۲۰

نشریه تخصصی امنیت سایبری مرکز آپا دانشگاه کردستان
شماره پنجم/ بهار ۹۹

- GitHub و امنیت
- امنیت در سرویس‌های ابری
- دفترچه تقلب Wireshark
- معرفی نسل‌های آینده فایروال
- ارزیابی امنیتی اپلیکیشن‌های موبایلی با ابزار MOBSF
- کدنویسی امن php - اعتبارسنجی و فیلتر داده‌های ورودی
- حوادث امنیت سایبری و اقدامات ضروری برای مقابله با آن‌ها

درباره

مرکز آپا دانشگاه کردستان

آپا مخفف عبارت آگاهی‌رسانی، پشتیبانی و امداد رخدادهای رایانه‌ای است و معادل بومی اصطلاح CSIRT می‌باشد. مرکز آپا دانشگاه کردستان، در راستای انجام فعالیت‌های خود در زمینه آگاهی و اطلاع رسانی، با بکارگیری نیروهای متخصص و پتانسیل‌های پژوهشی در استان کردستان اقدام به انتشار نشریه‌ای الکترونیکی در حوزه امنیت فضای سایبری نموده است.

مخاطبان اصلی نشریه کارشناسان و متخصصان فناوری اطلاعات و شبکه، دانشجویان و علاقمندان فضای سایبری است. مطالب این نشریه عموماً محورهای زیر را شامل می‌شود:

- اطلاع‌رسانی رخدادهای اخیر فضای سایبری
- آگاهی‌رسانی نسبت به آخرین تهدیدات و آسیب‌پذیری ابزارهای فضای مجازی
- آموزش‌های تخصصی و عمومی در جهت ارتقاء دانش امنیت

شایان ذکر است، ویرا، اسم نشریه، واژه‌ای در زبان کردی به معنی صاحب‌فکر و هوشمند است.

سردبیر: هادی گلباغی
سردبیر فنی: محمد حبیبی
ویراستار: نازیلا خسروی
طراحی و صفحه‌آرایی: بهار سرسیفی
نویسندگان (به‌ترتیب مطالب):
آرش یونسی/ محمد ساروقی/ سیروان اله‌ویسی/ هادی گلباغی/ محمد حبیبی/ نازیلا خسروی/ کسرا ریسمانچی/ محمدجواد عبدالملکی/ آزین زارعی
با تشکر از: مسلم حقیقیان

تلفن مرکز: ۰۸۷۳۳۶۱۱۴۱۵
نشانی مجله: کردستان، سنندج، بلوار پاسداران، دانشگاه کردستان، دانشکده مهندسی، ساختمان شماره ۳، طبقه همکف، مرکز آپا
وب‌سایت: www.cert.uok.ac.ir
ایمیل: apa@uok.ac.ir

راهنمایی:

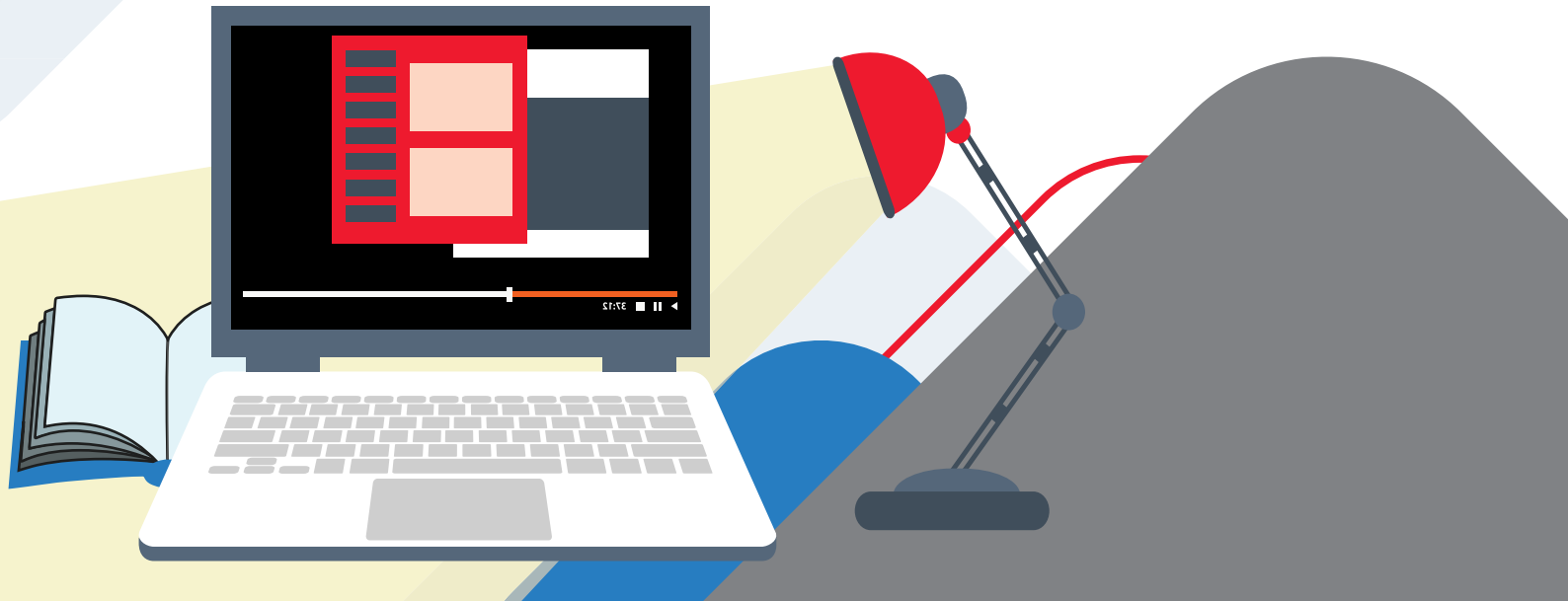
- در فهرست مطالب می‌توانید با کلیک بر روی هریک از بخش‌ها و مطالب به صفحه مورد نظر منتقل شوید.
- با کلیک بر روی QR کدها می‌توانید مستقیما به لینک‌ها منتقل شوید.

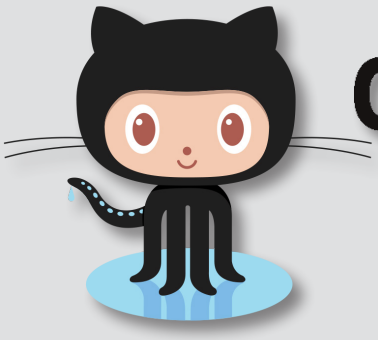
فهرست

مطالب

Tutorials

مقاله‌های آموزشی





GitHub و امنیت

تهیه و تدوین: آرش یونسی

معرفی GitHub (گیت‌هاب)

گیت‌هاب درواقع یک پلتفرم همکاری برای برنامه‌نویسان است که به برنامه‌نویسان اجازه می‌دهد باهم روی پروژه‌ها کار کنند، کدهای خود را از طریق گیت‌هاب به اشتراک بگذارند و در صورت نیاز از کدها، پروژه‌ها و کتابخانه‌هایی که توسط برنامه‌نویس‌های دیگر توسعه داده شده‌اند استفاده کنند.

امنیت در GitHub

با توجه به گستردگی GitHub در بین برنامه نویسان، کدنویس‌ها و توسعه‌دهندگان به‌طور تقریبی می‌توان گفت همه وب‌سایت‌ها، اپلیکیشن‌های موبایل، ربات‌های نرم‌افزاری و ... حداقل یک‌بار از کدهای متن‌باز GitHub استفاده کرده‌اند. لزوم برقراری امنیت در هنگام توسعه نرم‌افزارهای متن‌باز و همچنین رعایت موارد امنیتی هنگام استفاده از این نرم‌افزارها از اهمیت ویژه‌ای برخوردار است.
در ادامه راه‌هایی را بررسی می‌کنیم که تا حد زیادی به برقراری امنیت در استفاده از GitHub به ما کمک خواهند کرد.

راه‌های امن کردن حساب کاربری GitHub

۱. ذخیره نکردن اطلاعات ورود داخل فایل‌هایی که بر روی GitHub آپلود می‌شوند

جستجوها نشان می‌دهد که بسیاری از کاربران به‌اشتباه و یا از روی ناآگاهی اطلاعات ورود خود شامل نام‌های کاربری یا رمزهایشان را داخل فایل‌ها ذخیره کرده و بر روی GitHub آپلود کرده‌اند.
چنین اطلاعاتی می‌تواند به‌راحتی توسط افراد سودجو استفاده شود و شما را با مشکلات مختلفی مواجه کند.

راه‌حل‌ها:

- فایل‌های خود را قبل از ارسال به GitHub با ابزار git-secrets بررسی کنید.
- فایل‌های خود را بعد از ارسال با ابزارهای GitRob و truffleHug بررسی کنید.
- از ابزار Vault برای ذخیره اطلاعات حساس استفاده کنید.
- در صورت یافتن اطلاعات حساس بعد از آپلود، تمامی اطلاعات یافت شده از داخل فایل‌ها را حذف کنید، تاریخچه‌ها را پاک‌سازی کنید و از همه مهم‌تر اعتبار توکن‌ها، یوزرنیم‌ها و پسوردها را از بین ببرید.

۲. کنترل دسترسی‌ها

- احراز هویت دو مرحله‌ای را برای تمامی حساب‌های GitHub فعال کنید.
- دستگاه‌هایی که به حساب GitHub شما دسترسی دارند را امن کنید.
- دسترسی کاربرانی که همکاری شما با آن‌ها پایان‌یافته است را غیرفعال کنید.
- سطح دسترسی اعضای تیم را به آنچه نیاز دارند، محدود کنید.

۳. استفاده صحیح از کدها و برنامه‌های داخل GitHub

به یاد داشته باشید که برنامه‌ها و قطعه کدهایی که داخل GitHub موجود می‌باشند توسط برنامه نویسان نوشته‌شده‌اند نه خود تیم GitHub، هنگام استفاده از این کدها / برنامه‌ها:

- دسترسی‌های کد / برنامه را بررسی کنید و به بیشتر از آنچه نیاز دارند دسترسی ندهید.
- اعتبار برنامه‌نویس / سازمان تولیدکننده را بررسی کنید.
- وضعیت امنیتی کد / برنامه را بررسی کنید زیرا که حفره‌های امنیتی داخل این کدها / برنامه‌ها موجب نفوذ به داخل برنامه شما خواهد شد.

٤. انجام تست امنیتی هنگام ایجاد تغییرات در فایل‌ها

بعد از هر تغییر در فایل‌های پروژه (GitHub Push Request) از ابزارهای زیر برای تست مجدد فایل‌ها استفاده کنید تا از ایجاد آسیب‌پذیری جلوگیری شود.

- **SonarCloud**: بررسی کیفی کدها
- **CodeClimate**: بررسی جنبه‌های مختلف کد به‌صورت خودکار
- **Synk**: بررسی آسیب‌پذیری کدها، وابستگی‌ها و کتابخانه‌های استفاده‌شده

٥. افزودن فایل SECURITY.md

- **سیاست افشای اطلاعات**

بررسی‌ها نشان می‌دهد که تنها ۲۱ درصد از پروژه‌هایی که راهی برای افشای عمومی تعیین نکرده‌اند، در مورد آسیب‌پذیری‌های امنیتی به آن‌ها اطلاع داده‌شده است، درحالی‌که این عدد برای پروژه‌هایی که راه و روشی را برای افشای اطلاعات در خصوص آسیب‌پذیری‌ها و باگ‌های امنیتی تعیین کرده‌اند، ۷۳ درصد است. درنتیجه بسیار پراهمیت است که تعیین کنید در صورت یافتن آسیب‌پذیری توسط افراد مختلف (امنیت کارها، برنامه‌نویس‌ها و ...)، آن‌ها چگونه و به چه شکلی آن را به شما اطلاع دهند.

- **سیاست به‌روزرسانی امنیتی**

هر روزه از نرم‌افزارهای مختلف آسیب‌پذیری یافت می‌شود. زمانی که در برنامه یا کتابخانه شما آسیب‌پذیری یافت می‌شود شما وظیفه‌دارید که به کاربران خود در مورد آسیب‌پذیری اطلاع دهید، کاربران شما ممکن است از کد متن‌باز شما در پروژه‌ها و سیستم‌های مهمی استفاده کنند که آسیب‌پذیری ایجادشده باعث ضررهای جدی به آن‌ها شود.

برای به اشتراک گذاشتن اطلاعات مربوط به آسیب‌پذیری‌ها و به‌روزرسانی‌ها، ازجمله شدت آسیب‌پذیری، خطر ایجادشده و نحوه انتقال به نسخه ثابت کد خود، باید یک فرایند تعریف‌شده داشته باشید. این فرایند را به‌صورت پیش‌فرض تعریف کنید تا اطلاعات به کاربران شما منتقل شود و به آن‌ها اجازه دهید هر چه سریع‌تر درباره آسیب‌پذیری‌های امنیتی جدید در هنگام یافتن و رفع آن‌ها به‌روز شوند. برای مثال می‌توانید یک صفحه روی GitHub یا وب‌سایت خود ایجاد کنید و به‌روزرسانی‌های امنیتی را به اشتراک بگذارید. در صورتی که ایمیل کاربران خود را دارید ارسال ایمیل به آن‌ها در کنار صفحه وب یکی از بهترین گزینه‌ها می‌باشد.

- **پیکربندی امنیتی**

کاربران معمولاً درک کمی نسبت به امنیت دارند، بنابراین شما باید پیشنهاد تنظیماتی را در پروژه خود لحاظ کنید که کاربران با انجام آن‌ها حداقل‌های امنیت را در خصوص کد شما به دست بیاورند.

پیشنهاد تنظیماتی مانند فعال کردن HTTPS، جایگزین کردن گذرواژه‌های پیش‌فرض، اضافه کردن captcha و در کل هر پیشنهادی که به تأمین امنیت کمک کند.

- **شکاف‌های امنیتی شناخته‌شده و پیشرفت‌های آینده**

معمولاً به‌ندرت پیش می‌آید که در پروژه‌ی خود تمامی مسائل امنیتی را رعایت کنید.

به کاربران خود در مورد کنترل‌های امنیتی که در حال حاضر وجود ندارند اطلاع‌رسانی کنید تا آگاهانه در مورد پروژه شما تصمیم بگیرند. حتی ممکن است کاربران به شما در این مورد پیشنهاداتی ارائه دهند اما در نظر داشته باشید که مهاجمان نیز می‌توانند از گفته‌های شما علیه شما استفاده کنند.

درنتیجه اطلاع‌رسانی‌های شما نباید به‌گونه‌ای باشد که باعث سوءاستفاده مهاجمان شود و از طرف دیگر باید کاربران را تا جایی که امکان دارد از عدم وجود کنترل‌های امنیتی مطلع کنید.

٦. به‌روزرسانی کلیدهای SSH و توکن‌های دسترسی

دسترسی در GitHub با استفاده از کلیدهای SSH و توکن‌ها صورت می‌گیرد. درصورتی‌که این اطلاعات به سرقت برود مهاجمان به اطلاعات شما دسترسی خواهند داشت، درنتیجه همواره به‌صورت دوره‌ای این کلیدها و توکن‌ها را تازه کنید.

۷. استفاده از پیشنهادات GitHub برای نیازهای امنیتی خود

بسته به نوع پروژه یا مقررات سازمانی شما، ممکن است به نرم‌افزاری محدود شوید که فقط به‌صورت محلی قابل‌اجرا باشد یا شاید محدودیت‌ها در محل ذخیره کد منبع شما باشد ویا اینکه چه سازمان‌های دیگری می‌توانند به آن دسترسی داشته باشند.

این‌یک محدودیت مشترک برای مؤسسات مالی، ادارات دولتی یا سایر صنایع با نظارت دقیق است.

اما این بدان معنی نیست که شما نمی‌توانید از GitHub استفاده کنید.

نگاهی به پیشنهاد GitHub Enterprise بیندازید که به شما امکان می‌دهد مخازن GitHub را به‌طور کامل در سازمان خود به‌صورت محلی میزبانی کنید. این بدان معناست که شما می‌توانید از اینترنت جدا شوید و هنوز هم در داخل مخازن GitHub Enterprise به پروژه‌های خود دسترسی داشته باشید. حتی خود GitHub هم به پایگاه کدهای شما دسترسی ندارد.

۸. معرفی دو ابزار کاربردی

- git-secret
- Gitrob

git-secret

مشکلی که همواره در توسعه نرم‌افزارها وجود داشته ذخیره فایل‌های حساس مانند فایل‌های حاوی پسوردها، توکن‌ها، اطلاعات حساس مربوط به دیتابیس و ... بوده است.
شاید فکر کنید که چون repository (به زبان ساده به‌جایی از Git که فایل‌های پروژه‌ها در آنجا قرار دارند repository گفته می‌شود) شما خصوصی (private) است، پس مشکلی برای ذخیره چنین اطلاعاتی وجود ندارد.

در حقیقت این‌چنین نیست و ذخیره اطلاعات حساس در داخل Git repository باعث به خطر افتادن امنیت پروژه شما خواهد شد اما اگر این اطلاعات را داخل Git repository ذخیره نکنیم هر بار بعد از اجرای نرم‌افزار مجبور خواهیم بود که اطلاعات حساس را به‌صورت دستی اضافه کنیم که وقت‌گیر است و امکان دارد موجب ایجاد خطاهایی شود. همچنین اگر یک‌بار اشتباهاً فایل‌ها به‌صورت نادرست تنظیم شوند ممکن است باعث ایجاد آسیب‌پذیری‌هایی شود که کل پروژه را برای آپدیت‌های بعدی نیز ناامن کند.

ابزار git-secret درواقع یک bash-tool (ابزارهایی که با دستورات bash کار می‌کنند) می‌باشد که فایل‌های حساس شما را به روش GPG کدگذاری کرده و روی Git repository آپلود می‌کند. درنتیجه شما تمامی فایل‌های خود را بعد از هر بار آپدیت روی Git repository خواهید داشت.
در روش کدگذاری GPG از دو کلید استفاده می‌شود، یک کلید عمومی که برای همه‌کسانی که از فایل‌ها استفاده می‌کنند یکسان است و یک کلید خصوصی که برای هر فرد متفاوت است. این دو کلید باهم ترکیب‌شده و می‌توان از آن‌ها برای کدگذاری فایل و یا خارج کردن آن از حالت کد شده استفاده کرد.

Gitrob

ابزار Gitrob یک ابزار جستجوی فایل‌های حاوی اطلاعات حساس در داخل Git repository است که امکان جستجوی repositoryهای عمومی و خصوصی را داراست.

این ابزار ابتدا تمامی repositoryها و در مرحله بعد تمامی فایل‌های داخل این repositoryها را جستجو می‌کند سپس با استفاده از الگوهای مختلف اطلاعات حساس را در داخل فایل‌ها جستجو کرده و درنهایت اطلاعات مربوط به فایل‌های حساس را به شما ارائه می‌کند، درنتیجه از درصورتی‌که به اشتباه فایل‌های حاوی اطلاعات حساس را آپلود کرد باشید این ابزار به شما کمک می‌کند که آن‌ها را شناسایی کنید.

Website defacement

تهیه و تدوین: محمد ساروقی

Website defacement attack چیست؟

Website defacement یا به اختصار Defacement به نوعی از حملات گفته می‌شود که در آن فرد مهاجم، ظاهر یک وب‌سایت یا صفحه‌ای از وب‌سایت را تغییر می‌دهد. معمولا در اثر این تغییر ظاهر، دسترسی به محتوای اصلی سایت برای بازدیدکننده غیرممکن می‌شود. در این نوع حمله، تعدادی یا تمام صفحات یک سایت با یک صفحه یا محتوایی ثابت جایگزین می‌شود که حاوی لوگو، شعار، نام مستعار هکر (گروه هکری) و غیره می‌باشد. معمولا در این نوع حملات ضربه‌ی مهم یا غیرقابل جبرانی به سایت وارد نمی‌شود و اطلاعات پایگاه داده یا فایل‌های مربوط به آن نیز به‌صورت دست نخورده باقی می‌ماند. البته در برخی موارد ممکن است علاوه بر تغییر ظاهر، فایل‌های مهم سایت یا محتوای پایگاه‌های داده‌ای آن نیز حذف شود که بازگردانی آن‌ها می‌تواند به تعلیق چند ساعته‌ی سایت و حتی بدنامی برند یک سایت تجاری یا وب‌سایتی پرمخاطب منجر شود.

هدف فرد مهاجم در Defacement معمولا نمایش و به رخ‌کشیدن توانایی هک، انتشار یک متن یا شعار سیاسی و مذهبی، آزمودن میزان امنیت و هشدار به مالکین وب‌سایت برای بهبود ایمنی آن و غیره می‌باشد. از این‌رو معمولا سایت‌های پربازدید و مشهور نظیر سایت‌های دولتی و سایت‌های شرکت‌ها و فروشگاه‌ها در صورت عدم رعایت موارد امنیتی می‌توانند گزینه بسیار مناسبی برای این نوع حمله (به‌خصوص در مواردی که هدف فرد مهاجم انتشار یک شعار است) به‌شمار روند.

مثال‌های مختلف از Defacement سایت‌های دولتی و خصوصی در جهان:

۱. هکرهای اندونزیایی با Deface سایت دولت استرالیا در سال ۲۰۱۷ و درج پیام «Stop Spy on Indonesia» اعتراض خود را به جاسوسی دولت استرالیا از اندونزی اعلام کردند.

۲. در سال ۲۰۱۱ صفحه اصلی وب‌سایت دانشگاه‌هاروارد با عکس رئیس جمهور سوریه، بشار اسد، جایگزین شد.

۳. در سال ۲۰۱۲ حدود ۵۰۰ وب‌سایت کشور چین توسط یک گروه هکری ناشناس تغییر داده شدند.

٤. در سال ۲۰۱۳ کل وب‌سایت دانشگاه MIT، درایالات متحده پس از درگذشت هکر مشهور، آرون شوارتز، توسط حامیان این هکر تغییر داده‌شد.

و نمونه‌های مختلف دیگر در سال‌های اخیر که هر کدام از این حملات با انگیزه‌های متفاوت سیاسی، اجتماعی و غیره صورت گرفته است. اگرچه علت Defacement ذکر شد اما از علل اصلی که باید به آن اشاره کرد، آسیب‌پذیری‌های امنیتی می‌باشد که مورد سوء استفاده قرار می‌گیرند و به مهاجمان اجازه‌ی ایجاد تغییر در وب‌سایت را می‌دهند. آسیب‌پذیری‌های امنیتی رایج که در وب‌سایت‌ها و برنامه‌های وب وجود دارد شامل CSRF، XSS، SQL Injection نقص در بارگذاری فایل‌ها، مدیریت نامناسب حساب‌های کاربری و به‌روز نبودن نرم‌افزارهای وب‌سایت می‌باشد. Defacement وب‌سایت‌ها، پورتال‌های وب یا برنامه‌های وب برای صاحبان آنها می‌تواند پیامدهای اساسی به‌دنبال داشته باشد. این حملات می‌توانند باعث اختلال در عملکرد عادی وب‌سایت، خسارت به اعتبار صاحب وب‌سایت و احتمال خسارت به داده‌های ارزشمند آن‌ها شود که این امر ممکن است به هزینه‌های مالی سنگینی منجر شود. حمله به یک وب‌سایت بلافاصله عملکرد عادی آن را قطع می‌کند زیرا کارمندان سازمان و مشتریان قادر به دسترسی به ویژگی‌ها یا خدمات ارائه شده توسط وب‌سایت نیستند. علاوه بر این، اگر اقداماتِ متقابل مناسب و به‌موقع اعمال نشود، ممکن است در آینده‌ی نزدیک حملات جدید بیشتری به وب‌سایت وارد شود زیرا جزئیات امنیتی وب‌سایت فاش می‌شود و این باعث آسیب به شهرت مالکان وب‌سایت می‌گردد و همچنین ازدست رفتن داده‌های وب‌سایت از مهم‌ترین پیامدهای Defacement می‌باشد.

روش‌ها و ابزارهای مانیتورینگ و تشخیص Defacement

روش‌ها و ابزارهای متفاوتی وجود دارند که در عمل پیشنهاد و اجرا شده‌اند، این راه‌حل‌ها را می‌توان به دو دسته‌ی ۱) روش‌های مبتنی بر امضاء ۲) روش‌های مبتنی برناهنجاری تقسیم کرد.

در رویکرد مبتنی بر امضاء ابتدا مجموعه‌ای از امضاءهای حملات شناخته شده را از وب‌سایتی که ویژگی‌های مبتنی بر Deface شدن را دارا است، استخراج می‌کنیم سپس با استفاده از این ویژگی‌ها در صفحات تحت نظارت به دنبال امضاء مشخصی می‌باشیم. از ویژگی‌های اصلی این روش سرعت و تشخیص حملات شناسایی شده است اما قادر به تشخیص حملات جدید یا ناشناخته نیست. از طرف دیگر، رویکرد تشخیص مبتنی بر ناهنجاری ابتدا یک «پروفایل» از اطلاعات صفحات تحت نظارت یک وب‌سایت که در شرایط عادیِ کار است، ایجاد می‌کند.

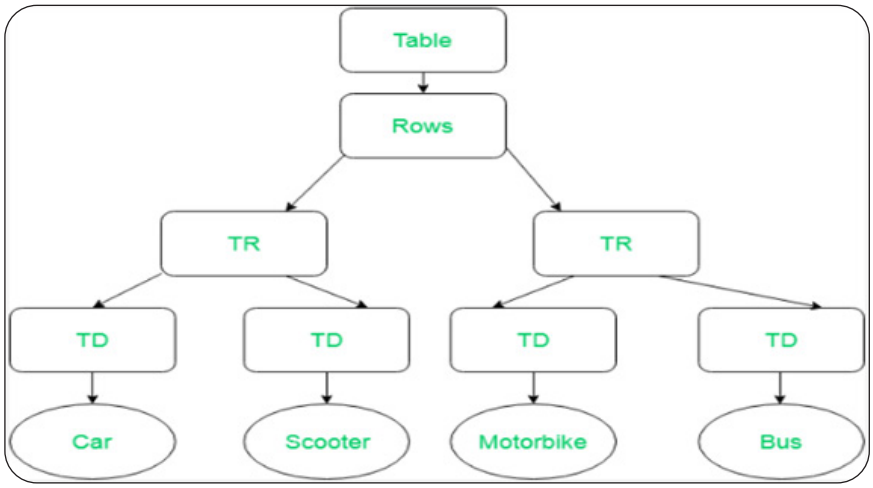
سپس صفحات برای استخراج اطلاعات مشاهده می‌شوند و اطلاعات صفحه با پروفایل مقایسه می‌شود. در صورت یافتن هرگونه علائم و تغییرات قابل توجه، هشدار حمله تشخیص داده می‌شود.

مهم‌ترین مزیت این رویکرد این است که توانایی شناسایی حملات جدید یا ناشناخته را دارد. با این حال، تصمیم‌گیری در مورد آستانه تشخیص مشکل است زیرا محتوای صفحات وب پویا به طور منظم تغییر می‌کند.

تشخیص Defacement براساس روش‌های غیرهوشمند

تشخیص تغییرات در وب‌سایت‌ها با استفاده از روش‌های قدیمی و غیرهوشمند مانند checksum برای یافتن تغییر در صفحات وب است، در مرحله اول محتوای صفحه‌ی وب با استفاده از الگوریتم‌های Hash مانند MD5، SHA1 محاسبه و ذخیره می‌شود سپس با بررسی صفحه مربوطه و Hash مجدد و بررسی، در صورتی که تغییراتی در وب‌سایت رخ داده‌باشد، به‌آسانی قابل تشخیص است. Document Object Model که به اختصار DOM نامیده می‌شود، روش دیگری است که توسط آن می‌توان تغییرات وب‌سایت را تشخیص داد، در این روش در صورتی که ساختار صفحه اصلی تغییر کرده‌باشد، می‌توانیم آن را تشخیص دهیم.

```
<Table>
<ROWS>
    <TR>
        <TD>Car</TD>
        <TD>Scooter</TD>
    </TR>
    <TR>
        <TD>Motor Bike</TD>
        <TD>Bus</TD>
    </TR>
</ROWS>
</Table>
```



ابزار diff

Diff مخفف Difference است، از این دستور برای مقایسه تفاوت‌های موجود در پرونده‌ها به صورت خطبه‌خط استفاده می‌شود.

```
diff [options] File1 File2
```

توسط این دستور می‌توانیم محتوای کد یک صفحه‌ی وب را ذخیره کنیم که با محتوای قبلی آن مقایسه شود، در صورتی که فرمت خروجی این دستور به صورت (numeric letter numeric) باشد اعداد شماره خط فایل‌ها هستند که می‌توانند یک یا چندین عدد باشند که توسط کاما (،) جدا می‌شوند.
Letter یکی از سه حرف: a(add) ,c(change) ,d(delete) است.

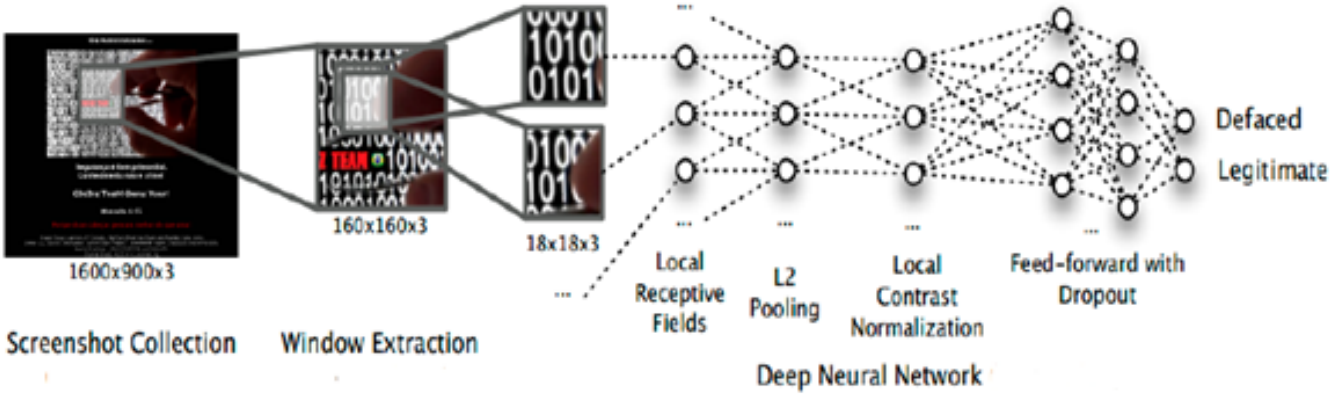
تشخیص Defacement وب‌سایت‌ها براساس روش‌های پیشرفته

یکی از روش‌های پیشنهادی که از دو مرحله تشکیل شده به این صورت می‌باشد که ابتدا با استفاده از الگوریتم N-GRAM محتوای فایل HTML وب‌سایت را به صورت 2-GRAM به بردار ویژگی‌ها تبدیل می‌کند تا بتوان با استفاده از داده‌ی ساختاریافته که استخراج نموده‌ایم و در مرحله دوم با استفاده از الگوریتم‌های طبقه‌بندی بتوانیم تغییرات در یک وب‌سایت را تشخیص دهیم.

با این‌حال از عمده کاستی‌های این روش:

۱) تعیین میزان آستانه هشدار با توجه به تغییرات مداوم محتوا روش پیشنهادی False positive زیادی تولید می‌کند.
۲) باتوجه به بردار ویژگی بزرگ و پویایی نیاز به منابع محاسباتی بالایی دارد.

روش دیگری که برای تشخیص تغییرات در وب‌سایت‌ها استفاده می‌شود مبتنی بر بینایی ماشین است، که با استفاده از شبکه عصبی عمیق با استخراج ویژگی‌هایی از تصاویری که از صفحات وب گرفته می‌شود، تصمیم به طبقه‌بندی به‌عنوان صفحه تغییریافته یا غیرتغییریافته می‌کند.



Site24x7 مانیتورینگ Defacement

این ابزار یکی از بهترین ابزارهای موجود است که می‌توان از آن استفاده نمود و به‌سرعت defacement وب‌سایت را تشخیص می‌دهد. شناسایی زودهنگام موارد امنیتی وب‌سایت، از جمله درج غیرمجاز یا اصلاح عناصر HTML صفحه وب مانند متن، اسکریپت، تصویر، پیوند و غیره از دیگر ویژگی این ابزار برای اسکن کل وب‌سایت جهت پیدا کردن پیوندهای غیر مجاز است. از مشکلات این ابزار نامناسب بودن برای وب‌سایت‌های پویا می‌باشد.

در روش‌ها و ابزارهای پایش وتشخیص مانیتورینگ Defacement وب‌سایت، می‌توان برخی از موارد را به شرح زیر عنوان کرد:

تکنیک‌های تشخیص Defacement با استفاده از Checksum، یا ابزارهای مختلف و تجزیه‌وتحلیل DOM فقط برای وب‌سایت‌های ثابت قابل استفاده هستند.

علاوه بر این، محاسبه آستانه تشخیص مناسب برای هر صفحه در روش‌های مبتنی بر ناهنجاری، بسیار مهم است. روش‌های تشخیص Defacement مبتنی بر یادگیری ماشین و داده‌کاوی دارای پتانسیل بالایی هستند زیرا مشخصات تشخیص یا آستانه را می‌توان از داده‌های آموزش مدل، آموخت.

نرم افزار آنالیز لاگ Flower fire Sawmill

نرم افزار Sawmill بدون شک یکی از بهترین نرم‌افزارهای آنالیز گزارش یا Log Analyzer است، محیط کاربر پسند و ساده از مهم‌ترین ویژگی و دلایل موفقیت این برنامه به‌شمار می‌رود. Sawmill محصول شرکت Flower fire است.

نرم‌افزار Sawmill علاوه‌بر پایگاه داده Internal خود از سه نسخه نرم‌افزار پایگاه‌داده MySQL, Oracle, SQL پشتیبانی می‌کند که امکان ذخیره‌سازی بلندمدت لاگ‌ها در داخل خود را به ما می‌دهد. برای آنالیز logها توسط این نرم‌افزار باید ابتدا لاگ‌ها را منتقل کنید، همچنین چون لایسنس از نوع Enterprise است، Unlimited Device می‌توان به برنامه اضافه کرد.

در نسخه جدید این نرم‌افزار تعداد پشتیبانی از Log Format ها از ۹۹۸ به ۱۰۲۱ مدل ارتقا پیدا کرده است و به‌آسانی می‌توان از نسخه‌های قدیمی‌تر بدون از دست‌دادن تنظیمات قبلی به نسخه جدید به‌روزرسانی کرد. با استفاده از این نرم‌افزار می‌توانیم log مربوط به وب‌سرور، فایروال و دستگاه‌هایی که جهت شناسایی Defacement به ما کمک می‌کنند را استفاده کنیم.

