نویسندگان: دریک راونتری
ایلینا کستریلو

مترجم: طیبه محمدی

مبانی رایانش ابری

درک اصول و مفاهیم رایانش ابری در تئوری و عمل

مبانی رایانش ابری

مولف کتاب:
دریک راونتری-ایلینا کستریلو

مترجم
طیبه محمدی

سرشناسه
عنوان و نام پدیدآور
مشخصات نشر
مشخصات ظاهری
شابک
وضعیت فهرست نویسی
یادداشت
یادداشت
موضوع
موضوع
موضوع
موضوع
شناسه افزوده
رده بندی کنگره
رده بندی دیویی
شماره کتابشناسی ملی

عنوان کتاب مبانی رایانش ابری
مولف کتاب: دریک راونتری-ایلینا کستریلو
مترجم طیبه محمدی
نوبت چاپ
شمارگان ۱۰۰۰ نسخه
ناشر
قیمت

مبانی رایانش ابری

درک مفاهیم بنیادی رایانش ابری در تئوری و عمل

فهرست مطالب

فصل‌های مشارکتی

فصل‌های ۶ و ۷ و همچنین گزیده های کوچک از فصلهای قبلی، در *Securing the Cloud* (ایمن کردن ابر) که توسط Vic Winkler نوشته شده است و *Moving to the Cloud* (حرکت به سوی ابر) نوشته شده توسط Dinkar Sitaram و Geetha Manjunath با مجوز مورد استفاده قرار گرفته‌اند.

مقدمه

از این کتاب چه انتظار می‌رود؟

محیط‌های ابری فراگیر هستند و انتظار می‌رود که حداقل بخشی از چشم انداز فناوری آینده هر سازمان را میزبانی کنند. مبانی رایانش ابری خط مشی است که به شما کمک می کند سوال‌هایی که، در زمان بررسی یا راه اندازی یک طرح اولیه یا محیط ابر هستید بوجود می آیند، هدایت کنید. ابر فقط برای شرکت های بزرگ و کسانی که دارای بودجه های بزرگ هستند در دسترس نیست؛ این تکنولوژی جایگزین با صرفه‌جویی در هزینه، در حال حاضر در دسترس اکثریت است. در برخی موارد، هر سازمان باید تصمیم‌گیری کند که آیا می‌خواهد از مزیت‌های ابر استفاده کند یا خیر. مصرف کنندگان به طور منظم تصمیم می گیرند که آیا عکس ها، موسیقی و فایل های داده خود را در سیستم محلی خود ذخیره کنند یا از ارائه دهندگان ابر استفاده کنند. بنابراین شما چه چیزی را انتخاب می‌کنید؟ پاسخ ساده نمی‌باشد. تمام این‌ها وابسته به نیازهای شما و منابع در دسترس شما است. هدف این کتاب کمک به شما در ایجاد آگاهانه‌ترین تصمیم ممکن در یک زمان محدود است. ما می‌خواهیم شما را مجهز به دانشی کنیم که شما نیاز دارید تا بهترین تصمیم را با توجه شرایط شخصی خود بگیرید، چه شما یک کاربر خانگی و چه مدیر شرکت باشید.

مخاطبان مورد نظر

این راهنمایی برای افرادی است که می خواهند با تکنولوژی رایانش ابری آشنا شوند. چه شما به دنبال بدست آوردن دانش کلی باشید و چه شما نیازمند تصمیم‌گیری برای استفاده از محیط ابری باشید، این کتاب شما را راهنمایی می کند. این کتاب حتی برای افرادی که در حال حاضر تصمیم به استفاده از ابر گرفتند اما نیاز به تصمیم گیری در مورد اینکه کدام ارائه دهنده ی ابر را انتخاب کنند، نیز مفید می‌باشد.

چرا این اطلاعات مفید است؟

تصمیم‌گیری برای حرکت به محیط ابری نباید با بی‌توجهی گرفته شود. به طور کلی برای بسیاری از دپارتمان‌های IT و سازمان‌ها، استفاده از محیط های ابری به معنای تغییر در راه کسب وکار است. شما نمی‌خواهید این تصمیم‌ها را با بی‌توجهی بگیرید. این مهم است که شما خود را به اطلاعات

بیشتری که قبل از تصمیم گیری میتوانید بدست آورید، مجهز کنید. این کتاب به شما در بدست آوردن این اطلاعات مهم کمک می‌کند.

ساختار این کتاب

این کتاب شامل هفت فصل می‌باشد. ابتدا با یک معرفی کلی از ابر و تکنولوژی‌هایی که آن را تشکیل داده‌اند شروع می‌کنیم. سپس درباره‌ی گزینه‌هایی که در هنگام پیاده‌سازی محیط ابری به دنبال آنها هستیم بحث می‌کنیم. و بعد شما را در تصمیم‌گیری راهنمایی می‌کنیم. بعد از تصمیم‌گیری، برخی از ملاحظات را که باید در پیاده‌سازی محیط ابری شما ایجاد شوند را پوشش می‌دهیم.

فصل ۱ معرفی مبانی ابر و مفاهیم مربوط به آن را ارائه می‌دهد و برخی از مزایایی را که از انتساب ابر بوجود می‌آید را شرح می دهیم. ما برخی از مسائل و نگرانی‌هایی را که برخی سازمان‌ها در هنگام حرکت به محیط ابری دارند را پوشش می‌دهیم. و هم‌چنین می‌گوییم که چگونه این مسائل و نگرانی‌ها می‌توانند کاهش یابند.

در فصل ۲ تکنولوژی‌ها و مفاهیمی را که در ایجاد محیط ابری استفاده می‌شوند را بررسی می‌کنیم و موارد احراز هویت، مفاهیم رایانش کلی، مجازی سازی و فن‌آوری های توسعه وب را توصیف می کنیم.

در فصل ۳ مدل‌های استقرار ابری متفاوت را معرفی می‌کنیم. ابرهای عمومی، خصوصی، گروهی و هیبریدی را شرح می دهیم. مزیت‌ها و مشکلات هر مدل را بررسی می‌کنیم. سپس به پیامدهای امنیتی هر مدل نگاه می کنیم. سرانجام، بررسی می کنیم که چه چیزی در حفظ هر محیطی دخیل است. در فصل ۴ مدل‌های متفاوت سرویس ابر را پوشش می‌دهیم، که در ابتدا با ۳ مدل سرویس اصلی آغاز می‌کنیم: نرم‌افزار بعنوان سرویس[۱] (Saas)، پلت‌فرم بعنوان سرویس[۲] (PaaS)، و زیرساخت بعنوان سرویس[۳] (IaaS). سپس مدل‌های سرویس جدیدتر را بررسی می‌کنیم که اخیرا توسعه یافته‌اند.

[۱] Software as a Service
[۲] Platform as a Service
[۳] Infrastructure as a Service

در فصل ۵ درباره‌ی تصمیم‌گیری حول ابر صحبت می‌کنیم. در ابتدا توصیف می‌کنیم که شما برای تصمیم‌گیری برای حرکت به سوی ابر چه چیزهایی را باید در نظر بگیرید. سپس درباره ی انتخاب یک مدل سرویس صحبت می‌کنیم. گام بعدی انتخاب مدل استقرار است. در نهایت درباره‌ی مواردی که در هنگام انتخاب فراهم‌کننده‌ی سرویس‌های ابری عمومی باید در نظر گرفته شود صحبت می‌کنیم.

در فصل ۶ بیشتر درباره‌ی امنیت ابر صحبت می‌کنیم. ما به دنبال یک چارچوب برای ارزیابی هستیم و امنیت بنیادین، ملاحظات کسب‌وکاری، و امنیت عملیاتی را پوشش می‌دهیم. هنگامی که محیط ابری ایجاد می‌شود، شما باید آن را اجرا کنید. در فصل ۷ عملیات یک محیط ابری را پوشش می‌دهیم و هم‌چنین دسترسی به محیط ، روش‌های عملیاتی و فرایندها را توصیف کرده‌ایم. در این فصل هزینه و کارآمدی را نیز پوشش می‌دهیم. ما باور داریم که موارد پوشش داده شده در این فصل‌ها نه تنها شما را در درک ابر، بلکه همچنین به شما در پیاده سازی ابر کمک خواهد کرد. با ابر، مانند بیشتر تکنولوژی‌های جدید، کلید انجام درست آن ایجاد اطمینان در درک درست چیزی است که شما با آن سروکار دارید. شما به این درک به منظور اطمینان از این که ابر برای سازمان شما مناسب است، نیاز دارید. هدف ما ایجاد اطمینان از این است که شما درک لازم را دارید.

فصل ۱. معرفی ابر

نکات این فصل

- ابر چیست؟
- درایورهای ابر (Cloud Drivers)
- موانع پذیرش ابر: چه چیز مانع افراد می‌شود؟

مقدمه

مفهوم رایانش ابری می‌تواند بسیار گیج‌کننده باشد. در این فصل، ما با ارائه‌ی یک دید کلی از ابر و مفاهیم مرتبط به آن آغاز می‌کنیم. سپس درباره‌ی برخی فاکتورهایی که سازمان‌ها را به سوی ابر هدایت می کنند صحبت می‌کنیم.

ابر چیست؟

مباحث زیادی درباره‌ی این که ابر چیست وجود دارد. افراد زیادی ابر را یک مجموعه از تکنولوژی‌ها می‌دانند. درست است که تکنولوژی‌های رایجی وجود دارند که عموما محیط ابر را ایجاد می‌کنند، اما این تکنولوژی‌ها ذات ابر نمی‌باشند. ابر در واقع یک سرویس یا مجموعه ای از سرویس‌ها می‌باشد. این تا حدودی دلیل این است که تعریف ابر سخت شده است. ابر به عنوان مجموعه‌ای مرکب از خدمات، تکنولوژی‌ها و فعالیت‌ها در نظر گرفته می‌شود. برای کاربر سرویس، این که در داخل ابر چه می‌شود شناخته شده نمی‌باشد. به این دلیل است که نامش ابر است. ارائه دهندگان ارائه دهندگان متوجه شده اند که اگر چه بعضی از کاربران در مورد آنچه که در پشت صحنه اتفاق می‌افتد اهمیتی نمی‌دهند، اما بسیاری از آنها به این موضوع توجه دارند. این باعث شده است که ارائه‌دهندگان بیشتر نسبت به آنچه انجام می‌دهند آماده باشند. در بسیاری از موارد، مشتریان مجاز به پیکربندی راه‌حل‌های کنترل سیستمشان نیز می‌باشند.

شکل ۱-۱: معمای ابر

برای بیشتر سرویس‌ها، ابر و سرویسی که ارائه می‌دهد در طول زمان تغییر می‌کند. برای انطباق با نیازهای مشتری برخی سرویس‌ها به سرعت تغییر می‌کنند. فکر کنید که شما چه سرویس‌هایی را تاکنون استفاده کرده‌اید (به ویژه سرویس‌های مرتبط با تکنولوژی) که به مرور زمان تغییر نکرده‌اند. قطعاً تعداد زیادی نیست. اگر شما یک ارائه‌دهنده‌ی سرویس باشید، برای این که برای مشتریان با ارزش باشید باید سرویس‌هایتان را تغییر دهید. قطعاً ابر نیز مستثنی نمی‌باشد. در این‌جا است که باعث گیج‌کنندگی می‌شود. هربار که فردی برای ابر تعریفی ارائه می‌دهد و فکر می‌کند آن مناسب است، مجددا خدمات تغییر می‌کنند. بسیاری فکر می کردند که هنگامی که موسسه ملی استاندارد و فناوری (NIST) یک تعریف رسمی برای محاسبات ابری ارائه داد، این تعریف نهایی خواهد بود. اما، همانطور که مشاهده کردیم، حتی NIST تعریف خود را در طول زمان تغییر داده است. حتی با تغییرات، تعریف NIST همچنان استانداردی است که اکثر مردم در هنگام صحبت کردن در مورد ابر به آن اشاره می کنند. تعریف ابر NIST دارای سه جزء اصلی است که ما در آن بحث خواهیم کرد:

۱. پنج خصوصیت کلیدی ابر
۲. چهار مدل استقرار ابر
۳. سه مدل سرویس ابر

خصوصیات کلیدی ابر

بسیاری از شرکت‌ها و ارائه‌دهندگان سرویس‌ها تلاش می کنند تا از محبوبیت ابر سود ببرند. بسیاری از ارائه دهندگان در تلاش برای پیشنهاد سرویس‌هایی هستند، اگرچه این کار را انجام ندهند. تنها به دلیل این که یک برنامه مبتنی بر وب است، به این معنا نیست که آن برنامه‌ی ابری است. برنامه وسرویس مربوط به برنامه باید خصوصیات مشخصی را نمایش دهند، پیش از این که آنها بتوانند یک پیاده‌سازی ابری صحیح در نظر گرفته شوند. تعریف NIST از محاسبه‌ی ابری بیانگر پنج خصوصیت ابر است: سلف سرویس مورد تقاضا، دسترسی گسترده به شبکه، اشتراک منابع، انعطاف پذیری یا قابلیت ارتجاعی سریع و سرویس اندازه گیری شده است. هر پنج خصوصیت باید وجود داشته باشند به ترتیبی که پیشنهاد، پیشنهاد ابری صحیح در نظر گرفته شوند.

سلف سرویس مورد تقاضا

سلف سرویس مورد تقاضا به معنای این است که مشتری می‌تواند دسترسی به سرویس پیشنهادی را دریافت و یا درخواست دهد، بدون این که مدیر یا نوعی از کارکنان پشتیبانی درخواست را به صورت دستی انجام دهند. فرآیند درخواست و تکمیل به صورت اتوماتیک هستند. این مزیت‌هایی را برای فراهم‌کننده و مشتری سرویس به همراه دارد.

پیاده‌سازی سلف سرویس کاربر به مشتریان امکان تهیه و دسترسی سریع به سرویس‌هایی را که آنها می‌خواهند را می‌دهد. این یک ویژگی جذاب ابر می‌باشد. این باعث می‌شود که شما منابعی را که نیاز دارید را به سرعت و به راحتی دریافت کنید. با محیط‌های سنتی، درخواست‌ها اغلب برای کامل شدن هفته‌ها یا روزها زمان می‌برند، و منجر به تاخیرهایی در پروژه‌ها و ابتکارات می‌شوند. شما نباید در این در محیط ابری نگران باشید. سلف سرویس کاربر نیز بار مسئولیتی را برای ارائه دهنده کاهش می دهد.

سلف سرویس کاربر هم چنین با مسئولیت مدیر را نیز کاهش می‌دهد. مدیران آزاد از فعالیت‌های روز به روز درباره‌ی ایجاد کاربران و مدیریت درخواست‌های کاربر هستند. این به کارکنان IT سازمان‌ها امکان تمرکز روی دیگران را می‌دهد. پیاده‌سازی‌های سلف سرویس می‌تواند برای ایجاد سخت باشند، اما برای ارائه دهندگان ابر قطعاً ارزش زمانی و مالی دارد. سلف سرویس کاربر عموماً از طریق پورتال کاربر پیاده‌سازی می‌شود. پورتال‌های خلاقانه‌ای وجود دارند که می‌توانند برای فراهم کردن عملکرد مورد نیاز مورد استفاده قرار گیرند، اما در برخی نمونه‌ها، یک پورتال سفارشی مورد

نیاز خواهد بود. علی‌الحساب، کاربران با یک واسط الگو[۴] که به آنها امکان قرار دادن اطلاعات مناسب را می‌دهد، نمایش داده می‌شوند. در انتها این پورتال با واسط‌های برنامه‌نویسی برنامه ها، مدیریت (API) که توسط برنامه‌ها و سرویس‌ها انتشار یافته‌اند ارتباط برقرار می‌کند. اگر در داخل سیستم APIها و یا روش‌هایی که برای ساده‌سازی اتوماسیون است وجود نداشته باشد، باعث مشکل می‌شود.

در هنگام پیاده‌سازی سلف سرویس کاربر، شما باید نسبت به مسائل نظارتی و موافقت بالقوه آگاه باشد. اغلب برنامه‌های انطباق[۵] مانند Sarbanes-Oxley (SOX) نیازمند کنترل‌هایی است که برای جلوگیری از این که کاربر قادر به استفاده از سرویس‌های خاص یا اجرای فعالیت‌های خاصی بدون مجوز باشد، است. در نتیجه برخی فرآیندها نمی‌توانند به طور کامل اتوماتیک شوند. این مهم است که شما متوجه شوید که کدام فرآیندها می‌توانند و یا نمی‌توانند در پیاده‌سازی سلف سرویس در محیط شما خودکار شوند.

دسترسی شبکه‌ی گسترده

سرویس‌های ابر باید به راحتی قابل دسترس باشند. کاربران تنها باید ملزم به داشتن اتصالات شبکه‌ای پایه برای اتصال به سرویس‌ها یا برنامه‌ها باشند. در بیشتر موارد، اتصال استفاده شده برخی انواع اتصال اینترنت خواهد بود. اگرچه اتصال‌های اینترنتی در پهنای باند قابل رشد هستند، اما آنها هنوز در مقایسه با اتصال‌های شبکه‌ی محلی نسبتا کندتر می‌باشند. از این رو، ارائه‌دهنده باید کاربران را ملزم به داشتم مقدار زیادی پهنای باند برای استفاده از سرویس کند.

اتصال‌هایی که در پهنای باند محدود می‌باشند منجر به بخش دوم این الزام می‌شوند: سرویس‌های ابری نباید به تین کلاینت[۶] نیاز داشته باشد. ابتدا دانلود یک فت کلاینت[۷] ممکن است زمان زیادی را ببرد، به ویژه در اتصال با پهنای باند کم. دوم این که، اگر برنامه‌ی کلاینت نیازمند ارتباط‌های زیاد

---

[۴] template interface

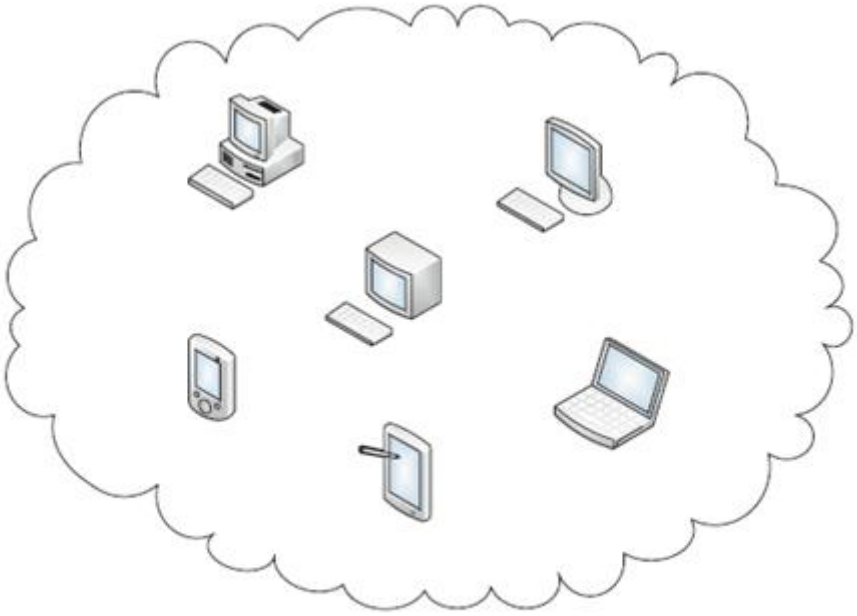
[۵] compliance programs

[۶] thin client: به کامپیوتر یا برنامه کامپیوتری اطلاق می‌شود که برای تحقق وظایف محاسباتی خود به کامپیوترهای دیگر وابسته است

[۷] fat client: در مدل مدل سرویس‌گیرنده-سرویس‌دهنده در شبکه‌های کامپیوتری، به سرویس گیرنده‌ای می‌گویند که امکانات و توانایی‌های زیادی دارد و از سرویس‌دهنده یا سرور مستقل عمل می‌کند

بین سیستم کلاینت و سرویس‌ها باشد، کاربران با مشکلاتی در زمان تاخیر روی اتصال‌های پهنای باند مواجه می‌شوند.

این ما را به بخش سوم این الزام می‌برد: سرویس‌های ابر باید قادر به دسترسی توسط یک تنوع گسترده از دستگاه‌های کلاینت باشند. لپ‌تاپ‌ها و دسکتاپ‌ها تنها دستگاه‌هایی نمی‌باشند که برای اتصال به اینترنت و شبکه استفاده شده‌اند. کاربران هم‌چنین می‌توانند از طریق تبلت‌ها، گوشی‌های هوشمند، و میزبان گزینه‌های دیگر نیز متصل شوند. سرویس‌های ابری نیاز به پشتیبانی تمام دستگاه ها دارند. اگر سرویس نیازمند یک برنامه‌ی کلاینت باشند، ارائه‌دهنده ممکن است مجبور به ایجاد برنامه‌های خاص از نظر پلت‌فرم باشد (یعنی ویندوز، MAC، IOS، و اندروید). اجبار در توسعه و نگهداری تعدادی برنامه‌های کلاینت پر هزینه است، بنابراین اگر راه حل را به گونه‌ای ساختاربندی کرد که به کاربر نیاز نداشته باشد بسیار مفید است.



شکل ۲-۱: دسترسی شبکه ی گسترده

اشتراک منابع

اشتراک منابع در کاهش هزینه‌ها کمک می‌کند و از جانب ارائه‌دهنده انعطاف‌پذیری را بوجود می‌آورد. اشتراک منابع براساس اسن واقعیت است که کلاینت‌ها نیاز ثابت به تمام منابع در دسترس برای آنها ندارند. هنگامی که منابع مورد استفاده قرار نمی‌گیرند، به جای این که بیکار باشند، می‌توانند توسط مشتریان دیگر مورد استفاده قرار بگیرند. این به ارائه‌دهنده امکان می‌دهد تا به بیش از یک مشتری خدمت ارائه دهند (اگر هر مشتری نیاز به منابع اختصاصی داشته باشد). اشتراک منابع با استفاده از مجازی‌سازی بدست می‌آید. مجازی‌سازی امکان افزایش حجم سیستم را به ارائه‌دهنده می‌دهد. در یک محیط مجازی‌سازی شده، منابع روی سیستم فیزیکی در یک مخزن قرار می‌گیرندکه می‌تواند توسط چندین سیستم مجازی مورد استفاده قرار بگیرد.

قابلیت ارتجاعی یا انعطاف پذیری سریع[۸]

قابلیت ارتجاعی یا انعطاف پذیری سریع توصیفگر توانایی یک محیط ابری برای رشد راحت به منظور ارضای درخواست‌های مشتری است. استقرارهای ابری باید برای گسترش ظرفیت خدمات زیر ساخت‌های لازم را داشته باشد. اگر سیستم به خوبی طراحی شده باشد، این ممکن است فقط مستلزم منابع کامپیوتر، هارد دیسک، و مانند آن باشد. کلید آنها این است که حتی اگر منابع موجود و در دسترس هستند، تا زمانی که نیاز نباشند مورد استفاده قرار نگیرند. این به ارائه‌دهنده امکان ذخیره‌ی هزینه‌های مصرف را می‌دهد (برای مثال برق و خنک کردن).

قابلیت ارتجاعی یا انعطاف پذیری سریع اغلب استفاده از خودکارسازی و هماهنگ‌سازی انجام می‌شود. زمانی که کاربرد منابع به نقطه‌ی خاصی می‌رسد، یک راه انداز[۹] فعال می شود. این راه انداز به صورت اتوماتیک شروع به پردازش گسترش ظرفیت می‌کند. وقتی که مصرف فروکش کرد، ظرفیت برای اطمینان ازاین که منابع هدر نمی‌روند ، کاهش می‌یابد. این ویژگی پیاده‌سازی ابری چیزی است که آنها را قادر به مدیریت ظرفیت پشت سر هم مورد نیاز توسط بسیاری از کابران آنها می‌کند. ظرفیت پیوسته یک ظرفیت افزایش یافته است که برای تنها یک زمان کوتاه مورد نیاز است. برای مثال یک سازمان ممکن است نیازمند افزایش ظرفیت پیش پردازش سفارش در انتهای سه ماهه مالی باشد. در یک سیستم سنتی، یک سازمان ممکن است برای پشتیبانی این حجم انتقال اطلاعات نیازمند

[۸] Rapid Elasticity
[۹] trigger

ظرفیت داخلی باشد. این به معنای این است که منابعی وجود دارند که همواره در دسترس هستند اما در یک بازه‌ی زمانی مورد استفاده قرار می‌گیرند. در یک محیط ابری، سازمان می‌تواند از مزیت‌های منابع ابری عمومی برای یک بازه‌ی زمانی کوتاه استفاده کند. نیازی به این نیست که همواره به صورت داخلی ظرفیت در دسترس باشد.

سرویس اندازه‌گیری شده

سرویس‌های ابری باید قادر به اندازه‌گیری مصرف باشند. مصرف می‌تواند با استفاده از معیارهای متفاوتی مانند پهنای باند استفاده شده، زمان استفاده شده، و داده‌های استفاده شده اندازه‌گیری شود. خصوصیت سرویس اندازه‌گیری شده چیزی است که ویژگی "پرداخت بر اساس مصرف[۱۰]" رایانش ابری را بوجود می‌آورد.هنگامی که یک معیارمناسب شناسایی می‌شود، یک نرخ تعیین می‌شود. این نرخ برای تعیین چگونگی پرداخت مشتری استفاده می شود. در این روش، کلاینت صورت‌حسابش را براساس سطح مصرف ایجاد می‌کند. اگر سرویس در یک روز مشخصی استفاده نشود، مشتری برای آن روز هزینه ای متحمل نمی شود. اگر شما هزینه‌های سرویس‌های ابری را می‌پردازید، شما باید اطمینان حاصل کنید که که دقیقا می دانید کدام سرویس‌ها برای شما اندازه‌گیری و منجر به هزینه می شوند. در یک سرویس اندازه‌گیری شده، بسیار مهم است که شما هزینه‌های مرتبط را درک کنید. اگر شما نسبت به هزینه ها درک خوبی ندارید، ممکن است به گونه‌ای ناخوشایند شگفت‌زده شوید.

مدل‌های استقرار ابر

نحوه استفاده از ابرها از سازمانی به سازمان دیگر متفاوت است. هر سازمان الزامات خودش را در این که می خواهد به چه سرویس‌هایی یک ابر دسترسی داشته باشد و این که چقدر می‌خواهد روی محیط کنترل داشته باشد، را دارد. برای تطبیق این الزامات متفاوت، یک محیط ابری می‌تواند با استفاده از مدل‌های مختلف سرویس پیاده‌سازی شود. هر مدل سرویس مجموعه از مقررات و مزایای خودش را دارد. تعریف NIST از رایانش ابری چهار مدل استقرار ابر متفاوت را طرح می‌کند: عمومی، خصوصی، گروهی و هیبریدی. ما در این‌جا خلاصه‌ای از هر کدام را ارائه می‌دهیم اما در فصل‌های بعدی با جزئیات بیشتری به آنها می‌پردازیم.

[۱۰] pay as you go

عمومی

هنگامی که بیشتر افراد درباره‌ی رایانش ابری فکر می‌کنند، آنها به مدل سرویس ابری عمومی فکر می‌کنند. در مدل سرویس عمومی، تمام سیستم‌ها و منابعی که سرویس را فراهم می‌کنند در یک ارائه‌دهنده‌ی سرویس خارجی قرار می‌گیرند. آن ارائه‌دهنده‌ی سرویس مسئول مدیریت سیستم‌هایی است که برای فراهم کردن سرویس استفاده شده‌اند. کلاینت تنها مسئول هر نرم‌افزار یا برنامه‌ی کلاینتی است که روی سیستم کاربر نهایی نصب شده است. اتصال‌ها به ارائه دهندگان ابر عمومی اغلب از طریق اینترنت ایجاد می‌شوند.

خصوصی

در ابر خصوصی، سیستم‌ها و منابعی که سرویس را فراهم می‌کنند در داخل سازمان یا کمپانی که از آنها استفاده می‌کنند قرار دارند. آن سازمان مسئول مدیریت سیستم‌هایی است که برای فراهم کردن سرویس استفاده شده است. بعلاوه، سازمان مسئول برنامه‌ی کلاینت و نرم‌افزارهایی است که روی سیستم کاربر نهایی نصب شده‌اند. ابرهای خصوصی اغلب از طریق LAN محلی و یا WAN در دسترس می‌باشند. در مورد کاربر کنترل از راه‌دور کاربران، این دسترسی از طریق اینترنت یا از طریق استفاده از یک شبکه‌ی خصوصی مجازی (VPN) فراهم می‌شود.

گروهی

ابرهای گروهی ابرهای نسبتا عمومی هستند که بین اعضای یک گروه انتخاب شده از سازمان‌ها به اشتراک گذاشته شده است. این سازمان‌ها عموما هدف و ماموریت یکسانی دارند. این سازمان‌ها نمی‌خواهند که از ابر عمومی استفاده کنند زیرا آن برای عموم آزاد است. آنها حریم خصوصی بیشتری نسبت به چیزی که ابر عمومی پیشنهاد می‌دهند می‌خواهند. بعلاوه هر سازمان نمی‌خواهد به تنهایی مسئول نگهداری ابر باشد؛ آنها می‌خواهند قادر به به اشتراک‌گذاری مسئولیت‌ها با یکدیگر باشند.

هیبریدی

مدل ابر هیبریدی ترکیبی از دو یا چندین مدل ابری است. این ابرها خودشان با هم ترکیب نمی‌شوند، بلکه هر ابر جدا است و آنها همه با هم متصل هستند. یک ابر هیبریدی پچیدگی بیشتری را در محیط ایجاد می‌کند، اما آن امکان انعطاف‌پذیری بیشتری در برآوردن اهداف سازمان ارائه می‌دهد.

مدل‌های سرویس ابری

هنگامی که شما دید عمیق‌تری به چیزی که سرویس‌ها توسط پیاده‌سازی ابری فراهم می‌کنند دارید، شما شروع به صحبت درباره‌ی مدل‌های سرویس ابری می‌کنید. تعریف NIST از رایانش ابری سه مدل سرویس پایه‌ای را طرح می‌کند: زیرساخت بعنوان سرویس، پلت‌فرم بعنوان سرویس، و نرم‌افزار بعنوان سرویس. در این‌جا به صورت خلاصه این مدل‌های را پوشش می‌دهیم و در فصل‌های بعدی به صورت دقیق‌تر درباره‌ی آنها بحث خواهیم کرد.

زیرساخت بعنوان سرویس

زیرساخت بعنوان سرویس یا IaaS سرویس‌های زیرساختی پایه‌ای را برای مشتری فراهم می کند. این سرویس‌ها شامل ماشین‌های فیزیکی، ماشین‌های مجازی، شبکه‌کردن، ذخیره‌سازی، و یا برخی ترکیب‌ها از این موارد است. سپس شما قادر به ایجاد هرآنچه که در بالای زیرساخت مدیریت شده نیاز دارید می‌باشید. پیاده‌سازی‌های IaaS به صورت داخلی جایگزین مراکز داده‌ای مدیریت شده می شوند. آنها به سازمان‌ها امکان انعطاف‌پذیری بیشتر اما در یک هزینه‌ی کاهش یافته را می‌دهند.

پلت‌فرم بعنوان سرویس

پلت‌فرم بعنوان سرویس یا PaaS یک سیستم عامل، پلت‌فرم توسعه، و یا پلت‌فرم پایگاه داده‌ای را فراهم می‌کند. پیاده‌سازی‌های PaaS به سازمان‌ها امکان توسعه‌ی برنامه‌ها بدون نگرانی درباره‌ی ایجاد زیرساخت مورد نیاز برای پشتیبانی محیط توسعه را می‌دهد. با این حال بسته به پیاده‌سازی PaaS که مورد استفاده قرار می دهید، در ابزارهایی که شما می‌توانید برای ایجاد برنامه‌یتان استفاده کنید محدود هستید.

نرم‌افزار بعنوان سرویس

نرم‌افزار بعنوان سرویس یا SaaS برنامه و سرویس‌های داده‌ای را فراهم می کند. برنامه ها، داده‌ها، و تمام پلت‌فرم ها و زیرساخت‌های مورد نیاز توسط ارائه‌دهنده‌ی سرویس فراهم می‌شوند. SaaS مدل سرویس ابری اصلی است و هنوز محبوبترین مدلی است که توسط بیشتر ارائه‌دهندگان توصیه می‌شود.

درایورهای ابر[۱۱]

ابر به افراد فرصت‌های جدیدی را ارائه می‌دهد. قبلا برای از حافظه جمع کردن برنامه‌ها، شما باید هزینه‌ی زیادی را پرداخت می‌کردید تا سیستم درست قرار بگیرد و کارکنان به خوبی آموزش داده شوند. اکنون بسته به این که شما کدام ارائه‌دهنده را انتخاب می‌کنید، این هزینه‌ها به شدت کاهش می‌یابند. ابر یک فاکتور بزرگ در معرفی این عصر مصرفی جدید است. کاربران نهایی نیازی نیست از برنامه‌هایی استفاده کنند که دوست ندارند و نیازهای آنها را پوشش نمی‌دهد. آنها می‌توانند خیلی راحت‌تر برنامه‌ی دیگری را انتخاب کنند که می‌خواند.

امروزه، برخی از برنامه‌های SaaS که بیشترین استفاده را دارند، مدیریت رابطه با مشتری (CRM) و برنامه‌ریزی منبع شرکتی (ERP) است. برنامه‌های CRM و ERP می‌توانند در پیاده سازی و پشتیبانی بسیار پیچیده و دشوار باشند. امروزه با SaaS بسیاری از سازمان‌ها به سوی میزبانی نمونه‌های این برنامه‌ها می‌روند، که با این کار زمان، هزینه و ... زیادی را ذخیره می‌کنند.

درایورهای سیستم

درایورهای سیستمی زیادی وجود دارند که سازمان‌ها را به سوی ابر هدایت می‌کنند. یک سازمان ممکن است خصوصیات سیستمی مشخصی را بخواهند که نمی‌توانند با ساختار فعلیشان آن را فراهم کنند. سازمان‌ها ممکن است که دانش یا بودجه‌ی رسیدن به خصوصیات مشخص محیط را به صورت داخلی نداشته باشند، بنابراین آنها بدنبال یک ارائه‌دهنده‌ی ابر برای ارائه هستند. این خصوصیات شامل چابکی، قابلیت اطمینان، مقیاس پذیری و کارایی می‌باشند.

چابکی

محیط‌های ابری می‌توانند چابکی زیادی را ارائه دهند. شما می‌توانید به راحتی در هنگامی که منابع نیاز هستند آنها را مجددا مناسب کنید. این به شما امکان اضافه کردن منابع به سیستمی که به آنها

[۱۱] CLOUD DRIVERS (درایور: یک سری دستورالعمل که کامپیوتر از آنها پیروی می کند تا اطلاعات را برای انتقال به دستگاه جانبی خاص یا بازیابی از آن دوباره قالب بندی کند)

نیاز دارد را می‌دهد و آنها را دور از سیستمی که آنها را نمی‌خواهد می‌کند. شما به راحتی می‌توانید سیستم‌هایی را برای بسط ظرفیت اضافه کنید.

محیط‌های ابری داخلی به شما امکان استفاده‌ی بهتر از منابع زیرساخت داخلی را می‌دهد. یک زیرساخت ابری که از مجازی‌سازی استفاده می‌کند می‌تواند به شما در افزایش ظرفیتتان و درصد مصرف از زیرساختتان کمک کند. در نتیجه احتمال کمتری از این که سیستم بی‌کار باشد وجود دارد.

قابلیت اطمینان

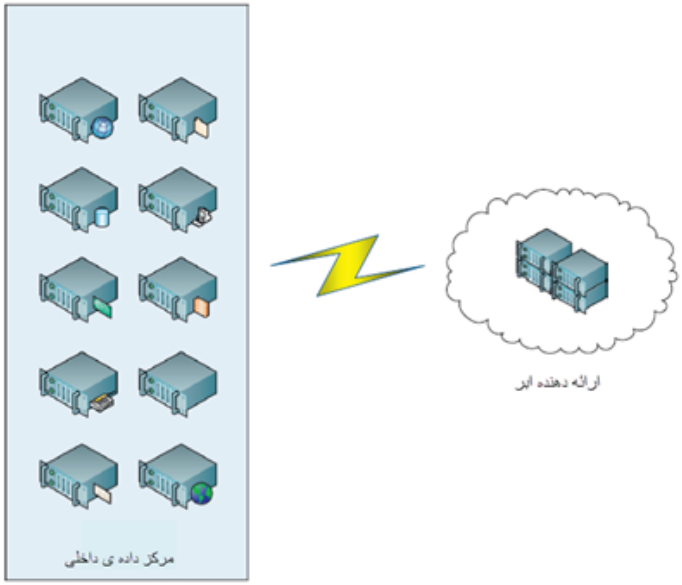
ایجاد اطمینان در محیط می‌تواند بسیار پرهزینه باشد. آن اغلب شامل داشتن چندین سیستم با مکان‌های مرکز داده‌ای است. شما باید بازیابی فاجعه[۱۲] (DR) و چندین برنامه‌ریزی تداوم و شبیه‌سازی را انجام دهید. بسیاری از ارائه‌دهندگان ابری چندین تنظیمات مکان دارد، بنابراین اگر شما از سرویس‌های آنها استفاده کنید، شما فورا می توانید قابلیت اطمینان را به محیط خود اضافه کنید. شما ممکن است درخواست داشته باشید که سرویستان از چندین مکان استفاده کند، اما حداقل آن یک گزینه است.

مقیاس پذیری و انعطاف پذیری

یک محیط ابر می‌تواند به صورت اتوماتیک برای برآورده کردن نیازهای مشتری مقیاس شود. منابع جدید می تواند به صورت اتوماتیک برای رسیدن به مصرف افزایش یافته اضافه شود. این به دو روش کمک می‌کند. ظرفیت افزایش یافته در اطمینان از این که نیازهای مشتری برآورده می‌شوند، کمک می‌کند. این واقعیت که منابع می‌توانند به صورت اتوماتیک برحسب تقاضا تخصیص یابند به این معنا است که آنها همواره در دسترس نیستند، که این به معنای این است که به سیستم‌هایی که بی‌کار هستند و انتظار می‌کشند نیازی ندارید. این سیستم ها هنوز از منابع استفاده می‌کنند. اگر شما به انتظار سیستم نیازی ندارید، شما می‌توانید مصرف منابع مانند برق و خنک کردن را ذخیره کنید. این مقیاس‌پذیری به شما امکان رسیدن به نیازهای مشتری به صورت بهتری را می‌دهد. شما می‌توانید به سرعت ظرفیتی را که مشتری نیاز دارد را برای گسترش موقت یا دائمی اضافه کنید. شما می‌توانید

[۱۲] disaster recovery

از محیط ابری خارجی برای ظرفیت موقت برای فراهم کردن منابع استفاده کنید در حالی که شما می‌توانید ظرفیت دائمی خود را گسترش دهید.



۱-۳: ظرفیت پیوسته

کارآیی

کارآیی در سیستم‌های ابری به صورت مداوم اندازه‌گیری و کنترل می‌شود. اگر کارآیی کمتر از یک سطح مشخصی باشد، سیستم می‌تواند به صورت اتوماتیک به منظور فراهم کردن ظرفیت بیشتر تنظیم شود. حضور توافق سطح سرویس نیز یک مزایا است. یک SLA (تواقنامه سطح سرویس) سطح مشخصی از کارآیی را تضمین می‌کند. اگر این سطح بوجود نیاید، فراهم‌کننده‌ی سرویس عموما برخی سطوح بازدهی را برآورده می‌کند. این بازدهی اغلب به صورت یک بازپرداخت یا کاهش هزینه می باشد. بنابراین اگرچه کارآیی تضمین شده نمی‌باشد، می‌توان یک تضمین را ایجاد کرد که هزینه‌ی کمبود کارآیی می‌تواند کاهش یابد.

راحتی در نگهداری

راحتی در نگهداری می‌تواند در رایانش ابری بسیار سودآور باشد. اگر فرد دیگری زیرساخت و سیستم‌هایی را که برای فراهم کردن سرویس استفاده شده است را مدیریت کند، آنها عموماً مسئول نگهداری هستند. این چندین معنا دارد. شما نباید نگران ردیابی و بروزنگهداری با آخرین سخت‌افزار و نرم‌افزار باشید. شما نباید نگران صرف زمان برای تلاش در مدیریت چندین سرور و بسیاری از سیستم های مشتری متفرقه باشید. شما نباید نگران خرابی ناشی از ویندوزهای تعمیر و نگهداری باشید. چند نمونه وجود دارد که مدیران باید پس از چند ساعت به سیستم وارد شوند تا تغییرات سیستم انجام شود. هم‌چنین نیاز به نگهداری و حمایت از توافق نامه های با چندین فروشنده می تواند بسیار پر هزینه باشد. در محیط ابری، شما تنها باید توافق با ارائه‌دهنده‌ی سرویس را نگهداری کنید.

امنیت و انطباق (پذیرش)[۱۳]

بسیاری از افراد خبره امنیت در یک محیط ابری را در نظر می‌گیرند که این برای این است که آن بسیار امن‌تر از محیط سنتی باشد. مدیران و مهندسان که محیط‌های ابری را اجرا می‌کنند نباید متخصص عمومی باشد، که این در محیط‌های سنتی وجود دارد. آنها می‌توانند روی ایمن کردن یک نوع محیط یا یک نوع داده تمرکز کنند. این تمرکز به مدیران امکان صرف کردن زمان بیشتر روی مقیاس‌های امنیتی بهتر را می‌دهد. بعلاوه یک ارائه‌دهنده‌ی ابری پول بیشتری برای ارائه‌ی حل مساله‌ی خاصی دارد. هم‌چنین آنها مسائل را برای چندین مشتری حل می‌کنند و نه فقط یک سازمان. بسیاری از سازمان‌ها به دنبال ابر برای سهولت در انطباق هستند. محدودیت‌های انطباق می‌تواند یک فشار بزرگ در محیط IT شما قرار دهید. آنها می‌توانند انعطاف‌پذیری و انتخاب‌هایی که شما می‌توانید برای ایمن کردن محیط انجام دهید را محدود کنند. اگر شما قادر به برون‌سپاری توابعی خاصی به یک ارائه‌دهنده‌ی خارجی باشید، شما قادر به کاهش انطباق سازمانتان نیز خواهید بود.

[۱۳] Compliance

درایورهای کسب‌وکار

ابر می‌تواند به شما در بکارگیری و اجرای سریعتر برنامه‌ها کمک کند. همچنین قابلیت مدیریت بهتر و نگهداری کمتر را فراهم می کند و فناوری اطلاعات را قادر می سازد تا منابع را سریعتر تنظیم کند تا تقاضای تجارتی نوظهور و غیر قابل پیش بینی را رفع کند. هنگامی که شما این مزایا را در نظر می‌گیرید، شما می‌توانید کسب‌وکارتان را به یک معماری ساده و چابک تبدیل کنید. مزایای کلیدی دیگری نیز هستند که مربوط به هزینه و مصرف‌گرایی می‌باشند.

هزینه

محیط‌های ابری می‌توانند یک منبع هزینه‌ی کاهش یافته باشند. یکی از بزرگترین ذخیره‌های هزینه انتقال از هزینه سرمایه به هزینه عملیات است. در هنگام تنظیم یک محیط سنتی، زیرساخت و ابزار باید جلوتر از زمان خریداری شوند. این ابزار اغلب بعنوان بخشی از بودجه سرمایه ای سازمان خریداری شود. در محیط ابری شما نباید نگران تهیه‌ی تجهیزات باشید؛ شما تنها برای سرویس هزینه پرداخت می‌کنید. هزینه‌ی سرویس معمولا در برابر بودجه عملیاتی سازمان حساب می شود. عموما دریافت تایید هزینه‌ی عملیاتی راحت‌تر از تایید هزینه‌های سرمایه است. بعلاوه محیط‌های ابری سنتی با استفاده از ذخیره‌ی مزایا و محاسبه‌ی مزایا ایجاد می‌شوند. این ها معمولا ارزان تر از اجزای تخصصی تر هستند.

مصرف‌گرایی

چشم‌انداز فناوری اطلاعات توسط نشانه‌ی مصرف‌گرایی تغییر کرده است. مصرف‌گرایی تمرکز روی نیازها و خواسته‌های مشتری است. مصرف‌کنندگان محدود به پارادایم خاصی نیستند؛ آنها در انتخاب روش‌های دسترسی و برنامه‌هایی که می‌خواهند آزاد هستند.برای رسیدن به این نیازهای مصرف‌کننده، محیط‌های فناوری اطلاعات باید منعطف باشند. آنها ممکن است نیازمند فراهم کردن یک میزبان از برنامه‌های مختلفی که یک عملکرد را فراهم می‌کنند، باشند. اجبار در پشتیبانی این برنامه‌ها می‌تواند سخت و هزینه‌بر باشد. استفاده از یک محیط ابری برای فراهم کردن این سرویس‌ها می‌تواند آن را بسیار ساده‌تر کند. بیشتر محیط‌های ابری دسترسی را از دستگاه‌های مختلف مانند کامپیوتر، لپ‌تاپ‌ها، تبلت‌ها و گوشی‌های همراه فراهم می‌کنند. آنها به کاربران انعطاف‌پذیری در دسترسی به سرویس به هر روشی که انها می‌خواهند را می‌دهد.

فراگیر شدن تکنولوژی

پیشرفت‌های اخیر در تکنولوژی دلیل بزرگی برای ابر به حرکت در آمدن هستند. در گذشته، ابر ایده‌ی خوبی بود اما یک امید واهی بود. تکنولوژی برای به واقعیت درآوردن رویا نبود. مدل ابر فاقد اجزای کلیدی بود تا یک گزینه قابل قبول باشد. بدست آوردن سرورهای کافی برای خدمت به مصرف‌کنندگان بسیار گران بود. و این که شما باید سرورهای جدا رای هر مشتری فراهم می‌کردید. برنامه‌ها یکپارچه بودند و نمی‌توانستند محدوده‌های را اندازه‌گیری کنند. بسیاری از برنامه‌ها نیازمند یک مقدار بیشتری از داده‌ها برای انتقال بین برنامه‌ها و کلاینت هستند. ارائه‌دهنده باید تمام خدمات را برآورده کند. اکنون تکنولوژی توسعه یافته است و برای بسیاری از این کمبودها راه‌حل ایجاد شده است.

مجازی‌سازی

مجازی‌سازی محرک بزرگی در حرکت به سوی ابر است. در واقع، هنگامی که افراد زیادی درباره‌ی ابر فکر می‌کنند، آنها فکر می‌کنند مجازی‌سازی یک الزام در محیط ابری است؛ اما این گونه نیست. مجازی‌سازی می‌تواند نقش گسترده‌ای را در یک پیاده‌سازی ابری داشته باشد، اما الزامی نمی‌باشد. با مجازی‌سازی، شما قادر به میزبانی چندین سیستم مجازی روی یک سیستم فیزیکی می‌باشید. این هزینه‌های پیاده‌سازی را به شدت کاهش می‌دهد. نیازی نیست که شما سیستم‌های فیزیکی جدا برای هر مشتری داشته باشید. بعلاوه، مجازی‌سازی به شما امکان جمع‌آوری منبع و مصرف افزایش یافته از سیستم فیزیکی را می‌دهد.

ساختار برنامه‌ها

تغییراتی در طراحی و ساختار برنامه‌ها نیز وجود دارد. پیش از این، یک برنامه نمی‌توانست به چندین کلاینت سرویس دهد. راهی برای جلوگیری از دسترسی یک مشتری به اطلاعات مشتریان دیگر یا بخشی از برنامه وجود نداشت. اکنون چندین مشتری می‌توانند به یک نمونه‌ی خاص از برنامه دسترسی داشته باشند، اما تعاملات آنها تقسیم‌شده است. برنامه‌ها هم‌چنین شروع به پیاده‌سازی ساختارهای سرویس‌گرا کرده‌اند. SOA به برنامه‌ها امکان تقسیم به عناصر را می‌دهد. این عناصر به صورت جداگانه قابل دسترسی هستند. SOA به برنامه‌ها امکان به اشتراک‌گذاری عناصر را می‌دهد. SOA ، APIهایی را نمایش می‌دهد که می‌توانند توسط سیستم‌های کلاینت یا برنامه‌های کلاینت مورد استفاده قرار بگیرند. رایانش منبع باز به ارائه‌دهندگان امکان سفارشی‌سازی برنامه‌های

پیاده‌سازی مانند تکنولوژی‌های مرتب‌سازی[۱۴] و هایپروایزر[۱۵] را برای برآورده کردن نیازهایشان را می‌دهد. شما با یک تنظیم برنامه‌ی پایه شروع می‌کنید، اما شما می‌توانید برنامه را به منظور متناسب کردنش با نیازهای سازمانتان سفارشی‌سازی کنید.

در رابطه با توسعه‌های وب نیز استانداردسازی بهبود یافته‌ای نیز وجود دارد. این استانداردسازی منجر به افزایش سازگاری و قابلیت همکاری شده است.

افزایش پهنای باند

سرعت دسترسی به اینترنت (پهنای باند) به شدت افزایش یافته است. این باعث افزایش سرعت کلی دسترسی به برنامه‌ها شده است. در بسیاری از موارد، دسترسی مبتنی بر اینترنت می‌تواند قابل قیاس با دسترسی مبتنی بر LAN محلی باشد. پهنای باند افزایش یافته می‌تواند به معنای زمان پاسخ بهتر باشد. این به ایجاد رشد در قابلیت استفاده‌ی برنامه‌های مبتنی بر وب کمک می‌کند.

محرک‌های ارائه‌دهندگان ابر

در سال‌های اخیر، تعداد سرویس‌های ابری و ارائه‌دهندگان ابر افزایش یافته است. محرک‌های برای مصرف‌کنندگان و ارائه‌دهندگان وجود دارد. و به این دلیل است که هر روزه ارائه‌دهندگان جدید بوجود آمده‌اند. آنها مزایایی را که می‌تواند با ارائه‌ی سرویس‌های ابری بدست آید را در نظر می‌گیرند.

اقتصاد مقیاس

ارائه‌دهندگان ابری از یک مفهوم به نام اقتصاد مقیاس استفاده می‌کنند، که مبتنی بر این واقعیت است که وقتی شما زیرساختی را برای برنامه یا سرویس ایجاد می کنید، اضافه کردن ظرفیت تنها نیازمند اضافه‌های افزایشی است. یعنی هرچقدر محیط بزرگتر باشد، بازده سرمایه‌گذاری پتانسیل بیشتری دارد. برای مثال نگاهی به سرویس‌های ایمیل داریم. پیاده‌سازی سرویس‌های ایمیل برای ۵۰۰۰ کارمند به طور داخلی حدود ۲۵ سنت برای هر جعبه‌ی ایمیل هزینه دارد. یک ارائه‌دهنده‌ی

[۱۴] orchestration
[۱۵] hypervisors

ابری که سرویس‌های ایمیل را برای ۱۰۰۰۰۰ کاربر پیاده‌سازی می‌کند برای هر جعبه‌ی ایمیل ۱۰ سنت هزینه دارد. ارائه‌دهنده می‌تواند هزینه‌ی ۱۵ سنت را برای هر جعبه‌ی ایمیل پیشنهاد دهد. این شرایطی است که همه برنده هستند. ارائه‌دهنده درآمد دارد و هزینه نیز ارزان‌تر از چیزی است که می‌تواند برای یک سازمان باشد.

درآمد متناوب

پیشنهاد خدمات مبتنی بر اشتراک می‌تواند سرویسی را برای ارائه‌دهنده‌ی با یک جریان درآمد متناوب فراهم کند. درآمد متناوب ثبات را به یک کسب و کار اضافه می کند. یک جریان درآمد قابل تخمین کمک به برآورد درآمد و بودجه بندی است.

موانع انتخاب ابر: چه چیز مانع افراد می‌شود؟

ابر مزایایی دارد، اما همه‌چیز عالی نمی‌باشد. مشکلاتی هستند که انتخاب ابر را کند می‌کند. در این بخش برخی از این مشکلات را بیان می‌کنیم.

ابهام

یکی از مشکلات که مانع افراد در انتخاب ابر می‌شود عدم درک مفهوم ابر و سرویس‌هایی است که ارائه می‌دهد. این عدم درک موجب ترس می‌شود. اغلب این ترس حول هزینه‌های پنهان، عدم کنترل، مسائل یکپارچه‌سازی، نگرانی‌های امنیتی و ... است. با این حال تمام این مشکلات قابل کاهش است اگر شما ردک درستی از چیزی که در ابر به دنبال آن هستید و از ارائه‌دهنده چه چیزی را انتظار دارید، داشته باشید. ما به شما دانش مورد نیاز برای غلبه بر این ترس را می‌دهیم. بسیاری از نگرانی‌ها تنها پرسش‌هایی هستند که پاسخ قطعی ندارند. وقتی شما با توانایی‌های سازمانتان برای انجام کارهای کسب‌وکاریش سروکار دارید، شما باید نسبت به ناشناخته‌ها محتاط باشید. شما نباید خطرهایی را انجام دهید که نمی‌توانید *آنها* را کاهش دهید. اگر شما نمی‌دانید خطرها چه هستند، پس قطعا شما نمی‌توانید آنها را کاهش دهید.

نگرانی‌های در رابطه با کمال و رشد[۱۶]

اغلب نگرانی‌هایی در رابطه با کمال ابر و ارائه‌دهندگان متفاوت ابری وجود دارد. بسیاری از ارائه‌دهنده‌های سرویس‌های عمومی جدید بسیاری از نیازهای سازمان‌ها را برآورده نمی‌کنند. ارائه‌دهندگان سرویس عمومی نه تنها باید خواسته‌های مشتریان را برآورده کنند بلکه باید سطوح درستی از سرویس‌ها را ارائه دهند و بتوانند آنها را پشتیبانی کنند.

سرویس‌ها به اندازه‌ی کافی قوی نیستند

بسیاری از سرویس‌های پیشنهاد شده توسط ارائه‌دهنده‌های ابر به اندازه‌ی کافی برای برآورده کردن نیازهای مشتری نیرومند نمی‌باشند. بسیاری از سرویس‌های عمومی ابر مشخص هستند. اگر سازمان شما نیاز به یک سرویس خاصی که که به گونه‌ی خاصی ارائه شده است ندارد، شما قادر به استفاده از مزیت‌های سرویس نخواهید داشت. ارائه‌دهندگان به طور متناوب خدمات را بروز می‌کنند یا به آنها اضافه می‌کنند که این برای برآورده کردن نیازهای مشتریان است.

SLAها (توافقنامه سطح سرویس)

بسیاری از ارائه‌دهنده‌های سرویس در نقطه ای قرار ندارند که بتوانند SLA های واقعی را ارائه دهند. برخی ارائه‌دهنده‌ها SLAها را به هیچ عنوان پیشنهاد نمی‌دهند. برخی دیگر SLAها را ارائه می‌دهند، اما این سرویس تضمین می کند که آنها برای بسیاری از سازمان ها مناسب نیستند. سازمان شما ممکن است در دسترسی ۲۴/۷ را برای سرویس یا برنامه‌ی ویژه‌ای نیاز داشته باشد، اما ارائه‌دهنده‌ای امکان ندارد آن را پیشنهاد دهد. یک نکته در این بخش این است که اگر سازمان شما نتواند یک سطح مشخصی از در دسترسی را فراهم کند (به دلیل محدودیت‌های فنی)، ارائه‌دهنده‌ی سرویس برای سرویس یا برنامه‌ی داده شده با همان محدودیت‌های فنی مواجه می‌شود.

[۱۶] Maturity

یکپارچه‌سازی

در هنگام کار با ارائه‌دهنده‌های سرویس، یکپارچه‌سازی یک عنصر کلیدی است. از آنجا که سیستم های استفاده شده توسط ارائه دهندگان خدمات خود را ندارید، شما دسترسی مستقیم به آنها را نخواهید داشت. بدون دسترسی مستقیم، برخی واسط‌ها برای بوجود آوردن امکان برای یکپارچه‌سازی با دیگر سیستم‌ها فراهم می‌شوند. شما ممکن است به یکپارچه‌سازی داده و برنامه نیاز داشته باشید.

