0 - 5 F - T	0 6 9 9	W. O. C.	0
Configuration Tag	Configuration Name	New Config [identity]	Special Comments
JDBCPersistenceManager.DataSource	Name	data_source="jdbc/WSO2IdentityDB"	
		[data source]	
JDBCPersistenceManager	SkipDBSchemaCreation	skip_db_schema_creation=false	
JDBCPersistenceManager.SessionDataPersist	Enable	[session_data.persistence] enable_persistance = true	
		[session_data.persistence]	
JDBCPersistenceManager.SessionDataPersist	Temporary	persist_temporary_data = true	
JDBCPersistenceManager.SessionDataPersist	PoolSize	[session_data.persistence] persistance_pool_size = true	
		[session data.cleanup]	
JDBCPersistenceManager.SessionDataPersist.SessionDataCleanUp	Enable	enable_expired_data_cleanup = true [session_data.cleanup]	
JDBCPersistenceManager.SessionDataPersist.SessionDataCleanUp	CleanUpTimeout	expire_session_data_after = true	
JDBCPersistenceManager.SessionDataPersist.SessionDataCleanUp	DeleteChunkSize	[session_data.cleanup] clean_expired_session_data_in_chunks_of = true	
JDBCPersistencemanager.SessionDataPersist.SessionDataCleanUp	DeleteChunkSize	[session_data_cleanun]	
JDBCPersistenceManager.SessionDataPersist.OperationDataCleanUp	Enable	[session_data_cleanup] clean_logged_out_sessions_at_immediate_cycle= true	
JDBCPersistenceManager.SessionDataPersist.TempDataCleanup	Enable	[session_data.cleanup] enable_pre_session_data_cleanup= true	
		[session_data.cleanup]	
JDBCPersistenceManager.SessionDataPersist.TempDataCleanup	PoolSize	[session_data.cleanup] pre_session_data_cleanup_thread_pool_size= true	
JDBCPersistenceManager.SessionDataPersist.TempDataCleanup	CleanUpTimeout	[session_data.cleanup] expire_pre_session_data_after= true	
abbot transcribental ager. Scanton batar transcribental occurring	oreanop rancour	copie_pre_session_data_arter rac	
		[session.timeout]	
TimeConfig	SessionIdleTimeout	idle_session_timeout= true	
TimeConfig	RememberMeTimeout	[session.timeout] remember_me_session_timeout= true	
		[key_mgt] keystore_dir= true	
Security	KeyStoresDir	Keystore_air= true	
	KeyManagerType	[key_mgt] key_manager_type= true	
		[key_mgt] trust_manager_type= true	
	TrustManagerType	trust_manager_type= true	
		[identity]	
Identity	IssuerPolicy	[identity] issuer_policy= true	
	TokenValidationPolicy	[identity] token_validation_policy= true	
OAuth.TokenCleanup	EnableTokenCleanup	[oauth.token_cleanup]	
OAuth. TokenCleanup	Enable l'okenCleanup	enable= true	
OAuth.TokenCleanup	RetainOldAccessToken	[oauth.token_cleanup] retain_access_tokens_for_auditing= true	
OAuth	IdentityOAuthTokenGenerator	[oauth.token_generation] access_token_type= "self_contained"	if access_token_type="self_contained" then IdentityOAuthTokenGenerator = "org.wso2 carbon.identity.oauth2 token.JWTTokenIssuer" if access_token_type not defined default will be "org.wso2 carbon.identity.oauth2.token.OauthTokenIssuerImpl"
OAuth	Identity OAuth Token Generator	access_token_type=_sen_contained	if access_token_value_type= "uuid" then AccessTokenValueGenerator = "org.apache.oltu.oauth2.as.issuer.UUIDValueGenerator"
			if access, loken value_type= "md5" then AccessTokenValueGenerator = "org.apache.oltu.oauth2.as.issuer.MD5Generator"
	AccessTokenValueGenerator	[oauth.token_generation] access_token_value_type= "uuid" #md5 #sha256	if access, bken_value_type="md5" then AccessTokenValueGenerator = "org apache oitu outhit2 as issuer.MD5Generator" if access_token_value_type=""ind256" then AccessTokenValueGenerator = "org wex2 carbon.identity outhit okenvaluegenerator.SHA256Generator" if access_token_value_type not defined default will be "org apache oitu.outhit2 as issuer.UIDIValueGenerator"
		[authentication]	if sign_auth_response_with_tenant_of= "sp" then UseSPTenantDomain = true
	UseSPTenantDomain	[authentication] sign_auth_response_with_tenant_of= "sp" #user	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain = true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	UseSPTenantDomain OAuth1RequestTokenUrl	[authentication] sign_auth_response_with_tenant_of="sp" #user [oauth_endpoints] oauth_request_token_url="	if sign_auth_response_with_tenant_of="sg" then UseSPTenantDomain = true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	OAuth1RequestTokenUrl	[authentication] sign_auth_response_with_tenant_of="sp" #user [oauth.endpoints] oauth! request_token_url="" [nauth.endpoints]	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain= true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain= false
		[authenication] sign_auth_response_with_lenant_of="sp" #user [sauth_endpoints] cauth_request_token_url="" [sauth_endpoints] cauth_reductse_url=""	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain= true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	OAuth1RequestTokenUrl	[authentication] sign_auth_response_with_tenant_of="sp" #user [aauth_endpoints] cauth_request_token_url="" [aauth_endpoints] cauth_reductse_url="" [aauth_endpoints] cauth_reductse_url=""	if sign_auth_response_with_tenant_of="sg" then UseSPTenantDomain = true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	OAuth AuthorizeUri OAuth AuthorizeUri OAuth AuthorizeUri	[authentication] sign_auth_response_with_tenant_of="sp" #user [aauth_endpoints] cauth_request_token_url="" [aauth_endpoints] cauth_reductse_url="" [aauth_endpoints] cauth_reductse_url=""	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain = true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	OAuth1RequestTokenUrl OAuth1AuthorizeUrl	[authentication] sign_auth_response_with_lenant_of="sp" #user [oauth endpoints] oauth_request_token_url="" [oauth endpoints] oauth_authorize_url="" [oauth endpoints] oauth_authorize_url="" [oauth endpoints] oauth_authorize_url="" oauth_endpoints] oauth_authorize_url=""	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	OAuth AuthorizeUri OAuth AuthorizeUri OAuth AuthorizeUri	[authenication] sign_auth_response_with_lenant_of="sp" #user [oauth endpoints] oauth_request_token_url="" [oauth endpoints] oauth_lendpoints]	if sign_auth_response_with_tenant_of="sg" then UseSPTenantDomain = true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	OAuth Request TokenUri OAuth TAuthorizeUri OAuth TAuthorizeUri OAuth TAuthorizeUri OAuth TAuthorizeUri OAuth TAuthorizeUri OAuth TAuthorizeUri	[authentication] sign_auth_response_with_tenant_of="sp" #user [oauth_endpoints] cauth_request_token_url="" [oauth_endpoints] cauth_reductse_url="" [oauth_endpoints] cauth_reductse_url="" [oauth_endpoints] cauth_reductse_url="" [oauth_endpoints] cauth_reductse_url="" [oauth_endpoints] cauth_reductse_url="" [oauth_endpoints] cauth_reductse_url="" cauth_endpoints cauth_reductse_url="" cauth_reductse_u	if sign_auth_response_with_tenant_of="sg" then UseSPTenantDomain = true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = fatse
