

Medgivande via hemsida

Teknisk manual för webbutvecklare

November 2010

1 Kontaktvägar och adresser

Postadress BGC, 105 19 Stockholm

Telefon, fax och e-post till BGC Kundservice hittar du på bgc.se under Kontakta oss:
www.bgc.se/kontaktaoss

Hitta information snabbt och enkelt på webben!

Besök gärna vår webbsida www.bgc.se. Där hittar du aktuell, produktinformation, frågor & svar samt manualer och trycksaker.

Innehåll

1	Kontaktvägar och adresser	2
2	Inledning.....	4
2.1	Om manualen	4
2.1.1	Del 1 Downloader:	4
2.1.1	Del 2 Verifier:	4
2.2	Övrig information.....	5
2.2.1	Testa formulär och medgivanden.....	5
2.2.1	Soap-protokoll.....	5
3	Downloader	6
3.1	Operationer.....	6
3.1.1	Logon	6
3.1.1	Query.....	6
3.1.1	Fetch.....	7
3.2	Exempel.....	7
3.2.1	(C#, .Net) för hämtning av medgivande.....	7
3.2.1	Exempel medgivandefil.....	8
3.3	Skicka medgivnadeunderlag till BGC.....	8
4	Del 2 Verifier	9
4.1	Inledning.....	9
4.2	Operationer.....	9
4.2.1	Logon	9
4.2.1	StoreSignature	10
4.2.1	VerifySignature	10
4.2.1	GetSignatureInfo	11
4.3	Exempel (C#, .Net).....	12
4.4	Felkommentarer	13
4.5	Skicka medgivnadeunderlag till BGC.....	13

2 Inledning

Medgivande via hemsida är en tilläggstjänst för dig som betalningsmottagare i Autogiro Privat. Tjänsten innebär att du kan låta dina betalare teckna sina medgivanden direkt från er hemsida.

Tjänsten är väldigt enkel och kan användas av alla företag med en hemsida som har/vill ha Autogirobetalande kunder. Eftersom tjänsten bygger på en unik länk till ditt formulär kan du även använda tjänsten för exempelvis riktade mailutskick. Genom nedanstående enkla åtgärder kan du nå dina betalare direkt på Internet:

1. Utforma ett medgivandeformulär utifrån egna preferenser och behov.
2. Lägg sedan upp den unika länken som skapas efter att ni gjort färdigt formuläret och placera den på er hemsida.
3. Betalaren fyller i uppgifterna och signerar med hjälp av en e-legitimation. Ni hämtar sedan medgivandet från BGC.

När du hämtat hem medgivandet och lagt upp/identifierat kunden i ditt system skickar du in ett medgivandeunderlag till BGC. Du kan välja att helautomatisera hela medgivandehanteringens och slippa manuella mellansteg eller fortsätta hantera/registrera dina medgivanden som vanliga pappersmedgivanden.

2.1 Om manualen

Manualen riktar sig till dig som har god kunskap om webbutveckling. Den är uppdelad i två olika avsnitt. Avsnitten beskriver helt olika funktioner som är oberoende av varandra. Båda tjänsterna är dock framtagna för att ni ska kunna underlätta er medgivandeprocess.

2.1.1 Del 1 Downloader:

Det finns två olika sätt att hämta nya medgivanden, antingen genom att logga in i Medgivande via hemsida-portalen eller hämta medgivandena via webbtjänst-funktionen Downloader. Filerna ser olika ut beroende på hur ert medgivandeformulär är utformat. All information som ni angett i formuläret samt allt som betalare fyllt i kommer med i medgivandet till er.

Del 1 av manualen beskriver hur du gör för att hämta hem medgivanden i XML-format via webbtjänsten ”Downloader”.

2.1.1 Del 2 Verifier:

För dig som har väldigt specifika önskemål om hur ett formulär ska vara utformat kan välja att bygga ditt eget formulär helt på egen hand och sedan endast kontrollera att signaturen var giltig mot BGC. På detta sätt kan du utnyttja ditt befintliga gränssnitt mot dina betalare men fortfarande verifiera att signaturen som kunden angett stämmer.

Del 2 av denna manual beskriver hur ni gör för att kontrollera att signaturen är korrekt mot BGC:s webbtjänst ”Verifier”.