یکپارچه‌سازی داده‌ها

یکپارچه‌سازی داده‌ها و گزارش بین سیستم‌های مبتنی بر ابر و سیستم‌های درون سازمانی[۱۷] می‌تواند بسیار پرهزینه باشد. شما باید وسیله ای برای کپی کردن مقادیر زیاد داده ها از یک مکان به مکان دیگر را کشف کنید . پهنای باند استفاده شده در طول فرآیند کپی قطعا هزینه‌ای که برای سرویس می‌دهید را تحت تاثیر قرار می‌دهد. فقدان در دسترسی بی‌درنگ داده‌ها می‌تواند ارائه‌دهنده‌ی مسائلی در بسیاری از شرایط باشد. داده‌های بی‌درنگ اغلب برای گزارش لازم هستند. حرکت داده‌ها به صورت بی‌درنگ می‌تواند پهنای باند زیادی را احتیاج داشته باشد. این مصرف پهنای باند می‌تواند بسیار پرهزینه باشد.

یکپارچه‌سازی برنامه/سرویس

گاهی واسط وب ارائه شده توسط ارائه‌دهنده‌ی سرویس به اندازه‌ی کافی به تنهایی خوب نمی‌باشد. شما ممکن است به برنامه یا سرویس وبی نیاز داشته باشید که باید از مزیت‌های ارائه‌دهنده‌های سرویس دیگر نیز بهره ببرد. بسیار از ارائه‌دهنده‌های سرویس واسط‌ها یا APIهایی را ارائه می‌دهند که می‌توانند برای دسترسی به عملکرد مورد استفاده قرار بگیرند. دسترسی امن به این واسط‌ها به شما امکان دسترسی به عملکرد مورد نیاز برای برنامه‌نویسی را می‌دهد.

[۱۷] on-premises

امنیت

اگرچه برخی فرض می‌کنند پیاده‌سازی‌های ابری نسبت به استقرارهای سنتی در برخی موارد بسیار امن‌تر می‌باشند، اما برخی جنبه‌ها ایمنی کمتری دارند و خطر بیشتری را دارا هستند. این خطر عموما از این واقعیت نشات می‌گیرد که شما کنترل مستقیم روی سیستم‌ها و داده‌ها ندارید. شما باید به آنچه که ارائه‌دهنده‌ی سرویس انجام می‌دهد اعتماد داشته باشید.

مالکیت داده‌ها

درباره‌ی مالکیت داده‌ها در ابر سوالات بسیاری وجود دارد. یک سوال بزرگ در پیاده‌سازی ابر اسن است که چه کسی صاحب داده‌ها است؟ شرکت شما ممکن است داده‌هایی را ایجاد کرده باشد، اما اکنون آیا آن در یک ارائه‌دهنده‌ی سرویس خارجی مرتب‌سازی شده است. آیا هنوز شما صاحب آن می‌باشید؟ چه اتفاقی می‌افتد اگر ارائه‌دهنده‌ی سرویس از این کسب‌وکار بیرون بیاید؟ شما چگونه می‌تواند به داده‌هایتان دست یابید؟ آیا شرکتی که مالکیت سیستم ها را می گیرد، اطلاعات خود را به دست می آورد؟ آیا این شرکت موظف است آن را به شما بدهد؟ این سوالاتی است که شما باید آنها را در هنگام در نظر گرفتن یک ارائه‌دهنده‌ی سرویس در نظر بگیرید. ارائه‌دهنده‌های سرویس متفاوت پاسخ‌های متفاوتی به این سوالات دارند، بنابراین شما باید نسبت به آنچه که از یک ارائه‌دهنده‌ی سرویس انتظار دارید آگاه باشید.

حسابرسی یا رسیدگی

توانایی انجام حسابرسی مناسب در میان محیط‌های ابری متفاوت است. بسته به پیاده‌سازی، شما ممکن است دسترسی مستقیم به سیستم و برنامه‌هایی که می‌خواهید حسابرسی کنید داشته باشید و یا نداشته باشید. ارائه‌دهنده‌ی سرویس ممکن است قادر به ارائه‌ی دسترسی به صورت عملیات مورد نظر برای شما باشد که این از طریق برخی واسط‌های برنامه و یا توسط گزارش صورت عملیات‌ها و ارسال آنها به طور مستقیم به شما ، انجام می‌شود.

مسائل مربوط به حریم خصوصی، حقوقی و پذیرش (تطابق)

حریم خصوصی یک مشکل بزرگی در پیاده‌سازی ابر است. ارائه‌دهنده‌ی ابر ممکن است دسترسی مستقیم به داده‌های سازمان شما داشته باشد. اگر داده خصوصی باشد، باید نگران مقیاس‌هایی باشید

که برای خصوصی نگه داشتن آن بکار گرفته می‌شود. در برخی موارد مشخص، شما ممکن است از استانداردهای شخصی بودن با ذخیره‌ی داده‌ها به همراه ارائه دهنده‌های خارجی تخطی کنید.

مسائل حقوقی و موافقت می‌توانند در هنگام کار با پیاده‌سازی‌های ابر بسیار بغرنج باشند. صلاحیت هنوز مشخص نشده است. اگر شما در ایالات متحده‌ی آمریکا باشید و به سرورها در اروپا دسترسی داشته باشید، چه مقرره‌ای اعمال می‌شود؟ به طور کلی راهنمایی ای اطمینان از این است که شما قوانین را در هر دو حوزه قضایی پیروی می کنید. یکی از روش هایی که می توانید برای اطمینان از اینکه ارائه دهندگان به مقررات مناسب پیروی می کنند، انتخاب یک ارائه دهنده است که یک حسابرسی SAS۷۰ Type II را تصویب کرده است . این حسابرسی اطمینان ایجاد می‌کند که ارائه‌دهنده به شاخص توافق مورد نیاز دسترسی پیدا می‌کند. این حسابرسی‌ها توسط یک سازمان مشاوره‌ی مستقل انجام می‌شود که به منظور نگهداری عدم نقص است.

چندمستاجری

چند مستاجری می تواند مسائل خود را ارائه دهد. شما باید زمانی که سازمان‌های مختلفی دارید که از یک سرویس یکسان استفاده می‌کنند مراقب باشید. بدون شک مسائل امنیتی و مسائل مربوط به سفارشی سازی وجود خواهد داشت.

امنیت

با چند مستاجری شما کنترل و دانش کمی به افرادی که سیستم‌هایی مثل سیستم شما را به اشتراک می‌گذارند دارید. شما ناخودآگاه می‌توانید رقبایی را داشته باشید که از همان سیستم استفاده می‌کنند. اگر رقبای شما قادر به استخراج برخی شکاف‌های امنیتی روی سیستم میزبان باشند، آنها ممکن است قادر به دسترسی به محیط شما نیز باشند. این همان هک است. هکرها نیز فضای ابری را خریداری می‌کنند. هدف اصلی آنها پیدا کردن و بهره‌برداری از حوزه‌هایی است که برای بدست آوردن دسترسی به دیگر محیط‌ها روی همان میزبان مورد استفاده قرار می‌گیرد.

کمبود یا فقدان سفارشی‌سازی

هنگامی که شما برنامه‌ها و سیستم‌ها را با دیگر سازمان‌ها به اشتراک می‌گذارید، محدودیتی بری مقدار سفارشی‌سازی که انجام می‌شود وجود دارد. در برخی موارد، ممکن است شما قادر به

سفارشی‌سازی بدون تحت تاثیر قرار دادن سازمان‌ها دیگر نباشید. در موارد دیگر، ارائه‌دهنده‌ی سرویس نمی‌خواهد یک برنامه‌ی سفارشی شده را پشتیبانی کند. شما باید به یاد داشته باشید که ارائه‌دهنده‌ی سرویس هزاران مشتری دارد. پشتیبانی سفارشی‌سازی برای هر کدام از آن مشتریان بسیار پرهزینه است.

به همین دلایل، شما ممکن است قادر به استفاده از یک ورژن مشخص از یک برنامه برای مدتی که می‌خواهید نباشید. شما ممکن است مجبور به گرفتن ورژن‌های دیگری از برنامه باشید. این ورژن‌های جدید ممکن است نیاز به آموزش‌های اضافه داشته باشند. ابر روی بهره‌وری کمپانی شما تاثیر دارد.

چالش‌های تکنولوژی

اگرچه بهبودهای بزرگی در تکنولوژی‌های ابری بوده است، اما جنبه‌های زیادی برای رشد وجود دارد. بسیاری از تکنولوژی‌ها هنوز رسما به عنوان استاندارد تصویب نشده‌اند. این منجر به مسائل سازگاری می‌شود. احراز هویت در این مورد مثال خوبی است. اگرچه پروتکل‌های احراز هویت استانداردی ایجاد شده است، اما آنها به طور گسترده‌ای مورد استفاده قرار نمی‌گیرند.

تغییر مقیاس افقی[۱۸]

محیط‌های ابری عموما از ابزارهای commodity برای زیرساخت‌هایشان استفاده می‌کنند. در بسیاری از موارد این به معنای اضافه کردن ظرفیت است، زیرا شما به جای بزرگ کردن به تغییر

[۱۸] تغییر مقیاس افقی
به تغییر مقیاس افقی که Scale out نیز گفته می شود، به معنی افزایش گره‌های بیشتر به سیستم است. به عنوان مثال می توان از افزایش تعداد وب سرورها از یک به سه عدد نام برد.
تغییر مقیاس عمودی
تغییر مقیاس عمودی که Scale up نیز گفته می شود، به معنی افزودن منابع به یک گره (Node) از سیستم - به طور نمونه ارتقای پردازنده یا رسانه ذخیره سازی در یک کامپیوتر- می باشد. افزایش بهره برداری از منابع نیز نوعی تغییر مقیاس عمودی است. به عنوان مثال می توان افزودن تعداد پروسه‌های در حال اجرای دمون (به انگلیسی: Daemon) های کارساز اچ‌تی‌تی‌پی آپاچی (به انگلیسی: Apache HTTP Server) را ذکر کرد.

مقیاس افقی نیاز دارید. تغییر مقیاس افقی می‌تواند منجر به بار افزایش یافته روی یک مرکز داده و هزینه‌های مرتبط به محیط افزایش یافته در منابعی مانند خنک کردن و برق بشود.

سیاست‌های شرکت

اگر پیش از این شرکت شما تنهای از منابع داخلی استفاده کرده است، سیاست‌ها و پروسه‌های شما باید به منظور در نظر گرفتن محیط‌های ابری بروزرسانی شود. شما ممکن است سیاست‌هایی را توسعه دهید که می‌توانند در هنگامی که شما کنترل روی محیط ابری را کامل کرده‌اید و هنگامی که شما این کار را انجام نداده‌اید اعمال شوند. شما به سیاست‌هایی برای تعیین این که چه چیز می‌تواند به ابر منتقل شود و چه چیز نمی‌تواند نیاز دارید. هم‌چنین شما به سیاست‌هایی حول این که چه چیز از ارائه‌دهنده‌های سرویس انتظار می‌رود نیز نیاز دارید.

انعطاف‌پذیری

انتخاب یک محیط ابری می‌تواند تا حدودی محدود کننده باشد. شما باید در نظر بگیرید که اگر از یک ارائه‌دهنده‌ی سرویس راضی نمی‌باشید، تغییر ارائه‌دهنده چقدر دشوار است. تغییر ارائه‌دهنده بسیار دشوار است. یک نگرانی بزرگ این است که انتقال داده‌ها از یک ارائه‌دهنده به دیگری تا چه میزان دشوار است. در برخی موارد این امر آنقدر پرهزینه است که این را را غیر ممکن می‌کند.

خلاصه

پنج خصوصیت ابری کلیدی وجود دارد: سلف‌سرویس مورد تقاضا، دسترسی به شبکه گسترده، توزیع منابع، انعطاف پذیری سریع و خدمات اندازه گیری شده. یک راه‌حل باید این پنج خصیصه را داشته باشد تا بعنوان یک راه‌حل درست در نظر گرفته شود. چهار مدل استقرار ابری وجود دارد: عمومی، خصوصی، گروهیی و هیبریدی. هر مدل با توجه به این که زیرساخت برای محیط کجا قرار گرفته می‌شود تعریف می‌شود. سه مدل سرویس ابری وجود دارد: نرم‌افزار بعنوان سرویس، پلت‌فرم بعنوان سرویس، و زیرساخت بعنوان سرویس. SaaS مدل ابری ارجینال است اما ابر به رشد و بسط ادامه داده است. اکنون مدل‌های سرویس بسیاری در دسترس است. فاکتورهای زیادی برای حرکت سازمان‌ها به سوی ابر وجود دارد، و هم‌چنین فاکتورهای زیادی نیز وجود دارد که آنها را از این موضوع دور نگه می‌دارد. هر سازمان باید ابرهایی را که برای آنها مناسب است را ارزیابی کند و بهترین را انتخاب کند.

فصل ۲. طرح مفاهیم اولیه

نکات این فصل

- احراز هویت
- مفاهیم رایانش
- مجازی‌سازی سخت‌افزار
- تکنولوژی‌های توسعه‌ی وب

مقدمه

ابر یک سرویس است، اما عناصر تکنولوژی متعددی وجود دارند که برای ممکن ساختن ابر کنار هم قرار گرفته‌اند. این تکنولوژی‌ها و پیشرفت‌های تکنولوژی مسئول رشد سریع ابر و در دسترسی برنامه‌های ابری هستند. ما به صورت جزئی درباره‌ی تکنولوژی‌ها صحبت نمی‌کنیم، اما این مهم است که شما درک عمومی از آنها داشته باشید. زیرا در هنگامی که شما باید یک ارائه‌دهنده و محصول ابری را انتخاب کنید، اگر شما بتوانید بین این تکنولوژی‌ها تمایز ایجاد کنید و بدانید هر کدام چه چیز را پیشنهاد می‌دهند بسیار مزایامند است.

احراز هویت

احراز هویت فرآیند تشخیص این است که آیا کاربران همانی هستند که می‌گویند. در بسیاری از سیستم‌ها پیش از دسترسی به منابع، شما باید ابتدا هویت خود را احراز کنید. هرگاه اطلاعات حساس مبهم باشند و هر زمان که نیاز به بررسی باشد، شما باید اطمینان ایجاد کنید که آن فرد، فرد درست است. اگر چنین کاری انجام نشود، شما نمی‌توانید به آن فرد اعتماد کنید و یا حتی نمی‌توانید به اطلاعات فراهم شده توسط آنها نیز اعتماد کنید. روش‌های متفاوتی وجود دارند که برای احراز هویت فرد یا شی مورد استفاده قرار می‌گیرد. این مهم است که شما احراز هویت درستی را بکار گیرید که

با توجه به شرایط است. احراز هویت بخش مهم هر محیطی محسوب می‌شود و ابر نیز از این مستثنی نمی باشد. در واقع احراز هویت در محیط ابری عمومی بسیار مهم‌تر از محیط‌های سنتی است. احراز هویت روش اصلی برای محدود کردن دسترسی به برنامه‌ها و داده‌ها است. از آنجایی که برنامه‌های ابری عمومی از طریق وب در دسترس می‌باشند، آنها می‌توانند به صورت نظری در دسترس همه باشند. به همین علت، ارائه‌دهنده‌های سرویس باید اطمینان ایجاد کنند که اقدامات احتیاطی مناسبی را برای حفاظت از برنامه‌ها و داده‌های کاربران به کار گرفته‌اند. این فرآیند با اطمینان از این که روش‌های احراز هویت مناسب انتخاب شده‌اند آغاز می‌شود. به طور مشابه، هنگامی که شما ارائه‌دهنده‌ی ابری را ارزیابی می‌کنید، باید اطمینان ایجاد کنید که آنها مقیاس‌های مناسب برای احراز هویت دارند. اطلاعاتی که در این بخش در اختیار شما قرار داده می‌شود به شما در ارزیابی کمک می‌کند. ابتدا ما اطلاعات پایه ای درباره‌ی احراز هویت و تایید هویت در اختیار شما قرار می‌دهیم؛ سپس ما ارائه‌دهنده‌ها و احراز هویت متعهد را شناسایی می‌کنیم .

شناسایی در برابر تأیید

هنگامی که شما به مساله‌ی احراز هویت می‌پردازید، شما می‌توانید آن را به عناصری تقسیم کنید: شناسایی و تایید. شناسایی فرآیندی است که شما بیان می‌کنید چه کسی هستید. این بیان می‌تواند به صورت یک نام کاربری، آدرس ایمیل، و یا روش‌های دیگری که شما را شناسایی می‌کند، باشد. اساسا شما می‌گویید، "I am drountre " یا "I am derrick@gmail.com " و "من دسترسی به منابعی را می‌خواهم که برای من در دسترس هستند". این سیستم نمی‌تواند تنها دسترسی برای کسی را که ادعا می کند که drountre است را فراهم کند. در این‌جا تایید باید انجام شود. تایید فرآیندی است که یک سیستم به بررسی این که شما واقعا کسی که می‌گویید هستید می‌پردازد. این فرآیند چیزی است که بیشتر افراد در هنگام اهراز هویت به آن فکر می‌کنند. آنها نمی‌توانند درک کنند که بخش اول این فرآیند که شما باید عبارتی درباره‌ی کسی که هستید را ایجاد کنید. تایید می‌تواند به روش‌های متفاوتی انجام شود. شما نرم‌افزار یا پینی را فراهم می‌کنید و یا از برخی از انواع شناساگرهای بیومتریک استفاده می‌کنید.

به این روش فکر کنید: شما می‌دانید که هنگامی که برای احراز هویت در یک سیستم تلاش می‌کنید و نام‌کاربری و پسورد خود را وارد می‌کنید، سیستم چک خواهد کرد که آیا این ترکیب درست است یا خیر. شما باید پسورد درستی را وارد کرده باشید که متناظر به نام کاربری وارد شده می‌باشد. اگر یکی از آنها نادرست باشد، احراز هویت دچار شکست می‌شود. سیستم اول بررسی می‌کند که آیا نام

کاربری وارد شده درست است. اگر چنین نباشد، سپس پیام خطا برگردانده می‌شود. اما اگر نام کاربری درست باشد، سیستم پسورد را بررسی می‌کند. ترکیب درست این‌ها (نام کاربری و پسورد) برای احراز هویت موفق الزامی است.

اجازه دسترسی[۱۹]

بعد از این که کاربران تایید هویت می‌شوند، اجازه و دادن مجوز آغاز می‌شود. مجوز فرآیند مشخص کردن چیزی است که کاربر اجازه‌ی انجام آن را دارد. مجوز تنها درباره‌ی سیستم‌ها و دسترسی سیستمی نمی‌باشد. مجوز امکان و توانایی است که در هر جا کاربر دارد. هر سازمان باید یک سیاست امنیتی داشته باشد که مشخص می‌کند چه کسی امکان دسترسی به چه منابعی را دارد و آنها اجازه‌ی انجام چه کاری را روی این منابع دارند. سیاست‌های مجوز می تواند توسط هر چیزی از نگرانی‌های حفظ حریم خصوصی برای انطباق با مقررات تحت تاثیر قرار بگیرد. این مهم است که سیستمی که شما در آن قرار دارید قادر به اعمال سیاست مجوز شما باشد؛ این شامل سیستم‌های مبتنی بر ابر عمومی است.

روش‌های تایید هویت پیشرفته

در ایمن کردن برنامه‌های داده‌ای، احراز هویت نام کاربری و پسورد کافی نیست. شما باید مراقبت‌های بیشتری را در شرایطی که شناسایی فرد مورد نظر مهم است داشته باشید، مانند درخواست‌های خارجی بخ سیستم‌های داخلی. سیستم‌های ابری عمومی می‌تواند هم‌چنین یک ریسک شدیدی را نشان دهند. از آنجایی که برنامه‌های ابری عمومی و داده‌ها به راحتی در اینترنت در دسترس هستند، شما ممکن است به دنبال ارائه‌دهنده‌ای باشید که روش‌های احراز همیت پیشرفته‌ای را برای ایمن کردن آنها ارائه می‌دهد. دو روش که عموما مورد استفاده قرار می‌گیرند: احراز هویت چند فاکتوره و احراز هویت مبتنی بر ریسک است.

[۱۹] Authorization

احراز هویت چند فاکتوره

یک روش برای اطمینان از امنیت احراز هویت مناسب استفاده از احراز هویت چند فاکتوره است. احراز هویت چند فاکتوره به دلیل این که از چندین فاکتور احراز هویت استفاده می‌کند این‌گونه نامیده شده است. شما ممکن است یک فاکتور را یک دسته از احراز هویت در نظر بگیرید. سه فاکتور احراز هویت وجود دارد که می‌تواند مورد استفاده قرار بگیرد: چیزی که شما می‌دایند، چیزی که شما دارید، چیزی که شما هستید. چیزی که شما می‌دانید پسورد، تولد، یا یک اطلاعات شخصی است. چیزی که شما دارید رمز یکبار استفاده، کارت هوشمند، و دیگر چیزهایی است که شما ممکن است به صورت فیزیکی در اختیار داشته باشید. چیزی که شما هستید هویت بیومتریکی شما می‌باشد، مانند اثر انگشت و الگوی صحبت. به منظور این که چیزی احراز هویت چند فاکتوره در نظر گرفته شود، آن باید از حداقل دوتا از سه فاکتوره بیان شده استفاده کند. برای مثال هنگامی که کاربر تلاش در تصدیق دارد، وی ممکن است پسورد و کد یکبار مصرفش را وارد کند.

احراز هویت چند فاکتوره توسط تعدادی ارائه‌دهنده‌ی سرویس که رو به رشد است پیشنهاد داده می‌شود، به ویژه آنهایی که داده‌های حساس را ذخیره‌سازی می‌کنند. بنابراین اگر شما حس می‌کنید احراز هویت چندفاکتوره برای شما لازم است، شما باید آن را از ارائه‌دهنده درخواست کنید.

احراز هویت مبتنی بر ریسک

احراز هویت مبتنی بر ریسک برای بدست آوردن شهرت شروع کرد. این احراز هویت به دلیل افزایش خطرهایی که برنامه‌های عمومی و سایت‌ها وب با آن مواجه می‌شدند بوجود آمد. احراز هویت مبتنی بر خطر از یک پروفایل ریسک برای تعیین این که درخواست احراز هویت می‌تواند مشکوک باشد استفاده می‌کند. به هر تلاش احراز هویت یک امتیاز خطر داده می‌شود. اگر امتیاز خطر فراتر از یک مقدار مشخص باشد، ارائه‌دهنده‌ی سرویس یا سایت وب می‌تواند درخواست اطلاعات بیشتری را پیش از امکان به دسترسی بدهد. این اطلاعات سنتی می‌توانند به صورت سوال‌های امنیتی یا فاکتورهای احراز هویت اضافه باشند.

یک خطر براساس خصوصیات سیستم و کاربر محاسبه می‌شود. سایت یک پروفایل برای هر کاربر براساس اطلاعاتی مانند زمان ورود معمول، سیستم مورد استفاده برای دسترسی به سایت یا روش دسترسی فراهم می‌کند. هنگامی که کاربری تلاش به دسترسی به سایت را دارد و خصوصیات مصرفی اخیرش با پروفایلش تطبیق ندارد، امتیاز ریسکش بازتاب‌کننده‌ی تغییر است.

احراز هویت مبتنی بر خطر در سایت‌های مالی و بانکی رایج است. اما، مانند احراز هویت چند فاکتور، احراز هویت مبتنی بر خطر تبلیغ نمی‌شود، بنابرایم شما باید از ارائه‌دهنده‌ی ابر بپرسید که آیا می‌توانند آن را تامین کنند.

ارائه‌دهندگان هویت

در عرصه‌ی تایید هویت، فراهم‌کننده‌ی سرویس خاصی وجود دارد که ارائه‌دهنده‌ی هویت نام دارد. یک ارائه‌دهنده‌ی هویت یا IdP، موجودیتی است که اطلاعات هویتی را نگهداری و مدیریت می‌کند. شما می‌تواند IdP را به صورت داخلی تنظیم کنید، و یا می‌تواند یک ارائه‌دهنده‌ی سرویس را بکار گیرید. کاربران، یا موجودیت‌ها، در برابر مخازن اعتبارنامه‌ی IdP احراز هویت می‌شوند. سپس IdP امکان دسترسی به اطلاعات هویتی کاربران را می‌دهد. لازم به ذکر است که IdP بیش از احراز هویت کاربر را انجام می‌دهد. این اطلاعات می‌توانند به هر کسی که به آن نیاز دارد ارسال شود. عموما، این یک ارائه‌دهنده‌ی سرویس است، که به آن بخش متکی[۲۰] گفته می‌شود. این بخاطر این است که سرویس مبتنی بر IdP برای احراز هویت و شناسایی اطلاعات است.

مخزن اعتبار[۲۱]

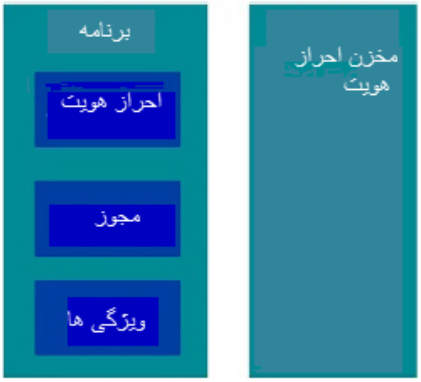
مخزن اعتبار، که گاهی انبار کاربر یا انبار احراز هویت نامیده می‌شود، جایی است که اعتبارهای واقعی کاربر در آنها ذخیره می‌شود. دو نوع اصلی مخزن احراز هویت که همراه با IdPها مورد استفاده قرار می‌گیرد: مخزن‌های دیتابیس و دایرکتوری هستند. به طور کلی، با دیتابیس‌ها، اعتبارها در جداول اختصاصی ایجاد شده توسط برنامه‌ی مدیریت کاربر ذخیره می‌شوند. یکی از دلایلی که دیتابیس‌ها اغلب بعنوان مخازن اعتباری انتخاب می‌شوند این است که اغلب توسعه‌دهنده‌ها تجربه‌ی برنامه‌نویسی در دیتابیس‌ها را دارند، بنابراین نوشتن کد برای احراز هویت کاربران در دیتابیس‌ها نسبتا آسان است. مخازن دایرکتوری شامل پروتکل دسترسی سبک‌وزن راهنما[۲۲] (LDAP) و پیاده‌سازی دایرکتوری فعال هستند. LDAP یک روش مبتنی بر استانداردهای ساده را برای دسترسی به اطلاعات از انبار

[۲۰] Relying party
[۲۱] Credential Store

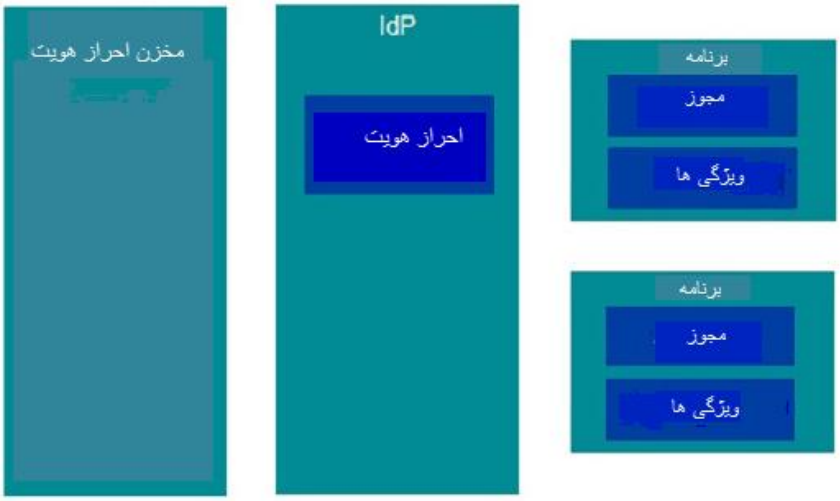
[۲۲] Lightweight Directory Access Protocol

اعتبار فراهم می‌کند. دایرکتوری فعال یک روش مبتنی بر دامنه‌ی مایکروسافت برای LDAP است. استفاده از یک مخزن اعتبار AD عموما نیازمند این است که شما از تکنیک‌های دسترسی اختصاصی استفاده کنید. بسیاری از ارائه‌دهنده‌های سرویس ابری اکنون گزینه‌ای برای استفاده از انبار اعتبار داخلی به جای استفاده از انبارهای سوم را می‌دهند. در این روش کاربر نیاز به بخاطر سپردن چندین مجموعه از اعتبارها ندارد.

به منظور کمک به شما برای درک این مفهوم، به شکل (۲-۱) و (۲-۲) مراجعه کنید. شکل (۲-۱) نشان‌دهنده‌ی یک ساختار احراز هویت سنتی است که در آن برنامه‌ها مستقیما با انبار احراز هویت در ارتباط هستند. شکل (۲-۲) نشان‌دهنده‌ی این است که آن با IdP به چه صورت کار می‌کند. برنامه‌ها با IdP ارتباط دارند و IdP هم با مخزن اعتبار در ارتباط است.



شکل ۲-۱: ساختار احراز هویت سنتی



شکل ۲-۲: ساختار احراز هویت متعهد

IdPهای عمومی

ارائه‌دهنده‌های هویت می‌توانند عمومی یا خصوصی باشند. استفاده از IdPهای عمومی به طور پیوسته رو به رشد است. به جای ایجاد IdPهای داخلی، بسیاری از سازمان‌ها استفاده از ارائه‌دهنده‌ی سرویس IdP را انتخاب کرده‌اند. استفاده از IdP خارجی می‌تواند زمان و هزینه‌ی زیادی را برای شما ذخیره کند. IdPهای عمومی متفاوتی برای استفاده در دسترس هستند. ما برخی از آنها را پوشش می‌دهیم.

OpenID

ما با OpenID آغاز می‌کنیم. OpenID یک استاندارد برای احراز هویت می‌باشد. آن یک چارچوبی را فراهم می‌کند که ارائه‌دهنده‌ها می‌توانند برای اطمینان از قابلیت همکاری راه‌حل‌هایشان از آن استفاده کنند. OpenID تکنولوژی است که به شما امکان پیاده‌سازی محیطی را می‌دهد که در آن

احراز هویت از مجوز گرفته می‌شود. با OpenID، احراز هویت می‌تواند جدا از یک برنامه یا دیگر منابع باشد. شما می‌توانید از یک موجودیت مرکزی استفاده کنید، مانند IdP ، که این برای انجام احراز هویت برای چندین وب‌سایت و منبع است. از آنجایی که IdP به استاندارد OpenID اضافه می‌شود و ارائه‌دهنده‌ی سرویس آن را پشتیبانی می‌کند، قابلیت همکاری به خوبی کار خواهد کرد. OpenID چندین مزیت کلیدی را فراهم می‌کند. اولا، ارائه‌دهنده‌ی سرویس نباید دیگر نگران نگهداری قابلیت‌های احراز هویت باشد. آنها نباید پشتسبانی احراز هویت را در برنامه یا سرویسشان ایجاد کنند. هم‌چنین آنها نباید نگران نگهداری مخازن اعتباری یا مدیریت کاربر باشند. تنظیم مجدد پسورد و چیزهایی که به سرعت هزینه های پشتیبانی را افزایش می دهد، حذف می‌شوند. دوما، با OpenID، ارائه‌دهنده‌ی سرویس به این که چه روش‌هایی برای احراز هویت کاربر به کار گرفته شده است، اهمیت نمی‌دهد. این به شما امکان انتخاب یک طرح اهراز هویت را که نیازهای سازمان شما را بدون نگرانی درباره‌ی این که با برنامه‌ی شما کار خواهد کرد یا خیر، برآورده می‌کند. شما هم‌چنین در تغییر طرح احراز هویت خود در هنگامی که فکر می‌کنید لازم است، آزادی عملی بیشتری را احساس می‌کنید. آن می‌تواند تغییر در همان IdP باشد، یا شما می‌توانید یک IdP جدیدی را انتخاب کنید. از آنجایی که IdP ، OpenID را پشتیبانی می‌کند، برنامه‌ی شما به این که مکانیزم احراز هویت تغییر کرده است اهمیت نمی‌دهد. اگر شما IdP جدیدی را انتخاب کرده باشید، شما باید اعتبار جدیدی را بین برنامه یا سرویس و IdP جدید ایجاد کنید، اما آن احراز هویت در سرویس یا برنامه را تغییر نمی‌دهد. این نوع انعطاف‌پذیری می‌تواند یک مزیت بزرگ در چشم انداز همیشه در حال تغییر امروزی باشد.

شکل ۲-۳: آرم OpenID

Google

IdP گوگل مبتنی بر استاندارد OpenID است. آن مطابق با OpenID ۲,۰ است. IdP گوگل هم‌چنین بسط‌های زیر را نیز پشتیبانی می‌کند: مبادله‌ی ویژگی OpenID ۱,۰[۲۳]، واسط کاربری OpenID ۱,۰، پروتکل هیبریدی OpenID + Oauth، و بسط‌های سیاست احراز هویت ارائه‌دهنده (PAPE). هنگامی که شما از حساب گوگل خود استفاده می‌کنید تا به سایتی وارد شوید (مانند YouTube)، شما در واقع از IdP گوگل استفاده می‌کنید.

Facebook

Facebook یک فراهم‌کننده‌ی هویت است که به شدت رو به رشد می‌باشد. Facebook اخیرا از OAuth ۲,۰ برای فراهم کردن احراز هویت و اختیار استفاده می‌کند. Facebook چندین API و کیت توسعه ی نرم‌افزار را ارائه می‌دهد که به شما در یکپارچه‌سازی ورود Facebook با برنامه‌یتان کمک می‌کند. شما می‌توانید از جاوا اسکریپت کلاینت، فراخوانی‌های دستگاه طبیعی (آندروید، IOS و غیره) و یا اجرای سمت سرور استفاده کنید. به منظور کسب اطلاعات بیشتر درباره‌ی استفاده از ارائه‌دهنده‌ی هویت Facebook به سایت www.facebook.com/developers مراجعه کنید.

حساب مایکروسافت

مایکروسافت یک فراهم‌کننده‌ی هویت است، که از پیش با نام Windows Live شناخته شده است. IdP پیش فرض آن در تمام وبسایت‌های مرتبط با مایکروسافت مورد استفاده قرار گرفته است. آن هم‌چنین IdP پیش فرض برای سرویس کنترل دسترسی مایکروسافت، سرویس ارائه دهندگان هویت متعهد مایکروسافت[۲۴] می‌باشد.

[۲۳] OpenID Attribute Exchange
[۲۴] Microsoft’s federated identity provider service

هویت فدرال یا یکپارچه

هویت فدرال یا یکپارچه یک روش امن برای سیستم‌های متفاوت به منظور بدست آوردن دسترسی به اطلاعات هویتی شما است. اما با هویت یکپارچه، سیستم های دیگر نیز می‌توانند به این اطلاعات دست یابند. کلید هویت یکپارچه اعتماد است. سیستمی که اطلاعات شما را نگهداری می‌کند و سیستمی که اطلاعات شما را درخواست می‌کند باید به یکدیگر اعتماد کنند. برای اطمینام از این که اطلاعات شما به مکان قابل اعتمادی منتقل شده است، سیستمی که اطلاعات را نگهداری می‌کند، سیستمی که اطلاعات شما را نگهداری می‌کند باید به سیستمی که اطلاعات شما را درخواست می‌کند اعتماد کند. سیستمی که درخواست اطلاعات را می‌دهد باید به ارسال‌کننده به منظور اطمینان از این که آنها اطلاعات درست و دقیق را دریافت می‌کنند، اعتماد کند.

اصولا یک برنامه به موجودیت دیگر اعتماد می‌کند، که آن IdP است، یعنی زمانی که موجودیت می‌گوید کاربر چه کسی است، می‌پذیرد. در واقع برنامه خودش اقدامی را انجام نمی‌دهد تا هویت کاربر را تصدیق کند. آن به سادگی آنچه را که IdP می‌گوید باور می‌کند. پیش از اعتماد برنامه به IdP، یک رابطه‌ی اعتماد باید بین آنها بوجود بیاید. برنامه باید با آدرس IdP که به آن اعتماد دارد پیکربندی شود. IdP باید با آدرس برنامه پیکربندی شود. در بیشتر موارد، برخی کلیدها بین دو موجودیت تبادل می‌شود که این برای برقراری رابطه است. این کلیدها توسط موجودیت‌ها برای شناسایی یکدیگر استفاده می‌شود.

سرویس‌های کنترل دسترسی مایکروسافت (ACS)

شما ممکن است تصمیم بگیرید که صادرکننده‌ی میزبانی‌شده‌ی خارجی مانند سرویس کنترل دسترسی مایکروسافت را انتخاب کنید. ACS یک سرویس وب مبتنی بر ابر Azure ویندوز است که برای شناسایی و مدیریت دسترسی استفاده شده است. ACS می‌تواند برای فراهم‌کردن عملکرد احراز هویت و اختیار برای سرویس‌ها و برنامه‌ها وب مورد استفاده قرار بگیرد. به این صورت، آن توابع نباید به صورت مستقیم در کد برای سرویس و برنامه ایجاد شوند. یک مزیت کلیدی ACS این است که به دلیل این که آن یک نمونه‌ی مبتنی بر ابر است، هیچ نصبی مورد نیاز نمی‌باشد. شما هنوز باید نمونه‌هایی را برای محیط خود پیاده‌سازی کنید، اما هیچ چیز نیازی به نصب ندارد.

ACS بسیار منعطف است. آن مطابق با تعداد زیادی از پروتکل‌ها و محیط‌ها می‌باشد. این به شما امکان می‌دهد تا به راحتی ACS را در محیط خود بکار گیرید. ACS پروتکل‌های استاندارد صنعتی

را پشتیبانی می‌کند مانند Oauth، OpenID، WS-Federation، و WS-Trust. هم‌چنین ACS چندین نوع فرمت توکن(رمز) را نیز پشتیبانی می‌کند. آن فرمت‌های ۱,۲ SAML، SAML ۲,۰، JWT و SWT را پشتیبانی می‌کند. ACS توسعه را با استفاده از بسیاری از پلت‌فرم‌های وب پشتیبانی می‌کند. شما می‌توانید از NET، PHP، Python، Java و ... استفاده کنید. ACS شامل یک میزبان از عملکردهایی است که برای بیشتر محیط‌های هویت متعهد حیاتی است. ACS به شما امکان پیاده سازی تنها یک عملکردی را که برای پیاده‌سازی شما ضروری است را می‌دهد. ACS عملکرد زیر را فراهم می‌کند: تأیید هویت، مجوز، متعهدسازی، جریان و انتقال رمز امنیتی، مدیریت اعتماد، مدیریت و اتوماسیون.

مفاهیم محاسباتی

دو مفهوم رایانشی کلیدی در پیاده‌سازی ابری نقش دارد. این مفاهیم به ایجاد فلسفه بسیاری از پیاده سازی ابر کمک می‌کند.

محاسبات خدمات همگانی[۲۵]

مفهوم محاسبات خدمات همگانی قدمت زیادی دارد، اما در به تازگی مورد استفاده قرار می‌گیرد. محاسبات خدمات همگانی عمل آموزش محاسبه منابع مانند سرویس اندازه‌گیری شده است، که ما این کار را برای برق و آب انجام می‌دهیم. یک شرکت خدمات همگانی تنها برای شما، آب و برق مصرفی را محاسبه می کند. ارائه‌دهنده‌ی سرویس نیز برای شما تنها منابع مورد استفاده را محاسبه می کند. این مفهوم محاسبه هزینه بر اساس مصرف در مرکز متدولوژی ابر عمومی است. منابعی در دسترس شما می‌باشند، اما شما باید برای آنهایی که استفاده می‌کنید هزینه پرداخت کنید. هم‌چنین یک پرداخت ماهانه نیز وجود دارد که مربوط به در دسترس بودن منابع است، اما بخش عمده‌ای از هزینه براساس استفاده واقعی شما است.

[۲۵] Utility: آب، برق، گاز و ...

سرورهای Commodity یا مصرفی

مفهوم سرورهای Commodity شامل استفاده از سرورهای عمومی و غیر تخصصی برای انجام یک کار می‌باشد. به جای استفاده از سرورهای مختلف برای انجام کار، شما برای همه‌ی کارها تنها از یک سرور استفاده می‌کنید. عموما، سرورهای Commodity سیستم‌هایی با هزینه‌های کمتر می‌باشند. به جای قرار دادن بسیاری از وظایف روی یک سرور قدرتمند، شما می‌تواند کار را روی یکسری از سرورها با قدرتمندی کمتر انتشار دهید. این عمل هم‌چنین بعنوان تکه تکه کردن به جای بزرگ کردن شناخته شده است.

ارائه‌دهنده‌ی ابری اغلب از سرورهای Commodity برای ایجاد زیرساخت مجازی استفاده می‌کنند. این دقیقا کاری است که آمازون برای پیاده‌سازی ابریش انجام داده است. در واقع، آمازون به گونه موفق بوده است که ارائه‌دهنده‌های دیگر نیز در حال تلاش برای کپی کردن این مدل هستند.

محاسبات خود گردان

محاسبات خود گردان که توسط Paul Horn از IBM در ۲۰۰۱ ارائه شده است، این دید را که هر سیستم محاسبات خودش را به صورت خودکار مدیریت کند، را به اشتراک گذاشته است. آن اشاره به خصوصیت‌های خودمدیریتی منابع محاسباتی توزیع شده دارد، که تغییرات را در سیستم درک و تشخیص می‌دهند، اقدام درست را برای تصحیح مناسب آن به صورت اتوماتیک بکار می‌گیرند، که این همراه با هیچ‌گونه مداخله‌ی انسانی انجام می‌شود.

مزایا کلیدی کاهش شدید در پیچیدگی ذاتی سیستم های محاسباتی و بصری تر و راحت تر کردن محاسبات توسط اپراتورها و کاربران است. در این دید سیستم‌های محاسباتی خود گردان، خود بهینه‌سازی، خود محافظتی و خود بهبودی دارند.

مستقلا تلاش‌های مشابهی به سمت ساده سازی مدیریت فناوری اطلاعات منجر شد، مانند متدولوژی‌های ITIL (کتابخانه‌ی زیرساخت IT) و تکنولوژی‌های ITSM (مدیریت سرویس IT)، WSDM (مدیریت توزیع‌شده‌ی سرویس‌های وب) و غیره. چندین گروه تحقیقاتی هنوز روی سیستم‌های خود بهبود دهنده و سیستم‌های مدیریت سیاست که می‌توانند توافق‌های در سطح سرویس پیچیده‌ای را برای فعالسازی بهتر تصمیم‌گیری خودکار مدیریت کنند، کار می‌کنند. ما موفقیت‌هایی را با بسیاری از محصولات بدست آورده‌ایم و هم‌چنین کنترل‌پذیری یکی از اهداف مهم

ما می‌باشد. با توجه به این که هدف رایانش ابری ساده‌سازی سیستم‌ها رایانشی و فراهم‌کردن قابلیت ارتجاعی در رایانش و دسترسی بالای سیستم است، هر ابتکار جدید در بیشتر خودکار کردن ماشین‌ها مستقیما به زیرساخت‌های ابری اضافه می‌شود. تکنولوژی‌های مجازی‌سازی سطح درستی از انتزاع را برای مدیریت دینامیکی تغییرات منابع سخت‌افزاری و تهیه‌ی قابلیت ارتجاعی مورد تقاضا فراهم کرده اند. این که بگوییم رایانش ابری دید محاسبات خود گردان را به اشتراک می‌گذارد نادرست نمی‌باشد.

ارائه‌دهندگان سرویس برنامه های کاربردی

روند میزبانی برنامه های کاربردی بعنوان سرویس برای دیگر برنامه‌ها به منظور استفاده در اوایل دهه‌ی ۱۹۹۰ آغاز شد. فروشندگانی که میزبانی این برنامه‌ها را با استفاده ی کلاینت‌هایشان تنها از مرورگرهای وب امکان‌پذیر می‌ساختند، ارائه‌دهنده های سرویس برنامه های کاربردی نامیده می‌شدند. با این تعریف، این بسیار شبیه به SaaS به نظر می‌رسد و فروشندگان SaaS می‌توانند ASPها امیده شوند. با این حال در هنگامی که برنامه‌های خارج از قفسه[۲۶] با یک واسط مبتنی بر مرورگر بعنوان سرویس میزبانی می‌شدند محدودیت‌هایی وجود داشت. بسیاری از این برنامه‌ها قابلیت مدیریت چند مستاجری و مصرف سفارشی‌سازی شده برای هر کاربر را نداشتند، و هم‌چنین استقرار خودکار و قابلیت ارتجاع برای مقیاس مورد تقاضا را نداشتند. با این اوصاف می‌توان گفت که مدل Saas رایانش ابری از مدل ASP گرفته شده است.

برای فهمیدن تفاوت بین مدل‌های SaaS و ASP به سایت *www.luitinfotech.com/kc/saas-aspdifference.pdf* مراجعه کنید.

مجازی‌سازی سخت افزار

هنگامی که بسیاری از افراد درباره‌ی ابر فکر می‌کنند، در واقع آنها به مجازی‌سازی فکر می‌کنند. اما در واقع مجازی‌سازی ملزم به ایجاد محیط ابری نیست.اگر شما به خصوصیات محیط ابری فکر کنید، هیچ‌کدام از آنها لزوما به مجازی‌سازی نیاز ندارند. اگر آن الزامی نیست، اما مجازی‌سازی در بسیاری از پیاده‌سازی‌های ابری بکار گرفته شده است. این بخاطر آن است که مجازی‌سازی می‌تواند قابلیت تحویل خصوصیات الزامی در محیط ابری را افزایش دهد. برای مثال افزایش ظرفیت بسیار ارزان‌تر

[۲۶] off-the-shelf

خواهد بود که این با اضافه کردن ماشین مجازی جدید انجام می‌گیرد و نه تهیه‌ی سیستم‌های فیزیکی. مجازی‌سازی سخت‌افزار رایج‌ترین نوع مجازی‌سازی است. مجازی‌سازی سخت‌افزار برای ایجاد یک سیستم فیزیکی شبیه‌سازی شده در یک سیستم فیزیکی واقعی استفاده می‌شود. در بسیاری از موارد سیستم‌های فیزیکی شبیه‌سازی شده‌ای وجود دارد. به این صورت است که مجازی‌سازی سخت‌افزار برای ایجاد چگالی سیستم و افزایش مصرف سیستم بکار گرفته می‌شود. این سیستم‌های مجازی استفاده از منابع فیزیکی را به اشتراک می‌گذارند. بنابراین در هنگامی که یک سیستم مجازی از منابع سیستم فیزیکی استفاده نمی‌کند، منابع ممکن است توسط سیستم فیزیکی دیگری استفاده شود. در یک محیط غیرمجازی شده، منابع سیستمی برای بازه‌ی زمانی طولانی بیکار خواهند بود. شما برای سیستم پرداخت کرده‌اید اما از تمام پتانسیل آن استفاده نمی‌کنید.

هایپروایزرها[۲۷] (یا ناظران ماشین مجازی)

مجازی‌سازی سخت‌افزار از طریق استفاده از هایپروایزرها انجام می‌گیرد. هایپروایزرها مجازی‌سازی مخزن و شبکه را پیشنهاد می‌دهند، اما ویژگی‌های قوی توسط محصولات دیگر اضافه می‌شود. ما در این بخش برخی از هایپروایزرهای رایج استفاده شده در محیط‌های ابری امروزی را پوشش می‌دهیم. بسته به این که شما از چه سرویسی استفاه می‌کنید، هایپروایزر در تصمیم شما بسیار مهم است. شما باید اطمینان ایجاد کنید که هایپروایزر ویژگی‌های مورد نیاز پیاده‌سازی شما را پشتیبانی می‌کند.

اصول هایپروایزر

هایپروایزر چیزی است که قابلیت‌های مجازی‌سازی فراهم می‌کند. هایپروایزر بعنوان یک واسط بین سیستم فیزیکی، میزبان، و سیستم مجازی‌سازی شده، مهمان ، عمل می‌کند. هایپروایزرهای متفاوت نیازمند عناصر متفاوت هستند که باید روی سیستم میزبان برای فراهم‌سازی مجازی سازی، نصب شوند. بعلاوه هایپروایزرهای مختلف گزینه‌های متفاوتی را برای سیستم‌های عامل مهمان فراهم می‌کنند.

[۲۷] Hypervisors

انواع هایپروایزر

دو نوع هایپروایزر وجود دارد: نوع ۱ و نوع ۲. هایپروایزرها براساس جایشان در پشته دسته‌بندی می‌شوند. هایپروایزرهای نوع ۱ عموما مستقیما در بالای اسکلت سخت‌افزار قرار می‌گیرند. هایپروایزر نوع ۱ بعنوان سیستم عامل خودش عمل می‌کند. این به آنها امکان می‌دهد تا استفاده‌های کارآمدی از منابع سیستمی فیزیکیشان داشته باشند. به این دلیل بسیاری از محیط‌های ابری با استفاده از هایپروایزر نوع ۱ ساخته می‌شوند. هایپروایزر نوع ۲ عموما در بالای سیستم عامل دیگری قرار می‌گیرد. سیستم عامل دسترسی به سخت‌افزار فیزیکی را کنترل می‌کند. این هایپروایزر بعنوان یک سیستم کنترلی بین سیستم عامل میزبان و سیستم عامل مهمان عمل می‌کند. یکی از بزرگترین مزیت‌های هایپروایزرهای نوع ۲ این است که شما می‌توانید عموما آنها را روی سیستم دسکتاپ معمولی نصب کنید. و نیازی نیست سیستم مجزایی برای نصب هایپروایزر داشته باشید.

هایپروایزر Xen

دو ورژن از هایپروایزر Xen وجود دارد: ورژن منبع باز و ورژن تجاری پیشنهاد شده توسط Citrix، که سرورXen نام دارد. در این کتاب ما درباره‌ی سرورXen صحبت خواهیم کرد. سرورXen هایپروایزر نوع ۱ است؛ که اساسا ورژن سفارشی لینوکسی است که روی سخت‌افزار سرور شما نصب شده است. پیاده‌سازی سرور Xen شامل دو موجودیت اصلی است: هایپروایزر سرور Xen که روی سخت‌افزار سیستم[۲۸] نصب می‌شود، و کنسول مدیریت مرکز Xen که روی سیستم ویندوز نصب شده است.

[۲۸] bare-metal system: فقط سخت افزار کامپیوتر ((برنامه نویسی فلز محض)) یعنی کنترل مستقیم سخت افزار به جای تکیه بر سرویسهای سیستم عامل

شکل ۲-۴: آرم Xen Hypervisor

Hyper-V

Hyper-V یک هایپروایزر نوع ۱ است. این هایپروایزر بعد از این که سیستم عامل ویندوز نصب می‌شود فعال می‌شود، و ماشین‌های مجازی مبتنی بر آن از طریق سیستم عامل ویندوز دسترسی پیدا می‌کنند. اما واقعیت این است که هنگامی که شما Hyper-V را فعال می‌کنید، آن خودش را بین سخت‌افزار و سیستم عالم قرار می‌دهد. و مشکل در این جا است که سیستم عاملی که شما می‌بینید اساسا یک ماشین مجازی است که روی پلت‌فرم Hyper-V اجرا می‌شود.

vSphere

VMWare یک هایپروایزر نوع ۱ را پیشنهاد می‌دهد که vSphere نام دارد. این هایپروایزر در سازمان‌ها به طور گسترده‌ای استفاده می‌شود که این برای فراهم کردن یک زیرساخت مجازی و عملکرد ابری خصوصی است. اما استفاده‌ی آن در زیرساخت ابری عمومی تا حدودی باعث بوجود آمدن مشکل می‌شود که این به دلیل طبیعت خصوصی بودن آن است.

شکل ۲-۵ آرم vSphere Hypervisor

KVM

ماشین مجازی مبتنی بر هسته، که عموما با نام KVM شناخته شده است، یک هایپروایزر مبتنی بر هسته‌ی لینوکس منبع باز است. KVM از یک ماژول هسته‌ی قابل بارگذاری که kvm-ko نام دارد و مدل خاص پلت‌فرم استفاده می‌کند، یا همان kvm-intel.ko یا kvm-amd.ko. kvm سیستم‌های عامل های لینوکس و ویندوز متفاوت را برای سیستم عامل میزبان پشتیبانی می‌کند.

شکل ۲-۶: آرم KVM Hypervisor

تکنولوژی‌های توسعه‌ی وب

برنامه‌های وب از طریق اینترنت و اغلب با استفاده از مرورگر وب در دسترس هستند. برنامه‌های وب عموما به نصب کلاینت دیگری نیاز ندارند. این یکی از چیزهایی است که آنها را در سناریوهای مبتنی بر ابر جذاب می‌کند. آنها می‌توانند از هر جا قابل دسترس باشند و در بسیاری از موارد از طریق دستگاه قابل دسترس می‌باشند، زیرا که دستگاه یک مرورگر وب مناسب دارد . فروشندگان نرم‌افزارهای مستقل (ISVs) بیشتر از همه ورژن‌های مبتنی بر وب برنامه‌هایشان را پیشنهاد می‌دهند. در واقع برنامه‌های وب تبدیل به استاندارد بالفعل برای پیشنهاد برنامه شده‌اند. چندین

استاندارد و تکنولوژی باعث شده است که برنامه‌های وب به یک راه‌حل قابل اعتماد تبدیل شوند. ما در این‌جا تعدادی از آنها را پوشش می‌دهیم. بعلاوه بسیار مهم است که تکنولوژی‌های برنامه‌ی وب را در ارزیابی پلت‌فرم‌های SaaS در نظر بگیریم. از آنجایی که شما از پلت‌فرم SaaS برای توسعه‌ی برنامه‌ها استفاده می‌کنید، مهم است که شما اطمینان ایجاد کنید که پلت‌فرم SaaS انتخابی شما تکنولوژی‌هایی را که شما قصد استفاده از آنها برای پیاده‌سازی برنامه‌هایتان را دارید را پشتیبانی می‌کند.

HTML

زبان نشانه‌گذاری ابرمتنی یک استاندارد گسترده‌ای برای مدت طولانی بوده است. در واقع HTML برای ایجاد صفحات وب بهترین استاندارد است. تمام جستجوگرهای وب تفسیر صفحات وب HTML را می‌شناسند. HTML از تگ‌هایی برای فرمت و اضافه کردن ساختار به صفحات وب استفاده می‌کند. تعداد تگ‌ها و میزان عملکرد موجود در HTML رو به گسترش است. در واقع جدیدترین ورژن آن یعنی HTML۵ باعث شده است که HTML آن را در زبان برنامه‌نویسی وب بهترین ساخته است.

Adobe Flash

Adobe Flash یک زبان برنامه‌نویسی است که اصولا برای ایجاد انیمیشن و گرافیک‌های برداری مورد استفاده قرار می‌گیرد. Flash یکی از زبان‌های بسیار رایج است که روی برنامه‌های اینترنتی که به انیمیشن نیاز دارد مورد استفاده قرار می‌گیرد.

Flash به دلیل ثبات و امنیتش خیلی معروف نشد. به همین دلایل، برخی از سیستم‌ها از آن پشتیبانی نمی‌کنند. توسعه‌دهنده‌ها به دنبال راه‌های دیگری هستند که همین کارآیی را برای آنها فراهم کند. HTML۵ یکی از تکنولوژی‌هایی است که خیلی‌ها فکر می‌کنند پتانسیل جایگزینی Flash را دارد.

شکل ۲-۷: آرم Adobe Flash

SOAP

SOAP ، پروتکل دسترسی آسان به اشیاء[۲۹]، پروتکلی برای تبادل داده‌ها بین سرویس‌های وب می‌باشد. پیام‌های SOAP از تنظیم اطلاعات XML برای فرمت‌دهی استفاده می‌کنند. SOAP برای مذاکره و انتقالش به پروتکل‌های دیگر نیاز دارد. دو پروتکل لایه‌ی کاربرد که SOAP استفاده می‌کند HTTP و SMTP می‌باشد. سه خصوصیت SOAP که آن را پروتکل جالبی می‌کند، بی‌طرفی، استقلال، و گسترش آن است.

REST

REST[۳۰] یک ساختار کاربردی است. REST تبادلات برنامه را به سرور و کلاینت تقسیم می‌کند. کلاینت موجودی است که درخواست را ایجاد می‌کند و سرور موجودیتی است که به درخواست سرویس می‌دهد.

REST شش محدودیت را روی پیاده‌سازی برنامه تعریف می‌کند:

- *مدل کلاینت/سرور.* باید تمایز اکیدی از نگرانی‌ها بین کلاینت و سرور باشد.

