Ê · Z e Ä ¼ ^]

Ê f ¿ € f À Ë Y d y ½ Y Z { € | a ¿ € Ã Ë Z ³ • a { d Œ Ÿ € | Ë Â | f » ¶ À a

{ Z ³ • Z ‡ Z a ® ¿ Z] ® Ì ¿ Á € f ° · Y d y Y { € a

½ Z ³ | ¿ € Ë ~ a d Ë € Ë |

پذیرندگان سایت پرداخت اینترنتی شرکت پرداخت الکترونیک بانک پاسارگاد با مراجعه به آدرس https:// pep.shaparak.ir/Merchant می‌توانند از پنل مدیریت پذیرندگان استفاده کنند. صفحه ورود به پنل در تصویر ۱ نشان داده شده است.

تصویر ۱

پس از ورود به پنل، فروشنده می‌تواند از سرویس های زیر بهره بگیرد:

ZÅ ŠÀ¯Y€e [Zˆu ●•Â• d§Z

فروشنده می تواند با وارد کردن شماره ترمینال در یک بازه زمانی مشخص، صورت حساب (مبلغ کارمزد هر تراکنش، مبلغ واریزی به حساب فروشنده، زمان تسویه و ...) تراکنش های ارسالی را مشاهده کند. همچنین می‌تواند در صورت نیاز برای محدود کردن نتیجه جستجو از فیلترهایی مانند کمترین و بیشترین مبلغ تراکنش، شماره کارت و بانک صادر کننده کارت و زمان تسویه نیز استفاده نماید.(تصویر ۲)

تصویر ۲

ZÅ ŠÀ¯Y€e É€Ì Ìa

- بر اساس اطلاعات فاکتور

توسط صفحه پیگیری بر اساس اطلاعات فاکتور، فروشنده می‌تواند از دو طریق گزارش تراکنش های خود را مشاهده کند.(تصویر ۳)

۱. بر اساس شماره ارجاع داخلی
۲. بر اساس شماره و تاریخ فاکتور

تصویر ۳

- بر اساس تاریخ

توسط صفحه پیگیری بر اساس تاریخ فروشنده می تواند در یک بازه زمانی مشخص و در صورت لزوم با استفاده از فیلترهایی مانند: نوع، کمترین و بیشترین مبلغ تراکنش و نتیجه ، گزارش تراکنش های ارسالی خود را مشاهده کند.(تصویر ۴)

تصویر ۴

http web request ª Ë € — • Y ½ Z ³ | ¿ € Ë ‡ ‰ Á Ë Ë ‡ € Ä W Y ¶ Á

– ^ e € » ¦ Ë • Z e

- **requestKind** مشخص می‌کند که فروشنده درخواست دریافت اطلاعات مربوط به زمان تسویه و کارمزد تراکنش ها را دارد یا خواستار دریافت ریز اطلاعات تراکنش ها می‌باشد و یا مایل است گزارش تراکنش های یک پایانه فروش را دریافت کند. برای درخواست اول کد ۱ و برای درخواست دوم کد ۲ و برای درخواست سوم کد۳ درنظرگرفته شده است.
- StartDate تاریخ ابتدای بازه‌ی زمانی می باشد (به صورت شمسی) .
- EndDate تاریخ انتهای بازه‌ی زمانی می‌باشد (به صورت شمسی).
- MinAmount کمترین مبلغ تراکنش به ریال می‌باشد .
- MaxAmount بیشترین مبلغ تراکنش به ریال می‌باشد.
- CardPan شماره کارتی که تراکنش با آن انجام شده است.
- DesBankID شماره شناسایی بانک صادر کننده کارت است. این شماره شش رقم اول شماره کارت می‌باشد.
- StartSettleDate تاریخ ابتدای بازه‌ی زمانی تسویه تراکنش ها می باشد (به صورت شمسی).
- EndSettleDate تاریخ انتهای بازه ی زمانی تسویه تراکنش ها می باشد (به صورت شمسی).

- ResultType می‌تواند دارای مقدار true برای دریافت تراکنش‌های موفق و مقدار false برای دریافت تراکنش‌های ناموفق باشد. در صورتی که ResultType دارای مقدار نباشد همه‌ی تراکنش‌های موفق و ناموفق در بازه‌ی زمانی فرستاده شده مد نظر قرار میگیرند.
- Action می‌تواند دارای مقدار ۱۰۰۳ برای دریافت تراکنش‌های خرید و مقدار ۱۰۰۴ برای دریافت تراکنش های برگشت از خرید باشد.

فروشنده می‌تواند سرویس های ذکر شده در قسمت پنل مدیریت پذیرندگان را از طریق httpwebrequest نیز دریافت کند. سایت فروشنده برای دریافت اطلاعات تراکنش ها در یک بازه‌ی زمانی اطلاعات مربوطه را با PrivateKey خود امضا کرده و به صورت Post به سایت بانک (https://pep.shaparak.ir/Merchant/GetTransactionList.aspx) ارسال می‌کند.

