

GeoShop SAMLAdapter Benutzerhandbuch

Zusammenfassung

Diese Dokumentation beschreibt die Installation und Konfiguration des GeoShop SAMLAdpater.

Die Dokumentation darf nur mit Erlaubnis der infoGrips GmbH vervielfältigt werden.

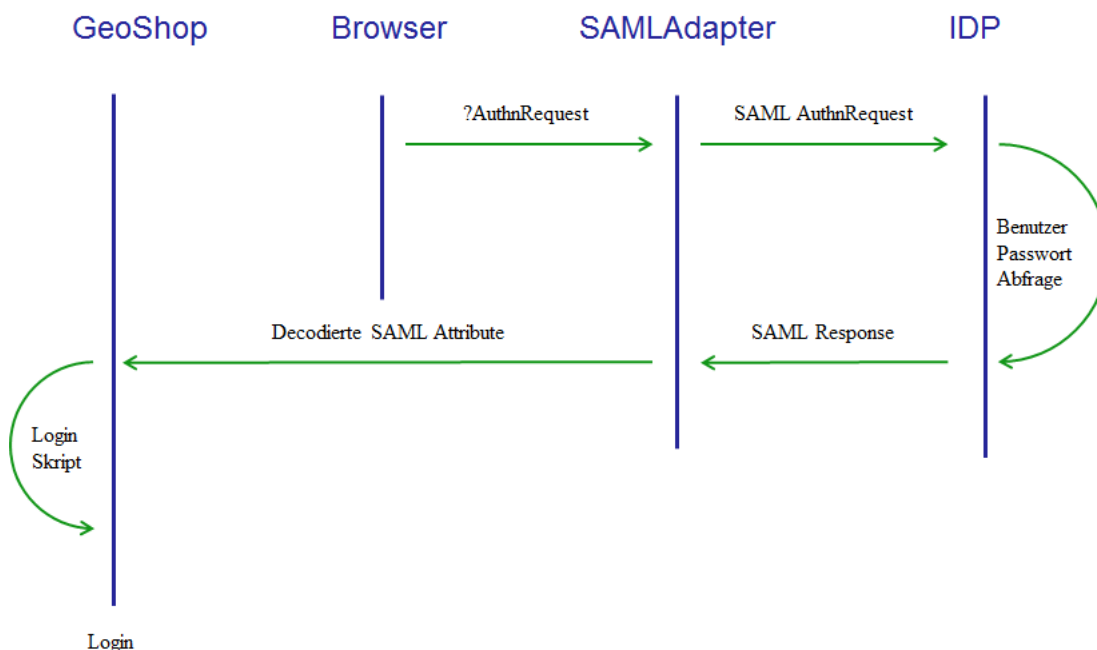
Inhaltsverzeichnis

1. Einleitung	4
1.1. Überblick	4
1.2. Begriffe	5
1.3. Aufbau dieser Dokumentation	5
1.4. Konventionen	5
2. Aktivierung und Konfiguration des SAMLAdapter	6
2.1. Überblick	6
2.2. Aktivierung des SAMLAdapter	6
2.3. Konfiguration des SAMLAdapter	6
2.4. Konfiguration des Login Skript	7
2.5. GeoShop Neustart	8
3. Metadatenaustausch	9
4. Hinweise zur Sicherheit	10
A. Anpassen des LOAD_USER Skript	10
1. Einfaches Beispiel (<i>simple.cfg</i>)	10
2. Erweitertes Beispiel (<i>extended.cfg</i>)	11
3. Fehlersuche	12
B. Erstellen von Private Key und Zertifikat	12

1. Einleitung

1.1. Überblick

Über den GeoShop Modul **SAMLAdapter** kann die interaktive Authentifizierung von GeoShop Client Benutzern an einen externen SAML IdP (SAML Identity Provider) ausgelagert werden. Die Kommunikation zwischen SAMLAdapter und dem IdP erfolgt über das SAML (Secure Assertion Meta Language) Protokoll. Die Kommunikation zwischen dem SAMLAdapter und dem IdP wird durch den gegenseitigen Austausch von Zertifikaten gesichert.



