**(( گزارش پروژه سوم مبانی امنیت اطلاعات ))**

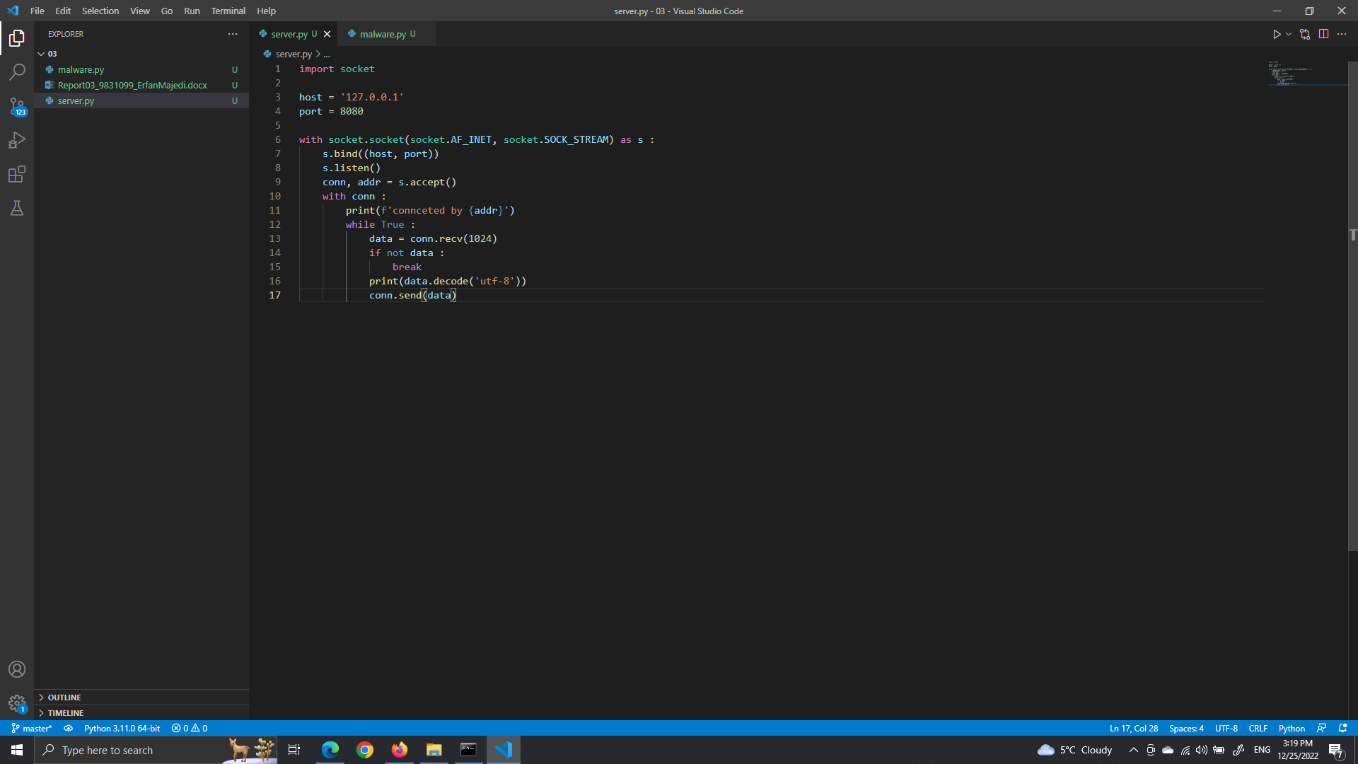
**گردآورنده : عرفان ماجدی 9831099**

**مبانی امنیت اطلاعات**

**دکتر شهریاری**

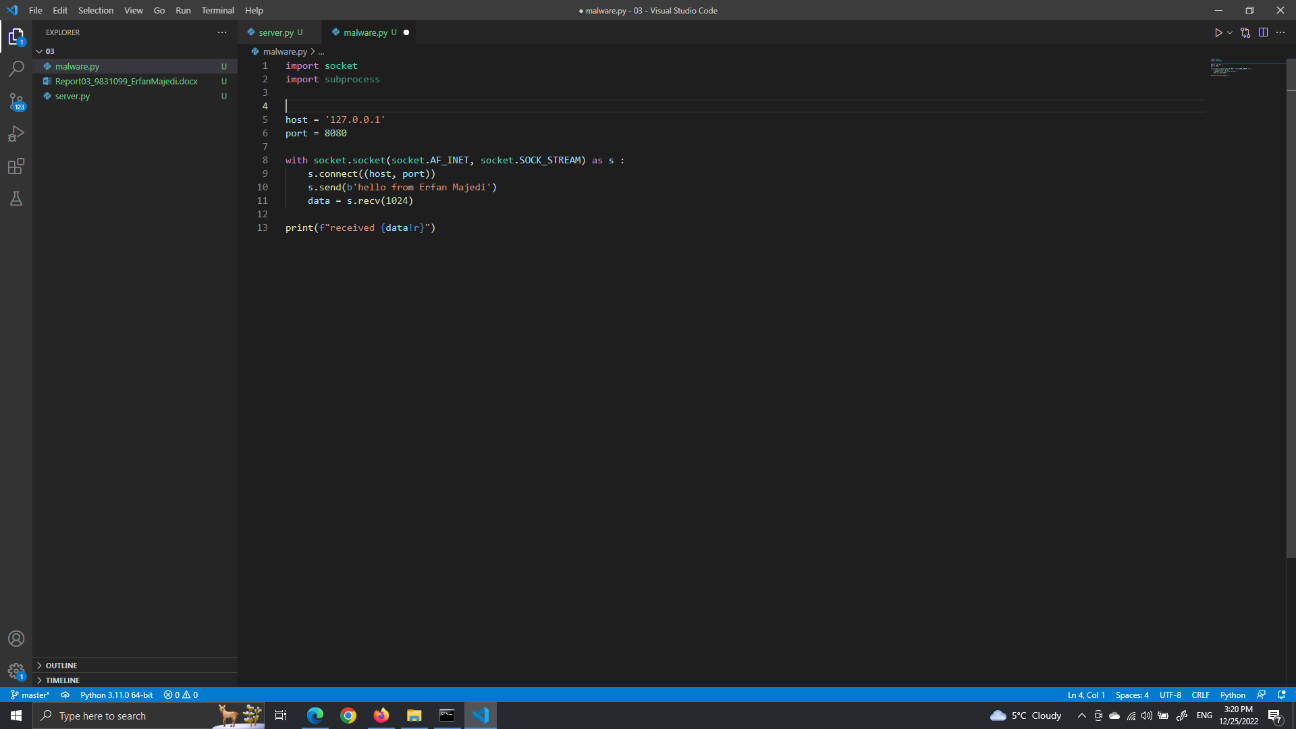
**بخش اول )**

**در این بخش ابتدا اسکرین شات کد server.py را می بینیم :**

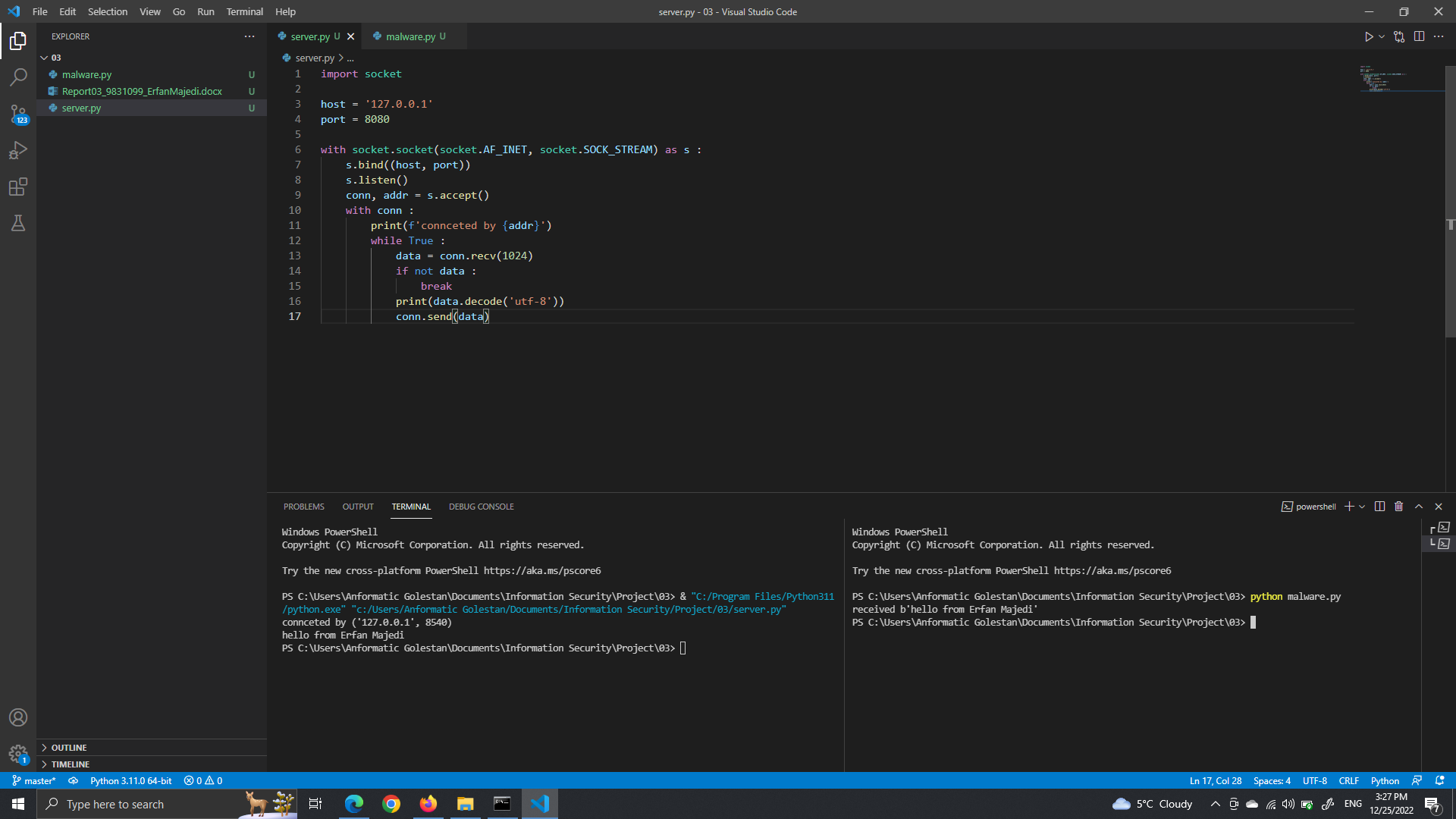