۱. اگر سایت فروشنده خواستار دریافت اطلاعات مربوط به کارمزد تراکنش ها باشد، مواردی که به صورت POST به وب سایت بانک ارسال می‌شوند عبارتند از:
   - RequestKind
   - MerchantCode
   - TerminalCode
   - StartDate
   - EndDate
   - MinAmount
   - MaxAmount
   - TimeStamp
   - CardPan
   - DesBankID
   - StartSettleDate
   - EndSettleDate
   - Digital Signature

   برای تولید امضای دیجیتال در این نوع درخواست نحوه اتصال داده ها به صورت زیر می باشد:

#requestKind#merchantCode#terminalCode#startDate#endDate#minAmount#maxAmount#timeStamp#cardPan#desBankID#startSettleDate#endSettleDate#

۲. اگر سایت فروشنده خواستار دریافت لیستی از تراکنش های موفق و ناموفق در بازه ی زمانی مشخصی باشد مواردی که به صورت post به وب سایت بانک ارسال می شوند عبارتند از:

- RequestKind
- TerminalCode
- MerchantCode
- StartDate
- EndDate
- MinAmount
- MaxAmount
- TimeStamp
- action
- ResultType
- Digital Signature

برای تولید امضای دیجیتال در این نوع درخواست نحوه اتصال داده ها به صورت زیر می باشد:

#requestKind#merchantCode#terminalCode#startDate#endDate#minAmount#maxAmount#timeStamp#action#resultType#

۳. اگر سایت فروشنده خواستار دریافت گزارش تراکنش های یک پایانه فروش باشد مواردی که به صورت post به وب سایت بانک ارسال می شوند عبارتند از:

- RequestKind
- TerminalCode
- MerchantCode
- StartDate
- EndDate
- MinAmount
- MaxAmount
- TimeStamp
- CardPan
- DesBankID
- Digital Signature

برای تولید امضای دیجیتال در این نوع درخواست نحوه اتصال داده ها به صورت زیر می باشد:

#requestKind#merchantCode#terminalCode#startDate#endDate#minAmount#maxAmount#timeStamp#cardPan#desBankID#

ادامه مراحل تولید امضای دیجیتال به صورت زیر می‌باشد:

- اجرای الگوریتم درهم‌سازی SHA1 بر روی رشته حاصل از بند اول یا دوم.

- امضای رشته‌ی حاصل به وسیله PrivateKey، که نتیجه آن یک رشته‌ی باینری می‌باشد.

تبدیل رشته‌ی باینری به رشته ای با فرمت base64String، که این رشته امضای دیجیتال پذیرنده برای استفاده از سرویس های فوق می‌باشد.

Ê Å { • Y | ¬ » Ä]f °•¿Z ↑ æ◂qY • {

- فرستادن مقادیر RequestKind، TerminalCode، MerchantCode، StartDate، EndDate، TimeStamp و DigitalSignature در هر سه نوع درخواست الزامی می باشد.
- مقادیر MinAmount، MaxAmount ، CardPan و DesBankID، StartSettleDate و EndSettleDateدر درخواست اول اختیاری می باشند.
- مقادیر MinAmount، MaxAmount ، Action و ResultType در درخواست دوم اختیاری می باشند.
- مقادیر MinAmount، MaxAmount ، CardPan و DesBankID در درخواست سوم اختیاری می باشند.

- لازم به ذکر است در صورتی که فیلدهای اختیاری دارای مقدار نیستند در تولید امضای دیجیتال دقت شود. به عنوان مثال اگر در درخواست اول فیلد MinAmount و CardPan دارای مقدار نیستند اتصال داده ها برای تولید امضا به صورت زیر است:

#resultType#merchantCode#terminalCode#startDate#endDate##maxAmount#timeStamp##desBankID#startSettleDate#endSettleDate#

در این مرحله داده‌های ارسالی پردازش گردیده و سایت بانک با توجه به نوع درخواست یکی از سه صفحه XML زیر را برای فروشنده ارسال می‌کند

۱. XML زیر در صورتی ارسال می‌شود که درخواست از نوع اول (دریافت صورت حساب و کارمزد تراکنش‌ها) باشد.

```
<?xml version=\"1.0\" encoding=\"utf-8\"?>\n>

<resultObj>
<resultMessage> {پیام حاصل از نتیجه جستجو}</resultMessage>
<transactions>
       <transaction>
       <referenceID>{شماره ارجاع}</referenceID>
       <traceNumber>{شماره پیگیری}</traceNumber>
       <transactionDate>{تاریخ تراکنش}</transactionDate>
       <amount>{مبلغ تراکنش}</amount>
       <totalFee>{مبلغ کارمزد}</totalFee>
       <totalAmount>{مبلغ واریزی}</totalAmount>
       <cardNumber>{شماره کارت}</cardNumber>
```

```
        <settleDate>{تاریخ تسویه}</settleDate>
        <invoiceNumber>{شماره فاکتور}</invoiceNumber>
        <invoiceDate>{تاریخ فاکتور}</invoiceDate>
        <cardPanHash>{ شماره کارت به صورت hash }</cardPanHash>
        </transaction>
        .
        .
        .
</transactions>
</resultObj>
```