[۲۹] Simple Object Access Protocol
[۳۰] Representational State Transfer

- *بی‌حالتی*[۳۱]: برنامه نباید مبتنی بر اطلاعات حالت در هنگام ارتباط با کلاینت باشد.
- *قابلیت جابه‌جایی:* محتویات دریافت شده توسط کلاینت باید قابلیت جابه‌جایی داشته باشند.
- *سیستم‌لایه‌ای:* کلاینت‌ها نمی‌توانند بگویند که آیا مستقیما به سرورها متصل هستند یا خیر؛ بنابراین در هنگام لزوم واسط‌هایی می‌تواند مورد استفاده قرار گیرد.
- *کد مورد تقاضا:* سرورها می‌توانند کد قابل اجرایی را به کلاینت‌ها ارسال کنند.
- *رابط یکسان یا استاندارد:* یک واسط استاندارد بین کلاینت‌ها و سرورها مورد استفاده قرار می‌گیرد.

Java

جاوا یک زبان برنامه‌نویسی شی‌گرا می‌باشد. برنامه‌های جاوا برای اجرا روی هر پلت‌فرمی طراحی شده‌اند. کد جاوا به یک زبان میانه تفسیر می‌شود که بایت‌کد جاوا نام دارد. این بایت کد سپس توسط ماشین مجازی جاوا اجرا می‌شود. از آنجایی که سیستم ورژن درستی از اجرای JVM را دارد، برنامه‌ی جاوا باید قادر به اجرا باشد.

جاوا اسکریپت

جاوا اسکریپت یک زبان برنامه‌نویسی شی‌گرا سبک وزن است. تمام ورژن های موجود مرورگر وب جاوا اسکریپت را می‌فهمد. گاهی شما خواهید دید که برای دلایل امنیتی، اجرای جاوا اسکریپت سمت کلاینت غیر فعال شده است. جاوا اسکریپت اساسا بعنوان یک زبان سمت سرور مورد استفاده قرار می‌گیرد، اما امروزه آن برای برنامه‌نویسی سمت کلاینت و سرور مورد استفاده قرار می‌گیرد. به دلیل سازگاری گسترده‌ی جاوا اسکریپت، در بسیاری از سایت‌ها و پیاده‌سازی‌های برنامه‌ی وب مورد استفاده قرار می‌گیرد.

[۳۱] Stateless

شکل ۲-۸: آرم جاوا

ASP.NET

ASP.NET یک زبان توسعه‌ی وب سمت سرور است که توسط مایکروسافت توسعه یافته است. آن به یک توسعه‌دهنده امکان ایجاد پویای صفحات را که web forms نام دارد را می‌دهد. این به معنای آن است که محتویات صفحه می‌توانند براساس خصوصیات یا الزامات مشخصی تغییر کنند. ASP.NET در بالای زمان اجرای زبان مشترک (CLR) مایکروسافت ایجاد می‌شود. CLR تدوین در لحظه‌ی برنامه‌ی نوشته شده با استفاده از هر زبان برنامه‌نویسی روی فریم کاری .NET. مایکروسافت را می‌دهد.

شکل ۲-۹: آرم Microsoft Net

Ruby on Rails

Ruby on Rails که روبی هم نامیده می‌شود، یک فریم کاری توسعه منبع باز است که می‌تواند برای ایجاد قالب‌ها، برنامه‌های توسعه، و پایگاه داده های پرس و جو بکار گرفته شود. روبی از کنترل‌کننده‌های دید-مدل یا ساختار M-V-C استفاده می‌کند. یک مدل به جدول موجود در دیتابیس نگاشت می‌شود. دید یک فایل ERB است که در زمان اجرا به HTML تبدیل می‌شود. یک کنترل‌کننده عناصری است که به درخواست‌های خارجی پاسخ می‌دهد.

شکل ۲-۱۱: آرم Ruby and Rails

JBOSS[۳۲]

JBOSS یک سرور کاربردی منبع باز است. آن برای پیاده‌سازی پلت‌فرم جاوا، نسخه‌ی شرکتی، استفاده می‌شود. JBOSS در جاوا نوشته می‌شود، که این به آن معنا است که آن می‌تواند روی سیستمی که برنامه‌های جاوا را پشتیبانی می‌کند اجرا شود.

شکل ۲-۱۲: آرم JBoss

PHP

PHP یک زبان برنامه‌نویسی و اسکریپت سمت سرور است. PHP برای صفحات خانگی شخصی است. بسیاری از زبان‌های برنامه‌نویسی سمت سرور به یک صفحه‌ی وب برای فراخوانی یک فایل مجزا نیاز دارند، اما کد PHP می‌تواند به صورت مستقیم در صفحه‌ی وب تعبیه شود.

شکل ۲-۱۰: آرم PHP

[۳۲] JavaBeans Open Source Software Application Server

JSON

نشانه‌گذاری شی جاوا اسکریپت[۳۳] (JSON) برای نمایش آرایه‌ها و ساختارها داده استفاده می‌شود. JSON به طور گسترده‌ای برای انتقال داده‌ها بین سرور و برنامه‌ی وب استفاده می‌شود. باید اشاره کنیم که اگرچه JSON از جاوا اسکریپت گرفته شده است، اما آن یک زبان مستقل است. این یکی از ویژگی‌های JSON است که آن را برای توسعه‌دهنده جذاب ساخته است.

شکل ۲-۱۳: آرم JSON

خلاصه

در برخی موارد، شما تنها به درک حداقل تکنولوژی‌های پشت ابر نیاز دارید. در موارد دیگر، شما به درک گسترده‌تری نیاز دارید. مهم است که شما بفهمید در هنگام تصمیم‌گیری درباره‌ی ارائه‌دهنده‌های ابری کدام تکنولوژی‌ها نقش دارند. اگر شما نیاز به یکپارچه‌سازی پیاده‌سازی ابری داشته باشید، درک این که شما باید کدام تکنولوژی‌ها را یکپارچه‌سازی کنید بسیار مهم است. این می‌تواند تکنولوژی‌های احراز هویت، تکنولوژی‌های رایانش، تکنولوژی‌های مجازی‌سازی یا تکنولوژی‌های توسعه‌ی وب باشد.

[۳۳] JavaScript Object Notation

فصل ۳. مدل‌های استقرار ابر

نکات این فصل

- **ابرهای عمومی**
- **ابرهای خصوصی**
- **ابرهای گروهی**
- **ابرهای ترکیبی**

مقدمه

NIST چهار مدل استقرار ابری را تعریف می‌کند: ابرهای عمومی، ابرهای خصوصی، ابرهای گروهی[۳۴] و ابرهای هیبریدی. یک مدل استقرار ابری با توجه به جایی که زیرساخت برای استقرار قرار می‌گیرد و این که چه کسی کنترل آن زیرساخت را میزبانی می‌کند، تعریف می‌شود. انتخاب این که کدام مدل استقرار را شما انتخاب می‌نمایید بسیار اهمیت دارد.

هر مدل استقرار ابری نیازهای سازمانی متفاوتی را تصدیق می‌کند، بنابراین مهم است که شما مدلی را انتخاب کنید که الزامات مدل شما را تصدیق کند. مهم‌تر از آن این است که هد مدل استقرار ابری گزاره ارزش متفاوت و هزینه های مختلف مرتبط با آن دارد. از این رو، در بسیاری از موارد، انتخاب شما در مدل استقرار ابری باعث کاهش هزینه‌ها می‌شود. در بسیاری از موارد، برای این که قادر به تصمیم‌گری درست باشیم، باید نسبت به خصوصیات هر محیط آگاه باشیم.

[۳۴] community clouds

ابرهای عمومی

ابرهای عمومی محیط‌هایی هستند که کاملا توسط یک ارائه‌دهنده‌ی سرویس خارجی مدیریت و سرویس‌دهی می‌شوند. در هنگامی که بیشتر افراد درباره‌ی ابرهای کامپیوتری فکر می‌کنند، آنها ابرهای عمومی هستند. در واقع بیشتر مقالات و ابزارهایی که شما می‌یابید مربوط به ابرهای عمومی می‌باشند. این بخاطر این است که محیط‌های ابری اولیه ابرهای عمومی بوده‌اند. ابرهای عمومی هنوز در محیط‌های ابری بیشترین استقرار را دارند.

مزیت‌ها

تعداد پیاده‌سازی‌ها ابری عمومی به دلیل مزایای زیادی که ابرهای عمومی پیشنهاد می‌دهند رو به رشد است. گزاره‌ی ارزش برای یک پشنهاد عمومی بسیار قوی است، اگرچه که ایرادهایی وجود دارد، که آنها را بررسی خواهیم کرد.

۱-دسترسی یا در دسترس بودن [۳۵]

استقرارهای ابر عمومی دسترسی افزایش یافته‌ای را پیشنهاد می‌دهد . هر سازمانی تعریفی از حد دسترسی دارد که مایل است به آن دست یابد. هم‌چنین هر سازمانی تعریفی از حد دسترسی دارد که قادر است به آن برسد. گاهی اوقات این دو با هم تطبیق دارند؛ و گاهی نه. مشکل این است که دسترسی هزینه‌بر است، چه هزینه‌ی نرم‌افزاری و چه هزینه‌ی سخت‌افزاری و هزینه‌ی کارمندان. هر کدام از این هزینه‌ها باشد، سازمان قادر به تهیه‌ی آن نخواهد بود، بنابراین آنها باید با آنچه که دارند کار خود را انجام دهند و از این رو قادر به دستیابی به سطح دسترسی مورد نظر خود نخواهند بود. بسیاری ارائه‌دهنده‌های ابر عمومی نرم‌افزار، سخت‌افزار و کارکنان را در دسترس دارند تا پیشنهادشان دسترسی بالایی داشته باشد. آنها ممکن است اندکی هزینه‌ی سرویسشان برای ارائه بیشتر باشد که این برای دسترسی بیشتر است، اما هرگز مانند هزینه‌ی انجام داخلی آن نخواهد بود. با این حال، به دلیل این که شما یک ارائه‌دهنده‌ی سرویس ابر عمومی را انتخاب می‌کنید، شما نباید دسترسی بالا و تحمل خطای بالایی را در نظر داشته باشید. شما باید از ارئه‌دهنده سوال کنید که به همراه سرویس چه چیزی ارائه می شود. اگر افزایش دسترسی قابل اضافه شدن باشد، شما باید در هنگام محاسبه‌ی

[۳۵] Availability

هزینه آن را بدانید. هم‌چنین شما باید اطمینان ایجاد کنید که دسترسی ایده‌آل شما بخشی از توافق سطح سرویس (SLA) شما می‌باشد. SLA شما می‌تواند به شما سطح تضمینی را بدهد که نیاز دسترسی شما ممکن است برآورده شود.

آگاه باشید که اگرچه ابرهای عمومی می‌توانند دسترسی شما را افزایش دهند، اما شما باید اطمینان ایجاد کنید که شما می دانید که چه چیزی دسترس خواهد بود. این بسته به پیشنهاد سرویس است. در پیشنهاد SaaS، برنامه به تنهایی دسترس خواهد بود. اما در پیشنهاد PaaS و IaaS، اگرچه پلت‌فرم یا زیرساخت ممکن است دسترس باشد، اما برنامه دسترس نخواهد بود. مسائل کاربردی با استفاده از یک پیشنهاد IaaS یا PaaS کاهش نخواهد یافت.

۲-مقیاس‌پذیری

پیاده‌سازی‌های ابری عمومی یک ساختار به شدت مقیاس‌پذیر را پیشنهاد می‌دهد. چیزی که پیاده‌سازی ابر عمومی پیشنهاد می‌دهد و ابرهای خصوصی آن را ندارند، توانایی مقیاس توانایی‌های سازمان شما بدون مجبور به ایجاد زیرساخت شخصیتان است. پیاده‌سازی‌های ابر عمومی می‌توانند ظرفیت بارگیری موقت یا ظرفیت دائمی را پیشنهاد دهند، که این بسته به این است که سازمان شما به چه چیزی نیاز دارد. اگر سازمان شما از سرویس SaaS استفاده کند، شما می‌تواند کاربران را بدون اضافه کردن زیرساخت مرتبط اضافه کنید. اگر از سرویس IaaS یا PaaS استفاده کنید، شما برای ایجاد برنامه‌ها و سرویس‌هایتان ظرفیت افزایش یافته خواهید داشت، اما شما هنوز نیاز به اطمینان این دارد که برنامه‌هایی که ایجاد کرده‌اید بار افزایش یافته را مدیریت می‌کنند.

۳-دستیابی پذیری [۳۶]

ارائه‌دهنده‌های ابر عمومی اهمیت زیادی را به در دستیابی پذیر بودن می‌دهند. برای افزایش مشتری، آنها تلاش در اطمینان از این دارند که می‌تواند به کلاینت‌های مختلفی می‌توانند سرویس دهند. هدف آنها اطمینان از این است که سرویس‌های آنها می‌تواند توسط هر دستگاهی روی اینترنت بدون نیاز برای VPNها یا هر نرم‌افزار کلاینت دیگری در دسترس باشد. امروزه افراد به اینترنت و برنامه‌های مبتنی بر اینترنت تنها از طریق مرورگرها سنتی روی لپ‌تاپ‌ها و کامپیوترها دسترسی ندارند. افراد

[۳۶] Accessibility

انتخاب‌های فراوانی را برای استفاده از مرورگرهای وب دارند. تبلته و گوشی‌های هوشمند استفاده‌ی زیادی دارند. اگرچه دستگاه‌های جدید مرورگر وب دارند، آنها مرورگرهای وب کامل نمی‌باشند. بنابراین برای داشتن توانایی پشتیبانی این دستگاه‌ها، برنامه‌ها و صفحات وب باید تا حدودی ساده‌سازی شوند و باید پایبند به استانداردهای توسعه باشند. پشتیبانی چندین سیستم عامل و مرورگر وب بسیار پرهزینه است. هزینه‌های تضمین توسعه و کیفیت می‌تواند به شدت زیاد باشد. بنابراین اگرچه بسیاری از سازمان‌ها می‌خواهند این نوع پشتیبانی را برای کاربران فراهم کنند، اما این هزینه‌ی زیادی دارد. هرچند از آنجایی که ارائه‌دهنده‌های سرویس بیشتر روی پیشنهاد یک تنظیم از سرویس‌ها تمرکز دارند، آنها بیشتر متمایل به پذیریش چنین هزینه‌هایی هستند.

٤-کاهش هزینه

ابرهای عمومی به دلیل هزینه‌ی کمی که دارند جذاب می‌باشند. اما شما باید در نظر بگیرید که این هزینه‌ی کم شاید آنقدرها هم که مشا تصور می‌کنید خوب نمی‌باشد. شما نه تنها باید درباره‌ی این کم بودن هزینه درک مناسبی داشته باشید بلکه باید درباره‌ی نوع این صرفه‌جویی‌ها هم اطلاعات داشته باشید. با استفاده از یک ابر دیگر سازمان‌ها نباید نگران صرف هزینه برای استقرارهای سخت‌افزاری و نرم‌افزاری باشند. مشتری تنها برای سرویسی که استفاده می‌کند هزینه می‌دهد. بیشتر هزینه‌های پیش‌پرداخت هزینه‌های عمده هستند زیرا این در واقع هزینه‌ای است که برای خرید سخت‌افزار است. هم‌چنین کاهش هزینه در نگهداری و پشتیبانی هم وجود دارد و البته هزینه‌های محیطی هم کاهش می‌یابد. از آنجایی که سرورها در مرکز داده‌ی شما قرار ندارند، شما فضا، هزینه‌های خنک کردن و برق را نیز کاهش می‌دهید. در واقع اگر شما تمام برنامه‌های خود را برون‌سپاری کنید، شما اصلا به مرکز داده نیاز نخواهید داشت. اما واقعیت این است که تعداد کمی از سازمان‌ها می‌توانند فعالیت‌های IT خود را برون‌سپاری کنند.

معایب

پیاده‌سازی‌های ابر عمومی معایب و محدودیت‌هایی را دارد. بسیاری از این‌ها می‌توانند بخاطر این باشند که زیرساخت برای سازمان دیگری است و توسط آن سازمان کنترل می‌شود.

۱- محدودیت‌های یکپارچه‌سازی

در ابرهای SaaS عمومی، سیستم ها خارج از سازمان شما هستند؛ یعنی داده‌ها نیز در خارج سازمان شما هستند. قرار دادن داده‌ها در یک مکان خارجی برای شما در هنگام گزارش‌دهی و یا حرکت به سیستم‌های درون‌سازمانی برای شما مشکل ایجاد می‌کند. اگر شما نیاز به گزارش‌ها یا انجام آنالیز هوش تجاری داشته باشید، شما می توانید به انتقال داده ها از طریق اینترنت پایان دهید. این نگرانی‌هایی را در مورد کارآیی و مسائل امنیتی برای شما ایجاد می‌کند. در هنگامی که گزارش‌ها در مکان یکسانی با داده‌ها ایجاد می‌شوند، آنها به سرعت تحویل داده می‌شوند.

یکپارچه‌سازی برنامه نیز می‌تواند در پیشنهادهای SaaS عمومی مشکل‌سازی باشد. در یک شرایط ایده‌آل، برنامه‌های متفاوتی می‌توانند از عملکرد به اشتراک‌گذاشته‌ای استفاده کنند. نیازی نیست که شما عملکرد یکسانی را در دو برنامه‌ی مختلف تکرار کنید. بنابراین اگر عملکردی در یک برنامه باشد، شما به برنامه‌ی دیگری نیاز دارید که باید قادر به فراخوانی آن عملکرد در یک برنامه‌ی دیگر باشد. این در برنامه‌های ابر عمومی یک مشکل است. ارائه‌دهنده‌ی برنامه باید APIها و سرویس‌های وبی را ارائه دهد که مشتری می‌تواند به برای انجام این کار از آنها استفاده کند. در غیر این صورت شما ممکن است در وضعیتی قرار بگیرید که در آن عملکرد آن تکرار شود.

۲-انعطاف‌پذیری کاهش یافته

هنگامی که شما از یک ارائه‌دهنده‌ی ابر عمومی استفاده می‌کنید، شما در معرض زمان ارتقاء آن ارائه‌دهنده قرار دارید. در بیشتر موارد، در هنگامی که ارتقادهی انجام می‌شود شما اصولا تحت تاثیر قرار نخواهید گرفت. بسیاری از ارائه دهندگان تمایلی به نصب چندین نسخه از یک برنامه یا سیستم آنلاین ندارند. چنین کاری باعث افزایش سربار اجرایی آنها می‌شود. کاربران در سیستم جدید آموزش داده می‌شوند، که این در بهره‌وری تاثیرگذار است.

۳- از کار افتادگی اجباری[۳۷]

در هنگامی که شما از یک ارائه‌دهنده‌ی ابر عمومی استفاده می‌کنید، در هنگامی که سیستم نگهداری و تعمیر آفلاین می‌شود، ارائه‌دهنده آن را کنترل می‌کند. تعمیر و نگهداری ممکن است در زمانی

[۳۷] Forced Downtime

انجام شود که برای شما و سازمان شما ناخوشایند است. بسته به این که سیستم چگونه تقسیم‌بندی می‌شود، شما قادر به به تعویق انداختن تعمیر برای یک دوره‌ی کوتاه و توافق روی زمانی که برای سازمان و ارائه‌دهنده مناسب است، خواهید بود. با این حال تعمیر و نگهداری نمی‌تواند برای یک مدت طولانی به تعویق بیافتد.

مسئولیت‌ها

با ابرهای عمومی، بیشتر مسئولیت‌ها گردن ارائه‌ی دهنده‌ی سرویس است. ارائه‌دهنده مسئول نگهداری و پشتیبانی است. ارائه‌دهنده هم‌چنین مسئول ایجاد اطمینان از این است که پرسنل به خوبی آموزش دیده‌اند. در ابر عمومی، ارائه‌دهنده‌ی سرویس مسئول تمام عناصر مورد نیاز برای پیاده‌سازی سرویس می‌باشد. این عناصر بسته به سرویس پیشنهاد شده متفاوت می‌باشند. آنها می توانند شامل سرورها، برنامه‌های کاربردی، مخازن و داده‌ها باشند. در یک ابر عمومی، مصرف‌کننده مسئول هر چیزی است که برای مصرف سرویس مورد نیاز است. تعدادی استثنا وجود دارد، مانند پیاده‌سازی‌ها در یک برنامه‌ی کلاینت سرور که درگیر آن است. مصرف‌کننده مسئول نصب برنامه‌های سمت کلاینت و اطمینان از این است که آن به خوبی عمل می‌کند. ارائه‌دهنده‌ی سرویس مسئول توسعه‌ی برنامه‌ی سمت کلاینت و پیشنهاد پشتیبانی برای بدست آوردن بهترین عملکرد است.
مشتری مسئول نگهداری کلاینت عمومی است. مشتری باید اطمینان ایجاد کند که بروزرسانی‌های لازم و برنامه‌های لازم روی سیستم کلاینت نصب شده است. هم‌چنین مشتری مسئول فراهم کردن اتصال اینترنتی به ارائه‌دهنده است.

ملاحظات امنیتی

اطمینان از امنیت در سناریوهای ابر عمومی بسیار دشوار است. از آنجایی که شما دسترسی به سیستمی را که سرویس‌ها را فراهم می‌کند را به درستی مدیریت نمی‌کنید، اطمینان از این که آنها ایمن هستند بسیار دشوار است. شما باید در این مورد کاملا به راهنمایی‌ها ارائه‌دهنده گوش دهید و به قابلیت‌های آن اطمینان داشته باشید.

داده‌ها

ارائه‌دهنده‌های ابر عمومی مسئله واقعی در مورد امنیت داده ها را افزایش می دهد. درباره‌ی مالکیت داده‌ها سوالاتی است. از آنجایی که ارائه‌دهنده‌ی سرویس مالک سیستم‌هایی است که داده‌ها شما در آن قرار دارد، ارائه‌دهنده می‌تواند مالک بالقوه‌ی داده‌ها در نظر گرفته شود.
هم‌چنین درباره‌ی دسترسی به داده‌ها نیز مسائلی وجود دارد. از لحاظ تئوری هر کسی که در ارائه‌ی‌دهنده‌ی سرویس کار می‌کند می‌تواند دسترسی به داده‌های شما داشته باشد.

پذیرش/انطباق

پذیرش یا انطباق می تواند یکی از بزرگترین نگرانیهای ارائه دهندگان سرویس ابر عمومی باشد. چه باید کرد با این حقیقت که شما دید کمی نسبت به آنچه در پشت صحنه اتفاق می افتد دارید. در اکثر موارد، شما باید از امکانات یا ابزار ارائه دهنده که با آنها سازگار است، استفاده کنید. ارائه دهنده ممکن است گواهینامه SAS-۷۰ را داشته باشد بدون امکان اینکه شما قادر باشید خودتان آن را بررسی کنید. شما مجبورید به بررسی کننده SAS اعتماد کنید که به اندازه کافی آن را بررسی کرده است.

حسابرسی

در مورد ارائه‌دهنده‌های سرویس ابر عمومی، شما قابلیت‌های حسابرسی محدودی را خواهید داشت. شما دسترسی مستقیم به حساب‌ها یا سیستم‌های مدیریت رویداد را نخواهید داشت. در بیشتر موارد شما قادر به پیاده‌سازی هشدار عقبه یا ثبت نام و ورود خود را نخواهید داشت. پس شما باید به آنچه که ارائه‌دهنده تامین می‌کند تکیه کنید. بسیاری از ارائه‌دهنده‌های ابری عمومی به شما امکان دسترسی به برخی فرم‌های ثبت برنامه را می‌دهند. این ثبت‌ها می‌توانند برای دیدن دسترسی کاربر و تصمیم‌گیری با توجه به مجوزدهی مورد استفاده قرار گیرند.

ابرهای خصوصی

ابرهای خصوصی کاملا توسط سازمان شما مدیریت و نگهداری می‌شوند. عمومی تمام زیرساخت مورد نیاز برای محیط شما در یک مرکز داده‌ای که شما آن را کنترل می‌کنید قرار می‌گیرد. بنابراین شما مسئول خرید، نگهداری و پشتیبانی هستید.

بسیار از افراد از ابر چنین فهمی را دارند که باور این که ابرهای خصوصی واقعا ابر هستند برایشان دشوار است. آنها فکر می‌کنند که فقط ابرهای عمومی ابر واقعی هستند. اما اگر شما به خصوصیات ابر توجه داشته باشید، متوجه خواهید شد که اهمیتی ندارد که ابر در کجا قرار گرفته است. گزاره‌ی ارزش ابر در هنگامی که شما درباره‌ی ابرهای خصوصی بعنوان مخالف ابرهای عمومی صحبت می‌کنید تغییر می‌کند؛ اما گزاره‌ی ارزش تعیین نمی‌کند که آن ابر است یا خیر.

مزیت‌ها

مدل‌های ابر خصوصی مزیت‌های زیادی دارند. بیشتر این مزیت‌ها در حول توانایی شما برای کنترل و نظارت چیزی که در محیط ابر اتفاق می‌افتد قرار دارند.

۱-پشتیبانی و عیب‌یابی

محیط‌های ابرهای خصوصی نسبت به ابرهای عمومی عیب‌یابیشان آسان‌تر است. در یک محیط ابر خصوصی شما دسترسی مستقیم به تمام سیستم‌ها دارید. شما می‌توانید به ثبت‌ها دسترسی داشته باشید، اجرای ردیابی شبکه، ردیابی اشکال زدایی را داشته باشید و یا می‌توانید دوره های بسیار سریعتر را فراهم کنید که به حفظ رضایت مشتری کمک می کند. در نهایت رضایت مشتری برای نگهداری موفقیت محیط شما اهمیت دارد.

۲-نگهداری

به وسیله‌ی ابرهای خصوصی، شما می‌توانید چرخه‌ی ارتقاء را کنترل کنید. در هنگامی که شما تمایلی به ارتقاء ندارید این کار انجام نخواهد شد. اگر ورژن جدید ویژگی‌ها و عملکردهایی را که شما می‌خواهید نداشته باشد شما مجبور به ارتقاء نخواهید بود. اگر سازمان شما زمانبندی خاصی را برای ارتقاء و نگهداری و تعمیر دارد شما می‌توانید آن زمان این کارها را انجام دهید. این به کاهش تاثیر قطعی سیستم کمک می‌کند. در برخی نمونه‌ها، شما ممکن است نیاز به اجرای چندین ورژن از یک برنامه را داشته باشید که این می‌توانید برای سازگاری باشد. اگر شما سیستم‌ها را کنترل نکنید، شما قادر به داشتن دسترسی به چندین ورژن از برنامه نخواهید داشت. با ابر داخلی، شما در اجرای چندین ورژن از یک برنامه آزاد هستید. این انعطاف‌پذیری به شما توانایی افزایش یافته برای خدمت‌دهی به نیازهای مشتری را می‌دهد.

۳-نظارت کردن

از آنجایی که شما در محیط ابر خصوصی خود دسترسی مستقیم به سیستم دارید، شما قادر به هر انجام هر نظارتی که می‌خواهید هستید. شما می‌توانید هر چیزی را از برنامه‌ها تا سیستم‌های سخت افزاری نظارت کنید. یک مزیت بزرگ این قابلیت این است که شما می‌توانید مقیاس‌های پیشگیرانه‌ای را برای جلوگیری از قطعی بکار گیرید، بنابراین شما در سرویس‌دهی به مشتریان فعال‌تر می‌باشید.

معایب

اگرچه این که شما به کل محیط کنترل دارین بسیار خوب است اما مشکلاتی هم دارد. وقتی شما یک محیط ابر خصوصی را پیاده‌سازی می‌کنید، شما با مشکلاتی مواجه می‌شوید که مانند مشکلات پیاده‌سازی راه‌حل داخلی سنتی است. شما باید این مشکلات را هم در نظر بگیرید و سپس تصمیم‌گیری کنید که آیا ابر داخلی برای شما انتخاب مناسبی است.

۱-هزینه

پیاده‌سازی یک ابر خصوصی نیازمند هزینه‌هایی است. شما مجبور به پیاده‌سازی زیرساختی هستید که نه تنها می‌تواند نیازهای جاری شما را پشتیبانی کند بلکه می‌تواند نیازهای آینده شما را نیز پشتیبانی کند. شما باید نیاز تمام بخش‌های کسب‌وکار را که آنها را پشتیبانی خواهید کرد را تخمین بزنید. هم‌چنین شما باید زیرساختی را پیاده‌سازی کنید که می‌تواند زمان‌های اوج را نیز پشتیبانی کند. تمام سیستم هایی که باید زمان‌های اوج را پشتیبانی ککند همواره مجبور به اجرا نیستند (البته اگر راهی برای اجرای آنها به صورت اتوماتیک در هنگام نیاز باشد).

۲-سازگاری نرم‌افزاری و سخت‌افزاری

شما باید اطمینان ایجاد کنید که نرم‌افزاری که پیاده‌سازی کرده‌اید با سخت‌افزار محیط شما سازگاری دارد. بعلاوه شما باید اطمینان ایجاد کنید که نرم‌افزاری که شما پیاده‌سازی کرده‌اید با کلاینت‌های محیط شما نیز سازگاری دارد. نمونه‌هایی وجود دارند که در آنها شما به سخت‌افزارهای خاصی نیاز دارید- برای مثال مخزن- که برای پیاده‌سازی برنامه‌ی خاصی مورد نیاز است.

۳-تخصص مورد نیاز

با ابرهای خصوصی شما به متخصصانی در تمام برنامه‌ها و سیستم‌هایی که شما می‌خواهید آنها را پیاده‌سازی کنید نیاز دارید. نیاز به متخصصان داخلی منجر به آموزش پرهزینه‌ای است. شما مسئول نصب، نگهداری و پشتیبانی آنها می‌باشید، بنابراین شما باید اطمینان ایجاد کنید که شما دانش انجام آن را دارید و یا توانایی استخدام کارمندان و پیمانکاران خارجی را دارید. ایجاد یک محیط ابری نیازمند کارمندانی با دانش سخت‌افزاری، ذخیره‌سازی، شبکه، امنیت و مجازی‌سازی است. پیدا کردن کارمندانی که تمام این دانش‌ها را دارا هستند بسیار دشوار است. بعلاوه سازمان شما به افرادی نیاز دارد که تخصص در پلت‌فرم ابری خاصی دارند که شما می‌خواهید آن را پیاده‌سازی کنید.

مسئولیت‌ها

در محیط ابر خصوصی، تقسیم مسئولیت‌ها ساده است. سازمان شما مسئول راه‌حل پایان به پایان[۳۸] است. شما مسئول سیستم‌هایی هستید که سرویس، برنامه‌های ملاینت، و نگهداری سیستم‌های کلاینت را فراهم می‌کند.

ملاحظات امنیتی

با پیاده‌سازی ابر خصوصی، سازمان شما کنترل کاملی روی کل سیستم، برنامه‌ها، و داده‌ها دارد. شما می‌توانید کنترل کنید هر کسی به چه چیزی دسترسی دارد. اطمینان از امینت در یک محیط ابری خصوصی ساده‌تر است. کنترل کامل روی سیستم وجود دارد و شما می‌توانید هر ابزار امنیتی را که می‌خواهید پیاده‌سازی کنید.

در محیط ابر خصوصی، شما قادر به اجرای حسابرسی‌های امنیتی و انطباق خود هستید. این به شما اطمینان بیشتری را در دانستن این که سیستم‌ها شما به امنیت و انطباق مورد نیاز می‌رسد را می‌دهد.

انطباق/پذیرش

در یک محیط ابر خصوصی، شما مسئول اطمینان از این می‌باشید که تمام قوانین انطباق رعایت می‌شوند.

[۳۸] end-to-end

اگر سازمان شما مهارت و توانایی اطمینان پایبندی به مقررات انطباق را دارد، داشتن اخلی سیستم و داده‌ها مزیت بزرگی است. اگر این چنین نباشد و شما مهارت‌ها و تکنولوژی‌های لازم را ندارید،باید آنها را بدست آورید در غیر این صورت با مشکلات بزرگی مواجه خواهید شد.
این که داده‌ها و سیستم‌های شما در یک مکان خارجی قرار گرفته باشند با انطباق می‌تواند به شرکت شما کمک کند.شما می‌توانید به ارائه‌دهنده‌های سرویس در فراهم کردن توانایی‌ها و خبره‌های مورد نیاز خود اعتماد کنید. انطباق صنعت کارت پرداخت[۳۹] (PCI) نمونه‌ی خوبی است. انطباق PCI نیازمند ملاحظات خاصی است که باید برای هر سیستمی که اطلاعات کارت اعتباری را پردازش می‌کند در نظر گرفته شود. این در ساده‌سازی برخی الزامات روی برخی سیستم‌های داخلی کمک می‌کند. با این حال شما باید احتیاط داشته باشید. شما نمی‌توانید به طور کامل به ارائه‌دهنده‌ی ابر اعتماد کنید. اگر مسائل امنیتی و انطباق وجود داشته باشد، از شرکت شما شکایت می‌شود و یا حداقل به شهرت شما آسیب رسیده می‌شود. خیلی از افراد بین شما و ارائه‌دهنده تمایز قائل نمی‌شوند. آنها شما را به دلیل انتخاب ارائه‌دهنده‌ی نامناسب سرزنش خواهند کرد.

داده‌ها

در محیط ابری خصوصی، شما صاحب داده‌ها و سیستمی هستید که داده‌ها را در خود قرار می‌دهند. این به شما اجازه‌ی کنترل بیشتر روی افرادی که می‌توانند به داده‌ها دسترسی داشته باشند و کارهایی که می‌توانند روی داده‌ها انجام دهند را می‌دهد.

حسابرسی

در محیط ابر خصوصی، شما دسترسی کامل به تمام برنامه‌ها و ورودی‌های سیستم دارید. شما می‌توانید ببینید که هر کس به چه جیزی دست یافته و با آن چه کاری را انجام داده است. بزرگترین مزیت این است که شما می‌توانید همه‌ی این‌ها را بی‌درنگ ببینید، بنابراین شما قادر خواهید بود تا اقدام صحیح مورد نیاز برای اطمینان از یکپارچگی سیستم خود را بکار گیرید.

[۳۹] Payment card industry

ابرهای گروهی

ابرهای گروهی مانند ابرهای عمومی و خصوصی زیاد مورد استفاده قرار نمی‌گیرند؛ در واقع آنها مدلی استقرار ابری هستند که حداقل استفاده و معروفیت را دارند. در یک ابر گروهی، ابر توسط یک گروه از سازمان‌ها که هدف مشخصی دارند به اشتراک گذاشته می‌شود.

مزایا

استفاده از این ابرها مزیت‌های فراوانی دارد. بسیاری از آنها به این خاطر هستند که زیرساخت و هزینه به اشتراک گذاشته می‌شود.

۱-هزینه

در ابر گروهی، هزینه بین اعضای گروه به اشتراک گذاشته می‌شود. این هزینه‌ی به اشتراک گذاشته شده به خرید زیرساختی که هر کدام از سازمان‌ها به تنهایی نمی‌توانستند آنها را تهیه کنند می‌انجامد. به این صورت اعضای گروه به صرفه‌ی اقتصادی بیشتری دست می‌یابند. اما شما باید دقت داشته باشید، چون مشکلاتی در زمینه‌ی این که هرکس هزینه‌ی چه چیز را باید پرداخت کند بوجود می‌آید. و مشکلاتی در زمینه‌ی این که هر عنصر زیرساخت برای چه کسی است نیز وجود دارد. در هنگام شروع تمام این مسائل باید به دقت بررسی شوند.

۲-چند مستاجری

در ابر گروهی، چند مستاجری در بدست آوردن مزیت‌های اقتصادی تاثیرگذار است. سازمان شما به تنهایی ممکن است آنقدر بزرگ نباشد که از برخی صرفه‌های مالی مزایا ببرد، اما با کار با یک سازمان دیگر یا چندین سازمان، شما را آنقدر که از این مزایا استفاده کنید بزرگ می‌کند. در ابر گروهی، چند مستاجری هم‌چنین به شما امکان به اشتراک‌گذاری فعالیت‌های پشتیبانی و نگهداری را می‌دهد. به جای این که نیاز باشد یک سازمان تمام مهارت‌ها برای نگهداری و پشتیبانی محیط را داشته باشد، هر سازمان می‌تواند در نواحی که در آن خبره است مشارکت داشته باشد.

معایب

معایب بالقوه‌ای برای پیاده‌سازی ابر گروهی وجود دارد. هر بار که شما چندین سازمان را دارید که با هم کار می‌کنند، امکان درگیری وجود دارد. گام های برای پیشگیری از این موضوع باید در نظر گرفته شود.

۱-مالکیت

مالکیت در پیاده‌سازی ابر گروهی باید به طور واضحی تعریف شود. اگر چندین سازمان در کنار هم قرار می‌گیرند تا زیرساخت را فراهم کنند، شما باید توافق‌هایی را برای مالکیت مشترک تعیین کنید. در برخی نمونه ها، سازمان‌ها در کنار هم قرار می‌گیرند تا ابر گروهی را ایجاد کنند که ممکن است یک سازمان مشترک را برقرار کند که می‌تواند مالک منابع باشد.

مسئولیت ها

در ابر گروهی، مسئولیت‌ها بین سازمان‌ها به اشتراک گذاشته می‌شود. ممکن است مشکلاتی در این که هر کس مسئول چه کاری است بوجود آید، اما بعد این که مشخص شد، این کار کاملا مزایامند است. این باعث کاهش مسئولیت مدیریتی روی هر سازمان می‌شود.

ملاحظات امنیتی

ابرهای گروهی یک تنظیم خاصی را در مورد امنیت ارائه می‌دهد زیرا چندین سازمان هستند که کنترل و دسترسی به محیط را دارند.

داده‌ها

در ابر گروهی، تمام مشارکت‌کننده‌ها در گروه به داده‌ها دسترسی دارند. به این علت، شما نمی‌خواهید داده‌هایی را که محدود به سازمان شما است را ذخیره کنید. اگر چنین کاری را کنید ممکن است مشکلاتی برای شما بوجود آید.

انطباق

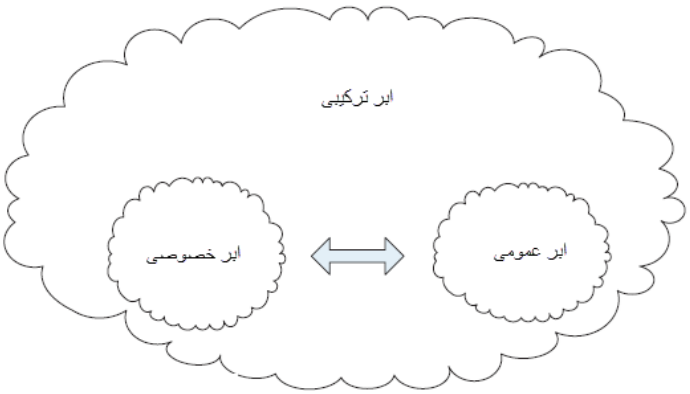
در ابر گروهی، هر سازمانی که تا حدودی آشنا با مقررات انطباق است، مسئول آن می‌باشد.

حسابرسی

در ابر گروهی، سازمان‌های عضو به تمام برنامه‌های و سیستم‌های به اشتراک گذاشته شده دسترسی دارند. شما ممکن است توافق‌هایی را برای مشخص کردن این که هرکس باید چه فعالیتی را انجام دهد بخواهید.

ابرهای هیبریدی

با کامل شدن رایانش ابری در گذر زمان، ابرهای هیبریدی به رایج‌ترین پیاده‌سازی ابری تبدیل خواهند شد. تصور جزئی نادرستی درباره‌ی این که ابر هیبریدی چیست وجود دارد. بسیاری از افراد تصور می‌کنند که ابر هیبریدی یک محیط ابری است که برخی عناصر خصوصی و برخی دیگر عمومی هستند. این نادرست است. یک محیط ابر هیبریدی، که در شکل ۳,۱ نیز آن را می‌بینید، ابری است که در آن چندین محیط ابری مجزا با هم متصل شده‌اند. ابرهای هیبریدی آزادی در پیاده‌سازی هرچیزی که برای رسیدن به نیازهای سازمان ضروری است را می‌دهند اما این ابرها می‌توانند در پیاده‌سازی بسیار گران و پیچیده باشند.



شکل ۳-۱: محیط ابر ترکیبی یا هیبریدی

مزایا

علاه بر مزایایی که از طریق هر ابر بدست می‌آید، مدل ابر هیبریدی انعطاف‌پذیری افزایش یافته‌ای نیز دارد. اگر هدف نهایی شما این است هر چیزی را به ارائه‌دهنده‌ی ابر عمومی انتقال دهید، ابر هیبریدی به شما امکان حرکت به سوی یک محیط ابری را می‌دهد که در آن شما مجبور به عمومی‌کردن چیزی تا زمانی که نخواهید نیستید. شما ممکن است برنامه‌ی مشخصی داشته باشید که پشنهاد سرویس عمومی برای آن گران است. شما می‌توانید این برنامه‌ها را به صورت داخلی تا زمانی که هزینه کاهش می‌یابد نگهداری کنید. هم‌چنین ممکن است شما نگرانی‌هایی را حول امنیت انتقال یکسری داده‌ها خاص به ارائه‌دهنده‌ی ابر عمومی داشته باشید. مدل ابر هیبریدی به شما امکان می‌دهد تا این داده‌ها را به صورت داخلی نگهداری کنید تا زمانی که مطمئن شوید آن در محیط ابر عمومی ایمن است.

سازمان‌های زیادی از مدل ابر هیبریدی استفاده می‌کنند زیرا که این مدل‌ها تلورانس خطا و در دسترسی بالایی دارد. شما می‌توانید برنامه‌های خاصی را داشته باشید که در دو محیط قرار دارند. در این صورت اگر یک محیط دچار ایرد شود، شما هنوز به آن برنامه دسترسی دارید.

معایب

یک محیط ابر هیبریدی می‌تواند پیچیده‌ترین محیط برای پیاده‌سازی باشد.ملاحظات متفاوتی با توجه به این که شما می‌خواهد چه نوع ابری را پیاده‌سازی کنید وجود دارد. تمام پروسه‌ها و قوانین شما به تمام محیط‌ها اعمال نخواهد شد. شما باید تنظیم‌های متفاوتی از قوانین و پروسه‌ها را برای هر محیط توسعه دهید.

۱- یکپارچه‌سازی

ممکن است برنامه‌ها وجود داشته باشند که به داده‌های یکسانی نیاز دارند. در این حالت شما باید یکی از موارد زیر را انتخاب کنید: شما می‌توانید دو کپی از داده‌ها را تهیه کنید، که این شما را ملزم به تنظیم برخی مکانیزم‌های تکثیر برای نگهداری داده‌ها در همگام سازی می‌کند، و یا شما می‌توانید داده‌ها را بعنوان انتقال دهید. انتقال داده‌ها در یک محیط ابر مشکلاتی دارد و شما باید نگران محدودیت‌های پهنای باند باشید.

ملاحظات امنیتی

ابرهای هیبریدی ملاحظات امنیتی خاصی دارند. شما نه تنها باید نگران مسائل امنیتی باشید بلکه باید نگران مسائل ایجاد شده بر اثر اتصال چند محیط هم باشید.

داده‌ها

حرکت داده ها در محیط‌های ابری بسیار خطرناک است. شما باید اطمینان ایجاد کنید که تمام محیط‌های درگیر داده‌های ایمن رضایت‌بخشی دارند. داده‌ای که مدام حرکت دارد تامین ایمنی بودنش دشوار است. هر دو طرف یک ارتباط باید پروتکل‌های امنیتی یکسانی را بکار گیرند، و باید با هم سازگار باشند.

حسابرسی

حسابرسی در محیط‌های ابر هیبریدی دشوار است. دسترسی کاربر به صورت داخلی و خارجی تغییر می‌کند. دنبال کردن یک فرآیند از شروع تا پایان ممکن است شما را به سیستم داخلی و خارجی ببرد. شما باید ثبت وقایع مرتبط را داشته باشید به طوری که بتوانید این وقایع داخلی و خارجی را با هم تطبیق دهید.

خلاصه

NIST چهار مدل استقرار ابری را بیان کرده است عمومی، خصوصی، گروهی و هیبریدی. ابرهای عمومی برای عموم باز است. ابرهای خصوصی برای یک سازمان مشخص هستند. ابرهای گروهی توسط چندین ساطمان به اشتراک گذاشته می‌شوند. ابرهای هیبریدی محیط‌هایی برای ایجاد یک ترکیب از مدل‌های ابری می‌باشند. هر مدل مزیت‌ها، ایرادها و پیامدهای امنیتی خودش را دارد.

فصل ۴. مدل‌های سرویس ابری

نکات این فصل:

- نرم‌افزار بعنوان سرویس (اجاره‌ی نرم‌افزار)
- پلت‌فرم بعنوان سرویس (اجاره‌ی پلت‌فرم)
- زیرساخت بعنوان سرویس (اجاره‌ی زیرساخت)
- مدل‌های سرویس دیگر

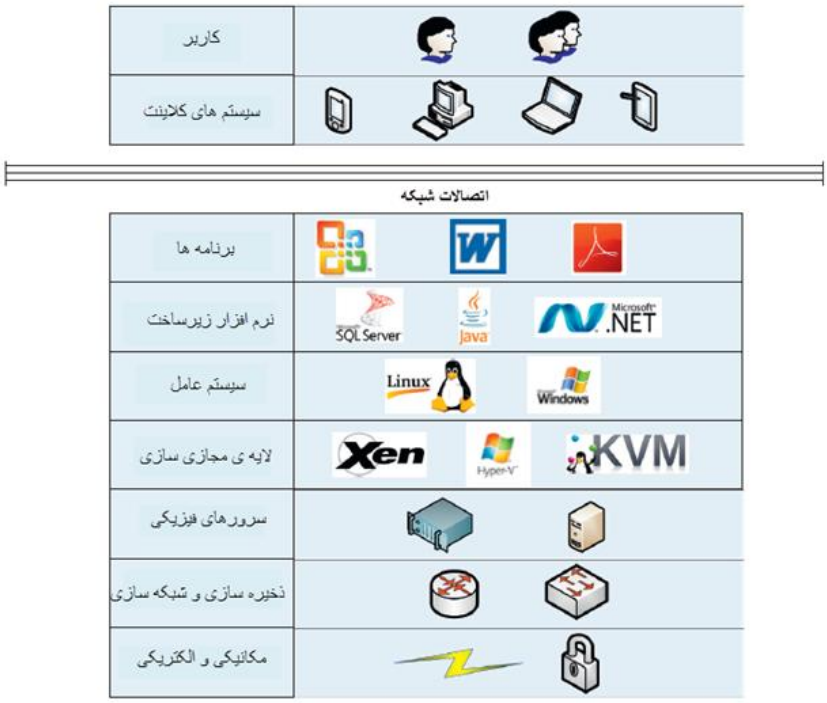
مقدمه

با توجه به تعریف مؤسسه ملی استاندارد و فناوری (NIST) از ابر، سه سرویس ابر اصلی وجود دارد: نرم‌افزار بعنوان سرویس، پلت‌فرم بعنوان سرویس و زیرساخت بعنوان سرویس. اما یک چیز را نباید فراموش کرد و آن این است که از آنجایی که ما با ارائه‌دهنده‌های سرویس سروکار داریم، باید برای همه‌چیز مذاکره شود. سرویس‌های موجود تغییر می‌کنند و سرویس‌های جدیدی برای برآورده کردن نیازهای مشتری اضافه می‌شود. و با رشد بازارهای ابری، باید نسبت به سرویس‌های جدید هم دانش داشته باشیم. ما برخی از این سرویس‌ها را در این مقاله پوشش می‌دهیم. هر مدل سرویس شاخص‌ها و خصوصیات منحصر به خودش را دارا است. شکل ۴,۱ یک بررسی کلی ار سرویس‌های مرتبط به کامپیوتر را نشان می‌دهد. در هر مدل سرویس، می‌بینیم که آنها دوباره کل پشته را تجزیه می‌کنند.

نرم‌افزار بعنوان سرویس

بسیاری از افراد SaaS را مدل ابری ارجینال می‌دانند. مدل SaaS شبیه به مدل ارائه‌دهنده‌ی سرویس برنامه است (ASP). اما یکسری تفاوت‌های کلیدی وجود دارد. اول این که در مدل ASP

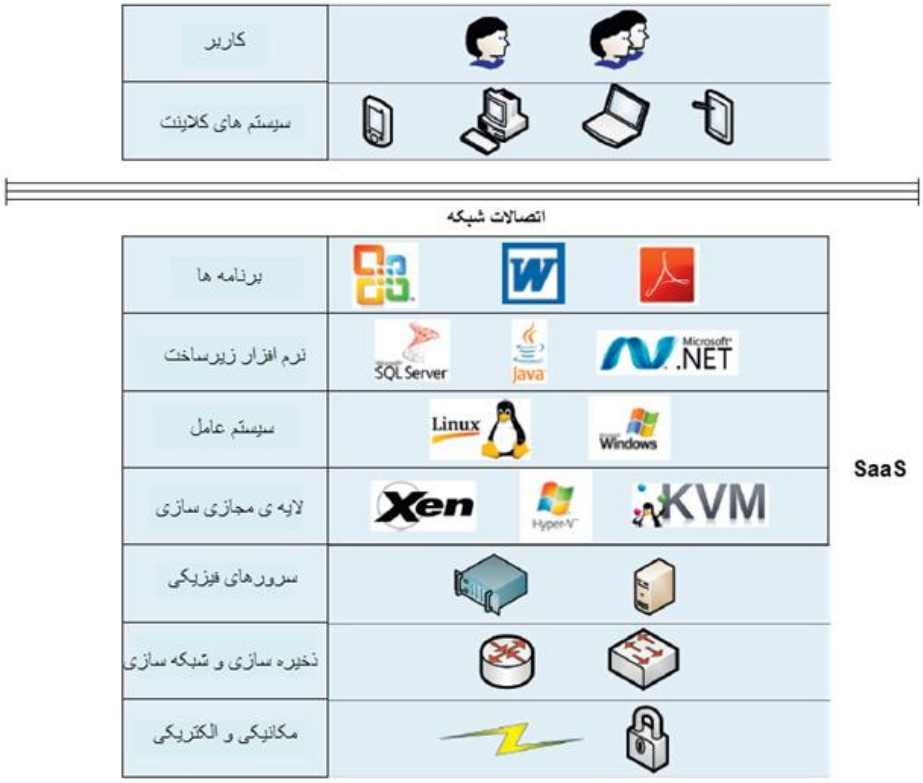
برنامه‌ها اغلب برنامه‌های کلاینت/سرور بودند. برخی از کلاینت‌ها و زیرساخت‌های خاص اغلب برای دسترسی به برنامه‌ها مورد نیاز بود. اما بیشتر برنامه‌های SaaS امروزی مبتنی بر وب هستند و به دسترسی به برنامه نیاز ندارند. این باعث ساده‌سازی فرآیند دسترسی به برنامه‌ها می‌شود. بعلاوه، در مدل ASP، مشتریان اغلب به برنامه‌های یکسانی دسترسی دارند؛ افرازهای نسبتا ساده‌ای از برنامه وجود دارد.



شکل ۴-۱: پشته ی سرویس های مبتنی بر کامپیوتر

خصوصیات SaaS

بسته به ارائه دهنده ی سرویس و سرویس پیشنهاد شده، خصوصیات ممکن است متفاوت باشند، اما ما در این‌جا یکسری از خصوصیات عمومی و رایج را پوشش می‌دهیم.



شکل ۴-۲: سرویس های SaaS

سفارشی‌سازی

با پیاده‌سازی SaaS ، ارائه‌دهنده‌ی سرویس اغلب به صورت مجازی همه‌چیر درباره‌ی برنامه را کنترل می‌کند. در بسیاری از موارد، آن محدود به هر سفارشی‌سازی می‌شود که می‌تواند انجام گیرد. اما بسته به پیاده‌سازی، شما ممکن است قادر به درخواست این باشید که واسط کاربری کمی تغییر کند. اغلب تغییرات عمده امکان پذیر نیست. در بیشتر موارد، مشتری قادر به ایجاد تغییرات نمی‌باشد؛ ارائه‌دهنده باید تغییرات را ایجاد کند. در محیط SaaS، امکان سفارشی‌سازی می‌تواند برای ارائه‌دهنده‌ی سرویس بسیار پرهزینه باشد و هم‌چنین برای مشتری. امکان سفارشی‌سازی گسترده به معنای میزبانی یک نمونه‌ی مجزا از برنامه تنها برای یک مشتری خاص است. داشتن سفارشی‌سازی

گسترده می‌تواند در هنگام ارتقاء نرم‌افزار نیز مسائلی را بوجود آورد. احتمالش زیاد است که سفارشی‌سازی در هنگام ارتقاء از بین برود. سپس باید توسط مشتری یا ارائه‌دهنده مجددا ایجاد شود. این نیازمند زمان و هزینه‌ی زیادی است.

پشتیبانی و نگهداری

در محیط SaaS، ارتقاءهای نرم‌افزار توسط ارائه‌ی دهنده‌ی سرویس متمرکز و انجام می‌شود. شما نباید نگران ارتقاء نرم‌افزار روی چندین کلاینت باشید. ارتقاءهای متمرکز شده امکان ارتقاءهای متناوب بیشتری را می‌دهند، که آنها امکان تحویل سریع را می‌دهند. استثنای این قانون زمانی است که نرم‌افزار کلاینتی وجود دارد که برای دسترسی به برنامه‌های متمرکز شده مورد استفاده قرار می‌گیرد. اما ارائه‌دهنده‌های SaaS در تلاش برای دسترسی به برنامه‌ها بدون نیاز به برنامه‌ی کلاینت هستند. ارتقاءهای متمرکز منجر به یک مساله می‌شوند. هنگامی که ارائه‌دهنده زمان ارتقاء را انتخاب می‌کند، شما در این مورد هیچ کمکی ندارید. در ابتدا اگر زمان خرابی مرتبط با ارتقاء وجود داشته باشد، شما باید آن را بپذیرید. بعلاوه این ارتقاء ممکن است نیازمند آموزش کاربران بیشتری باشد، بنابراین شما کاربران خود را آموزش دهید. این باعث خرابی یا کاهش سازندگی می‌شود.

آنالیزها

آمار مصرف و آنالیزها می‌تواند اطلاعاتی با ارزشی را درباره‌ی مصرف برنامه بکار گیرد. در پیاده‌سازی‌های SaaS، ارائه‌دهنده توانایی دیدن فعالیت‌های کاربر و تعیین گرایش‌ها را دارد. برای سازمان‌ها بزرگ، این اطلاعات می‌تواند ارزشمند باشد. از آنجایی که بیشتر محیط‌های ابری پرداخت به ازای هر ورود را پیشنهاد می‌دهند، درک گرایش‌های کاربر مهم است. درک گرایش‌ها باعث افزایش مصرف و درنتیجه باعث افزایش هزینه می‌شود.هم‌چنین مهم است که مصرف کلی و همزمان نیز درک شود. ممکن است شما قادر به کاهش هزینه‌ها مجوز باشید.

یکپارچه‌سازی

در SaaS، داده‌های در سایت ارائه‌دهنده ذخیره می‌شوند. در بیشتر موارد، مشتری دسترسی مستقیم به داده‌ها ندارد. این در هنگام گزارش‌دهی و کسب‌وکار هوشمند مساله‌ساز است. هم‌چنین اگر شما نیاز به تعمیر دستی داده‌ها یا لود کردن و یا مجدد لود کردن داده‌ها داشته باشید نیز مساله‌ساز است. در برخی موارد شما هیچ‌کاری در این باره نمی‌توانید انجام دهید. در برخی پیاده‌سازی‌ها شما

امکان حرکت داده‌ها و انتقال آنها را بین SaaS و سیستم‌های سازمانی داخلی خود دارید. در هنگام این نوع عملیات شما باید به پهنای باندی که مورد استفاده قرار می‌گیرد توجه داشته باشید. در این موارد شما باید هزینه‌هایی را به ارائه دهنده‌ی سرویس و ارائه‌دهنده‌ی اینترنت پرداخت کنید.

مسئولیت‌ها

در پیاده‌سازی‌های SaaS، بیشتر مسئولیت‌ها برگردن ارائه‌دهنده‌ی سرویس است. این یکی از دلایلی است که پیاده‌سازی‌های SaaS را معروف کرده است. سازمان‌ها قادر هستند تا منابع داخلیشان را برای انجام فعالیت‌های دیگر آزاد سازند. شکل ۴،۳ به شما ایده‌ای درباره‌ی مسئولیت کلی ارائه‌ی دهنده‌ی سرویس و وظایف مشتری می‌دهد.

