

Warszawa, 2023-04-23

Prowadzący: inż. Bodzan Marcin

Automatyzacja zadań w chmurze 2 (IZ08TC1)

Sprawozdanie z zajęć z dnia 2023-04-22

Dokumentacja skryptu tworzącego użytkowników w Azure AD na podstawie plików *.csv

Autor:	Mieszko Urbanowski
Numer albumu:	3421
Grupa:	IZ08TC1

Spis treści

Dokumentacja skryptu.....	3
Wprowadzenie.....	3
Opis skryptu.....	3
Struktura katalogów:.....	3
Format danych wejściowych.....	4
Konfiguracja skryptu.....	5
Logowanie do AD.....	5
Praca automatyczna.....	6
Dodatkowa konfiguracja.....	7
Użycie skryptu.....	8
Obsługa błędów.....	10
Pusty plik wejściowy user0.csv.....	10
Puste wartości w niektórych polach user01.csv.....	11
Błędny typ grupy user1.csv.....	12
Brak grupy w AD user2.csv.....	14
Użytkownik już istnieje user3.csv.....	15

Wykaz rysunków

Plik wejściowy csv.....	4
Identyfikator dzierżawy PowerShell.....	5
Identyfikator dzierżawy – Portal Azure.....	6
Logowanie do AD.....	8
Wynik działania skryptu.....	9
Wynik działania - pusty plik wejściowy.....	10
Dane wejściowe - brak niektórych pól.....	11
Wynik działania - brak niektórych pól.....	12
Dane wejściowe - błędny typ grupy.....	12
Wynik działania - błędny typ grupy.....	13
Dane wejściowe - brak grupy w AD.....	14
Wynik działania - brak grupy w AD.....	14
Dane wejściowe - użytkownik już istnieje.....	15
Wynik działania - użytkownik już istnieje.....	16

Format danych wejściowych

Dla każdego użytkownika tworzymy plik z rozszerzeniem **csv** w katalogu **.\in**. Struktura pliku:

```
1 imie,nazwisko,stanowisko,email,lokalizacja,dzial,numer
2 Marek,Platfoos,Dyrektor,marek@witkuo.onmicrosoft.com,Wołomin,Serwis,2343
```

Ilustracja 1: Plik wejściowy csv

Opis pól z danymi:

- **imie** – imię użytkownika
- **nazwisko** – nazwisko użytkownika
- **stanowisko** – stanowisko użytkownika
- **email** – e-mail użytkownika (w domenę obsługiwanej przez AD)
- **lokalizacja** – miasto użytkownika
- **dzial** – dział użytkownika oraz grupa AD do której zostanie przypisany użytkownik (grupa musi istnieć, nie może to być grupa dystrybucyjna)
- **numer** – identyfikator użytkownika w systemie HR zapisywany jest w AD jak nie-standardowy atrybut użytkownika (*extension_bf88dea2157f4090ad834774ae1602e6_externalid*)

Hasło użytkownika generowane jest automatycznie przez funkcję **New-Password**. Jest ono generowane na podstawie fragmentu **New-Guid**.

Imię i nazwisko mogą zawierać znaki narodowe. Są one usuwane dla pola -MailNickName. Znaki inne niż polskie mogą wymagać modyfikacji funkcji **Remove-StringLatinCharacters** (funkcja pochodzi ze strony <https://lazywinadmin.com/2015/05/powershell-remove-diacritics-accents.html> z drobnymi poprawkami)

