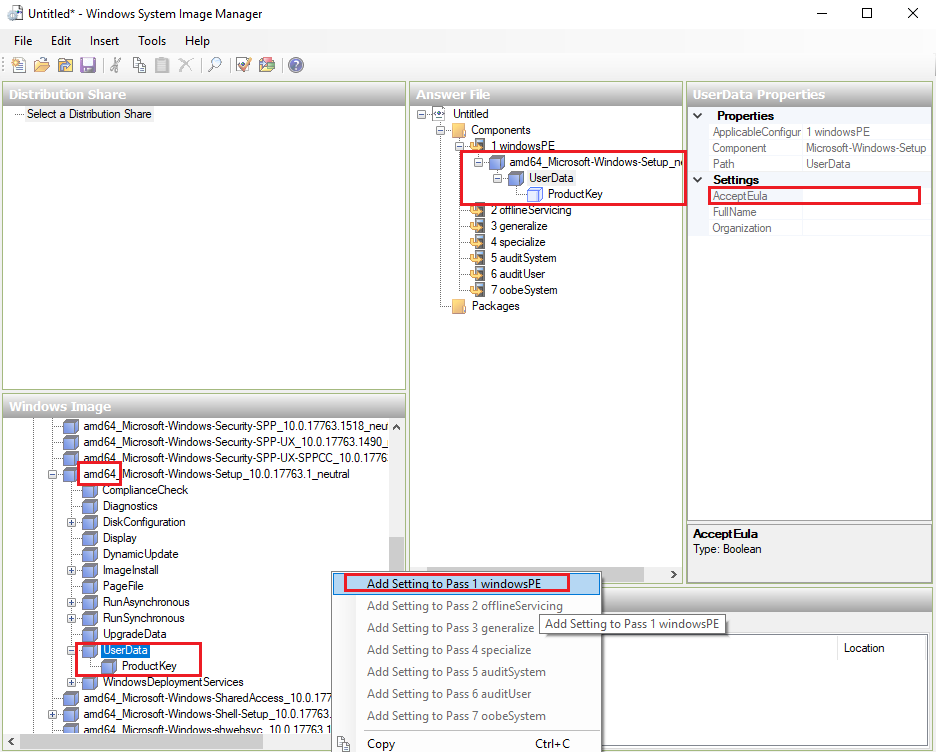
تمرین های مورخ 29/11/1402

**تمرین شماره 1 - یک سرویس مادر معادل آن در ویندوز چیست؟**

**بصورت عادی، شرایط مشابه INIT در لینوکس را می توان به محیط SafeMode در ویندوز مرتبط دانست. با این وجود در سیستم عامل ویندوز هم چنین وضعیت مشابهی را می توانیم در نظر بگیریم.**

****

**همانطور که مشاهده می شود با استفاده از ابزار Windows system Image Manager که برای ساخت AnswerFile استفاده می شود. وضعیت های Pass1 تا Pass7 را می توان مشاهده کرد. که Pass1 نصب بسته اولیه ویندوز بر روی سیستم است و آخرین مرحله Out-of-Box Experience است. هر چند این ابزار صرفا با رویکرد Modify کردن مراحل نصب ویندوز استفاده می شود. اما دید جالبی را در اختیار کاربر قرار می دهد.**

**تمرین شماره 2 - یک ویندوز Server بالا بیاورید و مفهوم ISCSI را در آن تست کنید.**

**بررسی و اقدام شد.**

**تمرین شماره 3 – حملات مرتبط با مجازی سازی را تشریح کنید ؟**

**حمله VM Escape: ماشین مجازی ها معمولاً در محیط های جدا از هم و در محیط میزبان اجرا می شوند. در امنیت رایانه ، فرار ماشین مجازی فرایند شکستن یک ماشین مجازی و تعامل با سیستم عامل میزبان است. هرگونه تلاش VM برای تعامل مستقیم با هایپروایزر از طریق مداخله در یک محیط جدا شده منجر به Escape VM می شود.**

**حمله VM hopping: یک حالت حمله رایج در حملات امنیتی مجازی سازی است. این بدان معنی است که یک مهاجم سعی می‌کند به دستگاه های مجازی دیگر در همان Hypervisor بر اساس یک ماشین مجازی دسترسی پیدا کند و سپس به آن حمله کند.**

**حمله Hyperjacking: در رایانش ابری، معمولاً از محیط های مجازی و ماشین های مجازی (VM) برای جداسازی سرویس ها استفاده می شود. Hyperjacking حمله ای است که هدف آن سوء استفاده از هایپروایزر (برنامه) مسئول ایجاد و اجرای ماشین های مجازی در فضای ابری است. اگر یک مهاجم کنترل هایپروایزر را به دست آورد، می تواند ماشین های مجازی در حال اجرا را تغییر داده و به آنها آسیب برساند. اگر این نوع حمله موفقیت آمیز باشد، می توان فعالیت های مخرب مختلفی از جمله نظارت بر داده های غیرمجاز را انجام داد.**

