

dtz<?php

```
// adresy pro server PayU a metodu Payment/get
// addresses for PayU server and the Payment/get method
$server = "secure.payu.com";
$server_script = "/paygw/ISO/Payment/get";

// parametry pozadovane pro odeslani pozadavku
// parameters required for checking and signing request
define("PAYU_POS_ID", 123);
define("PAYU_KEY1", "1234567890123456");
define("PAYU_KEY2", "9123456789012345");

// vraci pole s indexy:
// "code" (cislo statusu transakce nebo false v pripade chyby), "message" (popis statusu transakce nebo popis chyby)
// returns array with values:
// "code" (numerical transaction status or false in case of error), "message" (status description or error description)
function get_status($parts)
{
    // chybne cislo POS ID v odpovedi
    // incorrect POS ID number specified in response
    if($parts[1] != PAYU_POS_ID)
        return array("code" => false, "message" => "incorrect POS number");

    // vypocet podpisu pro porovnaní se sig odeslanym ze strany PayU
    // calculating signature for comparison with sig sent by PayU
    $sig = md5($parts[1].$parts[2].$parts[3].$parts[5].$parts[4].$parts[6].$parts[7].PAYU_KEY2);

    // chybný podpis v odpovedi v porovnaní s podpisem spocitanzm lokálne
    // incorrect signature in response in comparison to locally calculated one
    if($parts[8] != $sig)
        return array("code" => false, "message" => "incorrect signature");

    // ruzne zpravy dle statusu transakce. Popisy jednotlivých statusu jsou uvedeny v technické dokumentaci
    // different messages depending on transaction status. For status description, see documentation
    switch($parts[5])
    {
        case 1: return array("code" => $parts[5], "message" => "new");
        case 2: return array("code" => $parts[5], "message" => "cancelled");
        case 3: return array("code" => $parts[5], "message" => "rejected");
        case 4: return array("code" => $parts[5], "message" => "started");
        case 5: return array("code" => $parts[5], "message" => "awaiting receipt");
        case 6: return array("code" => $parts[5], "message" => "no authorization");
        case 7: return array("code" => $parts[5], "message" => "payment rejected");
        case 99: return array("code" => $parts[5], "message" => "payment received - ended");
        case 888: return array("code" => $parts[5], "message" => "incorrect status");
        default: return array("code" => false, "message" => "no status");
    }
}

// nektere parametry chybeji
// some parameters are missing
if(!isset($_POST["pos_id"]) || !isset($_POST["session_id"]) || !isset($_POST["ts"]) || !isset($_POST["sig"]))
    die("ERROR: EMPTY PARAMETERS");

// obdrzene cislo POS ID je jine, nez bylo ocekavano
// received POS ID is different than expected
if($_POST["pos_id"] != PAYU_POS_ID)
    die("ERROR: INCORRECT POS ID");

// verifikace obdrzeného podpisu
// verification of received signature
$sig = md5($_POST["pos_id"].$_POST["session_id"].$_POST["ts"].PAYU_KEY2);

// chybný podpis
// incorrect signature
if($_POST["sig"] != $sig)
    die("ERROR: INCORRECT SIGNATURE");

// podpis, který bude odeslan do PayU spolu s požadavkem
// signature that will be sent to PayU with request
$ts = time();
$sig = md5(PAYU_POS_ID. $_POST["session_id"]. $ts.PAYU_KEY1);

// priprava retezce (string) parametru k odeslani do PayU
// preparing parameters string to be sent to PayU
$parameters = "pos_id=".PAYU_POS_ID."&session_id=".$_POST["session_id"]."&ts=".$ts."&sig=".$sig;

// urceni metodu spojeni (socket nebo CURL)
// determining connection method (socket or CURL)
$fsocket = false;
$curl = false;
if((PHP_VERSION >= 4.3) && ($fp = @fsockopen("ssl://".$server, 443, $errno, $errstr, 30)))
    $fsocket = true;
elseif (function_exists("curl_exec"))
    $curl = true;
```

```

// odesilani pozadavku pomoci socket
// sending request via socket
if ($socket == true)
{
    $header = "POST ".$server_script." HTTP/1.0"."\\r\\n"."Host: ".$server."\\r\\n".
        "Content-Type: application/x-www-form-urlencoded"."\\r\\n"."Content-Length: ".
        strlen($parameters)."\\r\\n"."Connection: close"."\\r\\n\\r\\n";
    @fputs($fp, $header.$parameters);
    $payu_response = "";
    while (!@feof($fp))
    {
        $res = @fgets($fp, 1024);
        $payu_response .= $res;
    }
    @fclose($fp);
}

// odesilani pozadavku pomoci CURL
// sending request via CURL
elseif ($curl == true)
{
    $ch = curl_init();
    curl_setopt($ch, CURLOPT_URL, "https://".$server.$server_script);
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($ch, CURLOPT_HEADER, 0);
    curl_setopt($ch, CURLOPT_TIMEOUT, 20);
    curl_setopt($ch, CURLOPT_POST, 1);
    curl_setopt($ch, CURLOPT_POSTFIELDS, $parameters);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
    $payu_response = curl_exec($ch);
    curl_close($ch);
}

// neni k dispozici zadna pouzitelna metoda spojeni
// no connection method available
else
    die("ERROR: No connect method ...\\n");

// ziskavani odpovedi od PayU
// parsing PayU response
$result = false;
if (preg_match("/<trans>.*<pos_id>([0-9]*)</pos_id>.*<session_id>(.*?)</session_id>.*<order_id>(.*?)</order_id>.*".
    "<amount>([0-9]*)</amount>.*<status>([0-9]*)</status>.*<desc>(.*?)</desc>.*<ts>([0-9]*)</ts>.*<sig>([a-z0-9]*)".
    "</sig>.*</trans>/is", $payu_response, $parts))
    $result = get_status($parts);

// rozpoznany status transakce
// recognised status of transaction
if ($result["code"])
{
    $pos_id = $parts[1];
    $session_id = $parts[2];
    $order_id = $parts[3];
    $amount = $parts[4];           // v halerich (in hellers)
    $status = $parts[5];
    $desc = $parts[6];
    $ts = $parts[7];
    $sig = $parts[8];

    // TODO:
    // zmena statusu transakce v systemu shopu
    // change of transaction status in system of the shop

    /* naprikklad (for instance):
    if ($result["code"] == "99")
    {
        if(money_are_on_the_account)
        {
            // platba je uspesna, takLze posilAAme zpAAtky OK
            // payment sucessful so we send back OK
            echo "OK";
            exit;
        }
    }
    else if ($result["code"] == "2")
    {
        // transakce zrusena, muzeme rovnez transakci zrusit
        // transaction cancelled, we can also cancell transaction
    }
    else
    {
        // jine akce
        // other actions
    }
    */

    // pokud jsou vsechny operace ukoncene, posilame nazpet OK, v opacnem pripade vygenerujeme error

```

```
// if all operations are done then we send back OK, in other case we generate error
// if (everything_ok)
// {
    echo "OK";
    exit;
// } else {
//
// }
}
else
{
    // TODO:
    // sprava plateb se statusem error
    // error transaction status managment
    echo "ERROR: Data error ....\n";
    echo "code=".$result["code"]." message=".$result["message"]."\n";
    echo $payu_response;
    // informace o zmene statusu bude z secure.payu.com odeslana znovu, muzeme zapsat informaci do logu (logs)....
    // information about changing a status will be send again from secure.payu.com, we can write information to logs....
}
?>
```