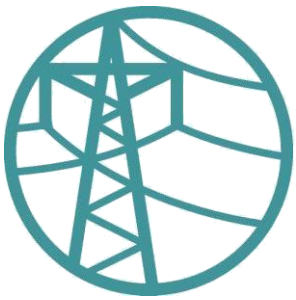


WYDZIAŁ
ELEKTROTECHNIKI
I INFORMATYKI
POLITECHNIKI RZESZOWSKIEJ

Jakub Skrzynecki

Technologie i oprogramowanie Chmurowe

Wdrożenie uwierzytelniania SSO w chmurze z wykorzystaniem Authentik i OIDC oraz integracja z aplikacjami SaaS w chmurze

Spis treści

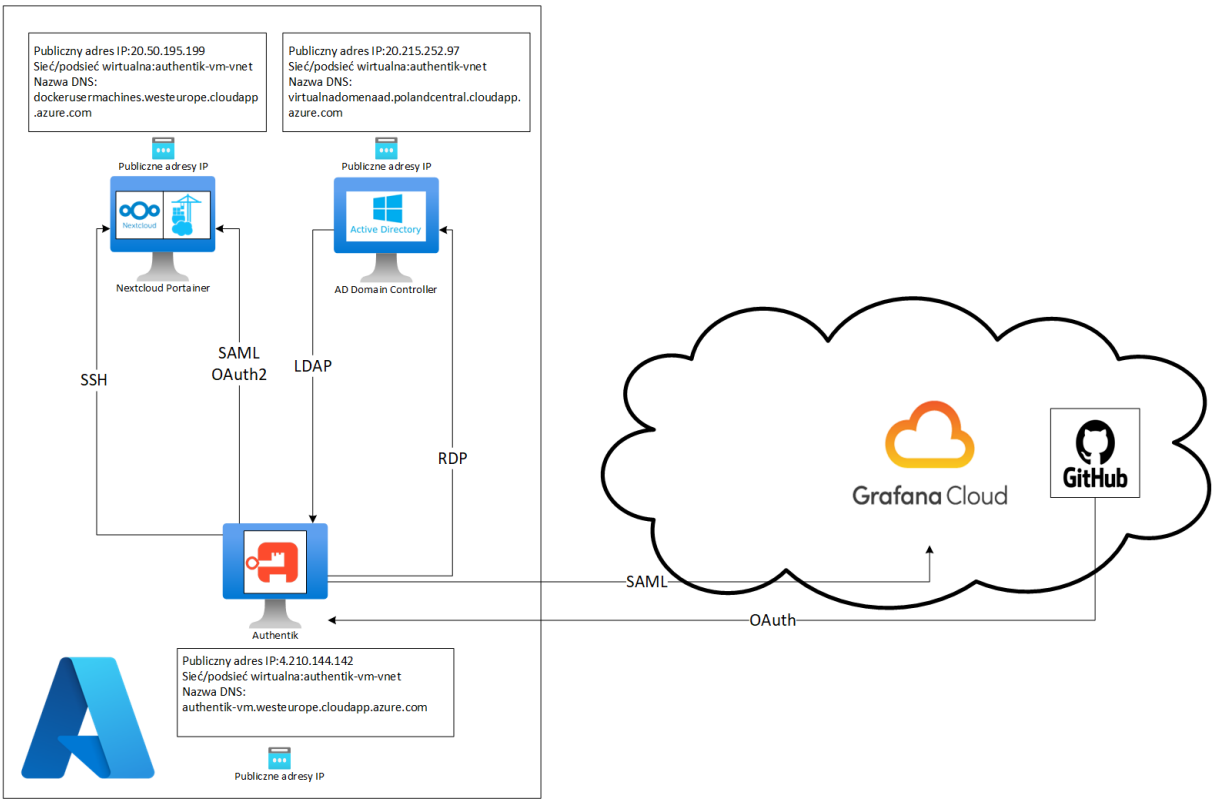
1 Wprowadzenie

Niniejsza dokumentacja przedstawia realizację kompleksowego systemu Single Sign-On (SSO) wdrożonego w chmurze Microsoft Azure z wykorzystaniem platformy Authentik jako centralnego Identity Provider (IdP). Projekt obejmuje implementację nowoczesnych protokołów uwierzytelniania OAuth2 i SAML 2.0 w celu integracji z aplikacjami typu SaaS (Software as a Service) w środowisku chmurowym oraz integrację z kontrolerem domeny Active Directory przez protokół LDAP.

Architektura rozwiązania opiera się na trzech głównych filarach: centralnym Identity Provider (Authentik), zintegrowanym źródle tożsamości (Active Directory) oraz aplikacjach SaaS działających jako Service Providers.

Projekt obejmuje także implementację trzech maszyn wirtualnych z różnymi systemami operacyjnymi oraz zarządzanie kontenerami.

2 Schemat Infrastruktury



Rys. 2.1 Schemat architektury systemu w chmurze Microsoft Azure przedstawiający trzy maszyny wirtualne i przepływy danych między nimi

Przedstawiony powyżej schemat infrastruktury ilustruje kompleksową architekturę systemu składającą się z trzech maszyn wirtualnych rozmieszczonych w różnych regionach Azure. Centralne miejsce w architekturze zajmuje maszyna z systemem Authentik, która pełni rolę głównego dostawcy tożsamości (Identity Provider) dla całego środowiska. Wszystkie komponenty zostały połączone w bezpieczną sieć komunikacyjną wykorzystującą protokoły HTTPS, OAuth2, SAML oraz LDAP.