	OAuth Request TokenUri OAuth Authorizeturi OAuth Authorizeturi OAuth AuthoEPUri OAuth 2 TokenEPUri OAuth 2 TokenEPUri	[authenication] sign_auth_response_with_lenant_of="sp" #user [oauth endpoints] oauth_request_token_url="" [oauth endpoints] oauth_reducting_url="" [oauth endpoints] oauth_authorize_url="" [oauth endpoints] oauth_acoss_token_url="" [oauth endpoints] oauth_acoss_token_url="" oauth_authopoints oauthopoints oauth_authopoints oauthopoints oauth_authopoints oauthopoints oauthopoints oauthopoints oauthopoi	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain= true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	OAuth Request TokenUri OAuth TAuthorizeUri OAuth TAuthorizeUri OAuth TAuthorizeUri OAuth TAuthorizeUri OAuth TAuthorizeUri OAuth TAuthorizeUri	[authenication] sign_auth_response_with_lenant_of="sp" #user [oauth endpoints] oauth_request_token_url="" [oauth endpoints] oauth_authorize_url="" [oauth endpoints] oauth_authorize_url="" [oauth endpoints] oauth_acoss_token_url="" [oauth endpoints] oauth_auth_authorize_url="" [oauth endpoints] oauth_authorize_url=""	if sign_auth_response_with_tenant_of="sg" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = fatse
	OAuth Request TokenUri OAuth AuthorizeUri OAuth AuthorizeUri OAuth AuthorizeUri OAuth AuthorizeUri OAuth 2TokenEPUri OAuth 2RevokeEPUri OAuth 2IntrospectEPUri	[authentication] sign_auth_response_with_tenant_of="sp" #user [oauth_endpoints] cauth_endpoints] cauth	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain= true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	OAuth Request TokenUri OAuth Authorizeturi OAuth Authorizeturi OAuth Access TokenUri OAuth ZhokenEPUri OAuth ZhokenEPUri OAuth ZhokenEPUri OAuth ZhokenEPUri OAuth ZhokenEPUri	[authenication] sign_auth_response_with_lenant_of="sp" #user [aauth_endpoints] [if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain= true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	OAuth Request TokenUri OAuth AuthorizeUri OAuth AuthorizeUri OAuth AuthorizeUri OAuth AuthorizeUri OAuth 2TokenEPUri OAuth 2RevokeEPUri OAuth 2IntrospectEPUri		if sign_auth_response_with_tenant_of="sg" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = fatse
	OAuth Request TokenUri OAuth AuthorizeUri OAuth AuthorizeUri OAuth Access TokenUri OAuth ZAuthzEPUri OAuth ZRevokeEPUri OAuth ZhevokeEPUri OAuth ZhevokeEPUri OAuth 2UserinfoEPUri OIDCCheck SessionEPUri	[authentication] sign_auth_response_with_lenant_of="sp" #user [auth endpoints] authentication	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain= true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	OAuth Request TokenUrl OAuth AuthorizetUrl OAuth AuthorizetUrl OAuth Access TokenUrl OAuth ZAuthzEPUrl OAuth ZRevokeEPUrl OAuth ZhevokeEPUrl OAuth 2IntrospectEPUrl OAuth 2UserIntoEPUrl OIDCCheck SessionEPUrl OIDCCheck SessionEPUrl	[authentication] sign_auth_response_with_tenant_of="sp" #user [oauth_endpoints]	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain= true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	OAuth Request TokenUri OAuth AuthorizeUri OAuth AuthorizeUri OAuth Access TokenUri OAuth ZAuthzEPUri OAuth ZRevokeEPUri OAuth ZhevokeEPUri OAuth ZhevokeEPUri OAuth 2UserinfoEPUri OIDCCheck SessionEPUri	[authentication] sign_auth_response_with_lenant_of="sp" #user [oauth_endponis] cauth_request_token_uri=" [oauth_endponis] cauth_reducts_curi=" [if sign_auth_response_with_tenant_of="sg" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	OAuth Request TokenUrl OAuth AuthorizetUrl OAuth AuthorizetUrl OAuth Access TokenUrl OAuth ZAuthzEPUrl OAuth ZRevokeEPUrl OAuth ZhevokeEPUrl OAuth 2IntrospectEPUrl OAuth 2UserIntoEPUrl OIDCCheck SessionEPUrl OIDCCheck SessionEPUrl	[authentication] sign_auth_response_with_lenant_of="sp" #user [oauth_endponis] cauth_request_token_uri=" [oauth_endponis] cauth_reducts_curi=" [if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain=false
	OAuth Request TokenUrl OAuth RuthorizeUrl OAuth RuthorizeUrl OAuth Access TokenUrl OAuth ZhokenEPUrl OAuth ZhokenEPUrl OAuth ZhokenEPUrl OAuth ZhokenEPUrl OAuth ZhokenEPUrl OAuth ZhokenEPUrl OIDCCReck SessionEPUrl OIDCCReck SessionEPUrl OIDCCReck SessionEPUrl OIDCLogoutEPUrl OAuth ZhokenPurl		if sign_auth_response_with_tenant_of="sg" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain=false
	OAuth Request TokenUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth Access TokenUrl OAuth ZAuthzEPUrl OAuth ZAvenkEPUrl OAuth ZrokenEPUrl OAuth Zinter Section EPUrl OIDCCReck Session EPUrl OIDCCReck Session EPUrl OIDCCLogoutEPUrl OAuth 2 ConsentPage		if sign_auth_response_with_tenant_of="sg" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	OAuth Request TokenUrl OAuth RuthorizeUrl OAuth RuthorizeUrl OAuth Access TokenUrl OAuth ZhokenEPUrl OAuth ZhokenEPUrl OAuth ZhokenEPUrl OAuth ZhokenEPUrl OAuth ZhokenEPUrl OAuth ZhokenEPUrl OIDCCReck SessionEPUrl OIDCCReck SessionEPUrl OIDCCReck SessionEPUrl OIDCLogoutEPUrl OAuth ZhokenPurl	Fauthentication sign_adh_response_with_lenant_of="sp" #user Fauthentication sign_adh_response_with_lenant_of="sp" #user Fauthentication	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain= true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	OAuth Request TokenUrl OAuth RuthorizeUrl OAuth RuthorizeUrl OAuth Rocess TokenUrl OAuth ZhuthzEPUrl OAuth ZhoenEPUrl OAuth ZherokeEPUrl OAuth ZherokeEPUrl OAuth ZherokeEPUrl OAUth ZherokeEPUrl OIDCCReck SessionEPUrl OIDCCReck SessionEPUrl OIDCCReck SessionEPUrl OAUth ZherorPage OIDCConsentPage	[authentication] sign_auth_response_with_tenant_of="sp" #user [oauth_endponits] oauth_request_token_url=" [oauth_endponits] oauth_subcen_url=" [oauth_endponits] oauth_subcen_url=" oauth_subcenurl=	if sign_auth_response_with_tenant_of="sg" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain=false
	OAuth Request TokenUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth Access TokenUrl OAuth ZhuthzEPUrl OAuth ZhokenEPUrl OAuth ZhevokeEPUrl OAuth ZhevokeEPUrl OAuth ZhevokeEPUrl OAUth ZhevokeEPUrl OIDCCReck SessionEPUrl OIDCCReck SessionEPUrl OIDCCReck SessionEPUrl OIDCLogoutEPUrl OAuth ZErnorPage OIDCConsentPage OIDCConsentPage	[authentication] sign_auth_response_with_lenant_of="sp" #user [oauth_endpoints]	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain= true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false
	OAuth Request TokenUrl OAuth RuthorizeUrl OAuth RuthorizeUrl OAuth Rocess TokenUrl OAuth ZhuthzEPUrl OAuth ZhoenEPUrl OAuth ZherokeEPUrl OAuth ZherokeEPUrl OAuth ZherokeEPUrl OAUth ZherokeEPUrl OIDCCReck SessionEPUrl OIDCCReck SessionEPUrl OIDCCReck SessionEPUrl OAUth ZherorPage OIDCConsentPage	suthentication sign_adh_response_with_tenant_of="sp" #user [oauth_endpoints] cauth_endpoints] cauth_endpoints cauth_endpoints	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain=false
