ضمیمه ب) چک لیست امنیتی OWASP

با توجه به این که یکی از روش‌های آزمون نرم‌افزار بررسی checklist های امنیتی است. بنابراین ما در این پروژه این checklist ها را بررسی کرده و تلاش کردیم تا برخی از آن‌ها را مرتفع سازیم و راه حل ارائه شده خودمان را تشریح کنیم.

Input Validation

- تمامی داده‌ها روی یک سیستم مورد اطمینان انجام شود.
  - در این پروژه ما از روش defense in depth استفاده کرده‌ایم بنابراین تمامی اطلاعات ورودی در سرور اصلی مورد بررسی مجدد قرار خواهند گرفت. همچنین اطلاعات فوق در لایه‌های بالاتر نیز مورد بررسی قرار میگیرند اما در با توجه به روش اتخاذ شده برای این روش محلی از اعراب باقی نیست.
- تفکیک اطلاعات به اطلاعات مورد اطمینان و غیر قابل اطمینان
  - در سیستم ما اطلاعات غیر قابل اطمینان عملا وجود نخواهد داشت بنابراین نگرانی از این بابت نخواهیم داشت.
- تهیه یک سیستم مرکزی برای بررسی اطلاعات ورودی
  - در سیستم ما تمامی اطلاعات ورودی از یک module مرکزی به نام marshal خواهد گذشت که تمامی مراحل marshalling و remarshaling توسط این سیستم رویت و مانیتور می‌شود.
- تمامی ورودی‌های invalid باید سیستم را به حالت input rejection ببرند
  - سیستم چجوری اگر اطلاعات ورودی برای سرور خوانا نباشد سیستم به حالت‌های خواستی طبق status code های http خواهد رفت.
- پشتیبانی از extended character ها
  - با توجه به این که زبان اصلی سیستم فارسی است بنابراین دایره characterهای مورد قبول سیستم بسیار بالاست.
- بررسی مقادیر مختلف header ها به ازای هر درخواست
  - با توجه به این که ما از headerهای خاصی برای ارتباط استفاده می کنیم بنابراین این اطلاعات به ازای هر درخواست مورد بررسی قرار می گیرند. یکی از مهم ترین مقادیر authorization key است.
- بررسی data range و data length
  - سیستم marshalling برای ما امکان مشخص کردن دو مورد ذکر شده را به ازای هر درخواست فراهم می سازد.

authentication and password management

- تفکیک درخواست‌های وابسته به auth و بقیه درخواست‌ها
  - تمامی api های expose شده توسط سیستم با استفاده از یک سیستم مرکزی محافظت می‌شوند و اگر اطلاعات مهمی بخواهد از سیستم گرفته شود فقط یک کاربر احراز هویت شده قادر به انجام این کار خواهد بود.
- تمامی مراحل رتق و فتق امور مربوط به auth باید روی یک سرور مورد اطمینان انجام شود

- در پروژه فعلی تمامی این اتفاقات در سرور انجام می‌شود و در صورت دخل و تصرف در این اطلاعات مربوط به auth در هر ماشین یا قسمتی از کار به غیر از سرور اصلی در صورت دریافت درخواست، آنها reject خواهد شد و مدیر سیستم از این اتفاق اطلاع خواهد یافت.

- تفکیک منطق auth از داده‌ها و بقیه اطلاعات سیستم
  - منطق auth صرفا در یک header از درخواست خلاصه می‌شود وbody در خواست که شامل اطلاعات اصلی برای حیات سیستم است، ارتباط خاصی با auth نخواهد داشت.
- فرایند هش کردن رمزها باید در یک سیستم مورد اطمینان و امن انجام شود.
  - تمامی فرایند hashing در سرور پیاده‌سازی شده‌اند و این اطلاعات توسط کلاینت‌ها از سرور گرفته می‌شود و مورد استفاده قرار میگیرد.
- فرایند تغییر پسورد به صورت ایمیلی و به وسیله یک لینک موقت انجام می‌شود به این ترتیب که ایمیل حاوی یک لینک موقت برای کاربر ارسال می‌شود و کاربر فقط از آن طریق قادر به تغییر پسوورد فراموش شده خواهد بود.
- تمامی لینک‌های مربوط به تایید اکانت‌های کاربری یک تاریخ انقضای کوتاه مدت دارند.

Communication Security

- تمامی درخواست‌ها طبق پروتکل ssl رمزنگاری خواهند شد.