na urządzeniach sieciowych. Serwery logowania to nic innego jak konkretne usługi sieciowe, korzystające z tych protokołów, które konfigurować możemy na dedykowanym oprogramowaniu np. CISCO Secure, ale również na Linuxach czy Windowsach. Na serwerze Windowsowym da się np. skonfigurować usługę serwera logowania i połączyć ją z Active Directory, dzięki czemu możemy logować się do urządzenia korzystając z kont domenowych. Protokół RADIUS jest to rozwiązanie otwarte wspierane przez większość firma zajmujących się technologiami sieciowymi, działające na protokole transportowym UDP. TACACS natomiast to rozwiązanie firmy CISCO, pracujące z wykorzystaniem protokołu transportowego TCP. W przypadku tego pierwszego protokołu nie mamy pełnego szyfrowania, to znaczy tylko część komunikacji pomiędzy klientem, a serwerem RADIUS realizująca uwierzytelnianie jest szyfrowana, pozostałe polecenia przesyłane są jawnym tekstem. TACACS natomiast szyfruje cała komunikację, dlatego można uznać, że jest zdecydowania bezpieczniejszy. Poniższa tabela przedstawia porównanie tych dwóch protokołów uwierzytelniania:

RADIUS	TACACS
Stosuje protokół transportowy UDP	Stosuje protokół transportowy TCP
Działa na portach 1812 lub 1813	Działa na porcie 49
Nie szyfruje całej komunikacji	Szyfruje cała komunikację
Jest to standard otwarty	Jest to standard będący własnością CISCO

Aby możliwa była realizacja logowania poprzez metodę AAA wymagany jest w sieci serwer, który odpowiadał będzie za zdalne uwierzytelnianie użytkowników. Zakładając, ze taki serwer w sieci już pracuje, do tego aby poprawnie skonfigurować metodę AAA potrzebować będziemy adres IP tego serwera, hasło (klucz) dostępu, a także login i hasło użytkownika. W programie Packet Tracer dostępna jest symulacja serwera wraz z usługą AAA. Po podłączeniu serwera do przełącznika, powinniśmy dokonać jego konfiguracji, może ona wyglądać tak:



W pierwszej tabeli widoczne są dane przełącznika sieciowego (jego nazwa, adres IP, rodzaj protokołu uwierzytelnia oraz hasło). Druga tabela to zbiór użytkowników, którzy będą mogli logować się do urządzenia. Adres IP tego serwera to 10.0.0.1/24. Teraz czas na konfigurację przełącznika. W przypadku programu Packet Tracer konfiguracja metody AAA możliwa jest tylko na przełączniku z serii 3500, dlatego też na tym sprzęcie ją zrealizujemy. Zaczniemy od tego, że zmienimy nazwę naszego przełącznika (Switch zamienimy na s1) i utworzymy sobie lokalnego użytkownika, który będzie mógł zalogować się do urządzenia jeśli z jakiś powodów konfiguracja AAA będzie nieskuteczna, albo utracimy możliwość komunikacji z serwerem:

```
Switch> enable
Switch# conf t
```

```
Switch(config)# hostname s1
s1(config)# username user secret hasło
```

Teraz musimy ustawić adres IP dla przełącznika. Przypominam, że na przełączniku adres IP nadajemy dla całej sieci VLAN, a że domyślnie wszystkie porty przełącznika pracują w sieci VLAN o identyfikatorze jeden, dla takiego interfejsu nadajemy adres IP:

```
s1(config)# interface vlan 1
s1(config-if)# ip address 10.0.0.2 255.255.255.0
s1(config-if)# no shutdown
s1(config-if)# exit
```

Na koniec uruchamiamy funkcjonalność AAA dla przełącznika wydając następujące polecenia:

```
s1(config)# aaa new-model
s1(config)# tacacs-server host 10.0.0.1 key trudnehaslo
s1(config)# aaa authentication login default group tacacs+
local
```

Pierwsze polecenie uruchamia nam uwierzytelniania za pomocą potrójnego A. W drugiej linii podajemy parametry serwera, a na koniec uruchamiamy możliwość uwierzytelniania serwerowego. Ostatnie parametr (***Local***) uruchamia możliwość lokalnego logowania, w razie braku komunikacji z serwerem uwierzytelniania. Podsumowując:

```
Switch> enable
Switch# conf t
Switch(config)# hostname s1
s1(config)# username user secret haslo
s1(config)# interface vlan 1
s1(config-if)# ip address 10.0.0.2 255.255.255.0
s1(config-if)# no shutdown
s1(config-if)# exit
s1(config)# aaa new-model
s1(config)# tacacs-server host 10.0.0.1 key trudnehaslo
s1(config)# aaa authentication login default group tacacs+
local
```