Konfiguracja skryptu

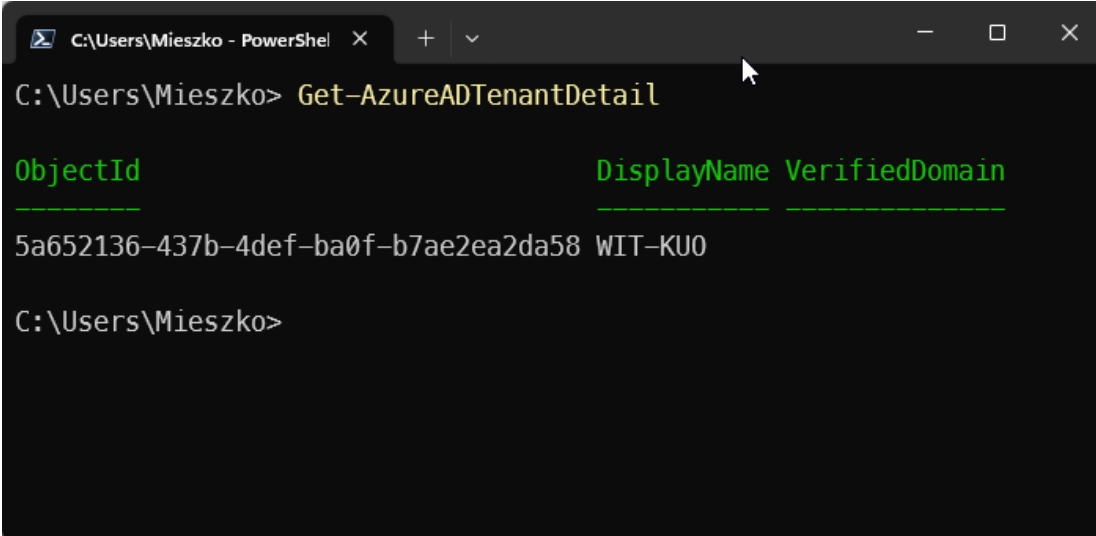
Logowanie do AD

Przed użycie w skrypcie należy ustawienie ID Tenant-a (Identyfikator dzierżawy):

```
$tenantid = "...."
```

Aktualną wartość możemy odczytać w PowerShell:

```
Import-Module AzureAD  
Connect-AzureAD  
Get-AzureADTenantDetail
```

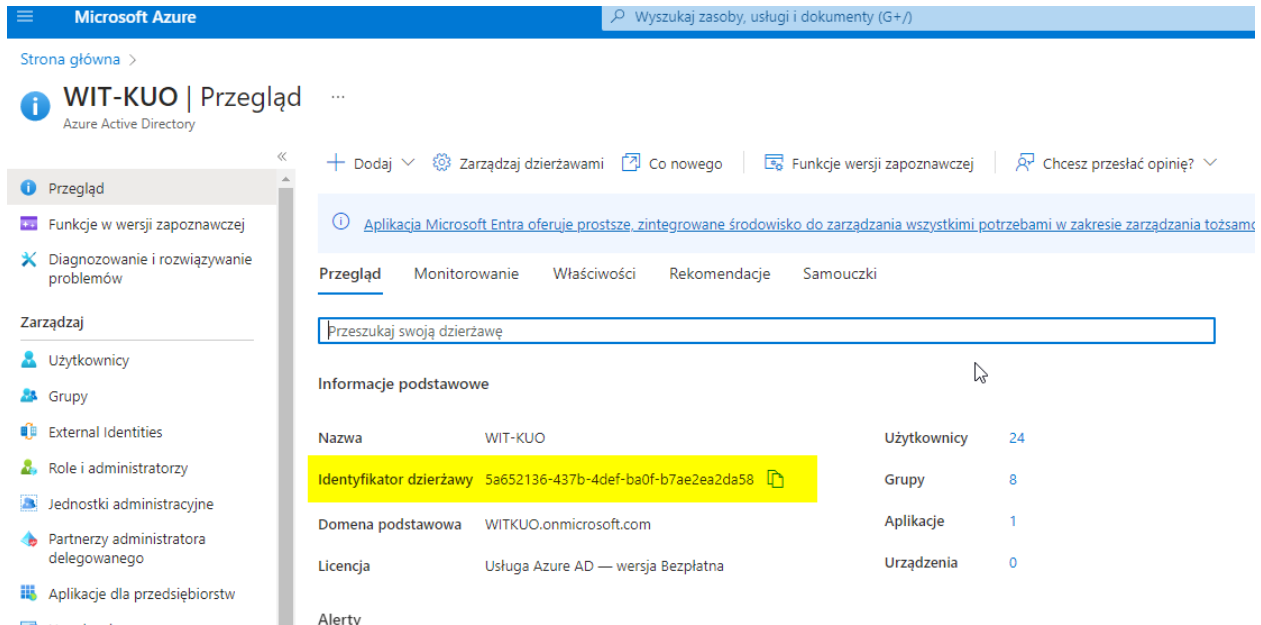


The screenshot shows a PowerShell terminal window with the title bar 'C:\Users\Mieszko - PowerShell'. The command 'Get-AzureADTenantDetail' has been executed, resulting in a table with three columns: 'ObjectId', 'DisplayName', and 'VerifiedDomain'. The output row shows the ObjectId '5a652136-437b-4def-ba0f-b7ae2ea2da58', the DisplayName 'WIT-KU0', and the VerifiedDomain is empty.