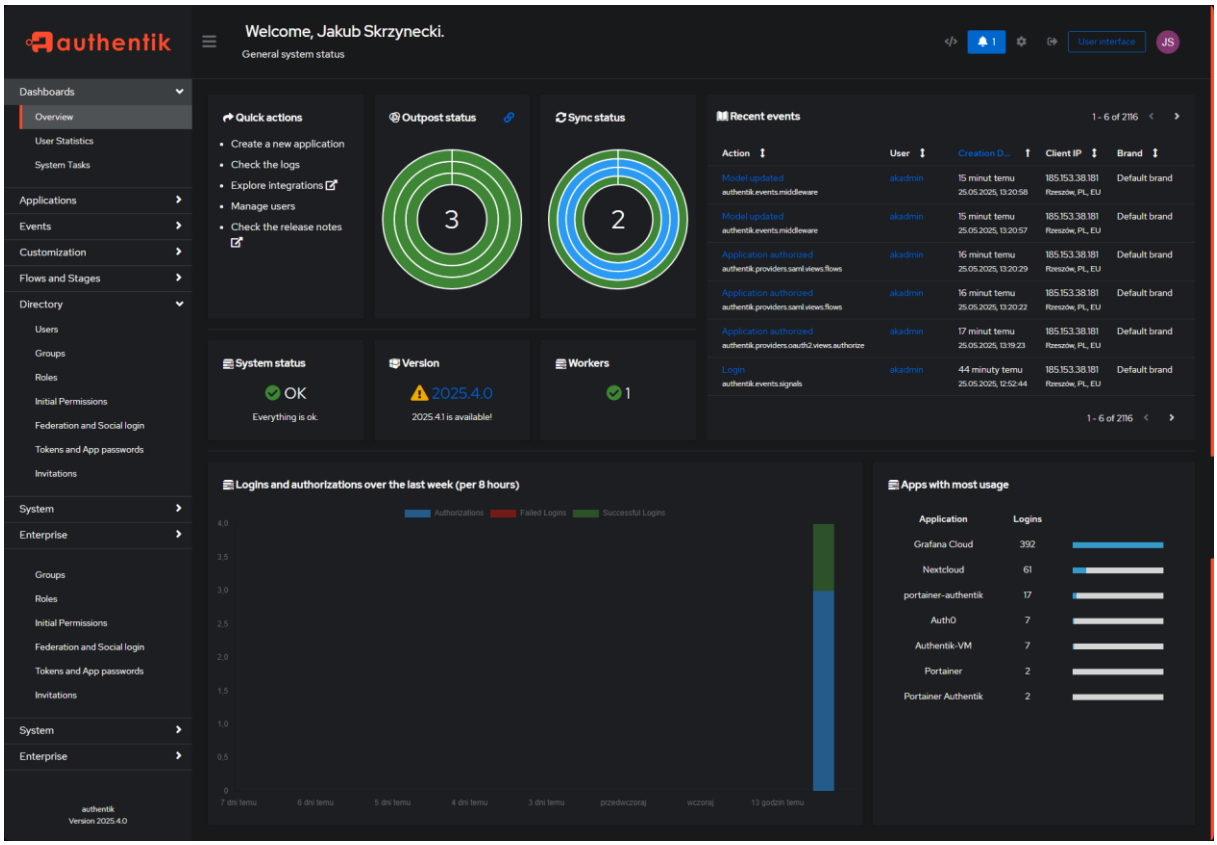
3 Komponenty systemu

3.1 Maszyna wirtualna z systemem Authentik

Główny serwer Identity Provider został wdrożony na maszynie wirtualnej z systemem Ubuntu Linux. Maszyna została skonfigurowana z publicznym adresem DNS authentik-vm.westeurope.cloudapp.azure.com, zapewniając globalny dostęp do usług uwierzytelniania. Na tej maszynie zostały wdrożone następujące komponenty:

3.1.1 Kontener Docker z Aplikacją Authentik

Aplikacja Authentik została wdrożona w kontenerze Docker jako główny dostawca tożsamości systemu. Authentik oferuje kompleksowe rozwiązanie do zarządzania uwierzytelnianiem i autoryzacją, obsługując nowoczesne protokoły bezpieczeństwa oraz umożliwiając integrację z zewnętrznymi źródłami tożsamości.



Rys. 3.1 Interfejs administracyjny systemu Authentik z widokiem głównego panelu zarządzania

Konfiguracja Authentik została przeprowadzona zgodnie z najlepszymi praktykami bezpieczeństwa, uwzględniając właściwe ustawienia protokołów uwierzytelniania oraz polityk dostępu. System został skonfigurowany do nasłuchiwania na porcie 9000, co zapewnia separację ruchu od standardowych portów HTTP.

3.1.2 Serwer Nginx jako reverse proxy

Na tej samej maszynie został zainstalowany i skonfigurowany serwer Nginx pełniący funkcję reverse proxy. Nginx przekierowuje ruch z portu 80 na port 9000, gdzie działa aplikacja Authentik. Konfiguracja umożliwia dostęp do systemu poprzez przyjazny adres DNS.

Adres DNS maszyny: authentik-vm.westeurope.cloudapp.azure.com

Konfiguracja Nginx została zoptymalizowana pod kątem bezpieczeństwa i wydajności, uwzględniając odpowiednie nagłówki bezpieczeństwa oraz konfigurację SSL/TLS.

3.2 Maszyna wirtualna z aplikacjami SaaS

Druga maszyna wirtualna hostuje aplikacje SaaS zintegrowane z systemem SSO. Serwer z systemem Ubuntu Linux został skonfigurowany z adresem DNS dockerusermachines.westeurope.cloudapp.azure.com i zawiera aplikacje demonstrujące różne scenariusze integracji SSO. Wszystkie aplikacje zostały wdrożone w kontenerach Docker, co zapewnia izolację i łatwość zarządzania.

