HDN DARTS

WEB AUTHENTICATIE

HDN Helpdesk – **T**: 0182 – 750 585 **F**: 0182 – 750 589 **M**: helpdesk@hdn.nl

CSNet

Inhoudsopgave

1. Inleiding op het ontwerp

1.1 – Het doel van dit document

Dit document heeft tot doel de implementie van het authenticeren/autoriseren met behulp van HDN certificaten te beschrijven.

1.2 – Doelgroep van dit document

De doelgroep bestaat uit ontwikkelaars van websites die bezoekers willen laten authenticeren met behulp van HDN certificaten.

Een basiskennis van implementaties van webservices met behulp van SOAP/WSDL en of HTTP/XML (XSD) wordt verondersteld. Tevens wordt globale kennis verondersteld met betrekking tot het configureren van webservers.

1.3 – Reikwijdte

De scriptvoorbeelden in dit document worden gepresenteerd in PHP. Van de lezer wordt verwacht dat deze de PHP scripts eventueel kan vertalen naar een voor hem of haar beschikbaar alternatief.

Verder wordt de webserver configuratie aan de hand van Apache gedemonstreerd. Indien de ontwikkelaar voor een andere webserver kiest wordt verwacht dat deze zelf de vertaalslag kan maken.

2. Algemene beschrijving

2.1 – Inleiding

DARTS is een uitbreiding van het Hypotheken Data Netwerk. Met DARTS kunnen dienstaanbieders diensten voor HDN aansluitnummers publiceren. Tevens definieert en implementeert DARTS de structuur en accounting betreffende de autorisatie en authenticatie voor het gebruik tot de diensten.

Binnen het HDN netwerk wordt de authenticatie gedaan op basis van HDN certificaten. Certificaten zijn aan een aansluitnummer gekoppeld. De aansluitnummers kunnen in groepen geplaatst worden. Groepen kunnen op hun beurt aan een dienst gekoppeld worden. Hiermee kan worden gedefinieerd welke groepen en dus welke aansluitnummers, toegang tot een dienst hebben. Tevens kunnen aan een dienst-groep combinatie parameters gekoppeld worden. Het configuren van DARTS gebeurt via een webapplicatie Dartsbeheer. Verdere informatie over DARTS en Dartsbeheer is te vinden in de handleiding DARTS.

Indien een gebruiker een website bezoekt kan de webserver zo geconfigureerd worden dat deze de browser om een HDN certificaat vraagt. Dit document beschrijft de implementatie om bezoekers van websites te authentiseren en autoriseren met behulp van HDN certificaten.

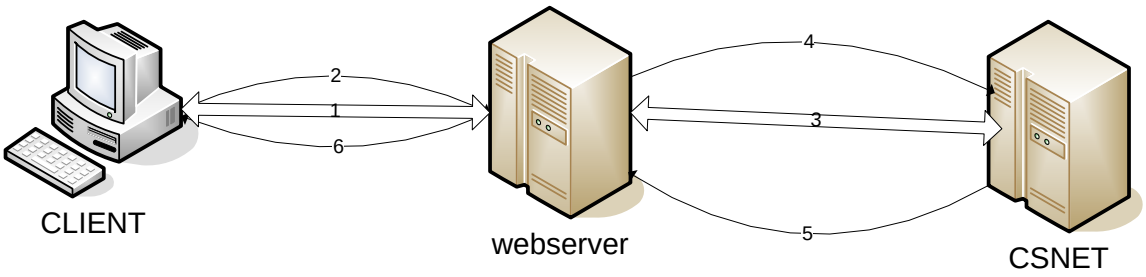
3. De Webserver

3.1 – Inleiding

Alle authenticatie via DARTS geschiedt op basis van certificaten. Aan een browser kan door de webserver om een cliënt certificaat gevraagd worden. Hiervoor dient de webserver dan geconfigureerd te worden.

Nadat de webserver het cliënt certificaat ontvangen heeft dient de webserver zichzelf ook met een eigen certificaat bij de CSNET server te authentiseren.

In onderstaand figuur is schematisch weergeven welke stappen doorlopen worden.



De cliënt maakt via een web-browser verbinding met de webserver. Hiervoor wordt een SSL verbinding opgezet (1). De webserver is zo geconfigureerd dat deze om een cliënt certificaat vraagt. Indien de cliënt een certificaat heeft en deze selecteert wordt het certificaat naar de webserver verzonden (2).