ObjectId	DisplayName	VerifiedDomain
5a652136-437b-4def-ba0f-b7ae2ea2da58	WIT-KU0	

Ilustracja 2: Identyfikator dzierżawy PowerShell

lub odnaleźć w Portalu Azure



Ilustracja 3: Identyfikator dzierżawy – Portal Azure

Skrypt wymaga dostępu do AzureAD, dlatego konieczne jest uprzednie zalogowanie za pomocą odpowiednich danych. Po uruchomieniu użytkownik każdorazowo musi podać dane do logowania w Azure.

Praca automatyczna

Jeśli pojawi się potrzeba zautomatyzowania działania skryptu należy wygenerować plik **credentials.xml** i umieścić go w katalogu ze skryptem.

Dodatkowo należy w skrypcie zamienić wiersz:

```
$Credentials = Get-Credential
```

na:

```
$Credentials = Import-CliXml "credentials.xml"
```

Plik **credentials.xml** można wygenerować wykonując następujące polecenia PS:

Dokumentacja skryptu tworzącego użytkowników w Azure AD na podstawie plików *.csv

```
$Credentials = Get-Credential  
$Credentials | Export-CliXml "credentials.xml"
```

Dodatkowa konfiguracja

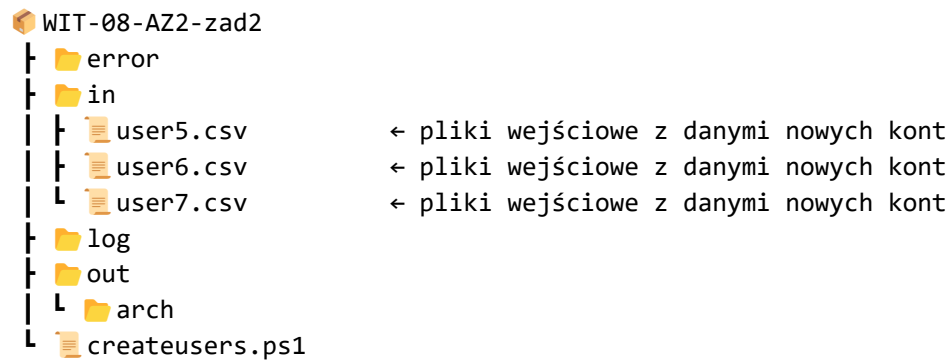
Modyfikując wartości zmiennych:

```
$inFolder = ".\in"  
$outFolder = ".\out"  
$errorFolder = ".\error"  
$logFolder = ".\log"
```

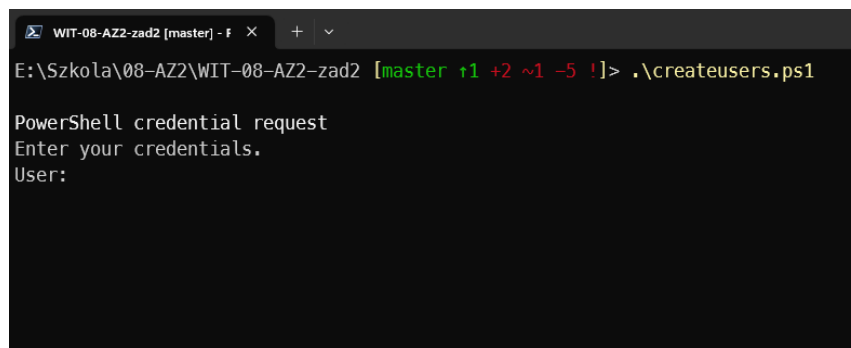
możemy zmienić miejsce odczytywania/zapisywania odpowiednich plików

Użycie skryptu

Przygotowane pliki ***.csv** z danymi użytkowników należy umieścić w katalogu **.\in** i uruchomić skrypt **createusers.ps1**.



