

**WCUnderwriting Web Service
Electronic Guidelines**

**Delaware Compensation Rating Bureau, Inc.
Pennsylvania Compensation Rating Bureau**

Issued: February 1, 2019

Revised: July 2021

WCUNDERWRITING ELECTRONIC GUIDELINES

A. WCUnderwriting Overview and Resources

This document is intended to explain the standard electronic procedures for carrier groups utilizing the WCUnderwriting web service product. This service provides carrier groups immediate access to experience modification factors, merit rating factors and other peripheral pricing information. The information will enable electronic commerce services designed for the validation of policy application information, policy quoting and to supplement member carriers' internal applications.

In addition to these guidelines, additional reference materials that may assist in the understanding of these data elements are the DCRB/PCRB Basic Manual and the Workers Compensation Data Specification Manuals (WCRATING and WCPOLS). These materials can be accessed at www.pcrb.com and www.wcio.org, respectively, to provide the basic definitions of the data elements in these guidelines.

B. Authentication

A User ID and password must be passed to the service for access and authentication. A User ID and password can be obtained by establishing the appropriate set-up from the DCRB/PCRB website as defined in **Section C. Web Service Set-Up**.

Access and authentication are established through the DCRB/PCRB website server ensuring only authorized users access the information. Note that the use of Secure Socket Layer (SSL) encryption protects the User ID and password.

To call or invoke the program, the carrier group must use a language and platform that supports XML running over HTTP and SOAP.

A unique User ID and Password (which does not expire) will be used for this web service product.

The SOAP header accepts User ID and Password elements for web service authentication. A User ID and Password is required in the SOAP header when web service is invoked using SOAP method, whereas User ID and Password is required in HTTP request header when web service is invoked using GET or POST method.

Required call parameters are shown in **Section H. Sample Requests and Responses**.

Basic authentication is used to connect to the web service. The authorization field must be passed in the request header. The authorization field has the following properties:

- The User ID and Password are combined with a single colon.
- The resulting string is Base64 encoded.
- The authorization method and a space are entered before the encoded string. Ex: "Basic "

An example of the authorization field is *Authorization: Basic cGNyYjEyMzQ1NjUdTE1NDdKayQh*

Sample code is shown in **Section I. C# Sample Code**.

The web service address is: <https://www.pcrbdata.com/WCUnderwriting/WCUnderwriting.asmx>

If the user is not authorized or if the User Id and/or Password are incorrect, an **HTTP 401** error will be returned with the following error message: *Credentials are not valid or account is inactive.*

C. Web Service Set-Up

Before data providers can request this service and establish a User ID and Password, a completed Carrier Group Administrator (CGA) Application must be on file.

If you are unsure if you currently have a CGA or you would like to know the name of the designated CGA within your organization, please contact DCRB/PCRB Central Support at (215) 320-4933.

Note that the WCUnderwriting web service is intended for DCRB/PCRB member carriers only and should not be shared with third party vendors and/or their applications.

It is the responsibility of the CGA to set-up the service and to establish the User ID and Password.

Once the CGA is signed into the DCRB/PCRB website, and while on the home page, the CGA must select the **Carrier Group Admin** menu item and then select the **Web Service Setup** option.



The CGA will be prompted to create a **User ID** and **Password** to be used by the web service. Once entered, the CGA must activate the **Authorized to Use** box, and then press **Update**.

After the initial set-up is complete, the web service product is ready to use.

D. WSDL

The WSDL file is available at the link below:

<https://www.pcrbdata.com/WCUnderwriting/WCUnderwriting.asmx?wsdl>

E. Request Parameters

Element	Required?	Valid Values	Length / Format	Notes
State Code	Yes	07 (Delaware) 37 (Pennsylvania)	2	
File Number (Risk ID Number)	Yes, if no FEIN provided No, if FEIN provided	Valid DCRB/PCRB file number	7	If no File Number (Risk ID), enter "0000000"
FEIN Number	Yes, if no File Number provided No, if File Number provided	Valid FEIN	9	If no FEIN Number, enter "000000000"
Start Date	No (See Notes)		8 YYYYMMDD	If not provided, the most current values will be returned
End Date	No (See Notes)		8 YYYYMMDD	If not provided, the most current values will be returned
User ID	Yes	Assigned by CGA	20	Does not expire
Password	Yes	Assigned by CGA	20	Does not expire