Övrig information

2.1.1 Testa formulär och medgivanden

Du kan genomföra ett fullskaligt test innan du börjar använda tjänsten fullt ut i ditt system. Genom att skapa ett formulär kan ditt företag testa hur medgivandefilerna för just ert företag kommer att se ut vid riktig produktion. Detta görs genom att testa formuläret. När formuläret är testat och korrekt signerat lagras ett testmedgivande i systemet som kan hämtas på vanligt sätt, alltså genom inloggning i portalen eller via webbtjänsten ”Downloader”.

Läs mer om test i Användarmanualen.

2.1.1 Soap-protokoll

Webbtjänsterna använder SOAP-protokollet för att överföra informationen; ett SOAP-anrop skickas till webbtjänsten som returnerar ett SOAP-svar. För att få informationen om SOAP-anrop och svar används länkarna:

<https://mh.bankgirot.se/MvHWS/Downloader.aspx>

<https://mh.bankgirot.se/MvHWS/Verifier.aspx>

För att bibehålla informationen mellan anropen måste ”session cookies” användas.

3 Downloader

3.1 Operationer

Webbtjänsten stöder 3 operationer:

Logon – logga in i webbtjänsten

Query - fråga hur många medgivanden som finns

Fetch - hämta hem medgivanden

3.1.1 Logon

Innan hämtning av medgivanden kan ske måste programmet logga in. *Logon* -operationen används på följande sätt för att utföra detta: (userName är bankgironummer i formatet 9999-9999). Användarnamn (userName) samt lösenord (password) skickas via ett välkomstbrev till ert företag när ni börjar använda tjänsten. Samma lösenord gäller för inloggning i portalen som vid inloggning till denna webbtjänst.

OBS! Efter 3 gånger med fel lösenord spärras möjligheten att logga in. Var därför uppmärksam på vilka koder som anropet returnerar. Vid spärrat lösenord måste BGC Kundservice kontaktas.

```
int Logon(string userName, string password)
```

Anropet returnerar en kod enligt tabellen nedan.

Kod	Beskrivning
0	Inloggningen lyckades
1	Fel användarnamn eller lösenord
2	Inloggningskontot har spärrats!
3	lösenordet måste ändras

3.1.1 Query

Query används för att initiera sökningar efter medgivanden. I anropet måste man specificera om testmedgivanden eller vanliga medgivanden ska hämtas:

```
int Query(bool test) test == true => hämtar test medgivanden  
test == false => hämtar äkta medgivanden
```

Anropet returnerar de antal medgivanden som är redo att hämtas.

3.1.1 Fetch

För att hämta medgivanden måste *Fetch* anropas en gång för varje medgivande. Samma testmedgivanden kan hämtas hur många gånger som helst, äkta medgivanden kan bara hämtas en gång.

```
string Fetch()
```

Operationen returnerar en xml sträng. Om man försöker hämta fler medgivanden än de som finns att hämta, returneras "null".

Om något medgivande skulle gå förlorat eller av någon anledning vilja hämtas fler än en gång, kan administratören av Medgivande via hemsida-portalen gå in och återaktivera medgivandet för ytterligare hämtning. Läs mer om återaktivering av medgivande i Användarmanualen.

3.2 Exempel

3.2.1 (C#, .Net) för hämtning av medgivande

```
// Skapa session cookie
if (Session["theSession"] == null)
    Session["theSession"] = new System.Net.CookieContainer();
System.Net.CookieContainer theSession =
(System.Net.CookieContainer)Session["theSession"];

// Skapa webservice objektet
MvHWS.Downloader md = new MvHWS.Downloader();
// Spara session cookie i objektet
md.CookieContainer = theSession;

// Logga in (userName är bgnr i formatet 9999-9999)
if (md.Logon("9999-9999","password")==0)
{
    // Query
    int count = md.Query(false);
    string xmlString;
    for (int i = 0; i < count; i++)
    {
        // Hämta
        xmlString = md.Fetch();
        // Bearbetar medgivanden
        // ...
    }
}
```