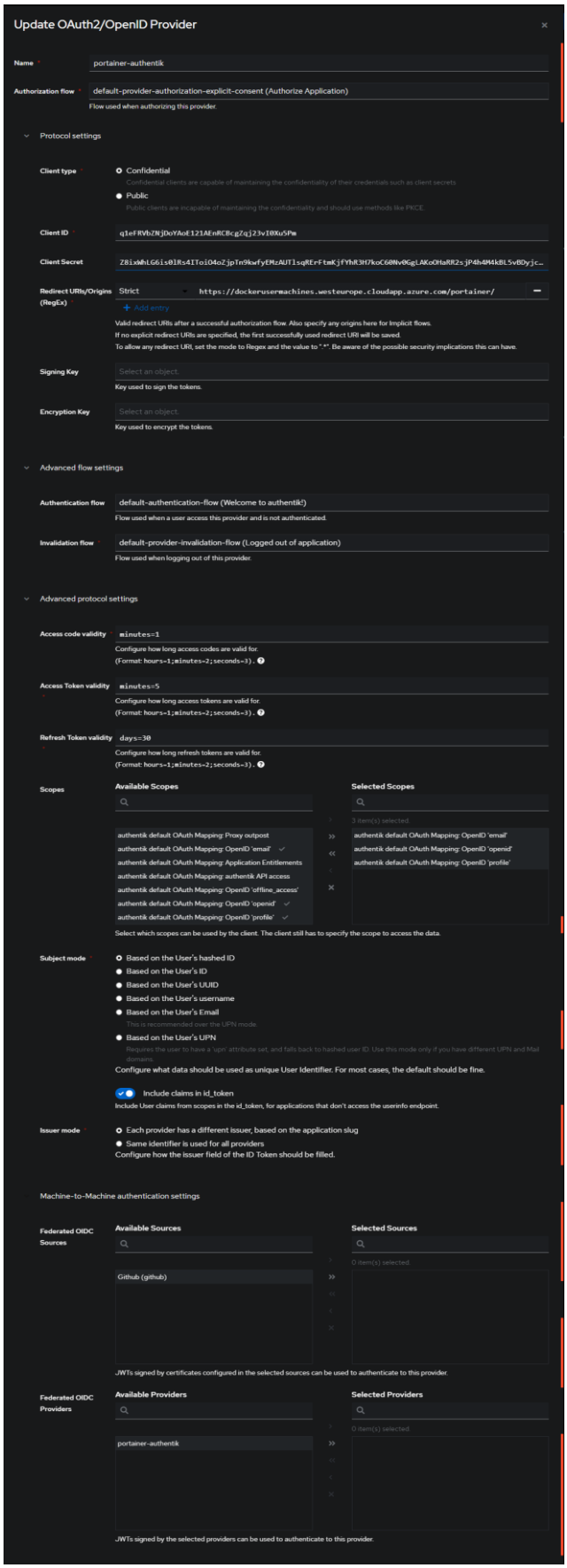
3.2.1 Portainer z integracją OAuth2

Portainer został skonfigurowany jako narzędzie do zarządzania kontenerami Docker z pełną integracją z systemem Authentik poprzez protokół OAuth2. Kluczowe rzeczy potrzebne do integracji:

Client ID – identyfikator aplikacji.

Client Secret – tajny klucz klienta.

Redirect URI – adres, na który Authentik odeśle użytkownika po zalogowaniu, w naszym przypadku: dockerusermachines.westeurope.cloudapp.azure.com/portainer/