****

**همانطور که می بینید کاری که انجام دادیم ایجاد یک local host بوده است با یک شماره پورت 8080 و سپس با استفاده از socket این host و port را بهم وصل کردیم و یک پیام هم برای کاربر نمایش دادیم که با چه پورتی و ادرس آی پی به سرور وصل شدیم و تا زمانی که true باشد ارتباط ما داده دریافت می کند و آن را decode کرده و به کاربر نمایش می دهد .**

**حال به سراغ کد malware.py می رویم :**

****

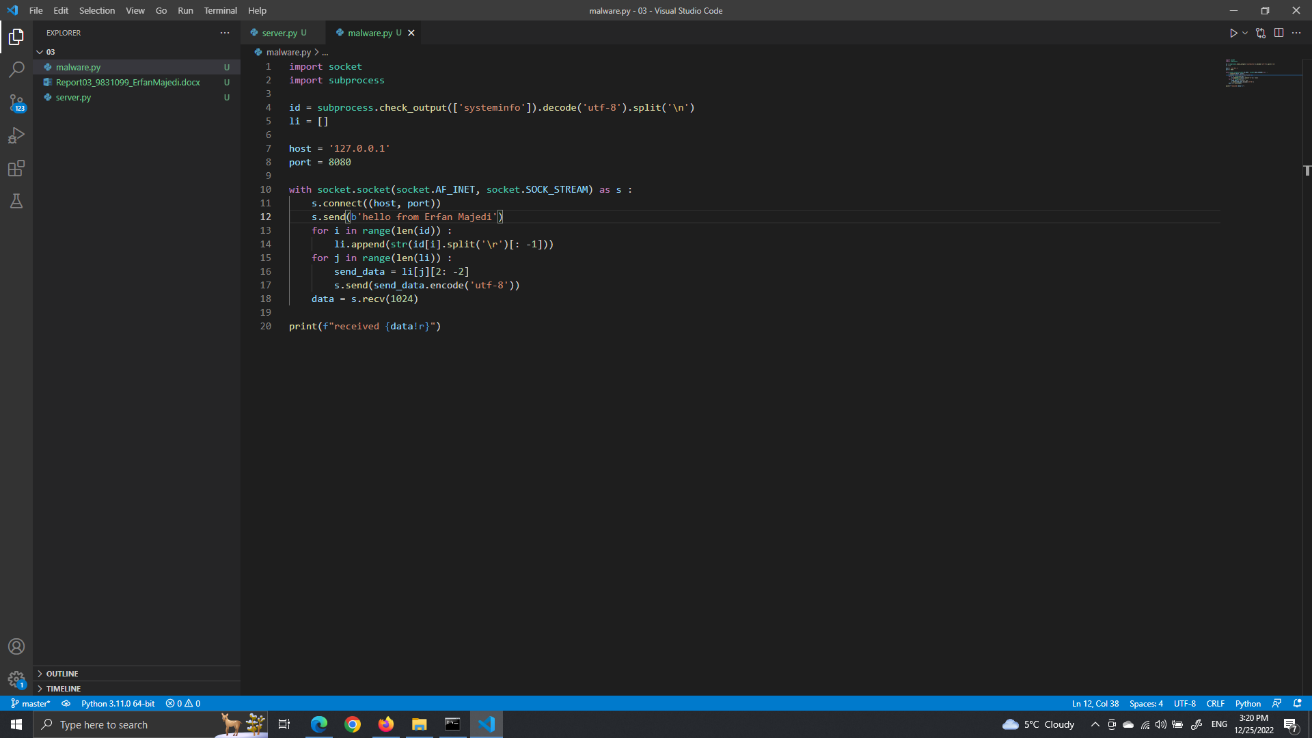
**در اینجا تنها تفاوتی که وجود دارد این است که با سرور ارتباط برقرار کرده و پیغام مناسب را چاپ کرده و آن را به سمت سرور می فرستد و سپس دوباره داده ای که سرور می دهد را گرفته و آن را به کاربر نمایش می دهد.**