	OAuth Request TokenUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth Access TokenUrl OAuth ZhuthzEPUrl OAuth ZhokenEPUrl OAuth ZhevokeEPUrl OAuth ZhevokeEPUrl OAuth ZhevokeEPUrl OAUth ZhevokeEPUrl OIDCCReck SessionEPUrl OIDCCReck SessionEPUrl OIDCCReck SessionEPUrl OIDCLogoutEPUrl OAuth ZErnorPage OIDCConsentPage OIDCConsentPage	suthentication sign_adh_response_with_tenant_of="sp" #user [oauth_endpoints] cauth_endpoints] cauth_endpoints cauth_endpoints	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain=false Second
	OAuth Request TokenUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthaEPUrl OAuth AevenbePUrl OAuth AevenbePUrl OAuth AevenbePUrl OAuth AevenbePUrl OAuth AevenbePUrl OAuth AevenbePUrl OIDCCheck SessionEPUrl OAuth 2ConsentPage OIDCCheck SessionEPUrl OAuth 2CONSENTPAge	jauthentication sign_auth_response_with_tenant_of="sp" #user [oauth_endpoints] cauth_endpoints] cauth	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain= true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain = false Second
	OAuth Request TokenUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthaEPUrl OAuth AevenbePUrl OAuth AevenbePUrl OAuth AevenbePUrl OAuth AevenbePUrl OAuth AevenbePUrl OAuth AevenbePUrl OIDCCneck SessionEPUrl OIDCCneck SessionEPUrl OIDCCassoniPuge OAUth ConseniPuge OAUth ConseniPuge OIDCCassoniPuge OIDCCassoniPuge OIDCCassoniPuge OIDCCassoniPuge	Suthentication	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain=false
	OAuth Request TokenUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthaEPUrl OAuth AevenbePUrl OAuth AevenbePUrl OAuth AevenbePUrl OAuth AevenbePUrl OAuth AevenbePUrl OAuth AevenbePUrl OIDCCheck SessionEPUrl OAuth 2ConsentPage OIDCCheck SessionEPUrl OAuth 2CONSENTPAge	jauthentication sign_auth_response_with_tenant_of="sp" #user [oauth_endpoints] cauth_endpoints] cauth	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain=false
	OAuth Request TokenUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeCEPUrl OAuth AuthorizeCEPUrl OAuth AuthorizeCEPUrl OIDCCReck Ses sionEPUrl OIDCCReck Ses sionEPUrl OIDCLogouIEPUrl OAuth 2ConsentPage OIDCCnsentPage OIDCCnsentPage OIDCCRepUrl OAuth 2ConsentPage	jauthentication sign_adh_response_with_lenant_of="sp" #user [oauth_endpoints] cauth_endpoints cauth_	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain=false
	OAuth Request TokenUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizePurl OAuth DesembePurl OAuth DesembePurl OAuth DesembePurl OIDCCheck Ses sionEPUrl OIDCCheck Ses sionEPUrl OIDCConsentPage	jauthentication sign_adh_response_with_tenant_of="sp" #user [oauth_endpoints] cauth_endpoints caut	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain=false
	OAuth Request TokenUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeCEPUrl OAuth AuthorizeCEPUrl OAuth AuthorizeCEPUrl OIDCCReck Ses sionEPUrl OIDCCReck Ses sionEPUrl OIDCLogouIEPUrl OAuth 2ConsentPage OIDCCnsentPage OIDCCnsentPage OIDCCRepUrl OAuth 2ConsentPage	[authentication] sign_auth_response_with_lenant_of="sp" #user [oauth_endpoints] oauth_stackses_pken_url=" oauth_scoss_pken_url=" oauth_scoss_pken_url= oauth_scoss_p	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain=false Second
	OAuth Request TokenUrl OAuth AuthorizeUrl OAuth DavenbeEPUrl OAuth DavenbeEPUrl OIDCCRes	jauthentication sign_adh_response_with_lenant_of="sp" #user [oauth_endpoints] cauth_endpoints caut	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain=false
	OAuth Request TokenUrl OAuth AuthorizetUrl OAuth AuthorizetUrl OAuth AuthorizetUrl OAuth AuthorizetUrl OAuth AuthorizetUrl OAuth AuthorizetUrl OAuth AuthorizetPurl OAuth AuthorizetPurl OIDCCRes SessionEPUrl OAuth AuthorizetBurl OIDCCRes SessionEPUrl OAuthorizetBurl OIDCCRes SessionEPUrl OAuthorizetBurl OAuthori	suthentication sign_adh_response_with_tenant_of="sp" #user [oauth_endpoints] cauth_endpoints] cauth	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain=true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain=false
	OAuth Request TokenUrl OAuth AuthorizeUrl OAuth DavenbeEPUrl OAuth DavenbeEPUrl OIDCCRes	Suthentication Sign_adh_response_with_lenant_of="sp" #user South endpoints	if sign_auth_response_with_tenant_of="spf then UseSPTenantDomain= true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain= false
	OAuth Request TokenUrl OAuth AuthorizetUrl OAuth AuthorizetUrl OAuth AuthorizetUrl OAuth AuthorizetUrl OAuth AuthorizetUrl OAuth AuthorizetUrl OAuth AuthorizetPurl OAuth AuthorizetPurl OIDCCRes SessionEPUrl OAuth AuthorizetBurl OIDCCRes SessionEPUrl OAuthorizetBurl OIDCCRes SessionEPUrl OAuthorizetBurl OAuthori	suthentication sign_adh_response_with_tenant_of="sp" #user [oauth_endpoints] cauth_endpoints] cauth	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain= true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain= false
	OAuth Request TokenUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizeUrl OAuth AuthorizePurl OAuth ZhevokeEPUrl OAuth ZhevokeEPUrl OIDCCheck Ses slonEPUrl OAuth Zhurk Spage OIDCCheck Ses slonEPUrl UseEntityldAs sloverin Oidchis covery AuthorizationCodeDefault/slidityPeriod User Access TokenDefault/slidityPeriod	jauthentication sign_adh_response_with_tenant_of="sp" #user [oauth_endpoints] cauth_endpoints] cauth_endpoints cauth_endpoints cauth_endpoints cauth_endpoints cauth_endpoints cauth_endpoints cauth_endpoints cauth_endpoints cauth_endpoints cauth_endpoints	if sign_auth_response_with_tenant_of="sp" then UseSPTenantDomain= true if sign_auth_response_with_tenant_of="user" then UseSPTenantDomain= false

		[oauth.token_renewal]	
	RenewRefreshTokenForRefreshGrant	renew_refresh_token= ***	
	TokenPersistenceProcessor	[oauth] hash_tokens_and_secrets= true #false	<module ref="addressing"/ >
	HashAlgorithm	[oauth] hash_token_algorithm= ""	
	EnableClientSecretHash	[oauth] hash_tokens_and_secrets= true #false	
	MapFederatedUsersToLocal	[oauth]	
	maprederatedUsers i oLocal	map_federated_users_to_local= true #false	
	RedirectToRequestedRedirectUri	[oauth] oauth.redirect_to_idp_error_page_on_error = "true"	
	redicerrorequesteredictor	[oauth.reponse_type.token] # Advanced	
OAuth.SupportedResponseTypes.SupportedResponseTypes	token	enable = true class = "org.wso2.carbon.identity.oauth2.authz.handlers.AccessTokenResponseTypeHandler"	
		[oauth.reponse_type.code] # Advanced enable = true	
	code	class = "org.wso2.carbon.identity.oauth2.authz.handlers.CodeResponseTypeHandler" [nauth_reponse_type_id_Inken] # Advanced	
	id token	[oauth.reponse_type.id_token] # Advanced enable = true class = "org.wso2.carbon.identity.oauth2.authz.handlers.IDTokenResponseTypeHandler"	
		[oauth.reponse_type.id_token_token] # Advanced enable = true	