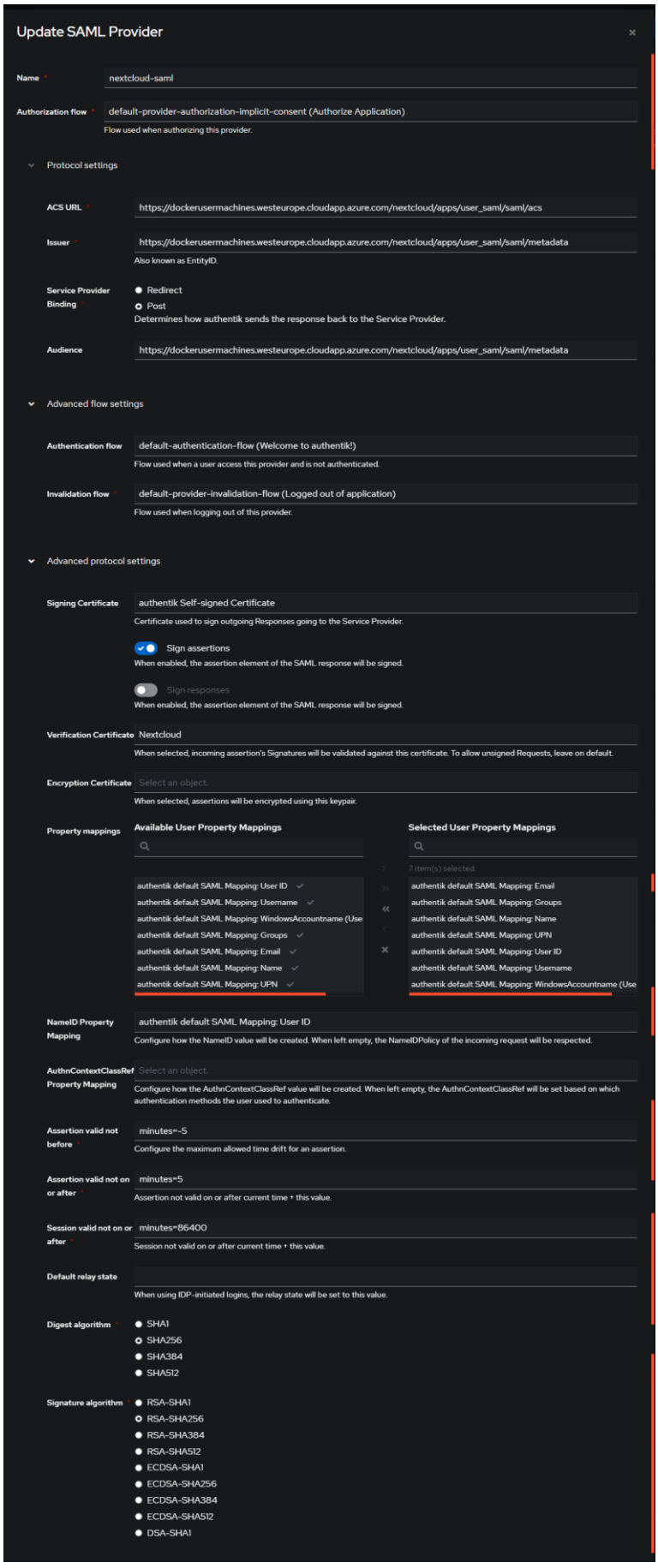

Update OAuth2/OpenID Provider
Name
portainer-authentik
Authorization flow
default-provider-authorization-explicit-consent (Authorize Application)
Flow used when authorizing this provider.
Protocol settings
Client type
Confidential
Confidential clients are capable of maintaining the confidentiality of their credentials such as client secrets
Public
Public clients are incapable of maintaining the confidentiality and should use methods like PKCE.
Client ID
q1eFRVbZNjDoYAoE121AEnRCBcgZqj23v10Xu5Pm
Client Secret
Z8ixWhLG6is0lRs4ITOiO4oZjpTn9kwfyEMzAUTlsqRErFtmKjfYhR3H7koC60Nv0GgLAKoOHaRR2sjP4b4M4kBL5vBDyjc…
Redirect URIs/Origins (RegEx)
Strict
https://dockerusermachines.westeurope.cloudapp.azure.com/portainer/
Add entry
Valid redirect URIs after a successful authorization flow. Also specify any origins here for Implicit flows.
If no explicit redirect URIs are specified, the first successfully used redirect URI will be saved.
To allow any redirect URI, set the mode to Regex and the value to ".*". Be aware of the possible security implications this can have.
Signing Key
Select an object.
Key used to sign the tokens.
Encryption Key
Select an object.
Key used to encrypt the tokens.
Advanced flow settings
Authentication flow
default-authentication-flow (Welcome to authentik!)
Flow used when a user access this provider and is not authenticated.
Invalidation flow
default-provider-invalidation-flow (Logged out of application)
Flow used when logging out of this provider.
Advanced protocol settings
Access code validity
minutes=1
Configure how long access codes are valid for.
(Format: hours=1;minutes=2;seconds=3).
Access Token validity
minutes=5
Configure how long access tokens are valid for.
(Format: hours=1;minutes=2;seconds=3).
Refresh Token validity
days=30
Configure how long refresh tokens are valid for.
(Format: hours=1;minutes=2;seconds=3).
Scopes
Available Scopes
authentik default OAuth Mapping: Proxy outpost
authentik default OAuth Mapping: OpenID 'email'
authentik default OAuth Mapping: Application Entitlements
authentik default OAuth Mapping: authentik API access
authentik default OAuth Mapping: OpenID 'offline_access'
authentik default OAuth Mapping: OpenID 'openid'
authentik default OAuth Mapping: OpenID 'profile'
Selected Scopes
3 item(s) selected.
authentik default OAuth Mapping: OpenID 'email'
authentik default OAuth Mapping: OpenID 'openid'
authentik default OAuth Mapping: OpenID 'profile'
Select which scopes can be used by the client. The client still has to specify the scope to access the data.
Subject mode
Based on the User's hashed ID
Based on the User's ID
Based on the User's UUID
Based on the User's username
Based on the User's Email
This is recommended over the UPN mode.
Based on the User's UPN
Requires the user to have a 'upn' attribute set, and falls back to hashed user ID. Use this mode only if you have different UPN and Mail domains.
Configure what data should be used as unique User Identifier. For most cases, the default should be fine.
Include claims in id_token
Include User claims from scopes in the id_token, for applications that don't access the userinfo endpoint.
Issuer mode
Each provider has a different issuer, based on the application slug
Same identifier is used for all providers
Configure how the issuer field of the ID Token should be filled.
Machine-to-Machine authentication settings
Federated OIDC Sources
Available Sources
Github (github)
Selected Sources
0 item(s) selected.
JWTs signed by certificates configured in the selected sources can be used to authenticate to this provider.
Federated OIDC Providers
Available Providers
portainer-authentik
Selected Providers
0 item(s) selected.
JWTs signed by the selected providers can be used to authenticate to this provider.

Rys. 3.2 Konfiguracja uwierzytelniania OAuth2 w aplikacji Portainer z automatycznym tworzeniem użytkowników

Konfiguracja OAuth2 w Portainer umożliwia automatyczne tworzenie nowych użytkowników podczas pierwszego logowania, co znacznie upraszcza proces onboardingu nowych członków zespołu. System automatycznie pobiera informacje o użytkowniku z Authentik i tworzy odpowiednie konto w Portainer z właściwymi uprawnieniami.

3.2.2 Nextcloud z integracją SAML

Platforma Nextcloud została zintegrowana z systemem SSO przy wykorzystaniu protokołu SAML 2.0. Konfiguracja umożliwia jednokrotne logowanie użytkowników z automatycznym provisioningiem kont oraz synchronizacją atrybutów użytkowników. Aplikacja została skonfigurowana jako Service Provider (SP) z Authentik pełniącym rolę Identity Provider (IdP).


Update SAML Provider
Name
nextcloud-saml
Authorization flow
default-provider-authorization-implicit-consent (Authorize Application)
Flow used when authorizing this provider.
Protocol settings
ACS URL
https://dockerusermachines.westeurope.cloudapp.azure.com/nextcloud/apps/user_saml/saml/acs
Issuer
https://dockerusermachines.westeurope.cloudapp.azure.com/nextcloud/apps/user_saml/saml/metadata
Also known as EntityID.
Service Provider Binding
Redirect
Post
Determines how authentik sends the response back to the Service Provider.
Audience
https://dockerusermachines.westeurope.cloudapp.azure.com/nextcloud/apps/user_saml/saml/metadata
Advanced flow settings
Authentication flow
default-authentication-flow (Welcome to authentik!)
Flow used when a user access this provider and is not authenticated.
Invalidation flow
default-provider-invalidation-flow (Logged out of application)
Flow used when logging out of this provider.
Advanced protocol settings
Signing Certificate
authentik Self-signed Certificate
Certificate used to sign outgoing Responses going to the Service Provider.
Sign assertions
When enabled, the assertion element of the SAML response will be signed.
Sign responses
When enabled, the assertion element of the SAML response will be signed.
Verification Certificate
Nextcloud
When selected, incoming assertion's Signatures will be validated against this certificate. To allow unsigned Requests, leave on default.
Encryption Certificate
Select an object.
When selected, assertions will be encrypted using this keypair.
Property mappings
Available User Property Mappings
authentik default SAML Mapping: User ID
authentik default SAML Mapping: Username
authentik default SAML Mapping: WindowsAccountname (Use
authentik default SAML Mapping: Groups
authentik default SAML Mapping: Email
authentik default SAML Mapping: Name
authentik default SAML Mapping: UPN
Selected User Property Mappings
7 item(s) selected.
authentik default SAML Mapping: Email
authentik default SAML Mapping: Groups
authentik default SAML Mapping: Name
authentik default SAML Mapping: UPN
authentik default SAML Mapping: User ID
authentik default SAML Mapping: Username
authentik default SAML Mapping: WindowsAccountname (Use
NameID Property Mapping
authentik default SAML Mapping: User ID
Configure how the NameID value will be created. When left empty, the NameIDPolicy of the incoming request will be respected.
AuthnContextClassRef Property Mapping
Select an object.
Configure how the AuthnContextClassRef value will be created. When left empty, the AuthnContextClassRef will be set based on which authentication methods the user used to authenticate.
Assertion valid not before
minutes=-5
Configure the maximum allowed time drift for an assertion.
Assertion valid not on or after
minutes=5
Assertion not valid on or after current time + this value.
Session valid not on or after
minutes=86400
Session not valid on or after current time + this value.
Default relay state
When using IDP-initiated logins, the relay state will be set to this value.
Digest algorithm
SHA1
SHA256
SHA384
SHA512
Signature algorithm
RSA-SHA1
RSA-SHA256
RSA-SHA384
RSA-SHA512
ECDSA-SHA1
ECDSA-SHA256
ECDSA-SHA384
ECDSA-SHA512
DSA-SHA1

Rys. 3.3 Konfiguracja dostawcy tożsamości SAML w aplikacji Nextcloud

Implementacja protokołu SAML w Nextcloud zapewnia bezpieczne jednokrotne logowanie (Single Sign-On) dla użytkowników. Konfiguracja uwzględnia mapowanie atrybutów użytkownika oraz automatyczne przydzielanie uprawnień na podstawie członkostwa w grupach zdefiniowanych w Active Directory.