****

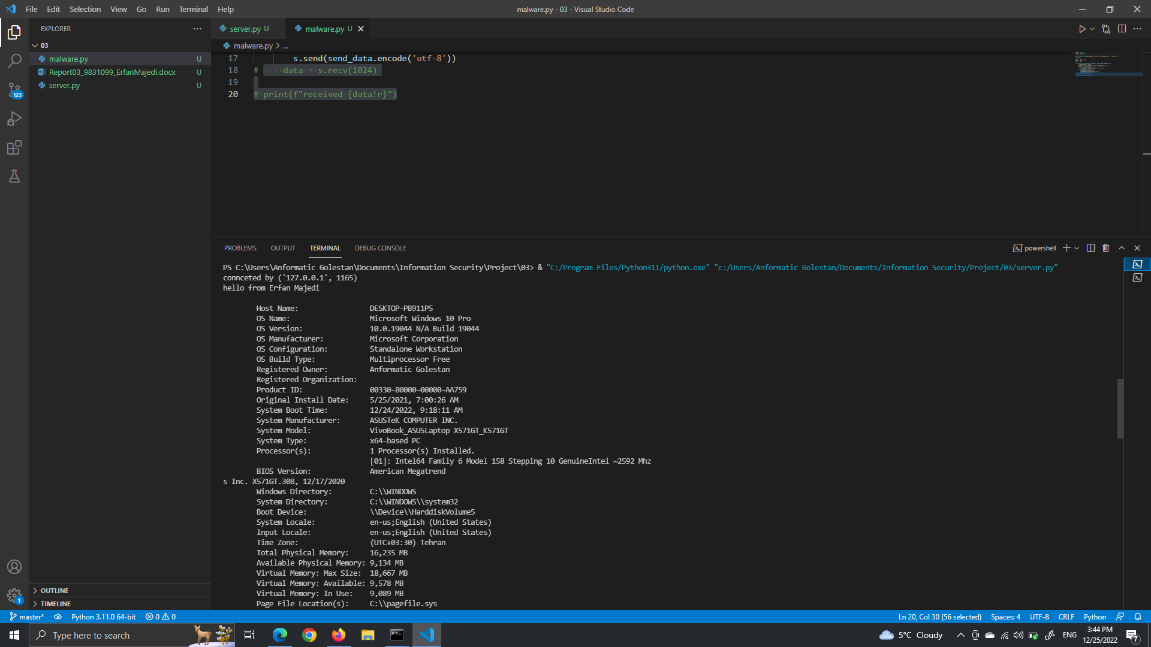
**نتیجه ی این کدها را نیز در تصویر بالا مشاهده می کنید .**