Ablauf einer SAML Authentifizierung:

1. Der Browser des Benutzers ruft die Adresse `GEOSHOP_BASE_URL/SAMLAdapter?SAMLRequest=AuthnRequest` auf.
2. Der SAMLAdapter erstellt einen mit seinem Zertifikat signierten SAML AuthnRequest und leitet den Browser an `IDP_SIGNON_URL` weiter.
3. Der IdP prüft die Signatur des AuthnRequest. Falls die Überprüfung erfolgreich war, zeigt der IdP dem Benutzer eine Login Maske an.
4. Der Benutzer gibt seinen IdP Benutzernamen und das IdP Passwort ein.
5. Der IdP erstellt eine SAML Response und füllt darin die gewünschten Benutzerattribute ab. Die Response Meldung wird mit dem Zertifikat des IdP signiert und an `GEOSHOP_BASE_URL/SAMLAdapter?Response=<Response>...` übermittelt.
6. Der SAMLAdapter überprüft die Signatur der SAML Response. Falls die Überprüfung erfolgreich ist, loggt sich der GeoShop Client ein.
7. Die vom IdP übermittelten Benutzerattribute werden vom Login Skript im GeoShop dekodiert.
8. Der Login Skript überprüft zur Sicherheit nochmals, ob die Signatur ursprünglich vom IdP kam.

9. Der Login Skript generiert aufgrund der SAML Attribute einen temporären GeoShop Benutzer.
10. Der Browser loggt sich unter dem generierten GeoShop Benutzer im GeoShop ein.

1.2. Begriffe

SAML	Security Assertion Meta Language. OASIS Standard für den Austausch von Identitäts Daten (z.B. Name, Vorname, EMail, etc.).
IdP	Identity Provider. Dienst im Netz, welcher die Benutzerinformationen in einer Datenbank führt (z.B. Benutzer, Passwort, Name, Vorname, E-Mail, etc.).
SAML Metadaten	XML formatierte Daten, welche bei der Installation zwischen IdP und GeoShop ausgetauscht werden.

1.3. Aufbau dieser Dokumentation

Diese Dokumentation ist wie folgt aufgebaut:

- Kapitel 1: Einleitung.
- Kapitel 2: Aktivierung und Konfiguration des SAMLAdpater.
- Kapitel 3: Metadatenaustausch.
- Kapitel 4: Hinweise zur Sicherheit.
- Anhang: Enthält Beispiele für Login Skripts und die Hinweise für die Erstellung von Zertifikaten.

1.4. Konventionen

In dieser Dokumentation werden folgende Konventionen eingehalten:

Kursiv	Namen von Dateien und URL's
fett	neue Begriffe, Namen von Funktionen oder Methoden
<code>courier</code>	Programmtext oder Eingaben im Betriebssystem

2. Aktivierung und Konfiguration des SAMLAdapter

2.1. Überblick

Die Installation des SAMLAdapter umfasst folgende Schritte:

- Aktivierung. Der SAMLAdapter muss im GeoShop aktiviert und als GeoShop Dienst ausgeführt werden.
- Konfiguration. Die notwendigen SAMLAdapter Parameter müssen in `appserver.opt` des GeoShop gesetzt werden.
- Login Skript: Es muss ein Login Skript konfiguriert / erstellt werden. Der Login Skript bildet die externen SAML Benutzer des IdP auf interne GeoShop Benutzer ab.
- GeoShop Neustart: Der SAMLAdapter wird geladen und die Konfigurationsparameter gelesen. Der GeoShop ist für SAML Anfragen bereit