Po uruchomieniu, należy podać dane logowania do AD:



Ilustracja 4: Logowanie do AD

Po poprawnym uwierzytelnieniu skrypt kontynuuje pracę:

```
WIT-08-AZ2-zad2 [master] - F x + v
E:\Szkola\08-AZ2\WIT-08-AZ2-zad2 [master +1 +2 ~1 -5 !]> .\createusers.ps1

PowerShell credential request
Enter your credentials.
User: mieszkourbanowski@witkuo.onmicrosoft.com
Password for user mieszkourbanowski@witkuo.onmicrosoft.com: *****

.\in\user5.csv
Account Environment Tenant TenantDomain AccountType
MieszkoUrbanowski@WITKU0.onmicrosoft.com AzureCloud 5a652136-437b-4def-ba0f-b7ae2ea2da58 WITKU0.onmicrosoft.com
.\out\2023-04-23_232639_326.txt
✔ Użytkownik Marek Platfoos został dodany.

.\in\user6.csv
.\out\2023-04-23_232640_367.txt
✔ Użytkownik Paweł Żonkiel został dodany.

.\in\user7.csv
.\out\2023-04-23_232641_242.txt
✔ Użytkownik Adaś Platfoos został dodany.

Naciśnij dowolny klawisz
```

Ilustracja 5: Wynik działania skryptu

W wyniku jego działania są tworzone pliki z hasłami dla nowych użytkowników oraz plik logów:

```
WIT-08-AZ2-zad2
├─ error
├─ in
├─ log
│  └─ 20230423_user_create.log ← logi
├─ out
│  └─ arch
│     ├── 20230423232639_user5.csv ← kopia plików wejściowych
│     ├── 20230423232640_user6.csv ← kopia plików wejściowych
│     └─ 20230423232641_user7.csv ← kopia plików wejściowych
├─ 2023-04-23_232639_326.txt ← pliki z danymi nowych kont
├─ 2023-04-23_232640_367.txt ← pliki z danymi nowych kont
└─ 2023-04-23_232641_242.txt ← pliki z danymi nowych kont
└─ createusers.ps1
```

Zawartość pliku logów:

```
2023-04-23 23:26:37 Przetwarzam plik .\in\user5.csv.
2023-04-23 23:26:39 SUCCESS: Użytkownik marek@witkuo.onmicrosoft.com Marek Platfoos został dodany.
```

2023-04-23 23:26:39 Przetwarzam plik .\in\user6.csv.
2023-04-23 23:26:40 SUCCESS: Użytkownik pawelzonk@witkuo.onmicrosoft.com Paweł Żonkiel został dodany.
2023-04-23 23:26:40 Przetwarzam plik .\in\user7.csv.
2023-04-23 23:26:41 SUCCESS: Użytkownik adamek@witkuo.onmicrosoft.com Adaś Platfoos został dodany.

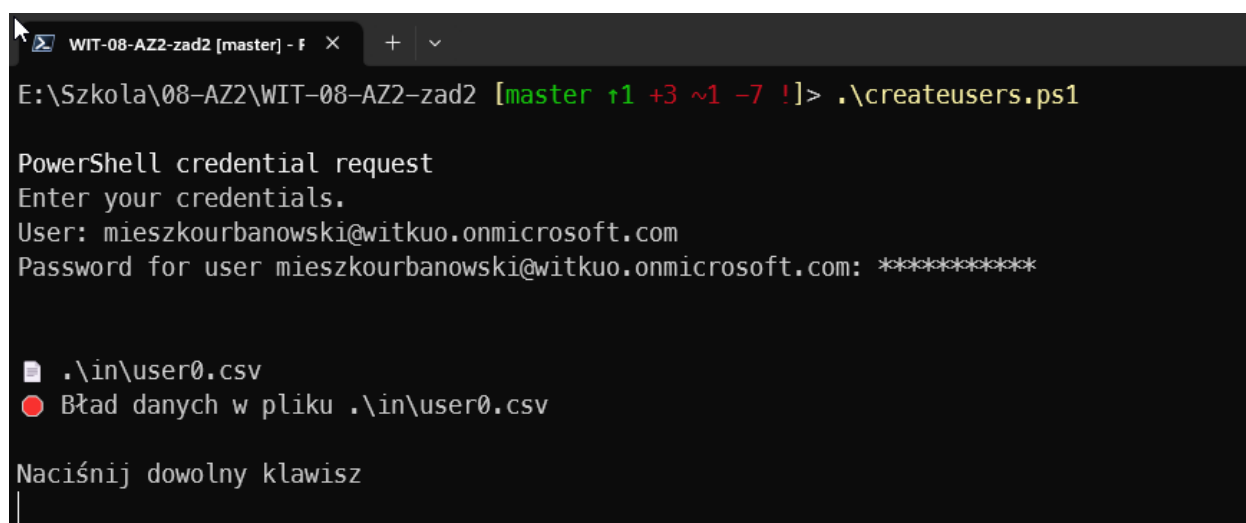
Obsługa błędów

Skrypt obsługuje różne przypadki błędów. W dalszej części omówienie kilku z nich.