3.2.3 Nginx jako kontener Docker

Na tej maszynie Nginx został wdrożony jako kontener Docker, pełniąc funkcję reverse proxy dla aplikacji Portainer i Nextcloud. Konfiguracja konteneryzowanego Nginx zapewnia wysoką dostępność i łatwość skalowania, a także umożliwia centralne zarządzanie konfiguracją routing-u dla wszystkich aplikacji użytkownych.

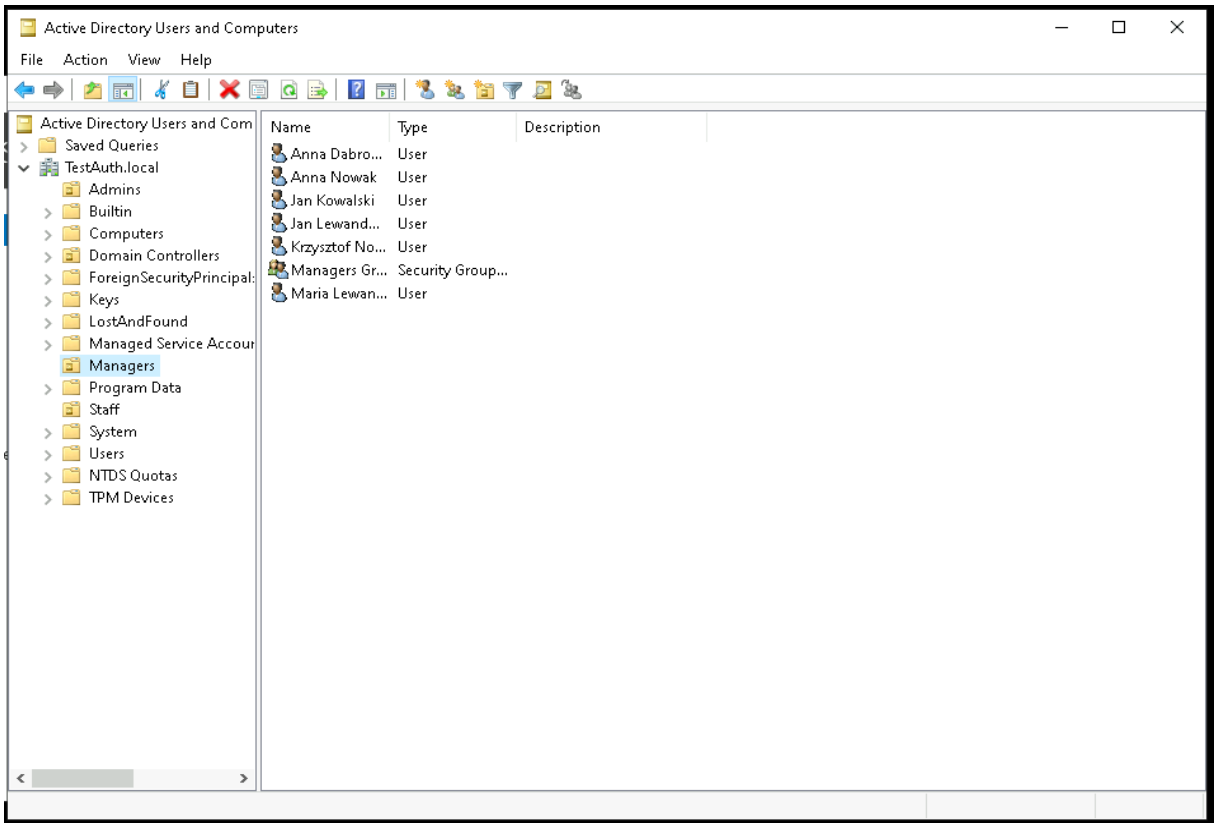
3.3 Kontroler domeny Windows Server 2022

Trzecia maszyna wirtualna została skonfigurowana z systemem Windows Server 2022 Datacenter jako kontroler domeny Active Directory.

3.3.1 Konfiguracja Active Directory Domain Services

Na maszynie została utworzona domena Active Directory z pełną funkcjonalnością kontrolera domeny.

Adres DNS kontrolera domeny: virtualnadomenaad.polandcentral.cloudapp.azure.com



Rys. 3.4 Struktura organizacyjna grup użytkowników w Active Directory Users and Computers

Struktura organizacyjna Active Directory przedstawiona na rysunku pokazuje hierarchię grup użytkowników zaprojektowaną z myślą o integracji SSO. Każda grupa została skonfigurowana z odpowiednimi atrybutami LDAP, które są następnie mapowane w systemie Authentik na role w aplikacjach SaaS.

3.3.2 Grupy użytkowników

W ramach domeny Active Directory zostały utworzone następujące grupy organizacyjne:

Staff Group - grupa przeznaczona dla pracowników standardowych z podstawowymi uprawnieniami dostępu do aplikacji użytkownych. Członkowie tej grupy mają dostęp do funkcji podstawowych w systemach Portainer i Nextcloud.