اقداماتی که برای جلوگیری و مقابله با Website Defacement باید انجام گیرد:

- از آخرین نسخه وب‌سرور و سرور بانک اطلاعاتی استفاده شود (سیستم عامل، وب‌سرور، پایگاه داده و سایر سرویس‌ها به‌روزرسانی شوند).
- تست نفوذ برنامه‌های کاربردی وب و وب‌سرور به‌صورت دوره‌ای یا پس از توسعه و یا ایجاد تغییرات ساختاری در برنامه کاربردی وب انجام گیرد.
- تغییر دوره‌ای رمزعبور حساب‌های کاربری
- استفاده از WAF نرم‌افزاری یا سخت افزاری برای جلوگیری و تشخیص حملات سمت وب
- پیکربندی صحیح و امن‌سازی سیستم عامل، وب‌سرور و پایگاه داده
- بازخوانی کدها و رعایت اصول کدنویسی امن
- پشتیبان‌گیری از فایل‌های برنامه کاربردی وب، پایگاه داده و فایل‌های پیکربندی برای بازیابی وب‌سایت در صورت رخدادن حمله
- برای نظارت خودکار بر تغییرات در پرونده‌ها و ساختار فهرست، از ابزارهای بررسی File Integrity استفاده کنید.

بخش دوم

کدنویسی امن php

اعتبارسنجی و فیلتر داده‌های ورودی

نویسنده: سیروان اله‌ویسی

مقدمه

اعتبارسنجی داده‌های ورودی کاربر برای وب‌سایت‌های پویا یکی از مباحث مهم و حائز اهمیت است. ورودی نامعتبر کاربر می‌تواند در پردازش اطلاعات خطایی را ایجاد کند و یا داده‌هایی را بدون داشتن مجوز دسترسی به سرور و بانک اطلاعاتی ارسال کند.
به‌عنوان مثال مهاجمان می‌توانند داده‌های ورودی را از طریق فرم HTML و یا از طریق پارامترهای API به‌عنوان مشتری REST (REST Client) ارسال کنند، این داده‌ها ممکن است نیاز به اعتبارسنجی (Validation) و یا فیلتر (Sanitizing) داشته باشند و باید قبل از پردازش فیلتر و تأیید اعتبار شوند. بنابراین اعتبار سنجی ورودی‌ها در بحث کدنویسی امن، امری بسیار ضروری است. زبان قدرتمند php به‌صورت پیش‌فرض دارای توابع داخلی متعددی است که اعتبار داده‌های ورودی کاربران را کنترل و عمل فیلتر (پاک‌سازی) را بر روی آن‌ها انجام می‌دهد.

همان‌طور که ذکر شد ارسال داده‌های نامعتبر می‌تواند باعث بروز خطرات جدی بر روی سیستم‌های نرم‌افزاری تحت وب و سرور شود. از این‌رو در فرآیند امن‌سازی یک سامانه نباید به هیچ‌کس و هیچ‌چیز اعتماد کنیم! عبارت رایجی که در PHP مشاهده خواهید کرد این است که هرگز به «ورودی کاربر» اعتماد نکنید. کاربران هنگام واردکردن اطلاعات ورودی غیرقابل‌اعتماد هستند، زیرا آن‌ها برای ما شناخته‌شده نیستند و ما هیچ کنترلی بر روی فعالیت آن‌ها نداریم.
در این مقاله جهت جلوگیری از ارسال داده‌های نامعتبر به سمت سرور، روش‌هایی برای اعتبارسنجی و اعمال فیلتر یا به‌اصطلاح فیلتر داده‌های ورودی کاربر، ارائه خواهد شد.

لیست توابع پیش‌فرض php جهت اعتبارسنجی و فیلتر داده‌های ورودی:

نام تابع	توضیحات / کاربرد
filter_has_var	بررسی می‌کند که متغیر از نوع تعیین‌شده باشد.
filter_id	شناسه (ID) فیلتر را بر اساس نام فیلتر به‌صورت عددی برمی‌گرداند.
filter_input_array	متغیرهای خارجی را در قالب آرایه دریافت و آن‌ها را اعتبارسنجی می‌کند.
filter_input	یک متغیر خارجی خاصی را بر اساس نام دریافت می‌کند و آن را فیلتر می‌کند.
filter_list	لیستی از همه فیلترهای قابل پشتیبانی را برمی‌گرداند.
filter_var_array	چندین متغیر را دریافت می‌کند و آن‌ها را اعتبارسنجی و فیلتر می‌کند.
filter_var	بر اساس یک فیلتر مشخص‌شده یک متغیر را فیلتر می‌کند.

یکی از تابع‌های پرکاربرد در زبان php تابع filter_var() می‌باشد که هم قابلیت اعتبارسنجی داده‌ها و هم قابلیت فیلتر آن‌ها را دارا می‌باشد.
در این‌جا چگونگی استفاده از این تابع جهت اعتبارسنجی و فیلتر ورودی‌ها آموزش داده خواهد شد.
نکته: در حال حاضر و تا زمان نوشتن این مقاله تعداد ۱۹ ثابت جهت استفاده در تابع filter_var() وجود دارد که لیست کامل فیلترهای قابل‌استفاده (ثوابت) در وب‌سایت رسمی php و از طریق لینک زیر قابل‌دسترس است:

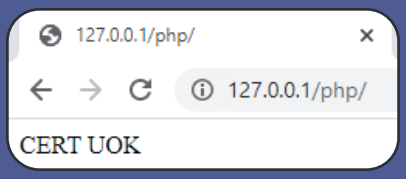
https://www.php.net/manual/en/filter.filters.validate.php

اعمال فیلتر بر روی یک‌رشته (string)

برای فیلتر یک رشته با استفاده از ثابت FILTER_SANITIZE_STRING به‌صورت زیر عمل خواهیم کرد:

```
1. <?php
2.
3. $str_name = "<span>CERT UOK<span>";
4.
5. $name = filter_var($str_name, FILTER_SANITIZE_
STRING);
6.
7. echo $name;
8.
9. ?>
```

در مثال فوق ما یک رشته را از طریق متغیر $str_name دریافت کرده‌ایم که در آن از تگ span استفاده شده‌است (خط ۳). با استفاده از تابع filter_var قبل از چاپ متغیر $str_name آن را فیلتر کرده‌ایم (خط ۵).
در تابع فوق متغیر $str_name به‌عنوان آرگومان اول و ثابت FILTER_SANITIZE_STRING به‌عنوان آرگومان (option) دوم به آن پاس داده شده‌است و در متغیر $name ذخیره شده‌است.
درنهایت با چاپ متغیر $name خروجی به‌صورت زیر است:

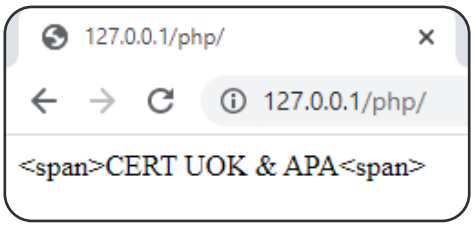


همان‌گونه که مشاهده می‌کنید کاراکترهای اضافی توسط تابع filter_var فیلتر و حذف شده‌است.
همچنین می‌توانیم از ثابت FILTER_SANITIZE_SPECIAL_CHARS برای فیلتر تگ‌های HTML استفاده نماییم. این ثابت تگ‌های HTML را در سورس فیلتر می‌نماید.

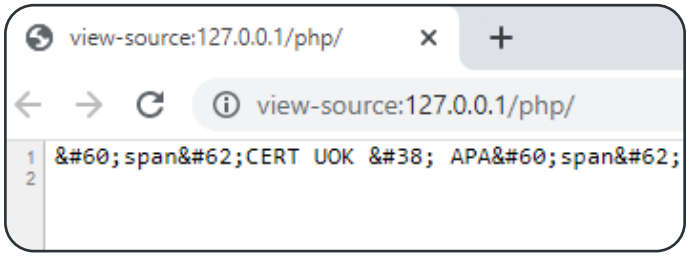
برای فیلتر یک رشته با استفاده از ثابت FILTER_SANITIZE_SPECIAL_CHARS به‌صورت زیر عمل خواهیم کرد:

```
1. <?php
2.
3. $str_name = "<span>CERT UOK & APA<span>";
4.
5. $name = filter_var($str_name, FILTER_SANITIZE_
SPECIAL_CHARS);
6.
7. echo $name;
8.
9. ?>
```

خروجی به‌صورت زیر است:



همان‌گونه که مشاهده می‌کنید در صفحه تغییراتی دیده نمی‌شود اما وقتی سورس صفحه را مشاهده کنید متوجه خواهید شد که کاراکترهای غیرمجاز فیلتر شده است:



اعتبارسنجی یک عدد صحیح (Integer)

برای اعتبارسنجی یک عدد صحیح به‌صورت زیر عمل خواهیم کرد:

```
1. <?php
2.
3. $int_age = 25;
4.
5. if (filter_var($int_age, FILTER_VALIDATE_INT)){
6.     echo "مقدار واردشده صحیح است";
7. }else{
8.     echo "مقدار واردشده صحیح نیست";
9. }
10.
11. ?>
```

در مثال فوق مقدار دریافت شده برای سن یک فرد در متغیر $int_age قرار دارد (خط ۳)، با استفاده از یک شرط مقدار واردشده توسط تابع موردنظر بررسی و اعتبارسنجی می‌شود. در اینجا ما عدد صحیح وارد کرده‌ایم پس خروجی به‌صورت مقابل است:

فیلتر و اعتبارسنجی آدرس ایمیل (Email)

برای اعمال فیلتر و اعتبارسنجی آدرس ایمیل به‌صورت زیر عمل خواهیم کرد:

```
1. <?php
2.
3. $user_email = "//apa@uok.ac.ir//()";
4.
5.    $email = filter_var($user_email, FILTER_SANITIZE_
EMAIL);
6.
7.    if(filter_var($email, FILTER_VALIDATE_EMAIL)){
8.        echo "ایمیل واردشده صحیح است"." ". $email;
9.    }else{
10.       echo "ایمیل واردشده صحیح نیست";
11.   }
12.
13. ?>
```

در مثال فوق در متغیر $user_email یک آدرس ایمیل به همراه کاراکترهای اضافی قرار داده‌ایم که در اصل یک فرمت ناصحیح از یک آدرس ایمیل می‌باشد و در ادامه جهت فیلتر کاراکترهای اضافی متغیر را به تابع موردنظر پاس داده‌ایم و ثابت FILTER_SANITIZE_EMAIL را به‌عنوان فیلتر بر روی آن قرار داده‌ایم (خط ۵).

سپس یک شرط تعیین کرده‌ایم که اگر متغیر موردنظر بعد از اعمال فیلتر و حذف کاراکترهای اضافی حاوی قالب صحیح ایمیل بود عبارت صحیح بودن آن به همراه مقدار فیلتر شده را برای ما چاپ کند (خط ۷ تا ۱۱).

مقداری که ما وارد کرده‌ایم با حذف کاراکترهای اضافی یک قالب صحیح از ایمیل می‌باشد پس خروجی مابعد از فیلتر و اعتبارسنجی بدین‌صورت خواهد بود:

همان‌گونه که مشاهده می‌کنید کاراکترهای اضافی توسط تابع filter_var فیلتر و اعتبارسنجی بر روی ایمیل انجام‌شده است.

فیلتر و اعتبارسنجی نشانی اینترنتی (URL)

برای اعمال فیلتر و اعتبارسنجی یک نشانی اینترنتی، به‌صورت زیر عمل خواهیم کرد:

```
1.  <?php
2.
3.  $user_url = "https://cert.uok.ac.ir";
4.
5.  $url = filter_var($user_url, FILTER_SANITIZE_
URL);
6.
7.  if (filter_var($url, FILTER_VALIDATE_URL)) {
8.     echo "نشانی اینترنتی صحیح است";
9.  } else {
10.    echo "نشانی اینترنتی صحیح نیست";
11. }
12.
13. ?>
```

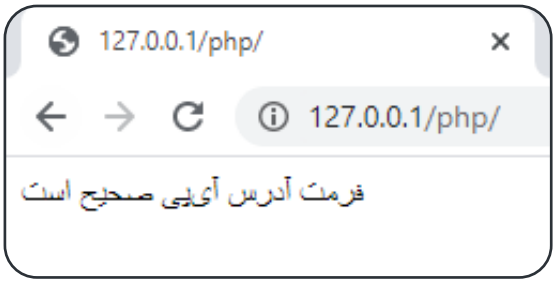
درنهایت خروجی ما بدین‌صورت است:

اعتبارسنجی آدرس IP

برای اعتبارسنجی یک آدرس IPV4 به‌صورت زیر عمل خواهیم کرد:

```
1. <?php
2.
3. $ip_address = "192.168.1.100";
4.
5. if (filter_var($ip_address, FILTER_VALIDATE_IP)){
6.     echo "فرمت آدرس آی‌پی صحیح است";
7. }else{
8.     echo "فرمت آدرس آی‌پی صحیح نیست";
9. }
10.
11. ?>
```

در مثال فوق مقدار آدرس IP دریافت شده در متغیر $ip_address قرار دارد (خط ۳)، با استفاده از یک شرط مقدار واردشده توسط تابع موردنظر بررسی و اعتبارسنجی می‌شود.
در اینجا یک آدرس IPV4 صحیح وارد کرده‌ایم پس خروجی به‌صورت زیر است:



برای اعتبارسنجی آدرس IPV6 به‌صورت زیر عمل خواهیم کرد:

```php
<?php

$ipv6_address = "2020:3ae4:85a3:0000:0000:8a2e:0370:5986";

if (filter_var($ipv6_address, FILTER_VALIDATE_IP, FILTER_FLAG_IPV6)){
    echo "فرمت آدرس آی‌پی صحیح است";
}else{
    echo "فرمت آدرس آی‌پی صحیح نیست";
}

?>
```

نوشتن تابع برای اعتبارسنجی فیلتر ورودی‌های کاربر

تا اینجا با نحوه فیلتر کردن متغیرها آشنا شدیم و نحوه استفاده از تابع filter_var را نیز یاد گرفتیم.
اما فرض کنید ما در یک وب‌سایت فرم‌های مختلفی را برای گرفتن اطلاعات از کاربران داشته باشیم و هریک از این فرم‌ها دارای فیلدهای مختلفی باشند، در اینصورت اعتبارسنجی و اعمال فیلتر به‌صورت تکی بر روی تمامی متغیرهای دریافتی کار چندان جالبی نیست و بسیار هم زمان بر و کسل‌کننده خواهد بود.
برای این کار در ادامه یک روش کاربردی به همراه یک مثال عملی جهت کاهش کدنویسی و سهولت در استفاده از تابع‌های فیلتر ورودی کاربر در قسمت‌های مهم برنامه ارائه خواهد شد.
در مثال زیر ٦ متغیر تعریف کرده‌ایم که دارای انواع مختلفی هستند و می‌خواهیم آن‌ها را به توابعی که برای فیلتر داده‌ها در ادامه می‌نویسیم پاس دهیم. جهت تست این توابع ما علاوه بر مقادیر صحیح تعدادی کاراکتر غیرمجاز نیز به آن اضافه کرده‌ایم:

```php
<?php

$name = "<h3>Sirwan</h3>";
$last_name = "<h5>Allahveisi</h5>";
$age = "25 abcd !@#";
$join_year = "1399 zxcv)(*&^%";
$website = "<span>www.sirvan.me</span>";
$email = "//info.@sirvan.me//////";
```

حال برای اعتبارسنجی و فیلتر هرکدام از انواع داده‌ها یک تابع تعریف می‌کنیم و متغیرهای هم‌نوع را به آن پاس می‌دهیم:

```php
// Filter String
function filter_string($input)
{
    $output = filter_var($input, FILTER_SANITIZE_STRING);
    if (filter_var($output, FILTER_SANITIZE_STRING)){
        return $output;
    }else{
        return "مقدار واردشده صحیح نمی‌باشد";
    }
}

// Filter & Validate Integer
function filter_int($input)
{
    $output = filter_var($input, FILTER_SANITIZE_NUMBER_INT);
    if (filter_var($output, FILTER_VALIDATE_INT)){
        return $output;
    }else{
        return "مقدار واردشده صحیح نمی‌باشد";
    }
}

// Filter & Validate URL
function filter_url($input)
{
    $output = filter_var($input, FILTER_SANITIZE_URL);
    if (filter_var($output, FILTER_VALIDATE_URL)){
        return $output;
    }else{
        return "آدرس اینترنتی واردشده صحیح نمی‌باشد";
    }
}

// Filter & Validate Email
function filter_email($input)
{
    $output = filter_var($input, FILTER_SANITIZE_EMAIL);
    if (filter_var($output, FILTER_VALIDATE_EMAIL)){
        return $output;
    }else{
        return "ایمیل واردشده صحیح نمی‌باشد";
    }
}
```

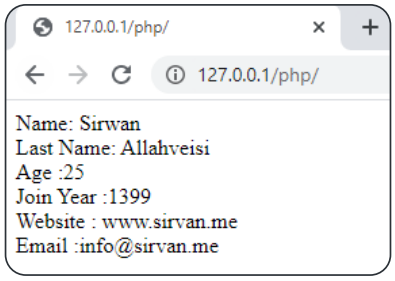
سپس برای پاس‌دادن این متغیرها به توابع موردنظر و اعتبارسنجی و اعمال فیلتر بر روی هرکدام، یک آرایه به اسم $user_data ایجاد می‌کنیم و این عملیات را درون آن انجام می‌دهیم، سپس خروجی را در عناصر آرایه ذخیره می‌کنیم:

```php
$user_data = [
    "name" => filter_string($name),
    "lname" => filter_string($last_name),
    "age" => filter_int($age),
    "join_year" => filter_int($join_year),
    "website" => filter_url($website),
    "email" => filter_email($email)
];
```

در آخر برای مشاهده تغییرات اعمال‌شده عناصر آرایه را به‌صورت مجزا چاپ می‌کنیم:

```
1. echo "Name: " . $user_data['name'] . "<br/>",
2.     "Last Name: " . $user_data['lname'] . "<br/>",
3.     "Age :" . $user_data['age'] . "<br/>",
4.     "Join Year :" . $user_data['join_year'] . "<br/>",
5.     "Website : " . $user_data['website'] . "<br/>",
6.     "Email :" . $user_data['email'] . "<br/>";
7. ?>
```

نتیجه به‌صورت مقابل خواهد بود:



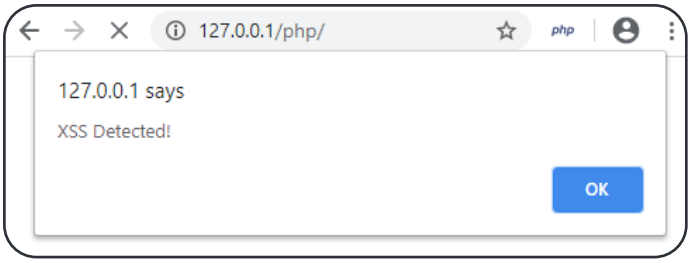
همان‌گونه که مشاهده می‌کنید کاراکترهای اضافی و غیرمجاز از مقادیر واردشده توسط توابعی که تعریف کردیم حذف‌شده‌اند و اعتبارسنجی آن‌ها نیز توسط همان توابع تأیید شده و شکل صحیح آن‌ها در خروجی چاپ‌شده است.

فیلتر عبارت‌های خاص با استفاده از تابع ()str_replace

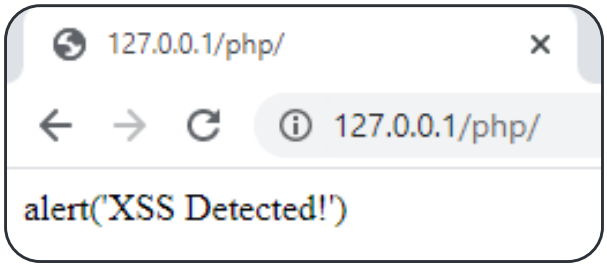
این تابع در php جهت جایگزینی یک عبارت با یک رشته به کار می‌رود. می‌توانیم از این تابع نیز جهت اعمال فیلتر بر روی ورودی‌ها استفاده نماییم.
این تابع ۳ پارامتر دریافت می‌کند، پارامتر اول عبارتی است که می‌خواهیم در رشته جستجو شود (مثلاً "script")، پارامتر دوم عبارتی است که می‌خواهیم با عبارت موردنظر جایگزین شود و پارامتر سوم متغیر یا رشته‌ای که می‌خواهیم عملیات جستجو و جایگزینی بر روی آن انجام شود. به مثال زیر توجه کنید:

```
1. <?php
2.
3. $input = "<script>alert('XSS Detected!')</script>";
4.
5. $filter_input = str_replace(«script», «/», $input);
6.
7. echo $filter_input;
8.
9. ?>
```

در مثال بالا یک متغیر به اسم $input ایجاد کرده‌ایم و یک کد جاوا اسکریپت در آن قرار داده‌ایم، در حالت عادی اگر بر روی ورودی فیلتر اعمال نشده باشد زمانی که این مقدار در صفحه چاپ شود کد جاوا اسکریپت اجرا خواهد شد که یک آسیب‌پذیری XSS محسوب می‌شود (خط ۳).
در مرحله بعد یک متغیر به اسم $filter_input ایجاد کرده‌ایم و تابع str_replace را در آن فراخوانی کرده‌ایم و عبارت "script" را به‌عنوان پارامتر اول و کاراکتر "/" را به‌عنوان پارامتر دوم و متغیر $input که رشته ما در آن قرارگرفته است را به‌عنوان پارامتر سوم به تابع پاس داده‌ایم (خط ۵).
حال تابع موردنظر بعد از اجرا در متغیر $input جستجو کرده و هر عبارت "script" را با کاراکتر "/" جایگزین می‌نماید.
در حالت عادی بدون اعمال فیلتر زمانی که متن دریافت شده در صفحه چاپ شود پنجره alert به کاربر نشان‌داده خواهد شد. نمونه خروجی:



حال بعد از اعمال فیلتر خروجی به‌صورت زیر است:



همان‌گونه که مشاهده می‌کنید تگ‌های "<script></script>" از رشته دریافتی حذف شده‌اند.

جلوگیری از ورودی‌های بیش‌ازحد بزرگ با استفاده از تابع ()strlen

تابع دیگری که می‌توانیم جهت اعمال محدودیت بر روی ورودی‌های کاربر استفاده نماییم تابع ()strlen است. این تابع در رشته یا متغیر دریافتی، تعداد کاراکترهای موجود را محاسبه می‌کند.

حال ما می‌خواهیم از این تابع جهت جلوگیری از دریافت ورودی بیش‌ازحد بزرگ که منجر به سرریز بافر می‌شود استفاده کنیم. به مثال مقابل توجه کنید:

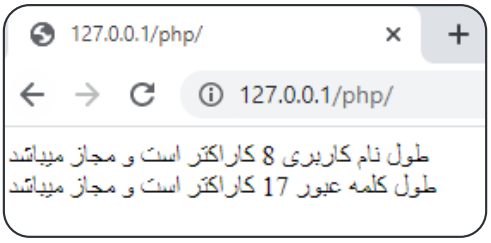
```
1. <?php
2.
3. $username = "user2020";
4. $password = "qwertyuiopasdfghj";
5.
6. if (strlen($username) > 10){
7.     echo "طول نام کاربری بیش‌ازحد مجاز است";
8. }else{
9.     echo "طول نام کاربری " . strlen($username) .
"کاراکتر است و مجاز می‌باشد" . "<br />";
```

```
10. }
11. if (strlen($password) > 20){
12.     echo "طول کلمه عبور بیش‌ازحد مجاز است";
13. }else{
14.     echo "طول کلمه عبور " . strlen($password) .
"کاراکتر است و مجاز می‌باشد";
15. }
16.
17. ?>
```

در مثال فوق یک متغیر به اسم $username و یک متغیر به اسم $password ایجاد کرده‌ایم و می‌خواهیم تعداد کاراکترهای آن را دریافت و با استفاده از دستورات شرطی، مجاز یا غیرمجاز بودن آن‌ها را تشخیص دهیم.

با استفاده از یک شرط تعیین کرده‌ایم اگر تعداد کاراکترهای دریافتی از طریق متغیر $username بیش از ۱۰ کاراکتر باشد پیغام خطا نشان داده شود و تعداد کاراکترهای واردشده نیز برای ما چاپ شود.

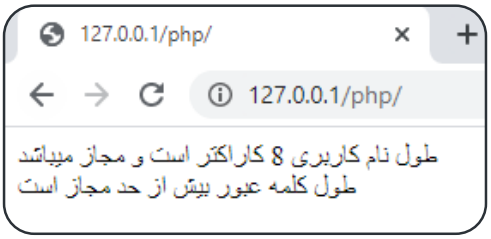
همچنین برای متغیر $password هم محدودیت کاراکتر را بر روی ۲۰ قرار داده‌ایم.
نتیجه به‌صورت مقابل است:



همان‌گونه که مشاهده می‌کنید طول ورودی‌ها مجاز بوده و تعداد کاراکترهای آن‌ها نیز توسط تابع strlen محاسبه‌شده است. حال می‌خواهیم یک رشته طولانی‌تر از حد مجاز را در متغیر $password قرار دهیم، به‌عنوان مثال رشته زیر را قرار می‌دهیم:

```
$password = "qwertyuiopasdfghjklzxcvb
nm-123456788990=!@#$%^&*()_+mmnbvcxzlkjhgfdsapoi
uytrewq";
```

بعد از بررسی نتیجه به‌صورت زیر خواهد بود:



اعتبارسنجی ورودی‌ها با استفاده از تابع ()preg_match

تابع preg_match یک الگو را دریافت می‌کند و آن را با رشته دریافت شده مقایسه می‌کند. این تابع دو پارامتر دریافت می‌کند، پارامتر اول شامل الگوهای کاراکتری است که تعیین می‌کنیم بر اساس آن مقایسه انجام شود و پارامتر دوم متغیر موردنظر که حاوی ورودی کاربر می‌باشد را به آن پاس می‌دهیم.

اعتبارسنجی رشته (string)

به مثال زیر توجه کنید:

```
1.  <?php
2.
3.  $name = "Cert Uok !@#";
4.
5.  if (!preg_match("/^[a-zA-Z ]*$/",$name))
6.  {
7.      echo "نام باید فقط شمال حروف الفبا و فاصله باشد";
8.  }else{
9.      echo "نام واردشده : " . $name . "صحیح است";
10. }
11.
12. ?>
```

در مثال فوق ما یک رشته را به همراه کاراکترهای غیرمجاز در متغیر $name قرار داده‌ایم و با استفاده از یک شرط و تابع preg_match بررسی می‌کنیم که رشته دریافت شده فقط شامل حروف الفبا باشد، در غیر این صورت پیغام خطا در صفحه چاپ شود. خروجی به‌صورت زیر است:

جمع‌بندی

در این مقاله با چند مورد از روش‌های کاربردی اعتبارسنجی و فیلتر داده‌های ورودی کاربر هنگام ورود و ثبت اطلاعات در php آشنا شدیم. روش‌های مختلفی برای تأیید اعتبار و اعمال محدودیت بر روی داده‌های ورودی در زبان برنامه‌نویسی php وجود دارد که در این مقاله سعی شد از میان روش‌های موجود بهینه‌ترین روشی که به‌صورت عملی نیز قابل‌استفاده باشد ارائه شود.

لازم به ذکر است تنها فیلتر کردن کاراکترهای مجاز و داده‌های ورودی نمی‌تواند به‌صورت کامل امنیت وب‌سایت شما را تضمین کند، ممکن است این روش‌ها نیز توسط مهاجمان دور زده شوند اما انجام این موارد اقدامی مثبت برای امنیت وب‌سایت شما محسوب می‌شود و تا حد زیادی از حملاتی که از طریق ورودی‌های وب‌سایت صورت می‌گیرد، مانند حملات Cross Site Scripting (XSS) جلوگیری خواهد کرد.

امنیت و .htaccess

تهیه و تدوین: آرش یونسی

.htaccess چیست؟

فایل .htaccess یک فایل پیکربندی وب‌سرور آپاچی است که می‌تواند دسترسی به سایت یا نحوه عملکرد آن را کنترل و همچنین تنظیمات پیش‌فرض سرور آپاچی را تغییر دهد. با استفاده از این فایل شما می‌توانید صفحه‌ای را به صفحه‌ی دیگر ریدایرکت کنید، دسترسی‌ها را محدود کنید، پوشه‌ها را رمزگذاری و مواردی از این قبیل را به‌سادگی انجام دهید. سیستم‌های مدیریت محتوا مانند وردپرس، دروپال، مجنتو (Magento) و ... برای بسیاری از تنظیمات امنیتی، بهینه‌سازی و سئو از این فایل استفاده می‌کنند.

نحوه ایجاد فایل .htaccess

فایل .htaccess را می‌توانید در هر مسیر دلخواهی قرار دهید و تنظیمات را روی آن مسیر مشخص اعمال کنید، اگر این فایل را در مسیری که قصد دارید تنظیمات را اعمال کنید نمی‌بینید، ابتدا مطمئن شوید که فایل‌های مخفی قابل‌مشاهده هستند (تیک گزینه Show Hidden Files را چک کنید) و درصورتی‌که کماکان فایل .htaccess را ندارید خودتان آن را ایجاد کنید (دقت کنید نام فایل دقیقاً برابر ".htaccess" می‌باشد، درواقع فقط یک پسوند بدون نام است).

کاربردها و نحوه استفاده صحیح از .htaccess

• جلوگیری از مشاهده مسیرها و فایل‌هایی که نباید از مرورگر برای سایرین قابل‌مشاهده باشند.

شاید گهگاهی برایتان پیش‌آمده باشد وقتی لینکی از یک وب‌سایت را بازکرده‌اید، صفحه‌ای شبیه مدیریت فایل (File Manager) باز شده که تعدادی فایل و گاهی پوشه داخل خود دارد.
این حالت عدم تنظیم باعث می‌شود که دیگران به مسیرها و فایل‌های وب‌سایت شما دسترسی داشته باشند و فایل‌های حساس شامل پسورد یا تنظیمات دیتابیس و ... را به‌راحتی دانلود کنند.

دستور زیر باعث غیرفعال شدن حالت فوق می‌شود:

```
# Disable directory browsing
Options All -Indexes
```

• محدودیت دسترسی با ipهای خاص

ممکن است بخواهید دسترسی ip یا رنج ip خاصی را قطع کنید و یا دسترسی به فایل‌هایی را فقط برای ip مشخصی تعیین کنید، برای مثال می‌توانید دسترسی فایل ورود به پنل ادمین را فقط برای یک یا چند ip آدرس مشخص تعیین کنید، درنتیجه سایر ipها نمی‌توانند به فایل ورود دسترسی داشته باشند.
دستور زیر دسترسی تمامی کاربران را (به‌جز ip آدرس‌های واردشده) به‌تمامی فایل‌ها و مسیرهای داخل پوشه‌ای که فایل .htaccess قرار دارد، می‌بندد:
این بخش دستورات متعددی دارد که با استفاده از آن‌ها می‌توانید دسترسی‌ها را بر اساس نوع متد (Get ,Post ,Put و ...) یا فایلی مشخص، برای ip آدرس‌های معین ببندید یا بازکنید.

```
# Filter access by ip address => allow access
to 302.143.54.102 and IP_ADDRESS_2

order deny,allow
deny from all
allow from 302.143.54.102
allow from IP_ADDRESS_2
```

• جلوگیری از دسترسی غیرمجاز به فایل‌های حاوی اطلاعات حساس و فایل‌های تنظیمات

عدم دسترسی افرادی غیر از شما به فایل‌های حساس یا فایل‌های حاوی تنظیمات، امری مهم در حفظ امنیت وب‌سایت شماست. با استفاده از دستور زیر می‌توانید دسترسی سایرین به این فایل‌های حساس را ببندید.
دستور زیر دسترسی همه افراد را به فایل config.php می‌بندد:

```
# Deny access to config file

<files config.php>
order allow,deny
deny from all
</files>
```

• جلوگیری از دسترسی غیرمجاز به خود .htaccess

.htaccess به‌شرط تنظیم صحیح، بخش‌هایی از امنیت وب‌سایت شما تأمین می‌کند.

درصورتی‌که دسترسی‌ها را به خود .htaccess نبندید، خود این فایل موجب ایجاد حفره‌های امنیتی خواهد شد.

دستور زیر دسترسی همه افراد به فایل‌های .htaccess را در تمامی مسیرها می‌بندد:

```
# Deny access to all .htaccess files

<files ~ "^.*\.([Hh][Tt][Aa])">
order allow,deny
deny from all
satisfy all
</files>
```

• محدود کردن نوع فایل‌های قابل‌اجرا و نمایش

برای حفظ امنیت و اطلاعات وب‌سایت می‌توانید پسوند فایل‌های قابل‌اجرا را تعیین کنید.
دستور زیر دسترسی همه افراد به فایل‌های با پسوند htaccess, htpasswd, ini, phps, fla, psd, log, sh را می‌بندد:

```
# Deny access specific file types

<FilesMatch ".(htaccess|htpasswd|ini|phps|fla|
psd|log|sh)$">
order allow,deny
deny from all
</FilesMatch>
```

• مسدود کردن حملات XSS

حملات XSS درواقع تزریق کدهای مخرب در صفحات وب است. در این روش کدهای جاوا اسکریپت به سایت تزریق می‌شوند و هدف کاربرانی هستند که به سایت مراجعه می‌کنند. درواقع هکرها با استفاده از این حملات اطلاعات کاربران یک سایت را بدون اینکه خودشان متوجه شوند، به سرقت می‌برند.
دستور زیر برخی از حملات XSS را مسدود می‌کند:

```
# Blocks some XSS attacks

<IfModule mod_rewrite.c>
RewriteCond %{QUERY_STRING} (\|3%E) [NC,OR]
RewriteCond %{QUERY_STRING} GLOBALS(=|\
[|\%[9-0A-Z]{0,2}) [OR]
RewriteCond %{QUERY_STRING} _REQUEST(=|\
[|\%[9-0A-Z]{0,2})
RewriteRule .* index.php [F,L]
</IfModule>
```

• محدود کردن سایز آپلود، Post و مدت‌زمان اجرای کد

.htaccess این امکان را به شما می‌دهد که برای حجم فایل‌های آپلودی، درخواست‌ها و زمان اجرای یک اسکریپت PHP محدودیت ایجاد کنید.
دستور زیر بیشترین حجم فایل آپلودی را روی ۲۰ مگابایت، بیشترین حجم اطلاعات post شده را روی ۱۰ مگابایت و بیشترین زمان اجرای یک اسکریپت PHP را روی ۲۰۰ میلی‌ثانیه تنظیم می‌کند:

```
# Limit max upload file size, max post size and
execution time in ms

php_value upload_max_filesize 20M
php_value post_max_size 10M
php_value max_execution_time 200
```

• تعریف ریدایرکت‌ها

یکی از پرکاربردترین موارد در .htaccess امکان تعریف ریدایرکت و فوروارد است. برای مثال یک سایت یا لینک‌های مختلفِ آن را به آدرس دیگری هدایت کنید یا مشخص کنید سایت با HTTP به HTTPS و یا با www به بدون www ریدایرکت شود. همچنین این موضوع که تمامی مسیرهای وب‌سایت شما فقط از طریق HTTPS در دسترس باشد ازلحاظ امنیتی از اهمیت بالایی برخوردار است.