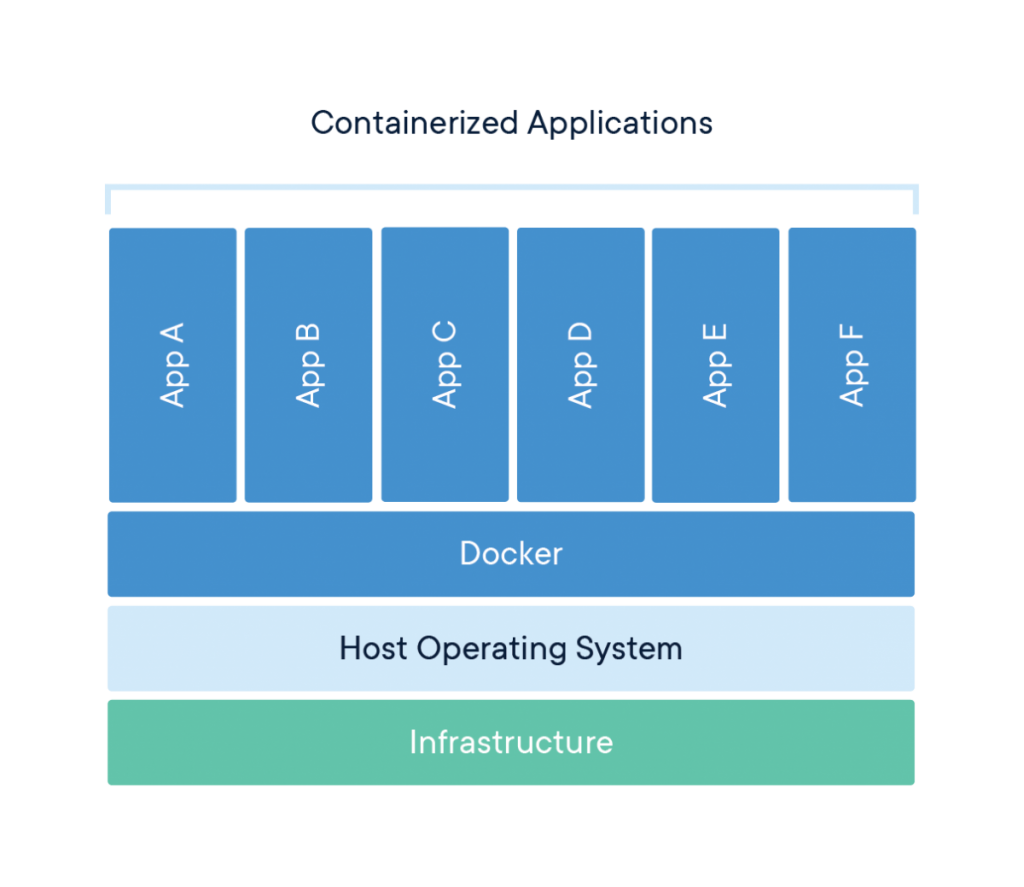
**حملات Hypercall: این نوع حمله شبیه هایپرجک است، با این تفاوت که بخش خاصی از مدیر ماشین مجازی که به عنوان هایپرجک شناخته می شود را هدف قرار می دهد تا به مجوزهای ماشین مجازی دسترسی پیدا کند. هنگامی که مهاجمان دسترسی پیدا کردند، می توانند کدهای مخرب را روی ماشین مجازی اجرا کنند.**

**تمرین شماره 4 - Docker و Countainer چیست و بر روی Ubuntu یک Container بالا بیاورید.**

**کانتینر داکر (Docker Container) از تصویر داکر (Docker Image) ساخته می شود. اگر فرض کنید تصویر داکر یک دستور آشپزی باشد، کانتینر داکر یک ظرف آماده شده از آن دستور آشپزی است.**

**یک تصویر کانتینر (Container Image) نسخه سبک، مستقل و بسته قابل اجرا از بخشی از نرم افزار می باشد که شامل هر آن چیزی است که نیاز به اجرا دارد: کد، ابزار سیستم، کتابخانه سیستم و تنظیمات.**

**کانتینر (Container) نرم افزار را از فضای اطراف آن ایزوله می کند، برای مثال فضای بین توسعه (Development) و راه اندازی نهایی نرم افزار و کمک می کند تداخل بین تیم های مختلف برای اجرای یک نرم افزار برروی یک زیرساخت مشترک کاهش یابد.**

****

**تمرین شماره 5: درباره Open Authentication یا SAML تحقیق کنید.**

**پروتکل OAuth یک استاندارد فنی برای مجوز دادن به کاربران است. در واقع OAuth پروتکلی برای انتقال مجوز از یک سرویس به سرویسی دیگر بدون به اشتراک گذاشتن اعتبار واقعی کاربر مانند نام کاربری و رمز عبور است. با استفاده از پروتکل OAuth یک کاربر می تواند وارد یک پلتفرم شده و سپس مجاز به انجام اقدامات و مشاهده داده ها در پلتفرمی دیگر باشد.**