شکل ۴-۳: مسئولیت های SaaS

در محیط SaaS، ارائه‌دهنده عموما مسئول همه‌چیز جز سیستم‌های کلاینت است. آن اطمینان ایجاد می‌کند که برنامه بروز است. اطمینان ایجاد می‌کند که برنامه به درستی سرهم‌بندی شده[۴۰] است. اطمینان ایجاد می‌کند که داده‌ها به خوبی ذخیره شده‌اند. آن سیستم را برای کارآیی و ایجاد هرگونه تنظیمی که مورد نیاز است کنترل می‌کند. در محیط SaaS، مشتری مسئول سیستم کلاینت است . مشتری باید اطمینان ایجاد کند که کلاینت به برنامه‌ی SaaS اتصال دارد. تمام نرم‌افزارهای لازم باید در سیستم کلاینت نصب شود. سیستم‌های کلاینت باید در سطح مناسبی اصلاح شده باشند.

درایورهای SaaS

درایورهای زیادی در رشد پیشنهادهای SaaS عمومی مشارکت داشته‌اند. رشد زیادی در ایجاد و مصرف برنامه‌های مبتنی بر وب وجود دارد. بیشتر ارائه‌های دهنده‌های SaaS سرویس‌های خود را به صورت برنامه‌های مبتنی بر وب پیشنهاد می‌دهند. بنابراین با پذیرش رشد برنامه‌های مبتنی بر وب، سرویس‌های SaaS نیز پذیرفته می‌شوند. نحوه‌ی دید به برنامه‌های تغییر کرده است و هم‌چنین کیفی و راحتی در توسعه‌ی برنامه‌ها نیز توسعه یافته است. کامل شدن پلت‌فرم‌های قدیمی‌تر و معرفی پلت‌فرم‌های جدید تنوع زیادی را در ابزارهایی که می‌توانند برای ایجاد برنامه‌های وب قوی بکار گرفته‌شود را ایجاد کرده است. برخی از این ابزارها HTML۵، JavaScript، CSS، Ruby on Rails و PHP هستند.

چالش‌های SaaS

اگرچه امروزه SaaS معروف‌ترین مدل سرویس ابری است، اما هنوز چالش‌هایی در انتخاب SaaS وجود دارد. ارائه‌دهنده‌های SaaS قادر به حل مجدد بسیاری از این چالش‌ها و نگرانی‌ها بوده‌اند، اما هنوز چالش‌هایی وجود دارد، که در بخش‌های بعدی توصیف می‌شود.

مکان های (موقعیت های) مختلف

برنامه‌های SaaS در یک مکان خارجی قرار دارند. این یعنی این که ارتباط‌های بین کلاینت و برنامه باید از طریق اینترنت عمومی صورت گیرد، که گاهی این مسیر طولانی است. این فاصله‌ی زیاد باعث

[۴۰] patch

بوجود آمدن تاخیر در محیط می‌شود. این یک فاکتور محدودکننده برای برخی برنامه‌ها می‌باشد. برخی برنامه‌ها نیازمند پاسخ در میلی‌ثانیه می‌باشند. این برنامه‌ها در محیط‌هایی که تاخیر زیادی وجود دارد کار نمی‌کنند.

چند مستاجری

چند مستاجری باعث بروز مشکلاتی می‌شود. از آنجایی که برنامه به اشتراک گذاشته می‌شود، عموما سفارشی‌سازی کمی انجام‌پذیر است. اگر سازمان شما به سفارشی‌سازی گسترده‌ای نیاز داشته باشد این می‌تواند یک مشکل باشد. شاید شما باید برنامه‌ی درون‌سازمانی را انتخاب کنید.
چند مستاجری هم‌چنین منجر به مشکلات امنیتی نیز می‌شود. زیرا مشتریان به یک نمونه از برنامه دسترسی دارند، و جریان برنامه ممکن است به یک مشتری این امکان را بدهد که به داده‌های مشتری دیگری دست یابد. ارائه‌دهنده‌های SaaSنسبت به این مساله هوشیار می‌باشند و در صورت بروز آن را رفع می‌کنند.

دیگر چالش‌های امنیتی

یکی از بزرگترین نگرانی‌هایی که سازمان‌ها با SaaS دارند درباره‌ی امنیت داده‌ها می‌باشد. کارمندان ارائه‌ی دهنده‌ی سرویس دسترسی مستقیم به سیستم‌هایی دارند که داده‌ها در آنها قرار گرفته است. یک راه برای کاهش این مساله حفاظت از داده‌ها در سطح نرم‌افزاری است. شاید نیاز باشد که شما داده ها خود را رمزگذاری کنید. بنابراین از امکان خواندن داده‌ها توسط کارمندان جلوگیری می‌شود.

ارائه‌دهنده‌های SaaS

ارائه‌دهنده‌های SaaS زیادی وجود دارد. در این‌جا ما برخی از آنها را بررسی می‌کنیم.

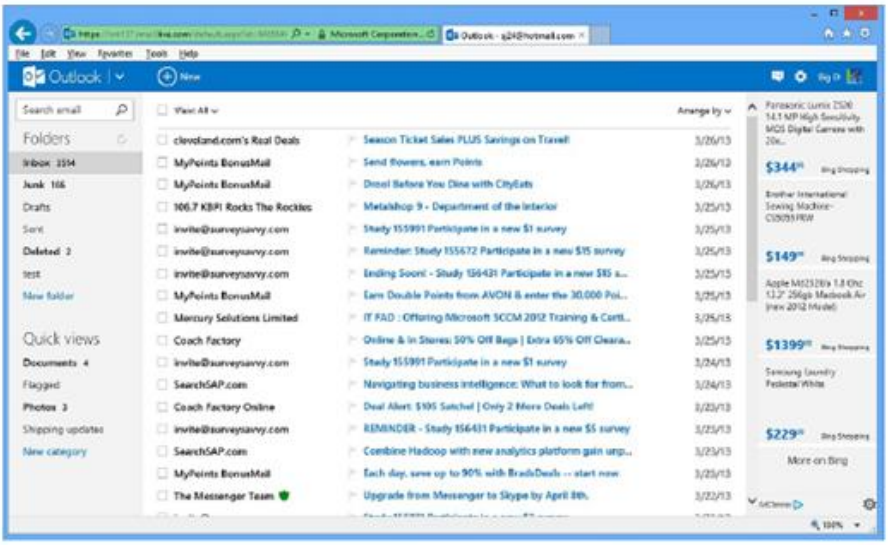
Outlook.com

ایمیل‌های مبتنی بر وب یکی از پیشنهادهای SaaS معروف است. برای مدت طولانی ارائه‌دهنده‌های زیادی ایمیل مبتنی بر وب را پیشنهاد داده‌اند. بیشتر ارائه‌دهنده‌ها سرویس رایگان و پولی را پیشنهاد می‌دهند. Outlook.com که در شکل (۴-۴) نشان داده شده است یک سرویس ایمیل مایکروسافت است که جانشین Hotmail و Live Mail است. یک ایمیل Outlook.com پیش فرض رایگان است. اما اگر شما ویژگی‌های پیشرفته‌ای را نیاز داشته باشید که شامل تبلیغات نیست، شما باید حساب ایمیل خود را ارتقاء دهید. این با انتخاب آیکون gear در گوشه‌ی بالا سمت راست و انتخاب

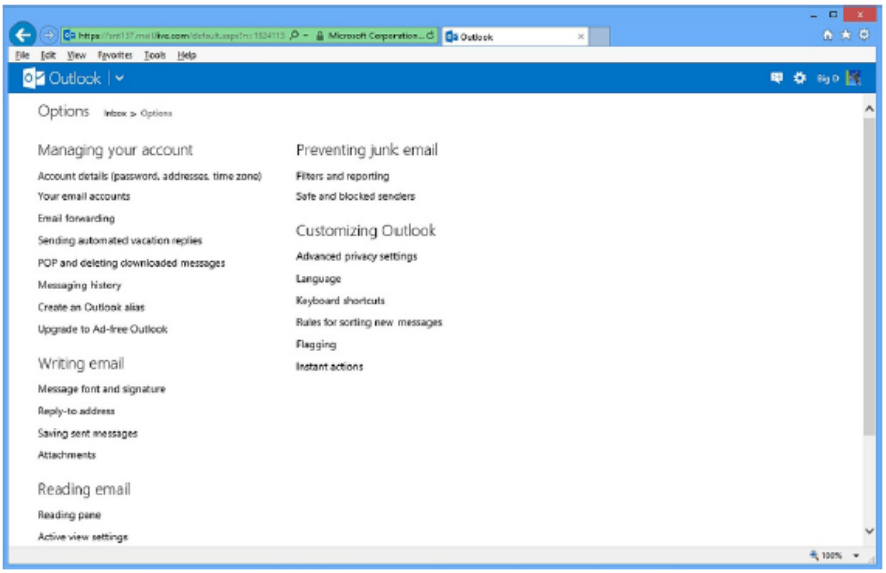
تنظیمات بیشتر Mail امکان‌پذیر است. صفحه‌ی گزینه‌های Inbox که در شکل (۴-۵) نشان داده شده است ظاهر می‌شود. در این‌جا شما گزینه‌ای برای ارتقاء Ad-free Outlook دارید. این شما را به سایت مایکروسافت می‌برد، که در آن شما می‌توانید دسترسی به ورژن‌های ارتقاء یافته برنامه را داشته باشید.

Google drive

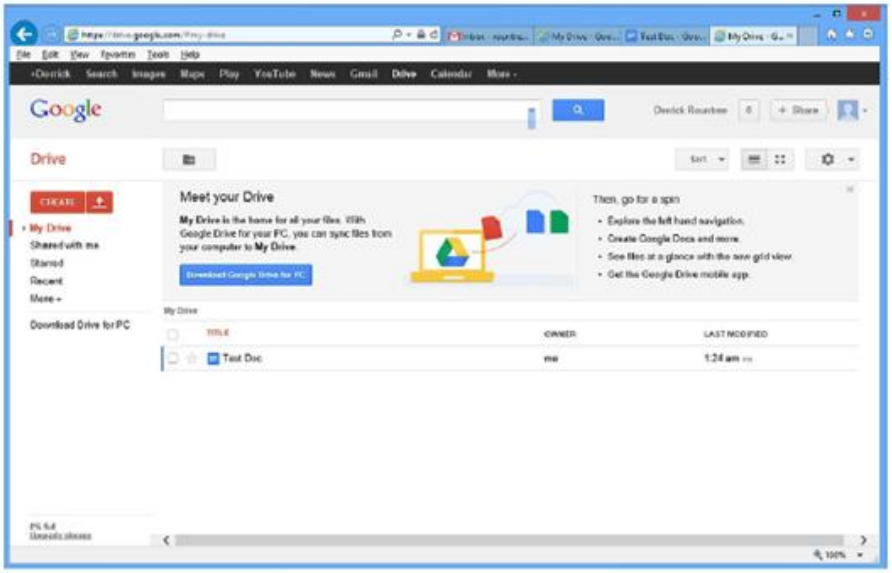
Google drive که در شکل (۴-۶) نشان داده شده است، به شما دسترسی آنلاین برای دیدن و ایجاد داکیومنت‌های پردازش متن، خواندنی ها، سخنرانی ها و چندین اسناد دیگر را می‌دهد. شما می‌توانید از یک نوع داکیومنت built-in استفاده کنید و یا نوع جدیدی را اضافه کنید. برای اضافه کردن انواع جدید داکیومنت، Creat را انتخاب کنید (سمت چپ) و Connect More Apps را انتخاب کنید. این صفحه‌ی Connect Apps را می‌آورد که در شکل (۴-۷) است.



شکل ۴-۴: چشم انداز ایمیل



شکل ۴-۵: چشم انداز گزینه های اینباکس ایمیل



شکل ۴-۶: google Docs

شکل ۴-۷: پنجره ی اتصال برنامه ها

Salesforce.com

Salesforce.com یک برنامه‌ی مدیریت ارتباط با مشتری شناخته شده می‌باشد که برای پردازش‌های سیستمی کسب‌وکار مرتبط با کارمندان، امور مالی و ارسال است. برنامه‌ی CRM یک مجموعه از جریان‌های کاری می‌باشد که با نرم‌افزاری به مدیریت فعالیت‌ها و اطلاعات مربوط به مشتری کمک می‌کنند. این فعالیت‌های می‌توانند مربوط به فروش‌ها (مانند استفاده از اطلاعات مشتری برای ایجاد راهنمایی در آینده)، کارهای بازاریابی (مانند استفاده از داده‌های فروش برای توسعه‌ی استراتژی فروش)، و یا فراهم‌سزای سرویس مشتری بهتر باشند. Salesforce.com لیست

جامعی از ویژگی‌ها برای هر سه نوع این فعالیت ایجاد می‌کند. در این بخش روی ویژگی‌های Salesforce.com برای پشتیبانی مشتری بعنوان یک مورد مطالعاتی در SaaS تمرکز داریم.

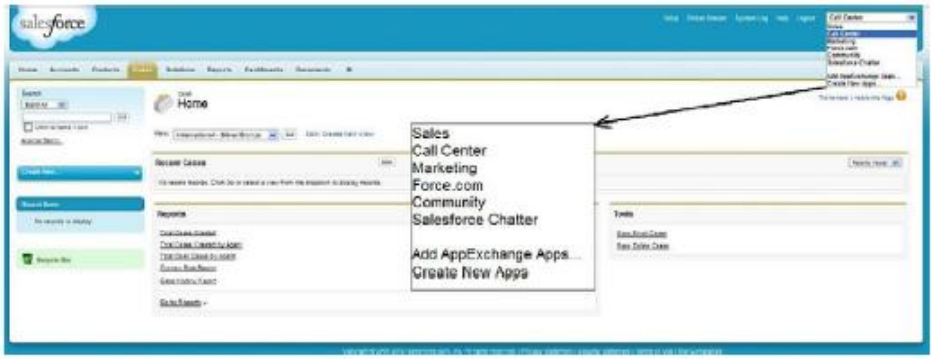
درک یک ویژگی

پیش از این که یک کسب‌وکار بتواند از Salesforce.com استفاده کند، یک فاز تنظیم کوچکی وجود دارد که کاربر کسب‌وکار می‌تواند Salesforce.com را با توجه به نیازهای کسب‌وکاری خودش سفارشی‌سازی کند. این در ابتدا شامل تعریف حساب Salesforce.com است. دوما مدیری سیستم باید داده‌های مشتریان فعلی را به Salesforce.com انتقال دهد، برخی اسکرین‌های Salesforce.com را سفارشی‌سازی کند، و دسترسی‌هایی را برای اسکرین‌های مناسب تعریف کند که این برای بکارگیری کسب‌وکار است. توصیفی که در ادامه داریم جزئیات این تنظیم را ذکر نمی‌کند و فرض می کند که پیکربندی لازم انجام گرفته است.

بعد از این که پورتال Salesforce.com ایجاد شد، نماینده‌های پشتیبانی مشتری می‌تواند وارد شوند و به صفحه‌ی وب مرکز فراخوانی[۴۱] بروند (در شکل E۴-۱) نشان داده شده است. این صفحه شامل توابعی برای مدیریت درخواست‌های مشتری است، مانند ضبط تماس‌های مشتری، اتخاذ مورد برای پشتیبانی پرسنل و جستجو برای راه‌حل‌ها. این صفحه‌ی وب شامل تعدادی تب است. شکل (E۴-۱) تب‌های را نشان می‌دهد، که این در پشتیبانی نماینده‌ها برای ردیابی و مدیریت شکایات مشتری کمک می‌کند. قابل رویت است که صفحه‌ی وب به ما امکان جستجو برای یک مورد خاص را می‌دهد، و یا جستجو برای یک موردی که اخیرا اتفاق افتاده است، ایجاد گزارش‌های کارآمد مانند تعداد کل موارد و ... . ابزار Mass Email به ما امکان ارسال ایمیل به ایمیل آیدی را که مربوط به هر مورد است را می‌دهد. این فیلدهای پیش‌فرض ببرای یک مورد می‌توانند با کلیک روی Create New bar که در سمت چپ صفحه‌ی وب است دیده شوند. با این کهر یک صفحه باز می‌شود (شکل (E۴-۲)) که می‌تواند توسط پشتیبان پرسنل مورد استفاده قرار گیرد. فیلدهای Contact Name و Account Name می‌توانند توسط جستجوی در دیتابیس Contacts و Accounts یافت شوند. بسیاری از این فیلدها، برای مثال، Priority و Case Origin، مقادیر هستند که از یک منو انتخاب می‌شوند. در Salesforce.com این را picklist (لیست انتخاب) می‌گوییم. فیلدهای اضافی می‌توانند به رکورد مورد توسط مدیر اضافه شوند، بنابراین سفارشی‌سازی این صفحه با توجه به نیازهای هر شرکت است.

[۴۱] Call Center Web page

توجه: برای تست عملکرد، می‌توانید به سایت www.salesforce.com رفته و برای حساب رایگان ثبت نام کنید.



شکل (E۴-۱): Salesforce.com

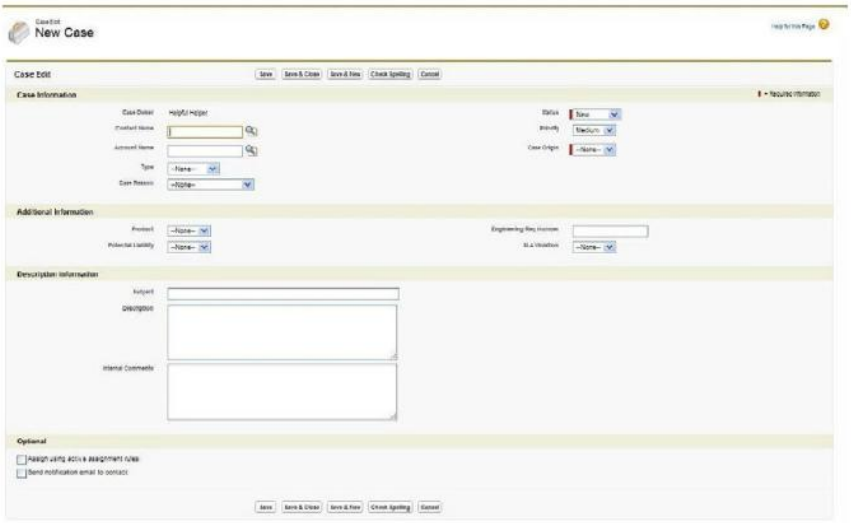
تب‌های دیگر روی صفحه شامل توابع جالبی هستند که برای مدیریت تماس‌های کارمندان به مشتریان کارآمد است. برای مثال تب Solution دسترسی به دیتابیسی را که شامل راه‌حل‌های اخیر برای مساله‌های مشتری بوده است، می‌باشد. این دیتابیس قابل جستجو است، و به کارمندان این امکان را می‌دهد تا به سرعت مساله‌های مشتری را حل کنند. لیست کامل تب‌ها با کلیک روی علامت "+" قابل پیدا کردن است. مدیر می‌تواند تب‌ها را روی هر صفحه قابل رویت کند. صفحات وب بازاریابی و فروش شامل توابعی کارآمد برای فروش و بازاریابی هستند. و مانند صفحه‌ی مرکز تماس می‌باشند. بعلاوه صفحات Community و Salesforce Chatter Web امکان پیام‌رسانی، انجمن ها و دیگر انواع همکاری بین کاربران را می‌دهد. می‌توان دید که واسط برنامه درخواست کسب‌وکار عمومی طراحی شده است و از این رو می‌تواند برای استفاده در یک کسب‌وکار جدید سفارشی‌سازای شود.

تب Add App Exchange App (شکل E۴-۴) کاربران را قادر به بسط عملکرد Salesforce.com می‌کند. پورتال AppExchange و تب Create New App به کاربران امکان ایجاد برنامه‌های جدید را می‌دهد و به آنها دانلود رایگان و خرید از طریق AppExchange را پیشنهاد می‌دهد. دسترسی به این تب‌ها می‌تواند توسط مدیر تحت کنترل باشد. ویژگی‌های

پیشرفته این پلت‌فرم می وانند با استفاده از لینک Force.com در دسترس قرار گیرند، که این در بخش بعدی توصیف شده است.

هنگامی که مورد جدیدی ایجاد می‌شود، می‌توان روی ID کلیک کرد تا جزئیات بدست اید. این صفحه هم‌چنین شامل فیلدهایی برای اختصاص کار به یک سازمان، تنظیم مهلت و غیره است.

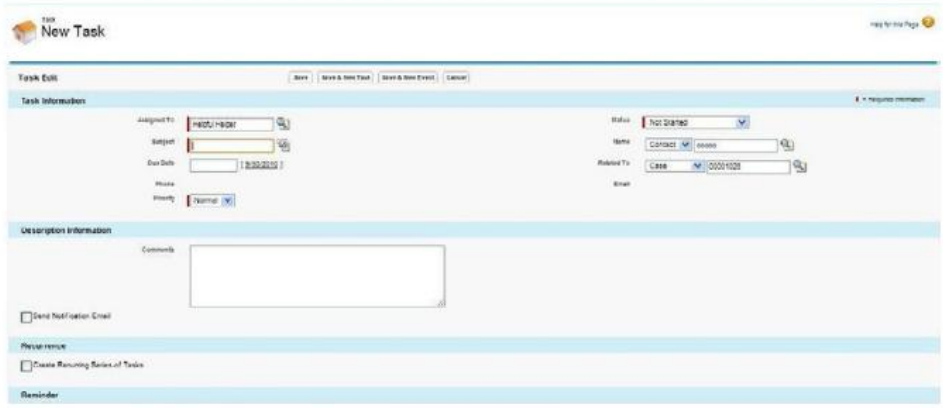
لازم نیست که همواره موارد را به صورت دستی وارد کنیم. Salesforce.com ویژگی‌هایی برای ایجاد موارد به صورت اتوماتیک از وب را دارد. برای ایجاد موارد به صورت اتوماتیک از یک صفحه‌ی وب سلف‌سرویس، مدیر می‌تواند یک اسکریپت وب را ایجاد کند که این با استفاده از برنامه‌ی Salesforce.com که ویژگی‌های پیشرفته‌ی دیگری برای کمک به نمایندگان پشتیبانی مشتری دارد انجام می‌گیرد. برای مثال موارد می‌توانند به صورت اتوماتیک با استخراج فیلدها از ایمیل مشتری ایجاد شوند. هم‌چنین ویژگی‌هایی برای پشتیبانی نرم‌افزارها، تیم‌های موردی که شامل کارمندان با نقشهای مختلف ، ایجاد سلسله‌ها است. جزئیات این ویژگی‌های پیشرفته می‌توانند در لینک زیر یافت شوند: https://na۳.salesforce.com/help/doc/user_ed.jsp?loc=help



E4.2
Salesforce.com: New Case screen.

شکل (E۴-۲): صفحه نمونه جدیدSalesforce.com

تاکنون درباره‌ی ویژگی‌های استاندارد و صفحات وب Salesforce.comصحبت کردیم. با این حال کسب‌وکارها تمایل به سفارشی‌سازی Salesforce.com دارند تا آن برای فرآیندهای کسب‌وکار آنها مناسب باشد. این ویژگی بسیار مهمی برای پشتیبانی چند مستاجری در برنامه‌ی SaaS است. یک خلاصه‌ای جزئیات و سفارشی‌سازی‌های مهم ارائه شده در ادامه ارائه داده می‌شوند.

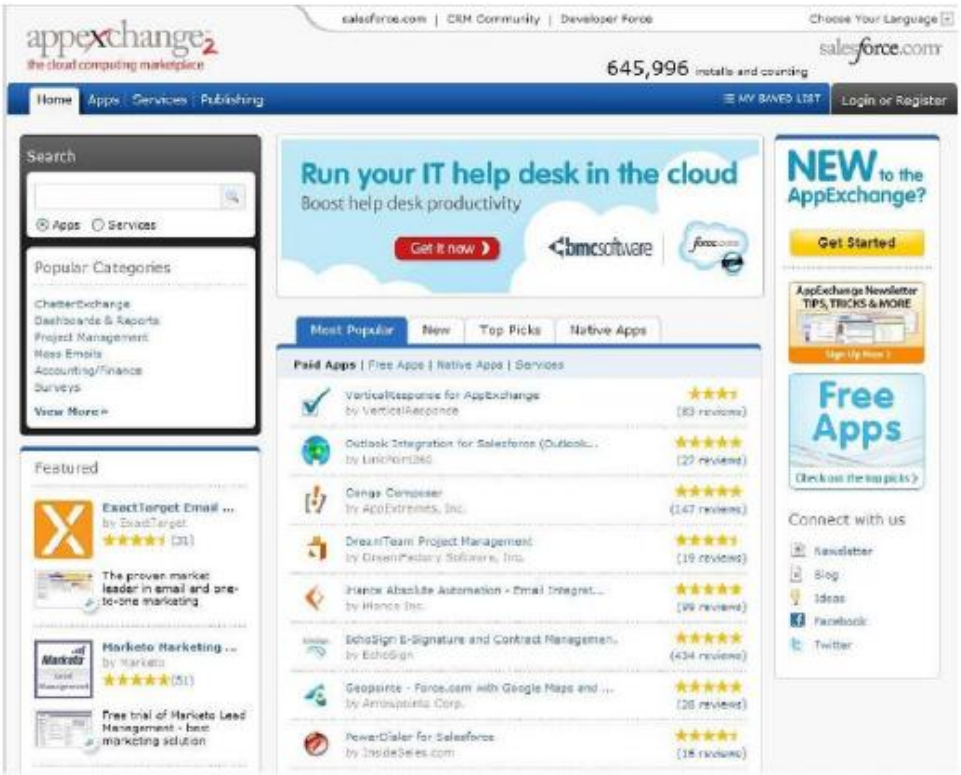


E4.3
Salesforce.com new task screen.

شکل (E۴-۳) : صفحه وظیفه جدید در Salesforce.com

نکته: سفارشی‌سازی برنامه

- تغییر نام فیلدها
- تنظیم شرط برای بروزرسانی فیلدها
- تنظیم شرط برای هشدار به ایمیل
- سفارشی سازی UI



E4.4

Salesforce.com AppExchange.

شکل E۴-۴: فروشگاه Salesforce.com

همانطور که قبلا هم بیان شد، Salesforce.com به کسب‌وکارها امکان تغییر نام فیلدهای Salesforce.com را می‌دهد، هم‌چنین امکان اضافه کردن فیلدهای خاصی را نیز می‌دهد. برای مثال کسب‌وکارها می‌توانند فیلدهایی را برای رکود خاصی اضافه کنند که در شکل (E۴-۲) نشان داده شده است که این‌ها برای ردیابی داده‌های منحصر به فرد کسب‌وکار است. فیلدهایی مانند فیلد محصول، که از طریق picklist انتخاب می شوند، می‌توانند کد محصول باشند. جریان‌های کاری (فرآیندهای کسب‌وکار) در Salesforce.com توسط یکسری قوانین ضبط می‌شوند. برای مثال انتخاب قانون (*assignment rules*) نشان داده شده در شکل (E۴-۲) برای اختصاص اتوماتیک

مورد به نماینده‌های پشتیبان است. با بروزرسانی *assignment rules* ، جریان موردی مشری می‌تواند به نیازهای کسب‌وکار راهنمایی شود. دیگر قوانینی هستند که می‌توانند پیاده‌سازی شوند:

۱. هشدار ایمیل (که تحت شرایط خاصی ایمیلی برای هشدار ارسال می‌کند)
۲. فیلدهای بروزرسانی (برای مثال هنگامی که مخاطبی در حال منقضی شدن است)
۳. پیام‌های خروجی (برای مثال ارسال پیام به سیستم مالی در هنگامی که یک فاکتور تایید شده است)

جزئیات در
https://login.salesforce.com/help/doc/en/creating_workflow_rules.htm
قابل رویت است.

در نهایت مدیران و کاربران می‌توانند واسط کاربری برنامه را سفارشی‌سازی کنند. این شامل آیتم‌هایی مانند قرار دادن و محتوای متن و گرافیک، نام و شماره تب‌ها در هر صفحه، و طرح کلی صفحه است. مدیر می‌تواند به پرسنل نیز اجازه‌ی سفارشی‌سازی دیدهای شخصیشان را بدهد.جزئیات بیشتر در این مورد https://na۳.salesforce.com/help/doc/user_ed.jsp?loc=help است. یک برنامه‌ی SaaS دیگر که عملکردش مانند Salesforce.com است Suger CRM می‌باشد که منبع باز است. مقایسه‌ی این دو نرم‌افزار در مقاله‌ی http://www.salesforce.com/ap/form/sem/why_salesforce:ondemand.jsp?d = ۷۰۱۳۰۰۰۰۰۰EN۱GandDCMP = KNC-Googleandkeyword = sugar٪۲۰CRMandadused = ۱۵۷۴۵۴۲۱۷۳andgclid = CNfqoLK۲uaQCFc۵R۶wod_R۳TbQ قرار داده شده است.

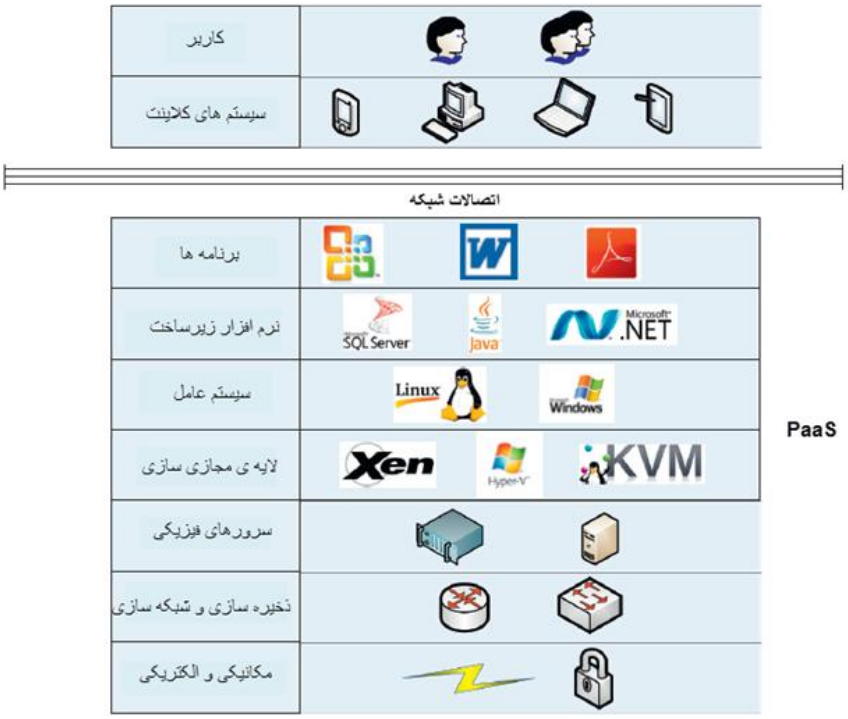
پلت‌فرم بعنوان سرویس

PaaS سرویسی است که به موجب آن مشتریان پلت‌فرمی برای استفاده برای نیازهای رایانشی خود دارند. در بیشتر موارد، این پلت‌فرم برای توسعه استفاده می‌شود. بسته به ارائه‌دهنده، پلت‌فرم توسعه می‌تواند یک سیستم پردازشی باشد و یا یک پلت‌فرم توسعه‌ی کامل که شامل سروهای وب و

کتابخانه‌های توسعه است. شکل ۴,۸ سرویس‌هایی را که شما می‌توانید از ارائه‌دهنده‌ی SaaS انتظار داشته باشید را بیان می‌کند.

خصوصیات PaaS

پیاده‌سازی‌های PaaS به سازمان‌ها امکان ایجاد و استقرار برنامه‌های وب بدون ایجاد زیرساختشان را می‌دهد. PaaS عموما ساده‌سازی‌هایی برای توسعه، یکپارچه‌سازی و تست را ارائه می‌دهد. در این‌جا ما برخی از این ویژگی‌ها را که در محیط PaaS است را پوشش می‌دهیم. پیش از این ابتدا باید درک کنیم که در محیط SaaS چه اتفاقی می‌افتد. هنگامی که سازمانی پیاده‌سازی PaaS را انتخاب می‌کند، آن برخی برنامه‌ها و سرویس‌ها را نیز در پلت‌فرمش پیاده‌سازی می‌کند. ارائه‌دهنده عمومی کنترلی روی چگونگی توسعه‌ی سرویس یا برنامه یا روی کیفیت توسعه ندارد. در بسیاری از توسعه‌ها، ارائه‌دهنده‌ی سرویس‌های اضافه را پیشنهاد می‌دهد، که این برای کمک به استقرار است.



شکل ۴-۸: سرویس های PaaS

سفارشی‌سازی

با PaaS، شما کنترل کامل روی برنامه‌ها خواهید داشت، بنابراین شما می‌توانید آنها را سفارشی‌سازی کنید. در پلت‌فرم توسعه شما قادر به ایجاد تغییرات زیادی نمی‌باشید. در بیشتر موارد، این پلت‌فرم توسط ارائه‌دهنده به شدت کنترل می‌شود. احتمالا گزینه های پیکربندی دیگری وجود دارد که می توانید تنظیم کنید، اما سفارشی‌سازی محدود است.

آنالیزها

از آنجایی که شما، مشتری، برنامه‌هایی را ایجاد می‌کند، شما توانایی دیدن مصرف برنامه‌های و تعیین گرایش‌ها را دارید. شما قادر به دیدن این که کدام عناصر بیشترین استفاده را دارند و کدام مورد استفاده قرار نمی‌گیرد، خواهید بود. در محیط PaaS، شما به پلت‌فرم هم دسترسی دارید. شما قادر خواهید بود که تعیین کنید چه زمان سیستم جدید برای مدیریت بار اضافه شود. اغلب ارائه‌دهنده‌ها به شما توانایی چرخاندن سیستم‌های جدید در هنگامی که سیستم جاری به آستانه‌ی بار رسیده است را می‌دهند.

یکپارچه‌سازی

در محیط PaaS، داده‌های در سایت ارائه‌دهنده ذخیره می‌شوند، اما مشتری به آن دسترسی مستقیم دارد. اداره‌ی هوش کسب‌وکار و گزارش‌دهی نباید از دید نقطه‌ی دسترسی مساله‌ای باشد، زیرا شما ممکن است مقدارهای زیادی از داده‌ها را بین محیط داخلی خود و محیط ارائه‌دهنده حرکت دهید. بنابراین نگرانی‌هایی در زمینه‌ی کارآیی وجود دارد که مخالف نگرانی‌های عملکرد و دسترسی هستند.

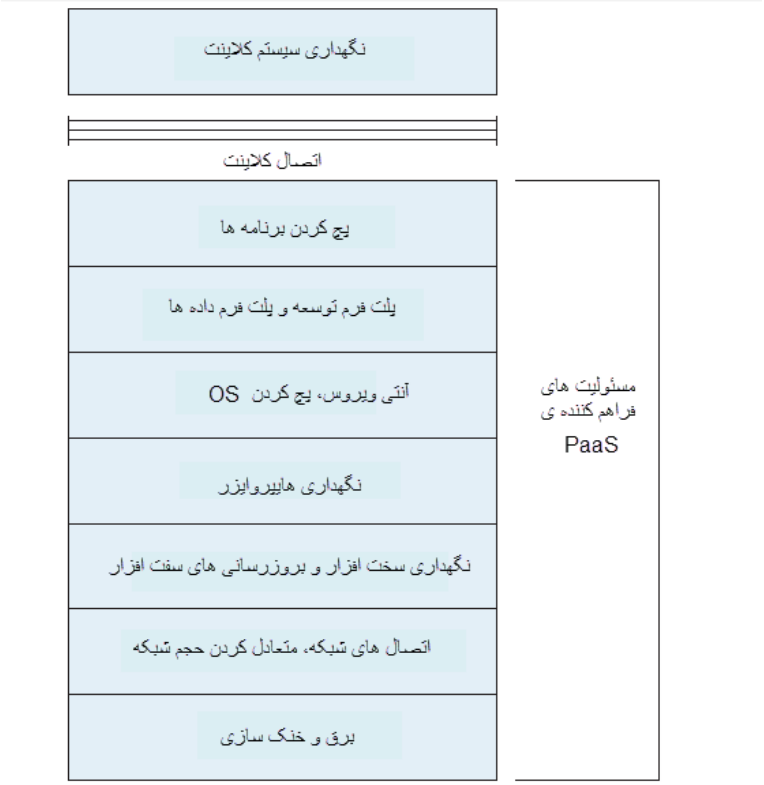
مسئولیت‌های PaaS

در پیشنهاد PaaS، مسئولیت‌ها تاحدودی بین ارائه‌دهنده‌ی سرویس و مشتری توزیع می‌شود (شکل ۴,۹). ارائه‌دهنده در سطح توسعه‌ی پلت‌فرم و زیر آن همه‌چیز را در نظر می‌گیرد. ارائه‌دهنده اطمینان ایجاد می‌کند که سیستم عامل وصله و بروز رسانی شده است (در هنگام تحویل به شما). ارائه‌دهنده هم‌چنین بروزرسانی‌های سیستم‌عامل دوره‌ای را انجام می‌دهد که در اختیار شما قرار می‌گیر. در پیاده‌سازی PaaS، مشتری عموما مسئول همه‌چیز در سیستم عامل و پلت‌فرم توسعه است. شما مسئول نصب و نگهداری برنامه‌های اضافی می‌باشید که نیاز دارید. این شامل وصله‌کردن برنامه و کنترل برنامه است. پلت‌فرم دیتابیس برای شما ممکن است تامین شده باشد، اما شما مسئول داده‌ها

می‌باشید. در پیاده‌سازی PaaS ، شما دسترسی مستقیم به داده‌ها دارید. اگر مشکلی با داده‌ها وجود داشته باشد، شما قادر به پیاده‌سازی تعمیر داده‌ها به صورت مستقیم هستید.

درایورهای PaaS

درایورهای زیادی هستند که رشد بازار PaaS را تحت تاثیر قرار داده‌اند. بسیاری از سازمان‌ها می‌خواهند به سمت مدل ابر عمومی بروند، اما نمی‌توانند ارائه‌دهنده‌ی SaaS عمومی را که پیشنهاد دهنده‌ی برنامه‌ی مورد نیاز آنها است را پیدا کنند. مدل PaaS به آنها امکان حرکت زیرساخت و پلت‌فرم‌هایشان به خارج از دیتاسنتر داخلی را می‌دهد در حالی که به آنها امکان توسعه‌ی برنامه‌های مورد نیازشان داده می‌شود.



شکل ۴-۹: مسئولیت های PaaS

چالش‌های PaaS

چالش‌هایی در محیط‌های PaaS عمومی وجود دارد، از جمله مسائل مربوط به انعطاف‌پذیری و امنیت.

چالش‌های انعطاف‌پذیری

شما ممکن است مشکلاتی در پیدا کردن ارائه‌دهنده‌ای که پلت‌فرم مورد نیاز شما را دارد، داشته باشید.بیشتر ارائه‌دهنده‌های PaaS پیشنهادهایشان را به تنظیمات پلت‌فرم خاصی محدود می‌کنند. اگر شما به تنظیم خاصی احتیاج داشته باشید و یا پیکربندی خاصی، شما قادر به پیدا کردن ارائه‌دهنده‌ای که چیزی را که شما می‌خواهید را ارائه دهد نخواهید بود.

چالش‌های امنیتی

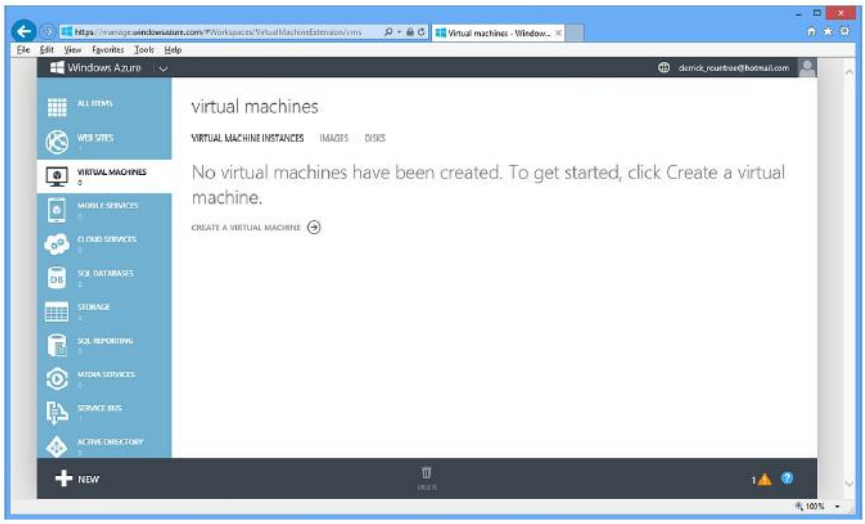
ارائه‌دهنده کنترل مدیریتی روی سیستم عامل و پلت‌فم دیتابیس دارد. از آنجایی که ارائه‌دهنده دسترسی مستقیم به سیستم‌ها دارد، آنها دسترسی مستقیم به تمام برنامه‌ها و داده‌ها دارند.

ارائه‌دهنده‌های PaaS

تعدادی از ارائه‌دهنده‌های PaaS در بازار به رشد خود ادامه می‌دهند. در ابتدا ما نگاهی به windows Azure داریم.

windows Azure

windows Azure که در شکل ۱۰-۴ نشان داده شده است، از اولین پیشنهادهای PaaS در بازار است. windows Azure پیشنهاد و ارتقاء رایگانی را دارد که ویژگی‌هایی مانند SLAهای افزایش یافته را پیشنهاد می‌دهد. windows Azure چرخاندن یک وب‌سایت یا توسعه‌ی پلت‌فرم را بسیار ساده می‌کند. windows Azure شامل گزینه‌های گسترده‌ای مانند سرویس‌های محاسباتی، سرویس‌های داده، سرویس‌های app و سرویس شبکه است.



شکل 4.10
Windows Azure

شکل ۴-۱۰: Windows Azuro

موتور Google App

موتور Google App یک راه‌حل PaaS است که کاربران را قادر به میزبانی برنامه‌هایشان روی یک زیرساخت یکسان مانند Google Docs، Google Maps، و دیگر سرویس‌های معروف گوگل می‌کند. مانند Microdoft Azure پلت‌فرمی را برای ایجاد و ساخت برنامه‌های Net. فراهم می‌کند، این موتور از توسعه و میزبانی برنامه‌های نوشته شده با Java، Python، و زبان‌های جدید دیگر[۴۲] اطمینان ایجاد می‌کند. این پلت‌فرم هم‌چنین دیگر زبان‌های زمان اجرای ماشین مجازی جاوا را نیز پشتیبانی می‌کند، مانند JavaScript (Rhino)، Jruby و Scala. برنامه‌ی میزبانی شده در این موتور می‌تواند در حافظه و رانش قابل بسط باشد، دقیقا مانند محصولات گوگل. این پلت‌فرم ذخیره‌ی توزیع‌شده‌ای را با تکثیر و متعادل سازی بار درخواست‌های کلاینت برقرار می‌کند. این برنامه می‌تواند به راحتی با استفاده از توسعه ی یکپارچه‌سازی‌شده‌ی Eclipse ایجاد شود . این بخش بررسی ساده‌ای را روی این پلت‌فرم انجام می‌دهد و برخی نکات کلیدی را در اختیار شما قرار می‌دهد.

[۴۲] http://golang.org/doc/go_tutorial.html.

شروع

دستورالعمل‌های گام به گامی برای استفاده از موتور گوگل App در این‌جا توصیف شده است، که این‌ها براساس پروسه‌های در دسترس این کتاب هستند[۴۳]. ابتدا توسعه‌دهنده ثبت نام می‌کند که این با استفاده از Gmail انجام می‌شود. شکل E۴-۵ نشان‌دهنده‌ی اولین اسکرین در هنگامی که برنامه پیکربندی می‌شود است.

موتور Google App به برنامه‌ای که جدیدا توسعه یافته است امکان می‌دهد از دامنه‌ی ود توسعه‌دهنده به خدمت گرفته شود. برای مثال، اگر توسعه دهنده myapp را بعنوان نام یک برنامه انتخاب کند، این برنامه در http://myapp.appspot.com به خدمت گرفته می‌شود این URL می‌تواند به صورت عمومی یا انتخابی با گروه‌های کوچکی از اعضا به اشتراک گذاشته شود. هر توسعه‌دهنده می‌تواند ۱۰ برنامه را به رایگان میزبانی کند، با ۵۰۰MB حافظه‌ی قابل تعریف. توسعه‌دهنده باید صوری را برای حافظه و منبع‌های پهنای باند استفاده شده توسط برنامه در فراتر از این محدوده‌های بدهد. داشبورد ساده ای که متریک‌ها را برای هر برنامه نشان می‌دهد می‌تواند روی پورتال دیده می‌شود، یک اسکرین از شکل E۴-۶.

نکته: استقرار و توسعه روی موتور Google App

۱. دانلود SDK
۲. ایجاد "Web Application Project " جدید
۳. پیکربندی برنامه
۴. توسعه‌ی کد
۵. تست در محیط موتور App شبیه‌سازی شده
۶. استقرار در موتور Google App

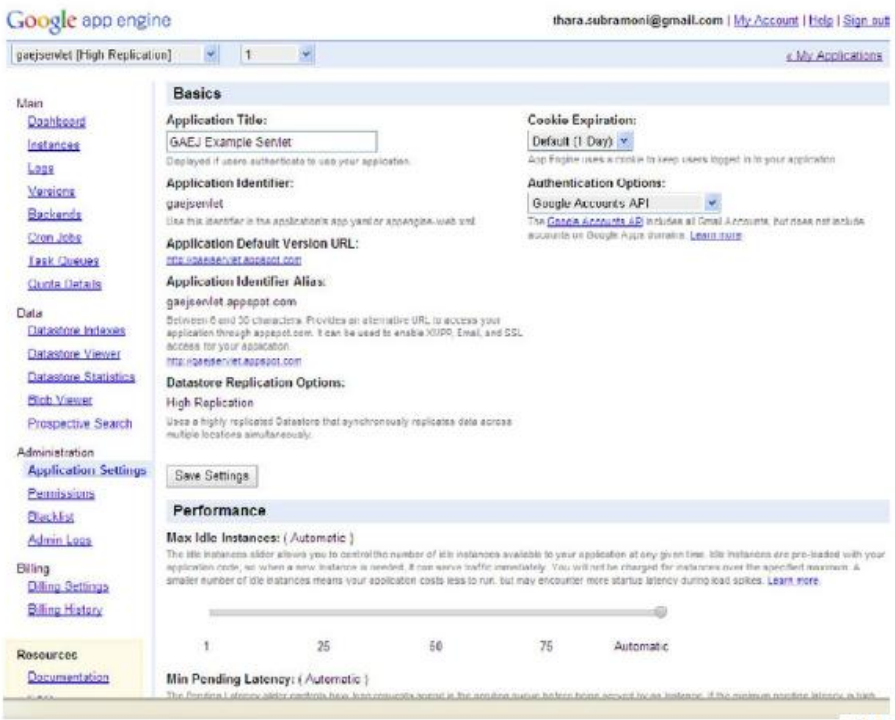
توسعه‌ی یک برنامه‌ی موتور Google App

برای توسعه‌ی برنامه‌های جاوا، کیت توسعه‌ی نرم‌افزار موتور Appباید نصب شود. SDK یک پلاگین Eclipse است (شکل E۴-۷) که شامل ایجاد، تست و استقرار محیط‌ها است و در http://dl.google.com.eclipse/plugin/۳.x در دسترس است. برای شروع، پروژه‌ی جدیدی

[۴۳] http://code.google.com/appengine/.

را ایجاد کنید بعنوان یک پروژه‌ی برنامه‌ی وب؛ روی اسم پروژه کلیک راست کنید و **Google** را انتخاب کنید و ID برنامه‌ی معتبری را برای پروژه وارد کنید. بعد از توسعه‌ی برنامه، در طول استقرار ما باید یک آیدی App را برای برنامه مشخص کنیم. برای استقرار روی موتور App، مانند ایجاد برنامه، روی نام پروژه کلیک راست کنید و گزینه‌ی **Deploy to App Engine** را انتخاب کنید، و برنامه روی موتور App آپلود می‌شود و استقرار می‌یابد.

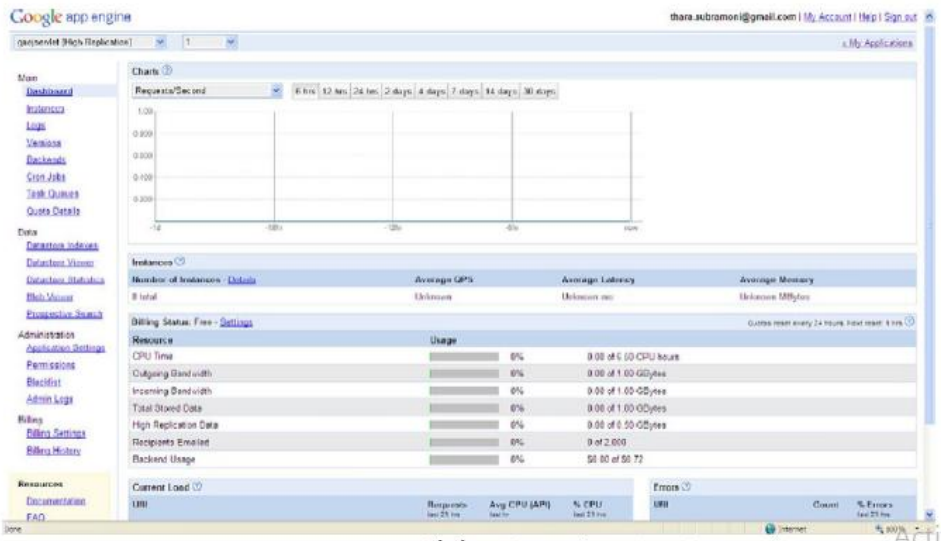
.



شکل E4.5

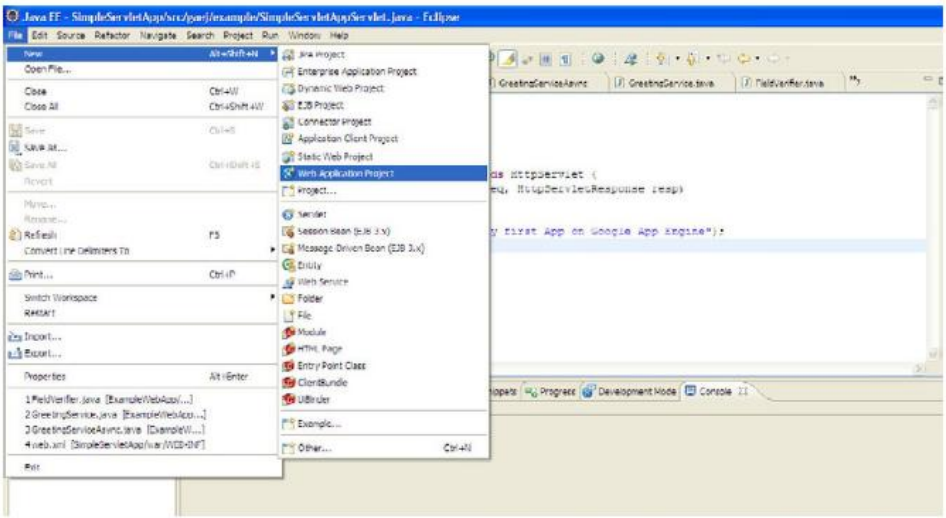
موتور برنامه ی گوگل: پیکربندی برنامه ها

شکل E۴-۵: موتور برنامه ی گوگل: پیکر بندی برنامه ها



شکل E4.6 : داشبورد برنامه برای موتور برنامه ی گوگل

شکل ۶-E۴ : داشبورد برنامه برای موتور برنامه گوگل



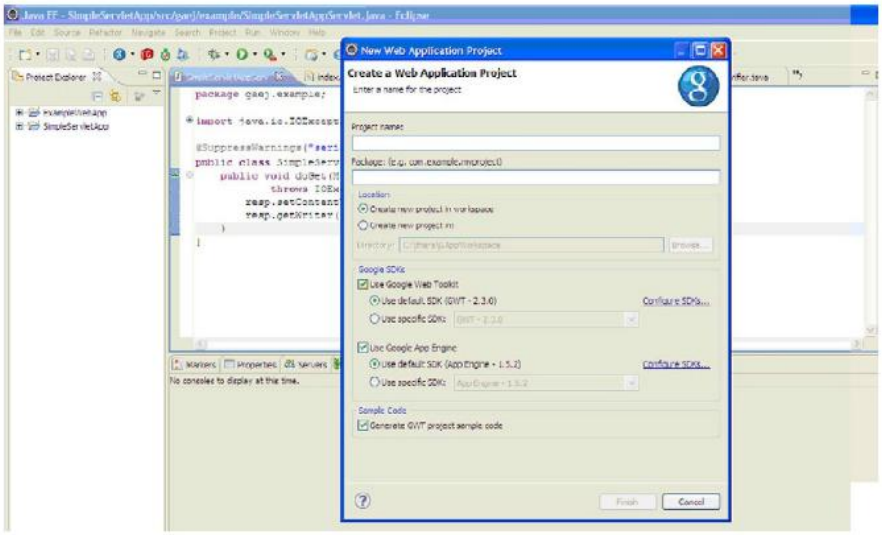
شکل E4.7 : پلاگین Eclipse موتور برنامه ی گوگل

شکل ۷-E۴: پلاگین Eclipse موتور برنامه گوگل

گزینه‌ی جالب دیگری در طول پیکربندی برنامه، گزینه‌ی ایجاد یک کیت ابزار وب گوگل (GWT) است. GWT عموما به شما امکان ایجاد برنامه‌های تعاملی با امکان drag و drop را می‌دهد تا یک واسط گرافیکی جدیدی سفارشی را ایجاد کنید. این ابزار سپس به صورت اتوماتیک گزینه‌ی UI را به جاوا اسکریپت با AJAX تبدیل می‌کند که برای دسترسی به منطق خادم[۴۴] روی سرور فراخوانی می‌شود. توجه داشته باشید که جاوا اسکریپت در داخل یک جستجوگر اجرا می‌شود و AJAX یک راه غیربلوک کردن را برای دسترسی به خادم فراهم می‌کند، تاثیر کلی تجربه‌ی خوبی در پاسخ سریع برای برنامه های تعاملی است. یک کد اسکلت برای GWT با استفاده از دستور زیر ایجاد می‌شود:

```
webAppCreator -out myFirstApp com.cloudbook.myFirstApp
```

توسعه‌دهنده هم‌چنین می‌تواند گزینه‌ی **Generate GWT Sample Code** را در طول ایجاد برنامه چک کند که این برای ایجاد پروژه‌ی خوش آمدید (شکل E۴-۸) است. اگر این گزینه غیر بررسی شده باشد، ما می‌توانیم کد servlet جاوای خود را بنویسیم و آن را روی موتور App قرار دهیم. بنابراین هر برنامه‌ی نوشته شده‌ای روی جاوا می‌تواند روی موتور App قرار گیرد.



شکل E4.8 : استقرار برنامه ی موتور برنامه ی گوگل

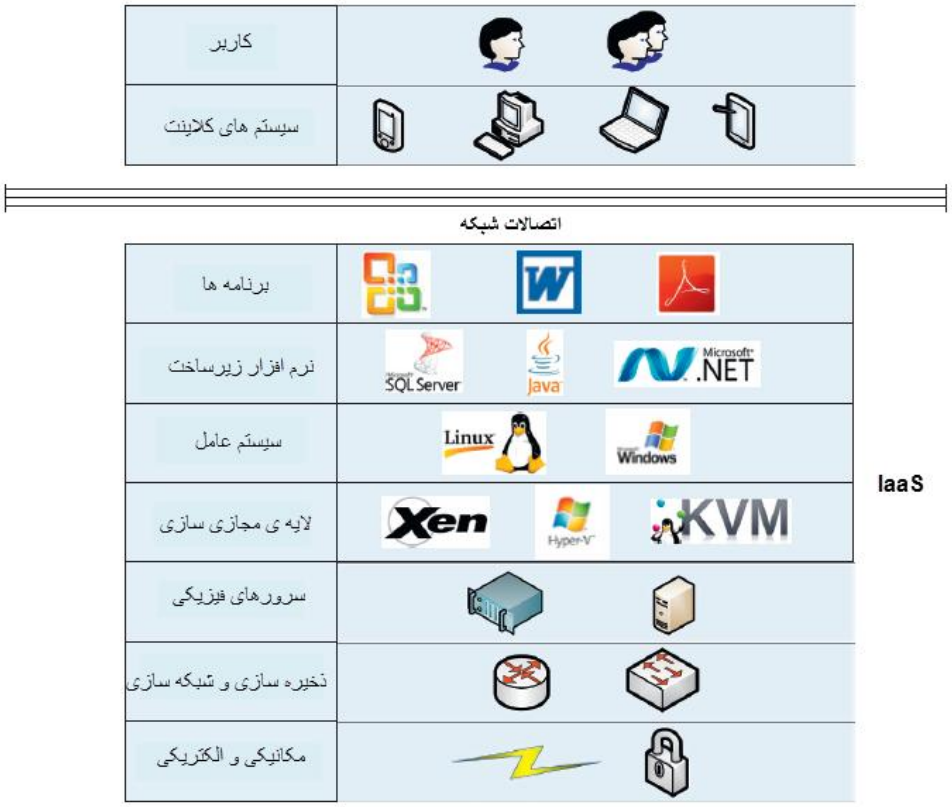
[۴۴] backend

شکل ۸-E۴: استقرار برنامه ی موتور برنامه گوگل

SDK همراه با سرور وب محلی برای تست استقرار است. این سرور وب محلی زمان اجرای ایمن یا محیط sandbox موتور App را با دسترسی محدود به سیستم‌عامل پایه شبیه‌سازی می‌کند. برای مثال برنامه می‌تواند تنها با استفاده از HTTP روی پورت خاصی در دسترس باشد. آن نمی‌تواند روی فایل سیستمی بنویسد و تنها می‌تواند فایل‌هایی را بخواند که با کد برنامه آپلود شده‌اند. محدودیت دیگر sandbox برنامه است، در هنگام دسترسی روی HTTP، باید کد پاسخی در ۳۰ ارسال شود. این محدودیت‌ها اساسا برای جلوگیری کردن از دخالت یک برنامه در برنامه‌ای دیگر است.