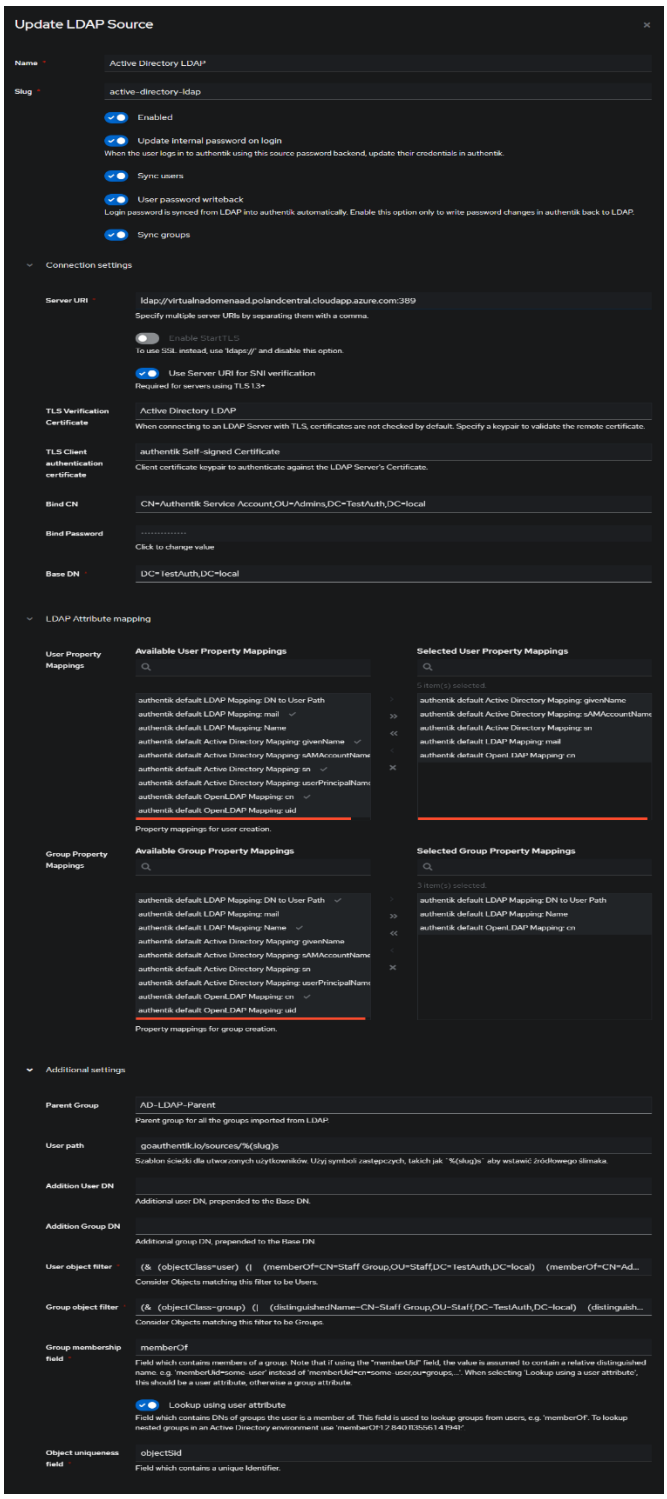
Admin Group - grupa administratorów systemu z pełnymi uprawnieniami do zarządzania infrastrukturą. Członkowie tej grupy posiadają uprawnienia administracyjne we wszystkich komponentach systemu.

Managers Group - grupa kierownicza z rozszerzonymi uprawnieniami do zarządzania zespołami i projektami. Członkowie tej grupy mają dostęp do funkcji zarządzania użytkownikami oraz zaawansowanych opcji konfiguracyjnych.

4 Integracje i protokoły komunikacji

4.1 Integracja LDAP z Active Directory

System Authentik został skonfigurowany do komunikacji z kontrolerem domeny poprzez protokół LDAP, co umożliwia synchronizację użytkowników i grup z Active Directory.

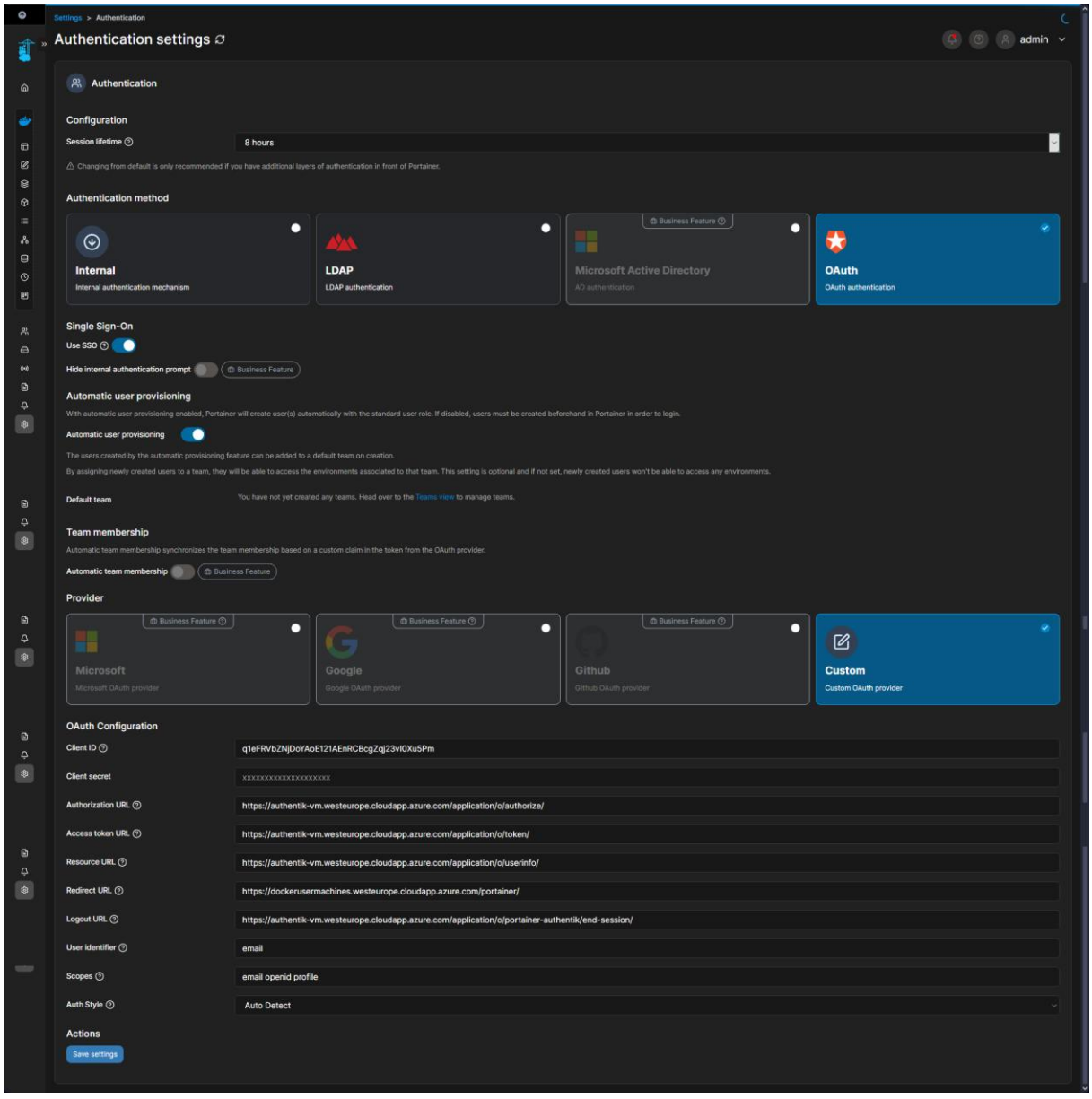


Rys. 4.1 Konfiguracja dostawcy LDAP w systemie Authentik z parametrami połączenia do Active Directory

