Diario di lavoro

|  |  |
| --- | --- |
| Luogo | Scuola d’arti mestieri Trevano |
| Data | 04.12.2020 |

|  |
| --- |
| Lavori svolti |
| Oggi ho implementato un sistema non richiesto ma importante per l’autenticazione peer accedere ad dei file di un sito. Lo script scheduler è accesibile da web e un utente malitenzionato potrebbe fare partire lo script. Invece utilizzando una Basic access authentication può accedere al file solamente un determinato utente. La Basic access authentication è un metodo per fornire credenziali di accesso tra client e server nel contesto di una trasazione HTTP. L'implementazione HTTP Basic authentication è la più semplice tecnica per consentire l'accesso a risorse web con controllo di accesso perché non richiede cookie, id di sessione e pagine di login. Basic authentication utilizza headers HTTP statici e standard che non richiedono handshake eseguito preventivamente. Ho dovuto installare l’estensione RESTClient che mi permette di creare il curl che dovrò inserire nel cron.    Il cron dovrà essere: curl -X GET -H 'Authorization: Basic QWRtaW46QWRtaW4uMTIz' -i 'http://www.samtinfo.ch/i17caspie/scheduler/scheduler.php'  Nello script dello scheduler per ricavare le credenziale devo utilizzare la variabile globale $\_SERVER che è un’array che contiene informazioni come intestazioni, percorsi e posizioni degli script. L’indice PHP\_AUTH\_PW è la password di autenticazione HTTP, invece PHP\_AUTH\_USER è lo username di autenticazione HTTP.  Immagine che contiene testo  Descrizione generata automaticamente  La password e lo username dell’autenticazione corrisponde alla password e lo username di un utente amministratore dell’applicativo. Se corrisponde l’utente può eseguire lo script. Per ora il codice del controller è commentato perché su infomaniak non so se esiste la possbilità di aggiungere l’autenticazione come opzione.  Oggi ho anche eleiminato la pagina modifica\_utente, perché per modificare il tipo di utente (amministratore o responsabile) può essere anche gestito direttamente nella pagina utente. Nella tabella utente sotto la colonna tipo per ogni utente c’è un menu a tendina con due opzioni (amministrore e responsabile).    Ho anche modficato la tabella visualizzando solamente il nome, cognome, email, tipo e elimina. Per modicare il tipo di utente ho utilizzando lo stesso scvript ajax della pagina gestione dove cambiavo il tipo di backup.  Immagine che contiene testo  Descrizione generata automaticamente  Oggi ho anche realizzato i vari controlli per le sql injection utilizzando i metodi preg\_match, ma avuto diversi problemi. |

|  |
| --- |
| Problemi riscontrati e soluzioni adottate |
| Oggi ho riscontrato dei problemi con le preg\_match. La funzione preg\_match permette di verificare una stringa con un pattern (regular expression). Se la stringa rispetta il pattern la funzione ritorna true altrimenti false. I pattern che creo però non funzionano e non so il motivo. Non ho ancora trovato una soluzione |

|  |
| --- |
| Punto della situazione rispetto alla pianificazione |
| Sono ad un buon punto della pianificazione. |

|  |
| --- |
| Programma di massima per la prossima giornata di lavoro |
| Per la prossima giornata di lavoro devo terminare il problema legato alla funzione preg\_match e modificare il sistema che permette di creare un utente. Quando viene creato un utente l’utente riceve come username l’e-mail e la password. Quando deve modificare la password provvisoria può anche scegliere lo username (nome.cognome).  Per finire posso continuare a scrivere la documentazione del progetto. |