

# Automatischer Login per „Einladungslink“

Andreas Fieger (@fiedschMuc)

Stammtisch Contao Bayern 2016-10-13

# Ausgangslage/Ziel

- Website stellt bestimmte Informationen nur für angemeldete User bereit
- E-Mail-Versand mit Link zur Website
- Automatischer Login
- Kein „Hochsicherheitstrakt“
- Möglichst viel Contao-Bordmittel
- Es sollen nicht alle benötigten User extra zuvor angelegt werden müssen

# Idee

- Link zur Seite mit Parameter `token`
- Auf der Zielseite:
  - Inhalte (geschützt)
  - Modul, das sich um den automatischen Login kümmert (ungeschützt)

# Implementierung

- Frontend-Modul ohne weitere Optionen  
(`ModuleLogin` erweitern)
- Login-Formular besteht nur aus Eingabefeld für das Token  
(Originaltemplatex minus Passwort-Feld)
- Username/Password → Token/<Konstante>

# Modul (1)

```
class ModuleTokenlogin extends \ModuleLogin {
    public function generate() {
        if (TL_MODE == 'BE') { /* Platzhalter anzeigen */ }
        if (\Input::post('FORM_SUBMIT') == 'tl_login') {
            // Fehlertexte anpassen ("Benutzername oder Passwort falsch" passt nicht)
            $token_from_URL = \Input::get('token');
            // Checks (z.B. ist das Token "gültig"?)
            \Input::setPost('username', $token_from_URL); // "username"
            \Input::setPost('password', self::TOKENUSERPASSWORD); // password dummy
        }
        return parent::generate();
    }

    // ...
}
```

# Modul (2)

```
class ModuleTokenlogin extends \ModuleLogin {
    // ...

    protected function compile() {
        parent::compile();
        if (!FE_USER_LOGGED_IN) {
            // anderes Template verwenden
            // Fehlertexte anpassen
        } else {
            // "logged in as ..." anpassen
        }
    }
}
```

# Hook

- Wird benötigt (und getriggert), weil der User zunächst nicht existiert

```
$GLOBALS['TL_HOOKS']['importUser'][] = array('MyHooks', 'myImportUser');
```

```
public function myImportUser($strUsername, $strPassword, $strTable) {  
    // front end only!  
    if ($strTable == 'tl_member') {  
        return $this->importFromTokenlist($strUsername, $strPassword);  
        // importFromTokenlist() ist im Wesentlichen:  
        // token (==$strUsername) prüfen und dann ggf.  
        // $newMember = new \MemberModel();  
        // ...  
        // $newMember->save();  
        return true;  
    }  
    return false;  
}
```

# Kleinere Probleme (Diskussion bitte!)

```
// Token aus der URL holen und damit quasi einen "autologin" durchführen
//
// Achtung: wenn das Token gültig ist wird der Login so funktionieren.
// Wenn nicht, geschieht durch parent::generate(); ein reload der Seite
// Da die Url dann aber noch die Gleiche ist haben wir eine Endlosschleife gebaut.
//
// Daher: Hier bereits prüfen "ob es funktionieren wird"
$token_from_url = \Input::get('token');

if ($token_from_url && !FE_USER_LOGGED_IN) {
    // Wird der Login mit diesem Token erfolgreich sein
    $sampleModel = \SampleModel::findByToken($token_from_url);
    if ($sampleModel) {
        /* ... */
    }
}
```

# Ausbau

- Datenbanktabelle für die Token (Backendmodul)
- „Reporting“ zur Nutzung (Besucherstatistik)
- Periodisches Löschen der generierten User

# Diskussion

- Ohne Token im Link reduziert sich die Aufgabenstellung auf: Loginmaske mit Token als Passwort (Zugangscod)
- ...

# Link zu Beispiel-Repository

- <https://github.com/fiedsch/contao-tokenlogin>