دستور زیر تمامی مسیرهای وب‌سایت شما را به HTTPS ریدایرکت می‌کند (یک‌بار نوشتن این دستور داخل فایل .htaccess در مسیر root وب‌سایت کفایت می‌کند):

```
# Force HTTPS redirect

RewriteEngine On
RewriteCond %{HTTPS} off
RewriteRule ^(.*)$ https://%{HTTP_
HOST}%{REQUEST_URI} [L,R=301]
```

• جلوگیری از Hotlink

Hotlink به عملی گفته می‌شود که طی آن فایل‌های وب‌سایت شما در وب‌سایت‌های دیگر به نمایش درمی‌آید. Hotlink Protection باعث می‌گردد از کپی شدن اطلاعات شما در وب‌سایت‌های دیگر جلوگیری شود، برای مثال اگر تصویری در یک مسیر از وب‌سایت شما وجود دارد، از نمایش آن تصویر در وب‌سایت‌های دیگر جلوگیری به عمل می‌آید.

دستور زیر از استفاده‌ی فایل‌های با پسوند gif, jpg, png و css, js در سایت‌هایی غیر از سایت شما جلوگیری می‌کند، در بخش domain.com دامنه خود را وارد کنید:

```
# Deny Hotlink

RewriteEngine On
RewriteCond %{HTTP_REFERER} !^$
RewriteCond %{HTTP_REFERER} !^http://
(www./)?domain.com/.*$ [NC]
RewriteRule \.(gif|jpg|png|css|js)$ - [F]
```

• تنظیم کردن هدر برای CORS

CORS مخفف کلمه Cross-Origin Resource Sharing یک مکانیزم است که توسط آن مرورگرها می‌توانند با استفاده از هدرهای HTTP بخش‌های یک وب‌سایت را از منابعی خارج از مبدأ اصلی سایت دریافت کنند. این مکانیزم همیشه مفید نیست و ممکن است شما نخواهید منابع وب‌سایت شما در سایت‌های دیگر استفاده شوند (مانند بخش Hotlink).
توسط فایل htaccess. می‌توانید استفاده از منابع وب‌سایت خود را فقط برای دامنه خود فعال کنید و درواقع مرورگرها فقط زمانی از منابع وب‌سایت شما استفاده خواهند کرد که وب‌سایت شما را بارگذاری کرده باشند (لازم به ذکر است مطلب فوق فقط یک قانون است که همه مرورگرهای مطرح آن را رعایت می‌کنند).

دستور زیر دسترسی به فایل‌های ذخیره‌شده بر روی هاست را فقط برای دامنه شما تنظیم می‌کند:

```
# Set CORS policy Header

Header set Access-Control-Allow-Origin- https://www.domain.com
```

• محافظت در برابر حملات content-sniffing

حملات content-sniffing به‌طورمعمول شامل فریب مرورگر برای اجرای اسکریپتی است که به پسوند دیگری (js, jpg, ...) تغییر قیافه داده است.

دستور زیر محافظت در برابر content-sniffing را فراهم می‌کند:

```
# X-Content-Type nosniff

<IfModule mod_headers.c>
      Header set X-Content-Type-Options nosniff
</IfModule>
```

.htaccess

Tool Review

معرفی ابزار

ارزیابی امنیتی اپلیکیشن‌های موبایلی
با ابزار MOBSF

نویسنده: هادی گلباغی

در چند سال اخیر به دلیل رخدادهای پیش‌آمده در حوزه‌های مختلف نرم‌افزاری و سخت‌افزاری، توجهات به مقوله تست و آزمون برنامه‌ها بیش‌ازپیش شده است و در اظهارنظری جالب از پروفسور جف اوفوت آمده است که «در آینده نزدیک چه بخواهیم، چه نخواهیم، هزینه‌های مربوط به تست در تولید نرم‌افزار شامل ۵۰ درصد از کل هزینه‌های پروژه خواهد شد!!!». این موارد همراه با نفوذها، نشت اطلاعات و هک شدن سامانه‌ها و اپلیکیشن‌ها در سال‌های اخیر باعث شده است که تست و ارزیابی‌های امنیتی جدی‌تر از قبل مدنظر قرار گیرند و در حوزه‌های مختلف؛ امنیت، کشف و شناسایی آسیب‌پذیری در محصولات نرم‌افزاری و سخت‌افزاری دارای اهمیتی بیش‌ازپیش باشند.

از طرفی دیگر به دلیل گسترده‌تر شدن دستیابی به تلفن‌های هوشمند و استفاده بسیار گسترده از اپلیکیشن‌ها در امور روزمره توسط درصد بالایی از مردم، توجه به ارزیابی امنیتی در اپلیکیشن‌های تلفن همراه بسیار بیشتر از قبل شده است. اگر فرآیند تحلیل و ارزیابی آسیب‌پذیری‌ها را بررسی کنیم، به‌طورکلی در دو بخش تحلیل ایستا و پویا دسته‌بندی‌شده‌اند که به‌عنوان‌مثال در بخش ایستا مهندسی معکوس و توجه به ساختار سورس کد و در بخش پویا مانیتورینگ ترافیک شبکه و بررسی روند اجرایی برنامه از مهم‌ترین موارد بررسی خواهند بود. همواره در ارزیابی‌های امنیتی، بخشی از بررسی‌ها با استفاده از ابزارهای خودکار انجام می‌شود که این خود جنبه‌های مثبت و منفی نیز دارد.

از جنبه‌های مثبت استفاده از ابزارها، خودکار بودن ارزیابی و صرفه‌جویی در زمان را می‌توان اشاره کرد و از جنبه‌های منفی می‌توان به پوشش کم سطح ارزیابی در ابزارها و عدم اطمینان قاطع به خروجی ارزیابی‌ها را می‌توان نام برد. بااینکه در حوزه ارزیابی امنیتی اپلیکیشن‌های تلفن همراه در چند سال اخیر ابزارهای مختلفی توسعه داده‌شده‌اند اما این حوزه‌ی کاری هنوز هم جای کار دارد و اصطلاحاً حوزه‌ی جدیدی محسوب می‌شود.

در ادامه به‌عنوان نمونه به چند مورد از ابزارهایی که در ارزیابی امنیتی اپلیکیشن‌های تلفن همراه مورداستفاده هستند، اشاره خواهیم کرد.

- ابزار Drozer
- ابزار AndroBugs
- ابزار Androwarn
- ابزار Qark-master
- ابزار tracedroid (آنلاین)
- ابزار ADB
- ابزار APKTool
- ابزار JADX
- ابزار Introspy-iOS
- ابزار SUPER Analyzer
- ابزار Appcritique (آنلاین)
- ابزار Amaaas (آنلاین)
- ابزار Frida
- ابزار MOBSF

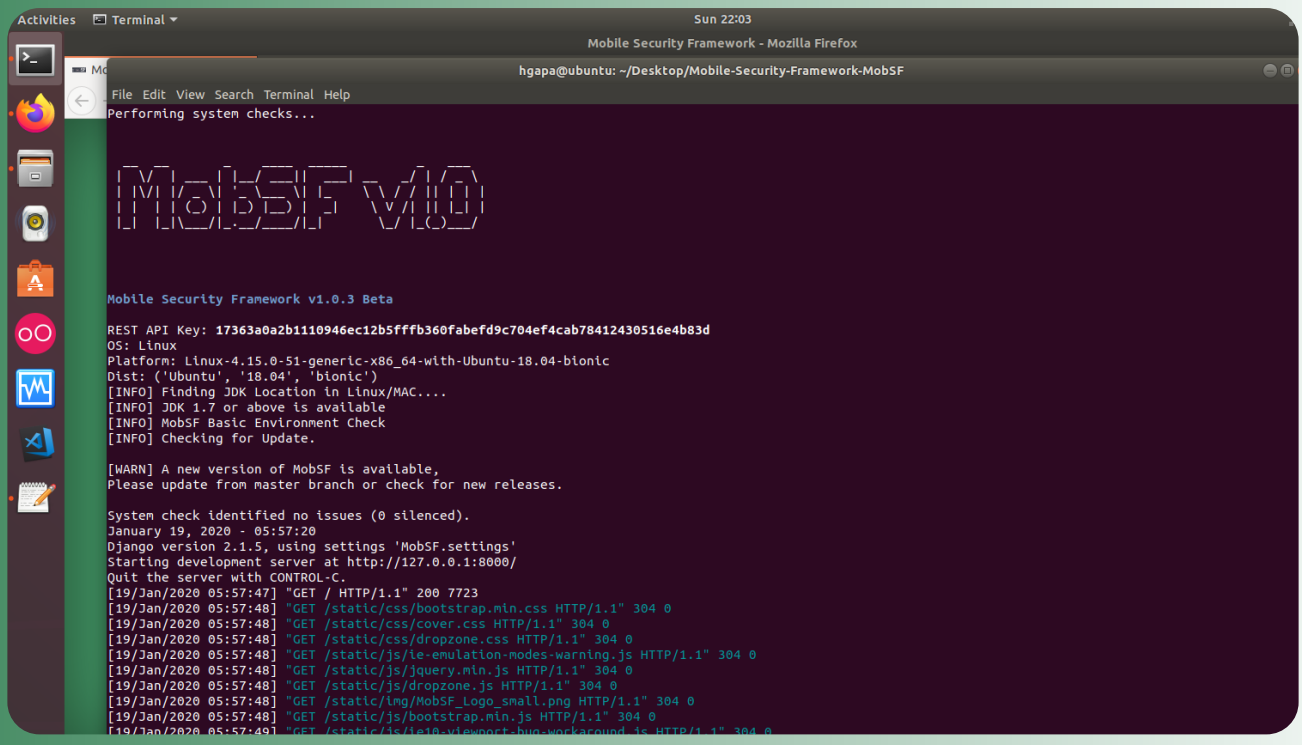
در این مقاله آموزشی ابزار قدرتمند (MOBSF) Mobile Security Framework بررسی خواهد شد.

ابزار MOBSF یک پروژه متن‌باز بوده که در گیت‌هاب برای دریافت موجود است و قادر می‌باشد تست نفوذ، تجزیه‌وتحلیل بدافزار را نیز انجام دهد. این ابزار خودکار هم تحلیل ایستا و هم تحلیل پویا را بر روی اپلیکیشن (Android-IOS) انجام می‌دهد. این ابزار دارای ویژگی‌ها و امکانات مختلفی از قبیل:

- رایگان و متن‌باز بودن ابزار
- محیطی ساده و کاربرپسند
- استفاده در محیط محلی
- تحلیل و ارزیابی بسیار سریع در سه سیستم‌عامل ویندوز، اندروید و IOS
- پشتیبانی از سورس کد به‌صورت باینری (APK-IPA-APPX) و فشرده
- پشتیبانی از Web API تست امنیتی API Fuzzer
- تحلیل بدافزار
- پشتیبانی از آسیب‌پذیری‌های OWASP Mobile Top 10
- ارائه خروجی مناسب با ارجاع و اطلاعاتی از آسیب‌پذیری شناسایی‌شده بر اساس CWE، CVSS، Severity

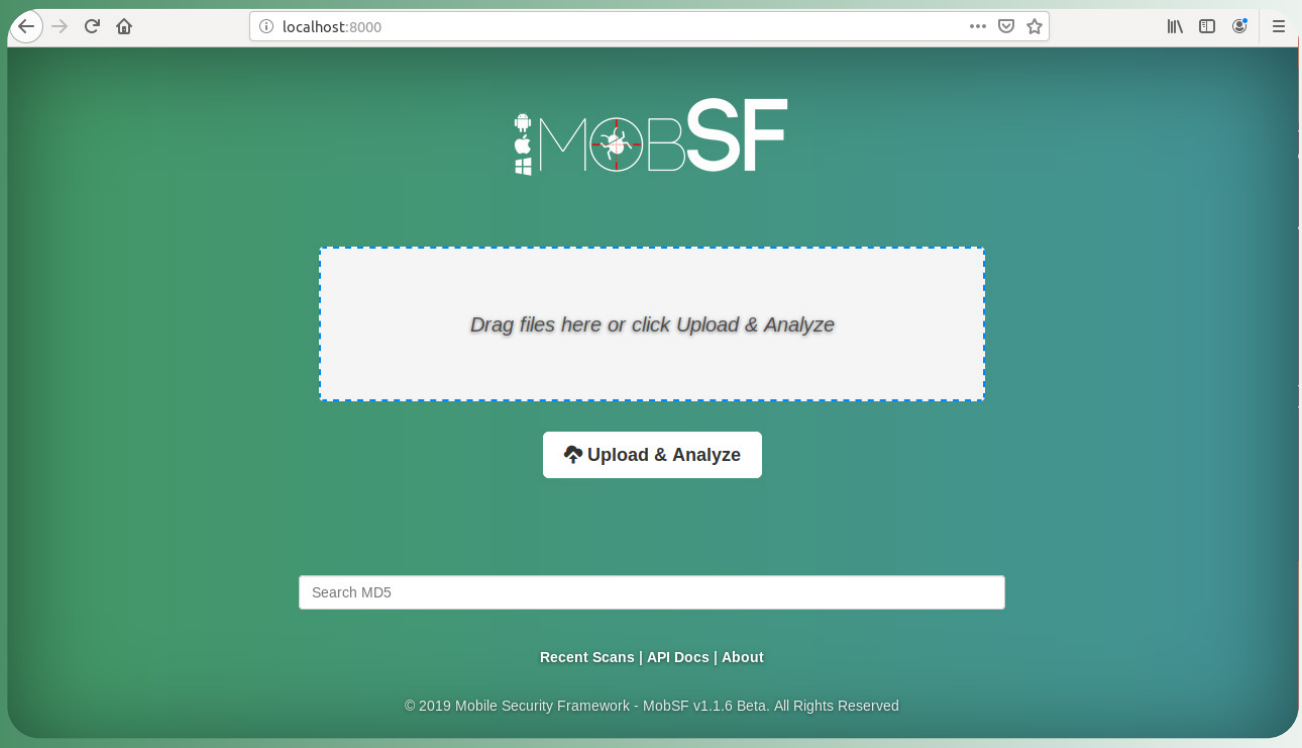
اجرای ابزار MOBSF

این ابزار را به‌راحتی با جستجوی نام آن از طریق گیت‌هاب می‌توان دانلود کرد و مورداستفاده قرارداد. در شکل ۱ نصب این ابزار در محیط سیستم‌عامل لینوکس (Ubuntu 18.04) نشان داده‌شده است که اگر به‌صورت صحیح این نصب انجام شود لوگوی ابزار در محیط ترمینال نشان داده می‌شود.



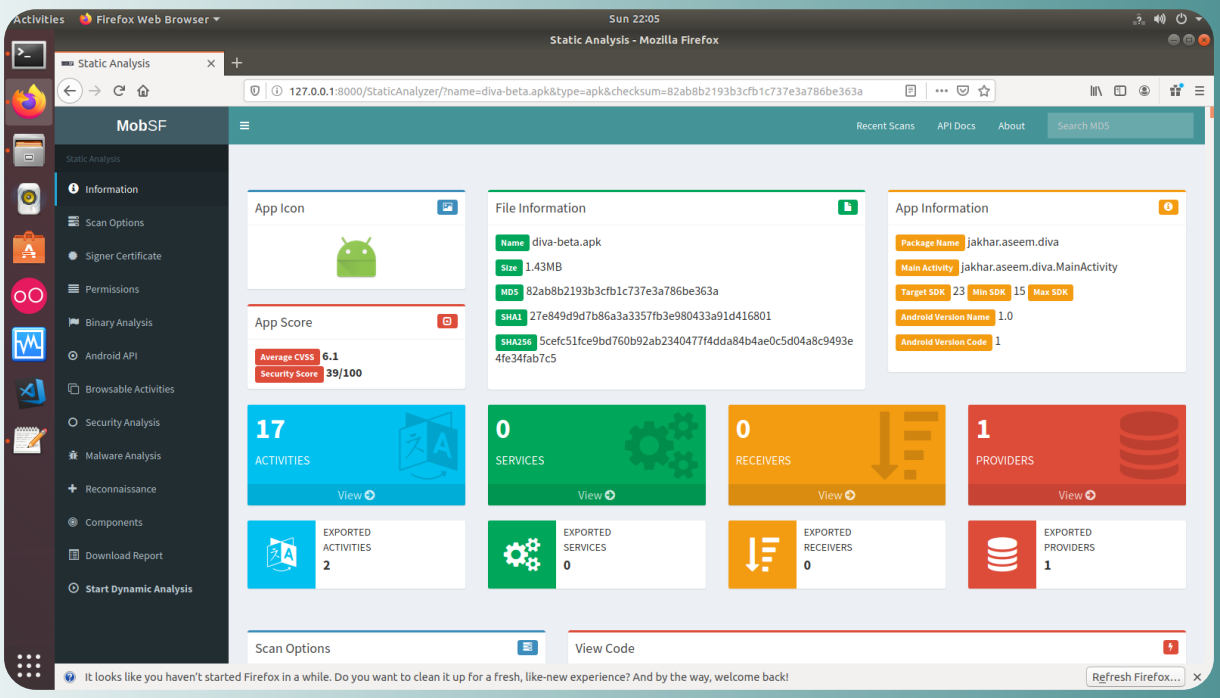
شکل ۱- نصب و اجرای MOBSF در محیط ترمینال لینوکس

با نصب به شکل صحیح و اجرای این ابزار می‌توان به‌صورت محلی و بر روی پورت ۸۰۰۰ به رابطکاربری آن دسترسی داشت که در شکل ۲ این اجرا نمایش داده‌شده است.



شکل ۲- صفحه اول ابزار MOBSF

طبق شکل ۲ مشخص است که می‌توان با drag کردن فایل اپلیکیشن و یا آپلود آن، فرآیند تحلیل و ارزیابي را آغاز کرد. همچنین در صفحه اصلي مي‌توان براي هش (MD5) اپلیکیشن‌هايي که قبلاً مورد ارزيابي قرارگرفته‌اند، جستجو نيز انجام داد. در منوي پايين نيز ارزيابي‌هاي اخير، داکيومنت مربوط به API و درباره ما مشاهده مي‌شود. در شکل ۳ ما به‌عنوان نمونه يک اپليکيشن (Diva) که براي موارد آموزشي به‌عنوان تست مورداستفاده قرار مي‌گيرد را براي ارزيابي به ابزار MOBSF به‌عنوان ورودي داده‌ايم و اولين صفحه خروجي نشان داده‌شده است.

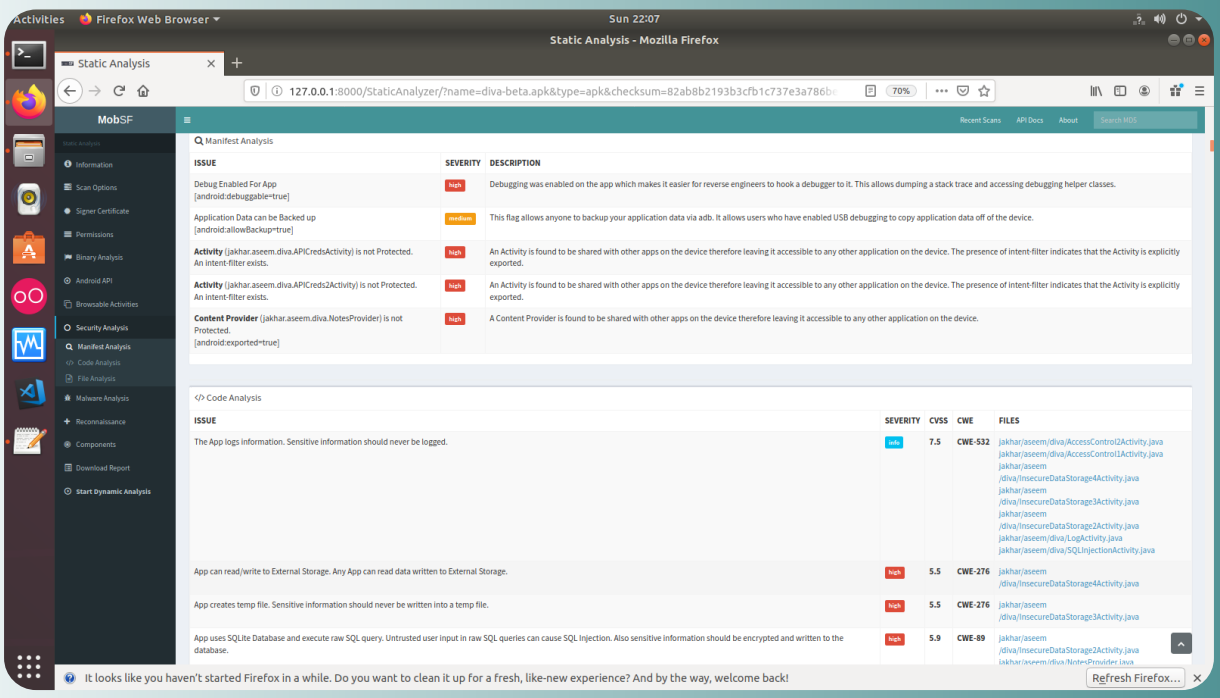


شکل ۳- صفحه اول خروجی ابزار MOBSF

همان‌طور که از شکل مشخص است اطلاعات مفیدي از اپليکيشن در خروجي ارزيابي قابل‌مشاهده بوده که در بخش‌هايي تقسيم‌بندي شده‌اند که مي‌توان به‌صورت جداگانه به هرکدام از بخش‌ها مراجعه کرده و اطلاعات خروجي ارزيابي در آن بخش را بررسي کرد.
اين بخش‌ها در دو دسته ايستا و پويا بوده که در دسته تحليل ايستا شامل:

- Information
- Scan Options
- Signer Certificate
- Permissions
- Binary Analysis
- Android API
- Browsable Activities
- Security Analysis
- Malware Analysis
- Reconnaissance
- Components
- Download Report

در هرکدام از بخش‌های مربوط به تحليل ايستا مي‌توان اطلاعات بسيار مفيدي در خصوص ارزيابي اپليکيشن موردنظر به‌دست آورد و خروجي را نيز به فرمت‌هاي مختلف از ابزار مي‌توان دريافت يا دانلود کرد. نکته جالب اين است که هرکدام از بخش‌هاي اصلي خود نيز زير بخش‌هايی داشته که به‌صورت جداگانه خروجي ارزيابي را در خصوص آن زير بخش مي‌توان ديد. به‌عنوان‌مثال در بخش مربوط به Security analysis زير بخش Manifest analysis وجود دارد که فايل مربوط به Manifest اپليکيشن را ارزيابي کرده و موضوعات حساس را با توضيحات و ميزان حساسيت بررسي مي‌کند. اين مورد در شکل ٤ نشان داده‌شده است.

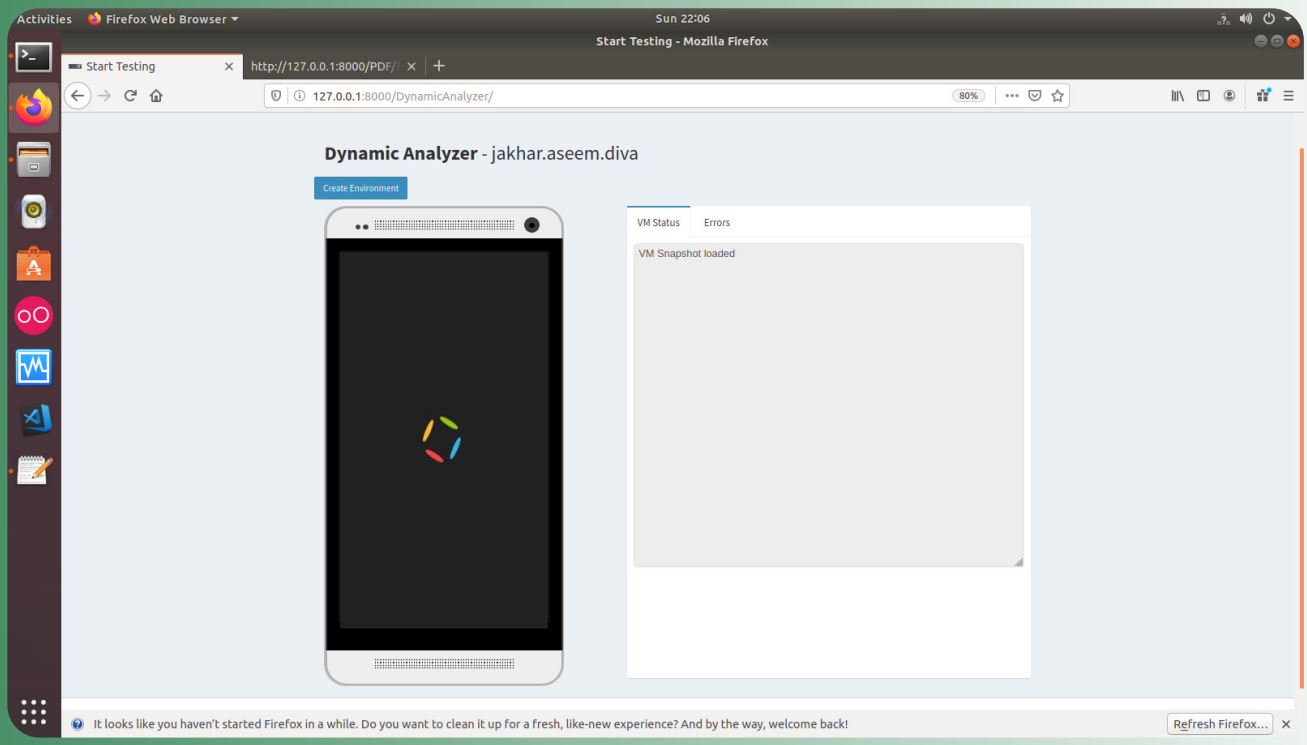


شکل ٤- زير بخش مربوط به Manifest Analysis

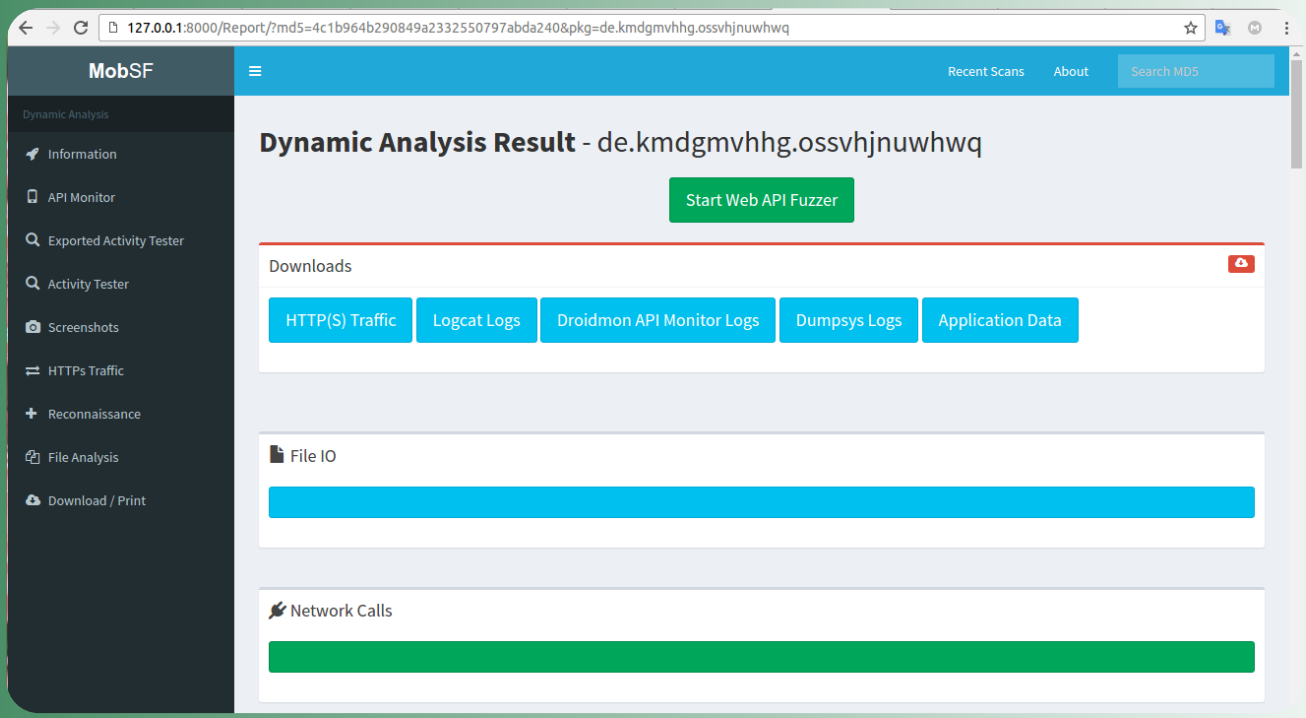
در بخش آخر نیز می‌توان ارزیابی پویا از اپلیکیشن را انجام داد و به‌عنوان‌مثال فرآیند اجرایی و تحلیل‌های این بخش را با اجرای اپلیکیشن در یک محیط شبیه‌سازی‌شده بررسی کرد. بخش‌ها در دسته تحلیل پویا شامل:

- Information
- API Monitor
- Exported Activity Tester
- Activity Tester
- Screenshots
- HTTPs Traffic
- Reconnaissance
- File Analysis
- Download/Print

در شکل‌های ۵ و ٦ ارزیابی پویا با استفاده از ابزار MOBSF و بخش‌بندی‌ها برای بخش پویا در خروجی ارزیابی اپلیکیشن نشان داده‌شده است.



شکل ۵- اجرای پویای اپلیکیشن در محیط شبیه‌سازی‌شده در ابزار MOBSF



شکل ٦- صفحه مربوط به خروجی ارزیابی پویا در ابزار MOBSF

سخن آخر

همواره براي ارزيابي امنيتي يک هدف (وب، شبکه، اپليکيشن موبايل) در ابتدا استفاده از ابزارها مي‌تواند اطلاعات مفيدي را به شخص تست‌کننده (Pentester) ارائه کند ولي بايد توجه داشت که طبق استانداردهاي موجود براي ارزيابي امنيتي، حدوداً ۲۰ درصد از کارِ ارزيابي را مي‌توان با ابزارها انجام داد و ۸۰ درصد از فرآيند ارزيابی امنيتی صحيح می‌بايستی به‌صورت دستی (Manual) صورت گيرد.

در اين مقاله توضيحاتی در خصوص ابزار قدرتمند MOBSF ارائه شد که برای بخش ارزيابی خودکار اپليکيشن موبايلی با استفاده ابزار، نقطه شروع خوبی است و چون هم ارزيابی ايستا و هم پويا را در خود جای‌داده است می‌تواند بررسی و ارزيابی اوليه مفيدی از اپليکيشن موردنظر برای شخص تست‌کننده فراهم آورد. برای دسترسی به اطلاعات بيشتر در خصوص استفاده از اين ابزار، می‌توانيد به لينک‌های بخش منابع اين مقاله که گيت‌هاب اين ابزار نيز در آن وجود دارد، مراجعه کنيد.

آخرين نکته اينکه سايت معتبر Udemy دوره‌ای را باعنوان Automated Mobile Application Security Assessment with MobSF در سايت خود دارد که در خصوص ارزيابی امنيتی خودکار اپليکيشن‌های موبايلی می‌باشد، لينک آن در منابع موجود است. طی کردن اين دوره برای تسلط بر اين ابزار مناسب به نظر می‌رسد.

منابع

- https://github.com/MobSF/Mobile-Security-Framework-MobSF
- https://nullcon.net/website/archives/ppt/goa-16/Automated-Mobile-Application-Security-Assessment-with-MobSF-by-Ajin-Abraham.pdf
- https://hydrasky.com/mobile-security/how-to-use-mobile-security-frameworkmobsf/
- https://www.udemy.com/course/automated-mobile-application-security-assessment-with-mobsf/

Cheat Sheet

دفترچه تقلب

Wireshark Cheat Sheet

تهیه و تدوین: محمد حبیبی

حالت‌های ذخیره‌سازی وایرشارک

Promiscuous mode

یک رابط انتخاب می‌شود و ابزار شروع به ذخیره Packetهای مربوط به شبکه آن رابط می‌کند.

Monitor mode

یک رابط برای شبکه بی‌سیم ایجاد می‌شود و تمامی ترافیک دریافتی از طریق آن را ذخیره می‌کند (فقط برای سیستم‌عامل‌های لینوکس و یونیکس).

انواع فیلترها

Capture filter

این فیلترها بر Packetهایی که ذخیره می‌شوند اعمال می‌شوند.

Display Filter

این فیلترها صرفاً برای سفارشی‌سازی خروجی ترافیکِ ذخیره‌شده استفاده می‌شوند.

لیست برخی از پروتکل‌های کاربردی در وایرشارک

Internet (TCP/IP) protocol family

ARP,RARP,IP,IPv6,ICMP,ICMPv6,IGMP,RIP,RIPng,DSR,AH,ESP,UDP,UDP-Lite,TCP,RTP,NetBIOS,HTTP,SMTP,MIME,DHCP,DNS,FTP,IMAP,NTP,SSH,SNMP

LTE Protocol Family

S1AP,NAS,Diameter, GTPv2,GTPv1.GTP-U

Voice over IP protocol Family

SIP,SDP,RTSP,Sigcomp,H323,H225,H235,H248/MEGACO,MGCP,SIGTRAN,SKINNY,IAX2,RTP,RTCP,T38,SRTP,MIKEY,RADIUS,DIAMETER

Network Filesystem Family

SMB,SMB2,ATSVC,DSSETUP,INITSHUTDOWN,LSA,NETLOGON,SAMR,BrowserProtocol,Mailslot,WINREG

فیلترهای نمایشی در وایرشارک

عملگر	توضیح	مثال
== یا eq	مساوی	ip.dest == 192.168.1.1
=! یا ne	نامساوی	ip.dest != 192.168.1.1
> یا gt	بزرگ‌تر	frame.len > 10
< یا lt	کوچک‌تر	frame.len <10
=> یا ge	بزرگ‌تر یا مساوی	frame.len >= 10
<= یا le	کوچک‌تر یا مساوی	frame.len<=10

عملگرهای منطقی

عملگر	عنوان	توضیحات
&& یا and	عملگر منطقی And	تمامی عبارات باید صحیح باشند.
|| یا or	عملگر منطقی Or	همه یا یکی از عبارات باید صحیح باشد.
^^ یا xor	عملگر منطقی Xor	فقط یکی از دو عبارت باید صحیح باشد.
! یا not	عملگر Not	دو عبارت باید نابرابر باشند.
[n] [...]	عملگر زیررشته	از زیر رشته‌ها برای فیلترکردن یک کلمه یا متن خاص استفاده می‌شود.

