# AA Testing

The following shell script uses *curl* to query a SAML2 Attribute Authority.

You need a valid principal (eduPersonPrincipalName) and the X.509 credentials of an existing Service Provider to use this script.

# Source

```
#!/bin/bash

basedir=$(dirname $0)

# URL of the Attribute Authority
AA_URI="https://hexaa.eduid.hu:8443/simplesaml/module.php/aa/attributeserver.php"

# Testing principal (subject)
Principal="bajnokk@niif.hu"

# HEXAA cert
AACert="$basedir/keys/hexaa.eduid.hu-aa.crt"

# EntityID and credentials of the SP on behalf of which
# the request is made
ReqSP="https://sp.hexaa.eduid.hu/test"
ReqCert="$basedir/keys/test.sp.hexaa.eduid.hu-fed.crt"
ReqKey="$basedir/keys/test.sp.hexaa.eduid.hu-fed.key"


usage () {
        cat <<EOS
Usage: $0 [options]

Options:
  -a uri       Attribute Authority URI. Defaults to '$AA_URI'
```

```
  -C certfile  Attribute Authority metadata certificate in PEM format. Defaults to '$AACert'.
  -p principal Testing principal (user name / subject). Defaults to '$Principal'.
  -s entity    EntityID of the SP on behalf of which the request is made. Defaults to '$ReqSP'
  -k keyfile   Key file in PEM format containing the key of the SP used for the request.
Defaults to '$ReqKey'
  -c certfile  Cert file in PEM format containing the certificate of the SP used for the
request. Defaults to '$ReqCert'
EOS
        exit 3
}

# Get command line arguments
while getopts "a:p:s:k:c:h" opt; do
        case $opt in
                a)
                        AA_URI=$OPTARG
                        ;;
                C)
                        AACert=$OPTARG
                        ;;
                p)
                        Principal=$OPTARG
                        ;;
                s)
                        ReqSP=$OPTARG
                        ;;
                k)
                        ReqKey=$OPTARG
                        ;;
                c)
                        ReqCert=$OPTARG
                        ;;
                h)
                        usage
                        ;;
                \?)
                        usage
                        ;;
        esac
```

```
done

DATE=$(date --utc +%FT%TZ)
ReqID=$(hexdump -n 16 -e '4/4 "%08x" 1 "\n"' /dev/urandom)


read -r -d '' REQ_XML <<EOS
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <samlp:AttributeQuery xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" ID="_$ReqID" IssueInstant="$DATE"
Version="2.0">
      <saml:Issuer>$ReqSP</saml:Issuer>
      <saml:Subject>
        <saml:NameID Format="urn:oid:1.3.6.1.4.1.5923.1.1.1.6">$Principal</saml:NameID>
      </saml:Subject>
    </samlp:AttributeQuery>
  </S:Body>
</S:Envelope>
EOS

#debug echo "$REQ_XML"

echo "$REQ_XML" | \
  curl --silent --show-error --cacert $AACert --cert $ReqCert --key $ReqKey \
       --header "Content-Type: text/xml;charset=UTF-8" --data @- $AA_URI
```

# Validation of response

Signature validation:

```
xmlsec1 --verify --id-attr:ID "urn:oasis:names:tc:SAML:2.0:protocol:Response" --trusted-pem
$aacert $response 2>/dev/null
```

Content validation:

```
xmllint --xpath "//*['Attribute'](local-name()#bkmrk-)[@Name'$attribute']/*[local-
name()='AttributeValue']/text()" $response
```

Változat #1
cziernorbert hozta létre 2026-04-14 13:22:25 CEST
cziernorbert frissítette 2026-04-14 13:24:34 CEST