2.2. Aktivierung des SAMLAdapter

Der SAMLAdapter muss zuerst als GeoShop Dienst aktiviert werden. Dazu muss ein zusätzlicher Eintrag in der Services Datei `services.srv` des GeoShop erstellt werden (s.a. GeoShop Benutzerhandbuch).

```
LIST
...
MAP
    name STRING samladapter
    description STRING 'SAMLAdapter'
    module STRING SAMLAdapter
    server STRING appserver
}
...
}
```

2.3. Konfiguration des SAMLAdapter

Als Nächstes können die Parameter des SAMLAdapter in `appserver.opt` des GeoShop konfiguriert werden:

```
MAP
...
SAMLAdapter MAP
    [ CERTDB_STORE STRING <CertDbStore> ]
    [ CERTDB_STORE_PASSWORD STRING <CertDbStorePassword> ]
    [ CERTDB_KEY_PASSWORD STRING <CertDbKeyPassword> ]
    [ CERTDB_ALIAS STRING <CertDbAlias> ]
    GEOSHOP_ENTITYID STRING <EntityId>
    GEOSHOP_CLIENT_URL STRING <ClientUrl>
    IDP_SIGNON_URL STRING <SignOnUrl>
    IDP_CERTIFICATE STRING <IdpCertificate>
    VALIDATE_SIGNATURE STRING <ON | OFF>
```

```

    LOG_SAML_REQUESTS STRING <ON | OFF>
    DEBUG STRING <ON | OFF>
  }
  ...
}

```

Folgende Felder sind einzutragen:

Parameter	req/opt	Typ	Beschreibung
CERTDB_STORE	o	STRING	Name der Zertifikatdatenbank (Default: <code>saml.pfx</code>). Die Zertifikatdatenbank muss sich im Ordner <code>\user\certs</code> befinden und im PKCS12 Format abgespeichert sein. Falls die Zertifikatdatenbank nicht existiert, wird <code>saml.pfx</code> beim Start des GeoShop automatisch aus einem selbst signierten Zertifikat erstellt.
CERTDB_STORE_PASSWORD	o	STRING	Passwort für die Zertifikatdatenbank.
CERTDB_KEY_PASSWORD	o	STRING	Passwort für den private Key in der Zertifikatdatenbank.
CERTDB_ALIAS	o	STRING	Alias unter dem das Zertifikat / private Key in der Zertifikatdatenbank abgelegt ist (Default: <code>geoshop</code>). Falls der Alias in der Zertifikatdatenbank nicht existiert, wird vom GeoShop das erste Zertifikat / der erste private Key in der Zertifikatdatenbank gewählt.
GEOSHOP_ENTITYID	r	STRING	Eindeutige Id, welche den GeoShop gegenüber dem IdP identifiziert. Es ist üblich dafür eine URL zu verwenden, z.B. <code>https://geoshop.com/SAMLAdapter</code>
GEOHOP_CLIENT_URL	r	STRING	URL der GeoShop Client Anwendung, z.B. <code>https://geoshop.com/client5/index.html</code>
IDP_CERTIFICATE	r	STRING	X509 Zertifikat des IdP als Base64 codierte Zeichenkette (ohne Header, Trailer, Zeilenumbrüche). Wird vom IdP beim Metadatenaustausch geliefert (s.a. Kapitel 3).
IDP_SIGNON_URL	r	STRING	URL an den der GeoShop einen SAML AuthnRequest weiterleiten soll. Wird vom IdP beim Metadatenaustausch geliefert (s.a. Kapitel 4).
VALIDATE_SIGNATURE	o	STRING	ON OFF. Signatur des IdP mit dem Zertifikat des IdP prüfen. Default: ON.
LOG_SAML_REQUESTS	o	STRING	ON OFF. SAML Meldungen in <code>appserver.log</code> aufzeichnen. Default: OFF.
DEBUG	o	STRING	ON OFF. Zusätzliche Meldungen in <code>appserver.log</code> ausgeben. Default: OFF.