کلیدهای میانبر در وایرشارک

کلید	عملکرد
Tab or Shift+Tab	حرکت بین اجزای صفحه
↓	حرکت به Packet یا آیتم بعدی
↑	حرکت به Packet یا آیتم قبلی
Ctrl+ ↓ or F۸	حرکت به Packet بعدی، حتی اگرفوکوس بر روی لیست Packetها نباشد.
Ctrl+ ↑ or F۷	حرکت به Packet قبلی، حتی اگرفوکوس بر روی لیست Packetها نباشد.
Ctrl+.	حرکت به Packet بعدی گفتگو (TCP, UDP or IP)
Ctrl+,	حرکت به Packet قبلی گفتگو (TCP, UDP or IP)
Backspace	در هنگام نمایش جزئیات، Packet به Node والد برمی‌گردد.

فیلترهای ذخیره‌سازی در وایرشارک

ساختار فیلتر	توضیحات
host 172.18.5.4	ذخیره‌سازی فقط ترافیک دریافتی یا ارسالی به IP ذکر شده
net 24/192.168.0.0	ذخیره‌سازی ترافیک دریافتی یا ارسالی به یک رنج IP مشخص
src net 24/192.168.0.0	ذخیره‌سازی ترافیک دریافتی از یک رنج IP مشخص
dst net 24/192.168.0.0	ذخیره‌سازی ترافیک ارسالی به یک رنج IP مشخص
port 53	ذخیره‌سازی ترافیک پروتکل DNS
host www.exam.com and not (port 80 or port 25)	ذخیره‌سازی ترافیک یک هاست با حذف Packetهای ارسالی و دریافتی از دو پورت 80 و 25 (SMTP and HTTP)
port not 53 and not arp	ذخیره‌سازی تمام ترافیک جز Packetهای ارسالی یا دریافتی توسط پروتکل‌های DNS و ARP

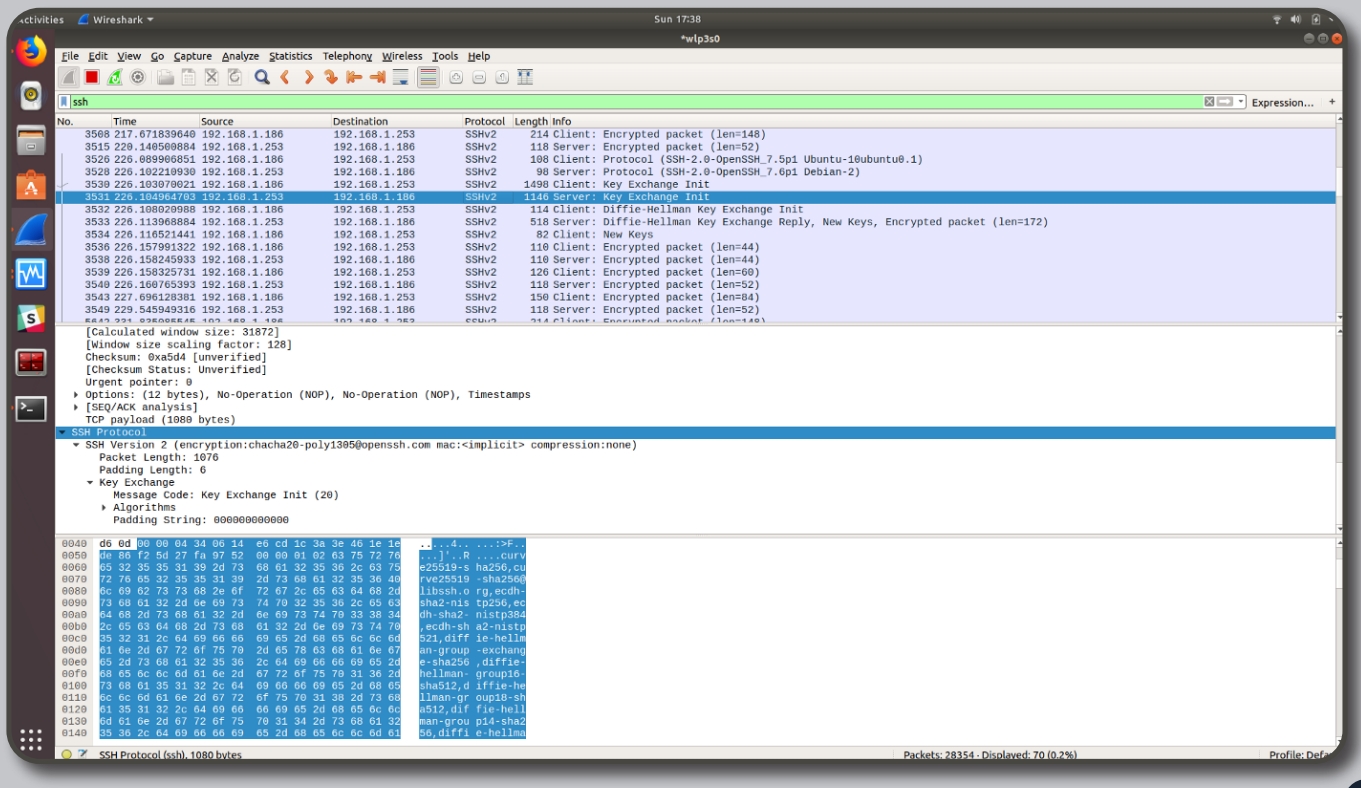
ساختار فیلتر	توضیحات
tcp portrange 1549-1501	ذخیره سازی ترافیک یک رنج خاص از پورت‌ها
ether proto 0x888e	ذخیره‌سازی فقط ترافیک پروتکل Ethernet از نوع EAPOL
ip	ذخیره‌سازی ترافیک ارسالی از طریق پروتکل IPv4
not broadcast and not multicast	ذخیره‌سازی ترافیک unicast و حذف Packetهایی که به صورت broadcast و multicast ارسال می‌شوند.
dst host ff02::1	ذخیره ترافیک ارسالی یا دریافتی از تمامی گره‌های IPv6 موجود در شبکه
dst port 135 and tcp port 135 and ip[2:2]==48	ذخیره‌سازی ترافیکی که ممکن است توسط Blaster worm ارسال شده باشد.
icmp[icmptype]==icmp-echo and ip[2:2]==92 and icmp[8:4]==0xAAAAAAAA	ذخیره‌سازی ترافیکی که ممکن است توسط Welchia worm ارسال یا دریافت شده باشد.

تعدادی از فرامین فیلترینگ کاربردی در وایرشارک

ساختار فیلتر	توضیحات
ip.addr == 10.10.50.1	فیلتر براساس یک آدرس IP
ip.dest == 10.10.50.1	فیلتر براساس آدرس IP مقصد Packet
ip.src == 10.10.50.1	فیلتر براساس آدرس IP مبدا Packet
ip.addr >= 10.10.50.1 and ip.addr <= 10.10.50.100	فیلتر براساس یک رنج IP
ip.addr == 10.10.50.1 and ip.addr == 10.10.50.100	فیلتر بر اساس چند آدرس IP
!(ip.addr == 10.10.50.1)	حذف یک آدرس IP خاص
ip.addr == 24/10.10.50.1	فیلتر بر اساس یک Subnet خاص

ساختار فیلتر	توضیحات
tcp.dstport == 23	فیلتر بر اساس یک پورت خاص در مقصد Packet
ip.addr == 10.10.50.1 and Tcp.port == 25	فیلتر براساس یک آدرس IP و پورت خاص
http.host == “host name”	فیلتر براساس یک URL
frame.time >= “June ,02 18:04:00 2019”	فیلتر براساس زمان و تاریخ دریافت Frame
tcp.flags.syn == 1	فیلتر کردن بر اساس فلگ SYN
eth.dst == ff:ff:ff:ff:ff:ff	فیلتر براساس Packetهای Brodcast
eth.addr == 00:70:f4:23:18:c4	فیلتر براساس یک آدرس فیزیکی(MAC) خاص

WIRESHARK



Course Description

معرفی دوره

معرفی دوره

Certified Information Systems Security Professional (CISSP)

تهیه و تدوین: نازیلا خسروی

با توجه به نقش فناوری اطلاعات در دنیای امروز و استفاده از انواع تکنولوژی‌های مرتبط، حفاظت از اطلاعات و امنیت سیستم‌های اطلاعاتی یکی از اصلی‌ترین ارکان پایداری هر ارگان و سازمانی است. با تدوین سیاست‌های امنیتی درست و دقیق می‌توان دستیابی به حفاظت و امنیت را تحقق بخشید همچنین نیاز به تربیت متخصصین کارآمد در این زمینه برای هر سازمان و مجموعه‌ی اطلاعاتی حیاتی و ضروری است.

آزمون و مدرک بین المللی Certified Information Systems Security Professional- CISSP- متعلق به شرکت $(ISC)^2$ -The International Information Systems Security Certification Consortium- به‌عنوان مدرکی بسیار کاربردی و مفید در شاخه امنیت اطلاعات، استراتژی‌های تامین امنیت شبکه را ارائه می‌نماید و هدف آن ایجاد یک سطح مهارت حرفه‌ای و عملی در زمینه امنیت اطلاعات می‌باشد. CISSP اولین مدرک در زمینه امنیت اطلاعات بوده که اعتبار آن در سازمان استانداردهای ملی آمریکا (ANSI) و سازمان استانداردهای جهانی (ISO) و همچنین در زمینه دانش فنی و مدیریتی تضمین اطلاعات، توسط وزارت دفاع آمریکا (DoD) تصویب و تایید شده است.

این مدرک به دلیل عدم وابستگی آن به محصولی خاص، به‌عنوان یک عنصر کلیدي در ارزشیابي داوطلبان کار در مؤسسات بزرگ و سیستم‌هاي Enterprise شناخته مي‌شود. اشخاص دارای مهارت و مدرک CISSP توانایی لازم در طراحی و پیاده‌سازي سیاست‌های کلان امنیتي را خواهند داشت. این افراد، دارای درک کامل و مستقلی از مسائل مربوط به مهندسی اجتماعی بوده و قادر به ایجاد امنیت اطلاعات در یک سازمان با ارائه خط مشی ویژه با سیاست‌های خاص امنیتی آن سازمان می‌باشند.

می‌توان دوره‌ی آموزشی Secutiry+ را پیش‌نیاز دوره‌ی CISSP دانست.

سرفصل مطالب CISSP از امنیت اطلاعات (Information Security) شامل عناوین زیر می‌باشد:

Security and Risk Management
Asset Security
Security Engineering
Communications and Network Security
Identity and Access Management
Security Assessment and Testing
Security Operations
Software Development Security

CISSP®

Certified Information
Systems Security Professional

می‌توان جزئیات مطالبی که تحت پوشش دوره فوق خواهند بود را به شرح ذیل نام برد:

- معرفی کلی امنیت و اقدامات مدیریتی امنیت اطلاعات
- مسئولیت‌های مدیریت، سیاست‌های امنیتی، طبقه‌بندی اطلاعات، مدیریت ریسک و آگاه‌سازی امنیتی
- سیستم‌ها و متدولوژی‌های کنترل دسترسی
- طراحی و معماری امنیت
- رمزنگاری
- امنیت ارتباطات، شبکه و اینترنت
- طرح تداوم کسب‌وکار (BCP) و طرح بازیابی حوادث (DRP)
- بررسی مسائل حقوقی و قانونی و جرایم رایانه‌ای
- امنیت توسعه سیستم‌ها و برنامه‌های کاربردی
- حاکمیت امنیت اطلاعات و مدیریت مخاطرات
- امنیت فیزیکی و محیطی

مخاطبین و متقاضیان دوره‌ی آموزشی و مدرک CISSP:

۱. مدیران امنیت اطلاعات سازمان‌ها و ارگان‌ها
۲. مدیران حراست سازمان‌ها و ارگان‌ها
۳. متخصصین سیستم‌های اطلاعاتی و امنیتی
٤. مدیران شبکه
۵. مشاورین سیستم‌های امنیت اطلاعات و کلیه علاقه‌مندان به مدیریت امنیت اطلاعات

CISSP Vs CISA

علیرغم اینکه گواهینامه‌های CISSP و CISA هر دو مبتنی بر سیستم‌های اطلاعاتی هستند، اما تفاوت‌های اساسی با یکدیگر دارند. CISSP بر مسائل امنیتی تمرکز دارد. وجه تشابه هر دوی این گواهی‌ها، نیازمندی به حداقل ۵ سال تجربه و سابقه کار است و این بدان معنا است که نه اخذ گواهی آسان است و نه باید آن را دست کم گرفت. داشتن هر کدام از این دو گواهی می‌تواند به داشتن یک کار با درآمد بالا منجر شود، در واقع افراد دارای گواهینامه CISA یا CISSP معمولاً پیشنهادات شغلی متعددی را دریافت می‌کنند.

CISA®

Certified Information
Systems Auditor®

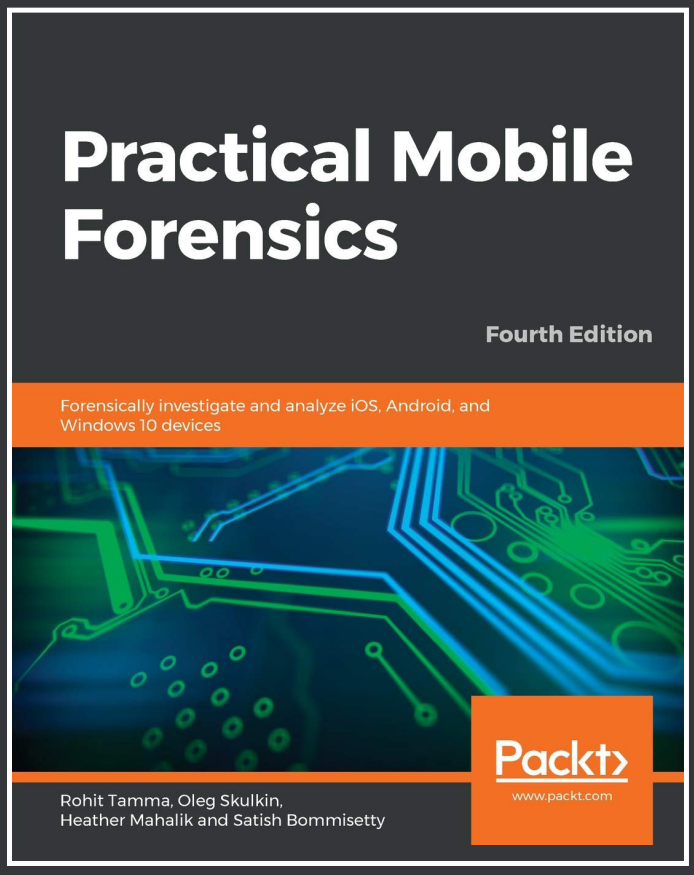
An ISACA® Certification

Book Suggestion

معرفی کتاب

معرفی کتاب

تهیه و تدوین: کسرا ریسمانچی



دانش جرم‌شناسي دستگاه‌هاي تلفن همراه به معناى به‌دست آوردن اطلاعات تلفن‌همراه با جزئيات و بررسي تحت شرايط قضايى است. در ويرايش چهارم اين كتاب مفاهيم جرم‌شناسي موبايل در عمل و اهميت آن در دنياى امروزى به‌شكل موشكافانه‌اى بررسى شده‌است.
تمركز اصلى اين كتاب در آموزش تكنيك‌هاى جرم‌شناسي موبايل به منظور بررسى تلفن‌هاى همراه با سيستم عامل‌هاى متفاوت است كه با خواندن آن، براى سيستم‌عامل‌هاى iOS11 تا iOS13، نسخه‌هاي ۸ تا ۱۰ سيستم‌عامل اندرويد و همچنين در ويندوز ۱۰ تكنيك‌هاى جرم‌شناسي را به صورت عملي خواهيد آموخت. بعد از موارد فوق، به‌روزترين ابزارهاى متن‌باز به منظور بررسى جرم‌شناسي موبايل مورد بررسى قرار مى‌گيرند كه در نهايت شما را در بازيابى و تحليل اطلاعات، به شكلى مؤثر توانمند خواهد ساخت. موضوعات كتاب شما را از بخش‌هايى چون بررسى دستگاه و به‌دست آوردن اطلاعات از فضاى ابرى به تهيه گزارش نهايى از اسناد بررسى‌هاى به‌وجود آمده، راهنمايى خواهد كرد.
در ادامه كتاب، درک درستى از مهندسى معكوس در اپليكيشن‌ها و راه‌هاى موجود براى شناسايى بدافزار به‌دست مى‌آيد. در بخش پايانى كتاب، تجزيه و تحليل اپليكيشن‌هايى چون فيس‌بوک و واتس‌اپ مورد بررسى قرار خواهد گرفت.

مشخصات کتاب

نام کتاب:	Practical Mobile Forensics
نویسندگان:	Rohit Tamma, Oleg Skulkin, Heather Mahalik, Satish Bommisetty
تعداد صفحات:	400
ناشر:	Packt Publishing (Apr 9, 2020)
زبان:	English

لینک کتاب

این کتاب مناسب چه کسانی است؟

۱- متخصصين حوزه قضايي و علاقه‌مند به حوزه جرم‌شناسي در تلفن‌هاي همراه
۲- متخصصين امنيت به خصوص حوزه امنيت تلفن‌هاي همراه و مهندسي معكوس اپليكيشن‌ها
۳- محققانى در جست‌وجوى يافتن دانش عميق در زمينه قطعات داخلى تلفن‌هاي همراه

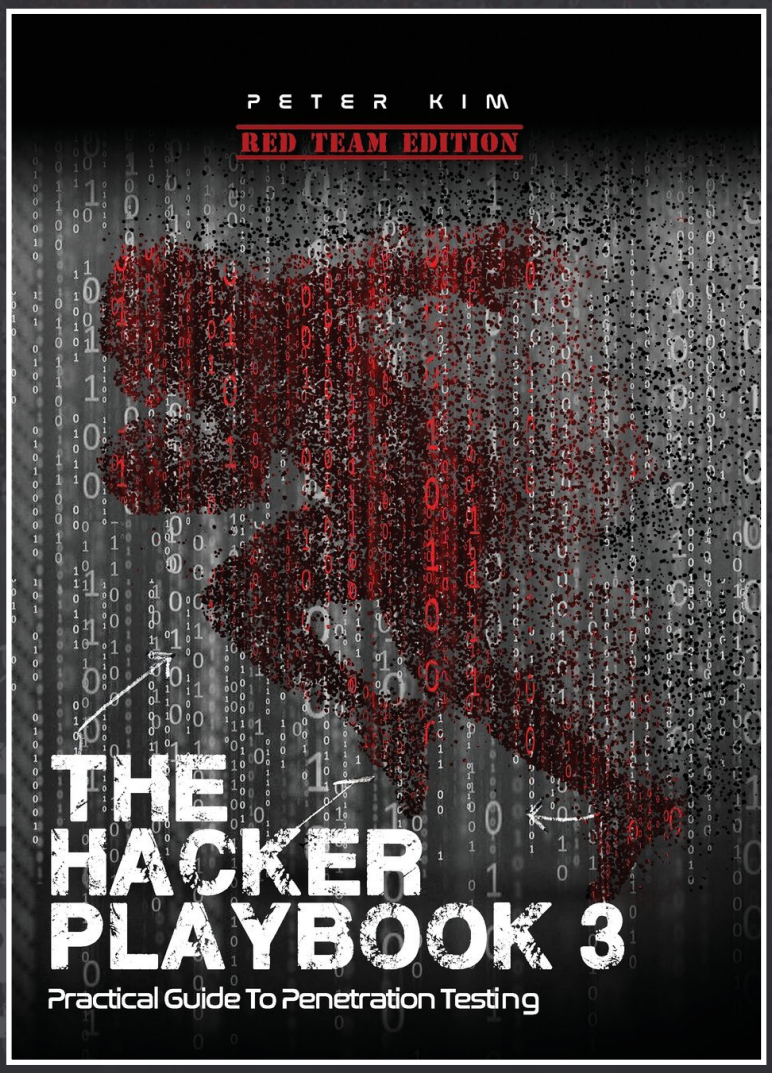
معرفی کتاب

تهیه و تدوین: محمد حبیبی

مشخصات کتاب

نام کتاب:	The Hacker Playbook 3: Practical Guide to Penetration Testing
نویسنده:	Peter Kim
تعداد صفحات:	291
ناشر:	Secure Planet LLC (1st May 2018)
زبان:	English

نسخه جدید این کتاب، The Hacker Playbook 3 (THP3) که توسط Peter Kim نوشته شده است در سال 2018 به انتشار رسید. این کتاب یک راهنمای عملی برای تست نفوذ است که با ترکیبی از استراتژی‌ها، حملات، اکسپلویت‌ها، نکات و ترفندهایی در این مسیر می‌تواند به شما کمک کند.
هدف اصلی این کتاب پاسخ دادن به این سوال است که چرا با وجود محصولات امنیتی متفاوت، بازبینی کدها از نظر امنیتی، مکانیزم‌های دفاعی و همچنین آزمایشات تست نفوذ، ما کماکان شاهد وجود رخنه‌های امنیتی در سازمان‌های بزرگ و دولت‌ها هستیم. سوال مهمی که باید از خود بپرسیم این است که آیا این مکانیزم‌ها، ابزار و موارد امنیتی که استفاده کرده‌ایم به درستی کار می‌کنند یا خیر؟! پاسخ این سوال تمامی چیزی است که در این کتاب آورده شده است.



لینک کتاب

بخشی از سرفصل‌ها و عناوین مطرح شده در کتاب به شرح زیر می‌باشند:

Red Team Recon

Monitoring an Environment
Regular Nmap Diffing
Web Screenshots
Cloud Scanning
Network/Service Search Engines
Manually Parsing SSL Certificates
Subdomain Discovery
Emails

Web Application Exploitation

Cross Site Scripting(XSS)
Blind XSS
DOM Based XSS
NoSQL Injections
Deserialization Attacks
Template Injections
Server Side Request Forgery (SSRF)
XML eXternal Entities (XXE)
Advanced XXE - Out Of Band (XXE-OOB)

Compromising The Network

Finding Credentials from Outside the Network
User Enumeration Without Credentials
Scanning the Network with CrackMapExec (CME)
Privilege Escalation
Pulling Clear Text Credentials from Memory
Getting Passwords from the Windows Credential Store and Browsers
Getting Local Creds and Information from OSX
Living Off of the Land in a Windows Domain Environment
Querying Active Directory
Lateral Movement with DCOM
Pass-the-Hash
Gaining Credentials from Service Accounts
Dumping the Domain Controller Hashes
Attacking the CSK Secure Network

Social Engineering

Building Your Social Engineering (SE) Campaigns
Doppelganger Domains
How to Clone Authentication Pages
Credentials with 2FA
Phishing
Microsoft Word/Excel Macro Files
Non-Macro Office Files - DDE
Hidden Encrypted Payloads
Exploiting Internal Jenkins with Social Engineering

Physical Attacks

Card Reader Cloners
Physical Tools to Bypass Access Points
LAN Turtle (lanturtle.com)
Packet Squirrel
Bash Bunny
Breaking into Cyber Space Kittens
QuickCreds
BunnyTap
WiFi

Cracking, Exploits, And Tricks

Automating Metasploit with RC scripts
Automating Empire
Automating Cobalt Strike
Password Cracking
Gotta Crack Em All - Quickly Cracking as Many as You Can
Disabling PS Logging
Windows Download File from Internet Command Line
Getting System from Local Admin
Retrieving NTLM Hashes without Touching LSASS
Building Training Labs and Monitor with Defensive Tools

Evading Av And Network Detection

Writing Code for Red Team Campaigns
The Basics Building a Keylogger
Compiling from Source
Obfuscation
THP Custom Droppers
Shellcode vs DLLs
Recompiling Metasploit/Meterpreter to Bypass AV and Network Detection
How to Build Metasploit/ Meterpreter on Windows
Creating a Modified Stage 0 Payload: SharpShooter
Application Whitelisting Bypass
Code Caves
PowerShell Obfuscation
PowerShell Without PowerShell: HideMyPS

Research Papers

مقاله‌های تحقیقاتی

امنیت در سرویس‌های ابری

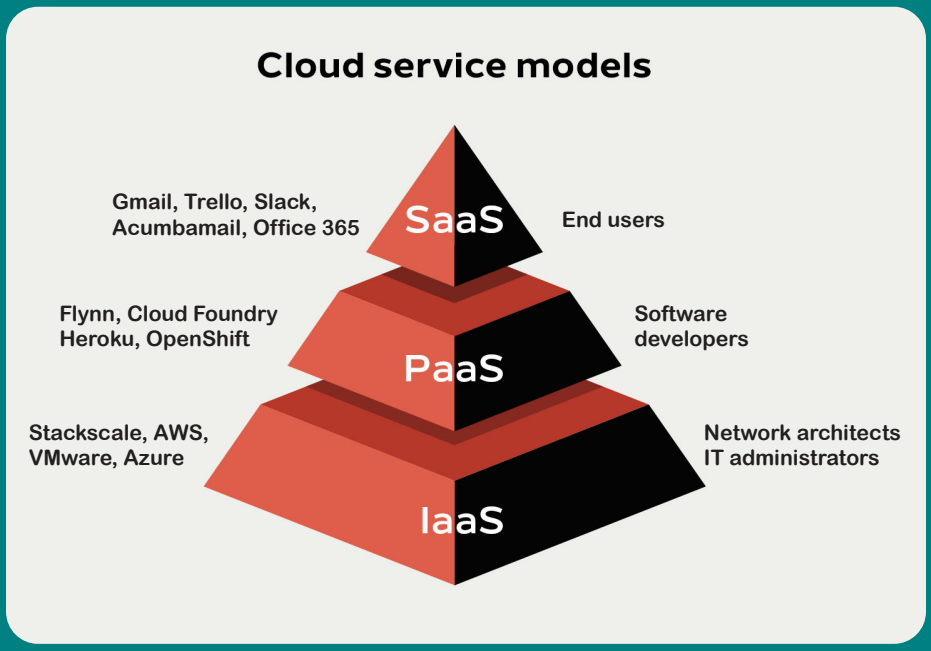
تهیه و تدوین: محمدجواد عبدالملکی

مقدمه

با ظهور شبکه گسترده اینترنت و ایجاد انواع جدید برقراری ارتباط، کسب‌وکارها نیز حول فضای جدید مجازی ساختار تازه‌ای به خود گرفتند. رقابت جهانی روزبه‌روز بیشتر شد و نهایتاً با ایجاد نیازهای جدید، روش‌های نوینی برای استفاده از توان رایانه‌ها مطرح‌شد و مورداستفاده قرار گرفت. روش جدید بهره‌بردن از توان رایانه‌ای، استفاده از رایانه‌ای در دوردست با کمک شبکه ارتباطی است که رایانش ابری نامیده می‌شود. درواقع رایانش ابری استفاده از توان رایانشی رایانه دیگر از طریق شبکه است.

روزبه‌روز بر اهمیت و کاربرد این فناوری با گسترش‌یافتن شبکه‌های بزرگ اینترنت و ایجاد نیازها و فرصت‌های جدید افزوده می‌شود. مشتریان رایانش ابری دارای زیرساخت‌های فیزیکی واقعی ابر نیستند، بلکه فقط با پرداخت هزینه اشتراک به فراهم‌کننده‌ی خدمات ابر، قادر به استفاده از منابع ابر و زیرساخت‌های آن می‌باشند. پس می‌توان نتیجه گرفت رایانش ابری موجب بهره‌وری کامل از سخت‌افزار و جلوگیری از هزینه اضافی می‌شود. وجود همه‌ی این عوامل موجب رشد سریع رایانش ابری در جوامع جهانی گردید.

لغت رایانش ابری اولین بار توسط جان مکارتی (John Mccarthy) در سال ۱۹۶۰ به‌صورت «رایانش ابر بالاخره روزی به‌عنوان یک نیازمندی عمومی در عرصه همگانی شناخته و سازماندهی خواهد شد» عنوان شد. در حقیقت این لغت به معنای تقسیم ویژگی‌ها در زیرساخت‌های مشترک می‌باشد. همچنین عبارت ابر در سال ۱۹۹۰ در حوزه تجارت به معنای شبکه بزرگ ATM شناخته می‌شد. در سال ۱۹۹۹ وب‌سایت salesforce.com توسط مارک بینوف و پارکر هرس راه‌اندازی شد. آن‌ها از فناوری‌های مختلفی در سایت‌هایی همچون google و yahoo استفاده می‌کردند. از دهه گذشته رایانش ابری موردتوجه زیادی قرارگرفته است. البته تمرکز اصلی در گذشته بیشتر بر روی نرم‌افزار به‌عنوان سرویس (SaaS) بود اما هم‌اکنون کاربردهای دیگری مانند زیرساخت به‌عنوان سرویس (IaaS) و همچنین پلتفرم به‌عنوان سرویس (PaaS) مطرح می‌شوند. شکل ۱ نمونه‌هایی از هر یک از این مدل‌ها، سطح کنترل و دسترسی در هر یک و کاربران خاص هر مدل را نمایش می‌دهد.



شکل ۱

علی‌رغم تمام امیدها جهت پیشرفت در حوزه امنیت رایانش ابری، به نظر می‌رسد که تهدیدات متنوعی در حال رشد و توسعه در این زمینه می‌باشند، به همین دلیل مطالعات و تحقیقات گسترده‌ای در زمینه حفظ امنیت داده در رایانش ابری صورت گرفته است، این امر نشان‌دهنده مبارزه بی‌وقفه در این زمینه می‌باشد. در این مقاله سعی شده است تا حد امکان مروری بر تهدیدات رایانش ابری و سپس راهکارهای ایمن‌سازی ارائه شود.

۱- مسائل امنیتی ابر

سازمان‌ها از مدل‌های گسترش‌یافته‌ای از محاسبات ابری نظیر مدل‌های خصوصی، عمومی، گروهی و ترکیبی استفاده می‌کنند. مسائل و نگرانی‌های امنیتی بسیاری در ارتباط با محاسبات ابری وجود دارد اما تمام این نگرانی‌ها به دو دسته کلی مسائل امنیتی مربوط به فراهم‌کنندگان محاسبات ابری و مسائل امنیتی مربوط به مشتریان آن تقسیم می‌شوند. در اغلب موارد، فراهم‌کننده باید از ایمن بودن زیرساختش مطمئن باشد و از داده‌های مشتریانش و برنامه‌های کاربردی محافظت کند. درحالی‌که مشتری باید از عملکرد فراهم‌کننده خدمات محاسبات ابری در راستای ایجاد معیارهای امنیتی مناسب برای حفاظت از داده‌هایش مطمئن شود. کاربرد گسترده مجازی‌سازی در پیاده‌سازی زیرساخت محاسبات ابری نگرانی‌های امنیتی یکسانی برای مشتریان و خدمات عمومی محاسبات ابری ایجاد کرده است.

مجازی‌سازی، جایگزین ارتباط بین سیستم‌عامل و سخت‌افزار در محاسبات، ذخیره‌سازی‌ها و حتی شبکه می‌شود. این امر باعث معرفی لایه جدیدی به نام لایه مجازی می‌شود که خود نیاز به تنظیم، مدیریت و امنیت صحیح دارد. نگرانی اصلی در این راستا شامل سازگاری نرم‌افزارهای مجازی یا hypervisor است.

اگرچه این نگرانی‌ها بیشتر به‌صورت تئوری مطرح می‌شوند اما می‌توانند در واقعیت وجود داشته باشند.

به‌عنوان‌مثال یک شکاف در ایستگاه کاری مدیریت که از نرم‌افزارهای مدیریتی مجازی استفاده می‌کند می‌تواند باعث ازکارافتادن یک پایگاه داده یا تنظیم مجدد آن به صورتی مطلوب برای مهاجم شود.

مسائل امنیتی متعددی برای محاسبات ابری که شامل بسیاری از فن‌آوری‌های شبکه ازجمله پایگاه‌های داده، سیستم‌عامل، مجازی‌سازی، برنامه‌ریزی منابع، مدیریت تراکنش، توازن بار، کنترل هم‌زمانی و مدیریت حافظه است وجود دارد. برای مثال، نگاشت از ماشین‌های مجازی به ماشین‌های فیزیکی باید به‌صورت ایمن انجام شود. امنیت داده‌ها شامل رمزنگاری داده‌ها و همچنین اطمینان از سیاست‌های مناسب برای به اشتراک‌گذاری داده‌ها، می‌باشد.

علاوه بر این، تخصیص منابع، مدیریت حافظه و الگوریتم باید امن باشد.

۲- کنترل امنیت ابر

معماری امنیتی ابر تنها در صورتی مؤثر است که پیاده‌سازی‌های دفاعی صحیحی در آن انجام گیرد. یک معماری امنیتی کارآمد، باید مسائلی که مدیریت امنیت را به وجود می‌آورند، تشخیص دهد.

مدیریت امنیت، این مسائل را با کنترل‌های امنیتی نشان می‌دهد. این کنترل‌ها برای حفاظت از نقاط ضعف در سیستم و کاهش اثر یک حمله در ابر لحاظ می‌شوند. درحالی‌که بسیاری از انواع کنترل پشت یک معماری امنیتی ابر وجود دارد، آن‌ها معمولاً در یکی از دسته‌های زیر قرار می‌گیرند:

کنترل بازدارنده (Deterrent Controls):

این کنترل‌ها برای کاهش حملات بر روی یک سیستم ابر در نظر گرفته‌شده است؛ مانند حصاری که به دور یک ملک کشیده شده است. کنترل بازدارنده به‌طورمعمول با توجه به سطح تهدید، با اطلاع‌رسانی به مهاجمان احتمالی گوشزد می‌کند که در صورت تداوم حمله عواقب نامطلوبی برای آن‌ها وجود خواهد داشت.

کنترل پیشگیرانه (Preventive Controls):

کنترل پیشگیرانه، تقویت سیستم در برابر حوادث است. به‌طورکلی اگر آسیب‌پذیری‌ها حذف نشوند، سطح امنیت ابر کاهش می‌یابد. به‌عنوان‌مثال، تأیید هوییت قوی از کاربران ابر که احتمال دسترسی کاربران غیرمجاز به سیستم‌های ابری را کمتر می‌کند، احتمال شناسایی مهاجمین را قبل از نفوذ افزایش می‌دهد.

کنترل تشخیصی (Detective Controls):

کنترل تشخیصی برای شناسایی و واکنش‌های مناسب به هر حادثه‌ای که رخ می‌دهد در نظر گرفته‌شده است. در صورت حمله، کنترل تشخیصی به کنترل پیشگیرانه و یا کنترل اصلاحی سیگنالی را جهت رسیدگی به این مسئله ارسال می‌کند. سیستم نظارت بر امنیت شبکه که شامل سیستم تشخیص نفوذ می‌شود، معمولاً تشخیص حملات در سیستم‌های ابر و زیرساخت‌های ارتباطی را پشتیبانی می‌کند.