**OAuth انتقال مجوز از برنامه ای به برنامه دیگر را صرف نظر از اینکه چه برنامه ای هستند ممکن می کند. OAuth یکی از رایج ترین روش هایی است که برای انتقال مجوز از یک سرویس احراز هویت یکپارچه (SSO) به یک سامانه دیگر استفاده می شود. پروتکل های دیگر نیز می توانند این کار را انجام دهند اگرچه پروتکل OAuth یک از پرکاربردترین آنها است.**

**تصور کنید هنگامی که صاحب خانه در خانه نیست بازدید کننده ای به خانه می آید و مالک به جای ارسال کلید برای بازدید کننده کدی موقتی می فرستد تا وارد صندوقی شود که کلید داخل آن است. OAuth نیز به روشی مشابه کار می کند. در OAuth یک سامانه به سامانه دیگر ، به جای ارسال اعتبارنامه کاربر برای اجازه دسترسی به وی، یک توکن دسترسی (authorization token) ارسال می کند.**

**SAML در مقابل OAuth : تفاوت احراز هویت با کنترل دسترسی چیست؟**

**احراز هویت و کنترل دسترسی اگرچه مشابه به نظر می رسند اما در مبحث مدیریت دسترسی کاملا یکسان نیستند و درک تفاوت بین آن ها برای درک نحوه عملکرد فناوری مدیریت دسترسی (از جمله OAuth) بسیار حائز اهمیت است. احراز هویت به هویت کاربر مربوط می شود در حالی که کنترل دسترسی مربوط به امتیازات کاربر است.**

**تصور کنید شخصی در یک تاسیسات امن که یک نگهبان در ورودی دارد کار می کند. تمام وسایل نقلیه ای که قصد ورود به تاسیسات دارند در محل نگهبانی متوقف شده و فقط کارمندان شناخته شده مجاز به ورود هستند. نگهبانی جایی است که احراز هویت افراد انجام می شود. ماموران امنیتی کارت شخص را در برابر لیستی از کارمندان و پلاک وسیله نقلیه وی را در لیست خودروهای مجاز بررسی می کنند.اگر آن ها بتوانند او و وسیله نقلیه اش را احراز هویت کنند می تواند در پارکینگ تاسیسات رفت و آمد یا پارک کند.**

**با این وجود تنها به این دلیل که این شخص می تواند در داخل تاسیسات رانندگی کند به این معنی نیست که می تواند وسیله نقلیه خود را در هر جایی که خواست پارک کند. در مقابل، برای هر کارمند پارکینگی مشخص شده است.شخص فقط می تواند در پارکینگ مخصوص خود پارک کرده و اجازه پارک در جای مدیرعامل را ندارد.**

**OAuth پروتکلی برای کنترل دسترسی است. این همان مسئله ای است که باعث می شود شخص به پارکینگ صحیح برود. در مقابل، SAML یک پروتکل احراز هویت است که به شخص اجازه عبور از نگهبانی را می دهد.**

**ارائه دهنده هویت (IdP) یا سرویس احراز هویت مرکزی (SSO) می تواند از هر دوی این پروتکل ها به طور هم زمان استفاده کند. به علاوه در بعضی از موارد استفاده از OAuth به تنهایی نیز امکان پذیر بوده در حالی که در بسیاری از موارد OAuth راهکاری برای احراز هویت ارائه نمی دهد.**

**به طور خلاصه OAuth و SAML پروتکل های مختلفی هستند و برای اهداف متفاوتی استفاده می شوند، اما هردو اغلب در SSO استفاده می شوند.**

|  |  |  |
| --- | --- | --- |
| **SAML** | **OAuth** | **امکانات** |
| یک پروتکل احراز هویت است. | یک پروتکل کنترل دسترسی است. | پروتکل |
| SAML توسط کسانی مورد استفاده قرار می‌گیرد که به دنبال مدیریت یکپارچه هویت هستند. آن‌ها از این هویت برای ورود کاربران به برنامه های کاربردی مختلف استفاده خواهند کرد. | OAuth توسط کسانی مورد استفاده قرار می‌گیرد که به دنبال حفظ هویت نیستند، بلکه بیشتر سعی دارند از اجرای پروتکل امن توسط برنامه‌های کاربردی مانند گوگل، مایکروسافت و Paypal اطمینان حاصل کنند که هویت‌های آن‌ها معتبر است. | سناریوی استفاده |
| SAML معمولا برای برنامه های مبتنی بر مرورگر مناسب است. | OAuth هم برای مرورگر، هم برای برنامه های موبایل و هم دسکتاپ مناسب است. | برنامه کاربردی هدف |
| SAML به طور گسترده ای برای برنامه های سازمانی استفاده می شود. | OAuth به طور گسترده ای برای برنامه های مشتری و دسترسی به API استفاده می شود. | موارد استفاده |