۲. XML زیر در صورتی برای فروشنده ارسال می‌شود که درخواست از نوع دوم(گزارش تراکنش ها بر اساس فیلترهای ارسالی) باشد.

```
<?xml version=\"1.0\" encoding=\"utf-8\"?>\n>
<resultObj>
<resultMessage>{پیام حاصل از نتیجه جستجو}</resultMessage>
<transactions>
        <transaction>
        <referenceID>{شماره ارجاع}</referenceID>
        <traceNumber>{شماره پیگیری}</traceNumber>
        <transactionDate>{تاریخ تراکنش}</transactionDate>
        <amount>{مبلغ تراکنش}</amount>
        <transactionReferenceID>[شماره ارجاع داخلی تراکنش]</transactionReferenceID>
        <invoiceNumber>[شماره فاکتور]</invoiceNumber>
        <invoiceDate>[تاریخ فاکتور]</invoiceDate>
        <result>{true|false}</result>
        <action>{1003|1004}</action>
        <cardNumber>{شماره کارت}</cardNumber>
        <cardPanHash>{ شماره کارت به صورت hash }</cardPanHash>
        </transaction>
        .
        .
        .
</transactions>
</resultObj>
```

۳. XML زیر در صورتی ارسال می‌شود که درخواست از نوع سوم (دریافت گزارش تراکنش های یک پایانه فروش) باشد.

```
<?xml version=\"1.0\" encoding=\"utf-8\"?>\n>

<resultObj>
<resultMessage> {پیام حاصل از نتیجه جستجو}</resultMessage>
<transactions>
        <transaction>
        <referenceID>{شماره ارجاع}</referenceID>
        <traceNumber>{شماره پیگیری}</traceNumber>
        <transactionDate>{تاریخ تراکنش}</transactionDate>
        <amount>{مبلغ تراکنش}</amount>
        <totalFee>{مبلغ کارمزد}</totalFee>
        <totalAmount>{مبلغ واریزی}</totalAmount>
        <cardNumber>{شماره کارت}</cardNumber>
        <invoiceNumber>{شماره فاکتور}</invoiceNumber>
        <invoiceDate>{تاریخ فاکتور}</invoiceDate>
        <cardPanHash>{ شماره کارت به صورت hash }</cardPanHash>
        </transaction>
        .
        .
```

```
        .
</transactions>
</resultObj>
```

‰ • Y , ³ d § Z Ë • { É Y|€¯] Æ Z ¿ Å Â d Ã ¼ { Â Ì { a µ Z ‡

```
<script language="C#" runat="server">

private ReadPaymentResult(){

string timeStamp = DateTime.Now.ToString("yyyy/MM/dd HH:mm:ss");
string resType ="true";
int requestKind = 1;
int merchantCode = 115;
int terminalCode = 12;
string startDate = "1390/1/1";
string endDate = "1390/10/1";
int minAmount = 10;
int maxAmount = 1000000;
RSACryptoServiceProvider rsa = new RSACryptoServiceProvider();
rsa.FromXmlString("<RSAKeyValue><Modulus>.... ");
string data = "#" + requestKind + "#" + merchantCode + "#" + terminalCode + "#" +
startDate + "#" + endDate + "#" + minAmount + "#" + maxAmount + "#" + timeStamp + "#";
if (requestKind == 1)
    data += cardPan + "#" + bankBin + "#"+ startSettleDate + "#"+ endSettleDate+"#";
if (requestKind == 2)
    data += action + "#" + resType + "#";
if (requestKind == ٣)
    data += cardPan + "#" + bankBin + "#";

byte[] sign = rsa.SignData(Encoding.UTF8.GetBytes(data), new
SHA1CryptoServiceProvider());
string mainSign=Convert.ToBase64String(sign);
string textGetTransaction = "MerchantCode=" + merchantCode + "&TerminalCode=" +
terminalCode +"&RequestKind=" + requestKind + "&StartDate=" + startDate +"&EndDate=" +
endDate + "&MinAmount=" + minAmount + "&MaxAmount=" + maxAmount +"&CardPan=" + cardPan
+ "&DesBankID=" + bankBin + "&ResultType=" + resType +"&action=" + action +
"&TimeStamp=" + timeStamp + "&sign=" + mainSign + "&StartSettleDate="+startSettleDate+
"&EndSettleDate="+endSettleDate;
byte[] textArray = Encoding.UTF8.GetBytes(textGetTransaction);
HttpWebRequest request =
(HttpWebRequest)WebRequest.Create("http://pep.shaparak.ir/Merchant/GetTransa
ctionList.aspx");
request.Method = "POST";
request.ContentType = "application/x-www-form-urlencoded";
request.ContentLength = textArray.Length;

request.GetRequestStream().Write(textArray, 0, textArray.Length);
HttpWebResponse response = (HttpWebResponse)request.GetResponse();
StreamReader reader = new StreamReader(response.GetResponseStream());
string result = reader.ReadToEnd();}

</script>
```

