Криптография/Примеры программных реализаций шифров — Urbanculture

обратно к статье «Криптография»

Примеры программных реализаций шифров

Содержимое этой подстраницы необходимо исправить и дополнить

Шифр Цезаря

Собственно пример шифра Цезаря. Переменная $num задаёт смещение.

```
<?php

$text = "This is a example";
$num = 3; // Смещение

// Шифрование
$result='';
for ($x = 0; $x < strlen($text); $x++) {
        $y = ord(substr($text,$x,1)) + $num;
        /* Да, если величина более 255, надо вычесть
           поскольку отсчёт тогда начинается с начала алфавита */
        if ($y > 255) $y = $y - 255;
        $result = $result.chr($y);
}

print "Text is '$text', result is '$result'<br>";

// Дешифровка
$result2 = '';
for ($x = 0; $x < strlen($result); $x++) {
        $y = ord(substr($result,$x,1)) - $num;
        if ($y < $num) $y = 255 - $num;
        $result2 = $result2.chr($y);
}
print "Decrypt text is '$result2'";

?>
</php>
```

Шифр Виженера

Пример реализации полиалфавитного Шифра Виженера на PHP. Для кодирования и дешифровки используется шестнадцатиричный алфавит. Программа позволяет работать с любыми двоичными данными и ключами.

```
<?php

$text = "This is an example";
$key = strtoupper(bin2hex("Sample key"));

$t0 = array ('0','1','2','3','4','5','6','7','8','9','A','B','C','D','E','F');
$t1 = array ('1','2','3','4','5','6','7','8','9','A','B','C','D','E','F','0');
$t2 = array ('2','3','4','5','6','7','8','9','A','B','C','D','E','F','0','1');
$t3 = array ('3','4','5','6','7','8','9','A','B','C','D','E','F','0','1','2');
$t4 = array ('4','5','6','7','8','9','A','B','C','D','E','F','0','1','2','3');
$t5 = array ('5','6','7','8','9','A','B','C','D','E','F','0','1','2','3','4');
$t6 = array ('6','7','8','9','A','B','C','D','E','F','0','1','2','3','4','5');
$t7 = array ('7','8','9','A','B','C','D','E','F','0','1','2','3','4','5','6');
$t8 = array ('8','9','A','B','C','D','E','F','0','1','2','3','4','5','6','7');
$t9 = array ('9','A','B','C','D','E','F','0','1','2','3','4','5','6','7','8');
$tA = array ('A','B','C','D','E','F','0','1','2','3','4','5','6','7','8','9');
$tB = array ('B','C','D','E','F','0','1','2','3','4','5','6','7','8','9','A');
$tC = array ('C','D','E','F','0','1','2','3','4','5','6','7','8','9','A','B');
$tD = array ('D','E','F','0','1','2','3','4','5','6','7','8','9','A','B','C');
$tE = array ('E','F','0','1','2','3','4','5','6','7','8','9','A','B','C','D');
$tF = array ('F','0','1','2','3','4','5','6','7','8','9','A','B','C','D','E');

$table = array ('0' => $t0, '1' => $t1, '2' => $t2, '3' => $t3, '4' => $t4, '5' => $t5,
                '6' => $t6, '7' => $t7, '8' => $t8, '9' => $t9, 'A' => $tA, 'B' => $tB,
                'C' => $tC, 'D' => $tD, 'E' => $tE, 'F' => $tF);

$khex = array ('0','1','2','3','4','5','6','7','8','9','A','B','C','D','E','F');
```

```
$kdec = array (0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15);

// ----------- Crypt -----------------
print "Original text - '$text', ";
$text = strtoupper(bin2hex($text));

$keylen = strlen($key);
$y = -1;
$result='';

for ($x = 0; $x < strlen ($text); $x++) {

    $y++;
    if ($y < $keylen) $ksimb = substr($key, $y, 1); else {
        $y = 0;
        $ksimb = substr($key, 0, 1);
    }

    $simb = substr($text, $x,1);
    print $simb;
    $ksimb = $kdec[array_search($ksimb, $khex)];
    $result = $result.$table[$simb][$ksimb];
}

// ----------- End crypt --------------

print "(HEX)\nCryped text - ".$result."(HEX), Key is '$key';\n";

// ----------- Decrypt ---------------
$dresult='';
$y = -1;

for ($x = 0; $x < strlen($result); $x = $x + 2) {

    $tmp = substr($result, $x, 2);
    $simb1 = substr($result, $x, 1);
    $simb2 = substr($result, $x+1, 1);

    $y++;
    if ($y < $keylen) $ksimb = substr($key, $y, 1); else {
        $y = 0;
        $ksimb = substr($key, 0, 1);
    }

    $ssimb1 = array_search($simb1, $table[$ksimb]);

    $y++;
    if ($y < $keylen) $ksimb = substr($key, $y, 1); else {
        $y = 0;
        $ksimb = substr($key, 0, 1);
    }

    $ssimb2 = array_search($simb2, $table[$ksimb]);

    $dresult = $dresult.chr($ssimb1 * 16 + $ssimb2);
}
// ----------- End decrypt ------------

print "Decrypted text - '$dresult'\n";

?>
```

Примечание: да, автор знает про функцию **hex2bin**, но она существует в версии PHP>=5.4.0, автор-же решил не привязываться к конкретным версиям языка программирования и сделать программу независимой от версий.