Protokół SSH jest to protokół zdalnego dostępu, który umożliwia zdalne łączenie się z hostami (serwerami, przełącznikami, ruterami) stosując szyfrowaną komunikację. Jest to bardzo popularna metoda komunikacji ze, dlatego warto ją również zastosować w przypadku konfiguracji dostępu do zarzadzania urządzeń sieciowych. Konfigurację zdalnego dostępu z wykorzystaniem protokołu SSH można skonfigurować na urządzeniu wydając następujące polecenia:

```
Switch> enable
Switch# conf t
Switch(config)# hostname s1
s1(config)# enable secret haslo
s1(config)# ip domain-name pasja.com
s1(config)# username user secret haslo
s1(config)# interface vlan 1
s1(config-if)# ip address 10.0.0.2 255.255.255.0
s1(config-if)# no shutdown
s1(config-if)# exit
```

Za pomocą powyższych poleceń zmieniliśmy nazwę urządzenia, ustawiliśmy hasło do trybu uprzywilejowanego, a także ustawiliśmy nazwę domeny. Te elementy konfiguracji są niezbędne do tego aby uruchomić SSH, ale pamiętajcie, że nazwa urządzenia oraz domena mogą być inne. Następnie utworzyliśmy sobie użytkownika oraz nadaliśmy adres IP dla przełącznika. Idziemy dalej wprowadzając polecenie:

```
s1(config)# crypto key generate rsa
```

które wymusi uruchomienie serwera SSH na przełączniku i pozwoli na generowanie kluczy szyfrujących. Po wykonaniu tego polecenia pojawi się komunikat:

"The name for the keys will be: s1.pasja.com
Choose the size of the key modulus in the range of 360 to 2048 for your
General Purpose Keys. Choosing a key modulus greater than 512 may take
a few minutes.

How many bits in the modulus [512]:"

Informujący nas o nadaniu nazwy dla kluczy. Teraz musimy podać wielkość takiego klucza, standardowy rozmiar to 1024 bity, taki też można podać. W dalszej kolejności wybieramy wersję protokołu SSH wydając polecenie:

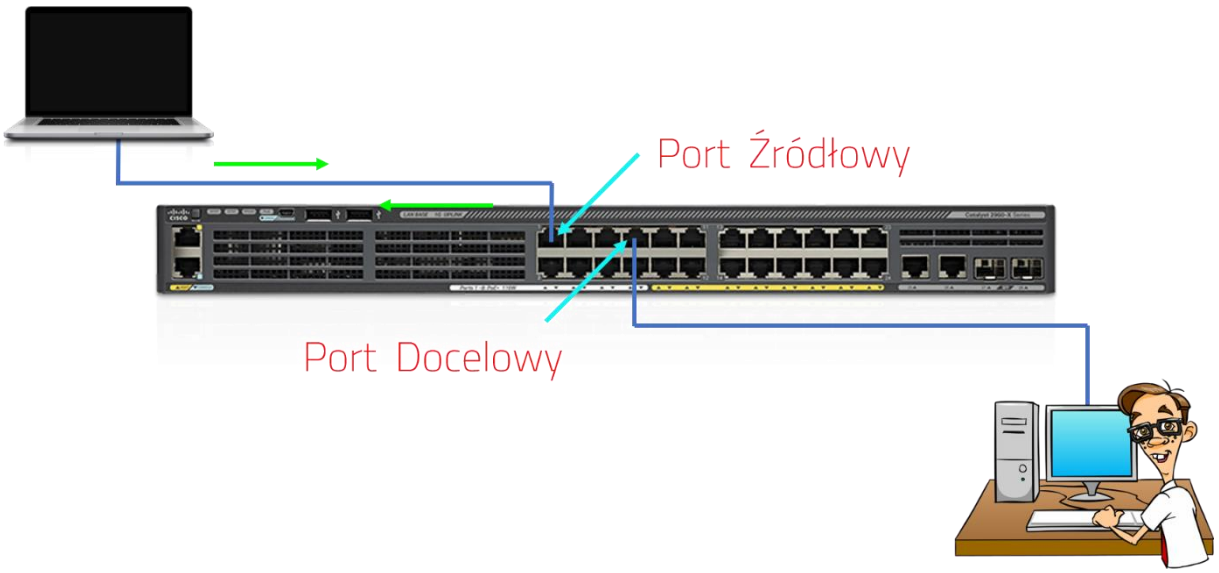
```
s1(config)# ip ssh version 2
```

Teraz możemy przystąpić już do uruchomienia możliwości logowania poprzez protokół SSH na wybranych liniach wirtualnych wydając następujące polecenia:

```
s1(config)# line vty 0 15
s1(config-line)# transport input ssh
s1(config-line)# login local
```

Jeśli konfigurujemy faktyczny sprzęt w naszej sieci i chcemy podłączyć się do niego poprzez protokół SSH, no to oczywiście możemy użyć do tego programu PUTTY. Jeśli chcemy natomiast pobawić się w konfigurację poprzez SSH w programie Packet Tracer, musimy uruchomić wiersz polecenia (command prompt) i wprowadzić polecenie: **`ssh -l user 10.0.0.2`**. Gdzie user to nazwa użytkownika, a 10.0.0.2 to adres IP urządzenia.

Port mirroring jest funkcją, która pozwala na kopiowania danych z danego, konkretnego portu lub grupy portów na inny. Chodzi o to, że dane, które transmitowane są np. z i do portu 1 mogą być kopiowane na np. port 7. Funkcja te może być wykorzystywane w celach diagnostycznych, np. jeśli chcemy analizować ruch sieciowych maszyn w naszej sieci. Wówczas wszystkie dane z portu źródłowego trafiają do naszego komputera dzięki temu możemy je analizować (wykorzystując np. program Wireshark) i szukać przyczyn błędnego działania usług sieciowych.



Medal ma oczywiście dwie strony, no i port mirroring może również zostać wykorzystany w celach, nazwijmy to mniej przyjaznych. Hacker, który skonfiguruje kopiowanie danych na port, do którego podłączony jest jego komputera będzie miał dostęp do informacji, z jakich korzysta usług sieciowych i na jakie strony WWW wchodzi użytkownik. To w jakim celu zostanie taka funkcja użyta no to już zależy od osób które z niej korzystają.

Zakładając, że chcemy aby do naszego komputera (podłączonego do portu 10 przełącznika), na którym zainstalowane mamy oprogramowanie do analizy ruchu sieciowego trafiały dane z komputerów podłączonych do portów 1 i 2, polecenia konfiguracyjne wyglądają następująco:

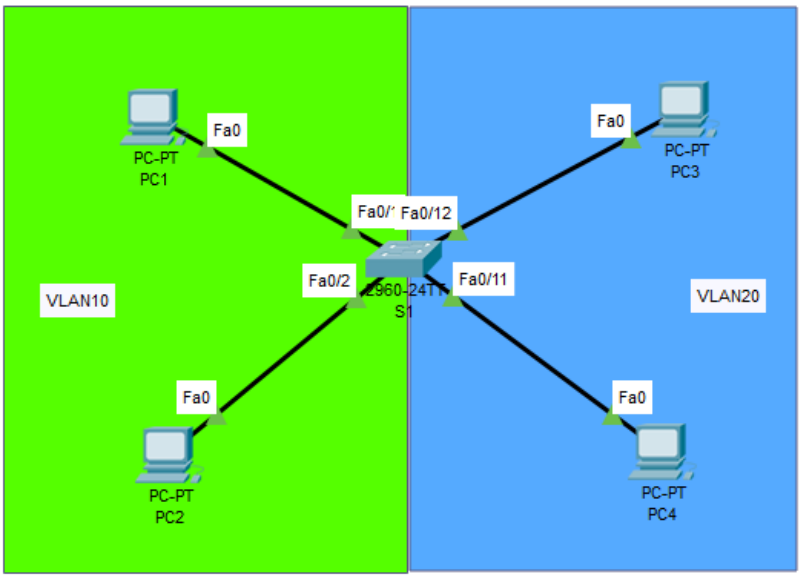
```
s1> enable
s1# conf t
s1(config)# monitor session 1 source interface fastEthernet 0/1
s1(config)# monitor session 1 source interface fastEthernet 0/2
s1(config)# monitor session 1 destination interface
fastEthernet 0/10
```

Od teraz wszystkie dane, zarówno wychodzące, jak i przychodzące z portów 1 i 2, kopiowane będą na port 10.

