

```
<HTML><HEAD><TITLE>..... Hacked By Ulow .....</TITLE></HEAD><link REL="SHORTCUT ICON" HREF="http://swararakyat.files.wordpress.com/2007/06/indonesia_flag.gif">
</head><body dir="ltr" text="#0033ff" vlink="#0033ff" alink="#0033ff" <div align="center"> <div style="border-width: 1px; border-style: none; border-color: rgb(0, 0, 0) rgb(0, 0, 0) black black; z-index: 1; left: 0px; width:
100%; position: absolute; top: 0px; height: 100%; background-color: black;" id="Layer1">
<div><br />
<HTML><HEAD>
<META http-equiv=Content-Type content="text/html; charset=windows-1254">
<STYLE type=text/css>BODY {
    SCROLLBAR-FACE-COLOR: #000000; SCROLLBAR-HIGHLIGHT-COLOR: #000000; SCROLLBAR-SHADOW-COLOR: #000000; SCROLLBAR-3DLIGHT-COLOR: #000000; SCROLLBAR-ARROW-COLOR:
#ffffff; SCROLLBAR-TRACK-COLOR: #ffffff; FONT-FAMILY: Verdana; SCROLLBAR-DARKSHADOW-COLOR: #000000
}
.Estilo10 {
    COLOR: #ffffff; FONT-FAMILY: Haettenschweiler
}
.Estilo8 {
    FONT-SIZE: 10px; COLOR: #ffffff; FONT-FAMILY: Haettenschweiler
}
</STYLE>
```

```
<SCRIPT LANGUAGE="JavaScript">
```

```
/*
```

An object-oriented Typing Text script, to allow for multiple instances.

A script that causes any text inside any text element to be "typed out", one letter at a time. Note that any HTML tags will not be included in the typed output, to prevent them from causing problems. Tested in Firefox v1.5.0.1, Opera v8.52, Konqueror v3.5.1, and IE v6.

Browsers that do not support this script will simply see the text fully displayed from the start, including any HTML tags.

Functions defined:

TypingText(element, [interval = 100,] [cursor = "",] [finishedCallback = function(){return}]):

Create a new TypingText object around the given element. Optionally specify a delay between characters of interval milliseconds. cursor allows users to specify some HTML to be appended to the end of the string whilst typing. Optionally, can also be a function which accepts the current text as an argument. This allows the user to create a "dynamic cursor" which changes depending on the latest character or the current length of the string. finishedCallback allows advanced scripters to supply a function to be executed on finishing. The function must accept no arguments.

TypingText.run():

Run the effect.

static TypingText.runAll():

Run all TypingText-enabled objects on the page.

```
*/
```

```
TypingText = function(element, interval, cursor, finishedCallback) {
  if((typeof document.getElementById == "undefined") || (typeof element.innerHTML == "undefined")) {
    this.running = true; // Never run.
    return;
  }
  this.element = element;
  this.finishedCallback = (finishedCallback ? finishedCallback : function() { return; });
  this.interval = (typeof interval == "undefined" ? 20 : interval);
  this.origText = this.element.innerHTML;
  this.unparsedOrigText = this.origText;
  this.cursor = (cursor ? cursor : "");
```

```

this.currentText = "";
this.currentChar = 0;
this.element.typingText = this;
if(this.element.id == "") this.element.id = "typingtext" + TypingText.currentIndex++;
TypingText.all.push(this);
this.running = false;
this.inTag = false;
this.tagBuffer = "";
this.inHTMLEntity = false;
this.HTMLEntityBuffer = "";
}
TypingText.all = new Array();
TypingText.currentIndex = 0;
TypingText.runAll = function() {
  for(var i = 0; i < TypingText.all.length; i++) TypingText.all[i].run();
}
TypingText.prototype.run = function() {
  if(this.running) return;
  if(typeof this.origText == "undefined") {
    setTimeout("document.getElementById('" + this.element.id + "').typingText.run()", this.interval); // We haven't finished loading yet. Have patience.
    return;
  }
  if(this.currentText == "") this.element.innerHTML = "";
  // this.origText = this.origText.replace(/<([^\>])*>/, ""); // Strip HTML from text.
  if(this.currentChar < this.origText.length) {
    if(this.origText.charAt(this.currentChar) == "<" && !this.inTag) {
      this.tagBuffer = "<";
      this.inTag = true;
      this.currentChar++;
      this.run();
      return;
    } else if(this.origText.charAt(this.currentChar) == ">" && this.inTag) {
      this.tagBuffer += ">";
      this.inTag = false;
      this.currentText += this.tagBuffer;
      this.currentChar++;
      this.run();
      return;
    } else if(this.inTag) {
      this.tagBuffer += this.origText.charAt(this.currentChar);
      this.currentChar++;
      this.run();
      return;
    } else if(this.origText.charAt(this.currentChar) == "&" && !this.inHTMLEntity) {
      this.HTMLEntityBuffer = "&";
      this.inHTMLEntity = true;
      this.currentChar++;
      this.run();
      return;
    } else if(this.origText.charAt(this.currentChar) == ";" && this.inHTMLEntity) {
      this.HTMLEntityBuffer += ";";
      this.inHTMLEntity = false;
      this.currentText += this.HTMLEntityBuffer;
      this.currentChar++;
      this.run();
    }
  }
}

```

```

    return;
} else if(this.inHTMLEntity) {
    this.HTMLEntityBuffer += this.origText.charAt(this.currentChar);
    this.currentChar++;
    this.run();
    return;
} else {
    this.currentText += this.origText.charAt(this.currentChar);
}
this.element.innerHTML = this.currentText;
this.element.innerHTML += (this.currentChar < this.origText.length - 1 ? (typeof this.cursor == "function" ? this.cursor(this.currentText) : this.cursor) : "");
this.currentChar++;
setTimeout("document.getElementById('" + this.element.id + ").typingText.run()", this.interval);
} else {
    this.currentText = "";
    this.currentChar = 0;
    this.running = false;
    this.finishedCallback();
}
}
}
</script>
<style>
td{align: center; font-family: Bradley Hand ITC; font-size: 18pt; color: black}
a{align: center; font-family: Bradley Hand ITC; font-size: 12pt; color: red}

</style>
<center>

<div
id="example1"></div>
<p id="example2">
<br>
<b>Hacked by ulow :"><br>
<br>
<br>
<b>@2012
</center>

</p><script type="text/javascript">
//Define first typing example:
new TypingText(document.getElementById("example1"));
//Define second typing example (use "slashing" cursor at the end):
new TypingText(document.getElementById("example2"), 50, function(i){
var ar = new Array("_", " ", " ", " "); return " " + ar[i.length %
ar.length]; });
//Type out examples:
TypingText.runAll();
</script>

<SCRIPT language=JavaScript type=text/javascript>

<!--
var rows=1; // must be an odd number
var speed=10; // lower is faster
var reveal=10; // between 0 and 2 only. The higher, the faster the word appears

```

```
var effectalign="default" //enter "center" to center it.
```

```
</script>
```

```
</body>
```

```
</html>
```

```
<blink><blink>
```