2.4. Konfiguration des Login Skript

Schliesslich muss noch in `appserver.opt` im Abschnitt `GeoShopServer` der Parameter `LOAD_USER` gesetzt werden.

```

MAP
  ...
  GeoShopServer MAP

```

```
...  
    LOAD_USER STRING saml,/script/saml/simple.cfg  
}  
...  
}
```

Im Standard Skript `/script/saml/simple.cfg` werden alle via SAML authentifizierten Benutzer auf einen GeoShop Benutzer gemappt, der die gleichen Rechte hat wie der Benutzer `\users\saml_template.usr`. Der Skript `simple.cfg` und die Benutzerdatei `saml_template.usr` können an die eigenen Bedürfnisse angepasst werden (z.B. Mapping auf verschiedene GeoShop Benutzer Gruppen, s.a. Anhang).

2.5. GeoShop Neustart

Der GeoShop muss frisch gestartet werden, damit der SAMLAdapter geladen wird. In `appserver.log` müssen folgende Meldungen erscheinen:

```
-----  
module SAMLAdapter 2022.0x  
-----  
testing environment ...  
  GeoShop EntityID is https://<GeoShop-URL>/SAMLAdapter  
  GeoShop Client URL is https://<GeoShop-URL>/client5/index.html  
  GeoShop private key loaded from saml.pfx  
  GeoShop certificate loaded from saml.pfx, valid until <Date>  
  GeoShop key and certificate match  
  IdP SignOn URL is <IdP-SignOnURL>  
  IdP certificate loaded, valid until <Date>
```

Falls Fehlermeldungen angezeigt werden (z.B. ungültiges Zertifikat), müssen die Ursachen behoben werden. Nach der Fehlerbehebung sollte der GeoShop nochmals gestartet werden.

3. Metadaten austausch

Um den SAMLAdapter mit einem konkreten IdP zu verbinden, müssen Metadaten zwischen GeoShop und IdP ausgetauscht werden. Sofern der SAMLAdapter wie beschrieben erfolgreich aktiviert und konfiguriert wurde, können die GeoShop SAML Metadaten im XML-Format via die URL

`GEOSHOP_BASE_URL/SAMLAdapter`

bezogen werden. Die so generierte XML-Datei muss an den IdP übermittelt werden. Der IdP übergibt ebenfalls eine XML-Datei mit seinen Metadaten. Aus der XML-Datei müssen die Werte für die Parameter `IDP_CERTIFICATE` und `IDP_SIGNON_URL` entnommen und in `appserver.opt` im Abschnitt `SAMLAdapter` eingetragen werden. Falls die Parameter in `SAMLAdapter` ändern, oder wenn die Zertifikate ablaufen, müssen die Metadaten erneut ausgetauscht werden.

4. Hinweise zur Sicherheit

Das SAML Protokoll ist ein *sehr sicheres Protokoll*, da es auf beidseitiger Signierung (GeoShop und IdP) der ausgetauschten Meldungen beruht. Die Signierung basiert auf allgemeinen kryptographischen Verfahren und den *gegenseitig* ausgetauschten Zertifikaten. Trotzdem kann es bei Fehlkonfiguration der Systeme zu Sicherheitsproblemen kommen. Es ist daher *sehr wichtig*, dass folgende Punkte beachtet werden:

- In der Produktion *müssen* die SAML Meldungen des IdP überprüft werden (VALIDATE_SIGNATURE ON), sonst könnte ein unbekannter IdP gefälschte Daten an den GeoShop übermitteln. Der Parameter VALIDATE_SIGNATURE darf *nur* zu Testzwecken auf OFF gesetzt werden.
- Es *muss* ein Login Skript unter LOAD_USER konfiguriert werden und der Skript *muss* die Bibliothek saml.lib einbinden.