½ • Z ¬ f » Z ¿ Œ Ë Z • Á ¿ d ¼ • Â Ì a

الگوریتم‌های رمز گذاری نامتقارن نوعی از الگوریتم‌های رمز نگاری هستند که دارای دو کلید مختلف می‌باشند که از یکی جهت رمزنگاری و از دیگری جهت رمز گشایی استفاده می‌شود. این الگوریتم‌ها در گستره وسیعی از کاربردها به کار می رود. در این الگوریتم‌ها کلید اول را کلید عمومی (**Public Key**) و کلید دوم را کلید خصوصی(**Private Key**) می‌نامند. یکی از کاربردهای مهم الگوریتم های رمز نگاری نامتقارن استفاده از آنها در تولید امضای دیجیتال می‌باشد.

µ Z f Ì n Ë»{ YÉ ŽÂ Æ ¨ ›

امضای دیجیتال روشی مبتنی بر الگوریتم های رمزنگاری نا متقارن می باشد که به کمک ان می‌توان اطمینان حاصل کرد که داده‌های ارسالی از جانب شخص مشخصی ارسال شده است. نمونه ای از این الگوریتم‌ها می توان به RSA و DSA اشاره کرد.

روال کار در امضای دیجیتال به این شکل است که پیش از ارسال داده ها، اطلاعات را با استفاده از الگوریتم‌های درهم سازی یک‌طرفه (Hash Algorithms) به یک کد درهم (Hash) تبدیل می‌شود. از نمونه این الگوریتم ها می توان به MD5, SHA1 و ... اشاره کرد. یک‌طرفه بودن در این الگوریتم‌ها به این معنی است که پس از کد شدن اطلاعات به هیچ عنوان نمی توان از روی این کدها، اطلاعات اصلی را به دست آورد. پس از در هم سازی اطلاعات، به منظور تولید امضای دیجیتال، باید از یکی از الگوریتم‌های رمز نگاری نامتقارن استفاده شود، و با استفاده از کلید خصوصی (Private Key) آن الگوریتم، رشته‌ی تولید شده توسط الگوریتم درهم سازی را امضا نمود.

: Ê • Â • y |Ì ¸ ¯ Á Ê » Â ¼ Ÿ |Ì ¸ ¯ 1

کلید عمومی بخشی از کلید است که بین همه توضیع می‌شود و هیچ نگرانی از لو رفتن و دزدیده شدن آن وجود ندارد به واقع لفظ "عمومی" نیز بیان‌گر همین مطلب است. اگر داده‌ای برای صاحب کلید عمومی (پخش کننده کلید عمومی) باید رمز شود با استفاده از این کلید رمز نگاری شده و ارسال می‌شود. نکته مهم الگوریتم‌های نامتقارن در این مطلب است که داده‌های رمز شده با کلید عمومی فقط و فقط با کلید خصوصی قابل رمز گشایی هستند و دوباره با همان کلید عمومی نمی‌توان آنها را رمزگشایی کرد به همین دلیل داشتن کلید عمومی کمکی به رمزگشایی داده‌ها نخواهد کرد.

کلید خصوصی در واقع بخشی از کلید است که به وسیله آن داده های رمز شده به وسیله کلید عمومی را می‌توان رمز گشایی کرد. صاحب کلید خصوصی باید حداکثر محافظت از این کلید را انجام دهد و به هیچ عنوان اجازه ندهد که این کلید در دست کسی غیر از خودش قرار گیرد. علاوه بر این با استفاده از کلید خصوصی می‌توان اسناد و مدارک مانند Documentها، Emailها و پیغامها را امضا کرد و امضای صورت گرفته را در انتهای Document، Email و یا پیغام قرار داد. در این حالت گیرنده پیغام با داشتن اصل پیغام، امضای دیجیتال زیر آن و کلید عمومی شما می تواند از صحت امضا اطمینان حاصل کند و مطمئن شود که داده‌ها از جانب شما ارسال شده است. اما با کلید عمومی به هیچ عنوان نمی‌تواند امضای شما را جعل کند.