کنترل اصلاحی (Corrective Controls):

کنترل اصلاحی به‌طورمعمول با محدود کردن آسیب، عواقب ناشی از حادثه را کاهش می‌دهد. آن‌ها در طول یا پس از وقوع حادثه اجرا می‌شوند. پشتیبان‌گیری یک سیستم سازشی به‌منظور بازسازی یک مثال از یک کنترل اصلاحی است.

۳- انواع حملات و تهدیدات امنیتی در رایانش ابری

بزرگ‌ترین چالش در اجرای موفق فن‌آوری‌های محاسبات ابری، مدیریت امنیت است. به‌محض اینکه برنامه‌های کاربردی حساس و داده‌ها به مراکز داده ابر، نقل‌مکان کنند، اجرا بر روی منابع محاسباتی مجازی در قالب ماشین مجازی ممکن است باعث بسیاری از نگرانی‌های امنیتی شود. شش تهدید امنیتی مهم مربوط به محاسبات ابری که توسط «اتحاد امنیت ابر» (CSA) کشف‌شده است عبارت‌اند از:

۱) استفاده نابجا از محاسبات ابری: این تهدید، تهدید برترِ شناسایی‌شده توسط CSA می‌باشد. در این روش مهاجمان می‌توانند به یک ابر عمومی نفوذ و با استفاده از قدرت زیرساخت‌های ابر و با آپلود نرم‌افزارهای مخرب، به هزاران کامپیوتر برای حمله به دستگاه‌های دیگر استفاده کنند. به‌عنوان‌مثال حملات DDoS.

۲) API ناامن (رابط برنامه‌نویسی کاربردی ناامن): APIها مجموعه‌ای از رابط‌های برنامه کاربردی هستند که توسط مشتریان برای تعامل با خدمات ابر استفاده می‌شوند. هنگامی‌که یک شخص ثالث (third party) شروع به ایجاد آن می‌کند، کاربر را با خطرات تهیه، محرمانه بودن و یکپارچگی داده‌ها مواجه می‌کند.

۳) آسیب‌پذیری فناوری به اشتراک گذاشته‌شده (shared): به‌عنوان ارائه‌دهنده پلتفرم ابر که توسط کاربران مختلف به اشتراک گذاشته می‌شود، این احتمال وجود دارد که این اطلاعات متعلق به مشتریان مختلف که در مرکز داده مشابه قرار دارند، باشند؛ بنابراین نشت اطلاعات غلط توسط یک مشتری می‌تواند به دیگران نیز انتقال داده شود.

٤) از دست رفتن (نشت) داده‌ها: از دست دادن داده یک مشکل شایع در محاسبات ابری است. اگر ابر ارائه‌دهنده خدمات رایانش ابری خدمات خود را به دلیل برخی از مشکلات مالی و حقوقی ببندد، پس‌ازآن احتمال از دست رفتن اطلاعات برای کاربرانش وجود خواهد داشت.

۵) ترافیک‌ربایی: ترافیک مشکل دیگری است که کاربران ابر باید از آن آگاه باشند. این تهدیدات شامل حملات man in the middle (MITM)، spamها و غیره می‌باشد.

٦) Insiderهای مخرب: این‌گونه تهدیدها شامل تقلب، آسیب و سرقت یا از دست دادن اطلاعات محرمانه ناشی از افراد خودی مورد اعتماد است. خودی‌های مخرب می‌توانند به‌وسیله توانایی خود برای نفوذ به سازمان و دارایی‌ها باعث ایجاد زیان بهره‌وری، آسیب نام تجاری و تأثیرات مالی و غیره شوند.

۷) دسترسی کاربری ممتاز: ارائه‌دهندگان ابر به‌طورکلی دسترسی نامحدودی به داده‌های کاربران دارند؛ بنابراین برای مقابله با خطر دسترسی کاربران ممتاز که منجر به خطر انداختن اطلاعات مشتریان می‌شود، لازم است کنترل‌هایی صورت گیرد.

۸) محل و تفکیک داده‌ها: کاربران و مشتریان ابر، نباید از محل ذخیره شدن داده‌هایشان مطلع باشند. چراکه در صورت اطلاع از محل داده‌ها می‌توانند آن‌ها را تفکیک کرده و به اطلاعات شخصی و محرمانه کاربران دیگر دست پیدا کنند.

۹) در دسترس بودن داده‌ها: حذف و یا گاهی اوقات در دسترس بودن داده‌ها یکی از خطرات مربوط به رایانش ابری می‌باشد. به‌خصوص زمانی که بخشی از سخت‌افزار ابر با توجه به نیاز مشتری به‌صورت پویا در اختیار او قرار می‌گیرد. در این صورت خطر حذف نشدن یک سری از داده‌ها که مشتری دیگری اقدام به حذف آن‌ها نموده است و یا داده‌های پشتیبان‌گیری شده قبلی وجود دارد.

۱۰) تهاجم به داده‌ها: یکی از خطرات امنیتی دیگر که ممکن است برای سرویس‌های رایانش ابری به وجود بیاید، هک شدن رمز عبور یا تهاجم به داده‌ها است. اگر فرد مهاجم به اطلاعاتی همچون رمز عبور حساب کاربری یک مشتری دسترسی پیدا کند، می‌تواند به‌تمامی منابع حساب کاربری تسلط پیدا کند؛ بنابراین رمز عبور به سرقت رفته به هکرها این اجازه را می‌دهد که تمامی اطلاعات بر روی دستگاه‌های مجازی را پاک‌کرده، تغییر داده و یا خدمات آن را غیرفعال کنند.

۱۱) از بین رفتن یکپارچگی داده‌ها: حفظ یکپارچگی داده‌ها جزء مهم‌ترین مسائل امنیتی قابل‌توجه در رایانش ابری است. چراکه داده‌های ذخیره‌شده در رایانش ابر ممکن است به هنگام انتقال و یا دریافت از فراهم‌کننده سرویس دچار مشکل شده و یکپارچگی داده‌ها از بین برود.

۱۲) چند مالکیتی: به اشتراک‌گذاری منابع سخت‌افزاری مانند CPU، تجهیزات حافظه، تجهیزات شبکه و غیره به‌وسیله چندین مشتری چند مالکیتی یک فناوری گفته می‌شود. برخلاف مدل‌های رایانش پیشین که منابع به یک مشتری اختصاص داده می‌شد، رایانش ابری مبتنی بر یک مدل شغلی می‌باشد که منابع به‌وسیله چندین مشتری در سطح شبکه، میزبان و اپلیکیشن به اشتراک گذاشته می‌شوند.

۱۳) مجازی‌سازی: مجازی‌سازی یک فناوری به‌منظور یکپارچه‌سازی و واقعی‌سازی منطقی منابع فیزیکی در یک سرور فیزیکی توسط چندین سرور مجازی می‌باشد. به‌عبارت‌دیگر مجازی‌سازی اجازه می‌دهد تا یک مجموعه سخت‌افزار بیش از یک ماشین مجازی را میزبانی کنند. این فناوری در بیشتر شرکت‌های بزرگ استفاده می‌شود و در تجارت‌های کوچک نیز رو به گسترش است.

۱٤) تمرکز داده‌ها و خدمات: خدمات و داده‌های مشتریان در منابع فیزیکی مشابهی مدیریت می‌شوند. به دلیل اینکه در مجازی‌سازی و چند مالکیتی همانند سرویس‌های کاربردی اینترنت موجود، بیشترین داده و خدمات متمرکزشده‌اند، لذا آسیب وارده از یک منبع فیزیکی یا سرویس فراهم‌کننده ابر می‌تواند به‌طور تصاعدی افزایش یابد. درواقع هنگامی‌که خدمات و داده‌های کلیدی بسیاری از مشتریان متمرکز می‌شود، آن‌ها به‌آسانی می‌توانند مورد هک یا حمله DDoS قرار گیرند. یک عملیات هک یا حمله DDoS می‌تواند منجر به اختلال در سرویس تمام مشتریان و لذا آسیب در مقیاس بالا شود. اگر چنانچه داده‌های ذخیره‌شده در سرور ابر سری و محرمانه باشند، با عملیات هک تمامی داده‌های مشتریان می‌تواند به بیرون درز پیدا کند.

۱۵) نرم‌افزار رابط مشتری و ابر: اگر رابط نرم‌افزاری که مشتری توسط آن به داده‌های خود در یک سرویس ارائه‌دهنده ابر دسترسی پیدا می‌کند، آسیب‌پذیر باشد، مهاجم می‌تواند با استفاده از این آسیب‌پذیری به ابر نفوذ کرده و یا حداقل موجب اخلال در عملکرد صحیح ابر شود.

۱٦) دسترسی غیرمجاز کارمندان ابر: ممکن است کسی از کارمندان شرکتی که سرویس رایانش ابری را ارائه می‌دهد، با توجه به انباشت داده‌های ارزشمند مشتریان در ابر، این انگیزه را پیدا کند که به داده‌های مشتریان دسترسی پیدا کند و اقدام به فروش یا هر سوءاستفاده دیگری از آن‌ها نماید.
با توجه به مواردی که در بالا ذکر شد، می‌توان به این نتیجه دست‌یافت که دلایل ریشه‌ای مشکلات امنیتی در رایانش ابری به‌صورت کلی به دو بخش تقسیم می‌شوند؛ بخش اول مربوط به اشتراک منابع زیرساختی بین مشتریان ابر (multi-tenancy) است، این یک حقیقت است که روش‌های جداسازی منابع بین مشتریان ابری آسیب‌پذیر می‌باشند. از سمت دیگر نیز آسیب‌پذیری‌ها و تهدیدات موجود، همان‌هایی هستند که به‌صورت عمومی وجود دارند با این تفاوت که در محیط ابر این تهدیدات مخاطرات بالاتری ایجاد می‌کنند. بخش دوم دلایل ریشه‌ای مشکلات امنیتی در ابر، به فقدان کنترل ناشی از برون‌سپاری داده‌ها و فرایندها برمی‌گردد. چراکه عدم اعتماد کامل به کارپذیری ابر وجود دارد. این عدم اعتماد ناشی از امکان درست‌کار ولی کنجکاو بودن و یا بالقوه خرابکار بودن مجموعه افرادی است که در سرویس ارائه‌دهنده ابر مشغول به کار می‌باشند.

٤- محدودیت‌ها و اقدامات متقابل برای تهدیدات و حملات در رایانش ابری

امروزه تحقیقات گسترده و متنوعی در حوزه‌های امنیتی رایانش ابری به‌منظور به حداقل رساندن تهدیدات امنیتی آن، درحال‌توسعه می‌باشد. تهدیدات امنیتی همانند ضوابط و کنترل سری مکالمات در بین ماشین‌های مجازی و بهره‌برداران، آسیب‌پذیر است و درمان آن به‌طور ویژه با فناوری امنیتی ابر اختصاصی در این زمینه توسعه داده‌شده است. در قسمت‌های بعد ابتدا محدودیت‌های فناوری امنیتی موجود برای مقابله با تهدیدات امنیتی رایانش ابری تشریح و سپس اقدامات تکنیکی و اجرایی برای تکمیل آن‌ها ارائه می‌شود.

٤-۱- محدودیت‌های فناوری امنیتی

کنترل و مدیریت دسترسی: فناوری‌های امنیتی موجود به‌منظور کنترل دسترسی به‌عنوان سدی در بالادست یک سیستم فیزیکی که توسط IP یا پورت آن شناسایی می‌شود، نصب می‌گردد؛ بنابراین با توجه به نقش هر مشتری برای میزان دسترسی موردنیاز آن، مدیریت می‌شود.

مانیتور کردن ماشین مجازی: فناوری‌های امنیتی موجود از روی IP و پورت، ماشین مجازی را تشخیص می‌دهند و آن ماشین را با استفاده از نرم‌افزارهای مدیریت ماشین مجازی کنترل و یا مانیتور می‌کنند. به عبارتی، این فناوری‌ها برای کنترل دسترسی هر ماشین مجازی و مانیتور هک داخلی در بین ماشین‌های مجازی ضروری است.

مانیتور هک بین ماشین‌های مجازی: مانیتور هک بین ماشین‌های مجازی یک سدی است که در ماشین مجازی مشتری نصب می‌شود و تنها می‌تواند همان ماشین مجازی که روی آن نصب‌شده است را مانیتور کند. راه‌حل امنیتی موجود نمی‌تواند هک را در بین ماشین‌های مجازی که باهم ارتباط داخلی دارند مانیتور کند.

آنتی‌ویروس: اگرچه فناوری‌های امنیتی متداول به‌وسیله نصب آنتی‌ویروس در تمام ماشین‌های مجازی چندگانه و یا در سیستم مدیریت این ماشین‌های مجازی به کار گرفته می‌شوند، ولی آن‌ها بار زیادی بر روی تجهیزات ایجاد می‌کنند. همچنین، تشخیص ویروس در بسیاری موارد مشکل است.

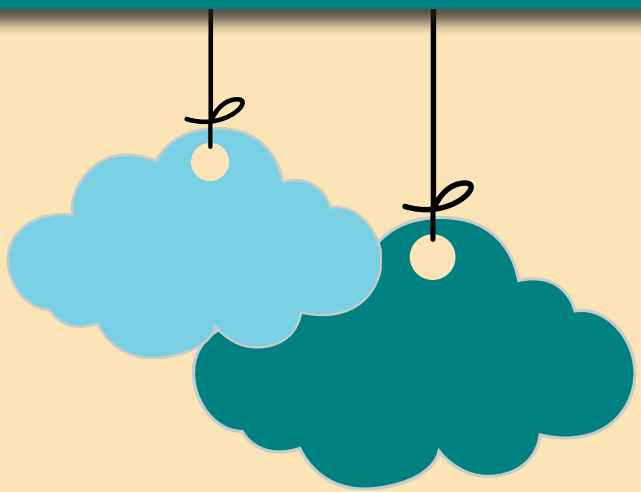
٤-٢- اقدامات متقابل امنیتی پیشنهادی

جداسازی شبکه‌های ماشین‌های مجازی: زمانی که بدافزار می‌تواند از یک ماشین مجازی هک شده به ماشین مجازی مجاور انتقال پیدا کند، نیاز است که شبکه‌های ماشین‌های مجازی از یکدیگر جدا شوند. یک روش معمول برای جداسازی شبکه‌های ماشین‌های مجازی استفاده از VLAN می‌باشد. باوجود جداسازی یک هاپ (hop) از VLAN، باز هم امکان حمله به آن وجود دارد؛ بنابراین علاوه بر VLAN بومی، مسیریاب و سوئیچ VLAN نیز می‌بایست به‌درستی پیکربندی شوند.

به‌کارگیری سیستم‌های IDS، IPS و مانیتورینگ: سیستم تشخیص نفوذ (IDS) بدون اجازه بالادست فیزیکی نصب و به‌کار گرفته می‌شود. این سیستم قادر به شناسایی ساختمانی فناوری مجازی نیست؛ اما به‌کارگیری سیستم تشخیص نفوذ، مانیتور و ورود بدون اجازه که مجازی شده باشد و از طرفی قادر به آنالیز ترافیک تولیدشده به‌وسیله ماشین‌های مجازی، تعیین و جلوگیری از حمله DDoS شود، موردنیاز است.

پنهان‌سازی داده‌های محرمانه: اگر داده‌هایی که درواقع به امانت گذاشته‌شده‌اند پنهان نگردند، احتمال بالایی وجود دارد که به علت اشتباه عمدی و یا بی‌دقتی توسط مجری به‌صورت غیرقانونی در دسترس قرار گیرند؛ بنابراین داده‌های کلیدی و مهم همانند داده‌های امانت‌گذاری شده بایستی پنهان شوند.

محدود کردن دسترسی به منابع: هنگامی‌که چندین مشتری با استفاده از فناوری مجازی از منابع فیزیکی معین استفاده می‌کنند، می‌توانند به‌واسطه محدود نمودن ظرفیت CPU اختصاص داده‌شده به یک مشتری از حمله DDoS و بدافزارها جلوگیری نمود. علاوه بر این برای یک فراهم‌کننده سرویس با سرعت‌بالای مجازی‌سازی، منابع و لودهای اضافی یک ماشین نسبت به ماشین‌های مجازی دیگر یک هشدار امنیتی محسوب می‌شود؛ بنابراین نرخ مجازی‌سازی بایستی در یک سطح مناسب با توجه به اهمیت آن سرویس فراهم گردد.



معرفی نسل‌های آینده فایروال

تهیه و تدوین: محمد ساروقی

فایروال‌ها تجهیزات سخت‌افزاری یا چهارچوب‌های نرم‌افزاری هستند که از دسترسی‌های غیرقابل‌تأیید جلوگیری می‌کنند. فایروال‌ها می‌توانند هم به‌صورت سخت‌افزاری و هم به‌صورت نرم‌افزاری یا ترکیبی از هر دو رویکرد پیاده‌سازی شوند.
از فایروال‌ها می‌توان برای جلوگیری از دسترسی سیستم‌های داخلی به شبکه اینترنت و همچنین بلعکس، یعنی دسترسی سیستم‌های خارج از شبکه محلی به سیستم‌های خصوصی داخلی استفاده کرد. تمام اطلاعات ورودی و خروجی بین شبکه محلی و اینترنت از طریق فایروال در دروازه ورودی شبکه عبور می‌کنند و بسته‌ها موردبررسی قرار می‌گیرند و مانع از عبور بسته‌هایی که در تضاد با معیارهای که از پیش تعریف‌شده هستند، می‌شوند. این امر باعث جلوگیری از نفوذ بسته‌های تعریف و ساخته‌شده توسط هکرها به سیستم‌های داخلی در شبکه شما می‌شود. فایروال‌ها عموماً دسترسی بین شبکه داخلی و اینترنت را محدود می‌کنند که قابلیت اعمال محدودیت‌ها بر روی ترافیک عبوری از شبکه داخلی به سمت اینترنت را نیز دارا می‌باشند. این محدودیت‌ها که به‌عنوان Policy شناخته می‌شوند توسط مدیران شبکه تعریف می‌گردد، فایروال‌ها یکی از حساس‌ترین تجهیزات شبکه می‌باشند؛ زیرا در صورتی که Policy مناسب تعریف نشود منابع اطلاعات داخلی با خطر مواجه می‌شوند. فایروال در یک شبکه علاوه‌بر اعمال محدودیت‌های تأییدشده از سوی مدیر شبکه، قابلیت مانیتورینگ حجم ترافیک و نوع فعالیت را به مدیر شبکه می‌دهد.

امنیت در کامپیوترها، کلمه فایروال یا دیوار آتش را از عبارات Fire fighting و Fire aversion به‌دست آورده‌اند و به مفهوم ساختن حدومرز برای جلوگیری از شعله‌های آتش است. از زمانی که سازمان‌ها اجازه دسترسی به سرورهای خود را به کاربران دادند، امکان کنترل دسترسی به یک نیاز تبدیل شد. قبل از ساخت فایروال‌ها در اواخر دهه ۱۹۸۰، اصلی‌ترین نوع کنترلِ امنیت سیستم‌ها توسط لیست کنترل دسترسی ACL که روی تجهیزات سوئیچینگ قابلیت پیاده‌سازی را داشت، استفاده می‌شد. توسعه‌دهندگان به این نتیجه رسیدند که ACL راهکاری برای جلوگیری از مهاجمان نیست و تنها روشی برای کنترل مسیردهی بین سیستم‌های مختلف می‌باشد، در سال ۱۹۹۲ اولین فایروال‌های تجاری DES SEAL با هدف خاص (جلوگیری از مهاجم سایبری) وارد بازار شدند.

فایروال‌های نسل بعد

فایروال‌های نسل بعد تجهیزات سخت‌افزاری یا چهارچوب امنیتی مبتنی بر برنامه‌نویسی هستند که می‌توانند با اجرای رویکردهای امنیتی در سطح برنامه‌های کاربردی و همچنین همانند فایروال‌های نسل اول در لایه انتقال و شبکه نیز قابلیت اعمال محدودیت بر روی بسته‌های عبوری را داشته باشند. فایروال‌های نسل دوم، یک سیستم هماهنگ و جامع هستند که تمام قابلیت‌های نسل اول را نیز دارا می‌باشند؛ به‌عنوان‌مثال، بازرسی بسته‌ها به‌صورت عمیق (DPI)، سیستم‌های جلوگیری از نفوذ (IPS)، واسط SSL و SSH، محدودسازی وب‌سایت‌ها، مدیریت پهنای باند و QOS. فایروال‌های نسل بعد در واقع نوعی مدیریت یکپارچه تهدید UTM هستند، بااین‌وجود اصطلاح UTM به مواردی اطلاق

می‌شود که دارای ابزارهای بیشتر مانند ضد ویروس، ضد هرزنامه و غیره می‌باشند؛ که به نسبت ابزارهای فایروال نسل دوم دارای کاربرد خاص و کمتر هستند. کاربرد اصلی مدیریت یکپارچه تهدید، در سازمان‌های کوچک و متوسط می‌باشد و فایروال‌های نسل دوم در سازمان‌های بزرگ استفاده بیشتری دارند.

ارزیابی فایروال‌های نسل بعد

تهدیدات مهم مانند حملات بدافزارها، حملات هدفمند به سرویس‌های خاص و حملات لایه کاربرد و غیره به‌سرعت در حال افزایش هستند. در حقیقت بیش از ۸۰ درصد کل بدافزارها و اقدامات نفوذی از آسیب‌های موجود در برنامه‌ها به‌جای نقاط ضعف در مؤلفه‌های مدیریتی سیستم‌ها استفاده می‌کنند. فایروال‌های نسل اول با اعمال فیلترینگ بر روی بسته‌ها بسیار خوب عمل می‌کنند و هر زمان که مدیر شبکه به محدود کردن Port یا IPها اقدام کند، سرویس‌ها از دسترس بسته‌ها خارج می‌شوند اما امروزه سرویس‌هایی مانند Share point و سرویس‌های وب که از ۸۰ Port برای انجام امور خود استفاده می‌کنند را نمی‌توانیم فیلتر کنیم زیرا باعث مسدودشدن برنامه‌ها می‌شوند. وجود این مشکل باعث شد سازمان‌ها به دنبال بهبود رویکرد امنیتی باشند، به همین دلیل نیاز به وجود فایروال‌هایی با توانایی بازرسی عمیق در بسته‌ها به وجود آمد.

در گذشته نیاز به یک چهارچوب یکپارچه شامل آنتی‌ویروس، سیستم‌های جلوگیری از نفوذ، فیلترینگ URL وجود داشت که این تنها بخشی از سرویس‌هایی است که در مدیریت یکپارچه تهدید UTM تصور می‌شد؛ اما این سیستم با این قابلیت‌ها سرعت لازم را ندارد بنابراین ایده ایجاد فایروال‌های نسل بعد ایجاد شد.

NGFWها برای بازرسی از بسته‌ها و جلوگیری از نفوذ در نظر گرفته‌شده‌اند، بسیاری از ابزارهای موجود در UTM به سرورهای مختص به خود منتقل شدند بنابراین عملکرد NGFW بهبود یافت و سرعت عملکرد تا چندین گیگ بیت افزایش پیدا کرد که بااین‌وجود هنوز سرعتِ مناسب بسیاری از سازمان‌های خاص نمی‌باشد.

فایروال‌های نسل دوم می‌توانند اطلاعات کاربردی و کنترل، جلوگیری از نفوذ، جلوگیری از بدافزار و بازرسی SSL را انجام دهند، همچنین به مدیران شبکه، قدرت کنترل و نظارت بر برنامه‌های کاربردی را خواهند داد. علاوه بر این فایروال‌های نسل دوم می‌توانند ترافیک رمزنگاری‌شده توسط SSL را اعمال کنند و اطمینان حاصل کنند که این کار باعث تبدیل‌شدن به یک بردار حمله نمی‌شود. نکته مهم که در انتخاب فایروال‌های نسل دوم وجود دارد انتخاب پردازنده متناسب با کاربرد شبکه است.

فایروال‌های نسل اول در لایه Network و Transport با مسدود کردن IP و Port مربوط به پروتکل‌ها برای مقابله با انواع حملات سایبری یک‌مرحله‌ای و پایدار که قابل‌مشاهده‌اند، سروکار دارند. اخیراً حملات سایبری ناپایدار به‌وجود آمده‌اند و بر روی برنامه‌ها و داده‌های حساس تمرکز دارند که برای دفاع در برابر حملات پیچیده مبتنی بر لایه کاربرد، نیاز فایروال‌ها به استفاده از لایه‌های مختلف در مدل OSI برای کشف حملات پیچیده شده است، شکل ۱ انواع مهاجمان و انگیزه چنین حملاتی را نشان می‌دهد.

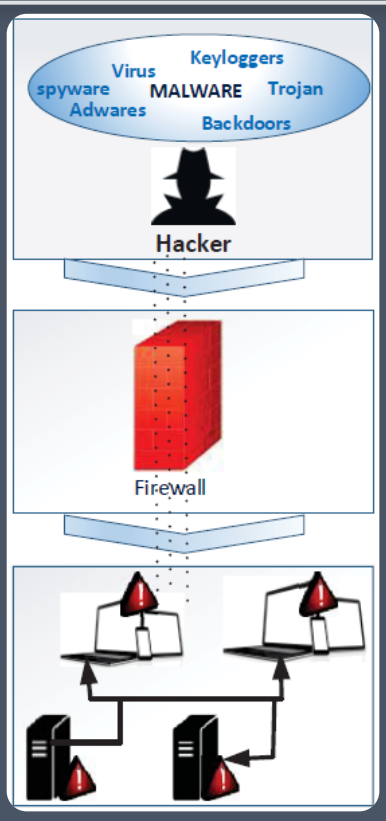
شکل ۱

آسیب‌پذیرهای فایروال‌های نسل اول که منجر به تکامل فایروال‌های نسل دوم شده‌اند:

۱. Advanced Evasion Technique

در این روش مهاجم بسته‌های حاوی محتوای مخرب را به تعداد بسته‌های زیاد تقسیم می‌کند، به‌گونه‌ای که وجود محتوای مخرب در بسته‌ها قابل‌تشخیص نباشد.

در شکل ۲ مهاجمان با استفاده از این روش حمله خود را تبدیل می‌کنند و به شبکه که توسط بسته‌های عادی کشف نشده بودند، نفوذ کرده و با موفقیت به اهداف خاص خود نفوذ می‌کنند.



شکل ۲

۲. Targeted Cyber Attack

حملات سایبری هدفمند، حملات مخربی هستند که در آن سیستم‌های خاص به‌جای یک شبکه کامل ضمن ناشناس بودن، در معرض حمله قرار می‌دهند. یک مهاجم ابتدا اطلاعات در دسترس عموم را شناسایی و جمع‌آوری می‌کند سپس بر اساس اطلاعات جمع‌آوری‌شده به شبکه‌ی هدف نفوذ و سطح دسترسی خود را افزایش می‌دهد تا به هدف خود دسترسی پیدا کند.
مهاجم می‌تواند از روش‌های مختلف ازجمله Phishing، Zero-day Attack و مهندسی اجتماعی و بسترهای نرم‌افزاری برای جذب افراد استفاده کند.
مراحل انجام این حملات به‌صورت زیر می‌باشد:

a) Intelligence Gathering
b) Point of Entry
c) Command and Control
d) Lateral Movement
e) Maintenance
f) Data Exfiltration

۳. حمله به برنامه‌های تحت وب

آسیب‌پذیری برنامه وب با استفاده از حملات Injection تقریباً بیش از یک دهه است که منبع شماره یک تهدید محسوب می‌شود. بردار اصلی تهدید برای یک وب‌سایت این است که برنامه‌های وب محدودیت دسترسی ندارند.

٤. حملات متمرکز بر داده‌ها

دلیل اصلی شروع حملات، سرقت داده‌های محرمانه و ایجاد جریان داده است که حتی قربانی در زمان به‌خطر افتادن داده‌های خود نمی‌داند کدام بخش از داده‌های حساس به خطر افتاده است؛ بنابراین توجه به جریان داده‌ها برای محافظت از هر فرد یا سیستم در برابر نقض جریان داده بسیار مهم است.

Gartner Magic Quadrant بیانگر چیست؟

روش و فرآیندی برای بررسی معیارهای میزان تکامل اهداف کلی و قدرت اجرایی اهداف که برای ارزیابی شرکت‌ها در زمینه فناوری اطلاعات بررسی‌های متفاوت توسط موسسه Gartner انجام می‌شود اطلاعات توسط موسسه فاش نمی‌شود و امتیازات کسب‌شده درمجموع جایگاه شرکت‌ها را در بازار، در یکی از ۴ بخش مربع تعیین می‌کند.
این بخش‌ها عبارت‌اند از:

رهبران:
این شرکت‌ها در هر دو زمینه امتیازات بیشتری را کسب کرده‌اند و عموماً دارای جمعیت بیشتر و پختگی در زمینه فعالیت تخصصی خود هستند.

چالشگران:
این دسته شامل شرکت‌هایی است که توانایی اجرایی بالایی دارند؛ اما در زمینه جامعیت بینش و تکامل اهداف درازمدت امتیاز پایین‌تری نسبت به رهبران کسب‌کرده‌اند و با کسب دید مناسبی در این راستا می‌توانند به‌عنوان چالشگر سهم بازار رهبران را به چالش کشیده و تهدید کنند.

ایده پردازان:
این شرکت‌ها امتیاز کمتری در زمینه اجرای اهداف به دست آورده‌اند اما در زمینه تکامل اهداف امتیاز بیشتری دارند. عموماً این شرکت‌ها کوچک‌تر از چالشگران هستند و علی‌رغم داشتن دید مناسبی از آینده، جای پیشرفت محسوسی در زمینه افزایش

قدرت اجرایی شرکت دارند.

فرصت جویان:
این دسته از شرکت‌ها در هر دو زمینه‌ی تکامل اهداف و قدرت اجرایی امتیاز کمتری نسبت به دیگران کسب کرده‌اند و عموماً شرکت‌های تازه‌وارد به بازار هستند و سهم کوچکی از بازار را به خود اختصاص می‌دهند که گاهاً انواعی خاص از مصرف‌کننده را هدف می‌گیرند و وارد بخش‌های کلی بازار نمی‌شوند.

Gartner Magic Quadrant که برای فایروال‌های نسل بعد در سپتامبر ۲۰۱۹ منتشر شد، در شکل ۳ نشان داده‌شده است:

شکل ۳

ویژگی‌های محصولات تجاری

فایروال‌های نسل بعد برای اینکه بتوانند وظایفی که بر عهده آن‌ها گذاشته می‌شود را به‌خوبی انجام دهند باید ویژگی‌های خاصی داشته باشند، درواقع محصولات تجاری برای شبکه‌هایی با مقیاس متفاوت تولید شده و با توجه به شرکت سازنده میتوانند دارای ویژگی‌های متفاوتی به‌صورت زیر باشند:

- **Dynamic Dashboard**

دارای یک داشبورد پویا می‌باشد که امکان مشاهده و رصد آخرین فعالیت‌های شبکه را میسر می‌سازد. منطقه بندی امنیت، موجب کاربرپسندی بیشتر مفاهیم امنیت و اعمال سیاست‌های امنیتی می‌شود.

- **Notification**

امکان ارسال ایمیل اخطار از فعالیت‌های شبکه و دستگاه در این بخش قابل تنظیم می‌باشد. همچنین آخرین اطلاعات شبکه و وضعیت سیستم با ارسال SMS به اطلاع مدیر امنیت سازمان می‌رسد.

- **New & Alerts**

آخرین اخبار و پیام‌های اخطار سیستمی و امنیتی در داشبورد دستگاه قابل‌مشاهده می‌باشد. با کلیک بر روی هر پیام می‌توان اطلاعات کامل‌تر را از سامانه دریافت نمود.

- **Licensing**

لایسنس قابل افزایش از دیگر ویژگی‌های آن می‌باشد.

- **CPU, RAM usage**

وضعیت لحظه‌ای و آنلاین مصرف CPU و حافظه در صفحه نخست پنل مدیریتی قابل مانیتور می‌باشد.

- **Routing (Static, Dynamic)**

علاوه بر جدول مسیریابی پیش‌فرض، می‌توان با تعریف قوانین سیاست‌گذاری روتین ایستا و پویا، فرایند مسیریابی را بهبود داد.

- **DHCP**

زمانی که یک کلاینت (یک کامپیوتر میزبان و یا یک دستگاه تحت شبکه مانند یک چاپگر با قابلیت شبکه) به شبکه متصل شود، به‌صورت خودکار یک IP از رنج شبکه تعریف‌شده در DHCP دریافت خواهد کرد.

- **Dynamic DNS**

مدیر شبکه می‌تواند از سرویس DNS پویا که توسط شرکت‌هایی مانند Dyn-DNS ارائه می‌گردد، استفاده نماید.

- **VLAN**

ایجاد VLANها، ترکیب شبکه‌های مجازی با مناطق فایروال و افزودن یک سطح برای جداسازی بین مناطق امنیتی مختلف می‌باشد.

- **Security**

امنیت در سطح دروازه، محافظت را به نقطه ورودی سازمان می‌برد. لایه‌های مختلف امنیتی در نظر گرفته‌شده است. Firewall، IPS،Gateway Antivirus، Content&URL Filtering،Email Security از مهم‌ترین ویژگی‌های امنیتی این دستگاه می‌باشد.

- **Web Security**

یکی از ماژول‌های امنیتی، امنیت وب می‌باشد. مدیر شبکه با استفاده از این ماژول می‌تواند فعالیت‌های وب کاربران را کنترل نماید.

- **Email Security**

یکی از راه‌های متداول انتشار آلودگی در سازمان، پست الکترونیکی می‌باشد. همه‌ی ورودی‌ها و خروجی‌های ایمیل از طریق پروتکل‌های POP۳ و SMTP نظارت می‌شوند.

- **Intrusion Prevention System**

دستگاه مجهز به سامانه سیستم تشخیص نفوذ (IDS) و سامانه پیشگیری از نفوذ (IPS) می‌باشد. این سامانه‌ها با شناسایی بسته‌های مهاجم از حمله و نفوذ آن‌ها به‌صورت پیشگیرانه جلوگیری می‌نمایند.

- **Cache Management**

دستگاه مجهز به سامانه مدیریت حافظه نهان می‌باشد. مدیریت Cache علاوه‌بر افزایش چشمگیر سرعت اینترنت، در میزان پهنای باند مصرفی و سهمیه اینترنت صرفه‌جویی قابل‌ملاحظه‌ای را برای سازمان به ارمغان می‌آورد.