	id_token token	class = "org.wso2.carbon.identity.oauth2.authz.handlers.IDTokenTokenResponseTypeHandler"	
		[[oauth.custom_response_type]] # Advanced name = "too_response_type" class = "com example.oauth.CustomResponseType"	
	custom response type	foauth grant, type authorization, codel	
OAuth.SupportedGrantTypes.SupportedGrantType	authorization_code	enable = "true" grant_handler = "org.wso2.carbon.identity.oauth2.token.handlers.grant_AuthorizationCodeGrantHandler"	
олиплогруппенний уреалирую пенений уре		[nouth grant type password]	
	password	enable = true grant_handler = "org.wso2.carbon.identity.oauth2.token.handlers.grant.PasswordGrantHandler"	
		[oauth.grant_type.refresh_token] enable = true grant_handler = "org.wso2.carbon.identity.oauth2.token.handlers.grant.RefreshGrantHandler"	
	refresh_token	grant_handler = "org.wso2.carbon.identity.oauth2.token.handlers.grant.RefreshGrantHandler" [oauth.grant_type.client_credentials]	
		anable = true	
	client_credentials	eracie - uder org. wso2.carbon.identity.oauth2.token.handlers.grant.ClientCredentialsGrantHandler* grant_handler = "org.wso2.carbon.identity.oauth2.token.handlers.grant.ClientCredentialsGrantHandler* allow_id_token = false	
		[oauth.grant_type.saml_bearer]	
	saml_bearer	enable = true grant_handler = "org.wso2.carbon.identity.oauth2.token.handlers.grant.saml.SAML2BearerGrantHandler"	
		[oauth.grant_type.iwa_ntlm] enable = true	
	iwa:ntlm	enable = true grant_handler = 'org.wso2 carbon.identity.oauth2.token.handlers.grant.iwa.ntlm.NTLMAuthenticationGrantH [oauth.grant_type.jwt_bearer]	andler"
		enable = true	
	jwt_bearer	enable = true grant_handler = "org.wso2.carbon.identity.oauth2.grant.jwt.JWTBearerGrantHandler" grant_validator = "org.wso2.carbon.identity.oauth2.grant.jwt.JWTGrantValidator"	
		[oauth.grant_type.uma_ticket] enable = true	
	uma_ticket.	enable = true grant handler = "org.wso2.carbon.identity.oauth2.token.handlers.grant.AuthorizationCodeGrantHandler" grant_validator = ""	
		[[nauth custom grant type]]	
		grant handler = "full qualified class name of grant handler" grant handler = "full qualified class name of grant handler" grant haidlator = "validator"	
	custom grant type	[oauth.custom_grant_type.properties] IdTokenAllowed = true	
OAuth	OAuthCallbackHandlers	[oauth.extensions] callback_handlers = ["org.wso2.carbon.identity.oauth.callback.DefaultCallbackHandler"]	
OABBI	Oxamcamaters	[oauth.token_validator.bearer]	
OAuth.TokenValidators	bearer	enable = true class = "org.wso2.carbon.identity.oauth2.validators.DefaultOAuth2TokenValidator"	
		[oauth.token_validator.jwt] enable = true	
OAuth.TokenValidators	jwt	class = "org.wso2.carbon.identity.oauth2.validators.OAuth2JWTTokenValidator"	
QAuth.TokenValidators	custom token validator	[[oauth.custom_token_validator]] type = "foo" class = "com.example.identity.TokenValidator"	
OAUU. I OKENVAIIGATOI'S	Custom token Validator	[oauth.scope_validator.jdbc]	
OAuth.ScopeValidators	jdbe	enable = true class = "org.wso2.carbon.identity.oauth2.validators.JDBCScopeValidator"	
		[oauth.scope_validator.xacml]	
OAuth.ScopeValidators	xacml	class = "org.wso2.carbon.identity.oauth2.validators.xacml.XACMLScopeValidator"	
OAuth.ScopeValidators	custom scope validator	[oauth.custom_scope_validator] class = "org.wso2.carbon.identity.oauth2.validators.xacml.XACMLScopeValidator"	
OAuth.EnableAssertions	UserName	[oauth.token_generation] include_username_in_access_token= true	
QAuth	BuildSubjectIdentifierFromSPConfia	[oauth] validation_response_subject_identifier_format= "fqn"#"app_configured" [oauth token validation]	if validation_response_subject_identifier_format= fqn then BuildSubjectIdentifierFromSPConfig= false if validation_response_subject_identifier_format= app_configured then BuildSubjectIdentifierFromSPConfig = true
QAuth AuthorizationContextTokenGeneration	Enabled Final Control of the Control	[cauth token validation]	
UAutn.AutnonzationContextTokenGeneration		include_validation_context_as_jwt_in_reponse= [oauth_extensions] token_context_generator=	
	TokenGeneratorImplClass	[oauth.extensions]	
	ClaimsRetrieverImplClass	token context claim retriever=	
	ConsumerDialectURI	[oauth.extensions] token_context_dialect_uri=	
	SignatureAlgorithm	[oauth.token_validation] validation_response_signing_algorithm=	
	AuthorizationContextTTL	[oauth.token.validation] validation_response_jwt_validity=	
OAuth CAM 2Creet		Validation [-espoint]" [Gauth_grant_type saml_bearer] [user_type="FEDERATED"	
OAuth.SAML2Grant	UserType	[oauth.oidc.claims]	
OAuth.OpenIDConnect	ConvertOriginalClaimsFromAssertionsToOIDCDialect	enable_oidc_dialect=	
	AddUnmappedUserAttributes	[oauth.oidc.claims] enable_ummapped_user_attributes=	
	IDTokenBuilder	[oauth.oidc.extensions] id_token_builder=	
	SignatureAlgorithm	[oauth.oidc.id_token] signature_algorithm=	
	IDTokenEncryptionAlgorithm	[oauth oids id, token]	First item from the list supported_encryption_algorithms will be configured as IDTokenEncryptionAlgorithm

		[an. th. aids id delice]	
	IDTokenEncryptionMethod	[oauth.oidc.id_token] supported_encryption_methods=[]	First item from the list supported_encryption_methods will be configured as IDTokenEncryptionMethod
	SupportedIDTokenEncryptionAlgorithms	[oauth.oidc.id_token] supported_encryption_algorithms=[]	
		[gauth.gidc.id_token]	
	SupportedIDTokenEncryptionMethods	supported_encryption_methods=[] [oauth.oidc.id_token]	
	Audiences	audiences=[] [oauth.oidc.id_token]	
	IDTokenIssuerID	issuer=	
	IDTokenCustomClaimsCallBackHandler	[oauth.oidc.extensions] claim_callback_handler=	
	IDTokenExpiration	[oauth.oidc.token validation]	
		id token_validity= [oauth.oidc.user_info] ynt_signature_algorithm=	
	UserInfoJWTSignatureAlgorithm	jwt_signature_algorithm= [oauth.oidc.extensions]	
	UserInfoEndpointClaimRetriever	user info claim retriever=	
	UserInfoEndpointRequestValidator	[oauth.oidc.extensions] user_info_request_validator=	
		[nauth oidc extensions]	
	UserInfoEndpointAccessTokenValidator	user_info_access_token_validator= [oauth.oidc]	if user_info.response_type= json then UserInfoEndpointResponseBuilder= "org.wso2.carbon.identity.oauth.endpoint.user.impl.UserInfoJSONResponseBuilder" if user_info.response_type= jwt then UserInfoEndpointResponseBuilder = "org.wso2.carbon.identity.oauth.endpoint.user.impl.UserInfoJWTResponse"
	UserInfoEndpointResponseBuilder	user_info.response_type= "json" #"jwt"	if user_info.response_type= jwt then UserInfoEndpointResponseBuilder = "org.wso2.carbon.identity.oauth.endpoint.user.impl.UserInfoJWTResponse" if consent promot= true then SkipUserConsent= false
	SkipUserConsent	[oauth] consent_prompt= true	if consent_prompt= false then SkipUserConsent = true
