**Improved Hash Algorithm**

The **Improved Hash Algorithm** is a PHP class to derive from a weak password a strong hash value which could be stored in a database.  
For an attacker it is realy hard to crack the hash value, because the class use a random salt and a key strengthening technique to generate a strong hash value.  
  
In normal applications, the passwords are hashed with a single call of md5() or sha1(), but an attacker can check arround 20 - 40 million passwords per secondes, so weak passwords are broken really fast.  
The **Improved Hash Algorithm** calls e.g. 2500 times md5() (sha1() is also possible), so an attacker can check only 10000 passwords per second instead of 25 million per second an his attack takes than e.g. 2500 days (nearly 7 years) instead of 1 day.  
The random salt which is used prevents dictionary attacks, which became popular through rainbow tables.  
  
**Usage of Improved Hash Algorithm**  
The usage is really simple:  
<?php  
require("iha.php");  
$iha = new iha();  
$pw = "secret password";  
  
//Calculate the hash value and store it in a database  
$hash = $iha->hash($pw);  
  
echo "Password: ".$pw." <br>  
Hash-Value: ".$hash."<br><br>";  
  
if($iha->compare($pw, $hash))  
   echo "Password ok";  
else  
   echo "Wrong password";  
?>