

Automatic cardtype detection – with VISA/Mastercard and some debug

This is a very simple example to show how to configure PAYONE hosted-iFrame mode:

- without cardtype-selection cardtype selection
- with automatic cardtype detection enabled
- with logos enabled to show the user the detected cardtype

Simply change:

- config.fields.*.style to your CSS layout
- request.*-fields to your mid, aid, portalid, mode, encoding and hash value
- set array "supportedCardtypes" to your cardtypes you'd like to process
- callback.function to enable / disable your logos if you want to

and finally

- initiate a preauthorization / authorization via received "pseudocardpan".

```
<!DOCTYPE html>
<html>
  <head lang="en">
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <link href="static/static/client_api.css" rel="stylesheet" type="text/css" media="screen" />
    <link rel="shortcut icon" type="image/x-icon" href="static/static/favicon.ico" />
    <meta charset="UTF-8">
    <title>Client API with the new hosted iFrames example</title>
  </head>
  <body>
    <h1>PAYONE Client API with new hosted iFrames example</h1>
    <form name="paymentform" action="#" method="post">
      <fieldset>

        <input type="hidden" name="pseudocardpan" id="pseudocardpan">
        <input type="hidden" name="truncatedcardpan" id="truncatedcardpan">
        <input type="hidden" name="cardtypeResponse" id="cardtypeResponse">
        <input type="hidden" name="cardexpiredateResponse" id="cardexpiredateResponse">

         <br />

        <label for="cardpanInput">Cardpan:</label>
        <span class="inputIframe" id="cardpan"></span> <!-- Container for PAYONE-Script -> PAYONE will place an
iFrame in here -->

        <label for="cvcInput">CVC:</label>
        <span class="inputIframe" id="cardcvc2"></span>

        <label for="expireInput">Expire date (mm/yyyy):</label>
        <span class="inputIframe" id="expireInput">
          <span id="cardexpiremonth"></span>
          <span id="cardexpireyear"></span>
        </span>

        <div id="error"></div>
        <br />
        <input id="submit" type="button" value="Run credit card check" />
        <input id="submitWithOutCompleteCheck" type="button" value="Run CCC (w/o complete check)" />
      </fieldset>
    </form>
    <h2>The JSON received from the server</h2>
    <div id="jsonResponse"><pre id="jsonResponsePre">Nothing received yet.</pre></div>
    <h2>Autodetection callback result</h2>
    <div id="autodetectionResponse"><pre id="autodetectionResponsePre">Nothing received yet.</pre></div>

    <script>
      var request,
```

```

supportedCardtypes = ["V", "M"],
config = {
  fields: {
    cardpan: {
      selector: "cardpan",
      style: "font-size: 14px; border: 1px solid #000;",
      type: "input"
    },
    cardcvc2: {
      selector: "cardcvc2",
      type: "password", // Could be "text" as well.
      style: "font-size: 14px; border: 1px solid #000;",
      size: "4",
      maxlength: "4",
      length: { "V": 3, "M": 3 } // enforce 3 digit CVC für VISA and Mastercard
    },
    cardexpiremonth: {
      selector: "cardexpiremonth",
      type: "text", // select (default), text, tel
      size: "2",
      maxlength: "2",
      iframe: {
        width: "40px"
      },
      style: "font-size: 14px; width: 30px; border: solid 1px #000; height: 22px;"
    },
    cardexpireyear: {
      selector: "cardexpireyear",
      type: "text", // select (default), text, tel
      iframe: {
        width: "60px"
      },
      style: "font-size: 14px; width: 50px; border: solid 1px #000; height: 22px;"
    }
  },
  defaultStyle: {
    input: "font-size: 1em; border: 1px solid #000; width: 175px;",
    select: "font-size: 1em; border: 1px solid #000;",
    iframe: {
      height: "22px",
      width: "180px"
    }
  },
  autoCardtypeDetection: {
    supportedCardtypes: supportedCardtypes,
    callback: function(detectedCardtype) {
      // For the output container below.
      document.getElementById('autodetectionResponsePre').innerHTML = detectedCardtype;

      if (detectedCardtype === 'V') {
        document.getElementById('visa').style.borderColor = '#00F';
        document.getElementById('mastercard').style.borderColor = '#FFF';
      } else if (detectedCardtype === 'M') {
        document.getElementById('visa').style.borderColor = '#FFF';
        document.getElementById('mastercard').style.borderColor = '#00F';
      } else {
        document.getElementById('visa').style.borderColor = '#FFF';
        document.getElementById('mastercard').style.borderColor = '#FFF';
      }
    }
  },
  // deactivate: true // To turn off automatic card type detection.
},
  language: Payone.ClientApi.Language.de, //, // Language to display error-messages (default: Payone.
ClientApi.Language.en)
  error: "error" // area to display error-messages (optional)
};

request = {
  request: 'creditcardcheck', // fixed value

```

```

    responsetype: 'JSON', // fixed value
    mode: 'live', // desired mode
    mid: '10001', // your MID
    aid: '10002', // your AID
    portalid: '2000002', // your PortalId
    encoding: 'UTF-8', // desired encoding
    storecarddata: 'yes', // fixed value
    // hash calculated over your request-parameter-values (alphabetical request-order) plus PMI portal key
    // hash:
'1cf456bf692453613ebb992a3fb859cc347ddc7e94e2ca764efbe8b0089de6964ab1266df0831e59de89dc5291070fe7'
    hash: '5c4014cebeb361d9e186fd42c810b9b1' // see Chapter 3.1.5.3
};
var iframes = new Payone.ClientApi.HostedIFrames(config, request);

// Function called by submitting PAY-button
function pay() {

    if (iframes.isComplete()) {
        // Perform "CreditCardCheck" to create and get a PseudoCardPan; then call your function
"payCallback"
        iframes.creditCardCheck('payCallback');
    } else {
        alert("Not complete. Nothing done.");
    }
}

document.getElementById("submit").onclick = function () {
    pay();
};

document.getElementById("submitWithOutCompleteCheck").onclick = function () {
    iframes.creditCardCheck('payCallback');
};

function payCallback(response) {

    console.debug(response);
    if (response.status === "VALID") {
        document.getElementById("pseudocardpan").value = response.pseudocardpan;
        document.getElementById("truncatedcardpan").value = response.truncatedcardpan;
        document.getElementById("cardtypeResponse").value = response.cardtype;
        document.getElementById("cardexpiredateResponse").value = response.cardexpiredate;
    }

    if (typeof response === 'object') {

        var responseAsString = 'time: ' + new Date().getTime() + "\n";
        for (var key in response) {

            if (response.hasOwnProperty(key)) {
                responseAsString += key + ': ' + response[key] + "\n";
            }
        }

        document.getElementById('jsonResponsePre').innerHTML = responseAsString;
    }
}

</script>
</body>
</html>

```