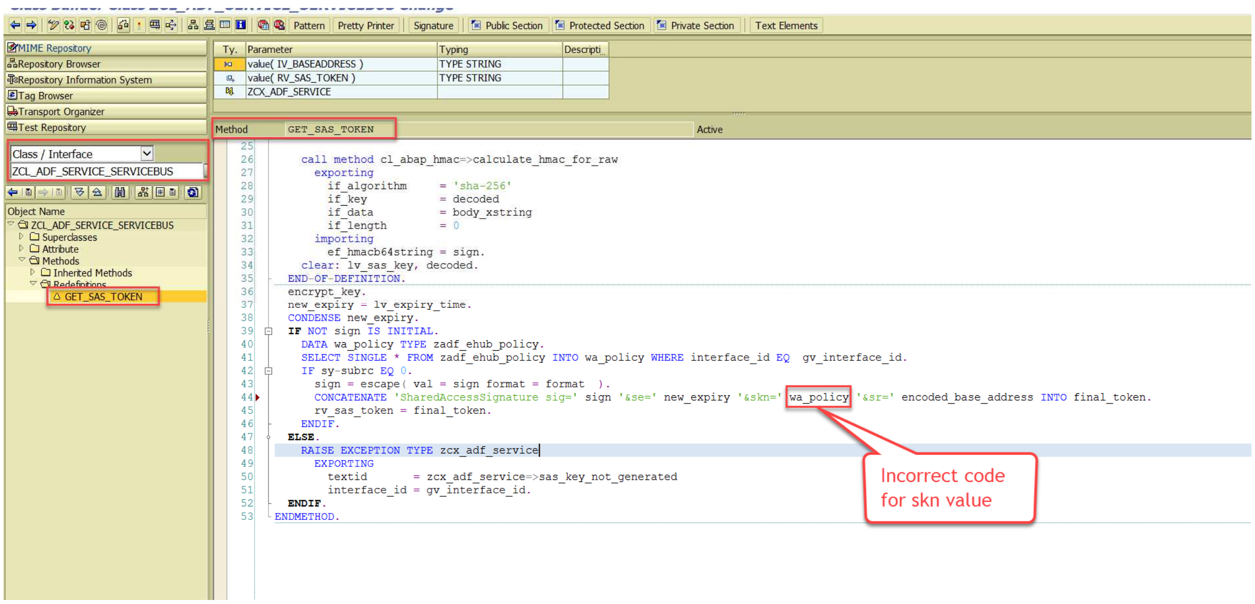
Before:



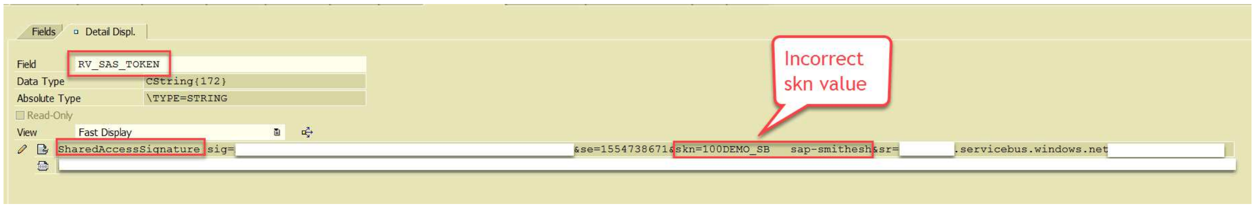
Class / Interface: ZCL_ADF_SERVICE_SERVICEBUS

Ty.	Parameter	Typing	Descripti
	value(IV_BASEADDRESS)	TYPE STRING	
	value(RV_SAS_TOKEN)	TYPE STRING	
	ZCX_ADF_SERVICE		

Method GET_SAS_TOKEN

```
    call method cl_abap_hmac=>calculate_hmac_for_raw
      exporting
        if_algorithm     = 'sha-256'
        if_key           = decoded
        if_data          = body_xstring
        if_length        = 0
      importing
        ef_hmacb64string = sign.
    clear: lv_sas_key, decoded.
  END-OF-DEFINITION.
  encrypt_key.
  new_expiry = lv_expiry_time.
  CONDENSE new_expiry.
  IF NOT sign IS INITIAL.
    DATA wa_policy TYPE zadf_ehub_policy.
    SELECT SINGLE * FROM zadf_ehub_policy INTO wa_policy WHERE interface_id EQ  gv_interface_id.
    IF sy-subrc EQ 0.
      sign = escape( val = sign format = format  ).
      CONCATENATE 'SharedAccessSignature sig=' sign '&se=' new_expiry '&skn=' wa_policy '&sr=' encoded_base_address INTO final_token.
      rv_sas_token = final_token.
    ENDIF.
  ELSE.
    RAISE EXCEPTION TYPE zcx_adf_service
      EXPORTING
        textid       = zcx_adf_service=>sas_key_not_generated
        interface_id = gv_interface_id.
  ENDIF.
ENDMETHOD.
```