Pusty plik wejściowy user0.csv

Zachowanie programu na pusty plik wejściowy. Pliki przed uruchomieniem skryptu:

```
WIT-08-AZ2-zad2
├── error
├── in
│   └── user0.csv
├── log
├── out
│   └── arch
└── createusers.ps1
```



```
WIT-08-AZ2-zad2 [master] - F X + v
E:\Szkola\08-AZ2\WIT-08-AZ2-zad2 [master +1 +3 ~1 -7 !]> .\createusers.ps1

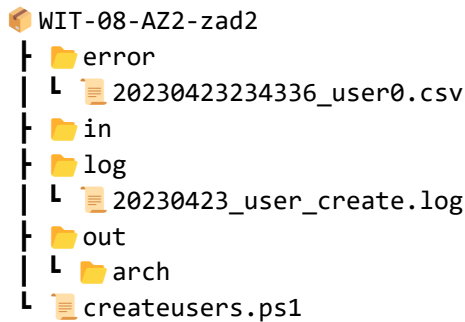
PowerShell credential request
Enter your credentials.
User: mieszkourbanowski@witkuo.onmicrosoft.com
Password for user mieszkourbanowski@witkuo.onmicrosoft.com: *****

. \in\user0.csv
● Błąd danych w pliku .\in\user0.csv

Naciśnij dowolny klawisz
```

Ilustracja 6: Wynik działania - pusty plik wejściowy

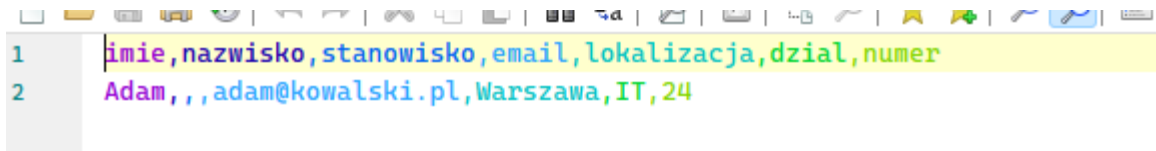
Skrypt wykrył że dane wejściowe nie są kompletne. Zgłosił błąd, taki sam komunikat dodał do pliku log-a, oraz przeniósł plik z błędnymi danymi do folderu **.\error** dopisując do jego nazwy bieżącą datę i godzinę.



Zawartość pliku logów:

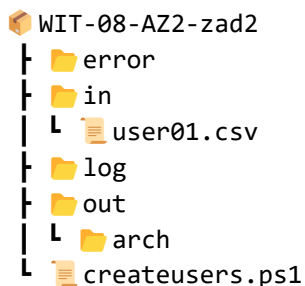
```
2023-04-23 23:43:36 Przetwarzam plik .\in\user0.csv.  
2023-04-23 23:43:36 ERROR Błąd danych w pliku .\in\user0.csv
```

Puste wartości w niektórych polach user01.csv



Ilustracja 7: Dane wejściowe - brak niektórych pól

Skrypt zachowuje się analogicznie jak powyżej.