Jeśli wolelibyśmy aby kopiowane do naszego komputera były dane tylko wychodzące, albo tylko przychodzące z portów 1 i 2 możemy użyć do tego parametrów **rx** oraz **tx**. Zakładając, że chcemy aby na nasz komputer kopiowane były dane tylko wychodzące z portu pierwszego i tylko przychodzące na port 2 wydamy następujące polecenia:

```
s1> enable
s1# conf t
s1(config)# monitor session 1 source interface fastEthernet 0/1
tx
s1(config)# monitor session 1 source interface fastEthernet 0/2
rx
s1(config)# monitor session 1 destination interface
fastEthernet 0/10
```

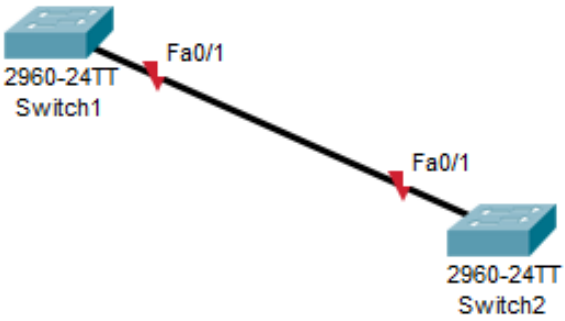
Zakładając, że komputery, z których dane chcemy otrzymywać pracują w jednej sieci VLAN, ale podłączone są do różnych przełączników, tak jak na grafice:



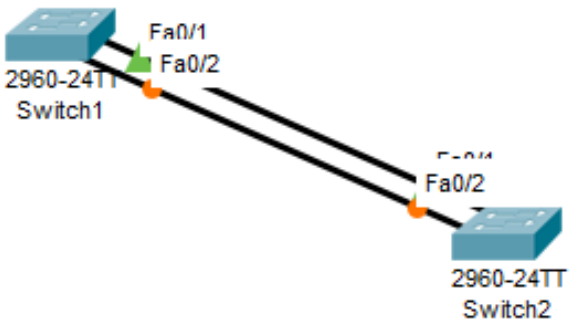
Zamiast konkretnego interfejsu możemy podać identyfikator sieci VLAN. Wówczas polecenia realizujące taką funkcjonalność będą wyglądać następująco:

```
s1> enable
s1# conf t
s1(config)# monitor session 1 source vlan 10
s1(config)# monitor session 1 destination interface
fastEthernet 0/1
```

Wyobraźmy sobie sytuację, w której musimy połączyć ze sobą dwa przełączniki. Żaden problem, wystarczy użyć odpowiedniego przewodu i będzie ok:



Co się jednak stanie kiedy łącze to ulegnie awarii? No lipa będzie bo urządzenia podłączone zarówno do jednego jak i drugiego przełącznika stracą możliwość wzajemnej komunikacji. Jak zatem zapobiec takiej sytuacji? Istnieje bardzo proste rozwiązanie a mianowicie użycie nadmiarowej ilości połączeń pomiędzy urządzeniami. Dzięki czemu awaria jednego przewodu nie spowoduje braku dostępu do sieci dla urządzeń, a dodatkowo zwiększy nam się ogólna przepustowość łącza pomiędzy urządzeniami:



Niestety takie rozwiązanie powoduje, że jedno z łącz, przez cały czas pozostaje nieaktywne. Dopiero po awarii jednego z nich, uruchomione zostaje drugie. Dzieje się tak dlatego, że na przełącznikach sieciowych działa protokół STP (ang. Spanning Tree Protocol), którego zadaniem jest zapobieganie pętli ramek, jakie mogę powstać w przypadku użycia nadmiarowej ilości połączeń pomiędzy przełącznikami. Aby można było wykorzystać większą ilość połączeń, możemy albo wyłączyć protokół STP, co jest bardzo ryzykowane, bo może doprowadzić do paraliżu całe sieci, albo możemy wykorzystać

funkcjonalność LAG (*ang. Link Aggregation Port*), zwaną w technologii CISCO kanałem ethernetowym (Etherchannel) i połączyć klika fizycznych przewodów w jeden logiczny. Dzięki czemu załatwimy sobie temat niebezpieczeństwa awarii związanego z wykorzystaniem tylko jednego przewodu łączącego urządzenia, a przy okazji nie będziemy zmuszeni wyłączyć STP na przełącznikach.