F. Returned Parameters

Element	Returned?	Values	Length / Format	Notes
Response Code	Yes	0 = Successful 1 = Data Not Found 2 = Error in parameters	1	See Response Message Notes
Response Message	Yes, if applicable	See <i>Section G. Response Message Description</i>		If Response Code = 0, then no message returned. If Response Code = 1 or 2, then a message with problem description returned
State Code	Yes	07 (Delaware) 37 (Pennsylvania)	2	
Risk ID Number	Yes	Valid DCRB/PCRB file number	7	File number
Name of Insured	Yes		72	Primary named insured
Federal Employer Identification Number (FEIN)	Yes	Valid FEIN number	9	
Rating Effective Date	Yes		8 YYYYMMDD	Rating Effective Date for each split period
Rating Issue Date	Yes, if applicable		8 YYYYMMDD	Rating Issue Date for each split period
Status of Rate Filing Code	Yes	F = Final	1	This field will always be returned as "F" (Final)
Contingent Indicator	Yes	Y = Yes N = No	1	If Y, this contingent experience modification was calculated using available information and is provided for informational purposes. A complete final rating will be issued upon receipt of specific data.

Experience Modification Factor	Yes, if applicable		5 xx.xxx	Experience Modification Factor for each split period
Merit Rating Factor	Yes, if applicable		5 xx.xxx	Merit Rating Factor for each split period
Workplace Safety Credit Percentage	Yes, if applicable		5 xx.xxx	Workplace Safety Credit for each split period
Construction Classification Premium Credit Percentage	Yes, if applicable		5 xx.xxx	Construction Credit for each split period

G. Response Message Description

If the Response Code equals “1” or “2”, a message with the error description will also be returned.

Sample Response Messages:

- *No data. Unable to retrieve information: missing key fields*
- *Invalid State Code (This is a numeric field)*
- *Invalid State Code. Valid state: 07, 37*
- *Invalid Risk ID (This is a numeric field)*
- *Invalid Risk ID (Maximum 7 digits field)*
- *FEIN must be numeric*
- *Invalid FEIN Number (Maximum 9 digits field)*
- *Invalid Start Date Format (yyyymmdd)*
- *Invalid End Date Format (yyyymmdd)*
- *Starting and Ending dates cannot have the same value.*
- *Invalid Starting date. (Starting date must be less than Ending date)*
- *Invalid Starting Date (Maximum 8 digits field)*
- *Invalid Ending Date (Maximum 8 digits field)*
- *Invalid Starting Date (This is a numeric field)*
- *Invalid Ending Date (This is a numeric field)*
- *Authorization Failed.*
- *Status Code 401: Credentials are not valid or account is inactive.*

H. Sample Requests and Responses

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The **placeholders** are replaced with actual values.

```
POST /WCUnderwriting/WCUnderwriting.asmx HTTP/1.1
Host: www.pcrbdata.com
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "SubmissionInfo/GetWCUnderwriting"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <WCUnderwritingHeader xmlns="SubmissionInfo">
      <UserId>string</UserId>
      <Password>string</Password>
    </WCUnderwritingHeader>
  </soap:Header>
  <soap:Body>
    <GetWCUnderwriting xmlns="SubmissionInfo">
      <StateCode>string</StateCode>
      <FileNumber>string</FileNumber>
      <FeinNumber>string</FeinNumber>
      <StartDate>string</StartDate>
      <EndDate>string</EndDate>
    </GetWCUnderwriting>
  </soap:Body>
</soap:Envelope>
```

```

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetWCUnderwritingResponse xmlns="SubmissionInfo">
      <GetWCUnderwritingResult xmlns="https://www.wcio.org/XML/WCUNDERWRITING">
        <WCUnderwriting>
          <Names>
            <Name xsi:nil="true" />
            <Name xsi:nil="true" />
          </Names>
          <ResponseCode>int</ResponseCode>
          <ResponseMessage>token</ResponseMessage>
          <StateCode>int</StateCode>
          <RiskIdNumber>token</RiskIdNumber>
          <FederalEmployerIdentificationNumber>token</FederalEmployerIdentificationNumber>
          <SearchStartingDate>date</SearchStartingDate>
          <SearchEndingDate>date</SearchEndingDate>
          <UnderwritingsInformation>
            <UnderwritingInformation xsi:nil="true" />
            <UnderwritingInformation xsi:nil="true" />
          </UnderwritingsInformation>
        </WCUnderwriting>
      </GetWCUnderwritingResult>
    </GetWCUnderwritingResponse>
  </soap:Body>
</soap:Envelope>