	SignJWTWithSPKey	[authentication] sign_auth_response_with_tenant_of= "sp" #user	if sign_auth_response_with_tenant_of= "sp" then Sign_JWTWithSPKey = true if sign_auth_response_with_tenant_of= "user" then Sign_JWTWithSPKey = false
		[oauth.oidc.token_validation] logout_token_validity=	
	LogoutTokenExpiration	[oauth.oidc.request object builder.request param value builder]	
	RequestObjectBuilder	enabled = true class = "org.wso2.carbon.identity.openidconnect.RequestParamRequestObjectBuilder"	
		[[oauth.oidc.custom_request_object_builder]]	
	RequestObjectBuilder custom	type = "foo" class = "com.example.oauth.CustomRequestBuilder"	
	RequestObjectValidator	[oauth.oidc.extensions.request_object_validator] request_object_validator =	
	requestosjectvandator	[Gauth.token_generation] retry_count_on_persistance_failures =	
OAuth.TokenPersistence	RetryCount	retry_count_on_persistance_failures = [oauth.token_renewal]	
OAuth	RenewTokenPerRequest	renew_access_token_per_request =	
		[sami]	
SSOService	EntityId	[saml] entity_id =	
	IdentityProviderURL	[saml.endpoints idp_url=	
		[saml.endpoints logout=	
	DefaultLogoutEndpoint	logouic [saml.endpoints notification=	
	NotificationEndpoint	notification= [saml.endpoints	
	ArtifactResolutionEndpoint	artifact_resolution=	
	SinaleLogoutRetryCount	[sami.slo] retry_attempts=	
		[saml.slo]	
	SingleLogoutRetryInterval	retry_interval= [saml.tenant_partitioning]	
	TenantPartitioningEnabled	enable=	
	AttributesClaimDialect	[saml] attribute_claim_dialect=	
	SAMLSSOAssertionBuilder	[saml.extensions] assertion_builder=	
	SAMLSSOEncrypter	[saml.extensions] encrypter=	
		[saml.extensions]	
	SAMLSSOSigner	signer=	
	SAML2HTTPRedirectSignatureValidator	[saml.extensions] redirect_signature_validator=	
	SAMLResponseValidityPeriod	[saml.response] validity=	
	UseAuthenticatedUserDomainCrypto	[saml] enable_user_domain_crpto=	
	OSEAGURIERUCATEGOUSEFDOMAINCTYPTO	enable_user_domain_crpto= [saml] signing_alg=	
	SAMLDefaultSigningAlgorithmURI	signing_alg= [sami]	
	SAMLDefaultDigestAlgorithmURI	[saml] digest_alg=	
	SAMLDefaultAssertionEncryptionAlgorithmURI	[saml] assertion_encryption_alg=	
	SAMLDefaultKeyEncryptionAlgorithmURI	[sami] key encryption alg=	
		[saml.slo]	
	SLOHostNameVerificationEnabled	host_name_verification= [saml.artifact]	
	SAML2ArtifactValidityPeriodInMinutes	validity=	
	SAMLECPEndpoint	[saml.endpoints]	
Consent	EnableSSOConsentManagement	[authentication.consent] prompt=	
SecurityTokenService	IdentityProviderURL	[sts.endpoint] idp=	
Passive STS	IdentityProviderURL	[passive_sts.endpoints] idp=	
		[passive_sts.endpoints] retry=	
	TokenStoreClassName	[passive_sts] token_store_class=	
	SLOHostNameVerificationEnabled	[passive_sts.slo] host_name_verification=	
EntitlementSettings.ThirftBasedEntitlementConfig	EnableThriftService	[entitlement.thrift] enable=	

		[entitlement.thrift]	
	ReceivePort	receiver_port=	
		[entitlement.thrift] client_timeout=	
	ClientTimeout	[entitlement.thrift.key_store]	
	KeyStore.Location	id=	
	KeyStore.Password	[entitlement.thrift.key_store] password=	
	KeyStore.Password	[entitlement.thrift]	
	ThriftHostName	hostname=	
SCIM	UserEPUrl	[scim.endpoints] users_endpoint=	
		[scim.endpoints]	
	GroupEPUrl	groups_endpoint=	
SCIM.SCIMAuthenticators	BasicAuthHandler	[scim.authentication_handler.basic.properties] priority = 5	
		[scim authentication, handler pauth properties]	
	OAuthHandler	priority = 10 authorization_server = "local://services"	
		[[scim authenticator]]	
		class = "com.example.identity.scim.Authenticator" priority = 10	
	Custom handler	[scim.custom authenticator.properties]	
		AuthorizationServer = "http://auth.example.com" [scim]	
	ComplexMultiValuedAttributeSupportEnabled	enable_complex_multivalued_attribute_support=	
SCIM2	UserEPUrl	[scim2.endpoints] users endpoint=	
		[scim2.endpoints]	
	GroupEPUrl	groups_endpoint=	
	ComplexMultiValuedAttributeSupportEnabled	[scim2] enable_complex_multivalued_attribute_support=	
Recovery.ReCaptcha	Password Fnable	[identity_mgt.password_reset_email]	
recovery.recaptona		enable_recaptcha= [identity_mot_username_recovery.email]	
	Username.Enable	[identity_mgt.username_recovery.email] enable_recaptcha=	
Recovery.Notification	Password.Enable	[identity_mgt.password_reset_email] enable_password_reset_email=	
necoral straincation	7 dosword.Enable	[identity_mgt_username_recovery.email] enable_username_recovery=	
	Username.Enable	enable_username_recovery=	
	InternallyManage	[identity_mgt] email_sender= "internal" #"external"	if email_sender= "internal" then InternallyManage= true if email_sender= "external" then InternallyManage = false
		[identity_mgt_password_reset_challenge_questions] enable_password_reset_challenge_questions =	
Recovery.Question	Password.Enable	enable_password_reset_challenge_questions =	
	Password.NotifyStart	[identity_mgt.password_reset_challenge_questions] notify_on_initiation =	
		[identity_mgt.password_reset_challenge_questions] question_separator =	
	Password.Separator	question_separator =	
	Password.MinAnswers	[identity_mgt.password_reset_challenge_questions] min_required_answers =	
		[identity_mgt.password_reset_challenge_questions] enable_recaptcha =	
	Password.ReCaptcha.Enable	enable_recaptcha = [identity_mat_password_reset_challenge_questions]	
	Password.ReCaptcha.MaxFailedAttempts	[identity_mgt.password_reset_challenge_questions] failures_before_recaptcha =	
	ExpiryTime	[identity_mgt.password_reset_email] reset_mail_validity =	
Recovery	ExpiryTime	[identity mot password reset email]	
	NotifySuccess	[identity_mgt.password_reset_email] notify_on_reset =	
Recovery.AdminPasswordReset	Offline	[identity_mgt.password_reset_by_admin] enable_offline_otp_based_reset =	
necovery. Admini assiroi aneset		[identity_mgt_password_reset_by_admin] enable_emailed_otp_based_reset =	
	OTP	enable_emailed_otp_based_reset =	
	RecoveryLink	[identity_mgt.password_reset_by_admin] enable_emailed_link_based_reset =	
EmailVerification	Enable	[identity_mgt.user_onboarding] enable_email_verification =	
Emanvernication	Enable	enable_email_vermcation = [identity_mot_user_onboarding]	
	ExpiryTime	[identity_mgt.user_onboarding] verification_email_validity =	
	LockOnCreation	[identity_mgt.user_onboarding] lock_on_creation}=	
	Location earth	[identity_mgt] email_sender= "internal" #"external"	if email sender= "internal" then InternallyManage= true
	Notification.InternallyManage	email_sender= "internal" #"external"	if email_sender="internal" then InternallyManage= true if email_sender= "external" then InternallyManage = false
	AskPassword.ExpiryTime	[identity_mgt.user_onboarding] ask_password_email_validity =	
		[identity_mgt_user_onloarding] password_generator =	
	AskPassword.PasswordGenerator	password_generator =	
		[identity_mgt.user_self_registration]	
SelfRegistration	Enable	allow_self_registration=	
	LockOnCreation	[identity_mgt.user_self_registration] lock_on_creation=	
