

PyHearing - Segnalazione #26

Permettere l'invio di mail a tutti gli utenti

04/11/2014 03:24 PM - Mark Caglienzi

Status:	Chiuso	Start date:	04/11/2014
Priority:	Normale	Due date:	
Assignee:	Christopher R. Gabriel	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			
Resolution:	fixed		

Description
A seguito di heartbleed, permettere di inviare una mail a tutti gli utenti del sito (ad esempio per consigliare di cambiare la password).

Associated revisions

Revision 850056c9 - 04/11/2014 04:03 PM - Mark Caglienzi

Send normal mails immediately, enqueue mass mails. refs #26

Revision 726f1c25 - 04/16/2014 01:04 PM - Mark Caglienzi

Check for is_staff and is_superuser in the mass_mail view. refs #26

History

#1 - 04/11/2014 03:33 PM - Mark Caglienzi

- Status changed from Nuovo to Commenti
- Assignee set to Christopher R. Gabriel

Ho guardato un po' di app django relative alle mail e ho trovato questa: https://github.com/ui/django-post_office che necessita di questo jsonfield (non ho indagato approfonditamente se sia possibile usare quello che usiamo di solito) <https://github.com/bradjasper/django-jsonfield>

Al solito ho scaricato da github la versione taggata più recente (la 0.8.2) e ho symlinkato la directory dentro al progetto, e ho fatto lo stesso con il jsonfield. I due file che ho usato sono questi:

- https://github.com/ui/django-post_office/archive/v0.8.2.zip
- <https://github.com/bradjasper/django-jsonfield/archive/master.zip>

e le due directory si chiamano rispettivamente post_office e jsonfield.

Ho committato in [896fd6fd](#) nel file settings.py l'aggiunta dell'app e EMAIL_BACKEND. Per i test ho usato in settings_local.py questo settaggio:

```
POST_OFFICE_BACKEND = 'django.core.mail.backends.console.EmailBackend'
```

in modo da avere l'output delle mail in console.

Praticamente post_office diventa il backend mail, e il setting che prima era EMAIL_BACKEND lo si passa a POST_OFFICE_BACKEND.

post_office supporta un bel po' di cose, ma come soluzione rapida ho aggiunto questo:

- Nel menu user (solo per chi è staff e superuser) c'è la voce "Send mail to all users"
- Questo porta a un form con subject e message
- Tasto Send.

Cliccando sul tasto vengono presi tutti gli indirizzi email degli user del sito senza nessun discrimine e vengono create le mail. Alla fine della creazione delle mail esce un message che dice il numero di mail accodate e il numero di user trovati (così si fa un confronto a occhio per vedere se qualcuno è sfuggito).

A me con una 40ina di email col mio database di test impiega non più di 5-6 secondi a tornare responsivo. Immagino che con centinaia di indirizzi i tempi si dilatino.

Attenzione che se si prova con un'installazione col fillrandom, questo va corretto in modo che le mail siano valide (e quindi gli indirizzi non devono finire con '.invalid' come sono attualmente).

Da questo momento in poi le mail sono nel database e si possono andare a vedere dall'admin.

Poi per effettuare l'invio vero e proprio post_office usa un comando di manage:

```
./manage.py send_queued_mail
```

(la documentazione di post_office consiglia di metterlo ad esempio in un cronjob così le mail vengono inviate viavia).

Con i settaggi globali e locali che ho detto, l'uso di quel comando a me fa vedere nella shell le mail inviate e alla fine dà un resoconto del tipo:

```
[INFO]11-04-2014 14:18:18 PID 24765: Process finished, 41 emails attempted, 41 sent, 0 failed  
[INFO]11-04-2014 14:18:18 PID 24765: 41 emails attempted, 41 sent, 0 failed
```

Lo stato della mail è un attributo dell'oggetto mail nel database (queued, sent, failed, eccetera. Vengono create dall'invio del form come 'queued' e poi passano in 'sent' una volta inviate dal comando di manage), così si può eventualmente implementare un modo per tentare di re-inviare le mail che hanno fallito.

#2 - 04/11/2014 04:07 PM - Mark Caglienzi

Dato che il default di post_office è quello di accodare le email, anche le mail di appuntamento fissato/cancellato venivano messe in coda (e quindi non partivano finché non veniva lanciato il comando di manage).

In [850056c9](#) ho configurato post_office in modo che invii subito le mail come default, e ho aggiunto la priorità medium nella chiamata che crea le mail di invio massivo in modo che quelle vengano messe in coda, in modo da rendere la webapp responsiva il prima possibile (che non deve attendere l'effettivo invio delle centinaia di mail). Per l'invio di queste mail bisognerà usare il comando di management.

#3 - 04/16/2014 12:48 PM - Christopher R. Gabriel

- Status changed from *Commenti* to *In elaborazione*

- Assignee changed from *Christopher R. Gabriel* to *Mark Caglienzi*

Nella view di mass mail manca il controllo sui permessi: e' vero che la voce del menu compare solo se sono superuser e staff, ma mi sembra che qualcuno conosca la url, gli basta avere una qualunque utenza per poter spammare. Almeno così' mi pare... verificiamo e mettiamo anche lì' il controllo?

#4 - 04/16/2014 01:08 PM - Mark Caglienzi

- Status changed from *In elaborazione* to *Commenti*

- Assignee changed from *Mark Caglienzi* to *Christopher R. Gabriel*

Confermo la dimenticanza. Aggiunto il controllo (devono essere presenti entrambi i flag, is_staff e is_superuser) anche nella view in [726f1c25](#)

#5 - 04/16/2014 01:12 PM - Christopher R. Gabriel

- *Status changed from Commenti to Chiuso*

- *Resolution set to fixed*

Verificato, ok, chiudo.