```

SOAP 1.2

The following is a sample SOAP 1.2 request and response. The **placeholders** are replaced with actual values.

```

POST /WCUnderwriting/WCUnderwriting.asmx HTTP/1.1
Host: www.pcrbdata.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-
envelope">
  <soap12:Header>
    <WCUnderwritingHeader xmlns="SubmissionInfo">
      <UserId>string</UserId>
      <Password>string</Password>
    </WCUnderwritingHeader>
  </soap12:Header>
  <soap12:Body>
    <GetWCUnderwriting xmlns="SubmissionInfo">
      <StateCode>string</StateCode>
      <FileNumber>string</FileNumber>
      <FeinNumber>string</FeinNumber>
      <StartDate>string</StartDate>
      <EndDate>string</EndDate>
    </GetWCUnderwriting>
  </soap12:Body>
</soap12:Envelope>

```



```

</GetWCUnderwriting>
</soap12:Body>
</soap12:Envelope>

```

```

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

```

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetWCUnderwritingResponse xmlns="SubmissionInfo">
      <GetWCUnderwritingResult xmlns="https://www.wcio.org/XML/WCUNDERWRITING">
        <WCUnderwriting>
          <Names>
            <Name xsi:nil="true" />
            <Name xsi:nil="true" />
          </Names>
          <ResponseCode>int</ResponseCode>
          <ResponseMessage>token</ResponseMessage>
          <StateCode>int</StateCode>
          <RiskIdNumber>token</RiskIdNumber>
          <FederalEmployerIdentificationNumber>token</FederalEmployerIdentificationNumber>
          <SearchStartingDate>date</SearchStartingDate>
          <SearchEndingDate>date</SearchEndingDate>
          <UnderwritingsInformation>
            <UnderwritingInformation xsi:nil="true" />
            <UnderwritingInformation xsi:nil="true" />
          </UnderwritingsInformation>
        </WCUnderwriting>
      </GetWCUnderwritingResult>
    </GetWCUnderwritingResponse>
  </soap:Body>
</soap:Envelope>

```

HTTP GET

The following is a sample HTTP GET request and response. The **placeholders** are replaced with actual values.

```

GET
/WCUnderwriting/WCUnderwriting.asmx/GetWCUnderwriting?StateCode=string&FileNumber=string&FeinNumber=string&StartDate=string&EndDate=string HTTP/1.1
Host: www.pcrbdata.com

```

```

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8"?>
<Submission xmlns="https://www.wcio.org/XML/WCUNDERWRITING">
  <WCUnderwriting>
    <Names>
      <Name nameReference="ID">
        <NameOfInsured>token</NameOfInsured>
        <FederalEmployerIdentificationNumber>token</FederalEmployerIdentificationNumber>

```

```

</Name>
</Names>
<ResponseCode>int</ResponseCode>
<ResponseMessage>token</ResponseMessage>
<StateCode>int</StateCode>
<RiskIdNumber>token</RiskIdNumber>
<FederalEmployerIdentificationNumber>token</FederalEmployerIdentificationNumber>
<SearchStartingDate>date</SearchStartingDate>
<SearchEndingDate>date</SearchEndingDate>
<UnderwritingsInformation>
  <UnderwritingInformation>
    <RatingEffectiveDate>date</RatingEffectiveDate>
    <RatingIssueDate>date</RatingIssueDate>
    <StatusOfRateFilingCode>F</StatusOfRateFilingCode>
    <ContingentIndicator>Y or N</ContingentIndicator>
    <ExperienceModificationFactor>decimal</ExperienceModificationFactor>
    <MeritRatingFactor>decimal</MeritRatingFactor>
    <WorkplaceSafetyCreditPercentage>decimal</WorkplaceSafetyCreditPercentage>
  <ConstructionClassificationPremiumCreditPercentage>decimal</ConstructionClassificationPremiumCr
editPercentage>
  </UnderwritingInformation>
</UnderwritingsInformation>
</WCUnderwriting>
</Submission>

```

HTTP POST

The following is a sample HTTP POST request and response. The **placeholders** are replaced with actual values.