Konfiguracja takiej funkcjonalności opisanej w standardzie **802.3ad** jest specjalnie skomplikowana, dlatego zanim ja zrealizujemy no to kilka słów na temat metod takiego łączenia portów. Do dyspozycji mamy dwie drogi, albo statyczną konfigurację bez użycia protokołów sieciowych, lub też dynamiczną, z wykorzystaniem jednego z dwóch protokołów sieciowych, które taką agregację, takie łączenie umożliwiają. Metoda statyczna ma tę przewagę nad dynamiczną, że nie wykorzystuje do tego żadnych protokołów sieciowych, dzięki temu w naszej sieci nie wysyłane są dodatkowe dane generowane przez te protokoły. W przypadku agregacji dynamicznej, stosuje się protokoły PAgP (*ang. Port Aggregation Protocol*) oraz LACP (*ang. Link Aggregation Control Protocol*). Zasadnicza różnica między nimi jest taka, ze ten pierwszy jest własnością CISCO i wspierany jest tylko na urządzeniach tej firmy, chociaż najnowsze urządzenia mogą go już nie obsługiwać, drugi natomiast to standard otwarty, z którego korzystają wszystkie urządzenia, w tym również urządzenia CISCO. Zasadniczo działanie tych protokołów jest zbliżone, dlatego decyzja który wybrać zależy tak naprawdę od tego jakie urządzenia macie w swojej sieci. Jeśli macie sprzęt tylko CISCO to śmiało możecie zastosować protokół PAgP, jeśli w sieci macie mieszankę, albo tylko sprzęt innych marek no to wyboru nie ma i należy użyć protokołu LACP.

Tworząc grupę portów należy mieć na uwadze kilka kwestii:

- porty muszą pracować z tą samą szybkością (nie łączymy ze sobą portów np. 100 Mb/s i 1Gb/s), i w trybie pełnego duplexu czyli muszą potrafić jednocześnie przesyłać i odbierać dane;
- maksymalnie możemy utworzyć na przełączniku do 6 grup portów, przy czym maksymalna liczba portów w grupie wynosi 8;
- biorąc pod uwagę algorytm rozdzielający obciążenia na poszczególne porty lepiej tworzyć grupy na które składa się 2, 4, albo 8 portów.

W tabeli poniżej widać jak procentowo rozkładana jest przepustowość w zależności od liczby portów jakie składają się na cała grupę:

Numer portu \ Ilość portów w kanale	1	2	3	4	5	6	7	8
1	100%	50%	37,5%	25%	25%	25%	25%	12,5%
2	-	50%	37,5%	25%	25%	25%	12,5%	12,5%
3	-	-	25%	25%	25%	12,5%	12,5%	12,5%
4	-	-	-	25%	12,5%	12,5%	12,5%	12,5%
5	-	-	-	-	12,5%	12,5%	12,5%	12,5%
6	-	-	-	-	-	12,5%	12,5%	12,5%
7	-	-	-	-	-	-	12,5%	12,5%
8	-	-	-	-	-	-	-	12,5%

