



گزارش پروژه مبانی رمز و امنیت شبکه

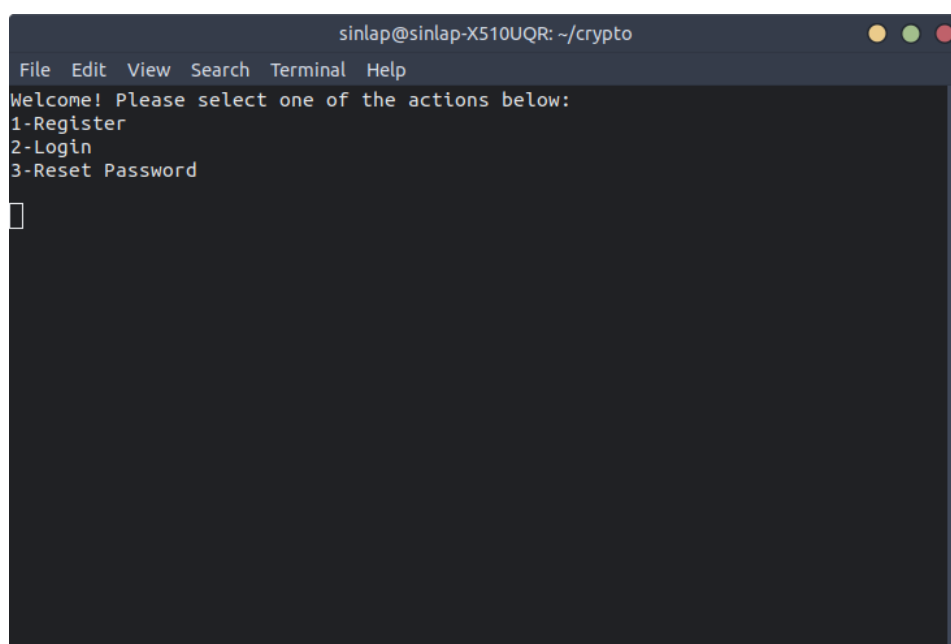
دکتر میرمحسنی

سینا کریمی 97105509

هدف پروژه:

هدف در این پروژه نوشتن کد برنامه مدیریت رمزهای عبور برای سایت های مختلف میباشد. این برنامه قابلیت افزودن کاربر جدید، افزودن سایت و رمز آن، درخواست رمز سایت موردنظر و تغییر رمز کاربر را دارد. این برنامه برای ذخیره اطلاعات از فایل های JSON استفاده میکند. برای رمزگذاری پسورد ها از AES ، برای هش کردن از تابع SHA-2 و برای تولید کلید از رمز کاربر از پروتکل PBKDF2 استفاده شده است.

در ادامه همراه با نمایش اجرای برنامه، نحوه کار آن توضیح داده میشود.



در ابتدا با اجرا کردن فایل main.py این صفحه مواجه میشویم که صفحه اول برنامه میباشد. در این قسمت میتوان عملکرد دلخواه را انتخاب کرد.

ابتدا میخواهیم کاربری با نام کاربری sina و رمز pass وارد کنیم.

کلمه Register را مینویسیم:

```
sinlap@sinlap-X510UQR: ~/crypto
File Edit View Search Terminal Help
Welcome! Please select one of the actions below:
1-Register
2-Login
3-Reset Password
Register█
```

بعد وارد کردن این کلمه از ما نام کاربری و رمز خواسته میشود.

```
sinlap@sinlap-X510UQR: ~/crypto
File Edit View Search Terminal Help
Please enter a unique username: sina
Please enter a password:
Please enter your password again:█
```

در این برنامه رمز به صورت ایمن وارد میشود. یعنی زمانی که رمز را وارد میکنید، رمز روی صفحه نمایش داده نمیشود. در اینجا کاربر با رمز pass ساخته میشود. در اینجا نام کاربری در فایل

users.json ذخیره میشود و رمز به صورت username+password+salt هش شده و ذخیره میشود.