Skrypt wykrył że dane wejściowe są błędne. Zgłosił błąd, taki sam komunikat dodał do pliku log-a, oraz przeniósł plik z błędnymi danymi do folderu **.error** dopisując do jego nazwy bieżącą datę i godzinę.

```
WIT-08-AZ2-zad2 [master] - F X + v
E:\Szkola\08-AZ2\WIT-08-AZ2-zad2 [master ↑1 +6 ~1 -8 !]> .\createusers.ps1

.\in\user01.csv
● Błąd danych w pliku .\in\user01.csv

Naciśnij dowolny klawisz
```

Ilustracja 8: Wynik działania - brak niektórych pól

```
WIT-08-AZ2-zad2
├── error
│   └── 20230423234736_user01.csv
├── in
├── log
│   └── 20230423_user_create.log
├── out
│   └── arch
└── createusers.ps1
```

Zawartość pliku logów:

```
2023-04-24 00:00:26 Przetwarzam plik .\in\user01.csv.
2023-04-24 00:00:26 ERROR Błąd danych w pliku .\in\user01.csv
```

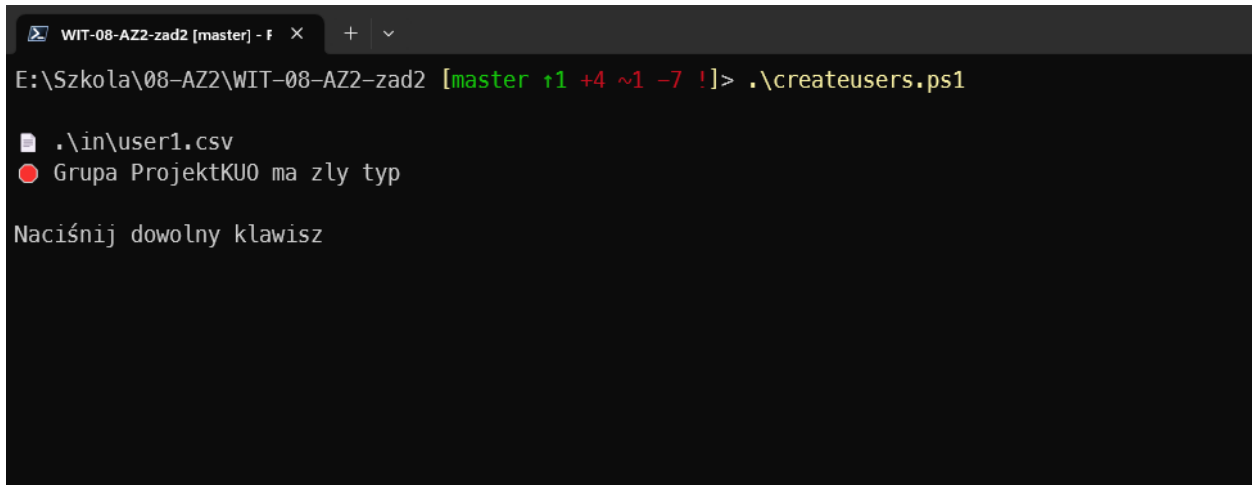
Błędny typ grupy user1.csv

```
1 | imie,nazwisko,stanowisko,email,lokalizacja,dzial,numer
2 | Adam,Kowalski,Support Junior,adam@kowalski.pl,Warszawa,ProjektKUO,24
```

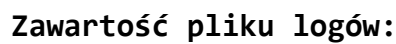
Ilustracja 9: Dane wejściowe - błędny typ grupy

Jako dział została podana poprawna nazwa grupy, ale jest to grupa dystrybucyjna. Skrypt zachowuje się analogicznie jak powyżej.

```
WIT-08-AZ2-zad2
├── error
```

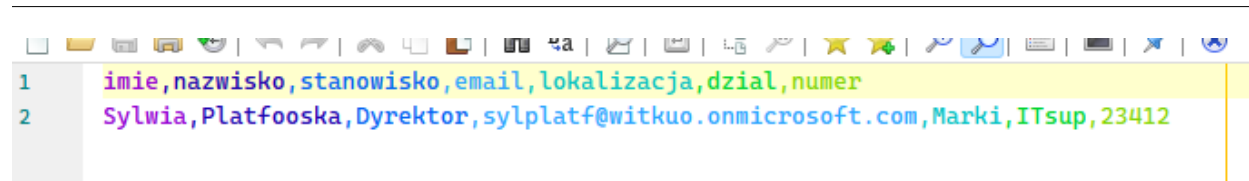


Skrypt wykrył że dane wejściowe są błędne. Zgłosił błąd, taki sam komunikat dodał do pliku log-a, oraz przeniósł plik z błędnymi danymi do folderu **.\\error** dopisując do jego nazwy bieżącą datę i godzinę.