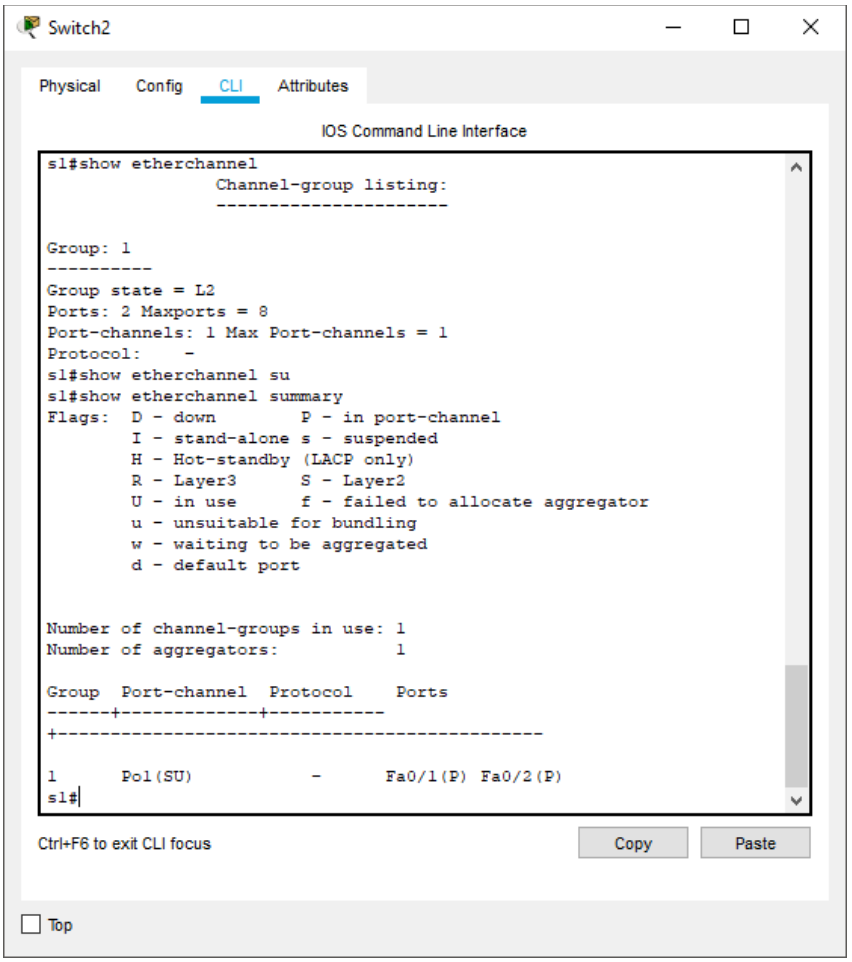
Podział ten, który na pierwszy rzut oka może wydawać się nielogiczny, wynika z działania algorytmu, który każdemu z portów przypisuje określoną wartość liczbową, przy czym suma liczb w całym kanale musi wynosić 8. Jeśli przykładowo mamy 3 porty w grupie to pierwszy i drugi port dostaną trójkę, a że do sumy 8 brakuje liczby dwa no to tez trzeci otrzyma właśnie dwójkę dlatego też w takim przypadku obciążenie będzie wynosiło po 37,5% na pierwszych dwóch portach, i 25% na trzecim. Inny przykład, jeśli w kanale jest pięć portów no to pierwsze 3 oznaczone zostaną dwójkami, a pozostałe dwa jedynkami, bo nie może wystąpić taka sytuacja, iż jeden z portów nie otrzyma żadnej liczby. Wówczas obciążenie dla poszczególnych łączy będzie takie, że po 25% przypadnie na 3 pierwsze porty, a po 12,5% na porty 4 i 5.

Przechodzimy do statycznej konfiguracja LAG, gdzie utworzymy sobie pierwszą grupę, na którą składać się będą porty 1 i 2:

```
s1> enable
s1# conf t
s1(config)# interface range fastEthernet 0/1-2
s1(config-if-range)# channel-group 1 mode on
```

Oczywiście taką konfigurację należy przeprowadzić również na drugim przełączniku.

Wydając polecenia w trybie uprzywilejowanym ***show etherchannel*** oraz ***show etherchannel summary*** możemy wyświetlić sobie podsumowanie wykonanych operacji:



Teraz konfiguracja dynamiczna z wykorzystaniem protokołu **PAgP**. Tym razem utworzymy sobie drugą grupę, na którą składać się będą porty 3 i 4:

```
s1> enable
s1# conf t
s1(config)# interface range fastEthernet 0/3-4
s1(config-if-range)# channel-group 2 mode desirable
```

Na koniec konfiguracja dynamiczna z wykorzystaniem protokołu **LACP**. Tworzymy 3 grupę, na którą składać się będą porty 5 i 6:

```
s1> enable
s1# conf t
s1(config)# interface range fastEthernet 0/5-6
s1(config-if-range)# channel-group 3 mode active
```

Jak widzicie konfiguracja nie jest specjalnie skomplikowana, ogranicza się do dwóch poleceń. Patrząc na to realnie i praktycznie, jeśli macie dostęp do konfiguracji obu urządzeń to tak naprawdę nie ma większego znaczenia, którą metodę wybierzecie. Równie dobrze sprawdzi się konfiguracja statyczna, jak i dynamiczna. Pamiętajcie, że po zagregowaniu łączy, awaria jednego przewodu wchodzącego w skład całej grupy nie powoduje awarii całego kanału ethernetowego.