Salt تولید شده رندوم میباشد و جایی ذخیره نمیشود و هربار تولید میشود. اینکار با استفاده از تابع random.seed انجام شده است و طول آن بین 12 تا 20 حرف متغیر میباشد. مقدار هش شده به دلیل اینکه شامل نام کاربری نیز میباشد، مانع حمله به فایل کاربری میشود زیرا حتی اگر این عبارت هش شده را جابه جا کنیم نیاز داریم که همان کاربر را داشته باشیم تا بتوانیم احراز اصالت بکنیم و وارد برنامه بشویم. این قسمت از کد در فایل register.py و در تابع userRegister() پیاده سازی شده است. نمونه ای از مقادیر داخل فایل:

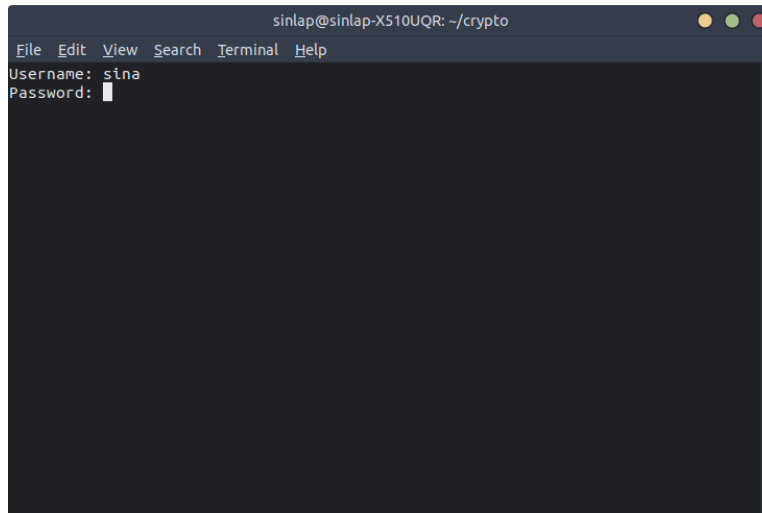
```
stricted mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

register.py  users.json X
D: > crypto > crypto > users.json > ...
1 [{"sina": "2f39a8118f343b674bf2d42e80a94846a23c32ae9b7da0ac066aa79ac50b46b6"}]
```

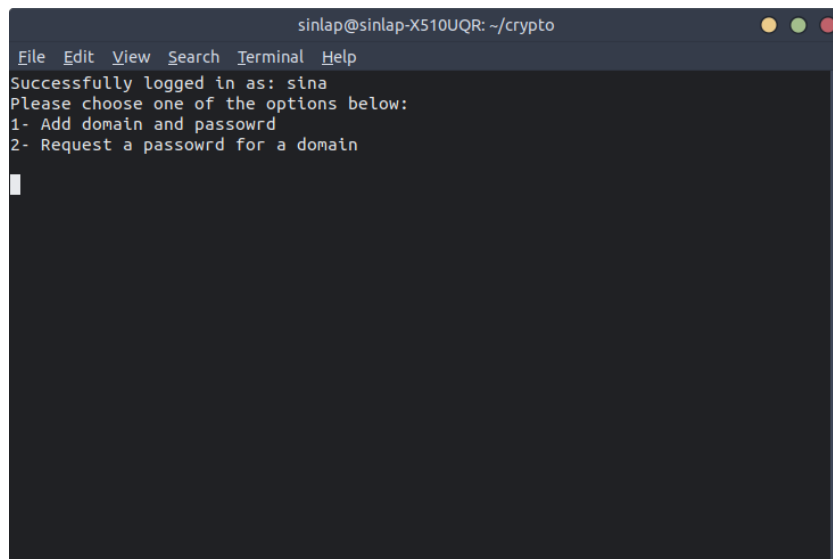
سپس به صفحه اول میرویم و عبارت Login را وارد میکنیم.

```
sinlap@sinlap-X510UQR: ~/crypto
File Edit View Search Terminal Help
Welcome! Please select one of the actions below:
1-Register
2-Login
3-Reset Password
Login
```

سپس از ما نام کاربری و رمز عبور خواسته میشود:



بعد از وارد کردن این اطلاعات تابع Login در فایل Login مقدار هش را بر اساس همان چیزی که قبلا گفته شده درست میکند و چک میکند که آیا مقدار هش تولید شده با مقدار هش داخل فایل برابر میباشد یا خیر. در صورت برابری با صفحه مقابل روبرو میشویم:



که به ما نشان میدهد تحت عنوان چه کاربری وارد شده ایم. در اینجا با نوشتن Add دامنه و رمز را وارد میکنیم.

```
sinlap@sinlap-X510UQR: ~/crypto
File Edit View Search Terminal Help
Enter domain: www.google.com
Enter password: 
```

در اینجا سایت گوگل با رمز salam وارد میشود. سپس دوباره وارد میشویم و رمز دیگری را برای سایت بینگ با رمز sina وارد میکنیم:

```
sinlap@sinlap-X510UQR: ~/crypto
File Edit View Search Terminal Help
Enter domain: www.bing.com
Enter password: 
```

این اطلاعات در فایل database.json ذخیره میشوند. ابتدا نام domain با سالت متفاوتی نسبت به رمز اضافه میشود و هش میشود. سپس با استفاده سالتی که بر اساس رمز کاربر تولید شده بود و خود رمز کاربر با استفاده از پروتکل PBKDF2 یک کلید 32 بیتی برای AES تولید میشود. رمز این سایت دوباره با سالت جمع میشوند و پد میشوند و با AES رمز میشوند و به صورت بیت در فایل ذخیره میشوند. این کارها در فایل userUI.py در تابع addPassword انجام میشود. نمونه فایل database.json بعد ذخیره سازی این 2 اطلاعات:

```
> crypto > crypto > { database.json > ...  
1 {"sina": {"f9a28964025ccf7cc2a70a13ba7f27ccd6c049f105e2eae522f8c0bfc3f9af65": "2Dhvn8YboE11RuT0TcW41yzFITnFdeL0weaUKHxgGY=",  
2 {"1fa9351ffe083d66f2f5587030933ec22e8f2db1c382b3bfe4bb175fd1b55334": "ZzAgFnZR2ikrE2oYKTEjHNVlqgVY6pt9rELTpr1iQSA="}}}
```

حال اگر بخواهیم این مقادیر را بیابیم دوباره وارد میشویم و اینبار عبارت Request را وارد میکنیم:

```
sinlap@sinlap-X510UQR: ~/crypto  
File Edit View Search Terminal Help  
Successfully logged in as: sina  
Please choose one of the options below:  
1- Add domain and passowrd  
2- Request a passowrd for a domain  
Request
```

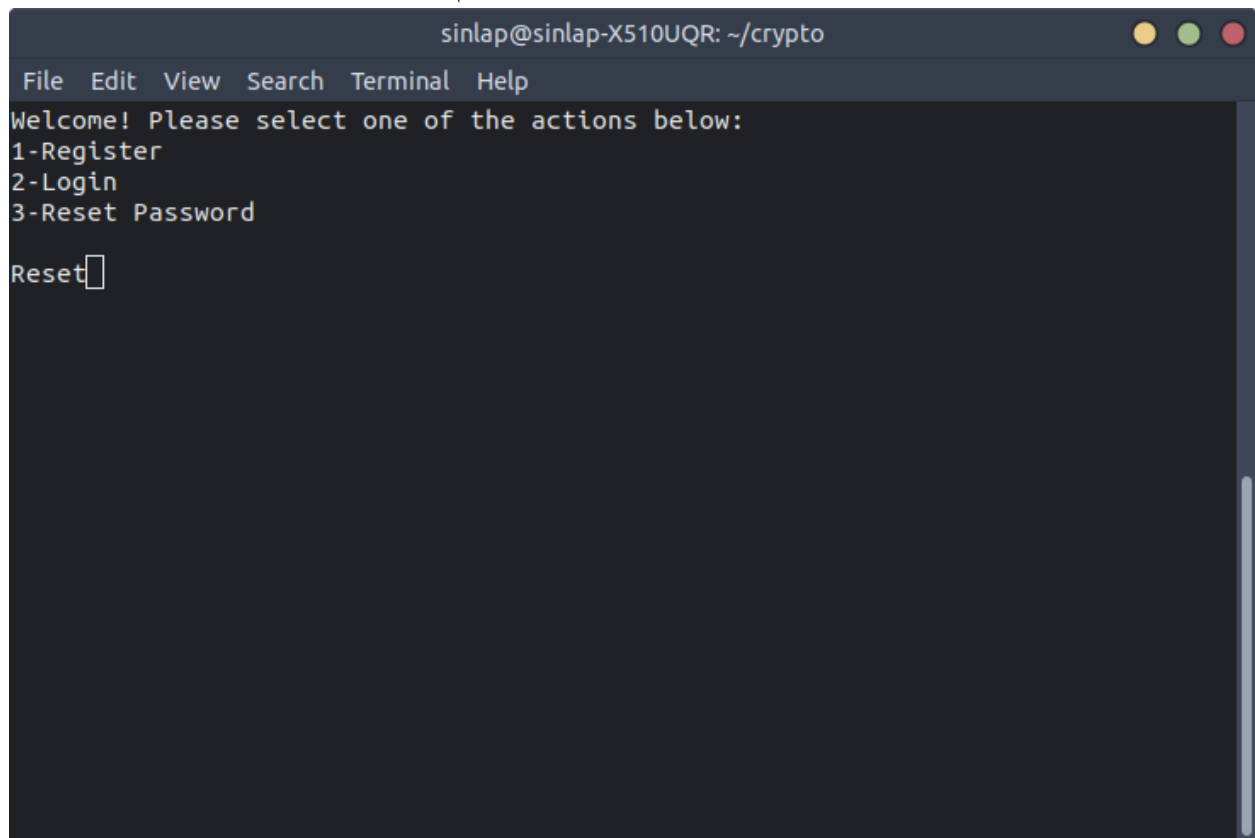
و برای هر دو سایت نتیجه مانند زیر میشوند:

```
sinlap@sinlap-X510UQR: ~/crypto  
File Edit View Search Terminal Help  
Enter domain name: www.google.com  
salam  
sinlap@sinlap-X510UQR:~/crypto$
```

```
sinlap@sinlap-X510UQR: ~/crypto
File Edit View Search Terminal Help
Enter domain name: www.bing.com
sina
sinlap@sinlap-X510UQR:~/crypto$
```