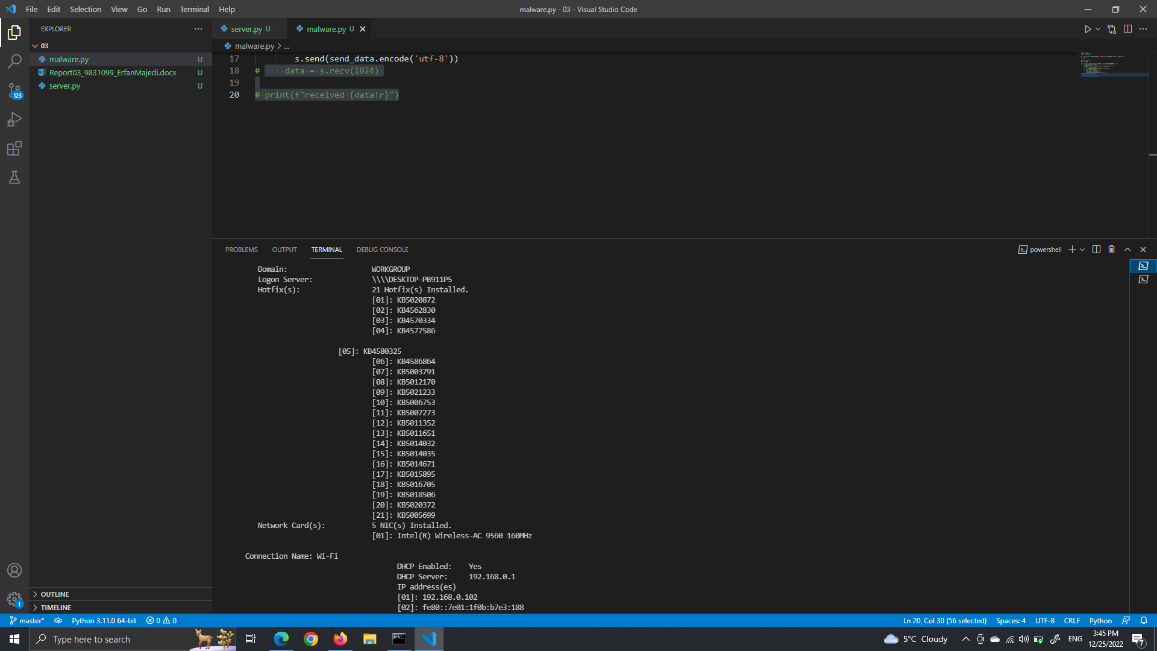
**بخش دوم )**

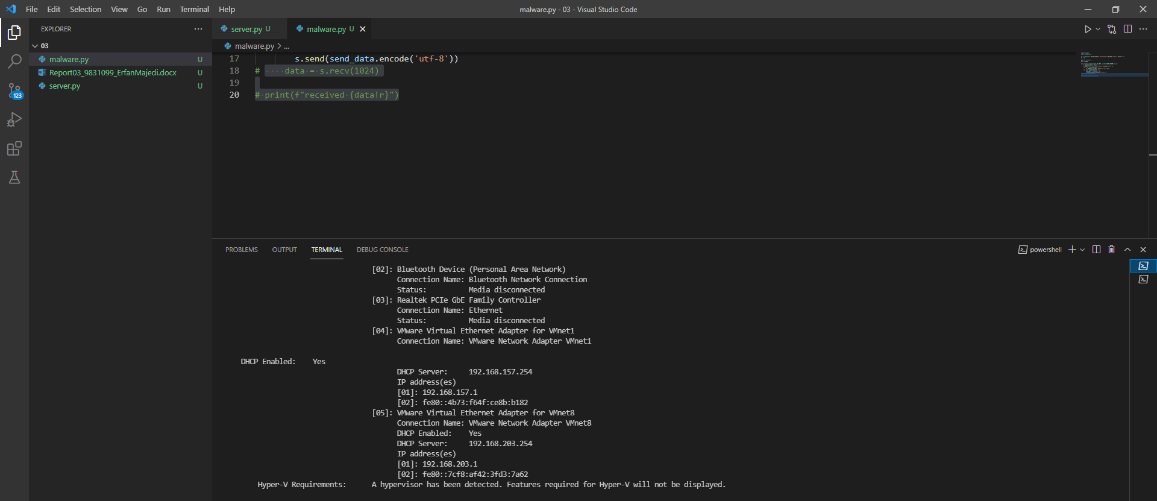
**در اینجا حرکتی که باید می زدیم این بود که کد malware.py را طوری تغییر دهیم که اطلاعات سیستم فرد قربانی را بدهد . کد به شکل زیر تغییر یافته است :**

****

**ابتدا کتابخانه ی subprocess را برای گرفتن اطلاعات فرد قربانی import کردیم و سپس یک متغیر id داریم که در واقع همان اطلاعات سیستم فرد قربانی است و سپس یک لیست خالی نیز ایجاد کردیم . حال در خط 13 یک حلقه روی طول id زدیم و محتویات آن را به صورت string تا ستون ماقبل اخری به لیست خود append کردیم . در مرحله ی بعد روی لیست خود یک حلقه ایجاد کردیم و داده هایی که میخواستیم برای سرور بفرستیم را در اینجا مشخص و داخل متغیر send\_data ریختیم و به صورت encode شده با utf-8 به سرور فرستادیم . نتیجه ی اجرای این کد را نیز در زیر می توانید ببینید :**