```

POST /WCUnderwriting/WCUnderwriting.asmx/GetWCUnderwriting HTTP/1.1
Host: www.pcrbdata.com
Content-Type: application/x-www-form-urlencoded
Content-Length: length
StateCode=string&FileNumber=string&FeinNumber=string&StartDate=string&EndDate=string

```

```

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8"?>
<Submission xmlns="https://www.wcio.org/XML/WCUNDERWRITING">
  <WCUnderwriting>
    <Names>
      <Name nameReference="ID">
        <NameOfInsured>token</NameOfInsured>
        <FederalEmployerIdentificationNumber>token</FederalEmployerIdentificationNumber>
      </Name>
    </Names>
    <ResponseCode>int</ResponseCode>
    <ResponseMessage>token</ResponseMessage>
    <StateCode>int</StateCode>
    <RiskIdNumber>token</RiskIdNumber>
    <FederalEmployerIdentificationNumber>token</FederalEmployerIdentificationNumber>
    <SearchStartingDate>date</SearchStartingDate>
    <SearchEndingDate>date</SearchEndingDate>
    <UnderwritingsInformation>
      <UnderwritingInformation>
        <RatingEffectiveDate>date</RatingEffectiveDate>
        <RatingIssueDate>date</RatingIssueDate>
        <StatusOfRateFilingCode>F</StatusOfRateFilingCode>
        <ContingentIndicator>Y or N</ContingentIndicator>
      </UnderwritingInformation>
    </UnderwritingsInformation>
  </WCUnderwriting>
</Submission>

```

```

    <ExperienceModificationFactor>decimal</ExperienceModificationFactor>
    <MeritRatingFactor>decimal</MeritRatingFactor>
    <WorkplaceSafetyCreditPercentage>decimal</WorkplaceSafetyCreditPercentage>
<ConstructionClassificationPremiumCreditPercentage>decimal</ConstructionClassificationPremiumCr
editPercentage>
    </UnderwritingInformation>
  </UnderwritingsInformation>
</WCUnderwriting>
</Submission>

```

Sample Responses with Data

If there is no data found, a Response Code of "1" is returned with a message description.

```

<?xml version="1.0" encoding="utf-8"?>
<Submission xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="https://www.wcio.org/XML/WCUNDERWRITING">
  <WCUnderwriting>
    <ResponseCode>1</ResponseCode>
    <ResponseMessage>DATA NOT FOUND</ResponseMessage>
    <StateCode>37</StateCode>
    <RiskIdNumber>603</RiskIdNumber>
<FederalEmployerIdentificationNumber>00000000</FederalEmployerIdentificationNumber>
  </WCUnderwriting>
</Submission>

```

If any of the required call parameters are not valid, a Response Code of "2" is returned with a message description. In this example, *State Code* was not valid.

```

<?xml version="1.0" encoding="utf-8"?>
<Submission xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="https://www.wcio.org/XML/WCUNDERWRITING">
  <WCUnderwriting>
    <ResponseCode>2</ResponseCode>
    <ResponseMessage>Invalid State Code. Valid state: 07, 37</ResponseMessage>
  </WCUnderwriting>
</Submission>

```

Example of a risk with a single experience modification and workplace safety credit:

```

<?xml version="1.0" encoding="utf-8"?>
<Submission xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="https://www.wcio.org/XML/WCUNDERWRITING">
  <WCUnderwriting>
    <Names>
      <Name nameReference="Name1">
        <NameOfInsured>CONFINED SPACE INDUSTRIAL RESCUE SERVICES INC</NameOfInsured>
        <FederalEmployerIdentificationNumber>273190547</FederalEmployerIdentificationNumber>
      </Name>
    </Names>
    <ResponseCode>0</ResponseCode>
    <ResponseMessage />
    <StateCode>37</StateCode>

```

```

<RiskIdNumber>3296225</RiskIdNumber>
<FederalEmployerIdentificationNumber>273190547</FederalEmployerIdentificationNumber>
<Insured>
  <Names>
    <Name nameReference="Name1" />
  </Names>
</Insured>
<SearchStartingDate>2008-01-19</SearchStartingDate>
<SearchEndingDate>2017-10-31</SearchEndingDate>
<UnderwritingsInformation>
  <UnderwritingInformation>
    <RatingEffectiveDate>2017-10-30</RatingEffectiveDate>
    <RatingIssueDate>2017-05-22</RatingIssueDate>
    <StatusOfRateFilingCode>F</StatusOfRateFilingCode>
    <ContingentIndicator>N</ContingentIndicator>
    <ExperienceModificationFactor>0</ExperienceModificationFactor>
    <WorkplaceSafetyCreditPercentage>0.05</WorkplaceSafetyCreditPercentage>
  </UnderwritingInformation>
</UnderwritingsInformation>
</WCUnderwriting>

```

The following example shows a risk with a split effective date. If the requested risk has a split period, all pertinent rating information will be returned.