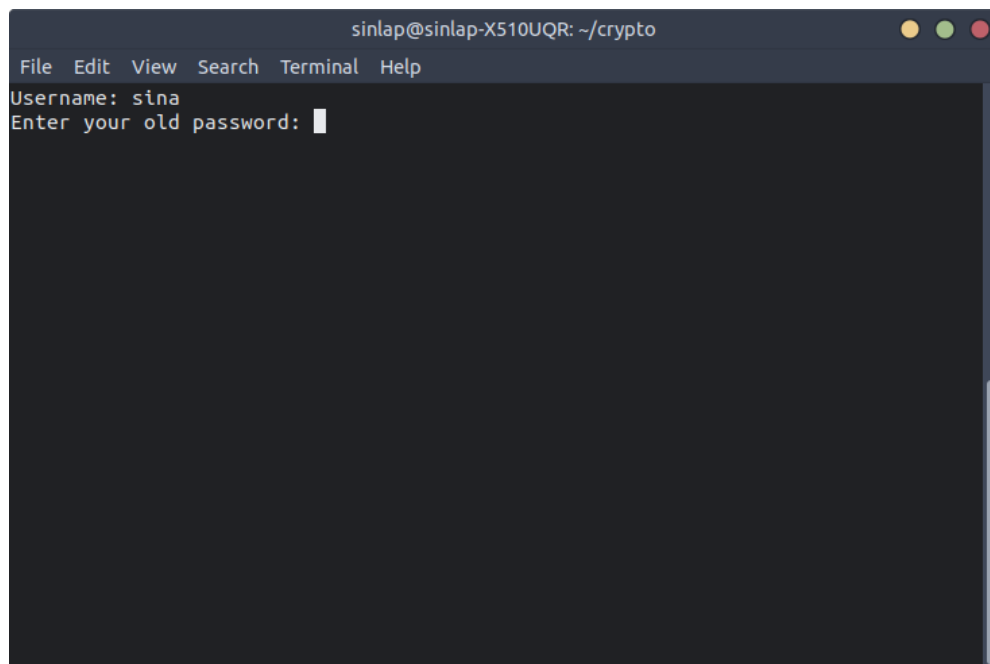
پیدا کردن رمز در فایل userUI در تابع returnPassword انجام میشود. در این قسمت سالت دو بار یکبار برای domain و یک بار برای رمز تولید میشود، domain و سالت مربوط به آن هش میشوند و مقدار مربوط به آن هش توسط کلید تولید شده بر اساس رمز کاربر و سالت آن باز میشود. در این حالت دیگر نیازی نیست که هیچ اطلاعات دیگری به جز سالتی که میخواهیم را بازگشایی بکنیم.