زیرساخت بعنوان سرویس

IaaS سرویس‌های هسته‌ای مانند قدرت رایانش، ذخیره‌سازی، شبکه و سیستم‌های عامل را فراهم می‌کند. سپس شما می‌توانید محیط خود را در بالای این منابع ایجاد کنید (شکل ۴-۱۱). یک ارائه‌دهنده‌ی IaaS منابع سخت‌افزاری مانند سرورها را نیز برای شما فراهم کند. این سرورها در مرکز داده‌ی ارائه‌دهنده قرار دارند، اما شما به آنها دسترسی مستقیم دارید. سپس شما می‌توانید هرچیزی را که می‌خواهید روی آن سرورها نصب کنید. این هزینه‌ی زیادی دارد، زیرا ارائه‌دهنده قادر به ایجاد استفاده از چند مستاجری نمی‌باشد. از این رو، مشتریان باید تمام هزینه‌ها را تحلیل کنند.



شکل 4.11: سرویس های IaaS

شکل ۴-۱۱: سرویس های IaaS

یک مدل رایج برای ارائه‌دهنده‌ی IaaS این است که فرای شما ماشین مجازی را فراهم کند و شما هر چیزی را که دوست دارید روی آن ماشین مجازی می‌توانید نصب کنید. این ماشین‌های مجازی می‌توانند Windows، Linux، و دیگر سیستم‌های عامل را اجرا کنند. به دلیل این که سیستم‌های مجازی‌سازی شده‌اند، ارائه دهنده قادر به استفاده از مزیت‌های چند مستاجری می‌باشد. این سیستم‌ها مشتریان زیادی را روی یک سخت‌افزار فیزیکی میزبانی می‌کنند. آنها می‌توانند به شدت ظرفیت خود را افزایش دهند. و این باعث کاهش هزینه می‌شود.

مسئولیت‌ها

در استقرار IaaS مشتری مسئول بیشتر محیط است (شکل ۴,۱۲). ارائه‌دهنده مسئول هایپروایزر و زیر آن است. این شامل سخت‌افزار فیزیکی، مخزن، و شبکه است. سخت‌افزار فیزیکی در دیتاسنتر ارائه‌دهنده ذخیره می‌شود. اما مشتری به آن دسترسی کامل دارد.

مشتری مسئول چیزهای دقیق مانند سیستم عامل و نگهداری برنامه است. با این حال، باید مواردی خاصی در نظر گرفته شوند، مانند آنتی ویروس. مشتری مسئول اطمینان از این است که سیستم آنتی ویروس را بروز رسانی کرده است.

شکل 4.12 مسئولیت های Iaas

شکل ۴-۱۲: مسئولیت های IaaS

درایورها

بسیاری از سازمان‌ها از IaaS برای بسط ظرفیتشان استفاده می‌کنند. بجای صرف هزینه‌های زیاد برای بسط مرکز داده و ایجاد مرکز داده‌ای جدید، سازمان‌ها سیستم‌هایی را اجاره می‌کنند که توسط ارائه‌دهنده‌ی IaaS فراهم شده است. برخی سازمان‌ها تنها در شرایط خاصی به ظرفیت افزایش یافته نیاز دارند. به همین علت، آنها نمی‌خواهند برای راه‌حل‌های دائمی گران هزینه‌ای بپردازند.

چالش‌ها

چالش‌هایی در انتخاب IaaS وجود دارد. بسیاری از سازمان‌ها مزایا را می‌بینند، اما آنها نسبت به از دست دادن کنترل نگران هستند. هزینه‌ی کلی می تواند یک مساله باشد. در بسیاری از محیط‌های IaaS، شما هزینه ی استفاده از منابع، مانند پردازنده و حافظه را می‌دهید.

چالش‌های امنیتی

چالش‌های امنیتی برای محیط‌های IaaS مانند چالش‌های دیگر ارائه‌دهنده‌ی سرویس است. با این حال از آنجایی که ارائه‌دهنده به دسترسی به سیستم عامل واقعی دسترسی ندارد (در سطح بالاتر)، برای آنها نیازی به داشتن حساب مدیریتی روی سیستم نیست. که این باعث امنیت بیشتری می‌شود.

ارائه‌دهنده‌های IaaS

ارائه‌دهنده‌های IaaS جایگاه رو به رشدی را در بازار دارند و تاثیر آنها رو به افزایش است. علاوه بر تقاضا، پلت‌فرم‌های IaaS مانند CloudStack و OpenStack برای ایجاد خودکارسازی و نظم توسعه یافته‌اند. در این‌جا دو ارائه‌دهنده‌ی IaaS معروف را پوشش می‌دهیم، Amazon EC۲ و Rackspace.

ابر رایانشی الاستیک یا انعطاف پذیر آمازون[۴۵] (EC۲)

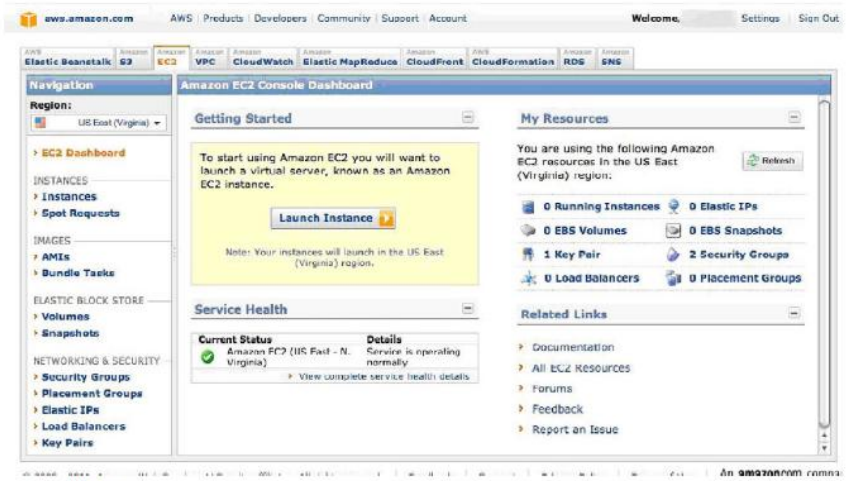
نوع دیگر مهم IaaS محاسبه بعنوان سرویس است، که به موجب آن منابعی رایانشی بعنوان سرویس پیشنهاد می‌شوند. البته برای یک پیشنهاد رایانش بعنوان سرویس کارآمد، امکان مرتبط کردن مخزن

[۴۵] Amazon Elastic Compute Cloud

با سرویس رایانش وجود دارد (به طوری که نتایج محاسبات می‌توانند پایدار باشند). شبکه‌ای کردن مجازی نیز مورد نیاز است به طوری که امکان برقراری ارتباط با نمونه‌ی رایانشی وجود دارد. همه‌ی این‌ها با هم زیرساختی را ب عنوان سرویس ایجاد می‌کند. ابر رایانش مرتجع آمازون، که یکی از پیشنهادهای معروف در زمینه‌ی رایانش بعنوان سرویس است، یک مثال ساده‌ای دارد که نشان می‌دهد که چگونه می‌تواند با پیشنهاد StaaS آمازون برای ایجاد پورتالی که مشتریان می‌توانند کتاب‌های خود را به اشتراک بگذارند مورد استفاده قرار بگیرد. در نهایت ما مثالی را داریم که نشان دهنده‌ی ویژگی‌های پیشرفته EC۲ است.

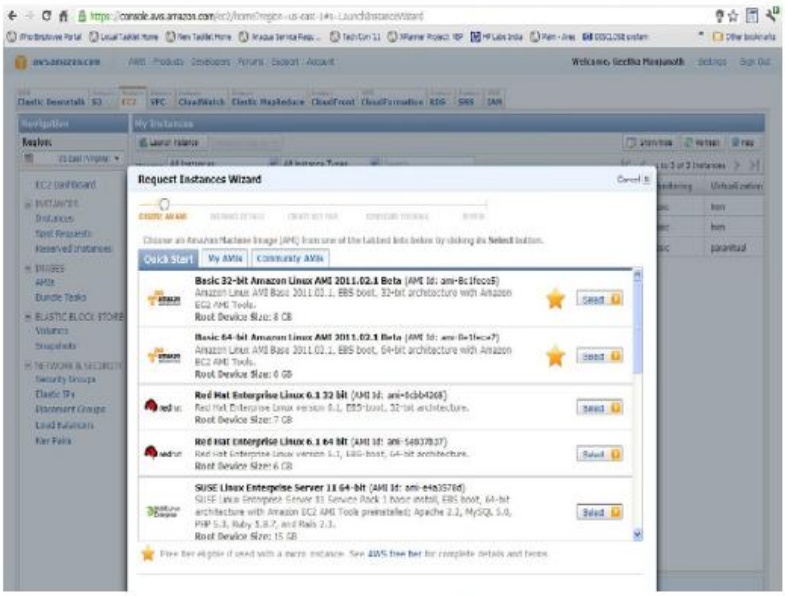
بررسی EC۲ آمازون

EC۲ آمازون به شرکت‌ها امکان تعریف سرور مجازی را می‌دهد، که این همراه با شبکه‌سازی مجازی و مخزن مجازی است. نیازهای رایانشی شرکت‌ها می‌تواند بسیار زیاد باشد؛ برخی برنامه‌ها ممکن است متمرکز روی محاسبه باشند، و دیگر برنامه‌ها روی مخزن تمرکز دارند. برنامه های شرکتی خاصی ممکن است به محیط های نرم‌افزاری مشخصی نیاز داشته باشند؛ دیگر برنامه‌ها به خوشه‌های محاسباتی برای اجرا به صورت کارآمد نیاز دارند. نیازهای شبکه‌سازی نیز ممکن است زیاد باشد. این گوناگونی در سخت‌افزار محاسبه، به همراه نگهداری خودکار و توانایی مدیریت مقیاس، EC۲ را یک پلت‌فرم منحصر به فرد کرده است.



شکل E4.9
AWS EC2 console.

شکل ۹-E۴: کنسول AWS



شکل E4.10 ایجاد نمونه ی EC2 با استفاده از کنسولAWS

شکل ۱۰-E۴: ایجاد نمونه EC۲ با استفاده از کنسول AWS

دسترسی به EC۲ با استفاده از کنسول AWS

مانند S۳، EC۲ می‌تواند از طریق سرویس‌های وب آمازون در دسترس باشد یعنی در http://aws.amazon.com/console . شکل E۴,۹ نشان‌دهنده‌ی داشبورد کنسول EC۲ است، که می‌تواند برای ایجاد یک نمونه‌ی، بررسی حالت نمونه‌های کاربر، و حتی خاتمه‌ی یک نمونه به کار گرفته شود. با کلیک روی **Launch Instance** کاربر به اسکرین نشان داده شده در شکل E۴-۱۰ می‌رود، که در آن مجموعه‌ای از تصاویر سیستم عامل پشتیبانی شده (که تصاویر ماشین آمازون یا AMIs نام دارند) برای انتخاب نشان داده شده است. هنگامی که تصویری انتخاب می‌شود، ویزارد نمونه‌ی EC۲ برای کمک در تنظیم گزینه‌های بیشتر به کاربر برای نمونه ظاهر می‌شود، مانند ورژن هسته‌ی OS خاص برای استفاده. سپس کاربر باید حداقل یک جفت کلید –مقدار را ایجاد کند که برای اتصال ایمن به نمونه است برای ایجاد یک جفت کلید و ذخیره‌ی فایل از دستورالعمل‌ها پیروی کنید . کاربر می‌تواند از کلید مقدار ایجاد شده در هنگامی که کاربر نمونه‌های زیادی دارد نیز

استفاده‌ی مجدد کند (استفاده از نام کاربری و پسورد برای دسترسی به ماشین‌ها نیز به همین صورت است).

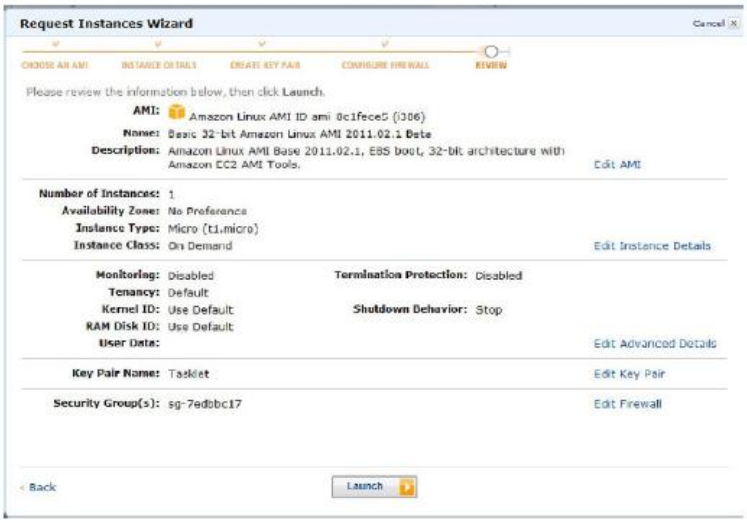
سپس گروه‌های امنیتی برای نمونه می‌توانند برای اطمینان از این که پورت‌های شبکه‌ی مورد نیاز باز یا بلاک هستند (برای نمونه) تنظیم شود. برای مثال انتخاب پیکربندی Web Server پورت ۸۰ را فعال می‌کند. قوانین فایروال‌های پیشرفته‌تری می‌توانند بکار گرفته شوند. آخرین اسکرین پیش از راه‌اندازی نمونه شکل نشان داده شده در E۴-۱۲ است. راه‌اندازی نمونه نام DNS عمومی را می‌دهد که کاربر می‌تواند برای وارد شدن به صورت خودکار استفاده کند اگر که ابر سرور و ماشین کلاینت در یک شبکه باشند. برای مثال برای شروع استفاده از کلاینت لینوکس، کاربر از دایرکتوری که فایل کلید مقدار در آن ذخیره شده است دستور زیر را وارد می‌کند. بعد از چندین صفحه‌ی تاییدیه، کاربر به ماشین برای استفاده از دستورات لینوکس وارد می‌شود. برای دسترسی ریشه، کاربر باید از دستور Sudo استفاده کند.

ssh -i my_keypair.pem ec۲-۶۷-۲۰۲-۶۲-۱۱۲.compute-۱.amazonaws.com

برای ویندوز، کاربر باید فایل my_keypair.pem را باز کند و از دکمه‌ی **Get Windows Password** که روی صفحه‌ی نمونه‌ی AWS وجود دارد استفاده کند. کنسول پسورد مدیریتی را که می‌تواند برای اتصال به نمونه با استفاده از یک برنامه‌ی دسکتاپ از راه دور استفاده شود را برگرداند (اغلب در **Start->All Programs -> Accessories -> Remote Desktop Connection** است).

شکل E4.11 نمونه ویزارد برای EC2

شکل E۴-۱۱: نمونه ویزارد برای EC۲



شکل E4.12 : پارمترهایی که می توانند برای نمونه ی EC2 فعال شوند

شکل ۱۲-E۴: پارامترهایی که می توانند برای نمونه ی EC۲ فعال می شوند

دسترسی به EC۲ با استفاده از ابزارهای خط فرمان[۴۶]

آمازون هم‌چنین فراهم‌کننده‌ی یک واسط خط فرمان برای EC۲ است که از EC۲ API برای پیاده‌سازی عملیات‌های خاصی که نمی‌توانند با کنسول AWS انجام شوند، استفاده می‌کند. در زیر به طور خلاصه توصیف شده است که چگونه باید سرویس های خط فرمان تنظیم و نصب شوند. جزئیات بیشتر در راهنمایی کاربر رایانش محاسبه‌ی مرتجع آمازون[۴۷] است. جزئیات ابزارهای خط فرمان در مرجع خط فرمان ابر رایانش مرتجع آمازون[۴۸] است.

نکته: نصب ابزارهای خط فرمان EC۲

- دانلود ابزارها
- تنظیم متغیرهای محیطی (برای مثال مکان JRE)

[٤٦] Command-Line
[٤٧] *http://docs.amazonwebservices.com/AWSEC ۲/latest/UserGuide/*
[٤٨] *http://docs.amazonwebservices.com/AWSEC ۲/latest/CommandLineReference/*

- تنظیم محیط امنیتی (برای مثال دادن اعتباردهی)
- تنظیم منطقه

تنظیم متغیرهای محیطی

اولین دستور تنظیم‌کننده‌ی متغیر محیطی است که مشخص‌کننده‌ی دایرکتوری است که جاوا آن را ایجاد می‌کند (در حین اجرا). PATHNAME باید نام کامل مسیر دایرکتوری باشد که فایل java.exe در آن جا است. دومین دستور مشخص‌کننده‌ی دایرکتوری است که در آن ابزارهای EC۲ قرار دارند؛ TOOLS_PATHNAME باید مسیر کامل دایرکتوری ec۲-api-tools-A.B-nnn باشد که در آن ابزارها از حالت فشرده خارج شده‌اند. (A و B و nnn دیجتال‌هایی هستند که براساس ورژن استفاده شده با هم متفاوت هستند). سومین دستور مسیر قابل اجرا برای اضافه کردن دایرکتوری در جایی که ابزارهای دستور EC۲ نشان داده شده‌اند را تنظیم می‌کند.
برای لینوکس:

```
$export JAVA_HOME = PATHNAME
$export EC۲_TOOLS = TOOLS_PATHNAME
$export PATH=$PATH:$EC۲_HOME/bin
For Windows:
C:\>SET JAVA_HOME = PATHNAME
C:\>SET EC۲_TOOLS = TOOLS_PATHNAME
C:\>SET PATH = %PATH%,%EC۲_HOME%\bin
```

تنظیم محیط امنیتی: مرحله‌ی بعد تنظیم محیط است به طوری که سرویس های خط فرمان می‌توانند در طول هر تکرار اعتبارسنجی شوند. برای این کار، لازم به دانلود اعتبار X.۵۰۹ و کلید خصوصی است که درخواست‌های HTTP را برای آمازون اعتبارسنجی می‌کند. X.۵۰۹ می‌تواند با کلیک روی لینک **Account** که در شکل E۴,۹ نشان داده شده است دانلود شود، سپس کلیک روی لینک **Security Credentials**، و پیروی از دستورالعمل‌های داده شده برای ایجاد اعتبار جدید. فایل‌های اعتبار باید به صورت دایرکتوری ec۲. دانلود شوند (در دایرکتوری home) روی لینوکس Linux/Unix و روی ویندوز C:\ec۲، بدون تغییر نام‌های آنها. دستورهای زیر برای تنظیم محیط باید اجرا شوند؛ دستورهای لینوکس و ویندوز داده شده است. در این‌جا f۱.pem فایل اعتبار دانلود شده از EC۲ است.

$export EC۲-CERT = ~/.ec۲/f۱.pem

یا

C:\> set EC۲-CERT = ~/.ec۲/f۱.pem

تنظیم محدوده: لازم است که در مرحله‌ی بعدی محدوده‌ای که ابزار دستور EC۲ با آن تعامل دارد، تنظیم شود- یعنی مکانی که در آن ماشین‌های مجازی EC۲ باید ایجاد شوند. به طور خلاصه، هر ناحیه نشان‌دهنده‌ی یک مرکز داده‌ی AWS است، و قیمت‌گذاری AWS با توجه به ناحیه متفاوت است. دستور ec۲-describe-regions می‌تواند در این نقطه برای آزمون نصب ابزارهای دستور EC۲ و لیست نواحی در دسترسی بکار گرفته شود. ناحیه‌ی پیش فرض استفاده شده، ناحیه‌ی US-East است که **us-east-۱** می‌باشد، به همراه URL نقطه‌ی پایان سرویس http://ec۲.us-east-۱.amazonaws.com ، اما آن می‌تواند به هر نقطه‌ی پایانی تنظیم شود که این با استفاده از دستور زیر انجام می‌شود که ENDPOINT_URL از نام حوزه استفاده می‌شود که برای **us-east-۱** نشان داده شده است.

$export EC۲-URL = https://<ENDPOINT_URL>

یا

C:\> set EC۲-URL = https://<ENDPOINT_URL>

بخش بعدی توضیح می‌دهد چگونه توسعه‌دهندگان می‌توانند از APIهای EC۲ و S۳ برای تنظیم یک برنامه‌ی وب برای پیاده‌سازی یم پورتال تاسیس ساده مانند پورتال Pustak استفاده کنند. لازم است که ما درک بیشتری درباره‌ی این که منبع مصرفی چیست و هم‌چنین پارامترهایی که برای هر منبع تنظیم می‌شوند داشته باشیم، که این‌ها در بخش بعدی توصیف می‌شود.

منابع محاسباتی EC۲

در این بخش خلاصه‌ای درباره‌ی منابع محاسباتی در دسترس روی EC۲ داریم.

منابع محاسباتی: منابع محاسباتی در دسترس روی EC۲ ، که به آنها نمونه‌های EC۲ می‌گویند، شامل ترکیب قدرت محاسباتی و منابعی مانند حافظه است. آمازون قدرت محاسباتی نمونه‌های EC۲ را اندازه‌گیری می‌کند که این محاسبات بر حسب بخش‌های محاسبه‌ی EC۲ است. بخش محاسبه‌ی

EC۲ (CU) یک مقیاس استاندارد قدرت محاسباتی است به گونه‌ای که بایت‌ها مقیاس استاندارد مخزن هستند. یک EC۲ CU قدرت محاسباتی به اندازه ی ۱,۰–۱,۲ GHz پردازنده‌ی Opteron یا پردازنده‌ی Xeon را فراهم می‌کند. از این رو، اگر توسعه‌دهنده یک منبع محاسباتی EC۲ CU۱ را درخواست داشته باشد، و منبع روی پردازنده‌ی ۲,۴ GHz قرار گرفته باشد، آنها ۵۰٪ از CPU را خواهند گرفت. این اندازه‌گیری به توسعه‌دهندگان امکان درخواست مقدارهای استاندارد قدر CPU بدون توجه به سخت‌افزار فیزیکی را می‌دهد. نمونه‌های EC۲ که آمازون برای بیشتر برنامه‌ها پیشنهاد می‌دهد متعلق به نمونه‌های استاندارد است. خصوصیات نمونه‌های استاندارد در جدول E۴,۱ نشان داده شده است، "انواع نمونه‌ی استاندارد EC۲ " . توسعه‌دهنده می‌تواند یک منبع محاسباتی را پیشنهاد دهد (از یکی از نوع‌های انواع نمونه‌های جدول، برای مثال نمونه‌ی محاسبات کوچک، که خصوصیاتش نشان داده شده است). شکل E۴,۱۰ نشان می‌دهد ما چگونه می‌توانیم این را با کنسول AWS انجام دهیم.

جدول E۴-۱ انواع نمونه استاندارد EC۲

پلت فرم	مخزن محلی	حافظه	ظرفیت محاسبات	نوع نمونه
۳۲ بیت	۱۶۰ گیگا بایت	۱/۷ گیگا بایت	یک هسته مجازی یک CU	کوچک
۶۴ بیت	۸۵۰ گیگا بایت	۷/۵ گیگا بایت	دو هسته مجازی، هر کدام دوCU	بزرگ
۶۴ بیت	۱۶۹۰ گیگا بایت	۱۵ گیگا بایت	۴ هسته مجازی، هر کدام دو CU	خیلی بزرگ

دیگر نمونه‌های در دسترس در آمازون برای برنامه‌هایی که به حافظه یا دیتابیس‌های زیادی نیاز دارند، مناسب است؛ نمونه‌هایی که به CPU زیادی نیاز دارند برای برنامه‌های رایانش گرا هستند؛ نمونه‌های رایانش خوشه‌ای برای رایانش با کارآیی بالا مناسب است، و هم‌چنین برای نمونه‌های GPU خوشه‌ای، که شامل بخش‌های پردازشی گرافیکی برای برنامه‌های HiPC است که به GPUها نیاز دارند.

نرم‌افزار: آمازون ترکیب‌های استاندارد خاصی از سیستم عامل و نرم‌افزار کاربردی دارد که به صورت تصاویر ماشینی آمازون[۴۹] هستند. AMI مورد نیاز در هنگام درخواست نمونه‌ی EC۲ باید مشخص شود. AMIای که روی یک نمونه‌ی EC۲ اجرا می‌شود AMI ریشه (root AMI) نامیده می‌شود. سیستم‌عامل‌های موجود در AMIها شامل بخش‌هایی از لینوکس، مانند لینوکس شرکتی Red Hat و SuSE، ویندوز سرور، و Solaris است. نرم‌افزار در دسترس شامل دیتابیس‌هایی مانند IBM DB۲، Oracle، و ویندوز SQL سرور است. نرم‌افزارهای کاربردی دیگری مانند Hadoop، Apache، Ruby on Rails نیز در دسترس هستند.

دو راه برای استفاده از نرم افزارهایی که در AMIها نیستند وجود دارد. می‌تواند یک AMI استاندارد را درخواست داد و سپس نرم‌افزار مورد نیاز را نصب کرد. این AMI می‌تواند بعنوان یک AMI در دسترس در آمازون ذخیره شود. روش دیگر وارد کردن یک تصویر Vmware بعنوان یک AMI با استفاده از دستور ec۲-import-instance و ec۲-import-disk-image است. برای جزئیات بیشتر می‌توانید به *http://aws.amazon.com/ec۲/faqs/ مراجعه کنید.*

فضاها و نواحی در دسترس: نواحی که EC۲ پیشنهاد می‌دهد، مانند فضاهایی هستند که S۳ پیشنهاد می‌دهد. در یک ناحیه، چندین فضای در دسترس وجود دارد، که هر فضای در دسترس متناظر با یک دیتاسنتر مجازی است که از دیگر فضاهای در دسترس ایزوله شده است. بنابراین شرکتی که می‌خواهد نمونه‌های رایانش EC۲ خودش را در Europe داشته باشد می‌تواند " Europe" را در هنگام ایجاد نمونه‌های EC۲ انتخاب کند. با ایجاد دو نمونه در دو فضای در دسترس متفاوت، شرکت می‌تواند پیکربندی با دسترسی بالایی را داشته باشد که تلورانس خطا در هر فضای در دسترس دارد.

متعادل کردن بار و مقیاس‌کردن: EC۲ متعادل‌کننده‌ی بار ارتجاعی را فراهم می‌کند که سرویسی است که بار را در میان چندین سرور متعادل می‌کند. سیاست متعادل کردن بار پیش فرض به گونه‌ای است که تمتم درخواست‌ها را مستقل می‌کند. با این حال امکان‌پذیر است که نشست‌های مبتنی بر زمان‌بند و تحت کنترل نرم‌افزار وجود داشته باشد، که به وسیله‌ی آن درخواست‌های متوالی از یک کلاینت به یک سرور مسیردهی می‌شوند که این براساس زمان و جهت نرم‌افزار انجام می‌شود. متعادل‌کننده‌ی بار هم‌چنین می‌تواند تعداد سرورها را افزایش یا کاهش دهد، که این بسته به بار است. این نیز بعنوان یک سیاست شکست مورد استفاده قرار می‌گیرد، زیرا شکست یم سرور توسط

[۴۹] *Amazon Machine Images* (AMIs)

متعادل‌کننده‌ی بار ارتجاعی کشف می‌شود. اگر بار روی سرور باقی مانده زیاد باشد، متعادل کننده‌ی بار ارتجاعی می‌تواند نمونه‌ی سرور جدیدی را راه‌اندازی کند.

نکته: منابع ذخیره‌سازی EC۲

- S۳ آمازون: فروشگاه اشیاء با دسترسی بالا
- سرویس بلاک ارتجاعی: مخزن بلاک پایدار
- مخزن نمونه: مخزن بلاک زودگذر

منابع ذخیره‌سازی EC۲ همانطور که قبلا هم ذکر شد، منابع رایانشی می‌توانند همراه با منابع شبکه و مخزن مورد استفاده قرار بگیرند. استفاده از فایل‌های S۳ مانند دسترسی به سرور HTTP است. با این حال اغلب یک برنامه چندین دیسک ورودی و خروجی را انجام می‌دد، و به دلیل کارآیی یا دلایل دیگر ما باید روی پیکربندی مخزن نیز کنترل داشته باشیم. در این بخش توصیف می‌کنیم که چگونه می‌توانیم منابعی را که به نظر دیسک‌های فیزیکی برای سرورEC۲ هستند را پیکربندی می‌کنیم (منابع ذخیره‌سازی بلاک نامیده می‌شوند). دو نوع منبع ذخیره‌سازی بلاک وجود دارد: سرویس بلاک ارتجاعی و مخزن نمونه.

سرویس بلاک ارتجاعی (EBS):

مانند همان روشی که S۳ سرویس‌های ذخیره‌سازی فایل را فراهم می‌کند، EBS نیز سرویس ذخیره‌سازی بلاک را برای EC۲ فراهم می‌کند. می‌توان یک حجم مشخصی از سایز دیسک EBS را فراهم کرد و این حجم را به یک یا چندین نمونه‌ی EC۲ ایندکسکرد که این با استفاده از ID نمونه‌ی برگردانده شده در طول ایجاد حجم انجام می‌شود. برخلاف این که نمونه‌های EC۲ در طول ایجاد نمونه‌ی EC۲ انتخاب می‌شوند، حجم EBS از نمونه‌های EC۲ استقلال دارد، که این برای ماندگاری داده‌ها که بعدا با جزئیات درباره‌ی آن صحبت می‌شود ضروری است.

مخزن نمونه: هر نمونه‌ی EC۲ یک مخزن محلی دارد که می‌تواند بعنوان بخشی از منبع محاسباتی پیکربندی شود (شکل E۴,۱۰)؛ که به آن مخزن نمونه گفته می‌شود. جدول E۴,۲ افراز پیش فرض مخزن نمونه‌ی مرتبط با هر نمونه‌ی EC۲ را برای نوع نمونه ی استاندارد نشان می‌دهد. این مخزن نمونه کوتاه مدت[۵۰] است- یعنی تا زمانی که نمونه‌ی EC۲ وجود دارد آن نیز وجود دارد و می‌تواند به هر نمونه‌ی EC۲ ایندکسشود. بعالوه اگر نمونه‌ی EC۲ خاتمه یابد، مخزن نمونه نیز متوقف

[۵۰] ephemeral

می‌شود. برای غلبه بر این محدودیت، توسعه‌دهنده‌ها می‌توانند از EBS یا S۳ برای مخزن پایدار و به اشتراک‌گذاری استفاده کنند.

جدول E۴-۳ : مقایسه ی نمونه مخزن EBS		
مخزن EBS	مخزن نمونه	
مستقل از نمونه های EC۲ ایجاد شده است.	هنگامی که نمونه ی EC۲ به صورت خودکار ایجاد می شود، تشکیل می شود.	ایجاد
می تواند بین نمونه های EC۲ به اشتراک گذاشته شود.	می تواند به نمونه EC۲ ای که با آن تشکیل شده است ایندکس شود.	اشتراک گذاری
به صورت پیش فرض به نمونه های EC۲ ایندکس نشده است.	به صورت پیش فرض به نمونه های S۳ ایندکس شده است و می تواند به نمونه های EBS نیز ایندکسشود.	ضمیمه
حتی اگر نمونه EC۲ نیز خاتمه یابد، ثابت است.	ثابت نیست، اگر نمونه ی EC۲ خاتمه یابد، خاتمه می یابد.	دوام
می تواند به S۳ اسنپ شات شود.	می تواند به S۳ اسنپ شات شود.	اسنپ شات

نمونه‌ی AMI، فایل‌های پیکربندی، و هر فایل پایدار دیگری می‌تواند در S۳ ذخیره شود، و در طول پردازش تصویر لحظه‌ای داده‌ها می‌تواند برای مدتی گرفته شود و به S۳ ارسال شود. اگر داده‌ها نیاز به به اشتراک‌گذاری داشته باشند، این می‌تواند از طریق فایل‌های ذخیره شده در S۳ انجام شود. یک مخزن EBS می‌تواند به نمونه‌ی دلخواه پیوست شود.

جدول E۴,۲ برخی از تفاوت‌های اصلی و شباهت‌های بین دو نوع مخزن را خلاصه می‌کند.

نمونه‌های پشتیبانی شده توسط S۳ در مقابل نمونه‌های پشتیبانی شده توسط EBS

EC۲ رفتار منابع را رایانش و ذخیره‌سازی می‌کند که وابسته به ریشه‌ی AMI برای نمونه‌ای است که در Amazon S۳ یا سرویس بلاک ارتجاعی آمازون ذخیره شده است. این نمونه‌ها، نمونه‌ی *S۳-backed* و *EBS-backed* گفته می‌شود. در یک نمونه‌ی *S۳-backed*،ریشه‌ی AMI در S۳ ذخیره می‌شود، که آن مخزن فایل است. از این رو آن باید در دستگاه ریشه در نمونه‌ی EC۲ پیش از این که نمونه‌ی EC۲ بتواند بوت شود، کپی شود. با این حال، به دلیل این که مخزن نمونه دائمی نمی‌باشد، هر تغییری که روی نمونه‌ی *S۳-backed* ایجاد شود، فراتر از طول عمر نمونه پایدار نخواهد ماند. بعلاوه از آنجایی که مخزن نمونه به صورت پیش فرض به یک نمونه‌ی *S۳-backed* پیوست داده می‌شود، مخزن نمونه به صورت پیش فرض به نمونه های *EBS-backed* پیوست داده نمی‌شود.

منابع شبکه‌سازی EC۲

علاوه بر منابع رایانش و ذخیره‌سازی، منابع شبکه‌سازی نیز برای برنامه ها مورد نیاز می‌باشند. برای شبکه‌سازی بین نمونه‌های EC۲ ، EC۲ آدرس عمومی و خصوصی را پیشنهاد می‌دهد. آن هم‌چنین یک سرویس‌های DNS را برای مدیریت نام‌های DNS مرتبط با این آدرس‌های IP پیشنهاد می‌دهد. دسترسی به این آدرس‌های IP توسط سیاست‌هایی کنترل می‌شود. ابر خصوصی مجازی برای فراهم‌سازی ارتباط امن بین اینترانت و شبکه‌ی EC۲ مورد استفاده قرار می‌گیرد.هم‌چنین ما می‌توانیم که زیرشبکه‌ی کاملا منطقی را ایجاد ایحاد کنیم و آن را با قوانین فایروال خودش نمایش دهیم. یک ویژگی جال دیگر EC۲ آدرس‌های IP ارتجاعی است، که مستقل از نمونه‌ها هستند، این ویژگی می‌تواند برای پشتیبانی شکست سرورها بکار گرفته شود. ویژگی‌های پیشرفته و این که برای تنظیم شبکه چگونه بکار گرفته می‌شوند در بخش بعدی ذکر می‌شود.

نکته: شبکه‌سازی EC۲

- آدرس‌های IP عمومی و خصوصی برای هر نمونه
- آدرس‌های IP ارتجاعی غیر مرتبط با نمونه‌ها
- مسیردهی DNS ۵۳ که URLهای ساده را فعال می‌کند(برای مثال www.mywebsite.com)
- گروه‌های امنیتی برای سیاست‌های امنیتی شبکه‌سازی

آدرس‌های نمونه: هر نمونه‌ی EC۲ دو آدرس IP دارد که به آن مرتبط است: آدرس IP عمومی و آدرس IP خصوصی. آدرس IP خصوصی و نام DNS می‌تواند تنها در ابر EC۲ حل شود. برای ارتباط بین نمونه‌های EC۲ ، آدرس‌های IP داخلی کارآمدتر هستند، زیرا پیام‌ها بعد از آن کاملا به داخل شبکه‌ی آمازون فرستاده می‌شوند. آدرس IP عمومی و نام DNS می‌تواند برای ارتباط در خارج از ابر آمازون مورد استفاده قرار بگیرد.

آدرس‌های IP ارتجاعی: این آدرس‌های IP مستقل از نمونه هستند اما مرتبط با حساب EC۲ آمازون خاصی هستند و می‌توانند به صورت پویا به هر نمونه‌ای انتساب یابند. از این رو آنها برای پیاده‌سازی failover مفید هستند. در اثر شکست یک نمونه‌ی EC۲، آدرس IP ارتجاعی می‌تواند به صورت پویا به نمونه‌ی EC۲ انتساب یابد. برخلاف آدرس‌های IP نمونه، آدرس‌های IP ارتجاعی به صورت اتوماتیک تخصیص می‌یابند؛ آنها در زمان مورد نیاز ایجاد می‌شوند.

مسیردهی ۵۳: شرکت‌ها ممکن است علاقمند به ایجاد مکان یکنواخت منبع (URL) به صورت www.myenterprise.com برای نمونه‌های EC۲ باشند. این به صورت پیش فرض امکان‌پذیر نمی‌باشد، زیرا نمونه‌های EC۲ در حوزه‌ی Amazon.com می‌باشند. مسیردهی ۵۳ یک سرور DNS است که می‌تواند برای ارتباط آدرس IP ارتجاعی یا آدرس IP عمومی با نامی به صورت www.myenterprise.com مورد استفاده قرار بگیرد.

گروه‌های امنیتی: برای امنیت شبکه، تعریف سیاست‌های امنیت شبکه‌سازی رایج است که از طریق آن پورت‌هایی که هر ماشینی که می‌تواند در دسترس قرار بگیرد و یا IPآدرس‌هایی که می‌توانند به سرور دسترسی داشته باشند را محدود می‌کنند. همین می‌تواند برای نمونه‌های EC۲ با استفاده از گروه‌های امنیتی مورد استفاده قرار بگیرد، به طور خلاصه قبلا هم ذکر شده است. هر گروه امنیتی مجموعه‌ای از سیاست‌های امنیت شبکه است. گروه‌های امنیتی متفاوتی باید برای انواع سرورهای مختلف ایجاد شوند؛ برای مثال، گروه امنیتی سرور وب می‌تواند تعیین کند که پورت ۸۰ برای اتصال‌های ورودی باز است. گروه امنیتی پیش فرض، در ایجاد نمونه‌ی EC، به نمونه‌ها امکان اتصال هر آدرس IP خارجی را می‌دهد اما امکان اتصال‌های ورودی را نمی‌دهد.

ابر خصوصی مجازی: شرکت‌هایی که نیاز به کنترل بیشتری روی پیکربندی شبکه‌ای خود دارند، می‌توانند از یک ابر خصوصی مجازی استفاده کنند. مثال‌های ویژگی‌های شبکه‌سازی پیشرفته‌ای که توسط VPCها پیشنهاد داده می‌شود به صورت زیر است:

- توانایی تخصیص آدرس‌های IP خصوصی و عمومی به نمونه‌ها از هر محدوده‌ی آدرس

- توانایی تقسیم آدرس‌ها به زیرشبکه ها و کنترل مسیردهی بین شبکه‌ها
- توانایی اتصال شبکه‌ی EC۲ با یک اینترانت با استفاده از تونل VPN

جزئیات VPCها فراتر از حوزه‌ی این کتاب است و می‌توانید آن را در ابر خصوصی مجازی آمازون[۵۱] پیدا کنید.

مثال ساده‌ای از EC۲: تنظیم یک سرور وب

اکنون به اصطلاحات و مفاهیم توصیف شده در دو بخش قبلی می‌پردازیم و مثال ساده‌ای از ایجاد وب سرور را بیان می‌کنیم. سرور وب بعنوان یک نمونه‌ی پشتیبانی شده توسط EBS ایجاد می‌شود، که این برای جلوگیری از لزوم داشتن بک‌آپ‌گیری دوره‌ای مخرن در S۳ است.

این فرآیند به چهار گام تقسیم می‌شود:

۱. انتخاب AMI برای نمونه
۲. ایجاد نمونه‌ی EC۲ و نصب سرور وب
۳. ایجاد حجم EBS برای داده‌ها، مانند فایل‌های HTML و غیره
۴. تنظیم شبکه‌سازی و قوانین دسترسی

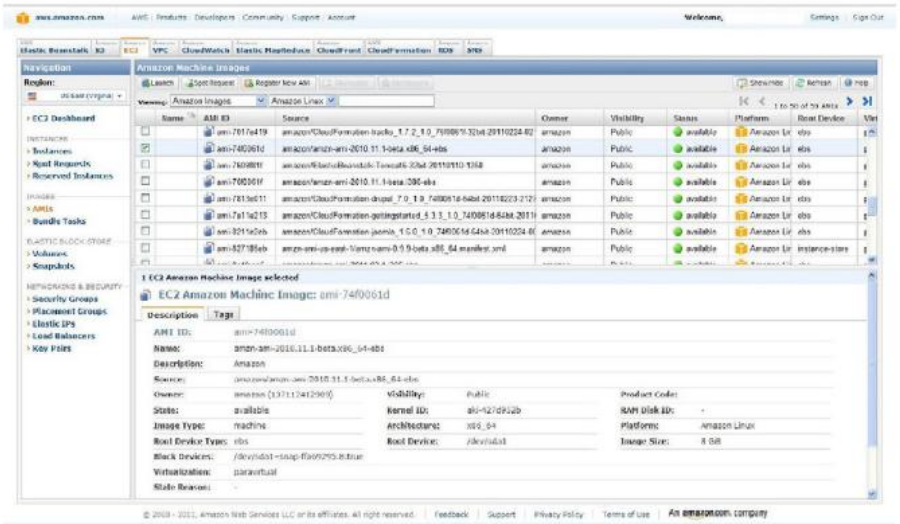
فرض می‌شود که داده‌های مورد نیاز برای سرور وب (فایل‌های HTML، اسکریپت‌ها و غیره) در دسترس هستند و به EC۲ آپلود شده‌اند. بعلاوه برای نشان دادن این که چگونه می‌تواند نرم‌افزار سفارشی را روی یک AMI استاندارد نصب کرد، فرض می‌شود که سرور وب الزامی نیز باید به EC۲ آپلود شود و سپس نصب شود.

انتخاب AMI: دستورالعمل برای ایجاد نمونه‌ی EC۲ جدید با استفاده از کنسول AWS پیش از این توصیف شده است. کاربر ممکن است آن مرحله را در اینجا فراخوانی کند . جزئیات بیشتر این فاز برای انجام عملکرد پیشرفته بعدا توصیف می‌شود.

استفاده از منوها برای انتخاب تصاویر آمازون و لینوکس آمازون لیستی از تصاویر لینوکسی را فراهم می‌کند که توسط آمازون عرضه شده است، که این در شکل ۱۳-E۴ نشان داده شده است. در این جا ستون دستگاه ریشه نشان دهنده‌ی این است که دستگاه ریشه برای تصویر EBS است یا خیر. برخی از پارامترهای مهم AMI در تگ توصیف در نیمه ی پایینی شکل است. می‌توان دید که تصویر یک تصویر لینوکس آمازون ۶۴ بیتی است که دستگاه ریشه ۱/dev/sda/ در EBS را دارد. ارزش true در فیلد دستگاه‌ها بلاک، پرچم DeleteUponTerminate است. با کلیک روی دکمه‌ی Launch ویزارد راه‌اندازی فراهم می‌شود، که این نیازمند انجام یکسری عملیات است که پیش از

[۵۱] Amazon Virtual Private Cloud, http://aws.amazon.com/vpc/

راه‌اندازی نمونه‌ی EC۲ انجام می‌شود.با این حال در این جا راهی برای ایجاد نمونه‌ی EC۲ با دستگاه ریشه ی ثابت از طریق کنسول AWS وجود ندارد. از این رو بخش بعدی توصیف می‌کند که چگونه یک نمونه‌ی EC۲ را با خط دستور راه‌اندازی کنیم.



شکل E4.13: انتخاب یک AMI

شکل ۱۳-E۴: انتخاب یک AMI

ایجاد مثالی از نمونه‌ی EC۲: دو گام مهم دیگر که در طول ایجاد نمونه انجام می‌شود به صورت زیر است:

۱. ایجاد یک جفت کلیدی که دسترسی به سرورهای EC۲ را فراهم می‌کند، که ایجاد می‌شوند

۲. ایجاد گروه‌های امنیتی که مرتبط با نمونه هستند و قوانین دسترسی شبکه‌سازی را مشخص می‌کنند.

در مثال ما از آنجایی که نمونه‌ی ایجاد شده نرم‌افزار مورد نیاز را نخواهد داشت (یعنی به صورت پیش فرض نصب نشده است)، گروه امنیتی ایجاد شده یک گروه امنیتی خالی است که امکان دسترسی به شبکه‌ی ورودی را نمی‌دهد. به همین ترتیب گروه امنیتی به حالت امکان HTTP تغییر داده می‌شود.

این جفت کلیدی از کنسول EC۲ با کلیک روی لینک **Key Pair** ایجاد می‌شود، که این طبق دستورالعمل‌ها است، و فایل‌ها را دانلود می‌کند (که در این مثال f۲.pem نام دارند.) اسکریپت زیر نشان دهنده‌ی این است که چگونه یک متغیر محیطی را که EC۲-PRIVATE-KEY نام دارد تنظیم کنیم به طوری که کلید دانلود شده برای نمونه‌های EC۲ جفت کلید پیش فرض باشد.
برای لینوکس:

```
$export EC۲-PRIVATE-KEY = ~/.ec۲/f۲.pem
$ec۲addgrp "Web Server" –d "Security Group for Web Servers"
$ec۲run ami-۷۴f۰۰۶۱d –b dev/sda۱=::false –k f۲.pem –g "Web Server"
```

برای ویندوز

```
C:\> set EC۲-PRIVATE-KEY = C:\.ec۲\f۲.pem
C:\> ec۲addgrp "Web Server" –d "Security Group for Web Servers"
C:\> ec۲run ami-۷۴f۰۰۶۱d –b "xvda=::false" –k f۲.pem –g "Web Server"
```

در این مثال، دستور ec۲addgrp که کوتاه شده‌ی ec۲-create-group است یک گروه امنیتی را که Web Server نام دارد را ایجاد می‌کند و تمام دسترسی‌ها خارجی را غیرفعال می‌کند. همانطور که قبلا نیز بیان شده است، این قانون بعدا به دسترسی فعال HTTP تغییر خواهد کرد. سپس دستور ec۲run که کوتاه شده‌ی ec۲-run-instances است برای شروع نمونه با حجم ریشه‌ی EBS ثابت مورد استفاده قرار می گیرد. این پارامتر AMI ID برای AMI انتخاب شده در شکل E۴,۱۳ است. مقدار false در پرچم b– نشان دهنده‌ی این است که پرچم DeleteUponTerminate برای این حجم false تنظیم شده است. این اشاره به این دارد که حتی اگر نمونه‌ی EC۲ خاتمه یابد نیز حجم حذف نمی‌شود. پارامترهای k– و g– جفت کلیدی را مشخص می‌کنند که می‌توانند برای ارتباط با نمونه و گروه امنیتی برای این نمونه‌ها مورد استفاده قرار بگیرد. تعداد نمونه‌هایی که باید راه‌اندازی شوند به صورت پیش فرض ۱ است. یک محدوده می‌تواند با استفاده از پارامتر instance-count– مشخص شود. جزئیات بیشتر درباره‌ی تمام گزینه‌ها برای EC۲ در مرجع دستورات ابری رایانشی ارتجاعی آمازون وجود دارد.

نام DNS برای نمونه‌ی اخیرا ایجاد شده از کنسول AWS در دسترس است. دستور ec۲-describe-instances می‌تواند برای گرفتن نام DNS عمومی نمونه نیز بکار گرفته شود. Ssh، PuTTY یا اتصال‌های دسکتاپ کنترل از راه دور می‌توانند برای وارد شدن به نمونه یا دانلود نرم‌افزاری که باید نصب شود بکار گرفته شوند. بعد از نصب نرم‌افزار اضافی، تصویر می‌تواند روی EBS بعنوان یک AMI با استفاده از دستور ec۲-create-instance ذخیره شود. پارامتر instanceId نمونه‌ی ID برای نمونه‌ی EC۲ است، و این دستور AMI ID برای EBS AMI جدیدا ایجاد شده را برمی‌گرداند. این مراحل در اسکریپت زیر نشان داده شده‌اند:
برای لینکوس

```
$ec۲din
$ssh –i f۲.pem instance-id
$ec۲-create-instance –n ‘‘Web Server AMI’’ instanceId
```

برای ویندروز

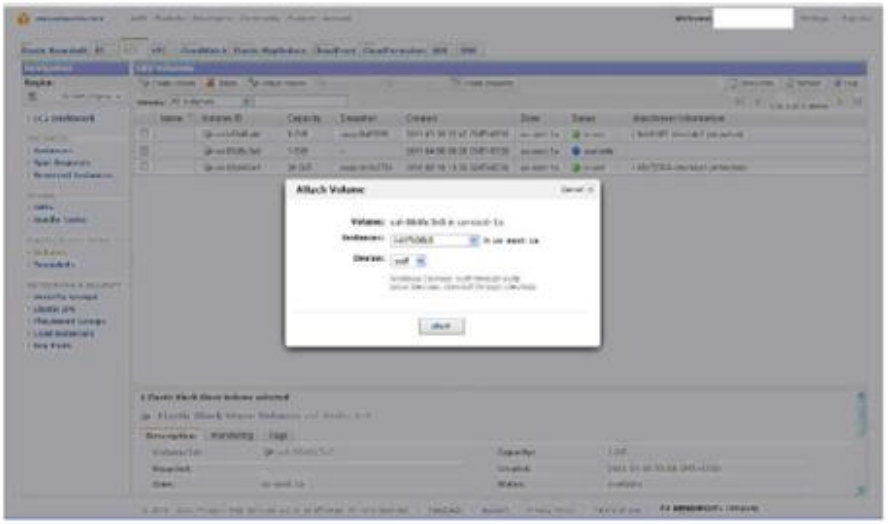
```
C:\>ec۲-describe-instances
C:\putty
C:\>ec۲-create-instance –n “Web Server AMI” instanceId.
```

پیوست یک حجم EBS

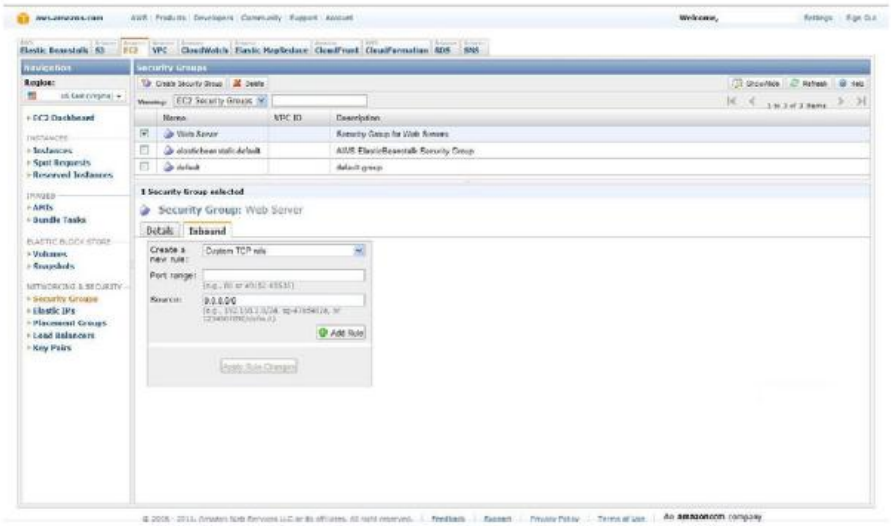
از آنجایی که صفحات HTML که از پورتال وب تامین می‌شود باید ثابت باشند، باید یک حجم EBS برای نگهداری صفحات HTML که باید از طریق سرور وب بکار گرفته شوند ایجاد شود. حجم‌های EBS می‌توانند از کنسول EC۲ با کلیک روی لینک **Volumes** ایجاد شوند. این کار یک لیستی از تمام حجم‌های EBS را که اخیرا کاربر صاحب آنها شده است را فراهم می‌کند. کلیک روی دکمه‌ی **Create Volume** اسکرین شکل E۴-۱۴ را فراهم می‌کند، که سایز حجم مورد نیاز می‌تواند پیش از ایجاد مشخص شود.
حجم جدیدی که ایجاد شده است در شکل اسکرین Volumes نشان داده شده است و حالتش نیز فعال است (شکل E۴-۱۵). با کلیک روی دکمه‌ی **Attach Volume** نیز نام دستگاه را نیز می‌تواند بدست آورد (xvdf به xvdp برای ویندوز، /dev/sdf to /dev/sdp برای لینوکس). بعد از انتخاب

مناسب، با کلیک روی دکمه‌ی **Attach** به صورت مجازی حجم را به نمونه‌ی انتخاب شده ایندکسمی‌کند. در این مرحله، نمونه‌ی EC۲ ایجاد می‌شود، سرور وب نصب شده است، و یک مخزن ثابت مجزا روی EBS ایندکسمی‌شود.

شکل E۴-۱۴ : ایجاد یک حجم EBS



شکل E۴-۱۵: ایندکس یک حجم EBS به نمونه EC۲



شکل E4.16 : تغییر یک گروه امنیتی

شکل ۱۶-E۴: تغییر یک گروه امنیتی

فعال کردن دسترسی به سرور وب: از آنجایی که سرور وب برای عملیات آماده است، دسترسی خارجی به آن می‌تواند اکنون فعال شود. با کلیک روی لینک **Security Groups** در قسمت چپ کنسول EC۲ یک لیستی از تمام گروه‌های امنیتی در دسترس بدست می‌آید. شکل ۱۶-E۴ نشان‌دهنده‌ی گروه‌های امنیتی در دسترس، که شامل گروه‌های اخیرا ایجاد شده سرور وب و دو گروه پیش فرض است. با کلیک روی تب **Inbound**، می‌تواند قوانینی را وارد کنیم که نوع ترافیک فعال را مشخص می‌کند. شکل ۱۶-E۴ نشان می‌دهد که چگونه باید قوانین جدیدی را که امکان ترافیک روی پورت ۸۰ از تمام آدرس‌ها را فراهم می‌کند را اضافه کنیم (که با آدرس IP صفر مشخص شده است). یک آدرس IP مشخص می‌تواند برای فعال کردن یک آدرس IP مشخص تعیین شود. با کلیک روی دکمه‌ی **Add Rule** این قانون جدید اضافه می‌شود. بعد از این که تمام قوانین اضافه می‌شوند، با کلیک روی دکمه‌ی **Apply Rule Changes** قوانین جدید فعال می‌شوند.

این استقرار سرور وب ساده روی EC۲ و EBS را کامل می‌کند.

مدل‌های سرویس دیگر

همانطور که قبلا هم بیان شد، ابر مجموعه‌ای از سرویس‌ها است. همواره سرویس‌های جدیدی برای فراهم کردن نیازهای کاربر ایجاد می‌شوند. این سرویس‌ها منجر به مدل جدیدی می‌شوند که علاوه بر سه مدل سنتی است. اگرچه تعدادی مدل‌های سرویس دیگری هم می‌باشد، ما در این‌جا تنها دو مدل را پوشش می‌دهیم: دیتابیس بعنوان سرویس و دسکتاپ بعنوان سرویس.