A. Anpassen des LOAD_USER Skript

Es können angepasste Varianten des LOAD_USER Skript erstellt werden. Damit kann u.A. Folgendes erreicht werden:

- Aufgrund der vom IdP gelieferten Benutzerattribute (z.B. Name, Vorname, EMail oder Rolle), können unterschiedliche GeoShop Benutzer Templates geladen werden.
- Es können Benutzer zurückgewiesen werden, weil sie sich in einer lokalen Sperrliste befinden.
- Benutzer welche lange inaktiv waren, können automatisch gesperrt werden.
- etc.

1. Einfaches Beispiel (*simple.cfg*)

Nachfolgend ein Beispiel für einen sehr einfachen Login Skript:

```
! saml.lib MUSS immer eingebunden werden !!!
|INCL \script\saml.lib

'saml_template.usr' => VAR.TEMPLATE_USER
IF OPT.user_dir . '\users\' . VAR.TEMPLATE_USER SERIAL.LOAD_OBJECT THEN
    => VAR.USER
    OPT.output &VAR.USER SERIAL.SAVE_OBJECT
ELSE
    'Der Template Benutzer ' . VAR.TEMPLATE_USER . ' existiert nicht' => VAR.MESSAGE
    VAR.MESSAGE SAML_APPEND_LOG
    VAR.MESSAGE SAML_SEND_ERROR
END_IF
```

Folgende Bemerkungen zum Skript:

- Der Login Skript *muss* die Bibliothek \script\saml.lib einbinden. In der damit automatisch aufgerufenen Prozedur SAML_INITIALIZE wird geprüft, ob der Skript via eine korrekt signierte

SAML Meldung vom IdP ausgelöst wurde. Dieser Test ist *sehr wichtig*, `saml.lib` muss daher immer im Login Skript eingebunden werden, selbst wenn keine weitere Prozedur aus `saml.lib` benötigt wird.

- Der Skript lädt den Template Benutzer `\users\saml_template usr` mit `SERIAL.LOAD_OBJECT`. Der Benutzer `saml_template` ist mit den gewünschten GeoShop Berechtigungen konfiguriert (z.B. Produkte, Views, etc.).
- Der Skript speichert den `saml_template` Benutzer unter `OPT.output`. Der GeoShop liest danach die generierte Benutzerdatei und führt den Login durch.
- Falls das Benutzer Template `saml_template usr` nicht gefunden wird, wird dem im GeoShop Client Benutzer mit `SAML_SEND_ERROR` eine Fehlermeldung angezeigt. Zusätzlich kann mit `SAML_APPEND_LOG` eine Meldung in die Logdatei `\data\logs\samllogs\samllogin.log` geschrieben werden.

Das obige Beispiel ist wie bereits erwähnt sehr einfach. Zusammen mit der Map `SAML_ATTRIBUTES` können im Skript zusätzlich die vom IdP gelieferten Attribute abgefragt werden. Damit ist es möglich abhängig von den Attributwerten unterschiedliche Benutzertemplates zu laden (z.B. abhängig vom Attribut `role`). Ausserdem können die Attributwerte dazu verwendet werden, den aus dem Template generierten Benutzer mit zusätzlichen Informationen zu ergänzen (z.B. Name, Vorname, EMail, etc.).