- **Stateful Inspection Firewall**

فایروال از چند ماژول برای مانیتورینگ و مسدود کردن انواع ترافیک‌های شبکه تشکیل‌شده است: Port forwarding / NAT، Outgoing traffic، Inter-Zone traffic، VPN traffic، System access.

- **Quality of Service (QoS)**

هدف از ماژول QoS اولویت بخشیدن به ترافیک IP بر اساس سرویس می‌باشد که به‌منظور افزایش کارایی و بهینگی شبکه ضروری می‌باشد. به‌عبارت‌دیگر QoS یک روش آسان برای رزرو و گارانتی مقدار پهنای باند (ترافیک ورودی و خروجی) تخصیصی برای یک سرویس معین می‌باشد. معمولاً کاربردهای کنترلی و تعاملی مانند SSH یا VOIP نیاز به اولویت‌بندی ترافیک شبکه دارند.

- **VPN (SSL & IPSec)**

یک ارتباط VPN می‌تواند دو شبکه محلی مجزا را از طریق بستر ارتباطی عمومی و معمولاً ناامن مانند اینترنت، به‌صورت مستقیم به هم متصل گرداند. همه‌ی ترافیک‌های شبکه در میان اتصال VPN به‌صورت امن در داخل یک تونل رمزنگاری‌شده، خارج از دید شخص ثالث منتقل می‌گردند. همانند یک پیکربندی Site-to-Site VPN، همچنین یک کامپیوتر راه دور در هر مکانی، می‌تواند از طریق اینترنت و با استفاده از یک تونل VPN به شبکه محلی LAN مطمئن متصل گردد. دستگاه علاوه بر پشتیبانی از ایجاد VPNها بر روی پروتکل IPsec که توسط اکثر سیستم‌عامل‌ها و تجهیزات شبکه پشتیبانی می‌گردد، از پروتکل OpenVPN نیز پشتیبانی می‌کند.

- **Multi-WAN (with Failover)**

امکان استفاده از چندین اتصال اینترنت به‌طور هم‌زمان وجود دارد.

- **Centralized Management**

مدیریت متمرکز با استفاده از صفحه وب میسر می‌باشد. امکان تعریف انواع سیاست‌گذاری‌های امنیتی و شبکه میسر می‌باشد.

- **Quote Management (Accounting for users)**

جهت بهینه‌سازی مصرف و سرعت اینترنت درون سازمان، مدیر شبکه می‌تواند علاوه بر اعمال سیاست‌های QoS، میزان سهمیه مصرفی برای هر فرد یا گروه را تعیین نماید. امکان مشاهده کاربران آنلاین و صفحه‌ی در حال مشاهده، میزان حجم مصرفی و باقیمانده و ویرایشِ قطع یا وصل اینترنت فرد/گروه در این ماژول قابل‌اعمال می‌باشد.

- **Monitoring**

مدیر سازمان می‌تواند به‌صورت آنلاین از وضعیت سیستم، کانکشن‌های درحال اجرا، شبکه، نمودارهای سیستم، پراکسی و اتصالات VPN مطلع گردد.

- **Logs & Reports**

یکی از بخش‌های مهم و کلیدی UTM گزارش‌گیری و وقایع‌نگاری آن می‌باشد. امکان مشاهده، چاپ و دریافت فایل خروجی با فرمت‌های PDF و XLS از گزارش‌های Content Filtering، Firewall، IPS، Visited Site میسر می‌باشد.

SANDBOX

سندباکس و انواع آن

تهیه و تدوین: محمدجواد عبدالملکی

مقدمه

با توجه به پیشرفت روزافزون تکنولوژی در کنار استفاده‌های مفید از آن مخاطرات و تهدیداتی نیز به‌وجود آمده است. فناوری اطلاعات به‌عنوان یکی از شاخه‌های مهم تکنولوژی از این قاعده مستثنی نیست و تهدیدات مربوط به این حوزه همواره افزایش یافته و پیچیده‌تر می‌شوند. طبیعتاً این امر موجب بی‌اعتمادی کاربران نسبت به نرم‌افزارها و یا حتی کدها و اسکریپت‌هایی می‌شود که کاربران به ایجادکننده و توسعه‌دهندگان آن‌ها اعتماد ندارند. از طرفی ممکن است کاربران به این موارد نیاز داشته باشند و درنتیجه مجبور به استفاده از آن‌ها شوند؛ بنابراین، نیاز به ابزاری که بتوان توسط آن از آلوده و مخرب نبودن نرم‌افزارها، کدها و اسکریپت‌ها اطمینان حاصل کرد به‌وجود می‌آید.

شکل ۱. سندباکس

۱- تعریف sandbox

به‌منظور جلوگیری از آسیب رسیدن به سیستم‌عامل و نرم‌افزارهای موجود در آن، ابزاری طراحی‌شده است که محیطی حفاظت‌شده را جهت اجرای نرم‌افزارهای مشکوک به داشتن ویروس یا هر کد مخربی در سیستم‌عامل ایجاد می‌کند. این محیط امن که در برخی آنتی‌ویروس‌ها و بسته‌های امنیتی نیز مشاهده می‌شود، بانام سطل شنی یا همان sandbox شناخته می‌شود. sandbox با ایجاد یک محیط حفاظت‌شده مانع از دسترسی فایل مشکوک اجراشده به بخش‌های حساس سیستم‌عامل به‌منظور تغییر دادن تنظیمات سیستم و مانیتور کردن پردازش‌ها می‌شود. به‌عنوان نمونه، برخی ویروس‌ها برای پنهان کردن خود حالت hidden یا مخفی دارند و با تغییری جزئی در registry ویندوز اقدام به غیرفعال کردن نمایش فایل‌های مخفی‌شده می‌کنند. برخی از آن‌ها حتی دسترسی کاربر به task manager و registry editor را نیز مسدود می‌کنند تا کاربر نتواند آن‌ها را اجرا کند و مانع از کارکرد ویروس شود. چرا که یکی از راه‌های ساده برای شناسایی ویروس، مشاهده و بررسی task manager بوده و به‌علاوه تنظیمات کاربر در registry ذخیره می‌شود. پس اگر ویروس مانع از دسترسی کاربر به آن‌ها شود، می‌تواند آسوده اجرا شود و به عملیات خرابکارانه خود ادامه دهد. البته کاربران می‌بایست توجه داشته باشند که برخی از بدافزارها طوری طراحی‌شده‌اند که در محیط sandbox رفتاری عادی از خود نشان بدهند و پس از اجرا در محیط اصلی سیستم‌عامل، ماهیت خرابکارانه خود را بروز دهند.

شکل ۲. سندباکس، محیطی نه کاملاً امن

۲- انواع سندباکس

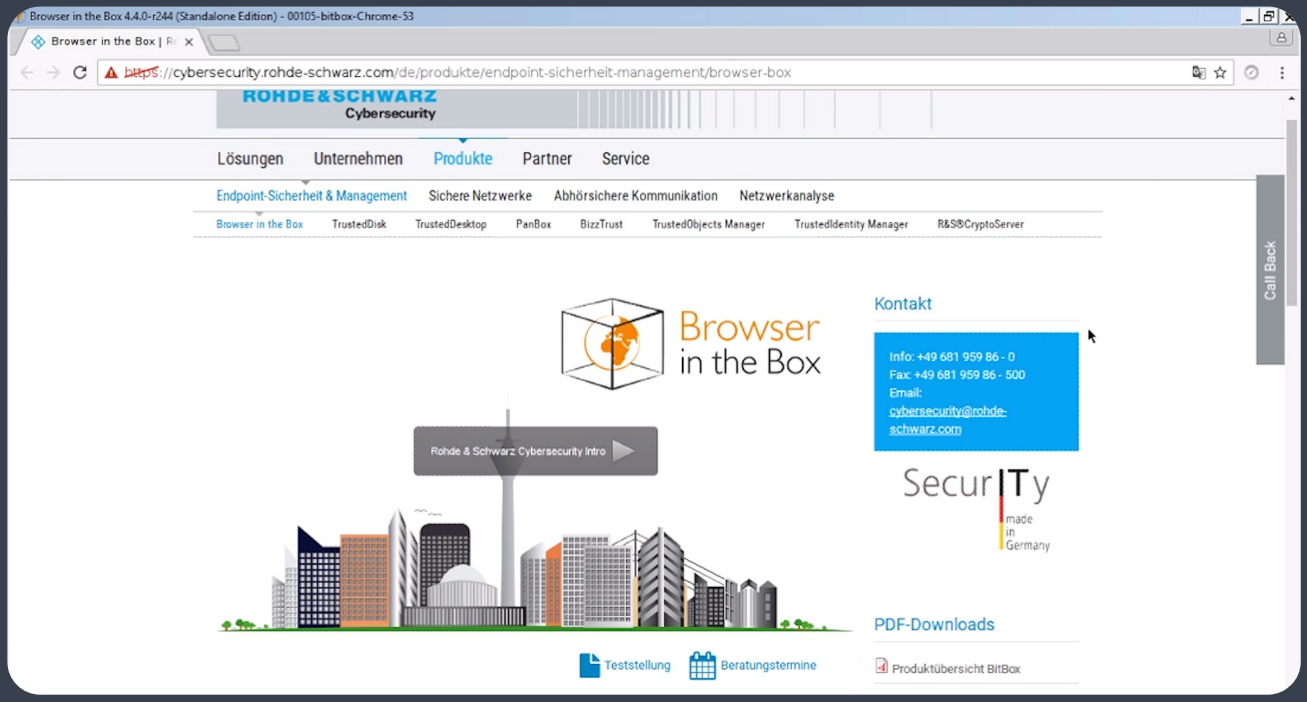
سندباکس‌ها بر اساس نحوه عملکرد به‌طورکلی به دو دسته آنلاین و آفلاین تقسیم می‌شوند. سندباکس‌های آنلاین برای بررسی صفحات وب یا کد-اسکریپت‌های مشکوک به‌کاربرده می‌شوند.

نحوه کار این نوع سندباکس‌ها بدین‌صورت است که ابزار ارائه‌دهنده سندباکس، قسمت کوچکی از سخت‌افزار سرور خود را برای ایجاد یک ماشین مجازی (Virtual Machine) تخصیص می‌دهد. این ماشین مجازی دسترسی‌های فایل مشکوک را به تنظیمات حساس سیستم قطع کرده و به بررسی رفتارهای فایل مشکوک می‌پردازد. سندباکس‌های آفلاین که به‌صورت یک نرم‌افزار بر روی سیستم‌عامل کاربر نصب و اجرا می‌شوند، به‌طور مشابه با استفاده از اختصاص دادن بخشی از سخت‌افزار سیستم کاربر به یک ماشین مجازی محیطی امن را ایجاد می‌کنند. این محیط امن بسته به نیازها و تنظیمات مدنظر کاربر می‌تواند محدودیت‌هایی برای نرم‌افزار یا فایل مشکوک، در دسترسی به تنظیمات حساس ایجاد کند تا در صورت آلوده بودن فایل، آسیبی به سیستم کاربر نرسد.

۲-۱- سندباکس‌های آنلاین

۲-۱-۱- سندباکس BitBox

از محبوب‌ترین سندباکس‌های موجود می‌توان به سندباکس BitBox که کوتاه شده عبارت Browser in the Box می‌باشد اشاره نمود. این سندباکس به‌طور خاص برای مرور وب در محیط سندباکسی طراحی شده است.
این نرم‌افزار دارای دو نوع Chrome و Firefox بوده و تقریباً شبیه نمونه‌ای از Virtualbox مورداستفاده در لینوکس می‌باشد. بدین معنی که این ابزار یک سیستم‌عامل مجازی لینوکس سبک جهت ایجاد محیط امن موردنیاز استفاده می‌کند. شکل ۳ نمایی از محیط نسخه chrome این ابزار را نشان می‌دهد.



شکل ۳. محیط سندباکس BitBox نسخه chrome

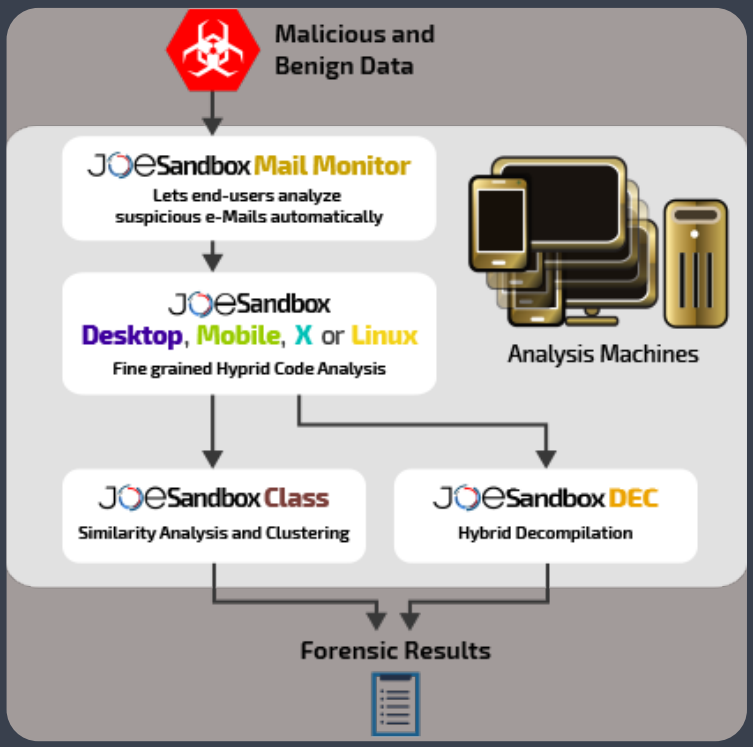
این نرم‌افزار به‌طور مستقل از دو مرورگر نامبرده کار می‌کند. یعنی کاربر می‌تواند مرورگرهای Chrome و Firefox را مستقلاً بر روی سیستم خود داشته باشد و از آن‌ها نیز استفاده کند. سندباکس BitBox برای کاربر این قابلیت را ایجاد می‌کند که تمام‌کارهایی که می‌توان با یک مرورگر انجام داد را در یک محیط ایمن انجام دهد.
این ابزار حتی فایل‌ها را از طریق وب در یک پوشه مجازی دانلود می‌کند و از طرفی نیز تمام تنظیمات مربوط به مرورگر هر کاربر را در پوشه مربوط به خودش ذخیره می‌کند؛ بنابراین می‌توان گفت این ابزار تا حد بالایی قادر است از حمله بدافزارها به سیستم کاربر از طریق مرور وب جلوگیری کند. در کل با اینکه حجم این نرم‌افزار مقداری بالاست، قادر است حریم خصوصی کاربر را تا مقدار قابل‌توجهی حفظ نموده و کاربر را از تروجان یا حملات مشابه مصون نگه دارد.

۲-۱-۲- سندباکس JOESecurity

یکی دیگر از سندباکس‌های آنلاین که اخیراً مورد توجه کاربران قرارگرفته است، سندباکس JOESecurity می‌باشد.
این سندباکس فایل‌ها و URLها را به‌طور کاملاً خودکار در یک محیط کنترل‌شده اجرا کرده و با مانیتور کردن رفتار برنامه‌ها و سیستم‌عامل، موارد آلوده را تشخیص می‌دهد. این ابزار تمامی فعالیت‌ها را در قالب یک گزارش جامع و کامل گردآوری و ارائه می‌کند.

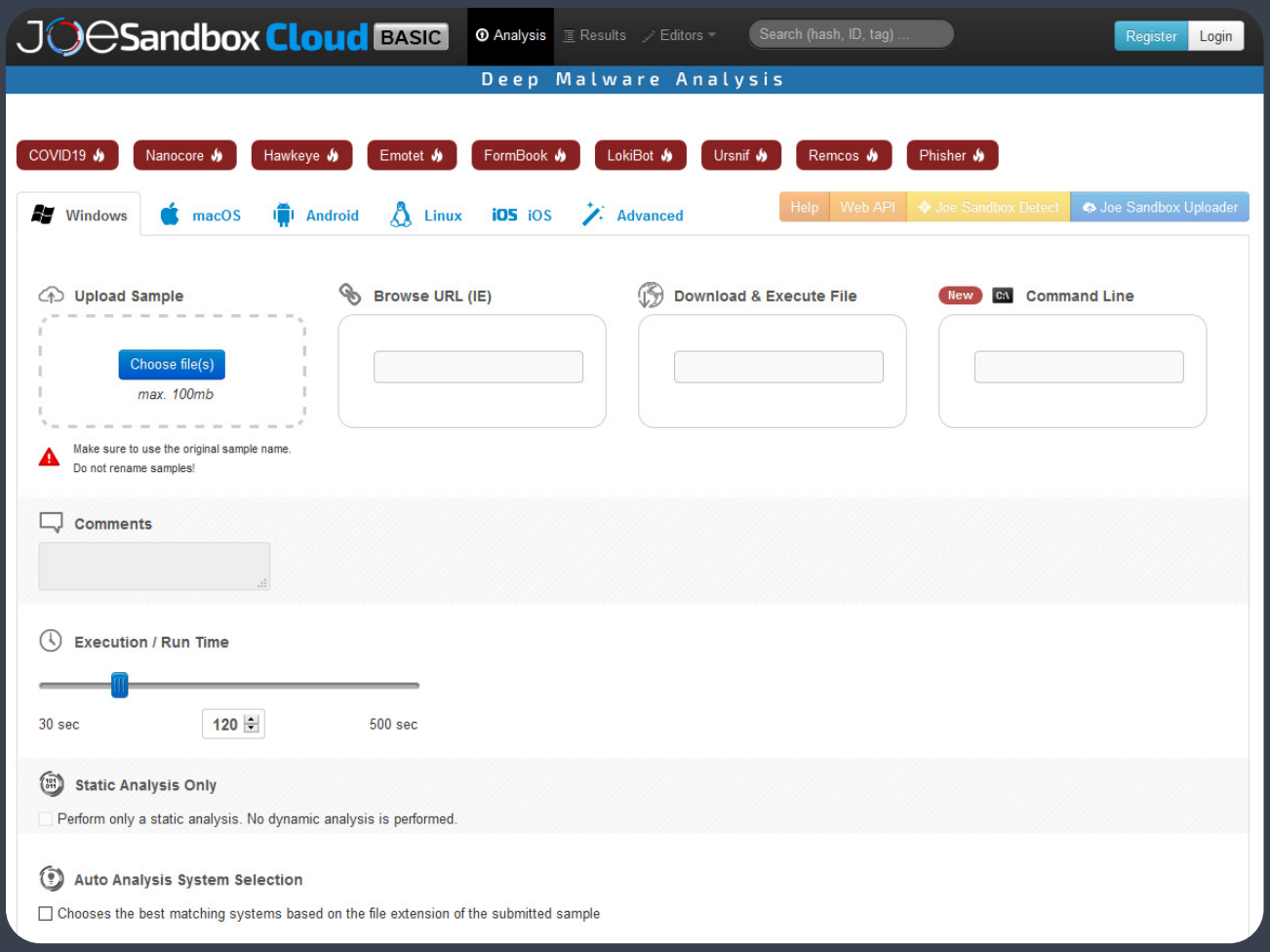
شکل ٤. سندباکس JOESecurity

شرکت JOESecurity انواع مختلفی از سندباکس‌ها را با توجه به نیاز کاربران ارائه کرده است، از انواع این سندباکس‌ها می‌توان به نسخه Desktop، نسخه Linux، نسخه Mail Monitor و غیره اشاره کرد. اخیراً این شرکت ابزار جدیدی را تحت عنوان JOE Snadbox Ultimate رونمایی کرده است که تلفیقی از چند سندباکس قبلی خودش است. شکل ٥ ساختار این سندباکس جدید را نشان می‌دهد. این ابزار شامل حداقل یک واحد کنترلی است که بر روی Linux و ماشین‌های مجازی چندگانه که سیستم‌عامل‌های مختلفی چون Windows و Android را دارا می‌باشند (مانند VMware)، اجرا می‌شود.



شکل ٥. ساختار JOE Sandbox Ultimate

ابزار سندباکس دیگری که همین شرکت تولید کرده است و با نام JOE Sandbox Cloud شناخته می‌شود، قادر است انواع فایل‌ها را از کاربر گرفته و در ابر خود آپلود کند و نهایتاً با استفاده از سندباکسی که در ابر خود ایجاد کرده است، چگونگی رفتار فایل مورد نظر را بررسی کند. شکل ٦ نشان‌دهنده صفحه کاربر در این سندباکس می‌باشد. همان‌طور که در شکل ٦ دیده می‌شود، قابلیت‌های زیادی در اختیار کاربر قرار داده شده است. به‌عنوان‌مثال، ابتدا کاربر می‌تواند سیستم‌عامل مورد نظر خود را برگزیند و سپس انتخاب کند که یک فایل را آپلود، یک آدرس سایت را بررسی، یک فایل را دانلود و اجرا و یا یک دستور برنامه‌نویسی را اجرا کند. در ادامه کاربر می‌تواند تنظیمات مدنظر خود را جهت چگونگی بررسی رفتار فایل مشکوک ایجاد کند. این تنوع، سندباکس JOE Sandbox Cloud را نسبت به سایر سندباکس‌های آنلاین موجود جذاب‌تر نشان می‌دهد.

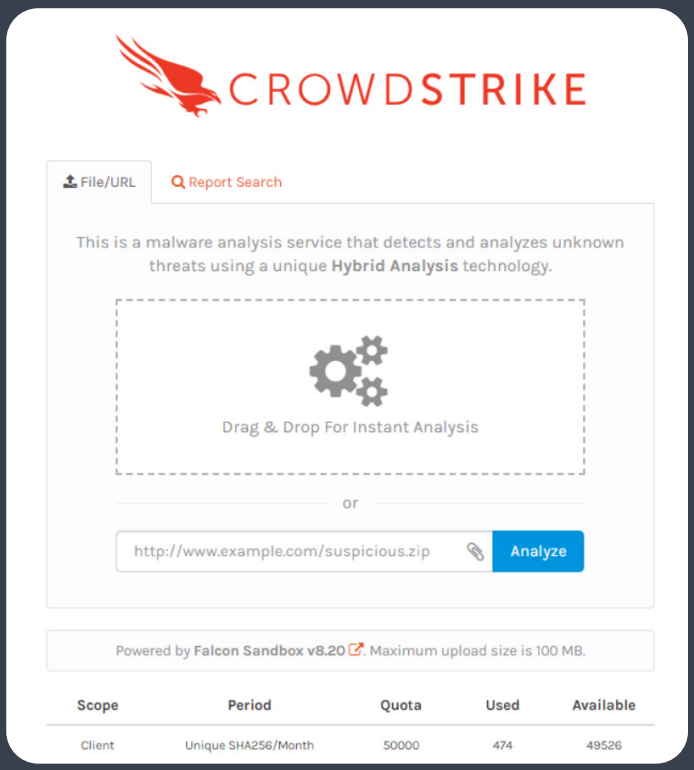


شکل ٦. محیط JOE Sandbox Cloud

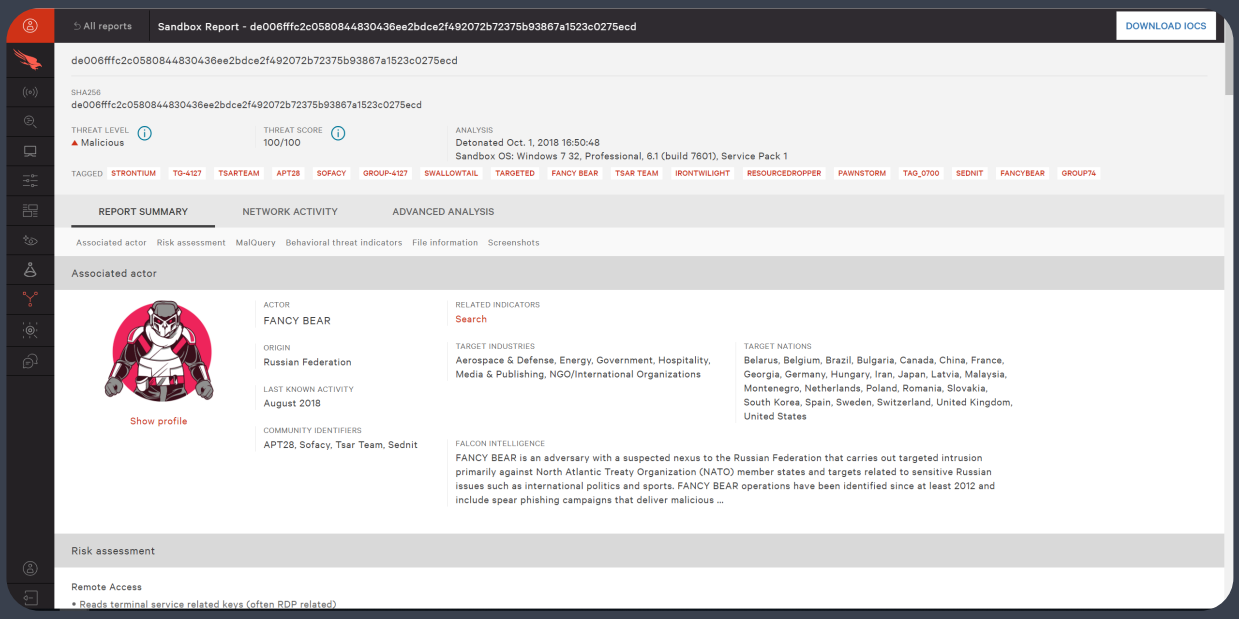
۲-۱-۳- سندباکس Falcon

یکی دیگر از سندباکس‌های آنلاین مبتنی بر ابر، سندباکس Falcon می‌باشد. این سندباکس که توسط شرکت CrowdStrike ساخته شده است، قادر است فایل موجود در یک آدرس سایت را بررسی و تحلیل رفتاری کرده و خرابکار بودن آن را تشخیص دهد. هر کاربر که حساب کاربری ایجاد کند و لایسنس این ابزار را خریداری نماید، این سندباکس یک بخشی از ابر خود را به آن کاربر اختصاص می‌دهد وکاربر می‌تواند فایل موجود بر روی سیستم خود را در آن ابر آپلود کند تا سندباکس به تحلیل رفتاری فایل مورد نظر بپردازد. پس از پردازش و تحلیل فایل، این سندباکس گزارش خود را برای کاربر ارسال می‌کند. یک قابلیت دیگر که این سندباکس در اختیار کاربران قرار می‌دهد این است که گزارش‌های خود را در اختیار تمام کاربران قرار می‌دهد و کاربران می‌توانند قبل از ارسال یک فایل برای بررسی، در میان گزارش‌های پیشین این سندباکس جست‌وجو کنند و اگر کاربر دیگری آن فایل را توسط سندباکس Falcon بررسی کرده باشد، نتایج گزارش را مشاهده کنند و نیاز مجدد به بررسی آن فایل نباشد. شکل ۷ محیط ورودی این سندباکس را نشان می‌دهد.

نمونه‌ای از گزارش خروجی این سندباکس نیز در شکل ۸ نشان داده شده است. این گزارش مواردی همچون مشخص کردن امتیاز برای آلودگی احتمالی برحسب درصد، نام مستعار و محل جغرافیایی سازنده آن، تاریخ آخرین فعالیت مشخص‌شده او و اهداف اصلی بدافزار (به‌عنوان‌مثال سازمان‌های مربوط به هوافضا و صنایع دفاعی و کشورهایی که این بدافزار سابقه فعالیت در آن‌ها را داشته است) را نشان می‌دهد.



شکل ۷. محیط ورودی سندباکس Falcon



شکل ۸. نمونه گزارش سندباکس Falcon

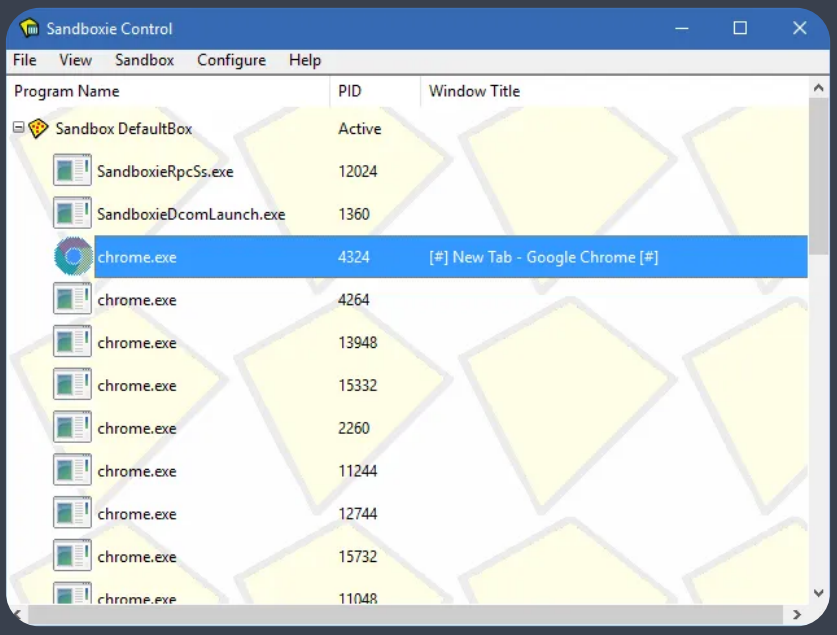
۲-۲- سندباکس‌های آفلاین

۲-۲-۱- سندباکس Sandboxie

می‌توان گفت پرطرفدارترین سندباکس، سندباکس آفلاین Sandboxie است. این سندباکس، دسکتاپ و منوی استارت خاص خود را دارد. کاربر می‌تواند برنامه‌های نصب‌شده بر روی سیستم‌عامل خود و همچنین کد-اسکریپت‌هایی را در این سندباکس اجرا نموده تا در صورت آلوده بودن آن‌ها، سیستم کاربر آسیب نبیند. یکی از مزایای سندباکس فوق این است که کاربر می‌تواند برای نصب نرم‌افزار جدید از طریق این سندباکس اقدام نموده و با خیال راحت نرم‌افزار مشکوک موردنظر خود را نصب نماید. استفاده از سندباکس‌ها طبیعتاً سرعت اجرای برنامه‌ها را نسبت به حالت عادی کاهش می‌دهند. این کاهش سرعت اجرا هرچه بیشتر باشد، سندباکس بهتر می‌تواند برنامه‌ها را تحلیل کند.

Sandboxie جزء سندباکس‌های سبک به شمار می‌آید. سبک بودن بدین معنی است که اجرای برنامه‌ها در آن با تأخیر و کندی کمتری نسبت به سایر سندباکس‌ها صورت می‌گیرد، به این علت که واکنش اصلی این سندباکس در مقابل برنامه‌های اجراشده در آن، صرفاً جلوگیری از دسترسی آن برنامه‌ها به تنظیمات Registry می‌باشد. از دلایل محبوب بودن این سندباکس میان کاربران غیرحرفه‌ای می‌توان به رایگان بودن، نصب و استفاده آسان از آن و همچنین قابلیت اجرای مرورگرها برای وبگردی به‌صورت ایمن اشاره نمود.

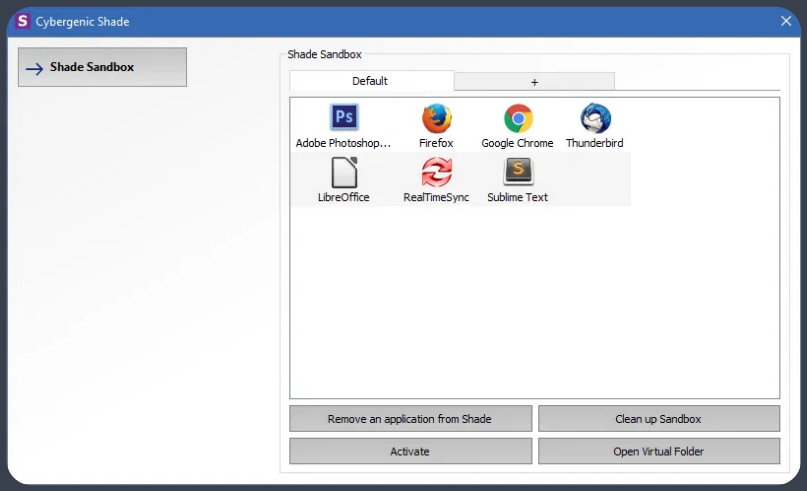
این نکته نیز جالب‌توجه است که نرم‌افزاری که توسط مرورگر اجرا شده در محیط حفاظت‌شده این سندباکس دانلود می‌شود، در همین محیط حفاظت‌شده سندباکس قابلیت نصب و اجرا دارد. شکل ۹ محیط این سندباکس را نشان می‌دهد.



شکل ۹. محیط سندباکس Sandboxie

۲-۲-۲- سندباکس Shade

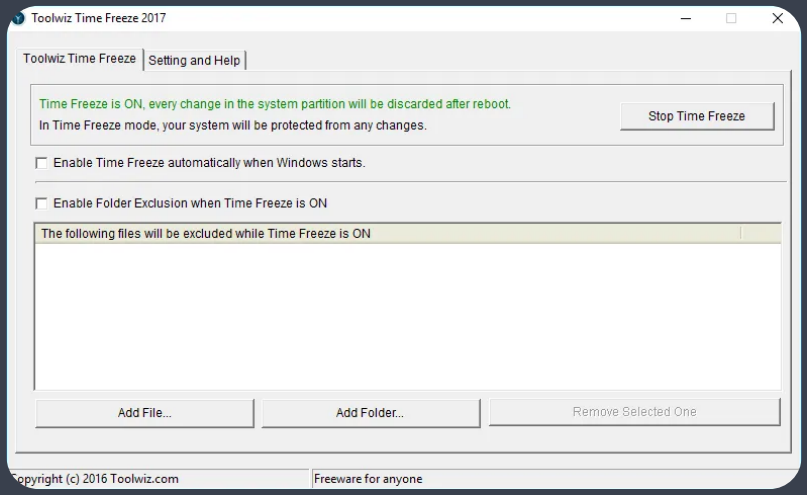
یکی دیگر از سندباکس‌های پرطرفدار و رایگان، سندباکس Shade است. عملکرد این سندباکس تقریباً مشابه سندباکس Sandboxie می‌باشد.
کار با این سندباکس در مقایسه با Sandboxie ساده‌تر و آسان‌تر است و برای کاربران تازه‌کار مناسب‌تر می‌باشد. برای کار با این سندباکس، کاربر کافی است آیکون نرم‌افزار موردنظر خود را جهت بررسی انتخاب کرده و آن را به داخل محیط سندباکس کشیده و رها کند، زین پس کاربر هر بار که بخواهد آن برنامه را اجرا کند، سندباکس Shade به‌طور خودکار آن را درون محوطه حفاظت‌شده سندباکس خود اجرا می‌کند. حین استفاده از این سندباکس تمامی تاریخچه‌ها، cookieها، تنظیمات Registry ویندوز، فایل‌های سیستمی ویندوز و سایر اطلاعات حساس دیگر توسط سندباکس ایزوله می‌شوند.
علاوه بر این، تمامی فایل‌های دانلود شده حین استفاده از سندباکس Shade در «پوشه مجازی دانلودها» ذخیره‌شده و دسترسی به آن‌ها تنها با استفاده از رابط Shade امکان‌پذیر خواهد شد. شکل ۱۰ نمونه‌ای از محیط این سندباکس را نشان می‌دهد.