دیتابیس بعنوان سرویس (DbaaS)

DbaaS سازمان‌های پلت‌فرم دیتابیسی را ایجاد می‌کند که برای ذخیره‌ی داده‌های آنها مورد استفاده قرار می‌گیرد. بسیاری از ارائه‌دهنده‌های PaaS نیز سرویس‌های دیتابیسی را فراهم می‌کنند، اما بسیاری از این سازمان‌های نمونه نیازی به توسعه‌ی پلت‌فرم ندارند؛ آنها تنها به مکانی برای ذخیره داده‌ها نیاز دارند. در این موارد، گزینه‌ی DbaaS انتخاب مناسبی است. اگرچه هزینه‌های ذخیره‌سازی کاهش یافته است، اما این هزینه هنوز هم زیاد است. یک پیاده‌سازی DbaaS پلت‌فرم دیتابیس و مخزن مورد نیاز شما را با هزینه‌ی کمتر از پیاده‌سازی داخلی آن فراهم می‌کند.

دسکتاپ بعنوان سرویس

DaaS یکی از مدل‌های سرویس جدید است. به طور کلی، DaaS برای کاربران یک دسکتاپ مجازی را فراهم می‌کند که می‌تواند برای انجام محاسبات دسکتاپ مورد استفاده قرار بگیرد. شرکت‌ها در حال تلاش برای کشف بهترین راه برای فراهم کردن این نوع از سرویس هستند و این که چه ویژگی‌ها و عملکردی آنها نیاز دارند. یکی از بزرگترین سوالات این است که آیا دسکتاپ‌های اختصاصی کاربری بهتری دارند و یا دسکتاپ‌های متصل شده.

بکارگیری ذخیره‌سازی (ذخیره‌سازی بعنوان سرویس)

داده‌ها نیروی حیاتی شرکت است. شرکت‌ها الزامات متفاوتی را برای داده‌ها دارند، از جمله داده‌های ساخت‌یافته در دیتابیس رابطه‌ای که کسب‌وکار الکترونیکی را قدرتمند می‌کند و داکیومنت‌هایی که داده‌های غیر ساخت‌یافته درباره‌ی فرآیندها، برنامه‌ها و دیدها را جمع می‌کند . هم‌چنین شرکت‌ها ممکن است به ذخیره‌سازی اشیاء نیز ممکن است نیاز داشته باشند، مانند یک آلبوم عکس آنلاین یا

پلت‌فرم ویرایش داکیومنت مشارکتی. بعلاوه برخی از داده‌ها ممکن است محرمانه باشند و باید به خوبی محافظت شوند؛ داده‌های دیگر باید به راحتی قابل به اشتراک‌گذاری باشند. در این موارد، داده‌های حیاتی کسب‌وکار باید امن باشد و در هنگام شکست نرم‌افزاری و سخت‌افزاری در هنگام نیاز مورد دسترس باشد.

نکات: سرویس‌های ذخیره‌سازی آمازون

- سرویس ذخیره‌سازی ساده (S۳): یک مخزن شی
- DBساده: مخزن کلید-مقدار
- سرویس دیتابیس رابطه‌ای (RDS): نمونه‌ی MySQL
-

سرویس ذخیره‌سازی ساده‌ی آمازون (S۳)

سرویس‌های وب آمازون، از Amazon.com، دنباله‌ای از محصولات سرویس ابری دارند که بسیار مشهور شده‌اند و بیشترین جستجو را برای استاندارد *de facto* برای تحویل IaaS دارند. شکل E۴,۱۷ اسکرینی از AWS را نشان می‌دهد، که نمایش‌دهنده‌ی محصولات IaaS متفاوت در تب‌های چندگانه است (S۳، EC۲، CloudWatch). آمازون S۳ بیشترین قابلیت اطمینان، در دسترسی، مقیاس پذیری و ذخیره‌سازی سریع را در ابر برای ذخیره‌سازی و بازیابی مقدار زیادی از داده‌ها از طریق سرویس‌های وب ساده دارد.در این بخش جزئیاتی از این پلت فرم ارائه داده می‌شود و سپس مثالی از S۳ داریم. استفاده‌های پیشرفته از S۳ در بخش‌های بعدی در آمازون EC۲ ارائه می‌شود با یک مثال که چگونه APIهای S۳ می‌توانند با هم توسط توسعه‌دهنده با دیگر سرویس‌های رایانش آمازون برای تشکیل یک راه‌حل IaaS کامل مورد استفاده قرار بگیرند.

ابتدا مثالی از این که چگونه می‌توانیم از S۳ ببعنوان یک ذخیره‌سازی ابری ساده برای آپلود فایل‌ها استفاده کنیم داده می‌شود.

دسترسی S۳: به سه روش می‌توان از S۳ استفاده کرد. رایج‌ترین عملیات از طریق کنسول AWS انجام می‌شود، واسط GUI به AWS که می‌تواند از طریق http:// aws.amazon.com/console در دسترس باشد. برای استفاده از S۳ در برنامه‌ها، آمازون REST-ful API را فراهم کرده است که عملیات HTTP آشنایی مانند GET، PUT، DELETE، و HEAD دارد. هم‌چنین کتابخانه‌ها و SDKهایی برای زبان‌های متفاوتی که این عملیات را تجزیه و خلاصه می‌کنند نیز وجود دارد.

شکل E4.17
AWS Console.

شکل E۴-۱۷: کنسول AWS

نکته: روش‌های دسترسی S۳

- کنسول AWS
- RESTful API آمازون
- SDKها برای Ruby و دیگر زبان‌ها

بعلاوه از آنجایی که S۳ یک سرویس ذخیره‌سازی است، جستجوگرهای S۳ متفاوتی وجود دارند که به کاربران امکان اکتضاف حساب‌های S۳شان حتی اگر دایرکتوری هم باشد را می‌دهد. چندین امکان برای وارد کردن دستور هم وجود دارد که می‌تواند در اسکریپت‌های دسته‌ای نیز بکار گرفته شوند که در انتهای این بخش توصیف خواهند شد.

آغاز با S۳: در ابتدا با یک مورد شخصی ساده آغاز می‌کنیم. فرض کنید که یک کاربر که دایرکتوری پر از عکس‌های شخصی دارد و می‌خواهد آنها را در ابر بعنوان بک‌آپ ذخیره‌سازی کند. در این‌جا ابر چگونه در دسترس قرار خواهد گرفت:

۱. برای S۳ در http://aws.amazon.com/s۳/ ثبت نام کنید. در هنگامی که ثبت‌نام می‌کنید، **AWS Access Key** و **AWS Secret Key** بدست می‌آید. این‌ها مانند ID و رمز کاربری است که برای شناسایی تمام تراکنش‌ها با سرویس‌های وب آمازون است.
۲. به **AWS Management Console** از طریق https://console.aws.amazon.com/s۳/home وارد شوید
۳. با دادن نام و مکان جغرافیایی که می تواند ذخیره‌سازی شود یک *bucket را ایجاد کنید.* *در S۳ تمام فایل‌ها در* bucket ذخیره‌سازی می‌شوند، که مجموعه‌ای از اشیاء مرتبط را نشان می‌دهد. Bucket ها و اشیاء بعدا در بخش "سازماندهی داده‌ها در S۳ ، باکت‌ها، اشیاء و کلیدها" توصیف می‌شوند.
۴. روی دکمه‌ی **Upload** کلیک کرده و برای آپلود فایل‌ها طبق دستورالعمل پیش بروید.
۵. تصاویر و دیگر فایل‌ها اکنون به صورت ایمن در S۳ بک‌آپ گیری می‌شوند و برای به اشتراک‌گذاری با URL در دسترس هستند، البته اگر اجازه‌های درست فراهم شوند.

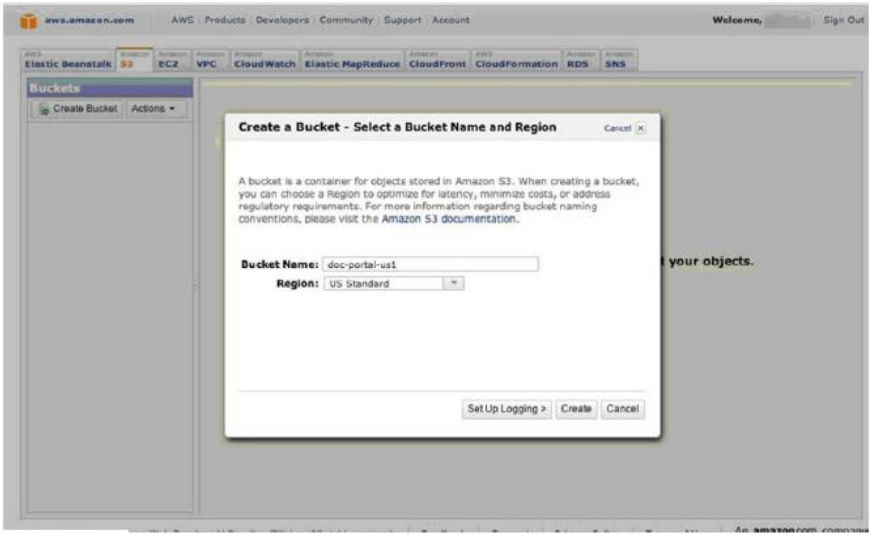
از دید توسعه‌دهنده، این می‌تواند به صورت برنامه‌ریزی شده در مواردی که نیاز به اضافه کردن این عملکرد به برنامه است، انجام می‌شود.

سازماندهی داده‌ها در S۳: باکت‌ها، اشیاء و کلیدها

فایل‌ها در S۳ اشیاء نامیده می‌شوند. به اشیاء با کلیدهایی اشاره می‌شود – اساسا یک نام مسیر انتخابی براساس نام شی است. اشیاء در S۳ در طول چندین مکان گرافیکی تکرار می‌شوند که این برای انعطاف‌پذیر کردن آنها در برابر چندین شکست است. اگر نسخه‌بندی اشیاء فعال شود، بازیابی از حذف و تغییرات ناخواسته امکان پذیر است. اشیاء S۳ می‌توانند در سایز تا ۵ ترابایت باشند و هیچ محدودیتی روی اعداد اشیائی که می‌توانند ذخیره شوند وجود ندارد. تمام اشیاء در S۳ باید در یک باکت ذخیره‌سازی شوند. باکت‌ها راهی را برای قرار دادن اشیاء مرتبط در یک مکان فراهم می‌کنند و آنها را از دیگر اشیاء جدا می‌کنند. در هر حساب ۱۰۰ باکت می‌تواند باشد و تعداد بی نهایت شی در هر باکت. هر شی یک کلید دارد، که می‌تواند بعنوان مسیری برای منبع شی در یک HTTP URL استفاده شود. برای مثال، اگر نام باکت johndoe باشد و کلید آن شی resume.doc باشد، سپس HTTP URL آن http://s۳.amazonaws.com/johndoe/resume.doc است یا http://johndoe.s۳.amazonaws.com/resume.doc . کلیدهایی که با slash جدا می‌شوند برای ایجاد طرح نام‌گذاری مانند دایرکتوری در جستجوگرهای S۳ مانند کنسول AWS

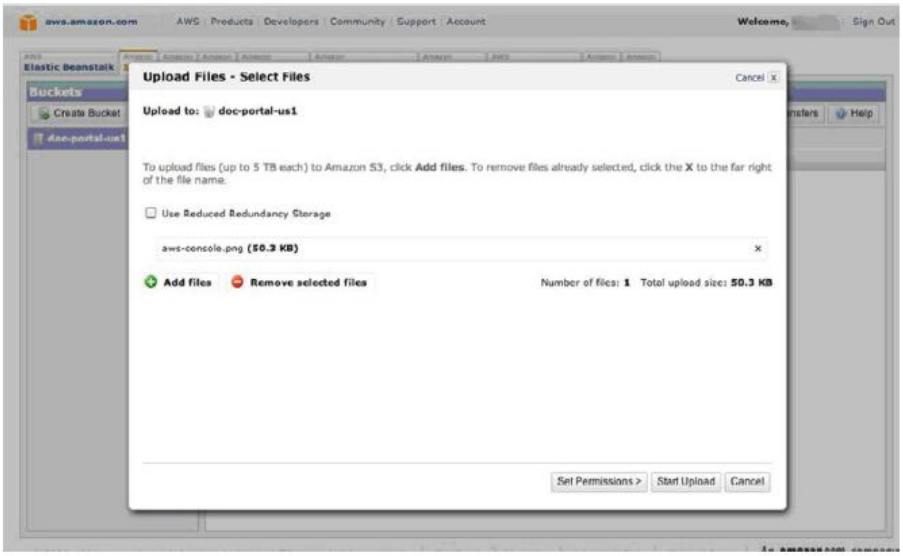
استفاده می‌شوند. برای مثال ما می‌توانیم URL مانند http://johndoe.s۳.amazon.aws.com/project۱/file۱.c ، http://johndoe.s۳.amazon.aws.com/project۱/file۲.c و http://johndoe.s۳.amazon.aws.com/project۲/ را داشته باشیم. با این حال این‌ها فایل‌هایی با کلیدهای project۱/file۱.c و ... هستند و S۳ یک سیستم فایلی سلسله مراتبی نیست. وجه داشته باشید که فضای نام[۵۲] باکت به اشتراک گذاشته می‌شود؛ ایجاد یک باکت با نامی که اکنون مورد استفاده است (توسط کاربر S۳ دیگر) امکان‌پذیر نمی‌باشد. توجه داشته باشید که وارد کردن URLهای قبلی به یک جستجوگر آن طور که انتظار می‌رود کار نخواهد کرد؛ نه تنها این مقادیر عملیاتی هستند بلکه اگر مقادیر واقعی برای کلید و باکت جایگزین شده باشند، نتیجه خطای " HTTP ۴۰۳ Forbidden " است. این به دلیل عدم وجود پارامترهای شناسایی URL است؛ اشیاء S۳ به طور پیش فرض خصوصی هستند، و درخواست‌ها باید پارامترهای تصدیق و شناسایی را داشته باشد که اثبات کند درخواست دهنده حق دسترسی به شی را دارد در غیر این صورت شی مجوزهای عمومی دارد. عموما کتابخانه‌ی کلاینت، SDK، و یا برنامه از کلید دسترسی AWS و کلید امنیتی AWS برای محاسبه‌ی امضایی که درخواست‌دهنده را شناسایی می‌کند و یا این امضا را به درخواست S۳ اضافه می‌کند، استفاده می‌کنند. برای مثال، *S۳ Getting Started Guide* در باکت awsdocs در کلید S۳/latest/s۳-gsg.pdf با مجوز خواندن بی‌نام[۵۳] ذخیره می‌شود؛ از این رو آن برای همه در http://s۳.amazonaws.com/awsdocs/S۳/latest/s۳-gsg.pdf در دسترس است.

[۵۲] namespace
[۵۳] anonymous



شکل E4.18: ایجاد bucket

شکل E۴-۱۸: ایجاد Bucket



شکل E4.19: بروزرسانی اشیاء

شکل E۴-۱۹: به روز رسانی اشیا

مدیریت S۳: در هر شرکتی، داده‌ها همراه با سیاست‌هایی هستند که مکان داده‌ها و دردسترسیشان یعنی این که چه کسی می‌تواند به آنها دسترسی داشته باشد و چه کسی نمی‌تواند را تعیین می‌کند. برای امنیت و انطباق با مقررات محلی، لازم است که حسابرسی و فعالیت‌های ورود فعال شوند و قادر به غیرفعال کردن فعالیت‌های کاربری غیرعمدی باشیم. S۳ امکان‌هایی را برای تمام این موارد فراهم می‌کند که به صورت زیر هستند:

امنیت: به دو روش کاربران می‌توانند از امنیت داده‌های S۳ را اطمینان حاصل کنند. اولا، S۳ کنترل دسترسی به اشیاء را فراهم می‌کند. کاربران می‌توانند مجوزهایی را تنظیم کنند که به دیگران امکان دسترسی به اشیاء آنها را می‌دهد. این از طریق کنسول مدیریت AWS قابل انجام است. با راست کلیک روی یک شی منوی فعالیت‌های شی فراهم می‌شود (شکل E۴-۲۰). اعطای دسترسی خواندن بی‌نام به شی آن را توسط هر فردی قابل خواند می‌کند؛ برای مثال این در محتویات پویای روی وب سایت مفید است. این با انتخاب گزینه‌ی ***Make Public*** روی منوی شی انجام می‌شود. هم‌چنین می‌توانیم دسترسی خواندن و نوشتن را به حساب‌های خاصی محدود کنیم. برای این کار گزینه‌ی ***Properties*** را انتخاب کنید و سپس وارد منوی دیگری می‌شوید که به کاربر امکان وارد کردن آیدی‌های ایمیل کاربران که به آنها دسترسی داده می‌شود را فراهم می‌کند. این هم‌چنین امکان‌پذیر است که به دیگران امکان دهیم اشیائی را به همین روش در باکت قرار دهند. یک استفاده‌ی رایج برای این اقدام مجهز کردن کلاینت‌ها به روشی است که بتوانند داکیومنت‌هایی را برای اصلاح تعیین کنند، که این‌ها بعدها در یک باکت متفاوتی نوشته می‌شوند که کلاینت مجوزهایی برای برداشتن داکیومنت اصلاح شده دارد.

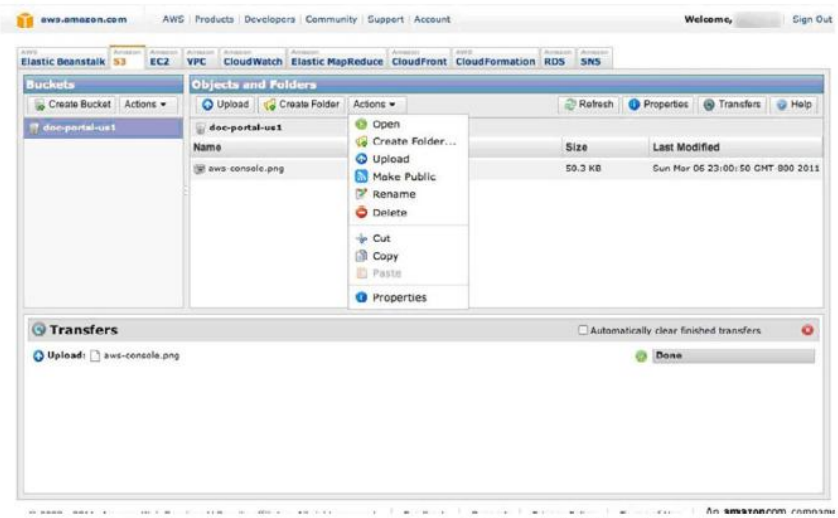
روش دیگر این است که ثبت‌های حسابرسی[۵۴] جمع‌آوری شوند. S۳ به کاربران امکان روشن کردن ورود برای یک باکت را فراهم می‌کند، که در این حالت ورودهای دسترسی برای باکت در یک باکت متفاوتی ذخیره می‌شود. این به کاربر اجازه می‌دهد که ببیند کدام حساب AWS به اشیاء دست یافته است، آدرس IP که دسترسی از آن اتفاق افتاده است. ورود می‌تواند از کنسول مدیریت AWS فعال شود (شکل E۴,۲۱). هم‌چنین ورود می‌تواند در هنگام ایجاد باکت فعال شود.

محافظت از داده‌ها: S۳ دو ویژگی برای جلوگیری در از دست دادن داده‌ها را ارائه می‌دهد. به طور پیش فرض S۳ داده‌ها را در چند دستگاه ذخیره‌سازی تکرار می‌کند. هم‌چنین می‌توان برای داده‌های

[۵۴] audit log

غیر حیاتی ذخیره‌سازی افزونه‌ی کاهش یافته[۵۵] (RRS) را درخواست دهیم. داده‌های RRS دوبار تکرار می‌شوند و برای باقی ماندن بعد از یک شکست طراحی شده‌اند.

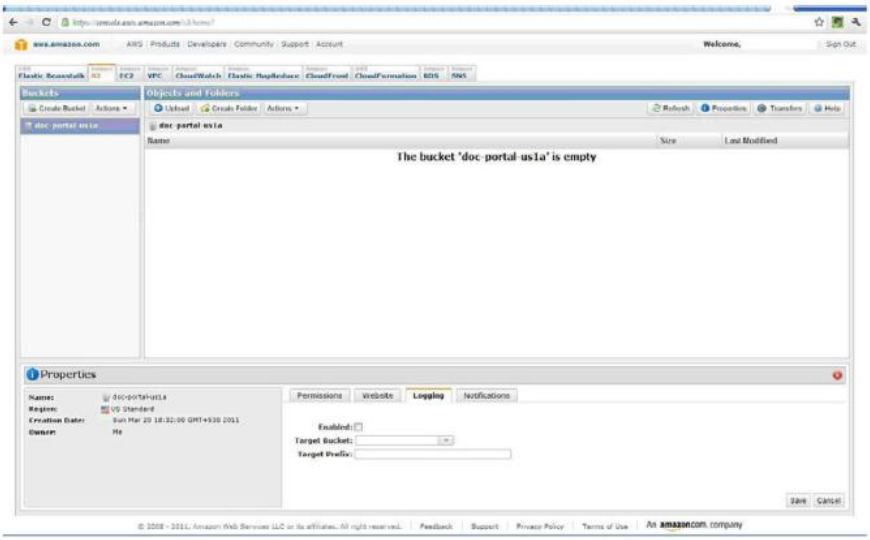
لازم به ذکر است که آمازون ثبات میان تکرارها را تضمین نمی کند؛ یعنی اگر سه تکرار از داده وجود داشته باشد، برنامه‌ای که یک تکراری را که بروزرسانی تاخیری را دارد می‌خواند ، می‌تواند ورژن قدیمی‌تر داده را بخواند.



شکل E4.20: آمازون S3 : انجام اقداماتی روی اشیاء

شکل ۲۰-E۴: آمازون S۳ (انجام اقداماتی روی اشیا)

[۵۵] Reduced Redundancy Storage



شکل E4.21: ورود Amazon S3 bucket

شکل ۲۱-E۴ : ورود Amazon S۳ bucket

نسخه‌بندی[۵۶]

اگر در یک باکت نسخه‌بندی امکان‌پذیر باشد، S۳ به صورت اتوماتیک تاریخچه‌ی تمام اشیاء را در یک باکت از آن زمان به بعد ذخیره می‌کند. شی می‌تواند به ورژن قبلیش مجددا ذخیره شود، و حتی حذف‌ها می‌توانند دوباره برگردانده شوند. این تضمین می‌کند که داده‌ها هرگز به طور تصادفی از دست نمی‌روند.

نواحی: برای کارآیی ، قانون‌مندی و دیگر دلایل، بهتر است که داده‌های S۳ در مکان‌های جغرافیایی مشخصی اجرا شوند. این می‌تواند در سطح باکت از طریق انتخاب ناحیه‌ای که در آن باکت در طول ایجادش ذخیره می‌شود، صورت گیرد. ناحیه متناظر با یک محدوده‌ی جغرافیایی بزرگ است، مانند آمریکا یا اروپا. لیست فعلی نواحی در وب سایت S۳ است[۵۷].

اشیاء بزرگ و آپلودهای چند بخشی: محدودیت اندازه‌ی شی برای S۳ به اندازه‌ی ۵ترابایت است، که این بیش از مقدار مورد نیاز برای ذخیره ی فیلم ۱۰۸۰p HD غیر فشرده شده است. اگر این مقدار

[۵۶] Versioning

[۵۷] http://aws.amazon.com/s۳

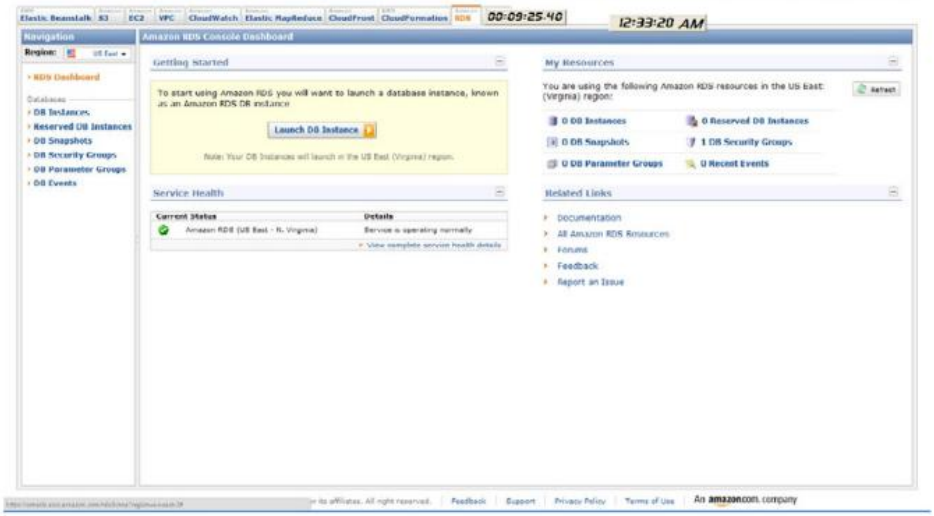
کافی نباشد، شی می تواند در قطعه‌های کوچکتری ذخیره شود و این با استفاده از داده در برنامه مدیریت می‌شود. اگرچه آمازون S۳ تراکم پهنای باند در دسترس بالایی دارد، اما آپلود اشیاء بزرگ زمان‌بر است. بعلاوه اگر آپلودی دچار شکست شود، کل شی باید دوباره آپلود شود. آپلود چندبخشی هر دو مساله را حل می‌کند. S۳ فراهم‌کننده‌ی APIهایی است که به توسعه‌دهنده امکان نوشتن یک برنامه‌ای را که یک شی بزرگ را به چندین بخش تقسیم می‌کند را می‌دهد و در این حالت هر بخش به صورت جداگانه آپلود می‌شود. این آپلودها می‌توانند موازی‌سازی شوند تا سرعتشان به ماکسیمم برسد. اگر بخشی در آپلود دچار شکست شود، تنها آن بخش باید تکرار شود. S۳ تا ۱۰۰۰۰ بخش برای هر شی را پشتیبانی می‌کند.

DB ساده‌ی آمازون: برخلاف S۳ آمازون، که شامل عملیات در سطح فایل است، DBساده یک واسط مخزن داده‌ی ساده را فراهم می‌کند که به صورت مخزن کلید-مقدار است. SDB (SimpleDB) امکان ذخیره‌سازی و بازیابی مجموعه‌ای از ویژگی‌ها براساس یک کلید را می‌دهد. استفاده از مخزن کلید-مقدار روش دیگری برای دیتابیس‌های رابطه‌ای است که از درخواست‌های مبتنی بر SQL استفاده می‌کند. این نوعی مخزن داده‌ی NoSQL است. بخش بعدی بررسی کوتاهی از SDB را فراهم می‌کند.

سازماندهی داده‌ها و دسترسی: داده‌ها در SDB به حوزه‌ها تقسیم‌بندی می‌شوند. هر آیتم در یک حوزه یک کلید منحصر به فرد دارد که باید در طول ایجاد فراهم شود. هر آیتم می‌تواند ۲۵۶ ویژگی را داشته باشد، که جفت‌های مقدار-نام هستند. براساس مدل رابطه‌ای، برای هر ردیف کلید اصلی به نام آیتم و نام‌های ستون ترجمه می‌شود و نام‌های ستون و مقدارها برای آن ردیف به جفت‌های نام-مقدار ترجمه می‌شوند. برای مثال اگر لازم باشد که اطلاعات کارمند را ذخیره کنیم، امکان ذخیره‌ی ویژگی‌های کارمند که توسط یک کلید مناسب نشان داده شده‌اند نیز وجود دارد، مانند ID کارمند. برخلاف یک سیستم مدیریت دیتابیس رابطه‌ای، ویژگی‌ها در SDB می‌توانند چندین مقدار را داشته باشند- برای مثال، اگر در یک دیتابیس محصول خرده‌فروشی، لیست ***keywords*** برای هر آیتم در کاتالوگ محصول می‌تواند بعنوان یک مقدار منفرد که متناظر با آن کلمات کلیدی ویژگی است ذخیره شوند، که انجام این کار با RDBMS بسیار پیچیده‌تر است. SDB یک زبان درخواستی را فراهم می‌کند که مشابه SQL است، هرچند روش‌هایی برای واکشی[۵۸] یک آیتم نیز وجود دارد. درخواست‌ها از این واقعیت که SDB به صورت اتوماتیک تمام ویژگی‌ها را نشان می‌دهد، استفاده می‌کنند.

۵۸: Fetch

در دسترسی SDB و مدیریت: SDB ویژگی‌هایی دارد که برای افزایش در دسترسی و قابلیت اطمینان است. داده‌های ذخیره شده در SDB به صورت اتوماتیک در مکان‌های متفاوتی برای در دسترسی بالا ذخیره می‌شوند. هم‌چنین آن به صورت اتوماتیک منابع محاسباتی را به نسبت نرخ درخواست اضافه می‌کند و به صورت اتوماتیک تمام فیلدها در دیتاست را برای دسترسی کارآمد نشان می‌دهد. SDB بی نظیر است؛ یعنی فیلدها می‌توانند به دیتاست در هنگامی که درخواست بالا می‌رود اضافه شوند.



شکل E4.22: کنسول AWS: سرویس دیتابیس رابطه ای

شکل ۲۲-E۴: کنسول AWS : سرویس دیتا بیس رابطه ای

سرویس دیتابیس رابطه‌ای آمازون (RDS): این سرویس فراهم کننده‌ی یک انتزاع پایگاه داده سنتی در ابر است، به ویژه یک نمونه‌ی MySQL در ابر. نمونه‌ی RDS می‌تواند با استفاده از تب RDS در کنسول مدیریت AWS ایجاد شود. AWS بسیاری از وظیفه‌های مدیریتی را که مرتبط با نگهداری دیتابیس برای کاربر هستند را انجام می‌دهد. دیتابیس در فواصل قابل تنظیم قابل بک‌آپ گیری است، که تناوبش می‌تواند ۵ دقیقه باشد. داده‌های بک‌آپ برای یک دوره‌ی قابل تنظیم از زمان می‌تواند حفظ شود، که این می‌تواند تا ۸ روز باشد. هم‌چنین آمازون فراهم‌کننده‌ی قابلیت اسنپ‌شات از دیتابیس در هنگام مورد نیاز است. تمام این وظایف مدیریتی می‌تواند از طریق کنسول AWS

انجام شود (شکل E۴-۲۲). هم‌چنین می‌توان یک ابزار سفارشی را که کار را از طریق APIهای RDS امازون انجام می‌دهد، توسعه داد.

خلاصه

NIST سه مدل سرویس ابری را مشخص کرده است: SaaS، PaaS، و IaaS . مدل SaaS قدیمی‌ترین است. هر مدل سرویس مزیت‌ها و معایب خودش را دارد. اما شما باید به انتخاب مدل سرویس توجه زیادی داشته باشید. چه از ارائه‌دهنده‌ی ابر استفاده کنید و چه خیر، باید همواره اطمینان ایجاد کنید که شما درباره‌ی نگهداری و کنترل سیستم و برنامه‌ها تدبیر دارید. تنها تفاوت با مدل ابر این است که ویژگی‌های مشخصی وجود دارند که ارائه‌دهنده مسئول انها است و شما باید آنها را در نظر بگیرید.

فصل ۵ تصمیم‌گیری

نکات این فصل:

- از ابر استفاده کنیم یا خیر؟
- انتخاب یک مدل سرویس ابر
- انتخاب یک مدل استقرار ابر

مقدمه

انتخاب یک سناریوی ابر درست و ارائه‌ی دهنده‌ی خوب برای اولین بار در موفقیت سازمان بسیار حیاتی است. بسته به ارائه‌دهنده، هنگامی که شما انتخاب خود را انجام می‌دهید، شما ممکن است گیر بیافتید، زیرا که انتقال داده ها به یک ارائه‌دهنده‌ی دیگر بسیار دشوار است. اگر دپارتمان IT ارائه‌دهنده‌ی درستی را انتخاب نکند، آنها نسبت به اعتبار کسب‌وکار خطر می‌کنند. کی از معایب محیط‌های ابری عمومی این است که کسب‌وکار می‌تواند از سرویس‌ها به صورت مستقیم استفاده کند. آنها نیازی به وابستگی به دپارتمان IT داخلی ندارند.

از ابر استفاده کنیم یا خیر؟

اولین گام در ارزیابی استفاده از ابر تعیین این است که شما قصد دارید چه مشکلی را حل کنید. شما می‌توانید برخی از مسائل عملیاتی و تکنیکی را حل کنید، و یا ممکن است شما تلاش در حل مسائل چگونگی پیشنهاد سرویس‌ها و قابلیت‌های جدید به مشتریان داشته باشید. شما باید تعیین کنید که سرویسی که نیاز دارید را خودتان می توانی تعیین کنید. اما اگر هم می‌توانید به این معنی نیست که شما باید خود آن را انجام دهید. برخی گمان می‌کنند که اگر آن پیشنهادی نیست که برای سازمان حیاتی باشد و یا کاری نیست که سازمان بتواند آن را به خوبی انجام دهد، پس باید به

ارائه‌دهنده‌ی سرویس انتقال یابد. این برای سرویس‌هایی که نگهداری و پشتیبانی بسیار گرانی دارند درست است.

شما باید تعیین کنید که از ارائه‌دهنده چه انتظارهایی را دارید. شما باید ارائه‌دهنده‌ای را انتخاب کنید که بتواند نیازهای شما را برآورده کند. گاهی انتظارات شما به گونه‌ای غیرواقعی است که هیچ ارائه‌دهنده‌ای نمی‌تواند آنها را برآورده کند.

یک نکته‌ی کلیدی دیگر این است که شما باید در نظر بگیرید که هر چند وقت یکبار می‌خواهید از سرویس‌ها استفاده کنید. اگر استفاده‌ی شما از آنها منظم است، بهتر است که آن را خود پیاده‌سازی کنید (هزینه‌ی کمتری دارد). در نظر داشته باشید که شما براساس استفاده‌هایتان در ابر هزینه پرداخت می‌کنید بنابراین استفاده از سرویس‌های به صورت منظم بسیار هزینه‌بر است.

انتخاب یک مدل سرویس ابر

بعد از مشخص کردن اصول این که شما چه می‌خواهید انجام دهید، شما باید مشخص کنید که کدام مدل سرویس بهتر نیازهای شما را پوشش می‌دهد. این به آن سادگی که شما فکر می‌کنید نیست. برای مثال تنها به دلیل این که شما به سرویس‌های برنامه‌های نیاز دارید به این معنا نیست که شما باید ارائه‌دهنده‌ی SaaS را انتخاب کنید. شما ممکن است ارائه‌دهنده‌ی PaaS را انتخاب کنید و برنامه‌یتان را خودتان بسازید. بنابراین مروری روی مواردی که شما در هنگام انتخاب مدل سرویس باید به آنها توجه داشته باشید خواهیم داشت.

تجربه‌ی کاربر

تجربه‌ی کاربر می‌تواند نقش مهمی را در تصمیم‌گیری داشته باشد. به طور کلی هدف نهایی شما خدمت رسانی به کلاینت‌ها است. اگر کلاینت‌های شما راضی نباشند، پیاده‌سازی شما موفق نخواهد بود.

اگر کنترل روی تجربه‌ی کاربر برای شما اهمیت دارد، مدل SaaS عمومی گزینه‌ی خوبی نمی‌باشد. در مدل SaaS شما کنترل کمی روی UI دارید. هم‌چنین شما فرصت کمی در سفارشی‌سازی برنامه برای کاربران دارید. اگر قصد داشته باشید از پیاده‌سازی PaaS یا IaaS استفاده کنید، شما کنترل کامل روی برنامه‌ها خواهید داشت. شما می‌توانید هر سفارشی‌سازی را که نیاز دارید برای برنامه را

داشته باشید. فاکتورهای دیگر مانند پهنای باند شبکه، نقش مهمی در تعیین تجربه‌ی کاربر دارند. بدون پهنای باند مناسب، سیستم کند به نظر خواهد رسید.

امنیت

در هنگام صحبت درباره‌ی ابرهای عمومی، مدل‌های سرویس ابری مختلف سطح‌های امنیت متفاوتی را پیشنهاد می‌دهند. این نشان می‌دهد چه کسی کنترل چه چیزی را دارد. دو سناریو وجود دارد: اولی امن نگه داشتن داده‌ها از تهدیدهای خارجی است و دومی امن نگه داشتن داده‌ها از تهدیدها بالقوه در سمت ارائه‌دهنده است. در محیط SaaSف ارائه‌دهنده کنترل و دسترسی کامل به تمام داده‌ها دارد، و کارهای کمی برای حفاظت از داده‌ها وجود دارد. و اما در IaaS، ارائه‌دهنده دسترسی فیزیکی به داده‌ها دارد، اما روش‌هایی وجود دارند که برای حفاظت از داده‌ها می‌توانید آنها را بکار گیرید، مانند رمزگذاری داده‌ها.

انطباق

بیشتر سازمان‌ها مقررات انطباق را دارند که باید آنها را اعمال کنند. مقررات انطباق می‌توانند مسئولیت‌هایی را برای سیستم‌های IT و زیرساخت IT قرار دهند. بسیاری از افراد از این مقررات انطباق برای نرم‌تر کردن این مسئولیت‌ها استفاده می‌کنند. هر مدل سرویس ابر براساس درجه‌ای که به شما در تطابق با مقررات تطبیق کمک می‌کند متفاوت است. در مدل SaaS ارائه‌دهنده‌ی یک نسبت بزرگ‌تری از محدوده‌ی تطبیق را فرض خواهد کرد. بسته به تطبیق مورد نیاز، ازائه‌دهنده‌ی SaaS مسئولیت را ممکن است فرض کند. سازمان شما ممکن است هنوز هم در مواردی مسئول باشد، اما ارائه‌دهنده مسئولیت ایجاد اطمینان از تطبیق سیستم‌ها را دارد. در مدل PaaS مسئولیت‌ها به اشتراک‌گذاری می‌شوند.مصرف‌کننده باید اطمینان ایجاد می‌کند که مقررات تطبیق با برنامه‌ای که ایجاد می‌شود فراهم خواهند شد. تلاش های زیادی باید برای تعیین این که پیاده‌سازی سازگار است صورت گیرد. در مدل IaaS بیشتر مسئولیت‌ها برای مشتری است. اما این یعنی در مدل IaaS مشتری بالاترین اعتماد را به مقیاس‌های سازگاری مناسبی که بکار گرفته شده‌اند دارد.

انتخاب یک مدل استقرار ابری

بعد از انتخاب مدل سرویس ابری که نیازهای شما را به بهترین شکل پوشش می‌دهد، شما باید مدل استقرار ابر را تعیین کنید. شما می‌توانید آن را میان هیبریدی، عمومی، خصوصی و گروهیی انتخاب کنید. بیشتر افراد باور دارند که مدل ابر هیبریدی مدلی است که در بیشتر سازمان‌ها مورد استفاده قرار خواهد گرفت. اما با این حال باید در نظر بگیرید که کدام برای سازمان شما بهتر است.

تجربه‌ی کاربر

ابر تجربه‌های کاربری متفاوتی را بسته به این که کدام مدل استقرار را باید انتخاب کنید، پیشنهاد می‌دهد. اگر شما ابر خصوصی را انتخاب کنید، شما کاملا روی کاربر کنترل دارید. و قادر خواهید بود تا برنامه، شبکه و در بسیاری از موارد سیستم‌های کلاینت را کنترل کنید. این به شما امکان منظم‌سازی همه‌چیز برای بهترین کارآیی و قابلیت استفاده را می‌دهد.
اما اگر شما ابر عمومی را انتخاب کنید، در بسیاری از موارد شما کنترلی روی تجربه‌ی کاربر ندارید. در محیط گروهیی یا گروهی، کنترل شما روی تجربه‌ی کاربر بسته به توافق شما با دیگر اعضای مجموعه است.

امنیت

امنیت همواره یک موضوع پیچیده است. حتی زمانی که شما با ابر کار می‌کنید پیچیده‌تر نیز می‌شود. این بیشتر مربوط به اعتماد است. بیشتر سازمان‌ها کمتر به سوم شخص اعتماد می‌کنند.

مسئولیت‌ها

مسئولیت‌ها بسته به این که مدل ابر شما چه است متفاوت است. این می‌تواند یک فاکتور کلیدی دیگر باشد. در واقع یکی از بزرگترین محرک‌های ابرهای عمومی گرایش سازمان برای کاهش مسئولیت‌های داخلیش است.
جدوول‌های زیر نشان می‌دهند که هر فرد مسئول چه چیزی در هر محیط است. جدول ۵-۱ مسئولیت‌های ارائه‌دهنده‌ی SaaS را بیان می‌کند.

جدول ۵-۱ : مسئولیت‌های SaaS توسط مدل استقرار ابری

ترکیبی	اجتماعی	خصوصی	عمومی	
متفاوت است	مشتری	مشتری	ارائه دهنده	بروزرسانی های برنامه
متفاوت است	مشتری	مشتری	ارائه دهنده	بروزرسانی ها OS
متفاوت است	مشتری	مشتری	ارائه دهنده	ذخیره‌سازی
متفاوت است	مشتری	مشتری	ارائه دهنده	آنتی ویروس ها
متفاوت است	مشتری	مشتری	ارائه دهنده	شبکه سازی
متفاوت است	مشتری	مشتری	ارائه دهنده	سخت افزار سرورهای فیزیکی
متفاوت است	مشتری	مشتری	مشتری	برنامه ی کلاینت
متفاوت است	مشتری	مشتری	مشتری	سیستم کلاینت

جدول ۵-۲: مسئولیت‌های ارائه‌دهنده ی PaaS

ترکیبی	اجتماعی	خصوصی	عمومی	
متفاوت است	مشتری	مشتری	مشتری	بروزرسانی های برنامه
متفاوت است	مشتری	مشتری	ارائه دهنده	بروزرسانی ها OS
متفاوت است	مشتری	مشتری	متفاوت است	ذخیره‌سازی
متفاوت است	مشتری	مشتری	ارائه هنده	آنتی ویروس ها
متفاوت است	مشتری	مشتری	ارائه دهنده	شبکه سازی
متفاوت است	مشتری	مشتری	ارائه دهنده	سخت افزار سرورهای فیزیکی
متفاوت است	مشتری	مشتری	مشتری	برنامه ی کلاینت
متفاوت است	مشتری	مشتری	مشتری	سیستم کلاینت

جدول ۵-۳: مسئولیت‌های ارائه‌دهنده‌ی IaaS

ترکیبی	اجتماعی	خصوصی	عمومی	
متفاوت است	مشتری	مشتری	مشتری	بروزرسانی های برنامه
متفاوت است	مشتری	مشتری	متفاوت است	بروزرسانی ها OS
متفاوت است	مشتری	مشتری	مشتری	ذخیره‌سازی
متفاوت است	مشتری	مشتری	ارائه دهنده	آنتی ویروس ها
متفاوت است	مشتری	مشتری	ارائه دهنده	شبکه سازی
متفاوت است	مشتری	مشتری	ارائه دهنده	سخت افزار سرورهای فیزیکی
متفاوت است	مشتری	مشتری	مشتری	برنامه ی کلاینت
متفاوت است	مشتری	مشتری	مشتری	سیستم کلاینت

انتخاب یک ارائه‌دهنده‌ی سرویس ابر عمومی

اگر شما تصمیم بگیرید که از ارائه‌دهنده‌ی ابر عمومی استفاده کنید، باید تعیین کنید که از کدام می‌خواهید استفاده کنید. موارد متفاوتی هستند که شما باید در ارزیابی انواع مختلف ارائه‌دهنده‌ها در نظر بگیرید. در این‌جا برخی از شاخص‌های مهم بیان شده است.

نکات انتخاب ارائه‌دهنده‌ی SaaS

در بیشتر موارد، ارائه‌دهنده‌های SaaS برنامه‌های متفاوتی را ارائه می‌دهند. این یعنی شما باید ارائه‌دهنده‌ی سرویس و برنامه را ارزیابی کنید. سوال‌های زیر را باید در هنگامی که ارائه‌دهنده‌ی SaaS را در نظر می‌گیرید از خود بپرسید:

- چگونه شارژ خواهید کرد؟
- آیا ورود و خروج داده‌های بالک می‌تواند صورت گیرد؟

- مهاجرت داده‌ها به چه صورت مدیریت می‌شود؟
- در صورت نیاز انتقال به یک ارائه‌دهنده‌ی سرویس دیگر چقدر دشوار است؟
- متوجه شوید که برای سفارشی‌سازی برنامه چه توانایی‌هایی دارید
- چه SLAهایی وجود دارند و جریمه‌ی تخطی از این SLAها چیست؟
- آیا شما می‌توانید کنترل خود را داشته باشید و متریک‌های خود را جمع‌آوری کنید؟

نکات انتخاب ارائه‌دهنده‌ی PaaS

اگرچه ارائه‌دهنده‌های سرویس پلت‌فرم‌های متفاوتی را ارائه می‌دهند، اما با PaaS باید شما متوجه می‌شوید که می‌تواند همان پلت‌فرم را از ارائه‌دهنده‌های متفاوتی دریافت کنید. در هنگامی که ارائه‌دهنده‌ی PaaS را انتخاب می‌کنید باید سوالات زیر را در نظر بگیرید:

- چگونه شارژ را انجام می‌دهید؟
- چه پلت‌فرم‌های دیتابیس، توسعه و سیستم‌عاملی را ارائه‌دهنده پیشنهاد خواهد داد؟
- SLAهای کارآیی و در دسترسی چه هستند و جریمه‌ی تخطی از آنها چیست؟
- آیا می‌توانید کنترل خود را داشته باشید و متریک‌های خود را جمع‌آوری کنید؟

نکات برای در انتخاب ارائه‌دهنده‌ی IaaS

- بیشتر ارائه‌دهنده‌های IaaS پلت‌فرم‌های زیرساختی یکسانی را پیشنهاد می‌دهند. بنابراین نکته‌ی کلیدی این است که ارائه‌دهنده‌ای را پیدا کنیم که سرویسی را پیشنهاد می‌دهد که بیشتر مورد درخواست شما است. در انتخاب ارائه‌دهنده‌ی IaaS بهتر است به سوالات زیر توجه داشته باشید. چگونه شارژ را انجام می‌دهید؟
- چه پلت‌فرم‌های دیتابیس، توسعه و سیستم‌عاملی را ارائه‌دهنده پیشنهاد خواهد داد؟
- SLAهای کارآیی و در دسترسی چه هستند و جریمه‌ی تخطی از آنها چیست؟
- آیا می‌توانید کنترل خود را داشته باشید و متریک‌های خود را جمع‌آوری کنید؟
- آیا نرم‌افزاری بعنوان بخشی از برنامه اضافه خواهد شد؟
- مهاجرت داده‌ها چگونه مدیریت می‌شود؟

فصل ۶ ارزیابی امنیت ابر: یک چارچوب امنیت اطلاعات

نکات فصل

- ارزیابی امنیت ابر
- چک لیست برای ارزیابی امنیت ابر
- متریک‌های چک لیست‌ها

مقدمه

در این فصل بحث قبلی را عملی می‌کنیم و اساس چارچوب ارزیابی امنیت ابر را ارائه خواهیم داد. آن باید از فعالیت‌هایی که ارزیابی، تصدیق یا اعتباربخشی ابر را بالا می‌برد استفاده کند. ما ابتدا مروری روی کار موجودی که در این زمینه است خواهیم داشت و سپس چک لیست‌هایی از شاخص‌های ارزیابی که محدوده‌ی فعالیت‌هایی را که باهم امنیت اطلاعات را برای رایانش ابری فراهم می‌کنند را تنظیم می‌کنیم. هدف این فصل تجهیز خواننده با مجموعه‌ای از ابزارهای سازماندهی شده‌ای است که می‌تواند برای ارزیابی امنیت ابر عمومی، خصوصی، هیبریدی و گروهیی بکار گرفته شود می‌باشد. ارزیابی امنیت ابر هیبریدی با مدیریت ارزیابی دو یا چند نمونه‌ی ابر با استفاده از مجموعه‌ای از چک لیست‌ها برای هر نمونه به خوبی انجام می‌شود. برای مثال اگر هیبریدی شامل یک ابر خصوصی و یک ابر عمومی باشد، عناصر خصوصی به سادگی با استفاده از مجموعه‌ای از چک لیست‌ها ارزیابی می‌شوند و ارزیابی عناصر عمومی با جداسازی مرزها صورت می‌گیرد.

ارزیابی امنیت ابر

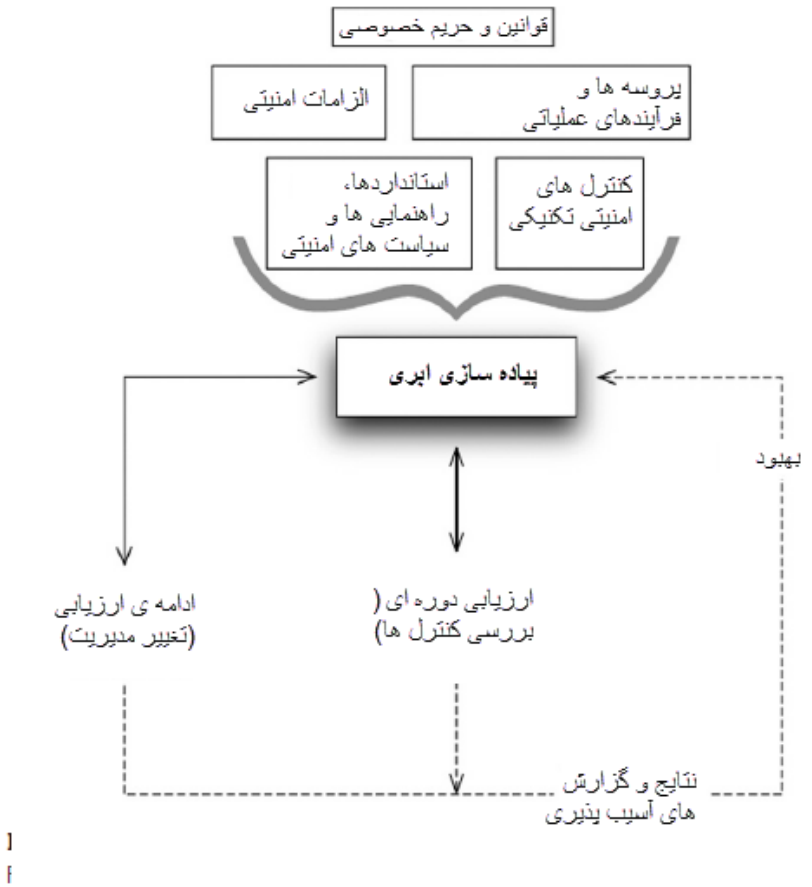
بیشتر کاربران ابر، چه ابر عمومی و چه ابر خصوصی، برای امنیت داده‌های خود ملاحظاتی را دارند. به همین صورت، صاحب و متصدی یک ابر مسئولیت‌ها را برای اطمینان از این که مقیاس‌های امنیتی قرار دارند و استانداردها و پروسه‌ها پیروی می‌شوند به اشتراک می‌گذارد. ما می‌توانیم انتظارات و مسئولیت‌های خود را توسط بیا آنها به صورت رسمی در الزامات داکیومنت شده ضبط کنیم. سیستم‌هایی که توسط سازمان‌های دولتی اجرا می‌شوند باید عموما با این الزامات NIST یا مربوطه تطبیق داشته باشند.ماتریس کنترل‌های تعهد امنیت ابری از همین روش در شرح الزامات امنیتی برای پیاده‌سازی‌های ابری استفاده می‌کند. یک نقطه‌ی شروع خوب در هنگامی که شما باید تاثیر و حضور امنیت را ابر را اندازه‌گیری کنید شامل لیستی از کنترل‌های امنیتی توصیه شده است.

برای شروع، برای کنترل‌های امنیتی در پیاده‌سازی‌های ابری دو ویژگی وجود دارد. اولی در ارتباط با حضور کنترل است. و دومی تاثیر و قدرت کنترل است. بعبارت دیگر، تنها حضور کنترل امنیتی کافی نیست؛ این کنترل باید موثر باشد. در واقع جلوتر به این تاثیر کنترل درجه‌ی اعتماد گفته می‌شود و یا دقت، که می‌تواند از این کنترل‌ها مورد انتظار باشد. برای مثال، یک ابر ممکن است ارتباط‌های رمزنگاری شده‌ای را بین ابر و کاربر خارجی برقرار کند- اما اگر ما تاثیر ارتباط‌های رمزنگاری شده را ارزیابی کنیمد، باید شناسایی کنیم که کنترل کاملا طراحی، پیاده‌سازی و ارزیابی شده است.

اندازه‌گیری حضور و یا تاثیر کنترل‌های امنیتی دقیقا همان چیزی است که ارزیابی‌های امنیتی گرایش به انجام آنها را دارند. ارزیابی‌های امنیتی مقدار گسترده‌ای بعنوان راهنما برای برنامه‌ریزی و توسعه امنیت دارند و هم‌چنین برای ارزیابی این که کنترل های مورد نیاز به درستی پیاده سازی شده‌اند. اما هم‌چنین ارزیابی‌ها ابزارهایی برای تهیه‌ی خدمات ابری نیز دارند: برای مثال، ارائه‌دهنده‌ی ابر ممکن است تصمیم به بکارگیری نتایج سطح بالای ارزیابی امنیتی سوم شخص داشته باشد. علاوه بر این، اگر ما مجبور به مقایسه‌ی امنیت دو یا چند ابر باشیم، باید مجموعه‌ای از شاخص‌ها برای ارزیابی را داشته باشیم.

براساس حساسیت داده‌ها یا ریسک مورد انتظار سیستم، ما باید فاز الزامات اساسی را متحمل شویم که در آن کنترل‌های امنیتی مناسبی شناسایی می‌شوند. اگر بعد از آن ارزیابی کامل فرآیند تصمیم‌گیری را که منجر به شناسایی کنترل‌ها می‌شود را انجام دهیم و آن را با ارزیابی امنیتی تاثیر آن کنترل‌هایی که پیاده‌سازی شده‌اند همراه کنیم، ما درک بسیار خوبی از این که آیا سرویس ابر کلی امنیت خوبی در برابر ریسک‌هایی که متحمل می‌شود دارد یا خیر را بدست خواهیم آورد.

شکل (۶-۱) نشان‌دهنده‌ی رابطه‌ی بین الزامات، ارزیابی امنیتی ابر، پیاده‌سازی ابر، اصلاح آسیب و ادامه‌ی کنترل‌های مدیریت پیکربندی را نشان می‌دهد.



شکل ۶-۱: از الزامات و ارزیابی تا بهبود مستمر امنیتی

کارهای موجود روی چارچوب‌ها یا راهنمایی امنیت ابر

در سال‌های اخیر که رایانش ابری بعنوان یک مدل جدید برای IT مورد استفاده قرار گرفته است، تلاش‌هایی برای راهنمایی درباره‌ی امنیت ابر صورت گرفته است. این‌ها تلاش‌ها شامل موارد زیر است:

- تعهد امنیتی ابر(CSA). CSA در موارد بسیاری فعال بوده است که از جمله‌ی آنها می‌توان به موارد زیر اشاره کرد:
  - ماتریس کنترل‌های ابر (CCM): این برای فراهم کردن اصول امنیت اولیه برای راهنمایی وندورهای ابر برای یاری به مشتریان ابر در دستیابی به خطر امنیتی کلی ارائه‌ی دهنده‌ی ابر طراحی شده است. ماتریس کنترل‌های ابر چارچوب کنترل‌هایی را فراهم می‌کند که فهم دقیقی از مفاهیم امنیتی را فراهم می‌کند و هم‌چنین اصولی را فراهم می‌کند که همراستا با راهنمایی تعهد امنیت ابر در ۱۳ حوزه هستند.
  - پرسشنامه ابتکاری ارزیابی مشارکتی: در این مورد تمرکز روی فراهم کردن راه‌های پذیرفته شده در صنعت برای داکیومنت‌سازی کنترل‌های امنیتی موجود در IaaS، PaaS و SaaS است و هم‌چنین هدف فراهم کردن شفافیت کنترل امنیتی است.
  - راهنمایی امنیتی برای حوزه‌ها حیاتی تمرکز در پردازش ابری. V۲,۱ در دسامبر ۲۰۰۹ منتشر شد، که راهنمایی امنیتی بری یک تعداد از حوزه‌ها در رایانش ابری را نشان داد؛ این حوزه‌ها شامل ساختار، حکومتداری، امنیت سنتی و مجازی سازی می‌باشند.
  - حوزه ۱۲: راهنمایی برای مدیریت شناسایی و دسترسی. V۲,۱ در آپریل ۲۰۱۰ این راهنمایی را منتشر کرد که در آن درباره‌ی توابع مدیریت شناسایی اصلی بحث شده است زیرا که آنها مرتبط با رایانش ابری هستند. این کار سنگ بنای اصلی Trusted Cloud CSA است.
- ممیزی ابر[۵۹]: به دنبال دادن ابزارهایی برای اندازه‌گیری و مقایسه امنیت سرویس‌های ابری به متقاضیان و عملگرها است. این با تعریف یک واسط و فضای نامی که به ارائه‌دهنده‌های

[۵۹] *Cloud Audit*

رایانش ابری امکان خودکاری سازی ممیزی، تایید، ارزیابی و اطمینان (A۶) زیرساختهای آنها (IaaS)، پلتفرم (PaaS) و محیط برنامه (SaaS) انجام می‌شود [۳].