```

<?xml version="1.0" encoding="utf-8"?>
<Submission xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="https://www.wcio.org/XML/WCUNDERWRITING">
  <WCUnderwriting>
    <Names>
      <Name nameReference="Name1">
        <NameOfInsured>EAST PHILADELPHIA FURNITURE SERVICES INC</NameOfInsured>
      </Name>
    </Names>
    <FederalEmployerIdentificationNumber>261997227</FederalEmployerIdentificationNumber>
    <ResponseCode>0</ResponseCode>
    <ResponseMessage />
    <StateCode>37</StateCode>
    <RiskIdNumber>3352946</RiskIdNumber>
    <FederalEmployerIdentificationNumber>261997227</FederalEmployerIdentificationNumber>
    <UnderwritingsInformation>
      <UnderwritingInformation>
        <RatingEffectiveDate>2018-06-26</RatingEffectiveDate>
        <RatingIssueDate>2018-03-06</RatingIssueDate>
        <StatusOfRateFilingCode>F</StatusOfRateFilingCode>
        <ContingentIndicator>N</ContingentIndicator>
        <ExperienceModificationFactor>0.758</ExperienceModificationFactor>
      </UnderwritingInformation>
      <UnderwritingInformation>
        <RatingEffectiveDate>2018-01-16</RatingEffectiveDate>
        <RatingIssueDate>2017-10-05</RatingIssueDate>
        <StatusOfRateFilingCode>F</StatusOfRateFilingCode>
        <ContingentIndicator>N</ContingentIndicator>
        <ExperienceModificationFactor>0.75</ExperienceModificationFactor>
      </UnderwritingInformation>
      <UnderwritingInformation>
        <RatingEffectiveDate>2017-01-16</RatingEffectiveDate>
        <RatingIssueDate>2017-09-20</RatingIssueDate>
        <StatusOfRateFilingCode>F</StatusOfRateFilingCode>
        <ContingentIndicator>N</ContingentIndicator>
      </UnderwritingInformation>
    </UnderwritingsInformation>
  </WCUnderwriting>
</Submission>

```

```

    <ExperienceModificationFactor>0.877</ExperienceModificationFactor>
  </UnderwritingInformation>
</UnderwritingsInformation>
</WCUnderwriting>

```

I. C# Sample Code

The sample code below targets .NET Framework 4.5.

```

Using System;
Using System.Net.Http;
Using System.Net.Http.Headers;
Using System.Text;
Using System.Threading.Tasks;

namespace ClientWebService
{
    class ClientService
    {
        static void Main(string[] args)
        {
            //Parameters - change text in red
            string username = " User Name ";
            string password = " Password ";
            string statecode = " State Code ";

            SearchPADEAsync(statecode,username,password,searchParameters).wait();

        }

        public async Task<string> SearchPADEAsync(string stateCode, string userName, string password, SearchParameters
searchParameters)
        {
            string baseAddress = "https://www.pcrbdata.com/WCUnderwriting/WCUnderwriting.asmx/";

            string requestUri =
string.Format("GetWCUnderwriting?statecode={0}&filenumber={1}&feinnumber={2}&startdate={3}&enddate={4}",
                stateCode, searchParameters.RiskID, searchParameters.FEIN, searchParameters.StartDate,
searchParameters.EndDate);

            string result = string.Empty;
            using (var client = new HttpClient())
            {
                client.BaseAddress = new Uri(baseAddress);

                //Base64 encode credentials.Add Authorization header to request.
                byte[] credentials = Encoding.ASCII.GetBytes(userName + ":" + password);
                client.DefaultRequestHeaders.Authorization = new AuthenticationHeaderValue("Basic",
Convert.ToBase64String(credentials));

                // HTTP GET - Execute the operation passed in
                HttpResponseMessage response = await client.GetAsync(requestUri);
                if (response.IsSuccessStatusCode)
                {
                    // Call was successful. A WCUnderwriting XML is passed in response body.
                    result = await response.Content.ReadAsStringAsync();
                }
                else
                {
                    // An error occurred.
                    result = await response.Content.ReadAsStringAsync();
                }
            }

            return result;
        }
    }
}

```