شکل ۱۰. محیط سندباکس shade

۲-۲-۳- سندباکس Toolwiz Time Freeze

این سندباکس برخلاف سندباکس‌های معرفی‌شده تاکنون برای تشخیص آلوده بودن یک برنامه مشکوک، دسترسی‌های آن برنامه را به تنظیمات سیستم و فایل‌های مهم مسدود نمی‌کند، بلکه یک نمونه کپی مجازی از کل سیستم و تنظیمات آن و فایل‌های مهم دیگر گرفته و آن را در محیطی امن نگه‌داری می‌کند. پس از اینکه کاربر برنامه مشکوک مورد نظر را اجرا کند، در صورت آلوده بودن برنامه و آسیب‌رساندن به سیستم کاربر، کاربر با rebot کردن سیستم خود می‌تواند تنظیمات قبلی سیستم را که توسط این سندباکس کپی گرفته شده بود بازیابی کند و به‌عبارت‌دیگر تنظیمات سیستم به حالت اولیه خود بازمی‌گردند.
از مزایای این سندباکس این است که آزادی اختیار کامل به برنامه مشکوک می‌دهد و آن برنامه متوجه حضور سندباکس نخواهدشد و عملکرد حقیقی خود را به نمایش می‌گذارد. درحالی‌که سایر سندباکس‌های معرفی‌شده با محدود کردن دسترسی‌های برنامه مشکوک، آن برنامه را از حضور خود آگاه می‌ساختند و موجب می‌شدند برنامه عملکرد واقعی خود را نشان ندهد و با انجام رفتارهای عادی سندباکس را فریب دهد.
محیط این سندباکس در شکل ۱۱ نشان داده‌شده است.



شکل ۱۱. محیط سندباکس Toolwiz Time Freeze

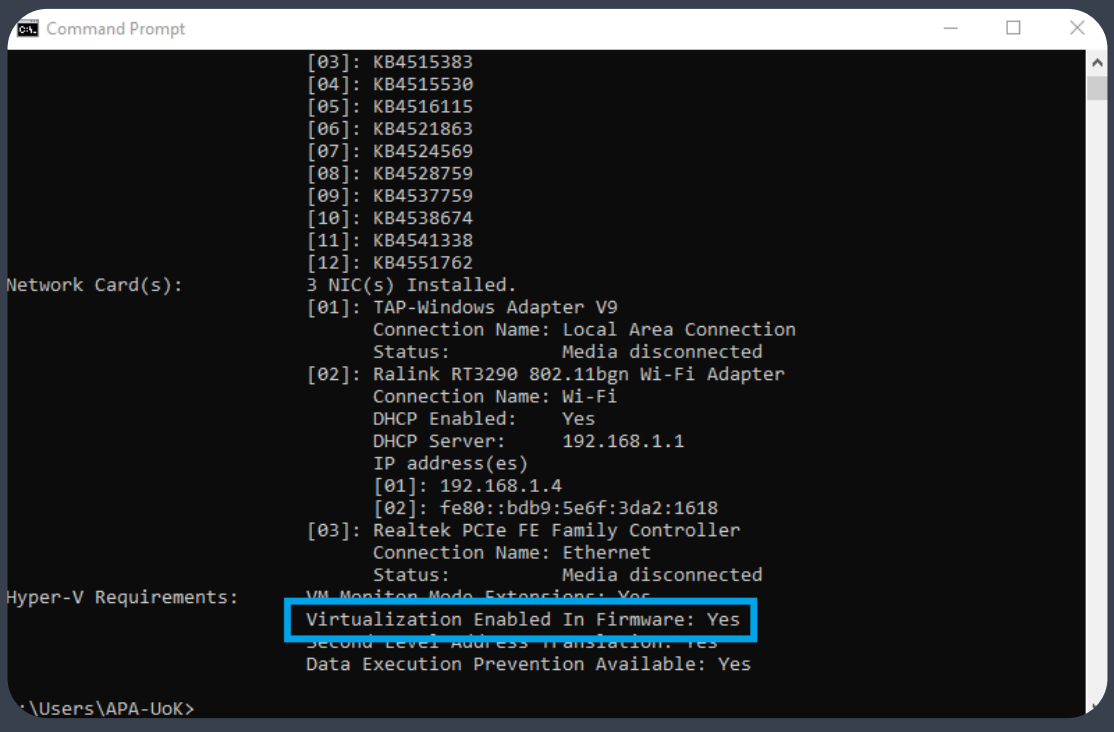
۲-۲-٤- سندباکس Windows 10

شرکت Microsoft در به‌روزرسانی May ۲۰۱۹ ویندوز، یک قابلیت جدید و جذاب به قابلیت‌های ویندوز اضافه نمود. این قابلیت همان سندباکس ویندوز (Windows Sandbox) است. سندباکس ویندوز که یک ماشین مجازی است از Microsoft Hyper-V که همان Image اصلی ویندوز می‌باشد بهره می‌گیرد. حجم این Image حدود ۱۰۰ مگابایت است که این موضوع منجر به این می‌شود تا سندباکس ویندوز با زمان‌بندی صحیح استفاده از هسته CPU، گرافیک مجازی و مدیریت RAM بسیار سبک با سرعت اجرای بالایی باشد. از مزایای این سندباکس می‌توان به عدم نیاز به نصب ماشین مجازی اشاره کرد. چراکه کاربر با اجرای سندباکس ویندوز، به‌طور عملی یک ویندوز مجازی را اجرا کرده است.
در این قسمت سعی می‌شود آموزش مختصری درباره استفاده از این سندباکس ارائه شود. در وهله اول مشخصات سیستم مورد نیاز برای اجرای سندباکس ویندوز به‌صورت زیر است:

- Windows Pro (or Enterprise)
- Hardware virtualization
- AMD64 architecture
- 2processor cores minimum (4 cores with hyperthreading is recommended)
- 4GB of RAM (8GB is recommended)
- 1GB of HDD space (SSD is recommended)

مراحل فعال‌سازی و استفاده از سندباکس ویندوز طبق گام‌های زیر اجرا می‌شوند؛

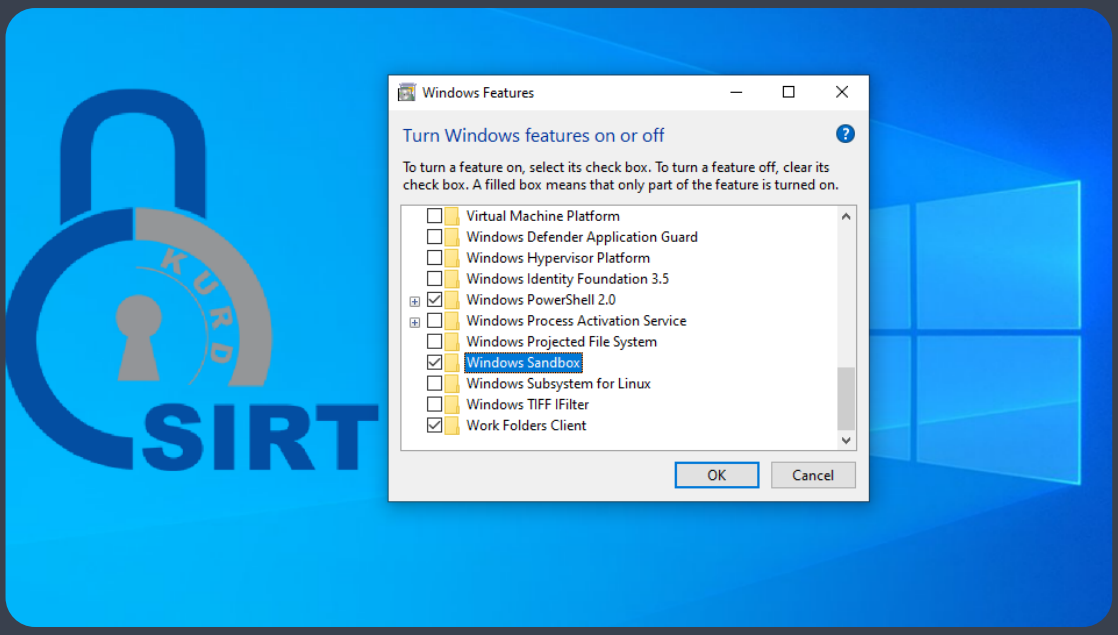
گام اول: ابتدا می‌بایست کاربر از پشتیبانی قابلیت مجازی‌سازی (Virtualization) توسط سیستم خود اطمینان حاصل کند. برای این کار ابتدا Command Window را با نوشتن cmd در قسمت جست‌وجوی ویندوز باز کند، پس از باز شدن Command Window، دستور «systeminfo.exe» را نوشته و اجرا کند که محیطی شبیه به شکل ۱۲ برای کاربر نمایان می‌شود.



شکل ۱۲. وضعیت فعال بودن Virtualization

با توجه به کادر مشخص‌شده با رنگ آبی در شکل ۱۲، کاربر می‌تواند تشخیص دهد که Virtualization در سیستم فعال است یا غیرفعال. در صورت غیرفعال بودن، کاربر می‌تواند با restart کردن سیستم، در قسمت BOOT و تنظیمات UEFI اقدام به فعال‌سازی آن نماید.

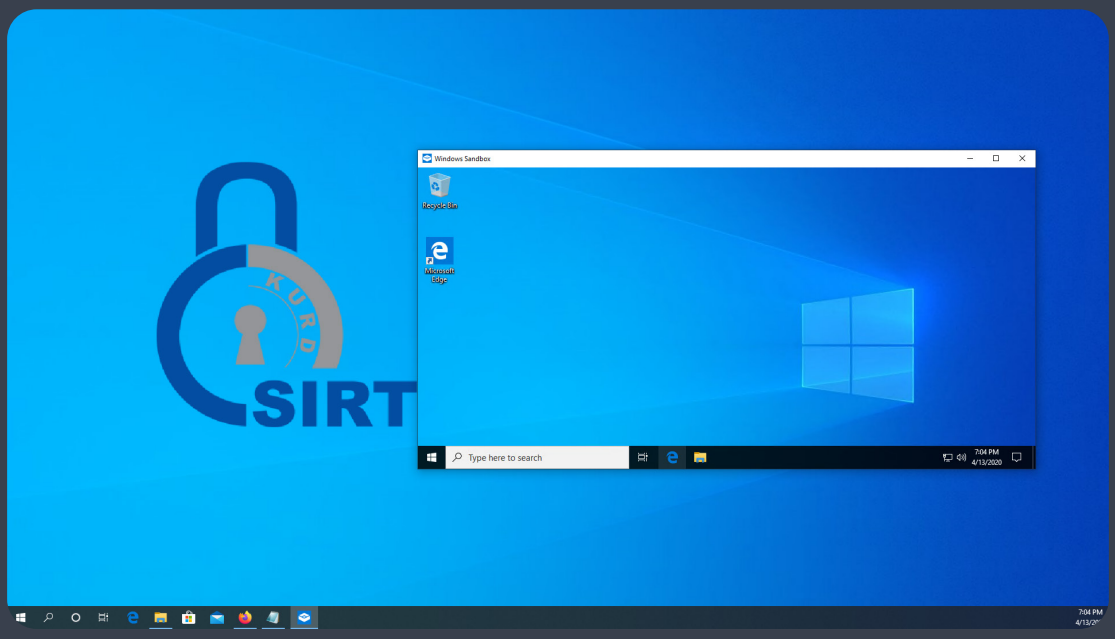
گام دوم: در این مرحله کاربر می‌بایست قابلیت سندباکس را در ویندوز فعال کند. برای این کار، عبارت Windows Features را در قسمت جست‌وجوی ویندوز تایپ نموده و پس از پایان جست‌وجو بر روی نتیجه آن کلیک کرده و پنجره Windows Features را که در شکل ۱۳ نشان داده‌شده است، مشاهده نمایش داده خواهد شد. همان‌طور که در این تصویر مشخص است، کاربر می‌بایست گزینه Windows Sandbox را فعال کند.



شکل ۱۳. Windows Features

پس از فعال کردن قابلیت Windows Sandbox و انتخاب گزینه Ok، سیستم restart شده و بعد از روشن شدن مجدد سیستم، کاربر با صفحه‌ای آبی همانند آنچه که در زمان نصب به‌روزرسانی‌های امنیتی در ویندوز ۱۰ مشاهده می‌شود روبرو می‌شود.
حدود سه دقیقه تا بسته شدن این صفحه و وارد شدن به ویندوز طول می‌کشد.

گام سوم: پس‌ازاینکه ویندوز بالا آمد، کاربر با نوشتن عبارت Windows Sandbox در قسمت جست‌وجو ویندوز می‌تواند سندباکس ویندوز را اجرا کند. شکل ۱۴ این سندباکس را نشان می‌دهد. همان‌طور که در تصویر مشاهده می‌شود، یک ویندوز مجازی برای کاربر ایجاد شده است و کاربر می‌تواند به‌راحتی اقدام به اجرای برنامه‌های مشکوک و یا حتی آلوده نماید.



شکل ۱۴. محیط سندباکس ویندوز

پس‌ازاینکه کاربر موارد مورد نظر خود را توسط این سندباکس انجام داد، می‌تواند سندباکس را ببندد و با بسته‌شدن سندباکس، هیچ یک از فعالیت‌های انجام‌شده توسط سندباکس ذخیره نشده و سیستم حالت عادی خود را دارد. بار بعدی که کاربر تصمیم به استفاده از سندباکس ویندوز بگیرد، مجدداً محیطی کاملاً نو همانند شکل ۱۴ را پیش روی خود خواهد دید.

علاوه بر دو دسته معرفی‌شده از سندباکس‌ها در فوق، دسته دیگری نیز وجود دارد که از آن‌ها تحت عنوان سندباکس‌های مرجع‌باز (open source) یاد می‌شود. سندباکس‌های مرجع‌باز، کد اسکریپت‌هایی هستند که کاربر می‌تواند مطابق میل و سلیقه شخصی خود نحوه عملکرد آن‌ها را تغییر داده و یا قابلیت‌های آن‌ها را گسترش دهد.
از این نوع سندباکس می‌توان به Cuckoo Sandbox، PyREbox و VxStream Sandbox اشاره نمود که اکثراً مبتنی بر زبان برنامه‌نویسی پایتون نوشته و ایجادشده‌اند.

Information Security

امنیت اطلاعات

حوادث امنیت سایبری و اقدامات ضروری برای مقابله با آن‌ها را بشناسیم!

(در صورت هک‌شدن چه اقداماتی باید انجام دهیم؟)

این مقاله در دو بخش به راه‌های مقابله و پاسخ‌گویی به حوادث امنیت سایبری پرداخته است؛ بخش اول در خصوص سازمان‌ها و بخش دوم در خصوص افراد.

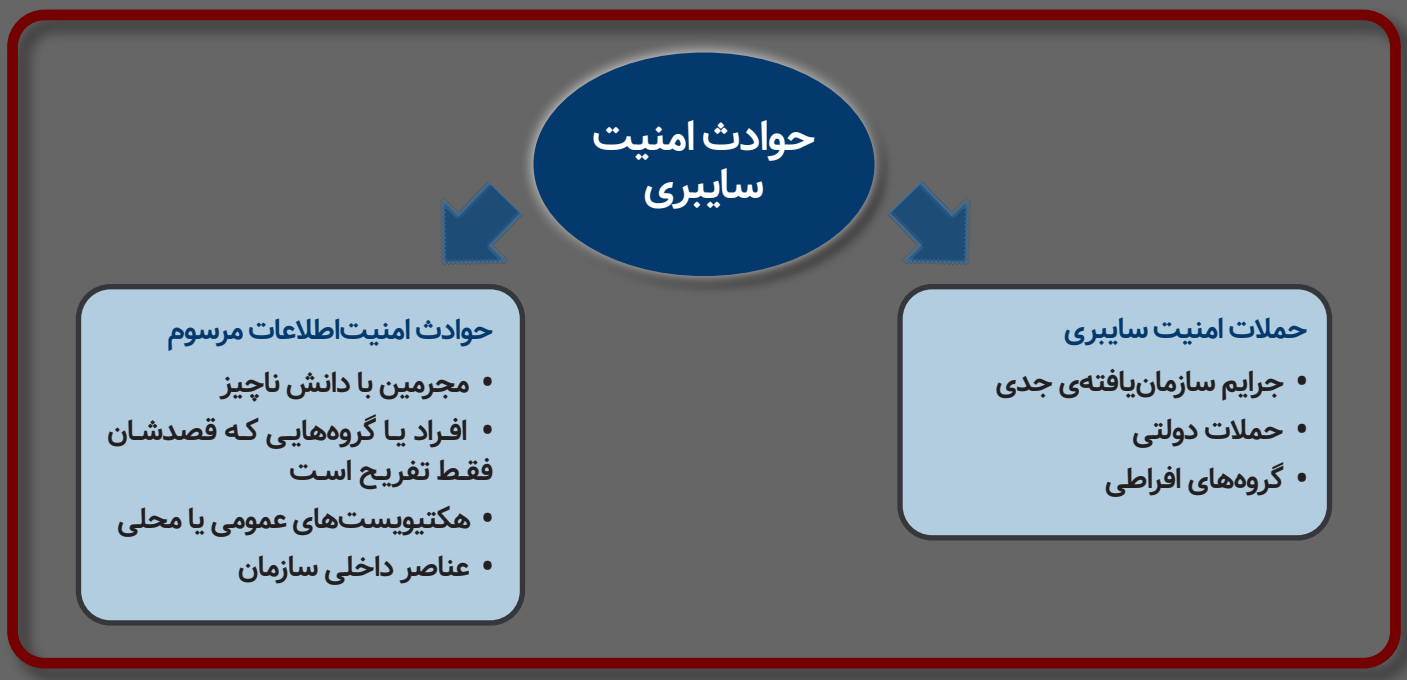
تهیه و تدوین: آزین زارعی

پیش‌درآمد

اصطلاح سایبر از بسیاری جهات قابل تفسیر است، مثلا در فرهنگ لغت cyber اصطلاحی مربوط به رایانه و اینترنت تعریف شده‌است که باز هم ممکن است افراد مختلف، برداشت مختلفی از معنی آن داشته باشند.
علاوه بر این، تحقیقات نشان می‌دهد که سایبر غالباً با مفهوم فضای مجازی همراه است. فضای مجازی یک حوزه‌ی تعاملی متشکل از شبکه‌های دیجیتال است و برای ذخیره‌سازی، تغییر و انتقال اطلاعات استفاده می‌شود، که نه‌تنها اینترنت، بلکه تمام سیستم‌های اطلاعاتی دیگری که از مشاغل، زیرساخت‌ها و خدمات ما پشتیبانی می‌کنند را شامل می‌شود.
فضای مجازی به طور مداوم در حال تکامل و ارائه فرصت‌های جدید است و تمایل افراد و مشاغل برای بهره بردن سریع از فناوری‌های جدید و استفاده از این فرصت‌ها، در کنار راحتی و منافعی که برای آن‌ها فراهم می‌کند خطرات پیش‌بینی نشده و عواقب ناخواسته‌ای را نیز به همراه می‌آورد که می‌تواند تأثیرات منفی به همراه داشته‌باشد. بسیاری از دستگاه‌های محاسباتی (رایانه‌های شخصی، لپ‌تاپ، تبلت و تلفن‌های هوشمند) تقریباً به طور مداوم به اینترنت متصل هستند. سوءاستفاده و بهره‌برداری‌های فنی، نه‌تنها آسیب‌پذیری‌های موجود در زیرساخت‌ها، بلکه در بسیاری موارد برنامه‌های مبتنی بر وب را نیز هدف قرار می‌دهند.
بنابراین می‌توان نتیجه گرفت که امنیت سایبری، امنیت فضای مجازی و یک حادثه‌ی امنیت سایبری، اتفاقی است که در فضای مجازی تأثیر می‌گذارد یا از فضای مجازی به‌عنوان بخشی از بردار (ناحیه) حمله استفاده می‌کند.

تعریف یک حادثه امنیت سایبری

حوادث در فضای تبادل اطلاعات و یا IT انواع مختلفی را شامل می‌شوند، از حملات جدی امنیت سایبری به زیرساخت‌های مهم ملی، جرایم سایبری سازمان‌یافته‌ی بزرگ از طریق هکتیویسم (hacktivism) و حملات اساسی بدافزارها گرفته، تا سوءاستفاده‌ی داخلی از سیستم‌ها و نقص نرم‌افزار، تمامی این‌ها می‌توانند در رده‌ی حوادث امنیت سایبری طبقه‌بندی شوند. با این حال، تحقیقات نشان داده است که هیچ تعریف مشترکی از یک حادثه‌ی امنیت سایبری وجود ندارد.
هیچ طبقه‌بندی معتبری برای تصمیم‌گیری و تشخیص میان حادثه‌ی امنیت سایبری، نقض یا حمله وجود ندارد.
اغلب حوادث امنیت سایبری با حملات مخرب یا تهدیدهای پیشرفته مداوم (APT) همراه است، اما هیچ توافق‌نامه‌ی روشنی در این زمینه موجود نیست. از آنجایی که بسیاری از سازمان‌های مختلف درک متفاوتی از معنای این اصطلاح دارند، رویکردهای پاسخگویی که اتخاذ می‌کنند ممکن است ناسازگار یا نامناسب باشد.



مقایسه‌ی انواع مختلف حوادث امنیت سایبری

به نظر می‌رسد تفاوت اصلی بین انواع مختلف حوادث امنیت سایبری در منبع آن‌ها باشد (یک جنایتکار جزئی در مقایسه با یک سندیکای جرم سازمان‌یافته‌ی بزرگ) و نه نوع حادثه (هک، بدافزار یا مهندسی اجتماعی).

بنابراین ممکن است تعریف‌کردن حوادث امنیت سایبری بر اساس نوع مهاجم، توانایی و قصد آن‌ها مفید باشد.
برخی از متداول‌ترین روش‌های مقایسه‌ی انواع مختلف حوادث امنیت سایبری در جدول زیر آورده شده است، هرچند با

توجه به انواع مختلف و متنوع حملات، مهاجمین و همچنین قربانی‌هایی که مورد هدف قرار می‌گیرند، این مقایسه‌ها در هر حادثه ممکن است متفاوت باشند.

عنوان	حادثه‌ی امنیت سایبری پایه	حادثه‌ی امنیت سایبری پیشرفته
انواع مهاجمین	• مجرمین با دانش ناچیز • افراد یا گروه‌ها به قصد تفریح یا پاسخگویی به یک چالش • هکتیویست‌های عمومی، محلی یا فردی • عناصر داخلی سازمان	• جرایم سازمان‌یافته‌ی جدی • حملات دولتی • گروه‌های افراطی
اهداف مهاجمین	• عمومی • بخش خصوصی • دپارتمان‌های دولتی غیر راهبردی	• سازمان‌های شرکتی بزرگ • سازمان‌های بین‌المللی • دولت‌ها • زیرساختارهای ملی مهم • امنیت ملی/ دفاع
انگیزه و قصد مهاجمین از انجام حملات	• سود مالی • قطع‌ارتباط محدود • تبلیغات • انتقام و تلافی	• پاداش مالی عمده • قطع‌ارتباط گسترده • کشف و افشای اسرار ملی • سرقت معنوی از منابع با اهمیت ملی • ارعابگری و ایجاد ترس • جنگ اطلاعاتی

قابلیت‌های هکرها	• مهارت پایین • منابع محدود • استفاده از ابزار حمله‌ی در دسترس عموم • سازمان‌دهی ضعیف • دسترسی ضعیف	• متخصصین بسیار ماهر • دسترسی قوی به منابع • ابزار حمله‌ی سفارشی • بسیار سازمان‌یافته • حضور بین‌المللی
الزامات پاسخگویی	• خدمات و سرویس‌های بازیابی • نظارت ویژه و سازمان‌دهی • به اشتراک‌گذاری بخشی از اطلاعات صنعت	• رهنمود مناسب و درخور برای متخصص صنعتی و قابلیت‌های ویژه • نتایج و پیامدهای مربوط به سرویس‌های امنیت دولت • پاسخگویی بخش اختصاصی صنعت

۱. سازمان‌ها

آیا سازمان شما برای یک حادثه امنیت سایبری آماده است؟

امروزه با تلاش روزافزون سازمان‌ها در بهره‌مندی هرچه بیشتر از فناوری اطلاعات جهت حفظ برتری خود در رقابت با سایرین، حملات مختلف سایبری نیز شایع‌تر شده‌اند. به‌دلیل عدم آمادگی سازمان‌ها در مقابله با وقفه‌ی ناگهانی در عملیات تجاری و کاری و یا از بین رفتن غیرمنتظره‌ی اطلاعات حساس و اختصاصی، این تهدیدها و حملات، پرهزینه‌تر و دردسر سازتر شده‌اند. بسیاری از سازمان‌ها پس از وقوع حمله‌ی سایبری، تصمیم به یافتن راه‌حل و یادگرفتن چگونگی پاسخ به چنین حادثه‌ای می‌گیرند. بنابراین داشتن یک برنامه‌ی ساخت‌یافته و تست‌شده برای چنین شرایطی باید همیشه بخشی از استراتژی مدیریت ریسک سازمان باشد.

برای پاسخگویی کارآمد به این‌گونه حوادث امنیتی که غالباً رخ می‌دهند سازمان‌ها به طور کلی باید:

- جهت به حداقل رساندن احتمال خطر و کاهش تأثیر حوادث امنیتی سایبری، بر اساس نتایجی که از ارزیابی ریسک‌پذیری و آسیب‌پذیری امنیتی خود به‌دست می‌آورند، اقدامات پیشگیرانه‌ی لازم را انجام دهند.
- با تعریف و دنبال کردن یک برنامه‌ی پاسخگویی که مشخص کند چه مراحل یا اقداماتی برای بهبودی در شرایط حادثه لازم است، قابلیت پاسخگویی و شناسایی حوادث را در سازمان خود توسعه دهند.

به حداقل رساندن شدت و امکان وقوع حوادث امنیت سایبری

«پیشگیری بهتر از درمان است» پس اتخاذ یک عملکرد پیشگیرانه معمولاً کم‌هزینه‌تر و بسیار مؤثرتر از واکنش و پاسخگویی به یک حادثه‌ی امنیتی پس از وقوع آن است. اگرچه جلوگیری از تمام حوادث امنیتی غیرممکن است اما می‌توان برای حصول اطمینان از به حداقل رساندن تأثیرات آن، اقدامات لازم را انجام داد. بنابراین سازمان‌ها باید موارد زیر را در نظر بگیرند:

۱	**ایمن‌سازی شبکه‌ها، سیستم‌ها و برنامه‌ها در حوزه IT** در صورت عدم اطمینان از امنیت سایبری این مؤلفه‌ها، ممکن است حوادث امنیتی بیشتری رخ دهند. سازمان‌ها باید تمام بسترها و دستگاه‌های شبکه و سیستم را بررسی کنند تا مطمئن شوند که جدیدترین وصله‌ها و کنترل‌های امنیتی مربوطه، تماماً نصب شده و الزامات گذرواژه‌ی قوی را نیز برای حساب‌های کاربری اعمال کرده‌اند.
2	**ارزیابی و نظارت بر سیستم‌ها و خدمات در حوزه IT** علاوه بر تلاش برای تأمین زیرساخت‌های فناوری اطلاعات و ارتباطات، سازمان‌ها باید به طور مرتب وضعیت امنیتی خود را نیز ارزیابی کنند. این کار را می‌توان با ارزیابی آسیب‌پذیری امنیتی توسط متخصصان امنیتی، نظارت و تحلیل ترافیک شبکه و عملکرد سیستم، بررسی مداوم کلیه‌ی گزارشات از قبیل گزارشات سیستم، رویداد، برنامه‌ها و گزارشات مربوط به تشخیص نفوذ، انجام داد. نظارت مستمر، به سازمان‌ها آگاهی بیشتری را نسبت به سیستم‌ها و خدمات ICT خود می‌بخشد، همچنین در صورت بروز حوادث امنیت سایبری آمادگی بیشتری خواهند داشت.

۳	**تقویت منابع و قابلیت پاسخگویی** سازمان‌ها می‌توانند با استفاده از منابع کافی، قابلیت‌های پاسخگویی خود در مقابل حوادث امنیتی را بالا ببرند و آن‌ها را به طور مؤثرتر و کارآمدتر اداره کنند. به همین جهت سازمان‌ها باید خط‌مشی‌های امنیتی واضحی را در حوزه‌ی فناوری اطلاعات و ارتباطات فراهم و اجرا کنند، همچنین باید تیمی برای واکنش به حوادث امنیت سایبری نیز تشکیل دهند. بسیاری از حوادث به‌طور قطع به‌دلیل پیروی نکردن برخی کارمندان از رویه‌های مناسب، ناآگاهی آن‌ها و یا بر اثر پیکربندی اشتباه دستگاه‌هایی مانند تجهیزات شبکه و سرویس‌های احرازهویت، ناشی می‌شود. بنابراین سازمان‌ها باید آموزش‌های مناسبی را برای کارمندان متقبل شوند تا مطمئن شوند که آنان ضمن آگاهی از سیاست‌های امنیتی ICT شرکت یا سازمان، از آن‌ها پیروی می‌کنند و در کارشان عملکردی مطابق و سازگار با این سیاست‌ها را پیش می‌گیرند.

اقداماتی جهت تشخیص حوادث و مقابله با آن‌ها

علائم و نشانه‌های حوادث سایبری می‌توانند از چندین منبع مختلف حاصل شوند. تشخیص تغییرات غیرمجاز ایجاد شده در سیستم‌های حساس و مهم یا فایل‌های کاربردی می‌تواند نشانه‌ی بارز وقوع حمله‌ی سایبری باشد. سازمان‌ها می‌توانند گزارشات و رویدادهای تشخیص نفوذ را نظارت و ردیابی کنند، همچنین می‌توانند از انتی‌ویروس‌ها برای تشخیص آلودگی به بدافزارها، از سیستم‌های پیشگیری برای شناسایی الگوهای حمله و از سوابق مربوط به فایروال‌ها برای شناسایی الگوهای غیرمعمول در ترافیک شبکه، استفاده کنند.

سازمان‌های بزرگ همچنین ممکن است تشکیل تیم پاسخگویی حوادث امنیتی رایانه‌ای (CSIRT) را به‌عنوان اصلی‌ترین رابط و نقطه‌ی تماس برای مقابله و سروکار داشتن با حوادث سایبری، در نظر بگیرند. این تیم‌ها به تشخیص و کاهش حوادث امنیت سایبری و بازگرداندن سرویس‌های ICT در آن سازمان می‌پردازند. اعضای CSIRT باید آموزش‌دیده و آماده‌ی مقابله با هر نوع حادثه‌ی امنیت سایبری باشند، همچنین نقش‌ها و مسئولیت‌های آن‌ها نیز باید به وضوح مشخص شود.

داشتن CSIRT مزایای مختلفی دارد، از جمله:

- نظارت فعال بر زیرساخت‌های فناوری اطلاعات و ارتباطات در مقابل نقض امنیت، ارائه و تأمین پشتیبانی هنگام رسیدگی و ممیزی سیستم، انجام ارزیابی و حتی تست‌های نفوذ.
- فعالیت به‌عنوان یک بستر ارتباط مرکزی در راستای دریافت و انتشار اطلاعات حیاتی به اشخاص مناسب مانند دستگاه‌های اجرایی، فروشندگان، مشتری‌ها و سایر تیم‌های پاسخگویی حوادث.
- مستندسازی و ضبط حوادث امنیت سایبری به‌عنوان تجربیات باارزش آموخته‌شده‌ای برای جلوگیری از تکرار مجدد آن‌ها، همچنین تهیه‌ی اسناد برای اعضای جدید تیم.
- بهبود و تقویت وضعیت امنیتی زیرساخت‌های فناوری اطلاعات و ارتباطات، با به‌روز کردن سیستم‌ها و رویه‌ها و تدوین تکنیک‌های جدید برای به حداقل رساندن آسیب‌پذیری‌ها و خطرات.

طرح و برنامه‌ی خود برای مقابله با حوادث را تعریف کنید

سازمان‌هایی که در حال آماده‌سازی برای یک حادثه‌ی امنیت سایبری هستند نیاز به طراحی یک برنامه برای پاسخگویی به حوادث دارند. این طرح باید رویه‌های لازم برای پیگیری را در فازهایی از قبیل موارد زیر پوشش دهد:

۱	**ارزیابی اولیه** تعیین اینکه آیا این حادثه ناشی از یک حمله‌ی سایبری واقعی است یا یک نتیجه FP (False Posetive)، همزمان با تعیین علت حادثه. همچنین جمع‌آوری اطلاعات کافی برای تصمیم‌گیری در مورد نحوه‌ی انجام ارزیابی اولیه، توسط پرسنل/ تیم پاسخگویی.
۲	**پیش‌بینی خسارات و به حداقل رساندن خطر** محدودکردن و جلوگیری از هرگونه آسیب احتمالی بیشتر، هدف اصلی این مرحله است. برای کاهش یک حادثه امنیتی، یک رویکرد سه جانبه لازم است: • مهار کوتاه‌مدت (جدا کردن شبکه‌ی آسیب دیده، خاموش‌کردن سرورها، تغییر مسیر ترافیک و غیره). • پشتیبان‌گیری از سیستم (حفظ‌کردن شواهد برای تجزیه و تحلیل). • مهار طولانی‌مدت (نصب اصلاحات موقتی مانند وصله‌ها و از بین بردن درپشتی‌ها برای ادامه‌ی استفاده از سیستم بدون خاموشی کامل).

۳	**حذف و بازیابی** اقدامات انجام شده در این مرحله برای حذف محتوای مخرب و بازیابی سیستم‌های آسیب‌دیده است. اقداماتی مانند استفاده از نسخه‌های پشتیبان دیسک اصلی برای بازیابی سیستم، نصب وصله‌ها، اسکن نرم‌افزارهای مخرب و غیرفعال‌کردن سرویس‌های بلااستفاده. همچنین آزمایش، نظارت و اعتبارسنجی سیستم بازیابی شده نیز برای جلوگیری از تکرار حادثه‌ی امنیتی ضروری است.
٤	**تجربیات کسب‌شده و درس‌عبرت‌ها** سرانجام، پرسنل/ تیم پاسخگویی باید موارد آموخته‌شده و تجربیات بدست‌آمده از حادثه‌ی امنیتی را به‌عنوان منابعی مستند کرده که در صورت بروز یک حادثه‌ی مشابه به آن‌ها رجوع شود، همچنین آن‌ها باید پیشنهاداتی را در مورد چگونگی بهبود اثربخشی سازمان در مقابله با حوادث احتمالی آینده ارائه دهند.