- سازمان امنیت اطلاعات و شبکه‌ی اروپا (ENISA): منجر به تلاش‌هایی در راهنمایی امنیتی در اروپا شده است، ENISA چندین انتشار در رابطه با راهنمایی را منتشر کرده است که برای انتخاب ایمن رایاش ابری است. که این‌ها شامل موارد زیر هستند:
  - رایانش ابری: چارچوب اطمینان اطلاعات. که این در نوامبر ۲۰۰۹ چاپ شد. نشان‌دهنده‌ی مجموعه‌ای از شاخص‌های اطمینان است که ریسک انتخاب محاسبه‌ی ابر را بررسی می‌کند.
  - رایانش ابری: مزایا، ریسک‌ها و توصیه‌ها برای امنیت اطلاعات. که این در نوامبر ۲۰۰۹ چاپ شده است.
  - ارزیابی و مجوز امنیت پیشنهاد شده توسط شورای CIO برای ابر رایانه دولت ایالات متحده. اهمیت اصلی آن این است که آن کنترل‌های امنیتی NIST ۸۰۰-۵۳R۳ را برای رایانش ابری در سیستم‌های ریسک کم و متوسط را دارند انتخاب می‌کند [۴].
- گروه رایانش مورد اعتماد (TCG). در سپتامبر ۲۰۱۰، TCG گروه کاری زیرساخت چند مستاجری مورد اعتماد را تشکیل داد که گرایش به توسعه‌ی یک چارچوب ایمن برای رایانش ابری داشت. گروه کاری زیرساخت چند مستاجری مورد اعتماد از استانداردهای موجود برای تعریف امنیت پایان به پایان برای رایانش ابری در یک چارچوب که می‌تواند بعنوان اساس انطباق و حسابرسی در خدمت باشد، استفاده می‌کند.

تمام این تلاش‌ها نسبتا جدید هستند و باید پذیرش گسترده‌ای را بدست آورند.

بیش از این، این‌ها فعالیت‌های اساسی هستند که گرایش به این دارند که بعنوان نقطه‌ی شروع کارهای رسمی یا محصول تلاش‌های گروهیی برای یک چارچوب مشترک برای امنیت ابری به خدمت گرفته شوند. به عبارت دیگر عدم قطعیت زیادی در این حوزه است. که این نشان‌دهنده‌ی سختی برای متقاضیان ابری است که نیاز به ارزیابی امنیت ابرهای گروهیی و خصوصی خود دارند و هم‌چنین برای کاربرانی که به ابزاری برای ارزیابی امنیت سرویس ابر نیاز دارند نیز دشوار است.

امروزه، کاربراین ابزارهای استاندارد و مشترکی برای ارزیابی امنیت ابر ندارند. امنیت ابر به سرعت رو به رشد است و تمام کارهایی که ما در این کتاب به آنها اشاره کرده‌ایم بین ۲۰۰۹ و ۲۰۱۰ صورت گرفته است.

ابزارها

ابزارهای زیادی برای تست امنیت مورد استفاده قرار گرفته‌اند. این‌ها شامل دسته‌های زیر هستند.

- پویش پورت برای باز کردن و پاسخ‌گویی به سرویس‌ها
- پویش پروتکل مدیریت شبکه‌ی ساده
- شمارش دستگاه یا فهرست بندی
- اسکن آسیب پذیر هاست
- آنالیز دستگاه شبکه
- تست پذیرش پسورد و عبور کردن

ابزارهای پایه‌ای وجود دارند که مورد آزمون قرار گرفته‌اند. این‌ها شامل نگاشت‌کننده‌ی شبکه[۶۰] برای پویش پورت و Nessus برای پویش آسیب‌پذیری هاست هستند. بعلاوه ابزارهای قدرتمندی هستند که ابزارهای دیگری نیز هستند که امکان تست دفاعی گسترده برای شناسایی کیفیت، انعطاف پذیری و آسیب پذیری های امنیتی مرتبط را می‌دهند. این ابزارها آزمون‌های مناسبی را برای رنج گسترده ای از شبکه‌های ابری را که نیاز دارند را پیشنهاد می‌دهند.

چک لیست برای ارزیابی امنیت ابر

گرایش توسعه‌ی یک چک لیست ارزیابی امنیت ابری داشتن یک سری ابزارهای مناسب برای شناسایی امنیت ابری و مشخص کردن اطمینان از یک CSP درباره‌ی امنیتش است. اما همانطور هم که در بخش مقدمه گفته شد، چنین چک‌لیست‌هایی می‌توانند توسط کاربران یا مشتریان برای مقایسه ی گزینه‌های امنیت ابر پیشنهاد شده توسط ارائه‌دهنده‌های متفاوت مورد استفاده قرار بگیرند. در ادامه‌ی این بخش چک لیست‌هایی را ارائه می‌دهیم که از چندین منبع استخراج شده‌اند، شامل ماتریس کنترل‌های ابری CSA [۵] فریم‌کاری تضمین اطلاعات رایانش ابری [۶]، NIST's ۸۰۰-۵۳R۳ [۷] است.

[۶۰] Network Mapper

هشدار

تست امنیت، به ویژه تست نفوذ و آزمایش آسیب پذیری ، می‌توانند به راحتی حس نادرستی از امنیت را ایجاد کنند. این مسئله به دو دلیل است:

۱. چنین تست‌هایی براساس دانش از آسیب‌پذیری هستند و نمی‌توانند برای صفر روی که دوره‌ای بوجود می‌آید محاسبه شوند. آسیب‌پذیری‌های جدید روزانه کشف می‌شوند. آسیب پذیری ها حتی سیستم های بسیار بالغ را نیز در معرض خطر قرار می‌دهند. دوباره چندین لایه از دفاع نیز استراتژی خوبی در برابر بهره‌برداری صفر روزه است.

۲. قبض سلامت در تست آسیب‌پذیری و نفوذ که بسیار دقیق است- شامل پروسه‌ها و کنترل‌های عملیاتی است که برنامه‌های اطلاعاتی امنیت بسته به آنها است- نمی‌توانند بعنوان یک مقیاس کلی برای امنیت اندازه‌گیری شوند.

بعبارت دیگر، آزمون تست- به ویژه تست نفوذ- تنها سیستم هدف را در نقطه‌ای از زمان و یک بسط محدودی تست می‌کند. سیستم‌ها و پیکربندی‌ها در طول زمان گرایش به تغییر دارند، و آسیب‌پذیری‌های جدید می‌توانند بعد از سال‌ها که سیستم آزمون و اثبات شد ظاهر شوند. مهندسان امنیت عموما توافق دارند که چنین تست‌هایی ارزشی را دارند. اما باید به یاد داشت که حریف‌های شما ممکن است زمان و علاقه‌ی بیشتری نسب به شما در تست سیستم شما داشته باشند، بنابراین تست باید جدی گرفته شود ولی نباید تنها به آن تکیه کرد.

یک کاربرد برای چک لیست این است که صاحب ابر می‌تواند از آن برای راهنمایی ارزیابی امنیت ابر استفاده کند. اگر ارائه‌دهنده‌ی ابر از چنین چک‌لیستی بعنوان یک چارچوب برای گزارش امنیت ابرها استفاده کند، مستاجرین آینده و کاربران می توانند امنیت نسبی ابرهای مختلف را مقایسه کنند. این چک لیست می‌تواند توسط مشتری ابر عمومی برای پرسش یکسری از سوالات که مرتبط با نیازهای تجاری مشتری هستند استفاده کند. هر یک از بخش‌های زیر حول نیازها و کنترل‌های مرتبطی شکل

گرفته است. شکل (۶-۲) نشان‌دهنده‌ی خلاصه‌ی بخش‌های چک‌لیست ارزیابی و لیست‌کردن گروه‌های کنترل‌ها یا الزامات برای هر بخش است.

امنیت بنیادین

سیاست امنیت، الزامات سازمان یا قوانین برای امنیت را تعریف می‌کند. سیاست امنیت محدودیت‌ها و الزاماتی را که افراد و گروه‌ها باید تحت آن عمل کنند را تعریف می‌کند، و بعنوان اعلامیه‌ای از گرایش مدیریت برای امنیت بکار گرفته می‌شود. اقداماتی که با توجه به امنیت صورت می‌گیرند باید به سیاست امنیت قابل ردیابی باشند. چندین ملاس از سیاست وجود دارد، که شامل سیاست امنیت کلی و سیاست‌های اضافه‌ای است که حوزه‌های محدود شده‌ی بیشتری را بررسی می‌کند. سیاست امنیت تمرکزش روی بدست آوردن نتایج ایده‌آل ، نه فقط روی یک پیاده‌سازی خاص، است.

چنین سیاست‌هایی بیانیه‌های دیگری از الزامات هستند که برای حوزه‌های خاصی می‌باشند. این‌ها اغلب بعنوان یک استاندارد تعریف می‌شوند و حوزه‌های خاصی مانند کنترل‌های تکنیکی یا الزامات سختی خاص را پوشش می‌دهند. راهنمایی‌ها یک داکیومنت کلاس سوم هستند که رسمیت کمتری دارند و بیشتر تمایل به بهترین شیوه های رویه ای دارند. این‌ها توصیف‌ها یا توصیه‌هایی از تمرین‌هایی است که اهداف سیاست امنیت را پشتیبانی می‌کنند که این توسط توصیف یک چارچوب برای پیاده‌سازی پروسه‌ها صورت می گیرد. بعبارت دیگر: یک سیاست بیان‌کننده‌ی چرایی، بیانیه‌ی استاندارد بیان‌کننده‌ی چه چیزی، و راهنمایی بیان‌کننده‌ی چگونگی است. چک‌لیست ۶-۱ عناصر امنیتی عملیاتی را پوشش می‌دهد که مربوط به سیاست، استانداردها، و راهنمایی‌ها است. چک‌لیست ۶,۲ شاخص ارزیابی را پوشش می‌دهد که تمرکزش روی شفافیت CSP است.

شکل ۶-۲. بررسی چک‌لیست ارزیابی

چک لیست ۶-۱: سیاست، استانداردها و راهنمایی‌ها [۱۰-۸]

- آیا سیاست امنیت به گونه‌ای شفاف مستند و تایید شده است و نشان‌دهنده ی تمام بخش‌های درگیر بعنوان نمایش گرایش مدیریت است؟
- آیا سیاست امنیت قانون، حریم خصوصی و دیگر بررسی‌های حکومتی را دارد؟
- آیا سیاست امنیت توسط استانداردهای امنیتی و یا راهنمایی‌ها نشان داده شده است؟
- آیا سیاست با سیاست خصوصی نشان داده شده است؟
- آیا سیاست‌های حریم خصوصی و امنیت با استانداردهای صنعت سازگار هستند؟
- آیا ارائه‌دهنده‌های سوم شخص همان سیاست‌ها و استانداردها را دارند؟

چک‌لیست ۶-۲: شفافیت [۱۳-۱۱]

- آیا CSP مشتریان را با یک کپی از سیاست‌های حاکم، استانداردها و راهنمایی‌ها پوشش می‌دهد؟
- آیا به مشتریان نسبت به تغییرات سیاست‌ها، استانداردها و راهنمایی‌های حاکم اطلاع داده‌اند؟
- آیا CSP مشتریان را با دید به حسابرسی های سازگاری با شخص ثالث آماده می‌کند؟
- آیا CSP برای مشتریان دید به ممیزهای داخلی و خارجی را فراهم می‌کند؟
- آیا CSP برای مشتریان دید مدیریت دارایی CSP و بازپرداخت تجهیزات را فراهم می‌کند؟

امنیت کارکنان برای یک ابر یک اساس است که براساس آن امنیت عملیاتی برقرار است. گرایش امنیت پرسنل جلوگیری از چندین کلاس ریسک امنیت و ایجاد یک محیط است که اهدافی را که در سیاست امنیت بیان شده است را تقویت می‌کنند. چک‌لیست ۶-۳ شاخص ارزیابی مرتبط با امنیت کارکنان را لیست کرده است.

چک لیست ۶-۳: امنیت کارکنان [۱۶-۱۴]

- آیا سیاست‌ها و پروسه‌ها برای موارد زیر است:
  - ✓ استخدام کارکنانی که دسترسی به مولفه های ابر و یا کنترل آنها را دارند؟
  - ✓ پیش بکارگیری برای کاربران با دسترسی ممتاز؟
- آیا سیاست امنیت کارمندان در همه‌جا سازگار است؟
- آیا آنها به سیستم‌های ابر آنلاین و داده‌ها بعنوان سیستم‌های آفلاینی که داده‌ها را ذخیره کرده‌اند و برای سیستم‌های آفلاینی که برای کاربرد آنلاین ارائه شده‌اند، بکار گرفته می‌شوند؟
- آیا برنامه آموزشی امنیتی وجود دارد، و اگر چنین است، این چقدر گسترده است؟
- اگر بخشی از یک ابر متعلق به قرارداد یا برون سپاری باشد، آیا طرف ارائه دهنده مطابق با همان سیاست و استانداردهای است که CSP اجرا می کند؟
- آیا پرسنل مجبورند گواهینامه های امنیتی داشته باشند ؟
- آیا دسترسی فیزیکی به امکانات CDP نیاز به بررسی‌های پیشین دارد؟

استفاده از قراردادی های متعارف یا ارائه دهندگان شخص ثالث می‌تواند خطر نامناسبی را برای مشتریان ایجاد کند در غیر این صورت چنین ارائه‌دهنده‌هایی به دنبال پیروی و اجرا با توجه به سیاست‌های CSP هستند. چک لیست ۶-۴ جزئیات شاخص برای ارائه‌دهنده‌های سوم شخص را بیان می‌کند.

چک لیست ۶-۴: ارائه‌دهنده‌های سوم شخص [۱۹-۱۷]

- آیا سرویس‌هایی و یا توابعی توسط سوم شخص فراهم می‌شوند؟
- اگر بخشی از یک ابر زیرقرارداد باشد و یا برون‌سپاری شده باشد، آیا بخش ارائه دهنده مطابق با همان سیاست و استانداردهایی است که CSP اعمال می‌کند؟
- در صورت استفاده، ارائه دهندگان شخص ثالث برای پیروی از سیاست ها و استانداردهای CSP حسابرسی می شوند؟
- آیا سیاست امنیت CSP و حکومت به تمام ارائه‌دهنده‌های سوم شخص بسط پیدا می‌کند؟

ملاحظات تجاری

ملاحظات تجاری متعددی هستند که همراه آنها ملاحظات امنیتی را داریم. ملاحظات امنیتی شامل مسائل قانونی، تداوم کسب و کار و تامین منابع هستند. شاخص ارزیابی برای این ملاحظات در چک لیست ۶-۵ و ۶-۶ و ۶-۷ فراهم شده است.

چک لیست ۶-۵: قانونی [۲۲-۲۰]

- داده‌ها در کجا ذخیره می‌شوند؟
- CSP در کجا گنجانده می‌شود؟
- آیا CSP از ارائه‌دهنده‌های سوم‌شخص که در محدوده ی یکسانی نیستند استفاده می‌کند؟
- آیا CSP با کارمند یا سرویسی قرارداد فرعی دارد؟
- آیا CSP از داده‌های مشتریان به روشی که بخشی از سرویس نمی‌باشد استفاده می‌کند؟
- آیا CSP پروسه های داکیومنت شده برای پاسخ به درخواست‌های قانونی برای داده‌های مشتری را دارد؟
- در هنگام رخ دادن سازگاری، CSP چگونه داده را برای مشتریان بدون فراهم کردن داده‌های ناسازگار فراهم می‌کند.
- آیا CSP در برابر زیان ها، از جمله حقوق و دستمزد برای تلفات مشتری به علت قطع شدن CSP یا قرار گرفتن اطلاعات، بیمه شده است؟

تداوم کسب و کار می تواند برای مشتریانی که از خدمات مبتنی بر ابر در یک شیوه ای مؤثر استفاده می کنند، حیاتی باشد. معیارهای مربوط به تداوم کسب و کار در لیست ۶-۶ فهرست شده است.

چک لیست ۶-۶: تداوم کسب‌وکار [۲۵-۲۳]

- آیا CSP فرآیند رسمی طرح احتمالی دارد که تداوم کسب‌وکار را راهنمایی و داکیومنت‌سازی کند؟
- هدف نقطه بازیابی خدمات (RPO) و هدف بازیابی (RTO) چیست؟
- آیا امنیت اطلاعات یکپارچه برای بازیابی و ترمیم است؟
- چگونه CSP یک اختلال در خدمات را به مشتریان می دهد؟
- آیا یک سایت ثانویه برای بازیابی فاجعه وجود دارد؟

تداوم کسب وکار یک مسئله‌ی پیچیده‌ای است که حوزه‌ی بسیار گسترده‌تری را دراد. علاقمندان در این زمینه می‌توانند برای کسب اطلاعات بیشتر در این حوزه به منابع زیر مراجعه کنند:

۱. ANSI/ASIS SPC.۱-۲۰۰۹. سازگاری سازمانی: سیستم های مدیریت امنیت، آمادگی و تداوم، با راهنمایی برای استفاده از استاندارد ملی آمریکا
۲. مؤسسه ملی علوم و فناوری (NIST) انتشارات ویژه ۸۰۰-۳۴، راهنمای برنامه ریزی احتمالی برای سیستم های فناوری اطلاعات
۳. دستورالعمل های عملی خوب که می تواند از www.thebcicertificate.org/bci_gpg.html دانلود شود.
۴. موسسه تداوم کسب و کار، در www.thebci.org .

شکست

در ۱۸ فوریه ۲۰۱۱ روزنامه‌ی آنلاین Zeit گزارش داد که خطایی در سیستم پرداخت ارائه‌دهنده‌ی ابر رخ داده است که دسترسی یک کمپانی آلمانی به ایمیل SaaS عمومیش و داکیومنت‌های آنلاینش را از کار انداخته است. اگر واقعیت‌ها در این مورد کاملا واضح نبودند اما باید همواره این را بعنوان یک هشدار در نظر داشت: هر حساب ارائه‌دهنده یا سیستم مدیریت مشتری می‌تواند خطا داشته باشد، و در یک مورد شدید این ممکن است منجر به یک شرایط رد سرویس تجاری شود. چنین خطای محاسباتی از نظر صدور صورت حساب و جمع‌آوری بدهی منحصر به فرد نمی‌باشد، اما در یک سیستم ارتباطات- مانند اینترنت- یا در یک شرایط سرویس‌های ابری، خطا می‌تواند به طور امکان‌پذیری اتفاق بیافتد، و در نتیجه به سرعت دچار شکست می‌شود. مدل سرویس‌های ابر دومین عامل پیچیده را بدست می‌آورند: بسیاری از سرویس‌های ابر به شدت به واسطهای سلف‌سرویس وابسته هستند. در مورد radio.d ، نشان داده شده است که CSP به طور ناگهانی دسترسی به رادیو را قطع می‌کند. Radio.de ظاهرا نمی‌تواند به اداره منطقه ای CSP در دوبلین برسد، و ایمیل‌ها و CSP برای روزها مشکل را حل نکردند. در این مورد خاص واقعیت‌ها واضح نمی‌باشد، بنابراین CSP در این‌جا شناسایی شده نمی‌باشد. با این حال اگر شما توابع تجاری حیاتی خود را برون‌سپاری کنید، اطمینان ایجاد کنید که هر شرایط مشابه دیگری می‌تواند با سرعت بیشتری با CSP مجددا حل شود. این مستلزم انجام کارهایی پیش از ایجاد رابطه ی کسب‌وکار با یک CSP است، هم‌چنین آن مستلزم نگهداری تماس با ارائه‌دهنده نیز می‌باشد، به طوری که شما همواره آگاه از هرگونه تغییر در تماس با روش‌ها یا جزئیات می‌باشید. در نهایت، باید در نظر گرفته شود که اگر

برنامه‌ی بازیابی فاجعه‌ی شما در سیستم‌های CSP ذخیره می‌شوند، شما اصلا برنامه‌ی بازیابی فاجعه ای ندارید. تامین منابع باید با اطمینان از این که سرویس ابر به طور کامل با افزایش تقاضای مشتری تامین می‌شود، صورت گیرد. برای انجام این کار، CSP باید اندازه‌های مشخصی را برای تحویل موفق به SLAها داشته باشد. برای مثال CSP ممکن است پروسه‌هایی برای اضافه کردن سرورها یا مخزن با افزایش تقاضا داشته باشد. چک لیست ۶-۷ شاخص‌های ارزیابی برای تامین منابع را لیست کرده است.

چک لیست ۶-۷: تامین منابع [۲۹-۲۷]

- برای مدیریت اشباع منابع چه کنترل‌ها و پروسه‌ها درگیر هستند(اشباع منابع شامل پردازش بیش از حد، حافظه و یا اشباع ذخیره سازی، و تراکم شبکه است).
- آیا CSP محدودیت اشتراک سرویس را برای حفاظت از SLA محدود می کند؟
- آیا CSP برای مشتریان اطلاعات و برنامه ریزی ظرفیت را فراهم می‌کند؟

حمایت عمیق

یکپارچه‌سازی و امنیت یک ابر عملیاتی بسته به یکپارچگی منابعی است که آن را تشکیل می‌دهند. نرم‌افزار یک عنصر اصلی برای آسیب‌پذیری و بهره‌برداری است. برای شروع چک لیست ۶-۸ شاخص‌های ارزیابی برای تضمین نرم‌افزار را لیست کرده است.

چک لیست ۶-۸: تضمین نرم‌افزار [۳۲-۳۰]

- برای حفظ یکپارچگی سیستم عامل ها، برنامه های کاربردی، به روز رسانی سیستم عامل، پرونده های پیکربندی و سایر نرم افزارها، چه کنترل هایی وجود دارد؟
- استانداردهای صنعتی، راهنمایی‌ها، یا بهترین عملیات‌ها چه هستند؟
- چه راهنمایی‌ها و پروسه‌هایی برای نگهداری یکپارچگی نرم‌افزار بکار گرفته می‌شود؟
- آیا تست نفوذپذیری و آسیب‌پذیری برای هر نسخه مورد استفاده قرار می‌گیرد؟
- آسیب‌پذیری‌های اصلاح شده چگونه شناسایی می‌شوند؟

یک تکنیک قدرتمند برای بهبود امنیت نرم‌افزار این است که به توسعه‌دهنده‌ها در طول فرآیند توسعه قدرت دهیم که این با دادن دسترسی به آنها در ابزارهای تست امنیت امکان‌پذیر است. بهترین تمرین این است که محیط توسعه را از نزدیک تست های نهایی، مرحله بندی و محیط های تولید باشد.
خاص‌ترین جنبه‌ی امنیت ابر پیاده‌سازی شبکه‌ای آن است. انتخاب‌های مربوط به معماری و ایزوله‌سازی که در این جا ایجاد شده‌اند مزایای دور از دسترسی دارند. انتخاب‌های شبکه‌ای با شبکه‌ی فیزیکی و قابلیت های تجهیزات آغاز می‌شود و به مجازی‌سازی شبکه ای و کنترل بسط می‌یابد. درجه‌ی ایزوله‌یازی بین کلاس‌های متفاوت ترافیک الزامات امنیتی دیگری را در سیستم‌ها و سطح‌های VM ایجاد می‌کند. چک لیست ۶-۹ شاخص امنیت شبکه را لیست کرده است.

چک لیست ۶-۹: امنیت شبکه [۳۵-۳۳]

- برای مدیریت حملات در داخل و خارج که شامل محروم‌سازی توزیع‌شده‌ی سرویس (DDoS) هستند، چه کنترل‌های بکار گرفته می‌شود؟
- برای مشتریان، ایزوله‌سازی بین ماشین‌های مجازی توسط هایپروایزر چگونه مدیریت می‌شود؟
- برای مشتریان، ایزوله‌سازی بین VMها توسط مسیریابی و سخت‌افزار شبکه چگونه مدیریت می‌شود؟
- چه استانداردها و تجربه‌ی برتری برای پیاده‌سازی زیرساخت شبکه‌ای مجازی مورد استفاده قرار می‌گیرد؟
- ایزوله‌سازی بین سیستم‌های قابل مسیریابی و در دسترس مشتریان و سیستم‌های مدیریت ابر به چه صورت مدیریت می‌شود؟
- آیا پردازش‌های مشتریان ابر وابسته به اجزای مستاجر غیرفعال مانند پروتکل دسترسی آسان دیتابیس (LDAP) است؟
- آیا CSP تست نفوذ دوره ای را در برابر ابر انجام می دهد؟
- اگر چنین است، آیا آزمون نفوذ انجام می شود؟
- آیا CSP آزمون آسیب‌پذیری زیرساخت ابر، مدیریت ابر، و اجزای قابل دسترس مشتری را انجام می‌دهد؟
- آیا اطلاعات آسیب‌پذیری در دسترس مشتریان است؟
- آیا CSP به مشتریان امکان انجام تست آسیب‌پذیری در برابر VMهای مشتری را می‌دهد؟

انواع و درجه‌های کنترل‌های امنیتی که برای محافظت هاست‌ها و VMها مور نیاز هستند بسیار گسترده می‌باشند. هرچقدر اتعطاف‌پذیری بیشتر باشد، کنترل‌های جبرانی بیشتری در سطح هاست و VMها مورد نیاز است. چک لیست ۶-۱۰ شاخص ارزیابی برای امنیت میزبان و VM را لیست کرده است.

چک لیست ۶-۱۰: امنیت هاست و VM [۳۸-۳۶]

- آیا VMهای مشتری رمزگذاری شده و یا در هنگام ذخیره‌سازی محافظت می‌شوند؟
- آیا تصاویر VM قبل از اینکه آنها مقرر شوند، تصحیح شده اند؟
- چگونه و با چه تناوبی تصاویر VM پیش از مقرر شدن تصحیح می‌شوند؟
- آیا مشتری می‌تواند تصاویر VM خودش را فراهم کند؟
- آیا CSP شامل اعتبار احراز هویت است، و اگر این‌چنین است آنها برای چه استفاده می‌شوند؟
- آیا تصاویر وصله شده و غیر قابل اصلاح VM شامل نمونه‌های فایروال عملیاتی به صورت پیش فرض می‌باشند؟
- آیا تصاویر VM غیر قابل اصلاح و پچ شده شامل سیستم های تشخیص نفوذ (IDS ها) یا سیستم های پیشگیری از نفوذ (IPS ها) هستند؟ اگر چنین است، آیا CSP به این ها در عملیات دسترسی دارد (و اگر چنین است، چگونه)؟
- چطوری جداسازی بین سرورهای VM مجزا برای مشتریان مختلف تضمین شده است؟
- چگونه ارتباط بین VM ها برای یک مشتری برقرار می شود؟
- امنیت چگونه برای داده های کاربر در سیستم های ذخیره سازی تضمین شده است؟
- چگونه از امنیت داده ها در بین سیستم های ذخیره سازی و VM های مشتری اطمینان حاصل می شود؟
- آیا CSP اطلاعات را برای مشتریان به منظور راهنمایی امنیت مصرف‌کننده فراهم می‌کند به طوری که آن برای محیط مجازی‌سازی مفید باشد؟

CSPها اغلب مسئول پلت‌فرم پشته‌ی نرم‌افزار می‌باشند، که از جمله‌ی آنها امنیت است. اگرچه یک CSP ممکن است نخواهد جزئیات درباره‌ی امنیت پشته‌ی PaaS را فراهم کند، یک CSP باید شفافیت‌هایی را درباره‌ی امنیت داشته باشد و هم‌چنین حوزه‌های کنترل امنیت. چک‌لیست ۶,۱۱ شاخص ارزیابی برای امنیت PaaS و SaaS را لیست می‌کند.

چک لیست ۶-۱۱: امنیت PaaS و SaaS

- CSP چگونه برنامه های چند مستاجره را جداسازی می کند؟
- CSP چگونه داده های کاربر یا مستاجر را جدا می کند؟
- آیا CSP امنیت را به عنوان یک ویژگی سرویس برای PaaS ارائه می دهد (مانند احراز هویت، تک نشانه، مجوز، و امنیت حمل و نقل)؟
- چگونه CSP اسیب‌پذیری‌های جدید امنیتی در برنامه‌ها و در داخل ساختار ابری را شناسایی می‌کند؟

مدیریت دسترسی و شناسایی عناصر حیاتی امنیت برای ابر می‌باشند. چک لیست ۶-۱۲ شاخص‌های ارزیابی برای مدیریت شناسایی و دسترسی را لیست کرده است که همراه با احراز هویت می‌باشند.

چک لیست ۶-۱۲: مدیریت دسترسی و شناسایی [۴۴-۴۲]

- آیا حسابهای کنترل شده CSP دارای اختیارات فراوانی هستند (و اگر چنین است، کدام عملیات)؟
- چگونه CSP حساب ها را با اختیارات مدیر یا بالاتر مدیریت می‌کند؟
- آیا CSP از کنترل دسترسی دو نفره استفاده می کند و اگر چنین است، برای کدام عملیات؟
- آیا CSP تقسیم اختیارات را اجرا می کند (به عنوان مثال، کنترل دسترسی مبتنی بر نقش، یا RBAC)، و اگر چنین است، چه نقش هایی برای محدود کردن اختیارات (امنیت، سیستم عامل، هویت و غیره) استفاده می شود؟
- آیا CSP پیاده‌سازی دسترسی break-glass را انجام می‌دهد و اگر چنین است تحت چه شرایطی آن را انجام می‌دهد و فرآیند برای پاکسازی پس از آن [1] چیست؟
- آیا CSP مزایای مستاجرین یا مدیران کاربران را اعطا می کند، و اگر چنین است، محدودیت هایی برای این اختیارات چیست؟
- آیا CSP هویت کاربر را در هنگام ثبت نام تایید می کند و اگر چنین است، آیا سطوح مختلف چک، بسته به منابعی که دسترسی به آنها اعطا می شود، وجود دارد؟
- دسترسی از راه دور چگونه مدیریت و پیاده‌سازی می‌شود؟

برای سیستم های مدیریت هویت و دسترسی به مشتری استفاده شده توسط CSP:

- آیا این از مدیریت هویت فدرال پشتیبانی می کند؟
- آیا سیستم CSP با سیستم های ارائه کننده هویت شخص ثالث سازگار است؟
- آیا مشتری می تواند یک ورودی را وارد کند؟
- آیا این سیستم از جداسازی نقش ها و اصل حداقل بودن اختیارات (LPP) پشتیبانی می کند؟

چگونه یک CSP هویتش را تحت سناریوهای زیر به مشتری می‌شناساند؟

- هنگامی که CSP در خارج از مرز با کاربر یا مشتری ارتباط برقرار می‌کند
- هنگامی که مشتری با CSP از طریق API ارتباط دارد
- هنگامی که مشتری از واسط مدیریت ابر استفاده می‌کند

احراز هویت

- چگونگی احراز هویت برای عملیات با کیفیت بالا CSP اجرا می شود؟
- آیا احراز هویت چند مستاجری مورد استفاده قرار می گیرد؟
- آیا دسترسی به عملیات با اطمینان بالا محدود به عملیات‌های شبکه‌های ابری است و تنها از آدرس‌های لیست سفید امکان‌پذیر است؟

رمزنگاری و مدیریت کلید باید به صورت دقیقی مدیریت شوند؛ در غیر این صورت امنیت رمزنگاری به سرعت تضعیف می‌شود. چک لیست ۶-۱۳ شاخص‌های امنیتی برای این حوزه‌ها را لیست کرده است.

چک لیست ۶-۱۳: رمزنگاری و مدیریت کلید [۴۷-۴۵]

مدیریت کلید برای کلیدهایی که CSP کنترل می‌کند:

- CSP چگونه از کلیدها محافظت می‌کند، و چگونه کنترل‌های امنیتی برای تاثیرگذاری روی آن بکار گرفته می‌شوند؟
- آیا ماژول های امنیتی سخت افزاری برای محافظت از چنین کلید هایی استفاده می شود؟
- چه کسی به این کلید دسترسی دارد؟
- آیا لغو کلید در سراسر ابر به صورت عملیات اتمی انجام می‌گیرد؟

رمزنگاری

- برای کدام عملیات (و کجا) رمزگذاری مورد استفاده قرار می گیرد؟
- آیا همه مکانیزم های رمزنگاری براساس محصولات آزمایش شده و ارزیابی شده توسط شخص ثالث هستند؟
- آیا سیاست امنیت به طور واضح تعریف می‌کند که چه چیزی باید رمزنگاری شود؟

امنیت عملیاتی

بسیاری از نگرانی‌ها حول ابر عمومی در رابطه با امنیت فیزیکی IT در کنترل شخص سوم است. با یک ابر عمومی، یک نقض فیزیکی چندین مشتری را تحت تاثیر قرار خواهد دارد. چک لیست ۶-۱۴ شاخص ارزیابی برای امنیت فیزیکی مرکز داده و شبکه‌سازی و قدرت مرکز داده را لیست کرده است.

چک لیست ۶-۱۴: قدرت، امنیت فیزیکی مرکز داده، و شبکه‌سازی[۵۰-۴۸]

امنیت فیزیکی دیتاسنتر:

- الزامات برای داشتن دسترسی فیزیکی به تسهیلات CSP چیست؟
- آیا ورود به تسهیلات محدود به محل کار و ورود است؟
- آیا تسهیلات به مناطق تقسیم شده است به طوری که هر کدام به مجوز دسترسی نیاز دارند؟
- احراز هویت قوی (مثلا کارت چند نفره و PIN یا کارت و بیومتریک) برای دسترسی فیزیکی مورد نیاز است؟
- آیا تمام دسترسی ها نظارت و مستند شده اند؟
- آیا تمام مکان‌های ورود تحت کنترل می‌باشند؟
- آیا کنترل ویدئویی برای تمام نواحی مشترک ر تسهیلات کامل است؟
- هر چند وقت یکبار ارزیابی خطر برای امنیت فیزیکی صورت می‌گیرد؟

شبکه‌سازی و قدرت دیتاسنتر

- آیا شبکه‌سازی و قدرت در داخل تسهیلات ایمن شده‌اند؟
- آیا سیستم‌های محیطی در استانداردهای صنعتی پیاده‌سازی شده‌اند؟
- آیا تهویه مطبوع به اندازه کافی خوب برای مقاومت در برابر دوره های طولانی از شرایط است؟
- آیا این مرکز در معرض خطر متوسط یا بالاتر از آسیب زیست محیطی یا آب و هوا است؟
- آیا تسهیلات برق اضافی (UPS) کافی برای قطع کوتاه مدت یا موقت وجود دارد؟
- آیا تسهیلات دارای اتصالات اینترنتی متعددی هستند و آیا از ارائه دهندگان مختلف سطح ۱ هستند؟ آیا این تسهیلات از منابع مختلف قدرت تامین می شود؟

CSP باید لیست کامل و اخیر تمام اطلاعات منابع را که برای پیاده‌سازی و اجرای ابر استفاده شده‌اند را نگهداری کند. یک تجربه‌ی جدید استفاده از دیتابیس مدیریت پیاده‌سازی برای نگهداری چنین اطلاعاتی است. چک لیست ۶-۱۵ شاخص برای مدیریت دارایی دیتاسنتر را لیست کرده است.

چک لیست ۶-۱۵: مدیریت دارایی مرکز داده [۵۳-۵۱]

- آیا CSP فهرست کامل و جاری تمام سخت‌افزار، شبکه، نرم‌افزار و عناصر مجازی را که ابر را تشکیل می‌دهند را نگهداری می‌کند؟
- آیا CSP چنین مدیریت و ردیابی انبارداری را خودکار می‌کند؟
- آیا CSP تمام دارایی‌هایی را که مشتری استفاده کرده است و یا مشتری داده‌ها را روی آن ذخیره کرده است را نگهداری می‌کند؟
- آیا CSP جداسازی یا جداسازی فیزیکی دارایی ها را در سطوح مختلف حساسیت حفظ می کند؟

امنیت موثر یک فرآیند در دست اقدام است که مستلزم پروسه‌های به خوبی تعریف شده و نقش‌هایی برای تمام کارمندان است. برای موثر بودن، چنین پروسه‌هایی باید انواع رویدادها را پیش بینی کنید. پروسه‌ها باید راهنمایی کافی برای اجازه به کارمندان برای بررسی یک محدوده‌ی گسترده از شکست‌ها در سیستم‌ها، پروسه‌ها و دیگر شرایط پیشنهاد دهند. چنین رخدادها و مسئولیت‌هایی باید ضبط شود. چک لیست ۶-۱۶ شاخص ارزیابی برای تجربه‌های عملیاتی را لیست کرده است.

چک لیست ۶-۱۶: تجربه‌های عملیاتی [۵۶-۵۴]

- آیا یک فرایند کنترل تغییر رسمی وجود دارد و آیا روش های به روشنی مستند شده اند؟
- آیا کنترل تغییر شامل یک ابزار برای هدایت تصمیمات در مورد تغییراتی است که نیاز به ارزیابی مجدد خطر دارند؟
- آیا روش های عملیاتی به وضوح مستند شده و دنبال می شوند؟
- محیط های جداگانه ای برای توسعه، تست، ساخت و تولید وجود دارد؟
- کدام سیستم و کنترل های امنیتی شبکه برای اطمینان از برنامه ها و اطلاعات کاربر نهایی یا مستاجر استفاده می شود؟
- کدام کنترل امنیتی برای کاهش کد مخرب استفاده می شود؟
- روش های پشتیبان چیست (چه کسی این کار را انجام می دهد، چه پشتیبان گیری می کند، چه زمانی انجام می شود، چه فرمتی را انجام می دهد)؟
- بک‌آپ‌ها در کجا ذخیره می شود و برای چه مدت ذخیره می‌شوند؟
- یا CSP تمامی نسخه های داده های مشتری را پس از خاتمه قرارداد مشتری به طور ایمن حذف خواهد کرد؟
- آیا CSP اصول امنیتی داکیومنت شده برای هر عنصر که زیرساخت ابر را تشکیل می‌دهد را دارد؟

در چک لیست ۶-۱۷ شاخص ارزیابی برای حوزه‌ی مدیریت حادثه لیست شده است.

چک لیست ۶-۱۷: مدیریت حادثه [۵۹-۵۷]

- آیا اطلاعات در ورودهای شبکه، ممیز و سیستم ضبط شدهاند؟
- این اطلاعات برای چه مدت نگهداری می‌شوند و چه کسی به آنها دسترسی دارد؟
- چه کنترل‌هایی برای پشتیبانی این ثبت‌ها از دسترسی احراز هویت نشده و حفظ زنجیره نگهداری مورد استفاه قرار می‌گیرد؟
- ثبت‌ها چگونه و هر چند وقت یکبار بررسی می‌شوند؟
- ثبت‌ها چگونه و هر چند وقت یکبار برای یکپارچگی و تکمیل بررسی می‌شوند؟
- آیا تمام سیستم‌ها و عناصر شبکه در یک زمان مشخص هماهنگ می‌شوند؟
- آیا CSP فرآیند رسمی برای پشتیبانی، شناسایی و پاسخگویی به حوادث را دارند؟
- آیا این فرآیندها به صورت دوره‌ای برای شناسایی این که آنها موثر و مناسب هستند تست می‌شوند؟
- فرآیند تشدید برای پاسخ حادثه چیست؟
- چه کسی به چنین ثبت‌ها دسترسی دارد؟
- ثبت‌های امنیتی برای چه مدتی نگهداری می‌شوند؟
- آیا CSP از تشخیص نفوذ، نظارت امنیتی، یا رویداد امنیتی و مدیریت حوادث (SIEM) برای تشخیص حوادث استفاده می کند؟
- آیا CSP می تواند یک تصویر قانونی از VM مشتری فراهم کند؟
- آیا CSP به مشتریان امکان پیاده‌سازی IDS مبتنی بر میزبان در VMها را می‌دهد؟
- آیا یک CSP رویدادهای مشتری و اطلاعات مربوط به حوادث را به نظارت امنیتی و فرآیند مدیریت حوادث خود می پذیرد؟

معیارها برای چک‌لیست‌ها

چک‌لیست‌ها به تنهایی توانایی قضاوت امنیت یک ابر را دارند، مشتریان ابر عمومی و صاحبان ابر خصوصی باید می‌خواهند که موارد زیر را بدانند:

- پیاده‌سازی چقدر ایمن است؟
- آیا CSP بهترین تجربه‌ها برای ایمنی بکار می‌گیرد؟
- CSP چطور الزامات و کنترل‌های امنیتی گسسته را بدست می‌آورد؟
- این سرویس چگونه با سرویس‌های مشابه دیگر مقایسه می‌شود؟

با توجه به چک لیست‌های ۶-۱ تا ۶-۱۷، تنوع خوبی در کار با کنترل‌هایی که می‌توانند پیاده سازی شوند دیده می‌شود. این باعث شده است که شناسایی متریک‌ها برای هر سوال دشوار شود. روش‌های موجود برای اندازه‌گیری امنیت توسط شرح دقیق کنترل‌های امنیتی به خوبی دسته‌بندی شده برای قلمروهای خاص و مشخص کردن این که کدام یک از این کنترل‌ها به سیستم عامل در سطح‌های مختلف تضمین با حساسیت اعمال می‌شوند، با این چالش روبرو می‌شوند.

ارزیابی واقعی امنیت یک پیاده‌سازی نیز زمان بر است و هم‌چنین هزینه‌بر نیز می‌باشد. صدور گواهینامه و اعتباربخشی سیستم منتج شده یک اسنپ شات از زمان است و در صورت تغییر سیستم باید تکرار شود. اصولا این ارزیابی‌ها تجربه‌های کاغذی هستند که تلاش زیادی روی آنها صورت گرفته است. چیزی که مورد نیاز است تکامل این فرآیند به خودی خود است، و رایانش ابری با اتوماسیون بیشتری روبرو است، که این به دلیل ماهیت قرارداد بین مصرف کنندگان فناوری اطلاعات و ابر است. در ابتدا باید اطلاعاتی که درباره‌ی امنیت، سیستم‌ها و فرآیند جمع‌آوری شده است در یک مخزن C&A سازماندهی شوند که این مخزن بیشتر شبیه به یک دیتابیس است تا یک داکیومنت رسمی سنتی. اهمیت سازماندهی و جمع‌آوری اطلاعات این است که آن بیانیه ها و ادعاهای مربوط به چگونگی بدست آمدن کنترل‌های امنیت گسسته را پشتیبانی می‌کند. داشتن چنین اطلاعاتی به صورت دیتابیس آن را برای چندین موجودیت مفید می‌کند. در یک پیاده سازی ابر، بخش‌های مختلفی از زیر ساخت و کنترل‌های یکسانی استفاده می‌کنند. ارزیابی امنیت باید قادر به استفاده‌ی مجدد از اطلاعاتی که درباره‌ی این کنترل‌ها هستند باشد و البته باید قادر به استفاده‌ی مجدد از اطلاعات تاثیر این کنترل‌ها. رایانش ابری باعث متحول شدن امنیت می‌شود، و کاملا واضح است که انتخاب یک ابر باعث توسعه‌ی اندازه‌گیری و ارزیابی امنیت و بدست آوردن امنیت مورد تقاضا می‌شود.

خلاصه

رشد در رایانش عمومی موجب افزایش نیاز به امنیت بیشتر شده است. سرویس‌های ابر عمومی نیاز به فراهم کردن سرویس‌های موثر از نظر هزینه و مجموعه‌ای از ویژگی‌ها که امکان انتخاب را آسان می‌کند ، دارند. اما برآورده کردن الزامات IT به گونه‌ای ایمن نیز بسیار اهمیت دارد. از این رو CSP راه‌هایی برای ارزیابی محصولات با استفاده از شاخص‌های مورد انتظار دارد. در این فصل چک‌لیست‌های امنیتی برای راهنمایی خوانندگان در توسعه‌ی لیست‌هایشان برای ارزیابی امنیت CSP یا ابر خصوصی داده شده است.

نکته

علاقمندان به ارزیابی امنیتی می‌تواند به گروه‌های زیر بپیوندند:

- اتحاد امنیت ابر[۶۱]
  - ✓ www.cloudsecurityalliance.org
  - ✓ www.linkedin.com/groups?mostPopular=&gid=۱۸۶۴۲۱۰
  - ✓ http://groups.google.com/group/cloudsecurityalliance
- ممیزی ابر
  - ✓ www.cloudaudit.org
  - ✓ http://groups.google.com/group/cloudaudit
- گروه رایانش مورد اعتماد[۶۲]
  - ✓ www.trustedcomputinggroup.org/solutions/cloud_security
  - ✓ www.linkedin.com/groups?mostPopular=&gid=۳۲۵۴۱۱۴
- CloudSecurity.org (http://cloudsecurity.org/forum/index.php) خیلی فعال نمی‌باشد ولی پتانسیل یک انجمن مستقل برای همکاری در تست امنیت ابر را دارد.

Endnotes

۱. CSA-GRC-Stack-v۱،۰-README.pdf; www.cloudsecurityalliance.org.
۲. Ibid.
۳. Ibid.

[۶۱] The Cloud Security Alliance
[۶۲] The Trusted Computing Group

٤. Proposed Security Assessment & Authorization for U.S. Government Cloud Computing, Draft version ۰,۹٦, CIO Council, US Federal Government; ۲۰۱۰.
٥. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.
٦. Catteddu D., Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹ [accessed ۲٤,۰۳,۱۱].
۷. NIST Special Publication ۸۰۰-٥۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۸. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.
۹. Catteddu D., Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۱۰. NIST Special Publication ۸۰۰-٥۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۱۱. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.
۱۲. Catteddu D., Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۱۳. NIST Special Publication ۸۰۰-٥۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۱٤. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.
۱٥. Catteddu D., Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۱٦. NIST Special Publication ۸۰۰-٥۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۱۷. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.
۱۸. Catteddu D, Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۱۹. NIST Special Publication ۸۰۰-٥۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۲۰. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.

۲۱. Catteddu D, Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۲۲. NIST Special Publication ۸۰۰-۵۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۲۳. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.
۲٤. Catteddu D, Hogben G. Cloud Computing Information Assurance Framework, European
Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۲۵. NIST Special Publication ۸۰۰-۵۳ Revision ۳, Recommended Security Controls for Federal
Information Systems and Organizations; ۲۰۰۹.
۲٦. Asendorpf D. “Ab in die Wolken”, Zeit Online, ۲۰۱۱; www.zeit.de/۲۰۱۱/۰۸/ Cloud Computing; ۲۰۱۱ [accessed ۲٤,۰۳,۱۱].
۲۷. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.
۲۸. Catteddu D., Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۲۹. NIST Special Publication ۸۰۰-۵۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۳۰. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.
۳۱. Catteddu D., Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۳۲. NIST Special Publication ۸۰۰-۵۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۳۳. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.
۳٤. Catteddu D., Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۳۵. NIST Special Publication ۸۰۰-۵۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۳٦. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.

۳۷. Catteddu D., Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۳۸. NIST Special Publication ۸۰۰-۵۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۳۹. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.
۴۰. Catteddu D., Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۴۱. NIST Special Publication ۸۰۰-۵۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۴۲. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.
۴۳. Catteddu D., Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۴۴. NIST Special Publication ۸۰۰-۵۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۴۵. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.
۴۶. Catteddu D, Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۴۷. NIST Special Publication ۸۰۰-۵۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۴۸. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.
۴۹. Catteddu D., Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۵۰. NIST Special Publication ۸۰۰-۵۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۵۱. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.
۵۲. Catteddu D., Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۵۳. NIST Special Publication ۸۰۰-۵۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۵۴. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.

۵۵. Catteddu D., Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۵۶. NIST Special Publication ۸۰۰-۵۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.
۵۷. Controls Matrix (CM), Cloud Security Alliance V۱,۰; ۲۰۱۰.
۵۸. Catteddu D., Hogben G. Cloud Computing Information Assurance Framework, European Network and Information Security Agency (ENISA). www.enisa.europa.eu; ۲۰۰۹.
۵۹. NIST Special Publication ۸۰۰-۵۳ Revision ۳, Recommended Security Controls for Federal Information Systems and Organizations; ۲۰۰۹.

فصل ۷ عملیاتی کردن ابر

نکات فصل:

- از معماری تا عملیات‌های ایمن و کارآمد
- فعالیت‌های عملیات ایمن

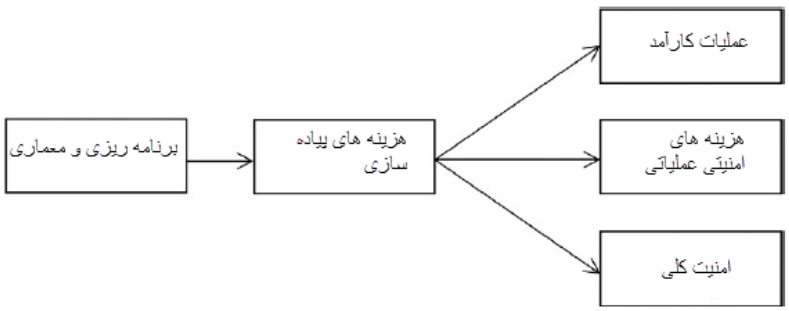
مقدمه

در طول این کتاب و به روش های مختلف ما بیان کرده‌ایم که رایانش ابری تکامل در مدل‌های IT است، انتخاب آن پیامدهایی دارد که بعدها بدست می‌آیند. از یک طرف ما مزیت‌هایی مثل راحتی و سریع بودن توسعه با هزینه‌های کمتر را داریم. از این رو متقاضیان ابر در پروژه‌های جدید IT با ریسک‌های کمتری روبرو هستند. استفاده از یک ابر عمومی، هر کس با یک ایده که نیازمند زیرساخت IT است می‌تواند بدون بدست آوردن زیرساخت و استخدام کارمند به آن دست یابد. اگر شما اتصال اینترنت، لپ‌تاپ، و کارت اعتباری دارید، شما می‌تواند زیرساخت‌های IT بی سابقه‌ای دست یابید و هم‌چنین زمان انتظار بسیار پایین‌تر از زیرساخت‌های سنتی است. از طرفی دیگر، ناکامی در پذیرش ابر عمومی عمدتا با کاهش انعطاف‌پذیری ذاتی سرویسهای ابر عمومی همراه است، که همراه با نگرانی های مربوط به لغو کنترل فیزیکی منابع اطلاعاتی است. هم‌چنین فاکتوری وجود دارد که تمام سرویس‌های ابر عمومی به راحتی امکان انتقال داده به ارائه‌دهنده‌ی دیگر را نمی‌دهند. اغلب فناوری سنتی از یک رابطه هم افزایی با سایر توابع کسب و کار لذت نمی برد. در رایانش ابری کاتالوگ سرویس باید به طور دقیق تعریف شود زیرا که SLAها را به هم مرتبط می‌کند. در ابرهای خصوصی، مصرف‌کنندگان سرویس‌های IT انتظار دارند که ماشین مجازی خود را داشته باشند که به اندازه‌ی ابرهای عمومی سریع است. با این تحولات و طبیعت سلف‌سرویس سرویس‌های ابری، IT نیاز دارد که شریک کسب‌وکار شود. اما هم‌چنین ما باید انتظار داشته باشیم که رشد رایانش ابری و تغییراتی که به همراه دارد باعث کاهش پرسنل زیرساخت IT می‌شود. این کاملا طبیعی است و درجه‌ی اتوماسیون‌سازی سرویس‌های IT را که توسط ابر بدست آمده را نشان می‌دهد. در فصل‌های قبلی ما

رایانش ابری را تعریف کردیم و سرویس‌های ابری و مدل ها را مورد بررسی قرار دادیم. هم‌چنین ما نگرانی‌ها امنیتی را نیز مورد بحث قرار دادیم و بسیاری از آنها را با بررسی دقیق ساختار و امنیت ابر بررسی کردیم. در این فصل ما روی عملیات یک ابر از جنبه‌ی امنیتی تمرکز داریم.

هدف پردازش یک ابر تحویل سرویس‌های ابر به صورت ایمن، کارآمد، با صرف از نظر هزینه و قابل اعتماد است. رسیدن به این هدف ممکن است بسیار دشوار باشد، و این بسته به فعالیت‌های پشتیبانی بسیاری است. معماری منجر به پیاده‌سازی و هزینه‌هایی می‌شود، که شامل هزینه‌های امنیت عملیاتی است. پردازش ایمن و موثر توسط یک برنامه‌ی دقیق امکان‌پذیر است. اقدامات امنیتی واکنشی ناشی از برنامه ریزی ناکارآمدی و پرهزینه است. شکل ۱-۷ نشان‌دهنده‌ی این رابطه‌ی کلی است.

متاسفانه برنامه ریزی اولیه و معماری اغلب به علت هیبریدی از عوامل، کوتاه می شوند. تجربه‌ها نشان می‌دهند که سرمایه‌گذاری در برنامه‌ریزی و ساختار می‌تواند هزینه های عملیاتی و حفاظت از برنامه ها از مسائل غیرقابل پیش بینی که بوجود می آیند را کاهش دهد. به نظر می‌رسد که دو انتخاب وجود دارد: صرف زمان زیاد روی برنامه‌ریزی ، و یا صرف زمان ناکافی برای برنامه‌ریزی و بعد از تجربه‌ی بحران‌ها و تاخیرها.



شکل۷-۱: عملیات و امنیت کلی تحت تاثیر تصمیم‌ها

نکات:

اهداف بازگشت سرمایه‌گذاری (ROI) زیر را برای امنیت در نظر بگیرید:

- امنیت باید زمان کار ضروری را کاهش دهد

- تکنولوژی‌های امنیت و فرآیندها باید هزینه‌ی کلی را کاهش دهند
- امنیت باید قادر به عملکرد و مدیریت سیستم‌ها را فعال کند

اگر مجبور به تعریف یکسری قوانین برای پشتیبانی ROI باشیم، باید با موارد زیر آغاز کنیم:

- امنیت باید درآمد را از طریق افزایش مطلوبیت مشتریان افزایش دهد
- امنیت باید زمان کارکنان را در اصلاحات اورژانسی کاهش دهد
- امنیت باید میزان منابع در دسترس را کاهش دهد
- امنیت باید احتمال مداخله نظارتی، از جمله جریمه و اختلال کسب و کار را کاهش دهد.

از معماری تا عملیات‌های امن و موثر

امنیت یک فاکتور کلیدی است که با همه‌ی جنبه‌های عملیاتی ابر مرتبط است. پیش از آنکه یک مهندس امنیت یک حساب کاربری زیرساختی سابق کارکنان را غیرفعال کند، نتیج اسکن آسیب‌پذیری را بررسی می‌کند ، تاثیرات این اقدامات عملیاتی در نظر گرفته شده است. اساس فرآیندهای عملیاتی بعدی هنگامی که معماری ابر تعریف می شود، معین می شود. کاملا درست است که یک ابر در سطح دپارتمان یا پروتوتایپ می‌تواند طراحی شود و در نتیجه عملیات‌هایی را با تلاش مداوم داشته باشد.

مسئله‌های امنیتی می‌توانند به راحتی تشدید شوند و تقاضای توجه و منابع زیادی را در هنگامی که پیاده‌سازی بزرگتر می‌شود، دارند. ا بر یک ترکیب بسیار پیچیده و پویا است که بر روی فن آوری های مختلف و اجزای سازنده امکان پذیر است. شیوه طراحی، پیاده سازی و حتی پیکربندی آن، مسیری طولانی به سوی فعال کردن عملیات کارآمد و ایمن دارد.