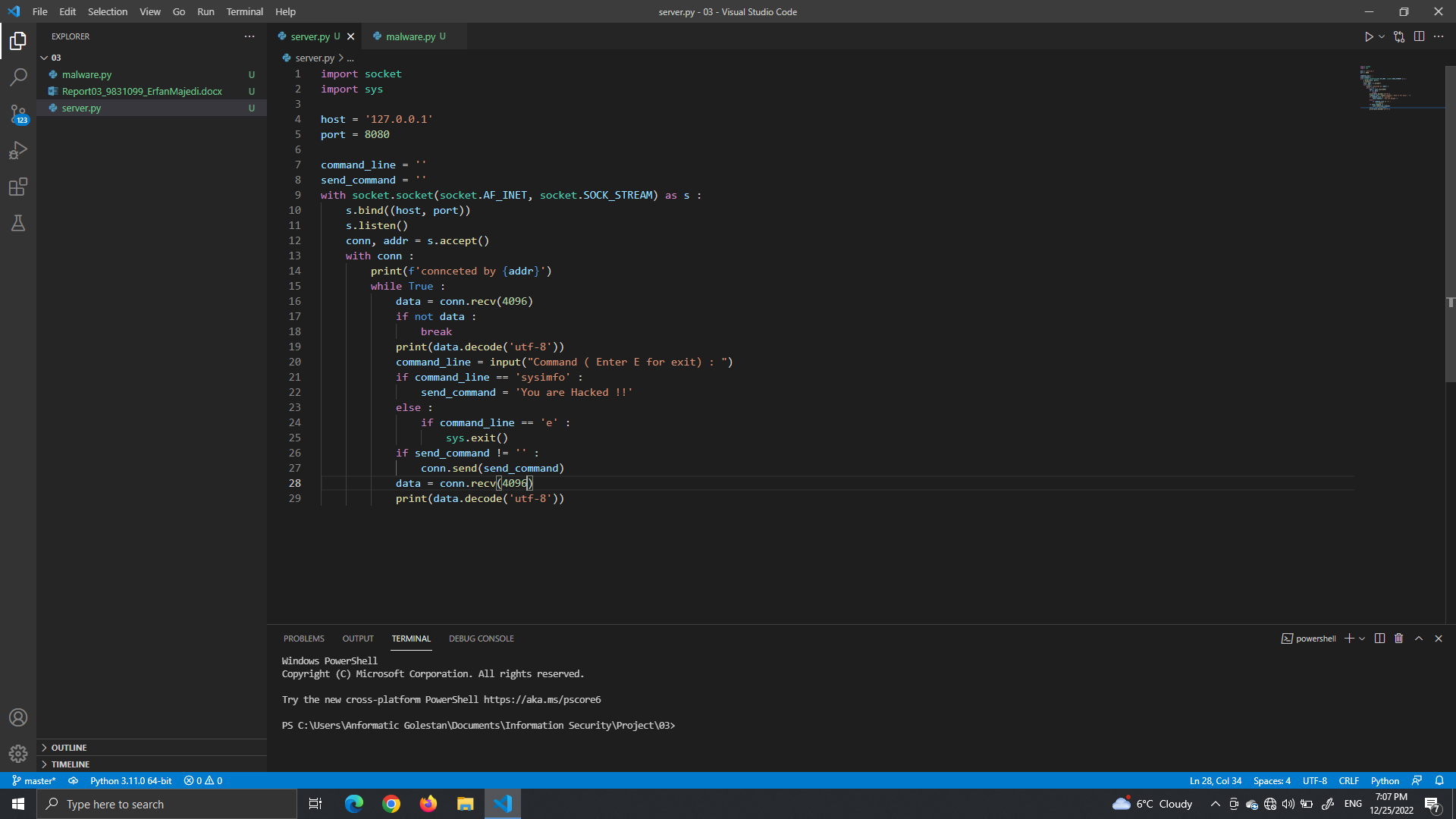
****

****

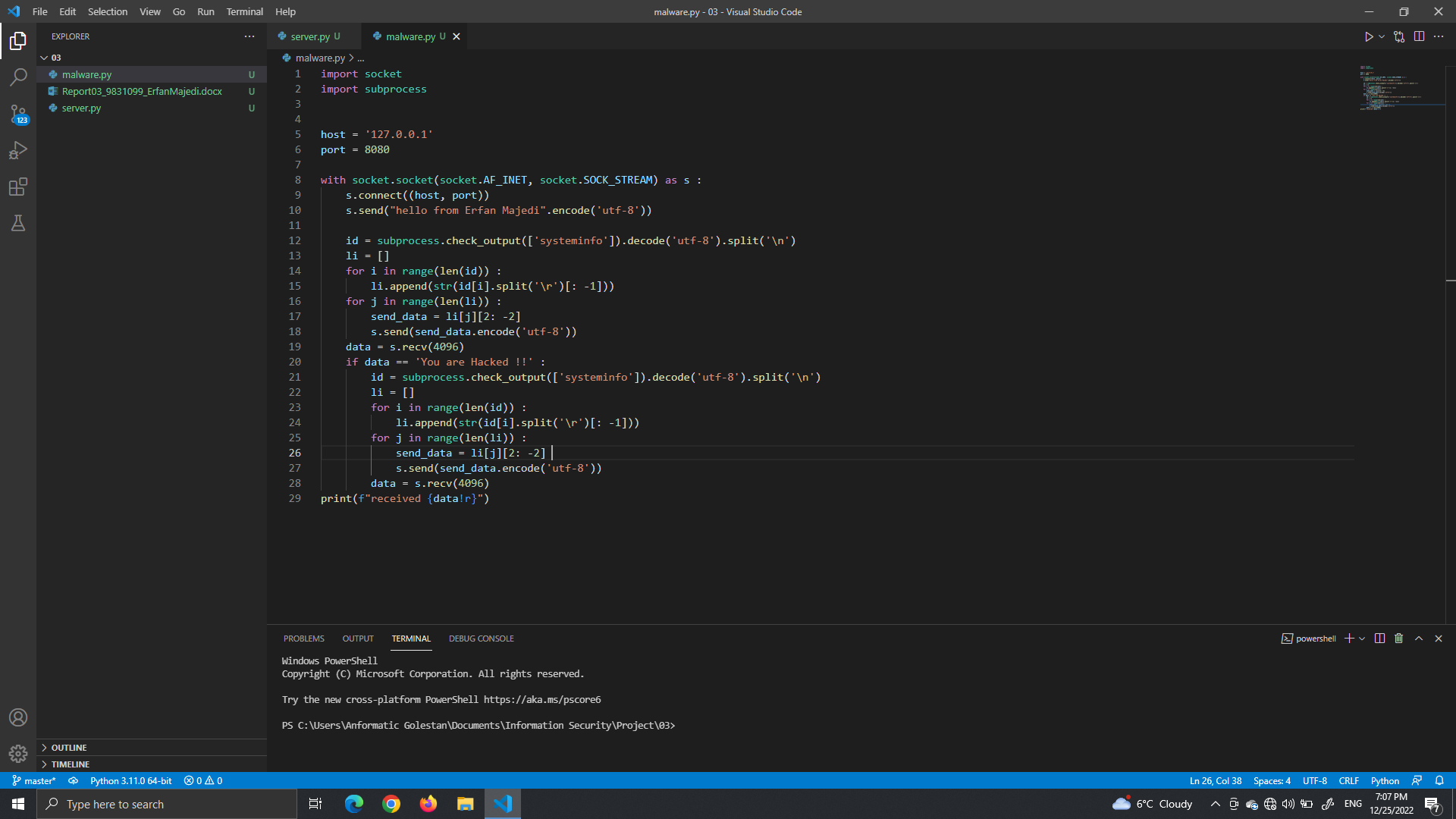
****

**بخش سوم )**

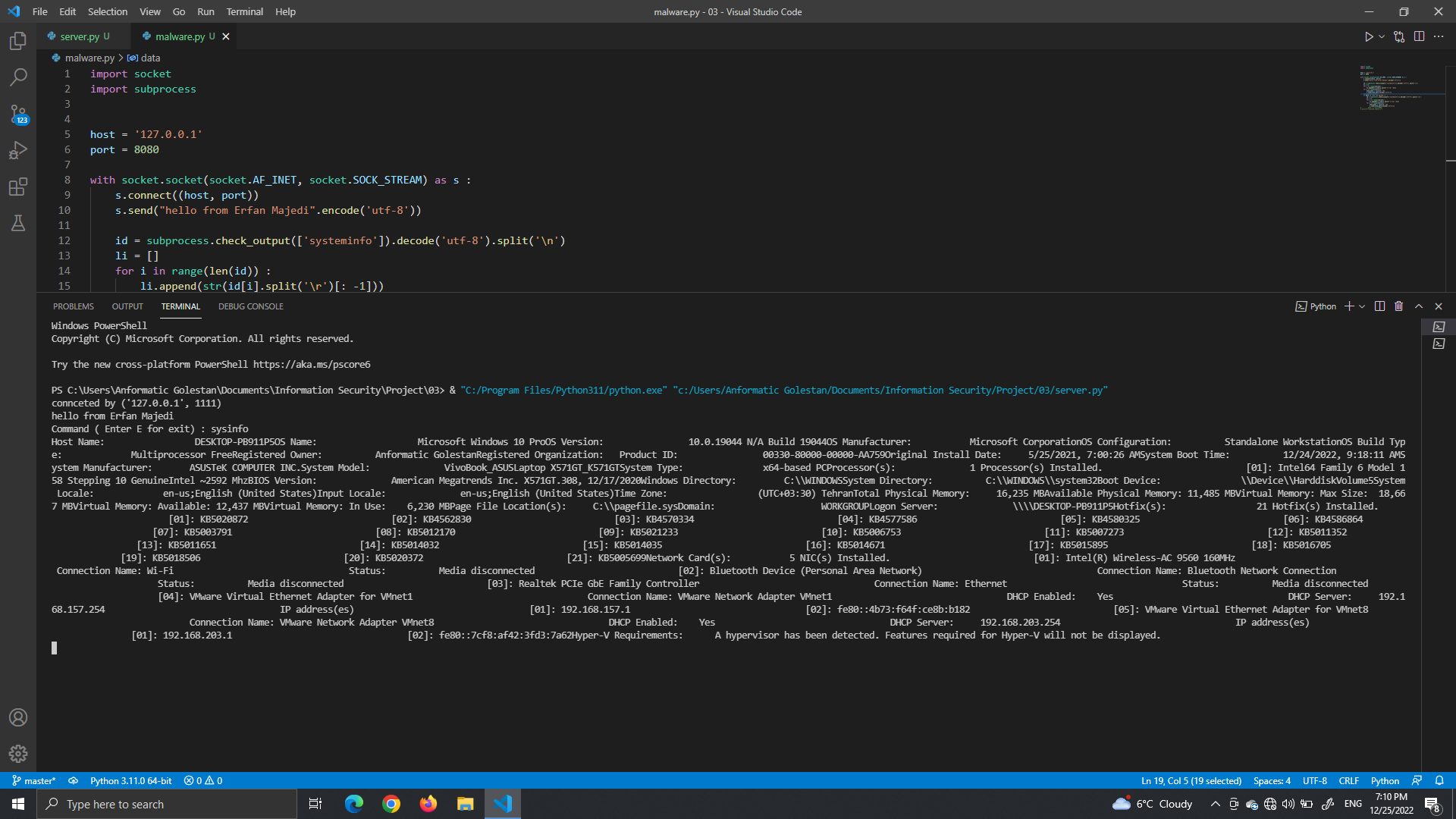
**در این قسمت باید یک سری تغییر در قسمت server.py ایجاد می کردیم تا ورودی از کاربر بگیریم و همچنین تغییر کمی را هم در قسمت malware.py ایجاد کردیم که باهم می بینیم :**

****

**تغییری که در server.py دادیم به این شکل است که دو string خالی به نام های command\_line و send\_command ایجاد کردیم سپس در خط 20 ام از کاربر خواستیم که command را وارد کند و آن را در متغیر command\_line ذخیره کردیم حال با توجه به دستور پروژه اگر کاربر sysinfo وارد می کرد باید عملیات گرفتن اطلاعات سیستم فرد قربانی اغاز می شد و به او یک پیغام نیز برای malware.py فرستاده می شود که در خط 22 قرار دارد حال اگر کاربر e را وارد می کرد از اجرای کد خارج می شدیم. حال در خط 26 گفتیم اگر send\_command خالی نباشد آن را برای malware.py بفرستید و سپس داده را از آن سمت گرفته و نمایش می دهد. حال به سراغ تغییرات سمت malware.py می رویم :**

****

**در این جا کاری که کردیم یک شرط قرار دادیم که اگر داده ی دریافت شده برابر با همان send\_command بود دوباره اطلاعات را بسازد و به همان روندی که در بخش 2 دیدیم به سمت سرور بفرستد. نتیجه ی این بخش هم به صورت زیر است :**

****