II. افراد

چقدر احتمال هک‌شدن شما وجود دارد؟

با همه‌گیری هک شدن در اینترنت و حتی دچار شدن سازمان‌ها و شرکت‌های بزرگ، هک‌شدن یک فرد مانند شما نه تنها دور از ذهن و تصور نیست بلکه در واقع یک تجربه‌ی کاملاً متداول است. برنامه‌های ضد بدافزار درمورد تهدیداتی که امروزه شاهد هستیم کمی آرامش‌خاطر به‌وجود می‌آورند، اما در حقیقت اسکنرهای ضد بدافزار در مواردی دارای عملکرد صحیح نیستند، به‌خصوص در مورد حملات و کدهای مخرب جدیدی که زمان زیادی از انتشارشان نگذشته است و اصطلاحاً در بانک‌های اطلاعاتی ضد بدافزارها، الگو یا امضایی برای این نوع حملات و بدافزارها وجود ندارد. هکرها و بدافزارهای مخرب می‌توانند هرموقع که بخواهند تاکتیک‌های خود را تغییردهند و یک بدافزار شناخته‌شده را تنها با جابه‌جا کردن چند بایت به یک بدافزار غیرقابل تشخیص تبدیل می‌کنند. هرچند دانستن معمول بودن این اتفاق لزوماً از ناراحتی که ایجاد می‌کند، نمی‌کاهد و ناآگاهی از اینکه باید چه اقداماتی را بلافاصله بعد از هک‌شدن انجام دهید باعث می‌شود شما احساس سردرگمی کنید، اما قبل از اینکه در مورد این اقدامات بخوانید بهتر است از هک‌شدن خود مطمئن شوید.

چگونه بدانید هک شده‌اید؟

برای اطمینان از هک‌شدن خود، ۱۵ مورد زیر به شما کمک خواهد کرد.

۱) دریافت یک پیام باج افزار

یکی از بدترین حالات، دریافت موارد مشابهی از پیام‌هایی‌ست مبنی بر اینکه تمام اطلاعات شما رمزنگاری شده و برای بازیابی و رمزگشایی آن‌ها باید مبلغی پول بپردازید. حدود ۲۵٪ از قربانیان با اطمینان از اینکه راه‌حل دیگری ندارند، باج را می‌پردازند. در حالی که پشتیبان‌گیری منظم از اطلاعات خود و نگهداری آن‌ها در جایی غیر از سیستم خود، می‌تواند برای نجات از این شرایط و باج‌گیری به شما کمک کند. بنابراین همواره از داشتن نسخه‌های پشتیبان خوب، قابل‌اعتماد، آفلاین و تست‌شده مطمئن شوید.

۲) دریافت یک پیام جعلی انتی‌ویروس

یک پیام پاپ‌آپ آلوده، بر روی کامپیوتر یا موبایل شما که تظاهر می‌کند اسکنر یک انتی‌ویروس است و ادعا می‌کند که تعداد زیادی آلودگی به بدافزار را بر روی دستگاه شما پیدا کرده‌است. با این‌که این روش خیلی رایج نیست ولی هنوز هم اتفاق می‌افتد و باید با آن مقابله کرد. اگر خوش‌شانس باشید، بدون اینکه اتفاقی بیافتد و سیستم‌تان آلوده شود، می‌توانید مرورگر را ببندید و مجددا راه‌اندازی کنید. اما بدترین حالت این است که پیام جعلی دستگاه شما را آلوده کرده باشد که در این صورت باید رایانه‌ی خود را خاموش کنید. اگر نیاز به ذخیره‌ی کاری دارید، قبل از خاموش‌کردن آن را انجام دهید و سپس سیستم خود را به یک نسخه پشتیبان سالم قبل از آن بازگردانید.

۳) داشتن نوار ابزار ناخواسته در مرورگر

ظهور چند نوار ابزار ناخواسته در مرورگر شما درصورتی که متعلق به محصولات قابل‌اعتماد و شناخته‌شده نباشند، یک علامت رایج از مورد بهره‌برداری واقع‌شدن است. بیشتر مرورگرها به شما اجازه‌ی بررسی و مشاهده‌ی نوارابزارهای فعالی که نصب‌کرده‌اید را می‌دهند، هرکدام از آن‌ها که توسط شما نصب‌نشده‌اند را پاک کنید. درصورتی‌که حذف هرکدام از نوارابزارهای جعلی برایتان دشوار یا ناممکن بود، مرورگر خود را به حالت و تنظیمات پیش‌فرض آن بازنشانی کنید.

٤) تغییرمسیر دادن جستجوهای اینترنتی شما

بسیاری از هکرها با هدایت مرورگر کاربران به آدرس‌های متفرقه و ناخواسته، افراد به جایی که قصد ندارند، امرار معاش می‌کنند. آن‌ها با هر کلیک افراد در وب‌سایت‌های موردنظرشان، درآمدزایی می‌کنند و افراد نیز از مخرب بودن این امر بی‌اطلاع هستند.

برای اطمینان از این امر معمولاً می‌توانید یک عبارت بدیهی و واضح مانند «ماهی‌قرمز» یا «توله‌سگ» را جستجو کنید و ببینید آیا نتایج جستجوی شما بازهم مشابه موارد قبلی و البته بی‌ربط به عبارات مورد نظر شماست یا خیر. متأسفانه امروزه بسیاری از جستجوهای تغییرمسیرداده‌شده، از طریق پروکسی‌های اضافه، به‌خوبی از کاربر مخفی شده و نتایج قلابی هرگز برگردانده نمی‌شوند و در نتیجه کاربر هشداری دریافت نمی‌کند.

۵) مشاهده‌ی پنجره‌های مکرر و تصادفی

به‌طور معمول پاپ‌آپ‌ها توسط یکی از سه مکانیزم قبلی ایجاد می‌شوند. ابهام در تشخیص وب‌سایت‌هایی که قابلیت دور زدن مکانیزم‌های ضد پاپ‌آپ در مرورگرها را دارند، این مورد را به یکی از موارد آزاردهنده بدل کرده است.

٦) دوستانتان از شما دعوت‌نامه‌هایی در رسانه‌های اجتماعی دریافت می‌کنند که شما ارسال نکرده‌اید.

به دوستانتان هشدار دهید که درخواست‌ها را قبول نکنند، ظاهراً هک شده‌اید. سپس به سایت رسانه‌ی مجازی مذکور در مورد درخواست‌های جعلی اطلاع دهید. این کار اغلب به سهولت با کلیک بر روی گزینه‌ای مانند گزارش‌دهی (reporting) در صفحات آن‌ها انجام می‌شود. سپس در صورتیکه یک حساب جعلی شبیه به شما ساخته نشده‌باشد و واقعاً هک شده‌باشید، گذرواژه‌ی خود را تغییر دهید. اگر طریقه‌ی عوض‌کردن گذرواژه را نمی‌دانید، می‌توانید به گزینه‌ی «کمک» (help) مراجعه کنید.

۷) گذرواژه‌ی آنلاین شما کار نمی‌کند.

اگر گذرواژه‌ی آنلاین شما باوجود اطمینان از درست واردکردن آن بازهم پذیرفته نمی‌شود، احتمال می‌رود که هک شده‌باشید. می‌توانید بعد از گذشت بازه‌ی زمانی ۱۰ تا ۳۰ دقیقه باز هم تلاش کنید و اگر مشکل فنی سایت نبود و واقعاً مطمئن شدید که گذرواژه‌ی شما کار نمی‌کند، پس یک هکر به حساب شما وارد شده و گذرواژه‌ی شما را تغییر داده است. این حادثه احتمالاً نتیجه‌ی پاسخ دادن شما به یک ایمیل فیشینگ به نظرمعتبر، است.

۸) مشاهده‌ی نصب‌های غیر منتظره‌ی نرم افزارها

نصب ناخواسته و خودسرانه‌ی نرم‌افزارها یکی از علائم بزرگ هک‌شدن است. امروزه اکثر بدافزارها تروجان و کرم هستند که مانند یک برنامه‌ی مجاز و قانونی خود را نصب و تکثیر می‌کنند. بنابراین قبل از قبول توافق‌نامه‌ها در حین نصب هر نرم‌افزار، ابتدا آن‌ها را با دقت بخوانید.

۹) موس شما بین برنامه‌ها حرکت و انتخاب می‌کند.

موس‌ها معمولاً به‌دلیل ایراد سخت‌افزاری حرکت‌های ناخواسته دارند، اما اگر این حرکات ناخواسته، شامل انتخاب‌کردن نیز باشد، شما قطعاً هک شده‌اید. این اتفاق ناشی از تکنیکی غیرمتداول است که هکرها را قادر می‌سازد به یک کامپیوتر راه پیدا کنند. آن‌ها بعد از دسترسی به سیستم تا زمانی که سیستم برای مدتی غیرفعال باشد مانند نیمه‌شب‌ها منتظر می‌مانند و سپس سعی می‌کنند به حساب‌های بانکی شما دسترسی پیدا کنند و به شما دستبرد بزنند.

یک شب قبل از خاموش کردن کامپیوترخود دقایقی آن را روشن رها کنید تا متوجه شوید که هکر به دنبال چیست و چه قصدی دارد و قبل از اینکه اجازه دهید از شما سرقت کند، برای مستند کردن این اعمال چند عکس از آن‌ها بگیرید، سپس کامپیوتر خود را خاموش و از شبکه جدا کنید یا روتر بی‌سیم را غیر فعال کنید و با متخصصین تماس بگیرید.

۱۰) غیر فعال‌شدن ضدبدافزار، Task Manager یا Registry Editor

این مورد نیز یک نشانه‌ی بارز از فعالیت مخرب است، بدین‌صورت که اگر متوجه شدید بدون اینکه بخواهید انتی‌ویروس شما غیر فعال شده‌است و برنامه‌هایTask Manager یا Registry Editor هم باز نمی‌شوند یا به‌محض باز شدن ناپدید می‌شوند، احتمالاً مورد بهره‌برداری قرار گرفته‌اید.

۱۱) کم‌شدن پول از حساب آنلاین شما

مجرمین آنلاین معمولاً در مقیاس کم دزدی نمی‌کنند بلکه ترجیح آن‌ها دزدیدن تقریباً همه‌چیز یا خالی‌کردن کل حسابتان است، بنابراین کل پول شما یا مبلغ قابل‌توجهی از حسابتان برداشت می‌شود. این امر اغلب بوسیله‌ی یک ایمیل فیشینگ از طرف بانک طرف حساب شما انجام می‌شود.

۱۲) توسط شخصی به شما اطلاع داده می‌شود که هک شده‌اید.

در این مورد، قبل از مطمئن‌شدن از صحت این خبر، سراغ نقشه‌های پاسخگویی و یا تیم پشتیبان خود نرفته و اقدامی نیز انجام ندهید.

۱۳) فاش‌شدن اطلاعات محرمانه

هیچ علامتی مانند ظاهرشدن اطلاعات محرمانه‌ی افراد بر روی اینترنت یا دارک‌وب، هک‌شدن کاربر را تأیید نمی‌کند! در این مورد هم باید قبل از هر اقدامی ابتدا از صحت اطلاعات اطمینان حاصل کنید.

۱٤) سواستفاده و نشت اعتبارنامه‌های شما

میلیاردها اعتبارنامه‌ی ورودبه‌سیستم معتبر روی وب یا دارک‌وب قرار دارند که غالباً از طریق فیشینگ، بدافزارها یا نقض پایگاه‌داده‌های وب‌سایت‌ها، در دسترس قرار گرفته‌اند. برعکس انواع دیگر افشای داده در این مورد شخص‌سومی به شما اطلاع نمی‌دهد و شما باید بطور فعال درباره‌ی این تهدیدها آگاه و هوشیار باشد. برای جلوگیری و کنترل پیامدها و خطرات ناشی از این نوع حوادث، تشخیص و فهمیدن هرچه سریع‌تر آن‌ها بهتر/ مهم است.

۱۵) مشاهده‌ی الگوهای عجیب در ترافیک شبکه

بسیاری از مواردی که در معرض هک‌شدن قرار گرفته‌اند، ابتدا با مشاهده‌ی الگوهای عجیب و ناخواسته در ترافیک شبکه (احتمالا ناشی از یک حمله‌ی منع از سرویس) متوجه خطر شده‌اند.

توجه داشته‌باشید که در تمام موارد، اولین توصیه قبل از هر اقدامی، بازنشانی سیستم به یک حالت قبلی امن و شناخته‌شده است.

فارغ از اتفاقی که از سر گذرانده‌اید، ازجمله فیشینگ، هک‌شدن، آلودگی به بدافزار، باج‌افزار یا هر حادثه‌ی امنیتی دیگر، چند قدم ساده وجود دارد که با انجام‌دادن بلافاصله‌ی آن‌ها، می‌توانید به حالت عادی و پاک سیستم برگردید.

اقدامات ضروری که افراد پس از هک‌شدن باید انجام دهند.

سعی کنید انگیزه‌ی هک را بیابید!

قطعاً شما در یک وضعیت استرس‌زا قرارداریدو آخرین کاری که در حال حاضر احساس می‌کنید قادر به انجام آن هستید، فهمیدن این است که چرا احتمالاً هک شده‌اید. با این حال، این اقدام می‌تواند بسیار مفیدتر از آنچه که تصور می‌کنید، باشد. به‌عنوان مثال در حالتی که حساب بانکی شما هک شده باشد، کاملاً واضح است که هکرها به احتمال زیاد به‌دنبال پول شما بوده‌اند.

موارد دیگری نیز وجود دارد، مانند هک‌شدن ایمیل شما که در این صورت احتمالات زیادی وجود خواهد داشت، مثلاً شاید هکرها بخواهند از طریق آدرس ایمیل شما به فرستادن هرزنامه (ایمیل اسپم) بپردازند یا اینکه بخواهند با مخاطبین شما تماس بگیرند و درخواست پول کنند.

از طرف دیگر، شاید قصدشان بازنشانی گذرواژه‌ی حساب‌های کاربری دیگرتان از طریق ایمیل شماست. حتی ممکن است در تلاش باشند تا از طریق ایمیل شما به کسب‌وکار شما راه یابند. در هر صورت صرف‌کردن دقایقی برای فهمیدن دلایل و انگیزه‌ی حمله، بسیار به شما کمک می‌کند تا بتوانید اقدامات مناسب را برای خنثی‌کردن هکرها انجام دهید و از وقوع چنین اتفاقی در آینده نیز جلوگیری کنید. علاوه بر این می‌تواند راهی را برای بهبود و بازیابی سریع شرایط شما، تعیین کند.

تنظیم مجدد گذرواژه

کاری که باید فوراً انجام دهید تغییر رمزعبورتان است. در واقع شما نه تنها در سرویس آسیب‌دیده، بلکه در سایر سرویس‌ها که از همان رمز عبور یا یک رمز عبور مشابه استفاده کرده‌اید باید گذرواژه‌ی خود را تغییر دهید. بطورکلی استفاده‌ی مداوم از رمزهای عبور و تکرار آن‌ها قطعاً ایده خوبی نیست و باید آن‌ها را هر چندوقت یکبار بطور منظم تغییر داد. در نتیجه اولین عکس‌العمل بعد از فهمیدن اینکه حسابتان نقض شده است تغییر فوری رمزهای عبور خود، هم در سرویس آسیب‌دیده و هم در هر جای دیگری که از رمزهای عبور مشابه آن استفاده کرده‌اید، است.

محققین امنیتی بر این عقیده‌اند که استفاده از رمزهای‌عبور مشابه یا تکراری چه بر روی یک سرویس چه بر روی سرویس‌ها و برنامه‌های مختلف که افراد برای کمک به حافظه و فراموش‌نکردن گذرواژه‌های خود، به آن تمایل دارند و بسیار هم معمول است، حساب‌های زیادی را در معرض خطر قرارمی‌دهد و خطرات گسترده‌ای را به‌دنبال دارد.

یک راه‌حل خوب برای این مسئله استفاده از یک برنامه‌ی مدیریت رمزعبور است. برای دسترسی به یک مدیرگذرواژه از یک رمز عبور استفاده می‌کنید و سپس این برنامه برای سایر سرویس‌های شما رمزهای عبور قوی و متفاوت تولید می‌کند. به‌این ترتیب تمام رمزهای‌عبور شما در یک مکان نگهداری می‌شوند و شما فقط باید اطمینان حاصل کنید که رمزعبور برنامه‌ی مدیریت رمزعبور شما شبیه به سایر موارد نباشد.

دستگاه خود را به‌روز و اسکن کنید.

این امکان وجود دارد که رایانه‌ی شما نقطه‌ی اصلی ورود مهاجم باشد. بیشتر اوقات، خود قربانی بدافزار را در رایانه‌ی شخصی خود نصب می‌کند، اولین کاری هم که بعد از مطلع‌شدن از حضور بدافزار باید انجام دهد، خلاص شدن از شر آن است. این کار را با به‌روزکردن سیستم‌عامل خود به جدیدترین نسخه شروع کنید. سپس یک نرم‌افزار انتی‌ویروس کاربردی، قابل‌اعتماد و شناخته‌شده را بارگیری و خریداری کرده و رایانه‌ی خود را برای ازبین‌بردن هرگونه بدافزار موجود بر روی آن، اسکن کنید.

با این حال، به‌یاد داشته‌باشید که هیچ نرم‌افزار انتی‌ویروسی کامل نیست. در واقع میزان موفقیت آن‌ها مابین ۵۰٪ تا ۷۵٪ متغیر است.

حساب خود را پس بگیرید.

خبر خوب این که اکثر پلتفرم‌های اجتماعی آنلاین روش‌های ساده‌ای دارند که به شما این امکان را می‌دهد تا حساب شخصی خود را از شخصی که کنترل آن را به‌دست گرفته‌است، پس بگیرید. این کار طی فرآیندی با احرازهویت شما و حصول اطمینان از تعلق داشتن حساب به شما، انجام می‌شود.

مطمئن شوید که هیچ دربِ‌پشتی وجود ندارد.

هکرهای ماهر فقط به رایانه یا حساب‌های آنلاین شما نفوذ نمی‌کنند، آن‌ها همچنین دربِ‌پشتی‌هایی از خود به‌جا می‌گذارند که حتی وقتی از شرشان خلاص شدید، باز هم بتوانند به سیستم شما راه پیدا کنند. بنابراین باید حتی بعد از پس‌گرفتن حساب خود، مطمئن شوید هیچ دربِ‌پشتی نیز برای دسترسی مجدد مهاجم در سیستم شما باقی نمانده‌باشد. به‌عنوان مثال، در ایمیل باید فیلترها و قوانین را بررسی کنید تا مطمئن شوید که بدون اطلاع شما و از جانب شما ایمیلی به حساب‌های دیگر ارسال نمی‌شود. همچنین باید سوالات امنیتی را بررسی کنید تا ببینید آیا پاسخ‌های آن‌ها تغییر کرده‌است یا خیر.

فعالیت مالی را مرور کنید.

اگر حساب مالی شما هک شده است، باید ریز تمام فعالیت‌های آن حساب مانند آدرس‌ها، روش‌های‌پرداخت، حساب‌های مرتبط و غیره را بررسی کنید، چون حتی ممکن است آن‌ها یک حساب و آدرس را به حساب شما پیوند زده باشند تا حتی پس از بازگرداندن حسابتان، بتوانند برای گرفتن وام یا پرداخت خریدهایشان، از حساب شما استفاده کنند.

به حساب‌های کاربری خود رسیدگی کنید.

بعضی اوقات هکرها یک حساب کاربری را هک می‌کنند تا بتوانند به چیز دیگری دسترسی داشته باشند. همان‌طور که گفته‌شد ممکن است آن‌ها برای بازنشانی گذرواژه‌ها، ایمیل شما را هک کنند یا مثلاً اگر یک حساب ابری از شما را هک کنند، ممکن است به هدف دسترسی به یک فایل یا پوشه‌ی خاص باشد. بنابراین باید همیشه تمام فایل‌ها، پوشه‌ها و حساب‌های متصل به آن را به‌دقت موشکافی و بررسی کنید. تمام گذرواژه‌ها را تغییر دهید، اطلاعات مهم را انتقال دهید و از آن‌ها به خوبی محافظت کنید.

عدم مجوز

در عصر OAuth و SSO (Single sign-on) احتمالاً برای ورود به بسیاری از حساب‌های اجتماعی خود از اطلاعات ثبت‌ورود (login) یکی از حساب‌هایتان استفاده می‌کنید. بطورمثال فیس‌بوک و توییتر به کاربرانشان اجازه می‌دهند از طریق Gmail وارد شوند. اگر یک هکر کنترل حساب شما را به‌دست گیرد و وارد سیستم شود و در حساب شما بماند، حتی تغییر رمز عبور نیز تأثیری روی دسترسی او نخواهدگذاشت.

بهترین کار در این مورد، لغو تمام مجوزهاست، به‌عنوان مثال به تمام پلتفرم‌های اجتماعی که از Gmail به‌عنوان مجوز استفاده می‌کنند رجوع کنید و مجوز Gmail را لغو کنید. سپس بعد از تغییر گذرواژه Gmail می‌توانید مجددا به آن مجوز دهید.

این فرآیند ظاهراً ممکن است پیچیده و طولانی به نظر برسد اما بدون شک بهتر از حضور مخفیانه‌ی یک هکر در یکی از حساب‌های شما و کمین برای یک لحظه‌ی مناسب و ضربه‌زدن به شماست.

امنیت اعتبار

یکی از جرایمی که مهاجمان سایبری دوست دارند مرتکب شوند، سرقت هویت است. بنابراین شما باید برای ارزیابی امنیت اعتبار خود وقت بیشتری بگذارید.

در صورت داشتن حساب کاربری رسمی و اداری ابتدا با کلیه‌ی مراکز مهم اعتبارسنجی تماس بگیرید و هک‌شدن خود را به آن‌ها اطلاع دهید و سپس می‌توانید اعتبار خود را قفل کنید.

به دوستان و آشنایانتان اطلاع‌دهید که هک شده‌اید.

متأسفانه خطرات هک شدن با شما پایان نمی‌یابد و فقط به شما بسنده نمی‌کند. این امکان وجود دارد که هکرها بخواهند از حساب شما استفاده کنند تا مثلاً با جعل هویت شما دوستانتان را برای ارسال پول فریب دهند.

همچنین ممکن است از طریق شما امکان دسترسی به داده‌هایی را داشته باشد که دوستان شما را تحت تأثیر قرار دهد، پس آن‌ها باید از خطر احتمالی اطلاع داشته باشند. با وجود تمام دلایل ذکرشده، مهم‌ترین دلیل برای اطلاع دادن به افراد و آشنایان، افزایش آگاهی است.

پیشگیری در نهایت بهتر از درمان است و با افزایش آگاهی دوستان و عموم مردم درباره‌ی آنچه برایتان اتفاق افتاده‌است، به آن‌ها فرصتی برای استفاده از روش‌های صحیح پیشگیری می‌دهید.

بطور مثال آن‌ها می‌توانند نرم‌افزارهای خود را به‌روز نگه‌داشته، از رمزعبورهای قوی و همچنین روش‌های درست اعمال آن‌ها استفاده کنند و برای امنیت بیشتر، همیشه از داده‌های خود نسخه‌ی پشتیبان تهیه کنند.

سخن آخر

طبیعتاً بعد از هک‌شدن احساس ناامیدی و سردرگمی خواهید داشت، با این حال بهتر است خونسردی خود را حفظ کنید، به اعصاب و احساسات خود مسلط باشید و اقدامات لازم را برای کاهش خسارت‌ها انجام دهید و هکرها را از فضای خصوصی خود بیرون کنید که مطمئناً با پیروی از مراحل بالا می‌توانید در این راستا پیشرفت خوبی داشته‌باشید.

چگونه در مقابل جرائم سایبری از خود محافظت کنیم؟

تهیه و تدوین: آزین زارعی

جرم سایبری چیست؟

جرم سایبری شامل هرگونه جرمی است که عمدتاً از طریق اینترنت صورت بگیرد. مجرمان سایبری از دانش و مهارت‌های خود در حوزه‌ی فناوری اطلاعات برای حمله به شبکه‌ها یا دستگاه‌های رایانه‌ای استفاده می‌کنند. هدف از این حملات، دستیابی به اطلاعات مختلف شغلی، دسترسی به حساب‌های کاربری و سرقت هویت است.
سایر جرائم سایبری بیشتر ماهیت شخصی دارند که مواردی همچون انتقام‌جویی، مزاحمت سایبری، آزار و اذیت، قلدری و سوءاستفاده‌ی جنسی را در بر می‌گیرند. همه‌ی ما حداقل یک‌بار فیلم یا سریالی با این مضمون دیده‌ایم و تا حدودی از قابلیت، دانش و عملکرد مجرمان فضای مجازی آگاهیم.
امروزه ما امور مالی و تجاری خود را بیشتر به‌صورت آنلاین و از طریق کامپیوتر، لپ‌تاپ، تبلت و حتی تلفن‌های خود کنترل می‌کنیم و از سوی دیگر جرائم سایبری گوناگون نیز به همین میزان در حال رشد هستند.
جرائم سایبری می‌توانند زندگی شما را دریک چشم به‌هم‌زدن زیرورو کنند. کافی است هکرها فرصت مناسبی بیابند آنگاه ظرف چند دقیقه موارد زیادی ازجمله اطلاعات ، هویت و امور مالی شما را به سرقت می‌برند.
همه باید ضمن استفاده از اینترنت، برخی اقدامات احتیاطی اساسی را رعایت کنند تا در برابر جرائم سایبری از خود محافظت کنند.

۱) از گذرواژه‌های قوی استفاده کنید.

به رمزهای عبور پیچیده فکر کنید، این‌که می‌خواهید رمزهای عبور خود را به‌راحتی به خاطر بسپارید کاملاً قابل‌درک است، اما این کار باعث می‌شود رایانه، اطلاعات و احتمالاً منابع مالی شما در معرض خطر بیافتند. گذرواژه‌های شما باید منحصربه‌فرد و حداقل دارای ۸ کاراکتر، شامل ترکیبی از کلمات نامربوط به شما، حروف، اعداد و نمادها باشند. پیشنهاد می‌شود به انتهای گذرواژه‌های خود یک نماد دلخواه اضافه کنید. (همان‌طور که در مطالب دیگر نیز اشاره شد، می‌توانید از برنامه‌های مدیریت گذرواژه نیز استفاده کنید.)
همچنین وب‌سایت‌هایی در سطح اینترنت وجود دارند که می‌توانید از طریق آن‌ها میزان امن و قوی بودن گذرواژه‌های خود را بسنجید. **Kaspersky Secure Password Check** با آدرس **https://password.kaspersky.com/** یکی از این وب‌سایت‌ها می‌باشد.
به‌خاطر داشته باشید که از واردکردن دقیق گذرواژه‌ی خود اکیداً بپرهیزید، صرفاً واردکردن یک کلمه‌بندی نزدیک و مشابه به مورد اصلی کفایت می‌کند!

۲) نرم‌افزارهای خود را به‌روز کنید.

احتمال اینکه هکرها با اعمال کدهای مخرب و اکسپلویت‌های شناخته‌شده روی نرم‌افزارها -به‌خصوص نرم‌افزاری به‌روز نشده- به سیستم شما دستیابی پیدا کنند، بسیار بالاست.

۳) به‌طور مرتب از فایل‌های خود نسخه‌ی پشتیبان تهیه کنید.

اگر بدترین اتفاق نیز رخ‌دهد، با داشتن نسخه‌های پشتیبان می‌توانید فایل‌ها و اطلاعات ازدست‌رفته‌ی خود را به‌راحتی بازیابی کنید.

٤) با احتیاط کلیک کنید!

بسیاری از افراد به‌سادگی با کلیک‌کردن روی یک پیوند یا دانلود یک فایل از منابع غیرقابل‌اعتماد، به دام مجرمان سایبری می‌افتند، بنابراین هنگامی‌که با منابع ناشناخته و غیرقابل‌اعتماد سروکار دارید، هرگز هیچ فایلی را دانلود نکرده و روی هیچ ابرپیوندی (hyperlink) کلیک نکنید.
توجه داشته باشید وب‌سایتی که استفاده می‌کنیم باید با « https» شروع شود. حتماً به وجود حرف «s» در انتهای https دقت کنید، زیرا مخفف کلمه‌ی امن (secure) است، بدین معنی که داده‌های ما رمزنگاری‌شده‌اند و ما یک ارتباط امن خواهیم داشت.

۵) مراقب تنظیمات رسانه‌های اجتماعی خود باشید.

غالباً دسترسی به اطلاعات شما برای مجرمان سایبری مهندسی اجتماعی کار سختی نیست، بنابراین نه تنها باید تنظیمات حساب‌های کاربری خود را در بالاترین سطح امنیتی حفظ کنید، همچنین هر چه اطلاعات کمتری به اشتراک بگذارید، به نفع شما، امن‌تر و بهتر است.

٦) شبکه‌ی خانگی خود را به یک رمزنگار قوی مانند VPN، مجهز کنید.

یک VPN تا زمانی که به مقصد موردنظر خود نرسد، تمام ترافیک خارج‌شده از دستگاه‌های شما را رمزنگاری می‌کند، بنابراین هکرها به‌جز ترافیک و اطلاعات رمزنگاری‌شده، چیزی عایدشان نمی‌شود.

۷) از اطلاعات و جزئیات خصوصی اقتصادی و مالی خود محافظت کنید.

افراد گاهی با واردکردن اطلاعات شخصی در فرم‌های آنلاین فریب می‌خورند. این فرم‌ها اغلب معتبر به نظر می‌رسند اما درواقع توسط کلاهبرداران برای سرقت پول یا هویت شما ایجادشده‌اند. سازمان‌های معتبری مانند بانک، هرگز جزئیات امنیتی را از شما نمی‌خواهند و یا از شما درخواست انتقال پول به یک حساب خاص را ندارند.

۸) کودکان خود را درباره‌ی استفاده از اینترنت و خطرات آن آگاه کنید.

به کودکان خود اطمینان دهید که در صورت تجربه‌ی هر نوع آزار و اذیت آنلاین، زورگویی، خشونت یا بددهنی می‌توانند با شما صحبت کنند و قضیه را با شما در میان بگذارند.

۹) رخنه و نقض‌های مهم امنیتی را دنبال کنید.

اگر در سایتی حساب کاربری دارید که تحت تأثیر یک رخنه و نقض امنیتی قرارگرفته است، از آن‌ها مطلع شده و فوراً گذرواژه‌ی خود را تغییر دهید.

۱۰) در خصوص اطلاعات و حساب‌های کاربری خود، راز نگه‌دار باشید!

اجازه ندهید دیگران بدون حضور شما، به حساب‌های کاربری و صفحاتی که با گذرواژه‌ی شما، محافظت‌شده‌اند وارد شوند. در غیر این صورت، رمز عبور خود را تغییر دهید. شما شاید از امنیت رایانه‌ی خود مطمئن باشید، اما اگر رایانه‌ی آنان بی‌خطر نبود چه خواهد شد؟
حتی اگر آن شخص مورد اعتمادترین و باهوش‌ترین فرد موردنظر شما باشد، بازهم ممکن است به‌طور تصادفی باعث قربانی‌شدن شما شود (به‌طورمثال دستگاه آن‌ها آلوده شود و اطلاعات تاریخچه‌های آن‌ها در اختیار مهاجمین سایبری قرار بگیرد).

۱۱) از فناوری‌های قدیمی و منسوخ استفاده کنید.

اگر صفحه گسترده‌ای از گذرواژه‌ها یا فایل‌های دیجیتالی بسیار حساس دارید، آن‌ها را در رایانه‌ای قدیمی که به اینترنت متصل نیست، نگه‌دارید. اگر یک کامپیوتر اضافی ندارید، فایل‌ها را با استفاده از یکی از ابزارهای رمزنگار رایگان، رمزنگاری کنید و یا حتی آن‌ها را چاپ و با خیال راحت نگهداری کنید و نسخه‌های الکترونیکی را از بین ببرید.

۱۲) تعبیه‌ی دو مکان مجزا

شکل دیگری از حفاظت می‌تواند شامل نگه‌داشتن فایل‌ها در دو مکان مختلف باشد. پرونده‌های رمزنگاری‌شده را مثلاً می‌توانید بر روی DVD یا فلش مموری کپی کنید و آن را به یکی از اعضای خانواده یا یک دوست قابل‌اعتماد بسپارید؛ بنابراین درصورتی‌که کامپیوتر شما به یک ویروس آلوده و به‌طور موقت غیرقابل استفاده شد، آن فایل‌ها و اطلاعاتشان هنوز در دسترس شما و قابل بازیابی هستند.

۱۳) از مکان‌های پرخطر دوری کنید.

می‌دانیم که برخی از رفتارها و اقدامات، به نسبت، ما را در معرض خطر بیشتری قرار می‌دهند. این امر درباره‌ی اینترنت نیز صدق می‌کند. رفتن به چت روم‌های هکر، مشاهده‌ی محتوای بزرگسالان یا مراجعه به سایت‌هایی که می‌دانید کلاهبرداری می‌کنند، شما را بیشتر در معرض خطر حملات سایبری قرار می‌دهند.

۱۴) به دام پاپ‌آپ‌ها نیفتید.

استفاده از ایمیل‌های تقلبی و پیام‌های متنی نه‌تنها روش متداولی هستند بلکه روزبه‌روز به‌طور فزاینده‌ای، قانع‌کننده‌تر نیز می‌شوند. اگر یک پنجره‌ی ایمیل یا پاپ‌آپ از شما می‌خواهد نام کاربری یا رمز عبور خود را وارد کنید، هرگز این کار را نکنید. در عوض مرورگر خود را بازکرده و مستقیم به سایت موردنظر مراجعه کنید. اگر هنوز قانع نشده‌اید، می‌توانید با شرکت یا سازمان موردنظر، تماس بگیرید و صحت ماجرا را جویا شوید. فراموش نکنید شرکت‌های معتبر هرگز از طریق ایمیل، اطلاعات ورود به سیستم را از شما درخواست نمی‌کنند.

کلام آخر

اگر در مورد بانکداری آنلاین، کارت‌های اعتباری یا سایر فعالیت‌های مالی خود نگران هستید و این امور را مستعد حملات سایبری می‌دانید، کاملاً حق با شماست!
پس برای جلوگیری از قربانی شدن، روی اقدامات خود و هر آنچه که برای محافظت از خود و رایانه‌ی خود می‌توانید انجام دهید، متمرکز شوید. حساب‌های خود، اعم از کاربری و مالی را مرتباً بررسی کنید تا اطمینان حاصل کنید که هیچ فعالیت مخرب یا کلاهبرداری صورت نگرفته باشد. اگر متوجه هرگونه مورد مشکوکی شدید، سریعاً با موسسه‌ی مالی یا اجتماعی مرتبط تماس بگیرید.

مرکز آپا دانشگاه کردستان

www.cert.uok.ac.ir