Konfiguracja LDAP zapewnia automatyczną synchronizację struktur organizacyjnych oraz atrybutów użytkowników między Active Directory a systemem Authentik. Proces synchronizacji uwzględnia mapowanie grup oraz przekazywanie informacji o członkostwie w grupach do aplikacji końcowych.

4.2 Przepływy uwierzytelniania OAuth2

Protokół OAuth2 został zaimplementowany dla aplikacji Portainer, zapewniając bezpieczny proces autoryzacji z automatycznym tworzeniem kont użytkowników.



Rys. 4.2 Konfiguracja dostawcy tożsamości OAuth2 w Portainer

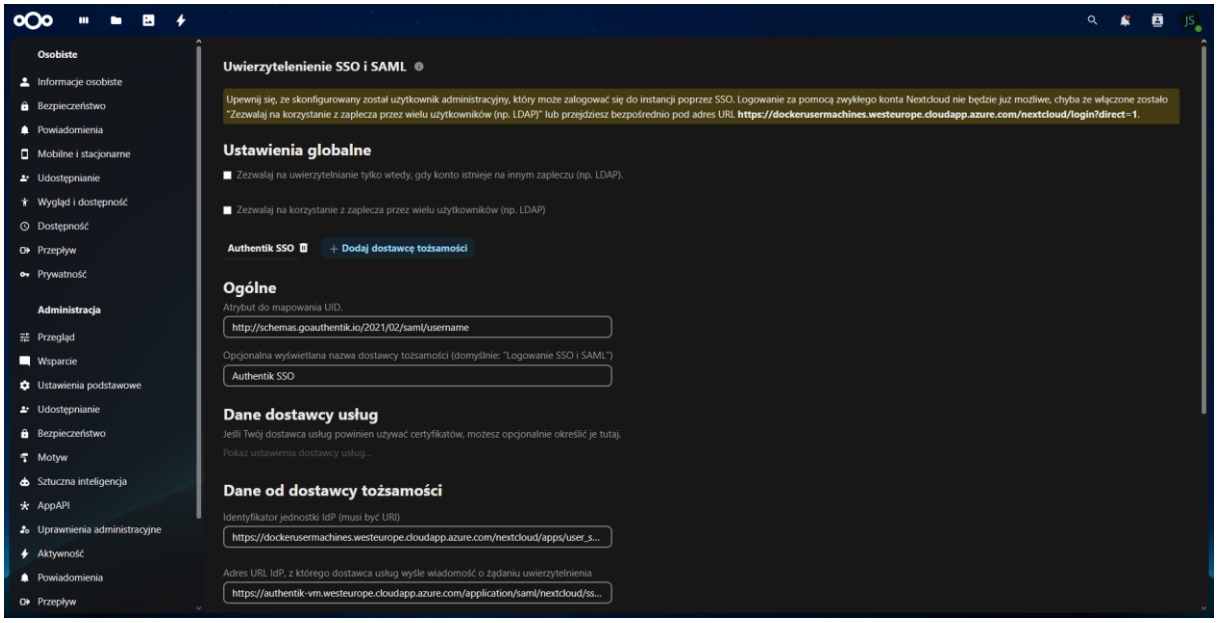
Do pełnej konfiguracji dostawcy tożsamości w portainer potrzebne były:

- Client ID
- Client Secret
- Adresy URL (Authorization URL, Access Token URL, itp.)
- User indetifier
- Scopes

Implementacja OAuth2 uwzględnia wszystkie najlepsze praktyki bezpieczeństwa, w tym użycie bezpiecznych tokenów dostępu, właściwą konfigurację zakresów uprawnień oraz mechanizmy odświeżania tokenów.

4.3 Integracja SAML z Nextcloud

Protokół SAML został skonfigurowany do zapewnienia jednokrotnego logowania w aplikacji Nextcloud z pełnym mapowaniem atrybutów użytkownika.



Rys. 4.3 Szczegółowa konfiguracja dostawcy tożsamości SAML w Nextcloud z mapowaniem atrybutów

Konfiguracja SAML umożliwia przekazywanie informacji o użytkowniku oraz jego przynależności do grup, co pozwala na automatyczne przydzielanie uprawnień w Nextcloud na podstawie struktury organizacyjnej zdefiniowanej w Active Directory.

5 Proces implementacji

5.1 Przygotowanie infrastruktury Azure

Implementacja rozpoczęła się od utworzenia infrastruktury chmurowej w Microsoft Azure. Proces obejmował konfigurację trzech maszyn wirtualnych w różnych regionach Azure, konfigurację sieci wirtualnej oraz zabezpieczeń sieciowych. Każda maszyna została skonfigurowana z publicznymi adresami IP oraz nazwami DNS umożliwiającymi dostęp z internetu.

5.2 Konfiguracja systemu Authentik

Na pierwszej maszynie wirtualnej została przeprowadzona instalacja i konfiguracja Authentik jako centralnego Identity Provider. Proces obejmował instalację Docker, utworzenie konfiguracji docker-compose oraz wstępną konfigurację systemu uwierzytelniania.

Kluczowe działania konfiguracyjne:

- Instalacja Docker Engine i Docker Compose na Ubuntu Linux
- Utworzenie pliku docker-compose.yml z konfiguracją Authentik
- Instalacja i konfiguracja Nginx jako reverse proxy z terminacją SSL
- Generowanie certyfikatów SSL self-signed dla szyfrowania komunikacji
- Wstępna konfiguracja Authentik przez interfejs webowy

5.3 Implementacja kontrolera domeny

Na maszynie z systemem Windows Server 2022 została przeprowadzona instalacja i konfiguracja Active Directory Domain Services. Proces obejmował promocję serwera do roli kontrolera domeny, konfigurację usług DNS oraz utworzenie struktury organizacyjnej.