3.2.1 Exempel medgivandefil

```
-<DocumentElement>
- <MedgivandeViaHemsida>
  <Formulärnamn>Betalmingsmottagarens testformular</Formulärnamn>
  <Betalmingsmottagares_x0020_namn>Storforetaget</Betalmingsmottagares_x0020_namn>
  <Betalmingsmottagares_x0020_adress>Storgatan 10</Betalmingsmottagares_x0020_adress>
  <Betalmingsmottagares_x0020_postnr>12345</Betalmingsmottagares_x0020_postnr>
  <Betalmingsmottagares_x0020_postort>Storstaden</Betalmingsmottagares_x0020_postort>
  <Bankgironr>1234-5678</Bankgironr>
  <Organisationsnr>556677-8899</Organisationsnr>
  <Autogiroanmälan_x002C__x0020_medgivande />
  <Betalares_x0020_namn>Doris Demosson</Betalares_x0020_namn>
  <Betalares_x0020_adress>Kungsgatan 10</Betalares_x0020_adress>
  <Betalares_x0020_postnr>12345</Betalares_x0020_postnr>
  <Betalares_x0020_postort>Smastaden</Betalares_x0020_postort>
  <Betalarnummer_x002F_Kundnummer>123123123</Betalarnummer_x002F_Kundnummer>
  <Kontoinnehavarens_x0020_bank>Storbanken</Kontoinnehavarens_x0020_bank>
  <Clearingnr>1234</Clearingnr>
  <Kontonr>1234567</Kontonr>
  <Kontoinnehavarens_x0020_personnr>191234-1234</Kontoinnehavarens_x0020_personnr>
  <Verifieringstid>2007-05-16 11:49:25</Verifieringstid>
  <Formulär_x0020_referens>1234-5678/MD/25</Formulär_x0020_referens>
  <Verifieringsreferens>123fbda4567890fb12b3</Verifieringsreferens>
</MedgivandeViaHemsida>
</DocumentElement>
```

3.3 Skicka medgivandeunderlag till BGC

När medgivandet är hämtat och lagrat hos er skickas ett vanligt medgivandeunderlag till BGC för registrering i BGC:s medgivanderegister enligt Layout A. Läs mer om denna fil i Autogiro Privat, Tekniska manual på www.bgc.se

4 Del 2 Verifier

4.1 Inledning

Medgivande via hemsida kan hantera dessa typer av certifikat:

- BankID
- Telia/SEB
- Nordea

För att utveckla denna tjänst krävs att du har kunskap om:

- Teknisk utveckling gällande certifikat (e-legitimationer)
- Utveckling av web-services-anrop
- Kunskap om webbutveckling och signering

4.2 Operationer

Webbtjänsten stöder 4 operationer:

Logon
StoreSignature
VerifySignature
GetSignatureInfo

4.2.1 Logon

Innan hämtning av medgivanden kan ske måste programmet logga in. *Logon* -operationen används på följande sätt för att utföra detta: (userName är bankgironummer i formatet 9999-9999). Användarnamn (userName) samt lösenord (password) skickas via ett välkomstbrev till ert företag när ni börjar använda tjänsten. Samma lösenord gäller för inloggning i portalen som vid inloggning till denna webbtjänst.

OBS! Efter 3 gånger med fel lösenord spärras möjligheten att logga in. Var därför uppmärksam på vilka koder som anropet returnerar. Vid spärrat lösenord måste BGC Kundservice kontaktas.

```
int Logon(string userName, string password)
```

Anropet returnerar en kod enligt tabellen nedan.

Kod	Beskrivning
0	Inloggningen lyckades
1	Fel användarnamn eller lösenord
2	Inloggningskontot har spärrats!
3	Lösenordet måste ändras

4.2.1 StoreSignature

StoreSignature används för att lagra signaturen och signerat data hos BGC:

```
int StoreSignature(string signature, SignatureType type, string data)
```

signatureType: specificerar signaturtyp och beror på signeringsklienten:

```
BankID thin client: bid  
Nexus Websigner: pkcs7  
Netmaker NetID: pkcs7
```

signature: beror på signaturtyp:

```
bid: base64-kodat xml signatur sträng  
pkcs7: base64-kodat 'pkcs7 detached signature' sträng  
data: base64-kodat original data
```

Noteringar:

- Nexus Websigner (och valfritt, Netmaker NetID) skapar en URL-kodat base64 signatur som måste avkodas.
- Nexus Websigner och Netmaker NetID kräver att originaldata ska vara URL-kodat. I anropen ska data inte vara URL-kodat

Utvecklare hänvisas till tekniska manualen till signeringsklienter för information om klientens användning.

