پروژه کارگاه امنیت نرم‌افزار

استاد :

جناب آقای دکتر محمودی

جناب آقای رجبی نسب

دانشجو :

سید عمید اسدالهی مجد (۹۴۱۲۲۱۱۰۲)

پاییز ۹۹

چکیده

این نرم‌افزار به منظور جستجوی سری‌های IP روی یک شبکه و یافتن آدرس (مودم) هایی که در آن‌ها یک پورت خاص (۹۱۷) باز بوده و دارای آسیب‌پذیری خاصی می‌باشند، پیاده سازی شده است. برای پیاده سازی از زبان برنامه‌نویسی پایتون[1] و کتاب‌خانه اصلی سوکت[2] استفاده شد.

این برنامه (checker.py) با دریافت ورودی چند خطی از کاربر به فرمت x.x.x.0/24 سری‌های IP را دریافت کرده و پس از تشکیل این سری‌ها و بررسی تست ping، به بررسی باز بودن و امکان برقراری ارتباط در پورت مورد نظر را با هر یک از IP ها به صورت موازی انجام می‌دهد. در انتها لیستی مرتب شده از IP هایی که با پورت مورد نظر قابل دسترسی بودند (آسیب‌پذیر بودند) را در فایل vulnerable_ip_list.txt ذخیره می‌نماید.

* همچنین یک فایل با نام test_servers.py نیز وجود دارد که صرفاً در طراحی و توسعه این نرم‌افزار از آن استفاده شده است. با اجرای این فایل پایتون، تعدادی سرور ساده در سری IP های 127.0.0.0/24 ایجاد می‌شوند که IP های زوج، با پورت باز مورد نظر (۹۱۷) در نظر گرفته می‌شوند.

[1] Python
[2] Socket

شرح توابع اصلی

توابع اصلی استفاده شده در این نرم‌افزار در ادامه شرح داده خواهند شد. تابع اصلی و اجرا کننده نهاییmain در انتها توضیح داده شده است.

get_input

وظیفه دریافت ورودی از کاربر در سطرهای مختلف و سری IP های مختلف بر عهده این تابع است. این تابع پس از ۲ بار فشردن دکمه Enter، آرایه‌ای شامل مقادیر ورودی در فرمت مناسب را برمی‌گرداند. به طور مثال ورودی 127.0.0.0/24 به صورت 127.0.0 به عنوان معرف این سری IP، در نظر گرفته می‌شود.

create_ip_list

این تابع با دریافت آرایه شامل معرف‌های سری IP های ورودی کاربر، لیستی از تمام IP های موجود در هریک از سری‌ها (از ۱ الی ۲۵۵) برمی‌گرداند.

ping_check

این تابع یک آدرسIP را به عنوان ورودی دریافت می‌کند و سپس تست ping را روی آن انجام می‌دهد. تست ping با اجرای دستور ping که به صورت پیشفرض در سیستم‌های عامل وجود دارد، (با فرمت مناسب هر سیستم‌عامل) در یک فرایند زیرین[3] انجام می‌شود. خروجی این تابع مقدار بولین بوده که True به معنای موفقیت‌آمیز بودن تست ping آن IP است.

port_check

این تابع یک آدرس IP و یک شماره پورت را به عنوان ورودی دریافت می‌کند و سپس با استفاده از کتاب‌خانه socket اقدام به ایجاد ارتباط با آن IP در پورت ورودی می‌نماید. اگر ارتباط با موفقیت ایجاد شود بنابراین پورت مورد نظر قابل استفاده برای ارتباط آن آدرس IP بوده و مقدار بولین True برگردانده می‌شود. در غیر اینصورت مقدار False خروجی تابع خواهد بود.

[3] Sub Process

sort_vulnerable_ip_list

این تابع یک لیست IP را دریافت کرده و سپس کلید مرتب‌سازی آرایه مقدار hex آن IP است که با استفاده از کتاب‌خانه socket بدست می‌آید. در نهایت لیست مربت شده به عنوان خروجی بازگردانی می‌شود.

save_vulnerable_ip_list

این تابع با دریافت یک لیست IP، آن‌ها را در فایل vulnerable_ip_list.txt و در کنار نرم‌افزار اصلی ذخیره می‌کند.

main

این تابع با استفاده از توابع یادشده بالا عملیات را از ورودی تا خروجی مدریت و اجرا می‌نماید. همچنین این تابع دو مقدار ورودی دریافت می‌کند که مقدار پورت، همان پورت مورد نظر برای بررسی است و مقدار max_workers تعداد کارگر (Thread) های موازی در نظر گرفته شده را مشخص می‌نماید.

مراحل اجرا :

1. دریافت سری IP های مورد نظر از کاربر با استفاده از تابع get_input
2. ساختن لیستی شامل تمام IP های موجود در هر سری IP با استفاده از تابع create_ip_list
3. با استفاده از کتاب‌خانه concurrent در پایتون عملیات چک کردن هر IP با استفاده از تابع (check__) به صورت موازی انجام می‌شود. تابع check__ خود به ترتیب به اجرای تابع ping_check و سپس در صورت موفقیت آمیز بودن تست ping، تابع port_check را روی IP ورودی اجرا می‌کند و در صورت موفقیت آمیز بودن، آدرس IP ورودی را برمیگرداند. در غیر این صورت مقدار null برگردانده می‌شود.
4. پس از اتمام عمل جستجوی هر IP، در صورت null نبودن خروجی، آن IP به لیست آسیب‌پذیر اضافه می‌شود.
5. مرتب‌سازی لیست IP های آسیب‌پذیر بدست آمده با استفاده از تابع sort_vulnerable_ip_list
6. ذخیره لیست IP های مرتب‌شده آسیب‌پذیر با استفاده از تابع save_vulnerable_ip_list