Met dit certificaat weet de webserver alleen dat het een geldig HDN certificaat is. Om te autoriseren dient de webserver een webrequest naar de CSNET server te maken (3). Dit verloopt ook over een SSL verbinding.

De CSNET server accepteert alleen een autorisatie-request van aansluitnummers die beheerder zijn van de betreffende dienst. De webserver dient dus ook geauthentiseerd en geautoriseerd te worden. Dit gebeurt ook met behulp van een HDN certificaat (4).

Indien de dienstbeheerder toegang heeft zal de CSNET webservice het in (2) verkregen certificaat authentiseren en het resultaat in (5) terugsturen naar de webserver. De webserver kan nu met de verkregen rechten voor de gebruiker de webpagina tonen (6).

3.2 – Webserver configuratie

Het valt buiten de scope van dit document om voor alle webservers te beschrijven hoe gebruik gemaakt kan worden van cliëntcertificaat authenticatie. Als voorbeeld wordt hier rudimentair uiteen gezet hoe Apache geconfigureerd kan worden.

De communicatie in (3),(4) wordt opgezet door de CSNET server. Hier hoeft dus op de webserver niets voor geconfigureerd te worden. Er wordt wel om een certificaat gevraagd. Dit wordt echter via een script gerealiseerd en wordt besproken in hoofdstuk 4.

(1) en (2) verlopen over een beveiligde verbinding. Dit gebeurt m.b.v. een SSL verbinding. Hiervoor is dus een Server Certificaat benodigd. Dit certificaat is niet per se gerelateerd aan HDN certificaten (mag wel).

Configureer een SSL verbinding op de webserver

De server is op de volgende manier geconfigureerd voor port# 443

SSLEngine ON

SSLCertificateFile [path]/server.pem

SSLCertificateKeyFile [path]/server.key

Server.pem/server.key zijn hier dus het server certificaat (en hoeft dus geen HDN certificaat te zijn).

Nadat de SSL verbinding geconfigureerd is kan de webserver de cliënt om cliënt-certificaten vragen. Hiervoor dienen additionele instellingen toegevoegd te worden.

Op webserver wordt optioneel om een cliënt certificaat gevraagd d.m.v. e.g.:

SSLCACertificateFile [path]/ hdnca.pem

SSLVerifyClient optional

SSLOptions +ExportCertData +StdEnvVars

SSLVerifyDepth 1

Indien de cliënt een certificaat heeft die dezelfde CN als het root CA certificaat hdnca.pem heeft zal de browser de cliënt vragen het certificaat te selecteren. Als de cliënt deze selecteert wordt deze naar de server verzonden. Indien dit certificaat ook daadwerkelijk door het root certificaat ondertekend is wordt de verbinding door de server geaccepteerd.

Door de optie +ExportCertData kan in e.g. PHP het serienummer via `$_SERVER['SSL_CLIENT_M_SERIAL']` uitgelezen worden.

3.2.1 – Certificaten formaten

HDN certificaten zijn standaard van binair (cer of p12) formaat. Apache vereist een BASE64 encoded DER certificaat. Op de volgende manier kan een p12 omgezet worden naar een .pem formaat:

openssl pkcs12 -in cert.p12 -out cert.pem

3.2.2 – HDN Root certificaat

Het HDN root certificaat kan via http://www.hdn.nl/downloads/basic-software gedownload worden. Dit is het certificaat die de webserver gebruikt om de HDN cliënt certificaten te valideren. Met

openssl verify -CAfile hdnca.pem -verbose xxx.pem

kan gecontroleerd worden of een cliënt HDN certificaat (xxx.pem) ook daadwerkelijk door dit CA certificaat ondertekend is.

4. Webservice aanroep

4.1 – Inleiding

Door de webserver te configureren zoals in hoofdstuk 3 is beschreven kan e.g. een php script het serienummer van het cliënt certificaat lezen. Het serienummer en de dienst waartegen geautoriseerd dient te worden kan via een webservice naar CSNET verzonden worden.

De webservice aanroep verloopt over een vergelijkbare verbinding als die van het cliënt certificaat. De CSNET server accepteert alleen een request indien deze ook een HDN certificaat gebruikt.

De URL van de webservice is:

https://pwws.hdn.nl/services/csnet_ws.php?wsdl

4.2 – Webservice operaties

De webservices die via een WSDL request aangeroepen kunnen worden hebben de volgende basis functionaliteiten:

Een aansluitnummer of certificaatserienummer voor een dienst autoriseren

Een record toevoegen aan het centrale DARTS accounting systeem

Een overzicht van alle groepen waar een aansluitnummer in zit opvragen

4.3 – WSDL Request

Het WSDL is volgens de rpc/encoded standaard.