Anropen returnerar en signaturID för signaturen (användas i *VerifySignature* och *GetSignatureInfo*).

4.2.1 VerifySignature

Efter att signaturen har lagrats kan den verifieras:

```
int VerifySignature(int signatureID, string personnummer)
```

Personnummer måste anges i formatet ååååmmddnnnn.

Anropen returnerar 0 om signaturen är OK annars en resultatkod (se tabell längst bak i manualen för alla koder)

4.2.1 GetSignatureInfo

För att hämta informationen om signaturen och verifikationen kan GetSignatureInfo anropas:

```
SignatureInfo GetSignatureInfo(int signatureID)
```

Anropen returnerar en signatureInfo-objekt som har följande egenskaper:

string certIssuer	certifikatets utgivare
string trustIssuer	utgivaren av certifikatet som är betrodd i systemet. (vanligtvis rootcertifikatet)
string verificationTime	den tid när den första veriferingen skedde
int verification	Result resultatet av veriferingen
int extendedError	koden som returnerades av TSS servern om verificationResult = 605
string verificationID	transaktionID som returnerades från TSS servern.
string ExternalException	om verificationResult = 101, 303, 402, 503 eller 606 (Exception), felmeddelanden från källan returneras här.

4.3 Exempel (C#, .Net)

```
// Skapa webservice objektet
MvHWS.Verifier m_verifier = new MvHWS.Verifier();
// Spara session cookie i objektet
m_verifier.CookieContainer = theSession;

// Skapa session cookie
if (Session["theSession"] == null)
    Session["theSession"] = new System.Net.CookieContainer();
System.Net.CookieContainer theSession = (System.Net.CookieContainer)Session["theSession"];

// Logga in (userName är bgnr i formatet 9999-9999)
if (m_verifier.Logon("9999-9999", "lösenord") == 0)
{
    string signature = getSignature();
    string signedData = getSignedData();

    // Lagrar en signature iform av ett BankID eller ett PKCS7 signatur.
    int signatureID = m_verifier.StoreSignature(
        signature,
        SignatureType.BankID,
        signedData);

    // Kolla om signaturen finns
    int correct = m_verifier.VerifySignature(signatureID, "191111111111");

    if (correct == 0)
    {
        //Hämtar signaturen
        SignatureInfo signatureInfo = m_verifier.GetSignatureInfo(signatureID);

        .....
        .....
        .....
    }
}
```

4.4 Felkommentarer

Klass	Kod	Beskrivning
PKIObjekt	101	Exception, för ytterligare info: se rubriken "GetSignatureInfo" i denna manual
	102	Signatur saknas/kan ej hittas
Verifier Initialisation	201	Signatur kan ej läsas
	202	Signering med BankID: Fel i signerat data eller signatur
	203	Signering med BankID: Originaldata stammer inte med signerat data
Personnummer	301	Personnummer finns inte i certifikatet
	302	Personnummer stämmer inte överens med det i certifikatet
	303	Exception, för ytterligare info: se rubriken "GetSignatureInfo" i denna manual
Trust	401	Certifikatet är inte giltigt för denna tjänst
	402	Exception, för ytterligare info: se rubriken "GetSignatureInfo" i denna manual
Signature	501	Felaktig signatur
	502	Certifikatet är inte korrekt installerat
	503	Exception, för ytterligare info: se rubriken "GetSignatureInfo" i denna manual
Certificate	601	Certifikatet är inte giltigt för denna tjänst
	602	Certifikatets giltighetstid har inte börjat gälla
	603	Certifikatets giltighetstid har gått ut
	604	Certifikatet är spärrat
	605	Övriga fel, för ytterligare info: se rubriken "GetSignatureInfo" i denna manual
	606	Exception, för ytterligare info: se rubriken "GetSignatureInfo" i denna manual

4.5 Skicka medgivandeunderlag till BGC

När medgivandet är hämtat och lagrat hos ert företag skickas ett vanligt medgivandeunderlag till BGC för registrering i BGC:s medgivanderegister enligt Layout A.

Läs mer om denna fil i Autogiro Privat, Tekniska manual på www.bgc.se