	Lockoncreation	[identity mat]	if email_sender= "internal" then InternallyManage= true
	Notification.InternallyManage	[identity_mgt] email_sender= "internal" #"external"	if email_sender= "external" then InternallyManage = false
	ReCaptcha	[identity_mgt.user_self_registration] enable_recaptcha=	
	·		
	VerificationCode.ExpiryTime	[identity_mgt.user_self_registration] verification_email_validity=	
		[sp role management]	
SPRoleManagement	ReturnOnlyMappedLocalRoles	[sp_role_management] return_only_mapped_local_roles=	
Enable&ekPassword&dminIII		[identity_mgt.user_onboarding] ask_password_from_user=	
EnableAskraSSW0rdAdminUl		asv_bassword_itotii_nsgi=	
		[identity_mgt.endpoint]	
EnableRecoveryEndpoint		enable_recovery_endpoint=	
		Educition and audorial	
EnableSelfSignUpEndpoint		[identity_mgt.endpoint] enable_self_signup_endpoint=	

AuthenticationPolicy	CheckAccountExist	[authentication_policy] check_account_exist=	
JITProvisioning	UserNameProvisioningUI	[authentication,it_provisioning] username_provisioning_url= [authentication,it_provisioning]	
	PasswordProvisioningUI	[aumentication.jit_provisioning] password_provisioning_url=	
		[event.default_listener.workflow] priority=	
EventListeners	org.wso2.carbon.user.mgt.workflow.userstore.UserStoreActionList	fevent default, listener identity, moth	
	org.wso2.carbon.identity.mgLldentityMgtEventListener	priority= enable =	
	org.wso2.carbon.identity.scim.common.listener.SCIMUserOperatio	[event.default_listener.scim] priority= enable =	
		[event.default_listener.scim2] priority=	
	org.wso2.carbon.identity.scim2.common.listener.SCIMUserOperati	[event.default_listener.governance_identity_store]	
	org.wso2.carbon.identity.governance.listener.ldentityStoreEventLi	priority= enable =	
	org. was zerous or a serous se	[event.default_listener.governance_identity_mgt] priority= enable =	
	org.wso2.carbon.identity.governance.listener.ldentityMgtEventList	data_store =	
	org.wso2.carbon.identity.data.publisher.oauth.listener.OAuthToker	[event.default_listener.oauth_listener] priority= enable =	
	, , , , , , , , , , , , , , , , , , , ,	[event.default_listener.user_deletion] priority= enable =	
		[event default listener consent mot handler]	
	org.wso2.carbon.identity.application.authentication.framework.har.	priority= enable =	
	org.wso2.carbon.identity.application.authentication.framework.har.	[event.default_listener.jit_provisioning_handler] priority= enable =	
	org.wso2.carbon.identity.application.authentication.framework.har	[event default_listener post_auth_association_handler]	
	org.wso2.carbon.identity.application.authentication.framework.ham	enable = [event.default_listener.subject_identifier_handler] prioritiv=	
	org.wso2.carbon.identity.application.authentication.framework.ham	nioniy= enable = [event.default_listener.user_mgt_audit_logger]	
	org.wso2.carbon.user.mgt.listeners.UserMgtAuditLogger	priority= enable =	
	org.wso2.carbon.user.mgt.listeners.UserManagementAuditLogger	[event.default_listener.user_management_audit_logger] priority= enable =	
		[event.default_listener.user_failure_audit_logger]	
	org.wso2.carbon.user.mgt.listeners.UserMgtFailureAuditLogger	enable = [event.default_listener.user_claim_audit_logger]	
	org.wso2.carbon.user.mgt.listeners.UserClaimsAuditLogger	priority= enable =	
		[[event_listener]] id = "custom_audit_listener" type = "org_wso2_carbon.identity.core.handler.AbstractIdentityMessageHandler"	
		Li = Custum; addia_instanti type = "org wso2_carbon.identity.core.handler.AbstractidentityMessageHandler" name = "com.example.identity_AuditListener" order = 99	
	custom event listener	[event_listener.properties] file_name = "login_audit.log"	
		[event.default_recorder.user_delete_event] name=	
<userdeleteeventrecorders></userdeleteeventrecorders>	UserDeleteEventRecorder	enable = write_to_separate_csv.path =	
		[[event_recorder]] name = "com.example.identity.Recorder" order = 99	
	custom event recorder	order = 90 [event_recorder_properties] file_name = "login_audit.log"	
		[cache.framework_session_context_cache]	
CacheConfig	AppAuthFrameworkSessionContextCache	enable = timeout = capacity =	
		[cache_authentication_context_cache] enable = timeout =	
	AuthenticationContextCache	capacity =	
		[cache.authentication_request_cache] enable = timeout =	
	AuthenticationRequestCache	capacity = [cache authentication_result_cache] enable =	
	AuthenticationResultCache	enable = timeout = capacity =	
		[cache.app_info_cache] enable =	
	ApplnfoCache	timeout = capacity = (capacity	
		[cache.authorization_grant_cache] enable = timeout =	
	AuthorizationGrantCache	capacity = [cache jwks_cache] enable =	
	JWKSCache	timeout = capacity =	
		[cache.oauth_cache]	
	OAuthCache	timeout = capacity = (aspha page aspha)	
		[cache.oauth_scope_cache] enable = timeout =	
	OAuthScopeCache	capacity =	

		[cache.oauth_session_data_cache] enable = timeout =	
	OAuthSessionDataCache	capacity =	
		[cache.saml_sso_participant_cache] enable =	
	SAMLSSOParticipantCache	timeout = capacity =	
		[cache.saml_sso_session_index_cache] enable = timeout =	
	SAMLSSOSessionIndexCache	capacity =	
		[cache.saml_sso_session_data_cache] enable =	
	SAMLSSOSessionDataCache	timeout = capacity =	
		[cache.service_provider_cache]	
	ServiceProviderCache	timeout = capacity =	
		[cache provisioning connector cache]	
	ProvisioningConnectorCache	enable = timeout = capacity =	
	•	[cache.provisioning_entity_cache]	
	ProvisioningEntityCache	enable = timeout = capacity =	
	,	[cache service provider provisioning connector cache]	
	ServiceProviderProvisioningConnectorCache	enable = timeout = capacity =	
	Service-rovisioning-connector-cache		
		[cache.idp_cache_by_auth_property] enable = timeout =	
	IdPCacheByAuthProperty	capacity = [cache.idp, cache_by_hri] enable =	
		timeout =	
	IdPCacheByHRI	capacity = [cache.idp_cache_by_name]	
		enable = timeout =	
	IdPCacheByName	capacity =	
EnableFederatedUserAssociation		[user.association] enable_for_federated_users =	
EnablereoeratedUserAssociation			
<tenantcontextstorewrite></tenantcontextstorewrite>	WebApp	[tenant_context.rewrite] webapps= []	
	Servlet	[tenant_context.rewrite] servlets= []	
	Service		
ClockSkew		[server] clock_skew=	
		[oauth.jwks_endpoint]	
JWTValidatorConfigs	Enable	enable=	
	JWKSEndpoint.HTTPConnectionTimeout	[oauth.jwks_endpoint] connection_timeout=	
	JWKSEndpoint.HTTPReadTimeout	[oauth.jwks_endpoint] read_timeout=	
	JWKSEndpoint.HTTPSizeLimit	[oauth.jwks_endpoint] size_limit_bytes=	
	·		
		[authentication.adaptive]	
<adaptiveauth></adaptiveauth>	EventPublisher.ReceiverURL	[authentication.adaptive] event_publisher.url=	
<adaptiveauth></adaptiveauth>	EventPublisher.ReceiverURL	event_publisher.url=	if authentication_type="basic" then Enable-true