برای تغییر رمز در صفحه اول برنامه مقدار Reset را وارد میکنیم:



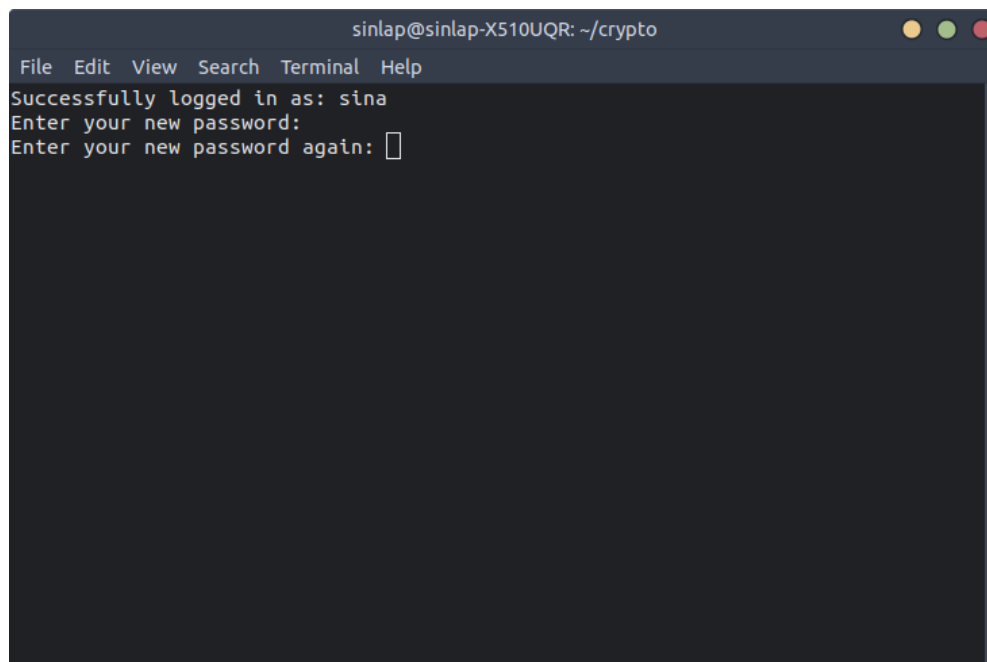
```
sinlap@sinlap-X510UQR: ~/crypto
File Edit View Search Terminal Help
Welcome! Please select one of the actions below:
1-Register
2-Login
3-Reset Password
Reset
```

بعد از آن نام کاربری و مقدار رمز قدیمی را وارد میکنیم.



```
sinlap@sinlap-X510UQR: ~/crypto
File Edit View Search Terminal Help
Username: sina
Enter your old password:
```

این اطلاعات دوباره به تابع Login داده میشود و اگر درست باشند با صفحه زیر مواجه میشویم:



```
sinlap@sinlap-X510UQR: ~/crypto
File Edit View Search Terminal Help
Successfully logged in as: sina
Enter your new password:
Enter your new password again: 
```

در اینجا رمز به code تغییر پیدا کرده است. این قسمت در فایل register.py در تابع userPassReset هندل میشود. در این قسمت اطلاعات مربوط به کاربر در فایل user.json با اطلاعات جدید جایگزین میشود و سپس همه اطلاعات مربوط به رمز domain ها بر اساس رمز قدیمی باز میشوند و بر اساس رمز جدید رمز گذاری شده و دوباره ذخیره میگردند. سالت مربوط به دامنه ها تغییری نمیکند اما سالت مربوط به رمز دامنه و رمز کاربر همگی دوباره ساخته میشوند. فایلی که به همراه این گزارش آمده است شامل کاربر sina با رمز code میباشد و دامنه ها نیز همان دامنه های بالا میباشند و آماده تست میباشد.

توجه: ممکن است در ویندوز با ارور نبود دستور clean مواجه شوید و آن به این دلیل است که این دستور برای ترمینال های UNIX موجود میباشد و صرفاً جنبه زیبایی دارد.