2. Erweitertes Beispiel (*extended.cfg*)

Im nachfolgenden Beispiel werden die vom IdP gelieferten Attribute auf Standardnamen gemappt und die Benutzer über das Attribute `role` in Gruppen eingeteilt:

```
MAP SAML_ATTRIBUTE_MAPPER
    Rolle => role
    Name => name
    Vorname => firstname
    Mail => email
END_MAP


MAP SAML_GROUPS ! by role
    Benutzer => user
    Administrator => admin
    DEFAULT => user
END_MAP

! saml.lib MUSS immer eingebunden werden !!!
|INCL \script\saml.lib

'saml_' . SAML_ATTRIBUTES.role . '.usr' => VAR.TEMPLATE_USER
IF OPT.user_dir . '\users\' . VAR.TEMPLATE_USER SERIAL.LOAD_OBJECT THEN
    => VAR.USER
    &VAR.USER 'order.name1' SAML_ATTRIBUTES.firstname . ' ' . SAML_ATTRIBUTES.name MAPINS
    &VAR.USER 'order.email' SAML_ATTRIBUTES.email MAPINS
    OPT.output &VAR.USER SERIAL.SAVE_OBJECT
ELSE
    'Der Template Benutzer ' . VAR.TEMPLATE_USER . ' existiert nicht' => VAR.MESSAGE
    VAR.MESSAGE SAML_APPEND_LOG
    VAR.MESSAGE SAML_SEND_ERROR
END_IF
```

Folgende Bemerkungen zum Skript:

- Über die Map `SAML_ATTRIBUTE_MAPPER` können die vom IdP gelieferten Attributnamen vereinheitlicht werden (z.B. Vorname => `firstname`).
- In der Map `SAML_GROUPS` werden die Benutzer via das Attribut `role` in Gruppen eingeteilt.
- Der Template Benutzer wird über den Inhalt des Attributs `role` dynamisch generiert (z.B. Administrator => `saml_admin.usr`).
- Der aus dem Template generierte Benutzer wird mit dem SAML Attributen `firstname`, `name` und `email` ergänzt. Alle verfügbaren SAML Attribute können über die Map `SAML_ATTRIBUTES` abgefragt werden.

 Von den IdP's werden nicht immer die gleichen Benutzerattribute geliefert. Das Attribut `role` ist z.B. nicht immer vorhanden. Typischerweise werden mindestens `name`, `firstname` und `email` geliefert. Fragen Sie dazu den Betreiber des IdP.


3. Fehlersuche


Die Authentifizierung mit SAML ist relativ komplex. Bei Problemen kann es u.U. schwierig sein die Ursache zu finden. Im GeoShop wurde daher unter `\data\logs\samllogs` ein sperates Verzeichnis für die Logdateien des Login Skript eingerichtet. Ausserdem können alle SAML Meldungen (eingehend und ausgehend) mit `LOG_SAML_MESSAGES ON` in `appserver.log` aufgezeichnet werden. Falls der Parameter `DEBUG` mit `ON` aktiviert wurde, werden noch mehr Details angezeigt.


B. Erstellen von Private Key und Zertifikat

Für das Signieren der SAML Meldungen benötigt der GeoShop einen Private Key. Das zugehörige Zertifikat muss dem IdP für die Verifikation der Signatur übergeben werden. Private Key und zugehöriges Zertifikat können wie folgt erstellt werden:

- Man bestellt ein Zertifikat bei einer offiziellen CA (Certificate Authority). Das Zertifikat wird von der CA zusammen mit dem Private Key in einer `.pfx` Datei im PKCS12 Format geliefert. Die `.pfx` Datei muss nach `\user\certs\saml.pfx` kopiert und die `CERTDB_` Optionen in `appserver.opt` gesetzt werden (s.a. Konfiguration SAMLAdapter).
- Man lässt den GeoShop ein selbst signiertes Zertifikat erstellen. Falls `\user\certs\saml.pfx` noch nicht existiert, muss man den GeoShop frisch starten und der GeoShop erstellt automatisch die notwendige `saml.pfx` Datei. Das Clientzertifikat kann über die Metadaten URL `GEOSHOP_BASE_URL\SAMLAdapter` abfragen.

 Die von einer offiziellen CA erstellten Zertifikate müssen jedes Jahr erneuert werden.

 Das vom GeoShop erstellte selbst signierte Zertifikat ist 10 Jahre gültig.

 Man kann auch den GeoShop SSL-Zertifikatspeicher `ssl.pfx` benutzen. Dazu muss man die Option `CERTDB_STORE` auf den Wert `ssl.pfx` setzen.