Een request waar de toegang wordt gevraagd voor het certificaat met serienummer '10CF00000C06' voor de dienst met de logische naam man_info kan als volgt:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<SOAP-ENV:Envelope>
        <SOAP-ENV:Body>
                <tns:authCertificateForGroup xmlns:tns="urn:darts">
                        <serial xsi:type="xsd:string">10CF00000C06</serial>
                        <LogDienstNaam xsi:type="xsd:string">man_info</LogDienstNaam>
                </tns:authCertificateForGroup>
        </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Voor de WSDL request geldt ook dat deze met behulp van een cliënt certificaat gedaan moet worden.

4.3.1 – WSDL Request voorbeeld applicatie

Appendix A bevat de source code van een voorbeeld script. Via CSNET kan de complete source code verkregen worden.

4.3.2 – WSDL functies

4.3.2.1 – authCertificateForGroup en authAansluitnummerForGroup

Met deze 2 functies kan een gebruiker voor een dienst geautoriseerd worden. Beide functies hebben als input parameter de logische naam van een dienst en een serienummer van een certificaat of het aansluitnummer van de gebruiker.

Indien de gebruiker toegang tot de dienst heeft wordt er een SOAP response geretourneerd die in de SOAP body het element aansluitnummer heeft. Hier kan dus op gecontroleerd worden.

Als de gebruiker geen toegang heeft wordt er een SOAP fault geretourneerd.

4.3.2.2 – getGroupsForAansluitnummer

Deze functie heeft als input het aansluitnummer van de gebruiker en de logische dienstnaam. De response bevat een soap array met alle groepen waarmee het aansluitnummer toegang tot de dienst heeft (dus niet allen de groep die de hoogste prioriteit heeft en die via de in 4.3.2.1 beschreven functies geretourneerd worden)

4.3.2.3 – getAllParamsForService en getAllParamsForServiceFromCert

Een aansluitnummer heeft via een groep toegang tot een dienst. Per groep die toegang heeft tot een dienst kunnen er andere parameters geretourneerd worden. De functie getAllParamsForService is een uitgebreide versie van getGroupsForAansluitnummer. Naast alle groepen worden ook per groep alle parameters geretourneerd.

getAllParamsForServiceFromCert heeft als input parameter niet het aansluitnummer maar het certificaat id van het client certificaat.

4.3.2.4 – doAccounting

Met doAccounting kan voor een aansluitnummer in de centrale dartsdatabase vastgelegd worden op welk tijdstip welke dienst opgevraagd is. Zie verder 4.4.

4.3.2.5 – getAansluitnummersInSysGroup

De input voor deze functie is de naam van de systeemgroep. De response is een soap array met alle aansluitnummers behorende tot die systeemgroep.

Een voorbeeld body van de aanvraag is:

```
<soap:Body>
        <getAansluitnummersInSysGroup xmlns="urn:darts">
                <GroupPath>de hypotheekmeester</GroupPath>
        </getAansluitnummersInSysGroup>
</soap:Body>
```

Een voorbeeld antwoord:

```
<SOAP-ENV:Body>
		<ns1:getAansluitnummersInSysGroupResponse xmlns:ns1="urn:darts">
			<Result xsi:type="SOAP-ENC:Array" SOAP-ENC:arrayType="xsd:string[5]">
				<item xsi:type="xsd:string">222518</item>
				<item xsi:type="xsd:string">222563</item>
				<item xsi:type="xsd:string">222589</item>
		<item xsi:type="xsd:string">222599</item>
				<item xsi:type="xsd:string">222600</item>
			</Result>
		</ns1:getAansluitnummersInSysGroupResponse>
</SOAP-ENV:Body>
```

4.4 – Accounting

Darts voorziet ook in een webservice operatie ten behoeve van accounting. Bij standaard autorisatie zoals eerder beschreven zal er geen accounting plaatsvinden. Voor de WSDL bestaat er de operatie 'doAccounting'. Deze operatie authentiseert net zoals 'authCertificateForGroup' maar verzorgt tevens de accounting binnen DARTS. Voor XML requests kan accounting worden verzorgd door 'DoAccouting' de waarde true te geven (zie XSD in appendix).

De operatie accounting legt het tijdstip, aansluitnummer, dienst en de groep waarmee het aansluitnummer toegang krijgt vast.

Het is de verantwoording van de dienstaanbieder om accounting op een voor de dienst relevante wijze aan te roepen.