<4daprire-tatis			if authentication type= "basic" then Enable- true username- admin password - admin
<dd>daprire-tatic></dd>	EventPublisher.BasicAuthentication	[authentication adaptive event_publisher] authentication_type="basic" [authentication_type="basic"] [authentication_adaptive]	if authentication_type="basic" then Enable= true username= admin password = admin url = https://localhost.8280
<dd><ddaprine dath=""></ddaprine></dd>	EventPublisher BasicAuthentication AsyncSequenceExecutorPoolSize	[authentication_adaptive_event_publisher] authentication_type="basic" [authentication_adaptive] async_executer_pod_isze=""	nassword = admin
<dd>datagrire-tatile></dd>	EventPublisher.BasicAuthentication	[authentication_adaptive.event_publisher] suthentication_type="basic" [authentication_adaptive] sayte_execute_pool_size=" [authentication_adaptive] sayte_execute_pool_size=" [authentication_adaptive.http_connections] mass=""	nassword = admin
<dd><ddoprine datilo=""></ddoprine></dd>	EventPublisher BasicAuthentication AsyncSequenceExecutorPoolSize	[authentication adaptive event_oublisher] authentication_type="basic" [authentication_adaptive] sayre_execute_pod_size=" [authentication_adaptive_http_connectione] max=" [authentication_adaptive_http_connectione] max_e="basic_pod_size=" max_e="basic_pod_size=" [authentication_adaptive_http_connectione] max_e="basic_pod_size=" " " " " " " " " " " " " " " " " " "	nassword = admin
<dd><ddagarine.tatilo></ddagarine.tatilo></dd>	EventPublisher.BasicAuthentication AsyncSequenceExecutorPoolSize Max TotalConnections	[authentication adaptive event_oublisher] authentication, type="basic" [authentication adaptive] [authentication adaptive] [authentication adaptive] [authentication adaptive, http_connectione] [authentication adaptive, http_connectione] [authentication adaptive http.connectione] [authentication adaptive https://doi.org//doi.o	nassword = admin
<dd><ddoprine latility<="" td=""></ddoprine></dd>	EveniPublisher.BasicAuthentication AsyncSequenceExecutorPoolSize MaxTotalConnections MaxTotalConnections	[authentication_adaptive_event_publisher] authentication_type="basic" [authentication_adaptive] [authentication_adaptive] [authentication_adaptive_http_connections] max=" [authentication_adaptive_http_connections] max=" [authentication_adaptive_http_connections] max_per_route== [authentication_adaptive_http_connections] [authentication_adaptive_http_connections] [authentication_adaptive_http_connections] [authentication_adaptive_http_connections] [authentication_adaptive_http_connections] [authentication_adaptive_http_connections]	nassword = admin
<dd><ddoprine dath=""></ddoprine></dd>	EveniPublisher.BasicAuthentication AsyncSequenceExecutorPoolSize MaxTotalConnections Max TotalConnectionsPerRoute HTTPConnectionImmout	[authentication adaptive event_publisher] authentication type="basic" [authentication daptive] async_executer_pod_size=" [authentication adaptive http_connections] max=" [authentication adaptive http_connections] max_er_poute=" [authentication adaptive http_connections] max_per_poute=" [authentication adaptive http_connections] connection_timeout=" [authentication adaptive http_connections] connection_timeout=" [authentication_adaptive http_connections]	nassword = admin
<dd><ddoprine datilo=""></ddoprine></dd>	EventPublisher.BasicAuthentication AsyncSequenceExecutorPoolSize MaxTotalConnections MaxTotalConnectionSPurRoute HTTPConnectionTimeout HTTPConnectionTimeout HTTPConnectionRequestTimeout	[authentication_adaptive_event_publisher] authentication_type="basic" [authentication_adaptive] [authentication_adaptive] [authentication_adaptive] [authentication_adaptive_http_connections] max="" [authentication_adaptive_http_connections] max="" [authentication_adaptive_http_connections]	nassword = admin
<dd><ddoprine datilo=""></ddoprine></dd>	EventPublisher.BasicAuthentication AsyncSequenceExecutorPoolSize MaxTotalConnections MaxTotalConnectionsPurRoute HTTPConnectionTimeout HTTPCannectionRequestTimeout Refreshinterval	[authentication adaptive event_publisher] suthentication_type="basic" [authentication_adaptive] symc_execute_pod_isse=" [authentication_adaptive] symc_execute_pod_isse=" [authentication_adaptive http_connections] max.e=" [authentication_adaptive http_connections] max.per_poutse=" [authentication_adaptive http_connections] connection_timeout=" [authentication_adaptive_http_connections] request_timeout=" [authentication_adaptive_http_connections] request_tim	nassword = admin
< datapatric tatilo*	EventPublisher.BasicAuthentication AsyncSequenceExecutorPoolSize MaxTotalConnections MaxTotalConnectionsPurRoute HTTPConnectionTimeout HTTPConnectionTimeout HTTPConnectionRequestTimeout PromptOnt.ongWatt	[authentication adaptive.event_publisher] authentication_typer-basic* [authentication adaptive] async_executive_pool_sizer= [authentication_adaptive] the promote of the pr	nassword = admin
< datapatric tatilo*	EventPublisher.BasicAuthentication AsyncSequenceExecutorPoolSize MaxTotalConnections MaxTotalConnectionsPurRoute HTTPConnectionTimeout HTTPCannectionRequestTimeout Refreshinterval	[authentication adaptive event_publisher] suthentication_type="basic" [authentication_adaptive] symc_execute_pod_isse=" [authentication_adaptive] symc_execute_pod_isse=" [authentication_adaptive http_connections] max.e=" [authentication_adaptive http_connections] max.per_poutse=" [authentication_adaptive http_connections] connection_timeout=" [authentication_adaptive_http_connections] request_timeout=" [authentication_adaptive_http_connections] request_tim	nassword = admin
	EveniPublisher BasicAuthentication AsyncSequenceExecutorPoolSize MaxTotalConnections MaxTotalConnectionsPerRoute MITPConnectionTimeout HITPConnectionTimeout HITPConnectionRequestTimeout PromptonLongWait LongWaitTimeout LongWaitTimeout	[authentication adaptive event_publisher] suthentication_type="basic" [authentication_adaptive] saync_execute_pod_size=" [authentication_adaptive] saync_execute_pod_size=" [authentication_adaptive http_connections] max.e="cutestication_adaptive http_connections] [authentication_adaptive] [authentication	nassword = admin
<a hre<="" td=""><td>EventPublisher BasicAuthentication AsyncSequenceExecutorPoolSize Max TotalConnections Max TotalConnectionsPerRoute MITPConnectionTimeout MITPConnectionTimeout MITPConnectionRequestTimeout Refessionterval PramptOnLongWalt LongWaltTimeout IntermediateCerts.CertCN</td><td>[authentication_type="basic" [authentication_type="basic" [authentication_adaptive] [authentication_adaptive] [authentication_adaptive] [authentication_adaptive_type="connections"] [authentication_adap</td><td>nassword = admin</td>	EventPublisher BasicAuthentication AsyncSequenceExecutorPoolSize Max TotalConnections Max TotalConnectionsPerRoute MITPConnectionTimeout MITPConnectionTimeout MITPConnectionRequestTimeout Refessionterval PramptOnLongWalt LongWaltTimeout IntermediateCerts.CertCN	[authentication_type="basic" [authentication_type="basic" [authentication_adaptive] [authentication_adaptive] [authentication_adaptive] [authentication_adaptive_type="connections"] [authentication_adap	nassword = admin