Incorrect code for skn value



Field: RV_SAS_TOKEN

Data Type: CString{172}

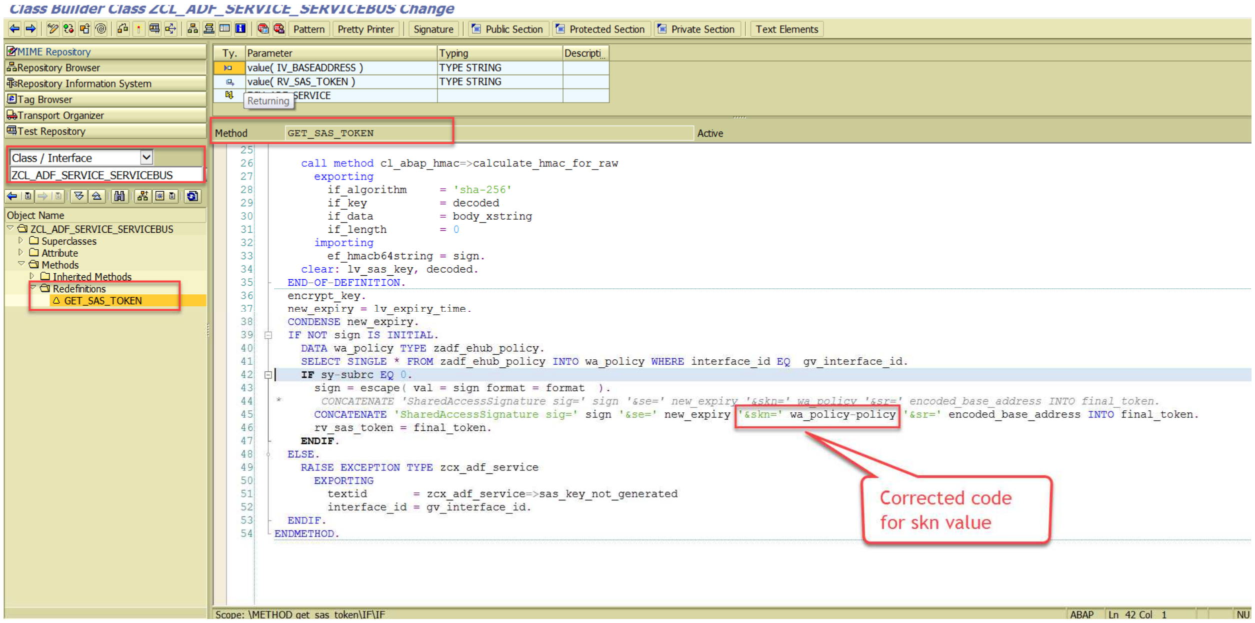
Absolute Type: \TYPE=STRING

□ Read-Only

```
SharedAccessSignature sig=[redacted]&se=1554738671&skn=100DEMO_SB    sap-smithesh&sr=[redacted].servicebus.windows.net[redacted]
```

Incorrect skn value

After:



Class / Interface: ZCL_ADF_SERVICE_SERVICEBUS

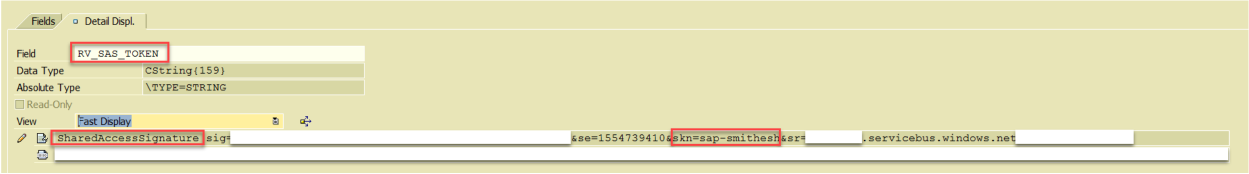
ZCL_ADF_SERVICE_SERVICEBUS > Methods > Redefinitions > GET_SAS_TOKEN

Ty.	Parameter	Typing	Descripti...
	value(IV_BASEADDRESS)	TYPE STRING	
	value(RV_SAS_TOKEN)	TYPE STRING	
	ZCX_ADF_SERVICE		

Method GET_SAS_TOKEN Active

```
    call method cl_abap_hmac=>calculate_hmac_for_raw
      exporting
        if_algorithm     = 'sha-256'
        if_key           = decoded
        if_data          = body_xstring
        if_length        = 0
      importing
        ef_hmacb64string = sign.
    clear: lv_sas_key, decoded.
  END-OF-DEFINITION.
  encrypt_key.
  new_expiry = lv_expiry_time.
  CONDENSE new_expiry.
  IF NOT sign IS INITIAL.
    DATA wa_policy TYPE zadf_ehub_policy.
    SELECT SINGLE * FROM zadf_ehub_policy INTO wa_policy WHERE interface_id EQ  gv_interface_id.
    IF sy-subrc EQ 0.
      sign = escape( val = sign format = format  ).
*      CONCATENATE 'SharedAccessSignature sig=' sign '&se=' new_expiry '&skn=' wa_policy '&sr=' encoded_base_address INTO final_token.
      CONCATENATE 'SharedAccessSignature sig=' sign '&se=' new_expiry '&skn=' wa_policy-policy '&sr=' encoded_base_address INTO final_token.
      rv_sas_token = final_token.
    ENDIF.
  ELSE.
    RAISE EXCEPTION TYPE zcx_adf_service
      EXPORTING
        textid       = zcx_adf_service=>sas_key_not_generated
        interface_id = gv_interface_id.
  ENDIF.
ENDMETHOD.
```

Scope: \METHOD get_sas_token\IF\IF   ABAP   Ln 42 Col 1   NU



Field: RV_SAS_TOKEN

Data Type: CString{159}

Absolute Type: \TYPE=STRING

□ Read-Only

View: Fast Display

```
SharedAccessSignature sig=[redacted]&se=1554739410&skn=sap-smithesh&sr=[redacted].servicebus.windows.net[redacted]
```