Zend_Session_ValidatorMagicString - Vladislav Rastrusny

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

Zend Framework: Zend_Session_ValidatorMagicString Component Proposal

Proposed Component Name	Zend_Session_ValidatorMagicString
Developer Notes	http://framework.zend.com/wiki/display/ZFDEV/Zend_Session_ValidatorMagicString
Proposers	FractalizeR@yandex.ru
Revision	1.0 - 13 November 2007: Initial release (wiki revision: 5)

Table of Contents

1. Overview
2. References
3. Component Requirements, Constraints, and Acceptance Criteria
4. Dependencies on Other Framework Components
5. Theory of Operation
6. Milestones / Tasks
7. Class Index
8. Use Cases
9. Class Skeletons

1. Overview

Zend_Session_ValidatorMagicString is almost a copy of Zend_Session_ValidatorHttpUserAgent with an exception, that it stores a custom value in client's cookie on session creation and verifies it on every session resume

2. References

- Component Page on SourceForge.net

3. Component Requirements, Constraints, and Acceptance Criteria

Nothing special requried to use this component

- This component **will not** save any data using Zend_Cache or the filesystem. All transient data **will be** saved using Zend_Session and client cookies

4. Dependencies on Other Framework Components

- Zend_Session

5. Theory of Operation

The component is instantized by executing Zend_Sesion::registerValidator(new Zend_Session_Validator_MagicString('Key', 'Value'));

6. Milestones / Tasks

The component is so simple, that it does not require any milestones

7. Class Index

- Zend_Session_ValidatorMagicString

8. Use Cases

UC-01	**You want to validate session against custom cookie value stored at session creation**

9. Class Skeletons

```
<?php
require_once "Zend/Session.php"
try {
  Zend_Session::start(); //Starting session
} catch (Zend_Session_Exception $e) {
  //Unable to create session. Possibly, session validation failed
  //(For example, UserAgent or IP address is changed on client. We can get reason from
$e->getMessage())
}

//Doing something and finding out that there is a need to remember this session data
if($needToKeepThisSession)  {
  Zend_Sesion::registerValidator(new Zend_Session_Validator_HttpUserAgent());
//Registering first validator
  Zend_Sesion::registerValidator(new Zend_Session_Validator_MagicString('MyKey',
'MyValue')); //Registering second validator
  Zend_Session::rememberMe();
}
?>
```

]]></ac:plain-text-body></ac:macro>
]]></ac:plain-text-body></ac:macro>