	EveniPublisher BasicAuthentication AsyncSequenceExecutorPoolSize MaxTotalConnections MaxTotalConnectionsPerRoute MITPConnectionTimeout HITPConnectionTimeout HITPConnectionRequestTimeout PromptonLongWait LongWaitTimeout LongWaitTimeout	fauthentication adaptive event_publisher] suthentication_type="basic" suthentication_adaptive suthentication_adaptive supprocessors supprocessors [authentication_adaptive_http_connections] rava." [authentication_adaptive_http_connections] rava.per_routes=" [authentication_adaptive_http_connections] [authentication_adaptive_h	nassword = admin
	EventPublisher BasicAuthentication AsyncSequenceExecutorPoolSize Max TotalConnections Max TotalConnectionsPerRoute MITPConnectionTimeout MITPConnectionTimeout MITPConnectionRequestTimeout Refessionterval PramptOnLongWalt LongWaltTimeout IntermediateCerts.CertCN	[authentication adaptive event_publisher] authentication _type="basic" [authentication adaptive] [authentication adaptive] [authentication adaptive http_connections] maxe=" [authentication adaptive http_connections] max.per_puble=" [authentication adaptive http_connections] [authentication adaptive long_wait] [authent	nassword = admin
Informediate Cert/alidation	EventPublisher BasicAuthentication AsyncSequenceExecutorPoolSize Max TotalConnections Max TotalConnectionsPerRoute MITPConnectionTimeout MITPConnectionTimeout MITPConnectionRequestTimeout Refessionterval PramptOnLongWalt LongWaltTimeout IntermediateCerts.CertCN	[authentication adaptive event_publisher] suthentication, type="basic" [authentication adaptive] [authentication adaptive] [authentication adaptive http_connections] [authentication adaptive long_wail] [authentication adaptive long_wail] [authentication adaptive long_wail] [authentication adaptive long_wail] [fletermediate_cert_validation] [fletermediate_cert_val	nassword = admin
Informediate Cert/alidation	EventPublisher BasicAuthentication AsyncSequenceExecutorPoolSize Max TotalConnections Max TotalConnectionsPerRoute MITPConnectionTimeout MITPConnectionTimeout MITPConnectionRequestTimeout Refessionterval PramptOnLongWalt LongWaltTimeout IntermediateCerts.CertCN	[authentication adaptive event_publisher] authentication _type="basic" [authentication adaptive] [authentication adaptive] [authentication adaptive http_connections] maxe=" [authentication adaptive http_connections] max.per_puble=" [authentication adaptive http_connections] [authentication adaptive long_wait] [authent	nassword = admin
IntermediateCertValidation FederatedIDPRoleClaimValueAttributeSeparator	EveniPublisher.BasicAuthentication AsyncSequenceExecutorPoolSize MaxTotalConnections MaxTotalConnections MaxTotalConnectionTimeout HTTPConnectionTimeout HTTPConnectionRequestTimeout ForepronectionRequestTimeout Refreshinterval PromptOnLongWalt LongWaltTimeout IntermediateCerts.CertCN ExemptContext.Context	[authentication adaptive event_publisher] authentication _type="basic" [authentication adaptive] sayro_execute_pod_size=" [authentication adaptive http_connections] max=" [authentication adaptive http_connections] max_per_pouts=" [authentication adaptive http_connections] connection_timeout=" [authentication adaptive http_connections] connection_timeout=" [authentication adaptive http_connections] request_timeout=" [authentication adaptive http_connections] request_timeout=" [authentication adaptive long_wait] purper_gentication_adaptive long_wait] purper_gentication_adaptive long_wait] purper_gentication_adaptive long_wait] finetimediate_cert_validation] finetimediate_cert_validation] [finetimediate_cert_validation] [finetimediate_cert_validation] gentication_adaptive_cert_validation] cert_cons=[] [finetimediate_cert_validation] gent_cert_validation_cert_cert_validation] gent_cert_validation_cert_validation] cert_cons=[] [finetimediate_cert_validation] gent_cert_validation_cert_validation] gent_cert_validation_cert_vali	nassword = admin
IntermediateCertValidation FederatedIDPRoleClaimValueAttributeSeparator	EveniPublisher.BasicAuthentication AsyncSequenceExecutorPoolSize MaxTotalConnections MaxTotalConnections MaxTotalConnectionTimeout HTTPConnectionTimeout HTTPConnectionRequestTimeout ForepronectionRequestTimeout Refreshinterval PromptOnLongWalt LongWaltTimeout IntermediateCerts.CertCN ExemptContext.Context	[authentication adaptive event_publisher] suthentication, type="basic" [authentication adaptive] [authentication adaptive] [authentication adaptive http_connections] [authentication adaptive long_wail] [authentication adaptive long_wail] [authentication adaptive long_wail] [authentication adaptive long_wail] [fletermediate_cert_validation] [fletermediate_cert_val	nassword = admin
IntermediateCortValidation FederatedIDPRoleClaimValueAttributeSeparator X509	EventPublisher.BasicAuthentication AsyncSequenceExecutorPoolSize MaxTotalConnections MaxTotalConnectionsPurRoute HITPConnectionTimeout HITPConnectionTimeout HITPConnectionRequestTimeout Refreshinterval PromptOnt.ongWait LongWaitTimeout IntermediateCerts.CertCN ExemptContext Context X509RequestFleaderName	[authentication adaptive event_publisher] authentication _type="basic" [authentication adaptive] sayro_execute_pod_size=" [authentication adaptive http_connections] max=" [authentication adaptive http_connections] max_per_pouts=" [authentication adaptive http_connections] connection_timeout=" [authentication adaptive http_connections] connection_timeout=" [authentication adaptive http_connections] request_timeout=" [authentication adaptive http_connections] request_timeout=" [authentication adaptive long_wait] purper_gentication_adaptive long_wait] purper_gentication_adaptive long_wait] purper_gentication_adaptive long_wait] finetimediate_cert_validation] finetimediate_cert_validation] [finetimediate_cert_validation] [finetimediate_cert_validation] gentication_adaptive_cert_validation] cert_cons=[] [finetimediate_cert_validation] gent_cert_validation_cert_cert_validation] gent_cert_validation_cert_validation] cert_cons=[] [finetimediate_cert_validation] gent_cert_validation_cert_validation] gent_cert_validation_cert_vali	nassword = admin

EnableAudiences			
GrantTypeName			
		[resource_access_control]] context ="nest1" secured = rue intp_method = rain* permissions = [*p1**,*p2**]	
		secured = true http method = "all"	
	Custom ResourceAccessControl	permissions = ["p1", "p2"]	If a user need to add a custom ResourceAccessControl section