Proces konfiguracji kontrolera domeny:

- Instalacja roli Active Directory Domain Services
- Promocja serwera do kontrolera domeny z nową domeną
- Konfiguracja usług DNS dla rozwiązywania nazw w domenie
- Utworzenie jednostek organizacyjnych (OU) dla różnych typów użytkowników
- Konfiguracja polityk bezpieczeństwa domeny

5.4 Integracja LDAP

Po skonfigurowaniu kontrolera domeny została zrealizowana integracja LDAP między Authentik a Active Directory. Konfiguracja obejmowała utworzenie dedykowanego konta usługi dla synchronizacji LDAP oraz skonfigurowanie parametrów połączenia w Authentik.

Elementy integracji LDAP:

- Utworzenie konta usługi dla synchronizacji LDAP w Active Directory
- Konfiguracja uprawnień konta usługi do odczytu obiektów katalogowych
- Konfiguracja LDAP Source w Authentik z parametrami połączenia
- Mapowanie atrybutów LDAP na pola użytkowników w Authentik
- Konfiguracja synchronizacji grup i automatycznego provisioning

5.5 Wdrożenie aplikacji SaaS

Piąty etap obejmował wdrożenie aplikacji Portainer i Nextcloud na drugiej maszynie wirtualnej. Proces rozpoczął się od konfiguracji Docker oraz utworzenia odpowiednich kontenerów. Następnie skonfigurowano Nginx jako reverse proxy oraz utworzono rekordy DNS dla dostępu do aplikacji.

Wdrożenie aplikacji obejmowało:

- Instalację Docker i Docker Compose na maszynie docelowej
- Konfigurację Portainer z integracją OIDC
- Wdrożenie Nextcloud z konfiguracją SAML SSO
- Konfigurację Nginx jako reverse proxy dla aplikacji SaaS
- Testowanie połączeń sieciowych między

5.6 Konfiguracja protokołów SSO

Ostatni etap polegał na konfiguracji protokołów OAuth2 i SAML dla integracji aplikacji z systemem Authentik. Proces obejmował utworzenie aplikacji w Authentik, konfigurację dostawców tożsamości w aplikacjach końcowych oraz testowanie pełnych przepływów uwierzytelniania.

Konfiguracja protokołów SSO:

- Rejestracja Portainer jako OAuth2 Application w Authentik
- Konfiguracja Nextcloud jako SAML Service Provider
- Wymiana metadanych SAML między Authentik a Nextcloud
- Konfiguracja mapowania atrybutów i grup dla każdej aplikacji

6 Wnioski

W ramach projektu zrealizowano wdrożenie systemu uwierzytelniania SSO (Single Sign-On) w środowisku chmurowym Microsoft Azure z wykorzystaniem platformy Authentik

jako centralnego dostawcy tożsamości. Konfiguracja systemu objęła integrację z kontrolerem domeny Active Directory przez protokół LDAP, implementację dwóch aplikacji SaaS (Portainer i Nextcloud) oraz konfigurację protokołów uwierzytelniania OAuth2 i SAML 2.0.

Zastosowanie platformy Authentik umożliwiło centralne zarządzanie tożsamością użytkowników oraz ich uprawnieniami w różnych aplikacjach. Integracja Portainera z wykorzystaniem OAuth2 pozwoliła na automatyczne tworzenie kont użytkowników przy pierwszym logowaniu, co usprawniło proces zarządzania dostępem i onboardingiem nowych użytkowników. W przypadku Nextcloud wdrożenie protokołu SAML zapewniło bezpieczne logowanie jednokrotne z pełnym mapowaniem atrybutów użytkowników i synchronizacją grup.

Warto podkreślić, że wszystkie komponenty zostały osadzone w kontenerach Docker, co zwiększyło elastyczność wdrożenia i uprościło zarządzanie infrastrukturą. Platforma Authentik okazała się niezwykle elastycznym rozwiązaniem, oferującym szerokie możliwości personalizacji – od modyfikacji interfejsu graficznego po dostosowywanie przepływów uwierzytelniania zgodnie z indywidualnymi wymaganiami organizacji.

Projekt napotkał jednak na kilka ograniczeń technicznych. Nie udało się zrealizować integracji z wykorzystaniem protokołu SCIM (System for Cross-domain Identity Management). Pomimo planów automatycznej synchronizacji użytkowników i ich atrybutów do aplikacji SaaS za pomocą SCIM, funkcjonalność ta okazała się niedostępna w wersji open-source. Usługa SCIM jest dostępna wyłącznie w płatnych wersjach Enterprise, co ograniczyło zakres automatyzacji w obszarze zarządzania tożsamościami.

Dodatkowo, nie udało się zaimplementować szyfrowanego połączenia LDAPS między systemem Authentik a kontrolerem domeny Active Directory. Aby uzyskać w aplikacji Authentik prawidłowo podpisany certyfikat do połączenia szyfrowanego, konieczne byłoby uruchomienie osobnego serwera certyfikującego (Certificate Authority) dla maszyny z aplikacją Authentik. Implementacja tej funkcjonalności wymagałaby dodatkowego czasu na konfigurację infrastruktury PKI (Public Key Infrastructure), na który zabrakło zasobów w ramach bieżącego projektu. W rezultacie komunikacja LDAP została zrealizowana w formie nieszyfrowanej, co w środowisku produkcyjnym stanowiłoby potencjalne zagrożenie bezpieczeństwa.

Mimo wspomnianych ograniczeń projekt należy uznać za w pełni udany – zaprojektowane i wdrożone rozwiązanie spełnia założone cele, zwiększa poziom bezpieczeństwa i centralizuje zarządzanie dostępem w środowisku chmurowym. Uzyskane doświadczenia wskazują na kierunki dalszego rozwoju systemu, w tym implementację

szyfrowanej komunikacji LDAPS oraz rozważenie migracji na wersję Enterprise platformy Authentik w celu wykorzystania zaawansowanych funkcji automatyzacji zarządzania tożsamościami.

7 Bibliografia

1) Microsoft Azure Documentation - Virtual Machines
2) Authentik Documentation - Identity Provider Configuration
3) Docker Documentation - Container Deployment Best Practices
4) nginx Documentation - Reverse Proxy Configuration
5) Microsoft Active Directory - Domain Services Implementation Guide
6) OAuth 2.0 Security Best Current Practice
7) SAML 2.0 Technical Overview