Dokumentacja skryptu tworzącego użytkowników w Azure AD na podstawie plików *.csv

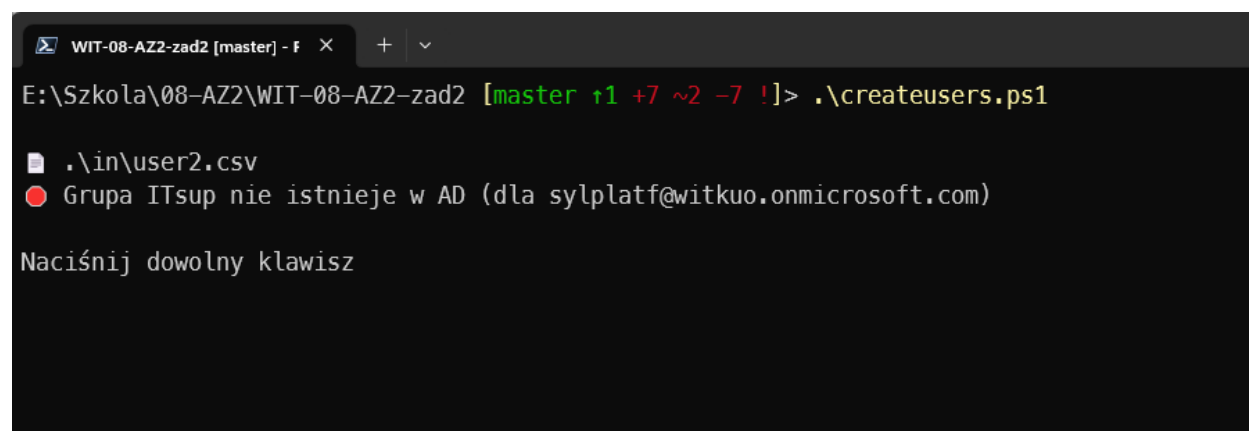
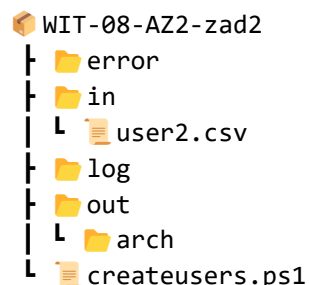
Brak grupy w AD user2.csv



1	imie,nazwisko,stanowisko,email,lokalizacja,dzial,numer
2	Sylwia,Platfooska,Dyrektor,sylplatf@witkuo.onmicrosoft.com,Marki,ITsup,23412

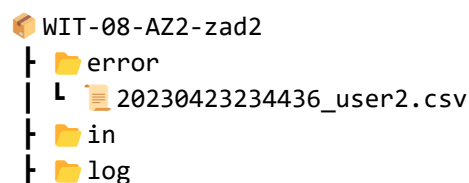
Ilustracja 11: Dane wejściowe - brak grupy w AD

Jako dział została podana nazwa która nie odpowiada żadnej grupie w AD.



Ilustracja 12: Wynik działania - brak grupy w AD

Skrypt wykrył że dane wejściowe są błędne. Zgłosił błąd, taki sam komunikat dodał do pliku log-a, oraz przeniósł plik z błędnymi danymi do folderu **.\error** dopisując do jego nazwy bieżącą datę i godzinę.

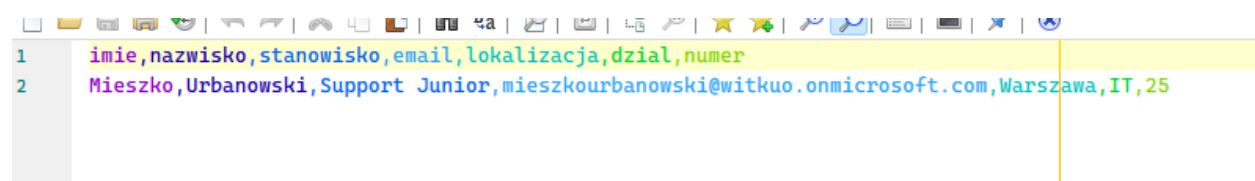


```
| L 20230423_user_create.log
|  }
|  L  }
|  L  out
|  L  arch
|  L  createusers.ps1
```

Zawartość pliku logów:

```
2023-04-24 00:04:59 Przetwarzam plik .\in\user2.csv.
2023-04-24 00:04:59 ERROR Grupa ITsup nie istnieje w AD (dla sylplatf@witkuo.onmicrosoft.com)
```

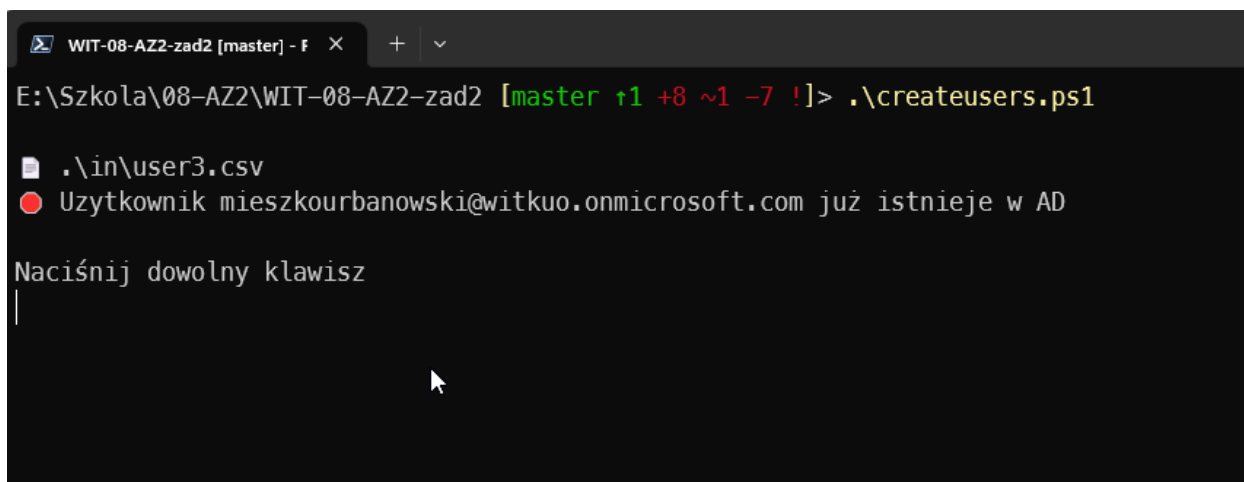
Użytkownik już istnieje user3.csv



Ilustracja 13: Dane wejściowe - użytkownik już istnieje

Skrypt sprawdza czy użytkownik o takim adresie e-mail już jest w bazie AD. Jeśli tak, nie zakłada nowego konta tylko zgłasza błąd.

```
WIT-08-AZ2-zad2
|  }
|  }
|  L  user3.csv
|  }
|  }
|  L  log
|  L  out
|  L  arch
|  L  createusers.ps1
```



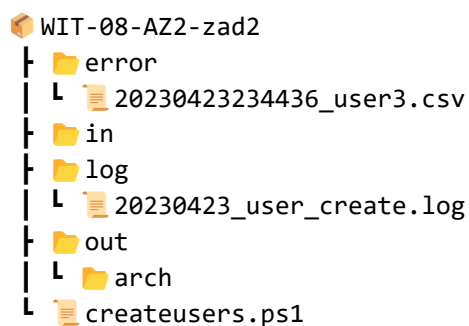
```
WIT-08-AZ2-zad2 [master] - F X + v
E:\Szkola\08-AZ2\WIT-08-AZ2-zad2 [master ↑1 +8 ~1 -7 !]> .\createusers.ps1

.\in\user3.csv
● Użytkownik mieszkourbanowski@witkuo.onmicrosoft.com już istnieje w AD

Naciśnij dowolny klawisz
|
```

Ilustracja 14: Wynik działania - użytkownik już istnieje

Skrypt wykrył że dane wejściowe są błędne. Zgłosił błąd, taki sam komunikat dodał do pliku log-a, oraz przeniósł plik z błędnymi danymi do folderu **.\error** dopisując do jego nazwy bieżącą datę i godzinę.



```
WIT-08-AZ2-zad2
├── error
│   └── 20230423234436_user3.csv
├── in
├── log
│   └── 20230423_user_create.log
├── out
│   └── arch
└── createusers.ps1
```

Zawartość pliku logów:

```
2023-04-24 00:08:23 Przetwarzam plik .\in\user3.csv.
2023-04-24 00:08:23 ERROR Użytkownik mieszkourbanowski@witkuo.onmicrosoft.com już istnieje w AD
```