حوزه‌ی برنامه‌ریزی

بسیار خوب است که فاز برنامه ریزی و معماری یک ابر با شرح فعالیت‌های عملیاتی که بعد از این که ابر آنلاین می‌شود اتفاق می‌افتند، آغاز شود. برنامه ریزی برای عملیات‌های ایمن در پیوستگی با برنامه‌ریزی دیگر جنبه‌های عملیاتی به خوبی انجام گرفته است. عملیات‌های ایمن نه تنها شامل حوزه هایی مانند مدیریت پیکربندی، میز خدمت[۶۳]، مدیریت مشکل، مدیریت ظرفیت و ارائه خدمات است،

[۶۳] service desk

بلکه عملیات امنیتی اغلب با این جنبه های دیگر عملیات مرتبط است. کتابخانه‌ی زیرساخت IT (ITIL) برای نشان دادن ارزش پیشنهادی بر حسب توصیف دقیق تجربه‌های IT اصلی که یک سازمان IT در عملیات با آنها مواجه می‌شود، شناسایی شده است. ITIL در رابطه با ضبط و سازماندهی بهترین تجربه‌ها حول تمام حوزه‌ی مدیریت سرویس‌های IT ، توسعه‌ی IT، و عملیات‌های IT است. از این رو ITIL یک نقطه‌ی آغاز خوبی را برای سازمان‌ها ایجاد می‌کند که آن فازهای برنامه‌ریزی و طراحی ایجاد ابر است. البته تمرکز ITIL روی عملیاتی کردن و مدیریت IT است، اما در هنگامی که ما برای ایجاد زیرساخت برنامه‌ریزی می‌کنیم و فرآیندهایی را تعریف می‌کنیم که به زودی سنگ بنای عملیات روزانه را تشکیل می دهد، آن ارزشمند است. مدیریت امنیت ITIL از کد برای مدیریت امنیت اطلاعات استنتاج می‌شود. هددف مدیریت امنیت اطمینان از امنیت اطلاعات است؛ به عبارتی دیگر، اطمینان از محرمانه بودن، صداقت و دسترسی به منابع اطلاعاتی. ITIL بعنوان یکسری از کتاب‌ها که هر کتاب تجربه‌های خاصی را پوشش می‌دهند انتشار یافته است. مجموعه‌ی کلی به هشت مجموعه‌ی منطقی سازماندهی شده است که با توجه به خط مشی‌های پردازشی مرتبط گروه‌بندی شده‌اند. در حال حاضر ورژن ۲ ITIL به صورت زیر است:

- پشتیبانی سرویس
- تحویل سرویس
- مدیریت زیرساخت تکنولوژی ارتباطات و اطلاعات
- مدیریت امنیت
- جنبه‌های تجاری
- مدیریت برنامه
- مدیریت دارایی‌های برنامه[۶۴]
- پیاده‌سازی ITIL در حوزه‌ای کوچک

اگرچه امنیت بخش‌های خودش را دارد، برنامه ریزی و ایجاد معماری برای امنیت نیز نیازمند درک دیگر حوزه‌ها است. امنیت خوب و دقیق شامل اقدامات امنیتی بالغ است که با سایر حوزه های عملی ادغام شده‌اند.

[۶۴] Software Asset Management

امنیت، هزینه‌های مداوم و دسترسی فیزیکی

برای کاهش هزینه‌های عملیاتی به پایین‌ترین سطح ممکن، دسترسی فیزیکی به زیرساخت IT باید بر اساس یک نیاز مستند شده محدود شود. اما حتی دسترسی اسکورت شده نیز ریسک‌هایی را دارد. چیزی که در این‌جا جالب است این است که هنگامی که زیرساخت ابر طراحی شده و ایجاد شده است (برای کارآیی عملیاتی)، سپس تمام دسترسی فیزیکی نباید روزانه مورد نیاز باشد.
مراکز داده با نظارت تصویری وسیع و سنسورهای محیطی تجهزی شده‌اند که آب، دود، رطوبت و درجه هوا را بدست می آورد. این‌ها بعدها می‌توانند با سنسورهای اضافی دیگر و دوربین‌های با وضوح بالا برای دیدن نورهای بصری از راه دور بکار گرفته شوند. کاهش نیاز برای کاهش نیاز به پرسنل عملیاتی برای داشتن حضور فیزیکی مداوم، هزینه های عملیاتی را کاهش می دهد، دوربین های با وضوح بالا سرمایه ای هستند که حداقل بازدید از مرکز داده را پشتیبانی می‌کنند و ضبط از این دوربین ها می تواند در صورت نیاز به عنوان یک ضبط قانونی باشد. هرچقدر این ویدئوها طولانی‌تر نگهداری شوند بهتر است زیرا حداقل یک تیم عملیاتی ممکن است به آنها نیاز داشته باشد.

دسترسی مجازی و منطقی

با توجه به اینکه ابرها بر روی شبکه مدیریت می شوند، کنترل دسترسی های فیزیکی و محدود کردن کنترل دسترسی به قلمرو فیزیکی مهم است. استفاده از سیستم های احراز هویت برای تعریف و مدیریت دسترسی توسط افراد برای دستگاه‌ها و توابع خاص یک راه موثر برای متمرکز سازی کنترل دسترسی داده‌ها است. اما کنترل‌های منطقی به تنهایی محدود به دسترسی محدود به سرورها و دیگر زیرساخت ابری نیستند. استفاده از ایزوله‌سازی شبکه بین حوزه‌های مختلف در زیرساخت ابر راه طولانی برای رسیدن به محدود کردن دسترسی هکر دارد؛ اما ایزوله‌سازی هم‌چنین دامنه‌ی پرسنل عملیاتی را محدود می‌کند. در واقع می‌توان گفت امنیت از پایین‌ترین لایه‌ی حفاظت کنترل را انجام می‌دهد ، و ایزوله‌سازی شبکه یک مکانیزم حفاظت ثانویه را فراهم می‌کند.

امنیت پرسنل

نه تنها باید دسترسی منطقی و فیزیکی پرسنل عملیاتی محدود شود، بلکه همه‌ی این افراد باید سیاست‌های پرنسل را رعایت کنند. دسترسی باید به گونه‌ای منظم نگهداری شوند بررسی های دوره ای از نیازهای دائمی دسترسی باید انجام شود و تمام پرسنل عملیاتی با دسترسی فیزیکی یا منطقی حداقل سالانه تحت آموزش قرار بگیرد. به همین ترتیب، تمام سیاست‌های کارکنان و پروسه‌ها باید

ارزیابی را ادامه دهند، به ویژه براساس حقوق دسترسی کاربران و امتیازات. هنگامی که پرسنل تیم عملیات را ترک می کنند، دسترسی آنها باید بلافاصله لغو شود؛ انجام این کار به طور موثر مستلزم استفاده از مدیریت شناسایی متمرکز است . لازم به ذکر است که در حالی که امنیت پرسنل ضروری است، تهدیدهای داخلی را متوقف نخواهد کرد. چه چیزی می تواند در مورد آن انجام شود؟ به عنوان مثال، مدیران امنیتی باید کار خود را به طور مستقل بر روی نتایج مورد انتظار آزمایش کنند.

آموزش

آموزش‌های خاص برای کارکنان IT برای تمام کارمندان مهم است، به ویژه پرسنل پشتیبانی عملیات ابری- که نه تنها شامل کارکنان زیرساخت است بلکه مدیران و کارکنان مختلف مرتبط با جنبه های دیگر عملیات مرتبط هستند. کارکنان عملیاتی ابر آموزش‌های مناسب برای اطمینان از این که آنها به تمام سیاست‌های کمپانی پایبند هستند را باید داشته باشند که از جمله‌ی آنها سیاست‌های امنیتی است. با وجود تعداد زیادی از سرورهای مجازی، پتانسیل ایجاد اختلال در سرورهای چندگانه یا به طور غیرمستقیم انجام انصراف سرویس بالا است . این نه تنها زمانی که سرویس کاملا عملیاتی است اعمال می‌شود بلکه ایجاد اولیه و در حال انجام نیز اعمال می‌شود. پیچیدگی و حوزه‌ی یک ابر تقاضا دارد که پرسنل‌ها تجربه‌ی بیشتری نسبت به مدیران سیستم‌های شرکتی معمولی داشته باشند.

دسته‌های کارکنان امنیت ابر

انواع امنیت پرسنل زیر وجود دارند که مرتبط با عملیات ابر هستند:

- امنیت فیزیکی یا کارکنان دیتاسنتر
- تحلیلگران امنیت مسئول کنترل هستند و یا مرتبط با یک مرکز عملیات امنیتی فیزیکی یا مجازی می‌باشند.
- اسکن یا تست نفوذ کارکنان
- مهندسان یا معماران سیستم‌های امنیتی
- افسر امنیتی و دیگر نقش های مدیریت امنیتی
- آنالیز جستجوی امنیت، توسعه‌دهندگان خودکارسازی امنیت و توسعه‌دهندگان محتوای امنیت

ابزار

گروهی منبع باز به گونه‌های مختلفی رایانش ابری را بکار می‌گیرد. در ابتدا بسیاری از پروژه‌های منبع باز در ابرهای مختلفی میزبانی می‌شوند. Google، Amazon، و دیگر ابرها توسعه‌ی فعال گروهی‌ها را پشتیبانی می‌کنند. دوم این که بسیاری از پروژه‌های منبع باز روی فعالسازی رایانش ابری تمرکز دارند. این تلاش‌ها برای توسعه‌ی نرم‌افزار شامل موارد زیر است:

- مدیریت پیکربندی: این ابزارها شامل Chef و Cfengine می‌باشند.
- نظارت: Zenoss، collectd ، CloudStatus در زمینه‌ی نظارت می‌باشند
- مدیریت: این دسته شامل OpenQRM، Bitnami و ControlTier است
- نرم‌افزار فعالسازی ابر: تلاش‌هایی که در این زمینه صورت گرفته است نرم‌افزارهایی را که کاربران را قادر به ایجاد، مدیریت و استقرار محیط‌های ابری می‌کند را فعال می‌کند.

حوزه‌ی نرم‌افزار فعال‌کننده‌ی ابر ابزارهای قدرتمندی را برای فیلد کردن ابرهای هیبریدی، خصوصی و عمومی دارد که در زیر برخی از آنها بیان شده‌اند:

- پشته‌ی ابر: یک پلت‌فرم نرم‌افزار IaaS است که توسعه‌ی خصوصی یا سرویس‌های رایانش ابری قابلیت ارتجاعی یا انعطاف پذیری تجاری را که با Amazon EC۲ رقابت دارند را فعال می‌کند. پلت‌فرم پشته‌ی ابر شامل یک سرور مدیریتی و هایپروایزرهایی برای مدیریت و پیاده‌سازی ابر IaaS است.
- *Eucalyptus* : یک زیرساخت منبع یاز برای پیاده‌سازی رایانش ابری روی خوشه‌ها است و هم‌چنین سازگار با Amazon EC۲، S۳ و EBS است.
- *OpenNebula*: ابزاری است که امکا ایجاد استقرارهای ابر خصوصی و عمومی را می‌دهد و هم‌چنین زیرساخت مجازی را مدیریت می‌کند. اما OpenNebula فراتر از این موارد است و مدل‌های ابری متفاوتی را پشتیبانی می‌کند که از جمله‌ی آنها استقرار ابر هیبریدی است.
- *پلت‌فرم رایانش قابلیت ارتجاعی یا انعطاف پذیری انحصاری:* که برای طراحی، استقرار، و مدیریت زیرساخت ابری مجازی قابل برنامه‌ریزی بکار گرفته می‌شود.
- ابر شرکتی *Ubuntu* : پروژه‌های منبع باز زیادی را یکپارچه‌سازی می‌کند، و هم‌چنین امکان استقرار آسان یک ابر خصوصی را نیز می‌دهد.

از محیط فیزیکی به محیط منطقی

محیط دیتاسنتر فیزیکی بعنوان ساختار پشتیبان اساسی برای یک ابر به خدمت گرفته می‌شود. ابر برای تمام ساختارهای ابری چه کوچک و چه بزرگ یکسان است. این محیط پشتیبان فیزیکی باید ایمن باشد. ابن به تنهایی نشان‌دهنده‌ی مجموعه‌ای از مسائلی است که باید در صورتی که اتصال اینترنت، برق، و دیگر ارتباط‌ها و دسترسی فیزیکی ایمن و قابل اعتماد است مورد بررسی قرار بگیرد. مقدار برنامه ریزی پیشرفته که برای دیتاسنتر باید انجام شودمشخص است- در واقع کمتر دیتاسنتری که تمام عناصر فیزیکی را داشته باشد کمتر یافت می‌شود و این از شکاف در شکست برنامه‌های احتمالی استنتاج می‌شود. بین محدوده‌ی فیزیکی دیتاسنتر و این ابزار IT لایه‌هایی از کنترل‌های دسترسی فیزیکی قرار دارند. به همین صورت، این رایانش پیچیده و زیرساخت ذخیره‌سازی نیز ثابت‌کننده‌ی تعدادی لایه از جداسازی منطقی است. هر کدام از این محدوده‌های فیزیکی و منطقی مانعی برای کارآمدی در رایانش ابری هستند، اما آنها برای جلوگیری و ایزوله‌سازی حوزه‌ی آسیبی که دسترسی را غیر مجاز می‌کنند وجود دارند . این محدوده‌ها نه تنها برای محافظت بلکه برای هزینه‌ها نیز باید طراحی شوند. ناکارآمدی در طراحی و فرآیندهای عملیاتی مرتبط باعث تضعیف کارآمدی هزینه ی مدیریت ابر پویا می‌شود. اگر ابری بخواهد به کارآمدی‌هایی که قبلا قول آنها را داده است برسد باید به خوبی طراحی شود.

راه‌اندازی مستقلانه‌ی[۶۵] عملیات ایمن

فرض این که یک ابر می‌تواند بدون شناسایی منشا و امنیت بسیاری از اجزاء تشکیل‌دهنده‌ی ابر پردازش شود بسیار غیر واقعی است. برای مثال، اگر بخشی از یک نرم‌افزار برای کنترل زیرساخت ابر به زیرساخت معرفی شود (بدون ارزیابی امنیت)، ما به گونه‌ای واضح ریسک تشکیل زیرساخت با بدافزار را متحمل می‌کنیم. از آنجایی که بیشتر نرم‌افزارهایی که امروزه استفاده می‌شوند منبع باز می‌باشند، پتانسیل نصب نرم‌افزار با دانلود مستقیم آن از اینترنت امکان‌پذیر است، بدون کنترل موثر روی اعتبار و امنیت آن. در هنگامی که ما سیستمی را برای محصولی ایجاد می‌کنیم این مناسب نمی‌باشد. عملیات‌های امنیتی بسته به پروسه‌ها و پردازش‌هایی هستند که امنیت را پشتیبانی می‌کنند- حتی پیش از این که ابر وارد پردازش‌ها شود.

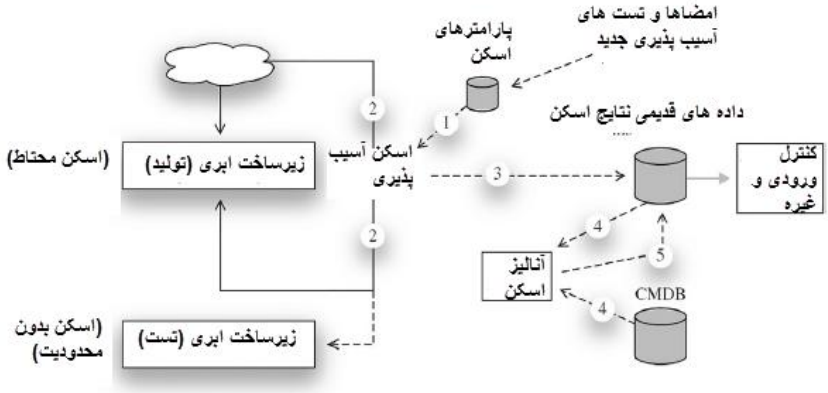
[۶۵] Bootstrapping

کارآمدی و هزینه

در پردازش‌ها و عملیات‌های امنیتی، چندین نوع فعالیت هستند که زمان‌بر هستند و در عین حال عمدتا قابل اجتناب می‌باشند. هم‌چنین فعالیت‌های امنیتی هستند که اجتناب‌پذیر نمی‌باشند اما می‌توانند ساده‌سازی شوند. در دسته‌ی اول توانایی انسان برای اختراع کار غیر ضروری می تواند تنها بخشی از مشکل را توضیح دهد .شناسایی، ارزیابی، ردیابی، اصلاح و گزارش دهی در مورد آسیب پذیری ها به نوعی مشابه مبارزه با آتش سوزی است. چندین استراتژی امکان‌پذیر است: ما می‌توانیم عناصر قابل اشتعال را کاهش دهیم و یا مرکز آتش را پیدا کنیم. جلوگیری از آتش سوزی امکان پذیر نیست، اما اگر ما در برخی از انواع پیشگیری سرمایه گذاری نکنیم، زمان بیشتری را برای تشخیص و گزارش دادن آتش سوزی ها صرف خواهیم کرد. هر محیط رایانشی به طور دوره‌ای آسیب‌پذیری‌های جدیدی را کشف می‌کند. حذف تمام آسیب‌پذیری‌ها (آنهایی که کشف شده‌اند) مناسب است، اما همواره منطقی و امکان‌پذیر نمی‌باشد.

اسکن کد برای آسیب‌پذیری‌ها در چرخه‌ی توسعه یک روش اثبات شده برای کاهش هزینه‌های امنیتی است.

همانطور که بیان شد، دیگر فعالیت‌های پردازشی امنیت قابل اجتناب نمی‌باشند، اما بسیاری از آنها می‌توانند ساده‌سازی شوند و کارآمدی بیشتری داشته باشند. برای مثال، یکی از فعالیت‌های دوره‌ای و ضروری که پردازش‌های امنیتی می‌توانند آنها را انجام دهند اسکن آسیب‌پذیری است. بعد از هر اسکن، نتایج باید ارزیابی شوند، که شامل گام‌های گسسته‌ای است، که از جمله‌ی آنها شناسایی مثبت‌های کاذب است. این فرآیند می‌تواند بعنوان یکسری از فعالیت‌های غیرساخت یافته مدیریت شود، و یا فرآیند می‌تواند بالغ‌تر و ساده‌تر باشد.



شکل 7.2. مدیریت اسکن آسیب پذیری داده ها

شکل ۷-۲: مدیریت اسکن آسیب پذیری داده ها

یک راه برای این کار ایجاد اطلاعات آسیب‌پذیری به گونه‌ای که برای ماشین قابل خواندن باشد، می‌باشد و یا حداقل می‌تواند آنها را به گونه نمایش داد که بتوانند به گونه‌ای نیمه خودکار مدیریت شوند. شکل ۷-۲ نشان‌دهنده‌ی این روش یکپارچه شده در مدیریت داده‌های اسکن آسیب‌پذیری است. توجه داشته باشید که اولین گام این فرآیند انتخاب پارامترهای اسکن است که برای محیط و هدف اسکن مناسب هستند. اگر این اسکن در برابر محیط آزمایش خالص صورت گیرد، اسکنر می‌تواند همه‌چیز را در نظر بگیرد- زیرا که آن محیط تولید نمی‌باشد، از این رو تست مخرب اطلاعات ارزشمندی را که می تواند در محیط تولید مشابه به کار برده شود، به منظور سخت شدن آن و جلوگیری از قطع تولید است را فاش می‌کند.اگر هدف قبلا اسکن شده باشد، منطقی است که با پارامترهای اسکنی که قبلا استفاده شده است آغاز کنیم. همانطور که در شکل ۷-۲ نشان داده شده است، گام‌های بعدی برای شروع اسکن و جمع‌آوری نتایج اسکن است. این نتایج نه تنها شامل داده‌های آسیب‌پذیر و نتایج مرتبط هستند بلکه مقیاسی برای این هستند که اسکن چقدر طول می‌کشد. این به خودی خود اطلاعات کارآمدی است. نتایج اسکن سپس در یک دیتابیس تبدیل یا ضبط می‌شوند که این برای انجام آنالیز نتایج اخیرو ارزیابی تغییرات از نتایج قبلی است. انجام این با دیتابیس ساده است و در غیر این صورت انجامش به صورت دستی بسیار زمانبر است.

باید اشاره شود که آنالیزی که می‌تواند انجام شود، می‌تواند به شدت با رشد شود البته اگر روتین‌های دیتابیس نیز دسترسی به اطلاعات CMDB مدیریت شده درباره‌ی زیرساخت ابر را داشته باشند.

در این روش ، اطلاعاتی که با آدرس IP در ارتباط هستند می‌توانند برای تامین محتوای یک هشدار خاص بکار گرفته شوند، و یک هشدار در ارتباط با سرور وب می‌تواند بعنوان یک مثبت کاذب دسته‌بندی شود، در حالی که همین هشدار اگر مرتبط با یک سرور دایرکتوری باشد بسیار حیاتی است.

فعالیت‌های عملیاتی امنیتی

رابطه‌ی مستقیمی بین مدیریت انتشار، مدیریت پیکربندی، مدیریت تغییر و امنیت است. با این حال، این رابطه اغلب به روش‌های درهمی تبدیل می‌شود، که این به دلیل فقدان کنترل‌های رسمی، و یا بررسی‌های ناموثر تغییرهای پیشنهاد شده است. CM و کنترل تغییر درجه‌ای از نظم در پردازش را که شامل درگیری‌های امنیتی است را تقاضا دارد. قبلا که مهندسان امنیتی در برنامه‌ریزی دخالت داشتند، شانس کمتری وجود دارد که چنین تغییراتی خطرات امنیتی ناخواسته را به همراه داشته باشد. معماران و مهندسان امنیت می‌توانند گام‌ها و پروسه‌هایی را که می‌توانند به شدت امنیت و اعتماد عملیاتی را بهبود بدهند را شناسایی می‌کنند. در بسیاری از روش‌ها، در عملیات، امنیت مجموعه‌ای از کیفیت‌ها است که در دسترسی و یکپارچه‌سازی مشارکت دارند. یکی از نشانه های امنیت مؤثر، کارکرد اقتصاد است . گام‌های پیچیده و پروسه عموما بهینه‌سازی نشده‌اند، و به صورت طبیعی، آنها نشان‌دهنده‌ی فرصت بزرگتری برای شکست و خطا هستند. در مقابل گام‌های ساده‌تر و اتمی‌تر می‌تواند قابل اعتمادتر باشد.

ساخت‌های سرور[۶۶]

بیشتر محیط‌ها تعداد استاندارد برای ساخت‌های سرور دارند. برای مثال، با ساخت سرور ویندوز مایکروسافت، شما با تعدادی گزینه‌های سرور روبرو می‌شوید که با ۳۲ یا ۶۴ بیت آغاز می‌شوند، و از آنجا شما ممکن است یک یا چندتا را از میان سرور اطلاعات اینترنت (IIS)، پروتکل انتقال پرونده های ناشناس (FTP)، مایکروسافت سیلورلایت، پروتکل پیکربندی میزبان پویا (DHCP) و سیستم نام دامنه (DNS) نصب کنید.

[۶۶] Server Builds

برای ابر خصوصی، شما ممکن است که بخواهید راهنماهایی را برای استفاده از محیط تنظیم کنید. برای مثال، یک مجموعه از ساخت‌های سیستم عامل استاندارد باید در نظر گرفته شود؛ این‌ها می‌توانند برای اطمینان از این که کاربر می‌تواند به راحتی و با سرعت آنها را مستقر کند توسعه و تست شوند. این‌ها هم‌چنین هیبریدی از سرورهای لینوکس و مایکروسافت ویندوز نیز هستند، مانند:

- ساخت لینوکس: Red Hat با سرور MySQL
- ساخت لینوکس: Ubuntu با سرور وب Apache
- ویندوز سرور مایکروسافت ۲۰۰۳
- ویندوز سرور مایکروسافت ۲۰۰۸

هر کدام از این‌ها با برنامه‌های استانداردی که شرکت فراهم می‌کند پیش ساخته و نصب شوند، مانند آنتی ویروس، به روز رسانی پچ، نرم افزار حسابرسی و غیره. اگر شما قصد استقرار یک محصول و محیط توسعه را دارید، این قوانین ممکن است برای محیط توسعه کمتر دقیق باشد، اما هر گونه ساخت خارج از حد معمول برای شرکت شما ممکن است به صورت رسمی تایید شود. خلاصه‌ای در مورد محیط های توسعه: به سادگی هیچ اتفاقی برای اتخاذ قطع ارتباط بین محیط های توسعه و تولید وجود ندارد.

رایانش ابری یک پاسخ موثر برای این مساله‌ی مداوم است. هر نمونه سرور باید مقیاس بندی شود تا اطمینان حاصل شود که آن را در حد مجاز قرار می دهد. قراردادن نمونه‌های مجازی زیاد روی یک سرور CPU که همه به یک مصرف CPU زیاد نیاز دارند منجر به نتایج رضایت‌بخشی نمی‌شود. درخواست کاربر برای نشان دادن CPU و حجم حافظه و ذخیره سازی پیش بینی شده بدون محدودیت تنظیم شده و یا شارژ بعید است که موفق باشد. اگر یک نمونه مجازی را روی یک پلت فرم ۳۲ بیتی با حافظه ۲ گیگابایتی یا یک پلتفرم ۶۴ بیتی با حافظه ۶ گیگابایت ارائه دهید، کاربر نمونه‌ای را که کارآیی بالایی دارد را انتخاب خواهد کرد مگر این که هزینه‌ی مربوط به آن وجود داشته باشد. هزینه باید معقول باشد، در غیر این صورت کاربران نمونه‌ای با کارآیی پایین‌تر را انتخاب خواهند کرد و این ممکن است در نتیجه منجر به مسائل مربوط به انکار سرویس محلی با توجه به سرور بارگیری شده شود.

بروزرسانی‌های سرور

مهم نیست که سرورهای شما چه پلت‌فرمی را اجرا می‌کنند، اما بروزرسانی‌های منظمی برای سیستم عامل و برنامه‌ها وجود خواهد داشت. پروسه‌های عملیاتی باید مشخص کنند که شما چگونه و چه

زمان بروزرسانی را روی سرورها انجام می‌دهید. بسته به ساختار ابر و روش تامین، شما ممکن است سرورهای زیادی را برای پچ داشته باشید. با این حال، با یک محیط مجاری، انتقال برنامه‌ها از VM قدیمی به VM بروزشده بهتر است. کاربران و اپراتورها ممکن است در نظر بگیرند که راحت‌تر است که برنامه‌های را به صورت انفرادی استقرار دهند و مدیریت کنند، به ویژه آنهایی که حیات مشخصی دارند. پس از پایان حیات، این سرورهای مجازی می‌توانند حذف شوند و برنامه بدون هیچ تعامل با سرور دیگری خاتمه می‌یابد. با روشن و خاموش کردن برنامه‌ها در صورت نیاز، ابر داخلی را می‌توان ایجاد که به همان روش ابر خارجی کار می‌کند، که این کارآیی کلی ابر را بهبود می‌دهد.

از آنجایی که شما زیرساخت ابری را مستقر می‌کنید، استنتاج این است که شما تعداد زیادی سرور برای استقرار دارید. استقرار پچ‌ها نیازمند تامل و بحث است. امنیت کلی ابر باید حفظ شود، اما این به معنای استقرار همه‌ی پچ‌هایی که منتشر می‌شوند، نمی‌باشد. برای مثال مایکروسافت را در نظر بگیرید، کمپانی مجموعه‌ای از پچ‌ها را در هر ماه منتظر می‌کند. این پچ‌ها توسط مایکروسافت بعنوان حیاتی، مهم و غیره رتبه‌بندی می‌شوند؛ با این حال، به دلیل فاکتورهای ممکن، ممکن است کمپانی شما پچ‌ها را به گونه‌ی دیگری رتبه‌بندی کند. بروزرسانی‌هایی که اساسی هستند نیازمند گسترده شدن[۶۷] می‌باشند. بسته به نرم افزار مجازی‌سازی که مورد استفاده قرار می‌گیرد، ابزارهای مدیریت پچ خودکار متفاوتی می‌تواند برای فعالسازی فرآیند بروزرسانی مورد استفاده قرار بگیرد. برای مثال با استفاده از Vmware (www.vmware.com) شما می‌توانید ابزارهای مدیریت پچی را برای مدیریت پچ کردن هاست و نمونه‌های مجازی مستقر سازید. اگر سرمایه گذاری شما در زیرساخت ابر داخلی برای مدت زمان طولانی با تعداد کافی از سرورها باشد، برخی از انواع اتوماسیون در فرایند به روزرسانی احتمالا به صرفه خواهد بود.

تداوم کسب و کار، پشتیبان گیری و بازیابی

برای اطمینان از این که سرویس‌های ابری برای مشتریان و کاربران در دسترس است، ما از تداوم کسب‌وکار[۶۸]، عبارتی که اشاره به مجموعه‌ی گسترده‌ای از فعالیت‌هایی که به طور مداوم انجام می شود . برای نگهداری سرویس‌ها و در دسترسی هستند، استفاه می‌کنیم. تداوم کسب‌وکار براساس استانداردها، راهنمایی‌ها، و پروسه‌هایی تخمین زده می‌شود که امکان عملیات‌های مداومی را می‌دهند، بدون توجه به شرایط.

---

[۶۷] rolled out
[۶۸] business continuity

بازیابی از حادثه زیر مجموعه‌ای از تداوم کسب‌وکار است و روی سیستم‌های IT و داده‌ها تمرکز دارد. از دید عملیاتی، فعالیت‌هایی که مربوط به تداوم کسب‌وکار به سایر روش ها و فرآیندهای عملیاتی متصل می شوند، از جمله کارآیی پشتیبان گیری مستمر و داده های معکوس به سیستم های بازیابی خارج از محل. ایجاد بک‌آپ باید به صورت بیمه مداوم دیده شود. اگرچه داده‌های پشتیبان‌گیری ممکن است به گونه‌ای ایمن در خارج از محل ذخیره شوند، اما بازسازی یک سیستم از چنین مخزنی بسیار زمان بر است. از دید زمان بازیابی، استفاده از چندین نمونه که بار پردازشی را به اشتراک می‌گذارد ولی آن دارای ظرفیت بیش از حد برای هر یک از سایت ها یا نمونه های آنلاین می باشد، موثرتر می‌باشد، چه برای نگهداری و چه برای وقفه در خدمات.

شکست‌ها

در اوایل دهه‌ی ۱۹۸۰، در یک دیتاسنتر طبقه‌بندی شده که در پنتاگون قرار داشت، یک اپراتور کامپیوتر شیفت شب وظیفه‌هایش در پشتیبان‌گیری یک سیستم حیاتی را آغاز کرد. بک‌آپ‌گیری به گونه طراحی شده است که سیستم در ابتدا باید آفلاین شود. سپس یک دیسک پشتیبان‌گیری در یک درایو دوم نصب خواهد شد، و برنامه ی پشتیبان‌گیری(*DSC روی* Digital Equipment PDP) اجرا خواهد شد. وقتی DSC اجرا می‌شود، محتویات دیسک منبع در دیسک هدف یا بک‌آپ کپی خواهند شد. هنگامی که این فرآیند به اتمام برسد، دیسک منبع ارجینال نصب می‌شود و روی قفسه‌ی ذخیره‌سازی برای پشتیبان‌گیری قرار خواهد گرفت. سپس سیستم با دیسک بک آپ راه‌اندازی مجدد می‌شود. هدف اصلی این سیستم شناسایی این است که بک‌آپ کامل است و منجر به یک نسخه قابل قبول شده است.

متاسفانه، در یک موقعیت که فرآیند شکست می‌خورد، و سیستم نمی‌تواند از دیسک پشتیبان‌گیری راه‌اندازی مجدد شود. اگر بک‌آپ شکست بخورد، و اپراتور دیسک پشتیبان بوت نشده را حذف کرده باشد، اپراتور دیسک ارجینال را بگیرد و آن را در درایو دیگری نصب کنید و تلاش در بوت کردن آن از آن دستگاه را داشته باشد، این‌ها همه شکست است. سپس اپراتور به قفسه‌ی ذخیره‌سازی می‌رود و دیسک پشتیبان اخیر را دریافت می‌کند و آن را در درایو دیگری نصب می‌کند و تلاش در راه‌اندازی آن می‌کند. این نیز یک شکست است. اپراتور مجدد به قفسه‌ی ذخیره‌سازی می‌رود و همین کار را با بک آپ جدید می‌کند. در همین زمان سوپروایزر شیفت می‌رسد و ۱۰ دیسک پراکنده هستند را می‌بیند. چیزی که اتفاق افتاده است این است که بک آپ ارجینال به دلیل سقوط نادر دچار شکست شده است. ایجاد تکنولوژی دیسک درایو کاملا مهروموم شده نمی‌باشد. خرابی هد باعث خراش

صفحه‌ی دیسک خواهد شد. تمام این کارهایی که اپراتورر انجام داده است خطا بوده است. با جایگزینی دیسک غیر قابل بوت شدن با یک دیسک دیگر، او دیسک دوم را نیز دچار همان خرابی کرد. با تغییر مکان دیسک از یک درایوی که هد خراب دارد به بک درایو دیگر، او یک پلاتر خراب را درایو دوم معرفی می‌کند، که باعث خرابی این درایو هم می‌شود. با بازیابی دیسک اخیرو قرار دادن آن و سپس یکی قدیمی تر از آن، اپراتور باعث خرابی تمامی بک‌آپ‌های اخیر برای روزهای گذشته شده است.

مدیریت تغییرات در محیط‌های عملیاتی

یک ارائه‌دهنده‌ی ابر به صورت دوره‌ای باید در ارائه‌ی خدمات و عملکردهای مهم روی سرویس‌هایی که ایجاد شده است تجدید نظر کند. پیش از این که یک نسخه‌ی جدید بتواند استقرار یابد، باید یک محیط تست شود. از آنجایی که یک ابر عملیاتی می‌تواند به عناصر مجزای زیادی برای مدیریت ابر نیاز داشته باشد، این یک سفارش طولانی خواهد بود. چنین عناصری شامل سوئیچ های حامل کلاس، روتر ها، سرورهای دایرکتوری، زیرساخت های امنیتی، تهیه‌کننده‌ها و سایر زیرساخت ها می‌باشند. چندین استراتژی می‌تواند در دستیابی به یک بهبود سیستمی قابل تخمین دست داشته باشد. یک روش ساده این است که توسعه، تست، مرحله بندی، و محیط های عملیاتی کاملا متمایز باشند. محیط‌های توسعه می‌توانند نسبت به حمایت از زیرساخت کاملا متواضع باشد. وقتی که یک نسخه‌ی جدید برای تست آماده است، یک محیط تست اختصاصی مورد نیاز است. بسته به طبیعت نسخه، این محیط تست ممکن است نیازمند استفاده‌ی خاصی از برخی از زیرساخت‌های گران داشته باشد- مانند ورودی روتر یا یک نمونه بزرگ ذخیره سازی. با این حال، نیاز برای محیط‌های تست اختصاص نباید مستلزم قربانی کردن درصد قابل توجه درآمد حاصل از زیرساخت ها باشد. برای یک ابر شرکتی خصوصی، چنین مسائلی وجود دارد. حرکت یک نسخه از تست به سمت تولید تمام خطاهای فایل ها، اسکریپت ها و رویه های پیکربندی را نمایش می دهد. مگر این که آن نسخه تغییری روی سری‌های قبلی باشد.

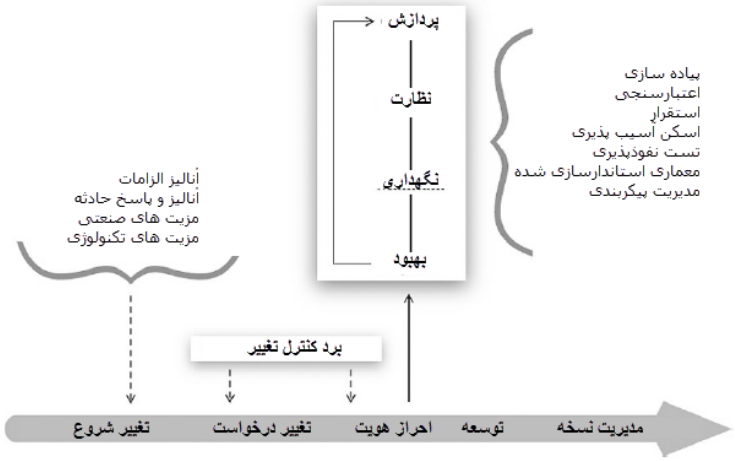
مدیریت نسخه

مدیریت نسخه برای ابر یعنی اطمینان از این که ورژن‌های درستی از نرم افزارها، سخت‌افزارها، فایل‌های پیکربندی، لایسنس‌ها، و دیگر فرآیندهای پشتیبانی بکار گرفته شده است و آنها به درستی نقش خود را ایفا می کنند. اهدف مدیریت نسخه شامل مدیریت موثر تمام فازها از برنامه‌ریزی نسخه تا توسعه‌ی پروسه‌هایی که در برنامه‌ریزی استفاده می‌شوند می‌باشد، که همراه با مدیریت انتظارات مشتری در طول برنامه ریزی است. شکل ۷-۳ نشان‌دهنده‌ی گام‌های رایج در مدیریت نسخه است و نشان‌دهنده‌ی نیازهای اساسی برای مدیریت پیکربندی برای پشتیبانی یک نسخه‌ی جدید است.

مدیریت نسخه‌ی موفق بسته به نظم در فرآیند، استفاده از پروسه‌های رسمی، و بررسی‌های متعدد و دروازه های پذیرش است.شکل ۷-۴ نیز نشان‌دهنده‌ی رابطه‌ی بین مدیریت نسخه و عملیات است، توجه داشته باشید که فعالیت‌های عملیاتی مانند پاسخ و تحلیل حادثه می‌تواند به نیاز به تغییر در ابر کمک کند.

شکل 7.3: گام های عمومی در مدیریت نسخه

شکل ۷-۳: گامهای عمومی در مدیریت نسخه



شکل 7. 4. رابطه ی بین پردازش ها و مدیریت نسخه

شکل ۷-۴: رابطه ی بین پردازش ها و مدیریت نسخه

نسخه‌های می‌توانند شامل تغییرات نرم‌افزاری و سخت‌افزاری عمده یا جزئی باشند و یا اصلاح‌های ضروری. اصلاح‌های ضروری اغلب محدود به بررسی یک تعداد کم از مسائل شناسایی شده یا پچ‌های امنیتی است.

اطلاعات درباره‌ی زیرساخت: مدیریت پیکربندی

یک پیاده‌سازی ابری پیچیده چندین دسته‌بندی از اطلاعات را دارد. این‌های در محدوده‌ی اطلاعات برنامه‌ریزی و طراحی تا اطلاعات پیکربندی ابر است. با این حال به دلیل پویا بودن ابر و به دلیل خودکار بودن بالا در عملیات IT، این نوع از داده‌ها درباره‌ی ابر باید برای مدیریت فرآیندها در دسترس باشند. تمرکز روی زیرساخت فیزیکی (سخت‌افزاری که شامل منابع رایانشی و ذخیره‌سازی است و شبکه‌سازی است)، یکی از آنها ممکن است وسوسه شود که از یک برنامه طراحی کامپیوتری (CAD) برای نشان دادن سرورها، ذخیره سازی و شبکه ها، همراه با کابل های قدرت و زیرساخت های فیزیکی همراه استفاده کند.

پیش از این درباره‌ی نقش CMDB در مدیریت دانش درباره‌ی پیکربندی احراز هویت شده‌ی عناصر، رابطه‌ها و ویژگی‌های آنها صحبت کردیم. همانطور که بحث شد، CMDB مزایای زیادی را به ابر

عملیاتی پیشنهاد می‌دهد. یک CMDB علاوه بر این که می‌تواند برای بازتاب حالت فعلی عنصر فیزیکی ابر بکار گرفته شود، می‌تواند رزیابی و یا حتی مدیریت عناصر ابر مجازی را نیز فعال کند. CMDB نیازی به ذخیره‌سازی اطلاعات منابع مجازی ندارد، اما لازم است دانش و مدیریت منابع مجازی را به قلمرو CMDB فیزیکی و سنتی برساند .

تست‌های آسیب‌پذیری و نفوذپذیری

تست آسیب‌پذیری و نفوذپذیری زیرساخت ابر باید به صورت منظم انجام شود. در بسیاری از موارد، پرسنل عملیاتی و امنیتی ممکن است تخصص و مهارتی برای انجام این کارها نداشته باشند، در این حالت باید برون‌سپاری شود و به شخص سومی واگذار شود. اگر این چنین باشد، باید اطمینان ایجاد کرد که سوم شخص حرفه‌ای است و مهارت‌های قابل اثباتی در این زمینه دارد. اگرچه اکثریت مهارت‌ها و تکنیک‌ها برای تست زیرساخت ابر استفاده می‌شوند یکسان هستند، اما باید مشخص کنید که تست‌کننده‌ها درک درستی از مجازی‌سازی و منظم‌سازی ابر دارند. تست نفوذپذیری باید هدفش کل زیساخت ابر باشد و نه تنها سرورها و عناصر خاص. عناصر شبکه که محیط ابر را فعالسازی می‌کنند باید برای اطمینان از این که امنیت پیکربندی شده است تست شوند. سوئیچ‌ها و روترها می‌توانند آسیب‌پذیری‌های قابل بهره‌برداری داشته باشند، و اگر به درستی پیکربندی نشوند، آنها می‌توانند ترافیک را به گونه‌ای مسیردهی کند که در مقابل نیاز به امنیت ابر است. تست نفوذپذیری و اسکن آسیب‌پذیری بیشتر آسیب‌پذیری‌ها را کشف می‌کند. آسیب‌پذیری‌های کشف‌شده باید درجه‌بندی شوند (حیاتی/بالا/متوسط/پایین). بعنوان یک قاعده‌ی کلی، هر آسیب‌پذیری که درجه‌ی بالا یا حیاتی را داشته باشد باید برای اطمینان از این که امنیت کل ابر نگهداری می‌شود، اصلاح شود. نیاز به اشاره است که بسیاری از آسیب‌پذیری‌هایی که با اسکت و یا تست نفوذپذیری کشف می‌شوند، ناشی از برنامه های توسعه و برنامه نویسی ضعیف است.

پاسخ و نظارت امنیتی

نظارت کلی می‌تواند به دو حوزه تقسیم شود: فیزیکی و سایبری. قطعا نیازهای امنیتی برای کنترل دیتاسنتر وجود دارد. یک دیتاسنتر به خوبی اجرا شده نظارت‌های مداومی دارد و در صورت رخدادن یک هشدار پروسه‌های تعریف شده‌ای را دارد. وقتی شما زیرساخت ابر خود را رشد می‌دهید، نیاز به نظارت افزایش می‌یابد و هم‌چنین پیچیدگی انجام این کار نیز افزایش می‌یابد. بسته به سایز و مکان

تسهیلات ابر، شما ممکن است کارکنان و تجهیرات خاصی را نیاز داشته باشید. نظارت فیزیکی شامل موارد زیر است:

- کنترل ویدئویی
- دسترسی به ورودی[۶۹]
- سنسورهای آبی، آتش و دیگر سنسورهای محیطی
- مصرف برق
- استفاده از امکانات

این فعالیت‌ها وظایف کارکنان امنیت دیتاسنتر است. شما باید پروسه‌های به خوبی تعریف شده‌ای داشته باشید تا اطمینان ایجاد کنید که ورودهای و ضبط‌های ویدئویی به گونه‌ای رای تحقق به نیازهای امنیتی هستند. این پروسه‌ها باید در هنگامی که ارزیابی ریسک انجام می‌شود مورد بررسی قرار گیرند، و بعد از آن تمام ریسک‌های فیزیکی درک شده باید کاهش یابند. عموما، دوربین‌های ضبط ویدئو در حال حاضر به راحتی در سراسر پروتکل کنترل پروتکل انتقال / پروتکل اینترنت (TCP / IP) در دسترس است، با دوربین های فعال بی سیم رواج بیشتری دارند. روشی که این دستگاه ها در اینترنت مشارکت دارند بسیار مهم است. از لحاظ امنیتی و پهنای باند شبکه، زیرا که فید های ویدئویی برای مصرف مقادیر زیادی پهنای باند شبکه مشهورند. یک روش بهتر داشتن شبکه‌ی امنیتی برای چنین ترافیک خارج از باند است و هم‌چنین برای اولویت بندی ترافیک در آن شبکه با توجه به نیازهای سایت است. کنترل سایبری می‌تواند به سه حوره تقسیم شود:

- Housekeeping[۷۰]
- نظارت بر تهدید
- پاسخ حادثه

Housekeeping

کنترل Housekeeping شامل نظارت بر تمام سرورها برای اطمینان از این است که سرور براساس پچ‌ها، بروزرسانی‌های آنتی‌ویروس، مصرف CPU و RAM و غیره بروز هستند. در اینجا دوباره، CMDB نشان دهنده‌ی فرصتی برای افزایش تاثیر در عملیات است. به جای اسکن هر سیستم و

[۶۹] Door access

[۷۰] عملیات کامپیوتری که مستقیماً کمکی برای بدست آوردن نتایج مطلوب نمیکند اما قسمت ضروری یک برنامه مانند راه اندازه مقدمه چینی و عملیات پاکسازی است

شناسایی سیستم‌هایی که به پچ نیاز دارند، تمام ورژن‌ها و اطلاعات مرتبط می‌توانند در CMDB نگهداری شوند، که این برای ایجاد جستجوی سریع است. به طور دوره ای، مهم است که تأیید کنیم که CMDB به طور دقیق محیط فیزیکی و منطقی را که اطلاعات آن را حفظ می کند بازتاب می‌دهد. انجام این کار برای کل ابر یک کار دلسرد کننده است، اما باید برای اجزای تشکیل دهنده زیرساخت های مدیریتی انجام شود. بعلاوه ما می‌توانیم به صورت انتخای سرورهای رایانشی و VMهایی که بارها تکرار شده‌اند را حسابرسی و نمونه‌برداری کنیم. یک راه برای انجام حسابرسی دوره‌ای در برابر محیط منطقی استفاده از نرم‌افزارهای کاتالوگ‌سازی است. Nessus مثال خوبی است که برای بسیاری از مهندسان امنیتی شناخته شده است.

کنترل تهدید

کنترل تهدیدها در ساختار شما هیبریدی از روش‌های دستی و خودکار است. در سطح پایه، شما باید داده‌های حوتدث و هشدار را از سنسورهای IDS/IPS، ثبت‌های آنتی ویروس‌ها، ثبت‌های سیستم از دستگاه‌های مختلف در ساختار شما و ... جمع‌آوری کنید که این‌ها در بخش‌های مختلف این فصل توصیف شده‌اند. با دیتاسنترهای سایز متوسط تا بزرگ، اگر پرسنل از روش‌های دستی برای جمع‌آوری و ارزیابی داده‌ها استفاده می‌کنند، مقداری از داده‌ها عملیات آنها را هدر خواهد داد. از آنجایی که مقدار داده‌ها افزایش می‌یابد، روش‌های دستی نیازمند هدهای اضافه‌ی زیادی می‌باشند، و یا این که شانس تهدید بوجود آید افزایش می‌یابد. ابزارهای خودکار متعددی می‌توانند در این زمینه بکار گرفته شوند. این ابزارها تهدید روی سیستم را محدود می‌کنند. اساسا این ابزارها قادر به کاهش تعداد مثبت‌های کاذبی هستند که در جریان حادثه‌ی خام اتفاق می‌افتد. سپس اپراتور قادر به تمرکز روی تعداد کمتری از تهدیدها است . علاوه بر این، این ابزار را می توان به گونه ای طراحی کرد که هشدارها به گروه های مناسب ارسال شوند: برای مثال هشدار ویروس به یک گروه و شکست سنسور IDS به گروهی دیگر. این گروه‌ها می‌توانند داده‌ها را از سنسورهای مختلفی جمع‌آوری کنند و سپس این داده ها را در یک مکان متصل و مرتبط می‌کنند.

در گذشته کنترل مقدار IT که یک ابر را تشکیل می‌داد می‌توانست شامل مرکز عملیات شبکه‌ی[۷۱] خاصی باشد و شاید یک مرکز عملیات امنیتی[۷۲]. اما امروزه این به طور گسترده‌ای می‌تواند به صورت مجازی و با استفاده از کنسول‌های مبتنی بر وب ایمن که به یک تیم امنیتی امکان اجرا از کل دنیا

[۷۱] network operations center (NOC)
[۷۲] security operations center (SOC)

برای پوشش کامل ۷/۲۴ را می‌دهد، صورت گیرد. NOC و SOC هنوز معقوا=ل هستند، اما مقیاس زیرساخت یا ریسک باید چنین سرمایه‌گذاری را تنظیم کند.

پاسخ حادثه[۷۳]

کنترل و کشف یک تهدید بالقوه تنها شروع کار است. بعد از تایید این که این یک مثبت کاذب نمی‌باشد، شما باید یک برنامه‌ی پاسخ به حادثه را طراح کنید. این‌ها به گونه‌های متفاوتی لیبل‌گذاری می‌شوند- زیاد، متوسط، پایین؛ اصلی/فرعی و غیره- و باید برای هر کدام پاسخ مناسب در نظر گرفته شود.

در پایین‌ترین سطح، حوادث می‌توانند توسط کارمندان عملیات بعنوان بخشی از فعالیت‌های روزانه مورد رسیدگی قرار گیرند و عموما نیازی به تشدید نمی‌باشد. این باید به منظور اطمینان از این که هیچ الگوی کلی وجود ندارد و اطمینان از این که هر گونه کار پیگیری (مانند نصب پچ های مهم) انجام می شود، باید ردیابی شود. سطح بعدی حادثه زمانی است که برخی از سرورها را تحت تاثیر قرار می‌دهد، مانند شکست در تامین برق در کل رک و یا شکست شبکه برای یک بخش از شبکه‌ی شما. اگرچه کارکنان بخش عملیات این شکست را رفع خواهند کرد اما احتمال دارد که برخی از شکل‌های ارتباط نیاز به ارسال به خارج از خوزه‌ی کارکنان داشته باشند.

در بالاترین سطح حوادث اصلی هستند که درصد زیادی از کاربران را تحت تاثیر قرار می‌دهد، چنین حوادثی شامل یک توافق هستند و در غیر این صورت اعتبار را تحت تاثیر قرار می‌دهند. مجددا در این مورد نیز برنامه‌ریزی کلید موفقیت است. پاسخ‌ها اغلب شامل محدوده‌ی بیشتری از افراد نسبت به فقط کارکنان بخش عملیات هستند و نیاز به مدیریت با دقت حوادث دارند.

بهترین تجربه‌ها

در دهه‌ی ۱۹۹۰، ISF[۷۴] استاندارد تجربه‌ی های خوب SoGP را منتشر کرد، که در آن مجموعه‌ای از اطلاعات امنیتی از تجربه‌های کارآمد مشخص شد. در سال ۲۰۰۷ این مجموعه به روزرسانی شد . SoGP از تحقیقاتی پیچیده و بررسی‌های تجربه‌های کارآمد در حوزه‌های امنیتی و مدیریت حادثه توسعه یافت. SoGP اغلب در پیوستگی با دیگر استانداردها مورد استفاده قرار می گیرد، مانند ISO/IEC ۲۷۰۰۲ و COBIT .

[۷۳] Incident Response
[۷۴] Information Security Forum

در ۱۹۹۶ ، Marianne Swanson و Barbara Guttman ، اصول عمومی پذیرفته شده و تجربه‌ها برای سیستم‌های تکنولوژی اطلاعات امنیتی را منتشر کردند. آنها هشت اصل زیر را مشخص کردند:

- امنیت کامپیوتر از ماموریت سازمان پشتیبانی می کند
- امنیت کامپیوتر یک عنصر یکپارچه از مدیریت خوب است
- صاحبان سیستم‌ها مسئولیت‌های امنیی در خارج از سازمان‌هایشان دارند.
- مسئولیت‌ها و پاسخگویی‌های امنیت کامپیوتر باید صریح باشد
- امنیت کامپیوتر نیازمند یک روش پیچیده و یکپارچه‌سازی شده است.
- امنیت کامپیوتر باید به صورت دوره‌ای ارزیابی مجدد شود.
- امنیت کامپیوتر متشکل از فاکتورهای گروهی است.

انعطاف‌پذیری در عملیات

بدون در نظر گرفتن تداوم و حکمرانی کسب‌وکار تعریف امنیت دشوار است. در جایی که تداوم کسب و کار به سوی غلبه بر هر گونه وقفه مهم خدمات (و پیامدهای آن) هدایت می شود، اداره فناوری اطلاعات یک شکل از فرمان و کنترل IT است. اختیارداری یک فرآیند یا مجموعه‌ای از اقدامات است که هدفشان تحویل نتایج IT مورد انتظار است. سازمان ها با ایجاد موانع متعدد در ایجاد امنیت به یک عامل موثر برای دستیابی به اهداف کلی سازمان می پردازند. برای شروع، اکثر سیستم ها واقعا قادر به مقاومت در برابر حتی شکست های بی اهمیت هستند بدون اینکه برخی از وقفه های سرویس را برطرف کنند. همانطور که در گزارش‌ها توسط دانشگاه Carnegie Mellon بیان شده است: حمایت از انعطاف پذیری عملیاتی نیاز به یک قابلیت اصلی برای مدیریت ریسک عملیاتی - خطرات ناشی از عملیات روزمره- دارد. مدیریت ریسک عملیاتی برای تضمین موفقیت ماموریت مهم است. برای برخی صنایع مانند بانکداریو امور مالی، آن نه تنها یک تابع کسب‌وکار ضروری است بلکه یک ابزار منظم‌سازی نیز به شمار می‌آید. فعالیت‌هایی مانند امنیت، تداوم کسب‌وکار و مدیریت عملیات IT به دلیل هدف اساسی که برای شناسایی، آنالیز و کاهش ریسک‌های عملیاتی دارند، مهم هستند. به نوبه‌ی خود، به دلیل این که آنها ریسک‌های عملیاتی را پشتیبانی می‌کنند، آنها به طور مستقیم انعطاف‌پذیری عملیاتی را تحت تاثیر قرار می‌دهند. یکی از اهداف انعطاف‌پذیری در IT کاهش تاثیر شکست و فاجعه است. کاهش احتمال فاجعه یک هدف اساسی است ولی توانایی پوشش از فاجعه نیز به همان اندازه اهمیت دارد.

خلاصه

بسته به نحوه‌ی انتخاب مدل ابری (خصوصی، عمومی، گروهیی و هیبریدی) و بسته به چگونگی تحویل سرویس‌های مبتنی بر ابر (IaaS، PaaS، و SaaS)، رایانش ابری فرصت‌های متفاوتی را برای تغییر ایجاد می‌کند. بعنوان یک مدل جدید برای IT، رایانش ابری ا ز مزایای مختلف توسط رقبا در همان صنعت، توسط فروشندگان و ارائه دهندگان سرویس های ابر، و توسط مصرف کنندگان و مشترکان استفاده می شود. روشی که توسط آن سازمان از رایانش ابری مزایا می‌برد بسته به چگونگی ارزیابی اطلاعات و منابع ارتباطی موجود و چگونگی رویارویی با این انتقال به مدل رایانش است. در حال حاضر می توانیم این رخداد را ببینیم و موفقیت آن بستگی به توانایی سازمان در درک فرصت ها و تغییر مسیردهی تغییرات به سوی فن آوری ها، محصولات و مفاهیم موجود و در حال ظهور است و این که ابر را به عنوان مدل جدید فناوری اطلاعات در نظر بگیریم. اگرچه ابرهای خصوصی می‌توانند به مقیاس زیادی برسند و مشتریان داخلی زیادی را سرویس‌دهی کنند، اما ابرها خصوصی نسبتا کوچک‌تر هستند. این باعث می‌شود که ابر عمومی مزیت‌هایی داشته باشد که از جمله‌ی آنها بازگشت سرمایه‌گذاری روی ابزارها و قابلیت‌های امنیتی است که ذاتا گران هستند و یا این که نیازمند سرمایه‌گذاری در خبره‌ها برای پیاده‌سازی و پردازش آن است. یکی از مزیت‌های IT با مدل ابری این است که هنگامی که زیرساخت مناسب است، بیشتر سخت‌افزارهای فیزیکی IT و فعالیت‌های شبکه‌سازی دیگر با مشکل اجرا نمی شوند.

با انتخاب رایانش ابری بعنوان یک مدل برای IT، سازمان ها می‌تواند به دور شدن از جنبه‌های سخت‌افزار محور سنتی به سمت استراتژی های مبتنی بر سرویس حرکت کنند. ابرها مزایای زیادی را پیشنهاد می‌دهند که فراتر از ساختار IT ای است که آنها استفاده می‌کنند. تعاملات واضحی وجود دارند که شامل کنترل روی برنامه‌ها و داده‌ها است، و انطباق با قوانین و مقررات و حتی امنیت دارد. مدل ابر نیز مقیاس‌پذیری بیشتری را به همراه دارد، و با استفاده از ***fail in place***، ابر قابلیت اعتماد و افزونگی بیشتری را نیز به همراه دارد. نیازهای هیبریدی برای قدرت محاسباتی، ذخیره سازی داده ها و پهنای باند همچنان به تقاضای سیستم های بسیار قدرتمند ادامه می دهد. برنامه های کاربردی با شدت زیاد به دسترسی به مقیاس های ذخیره سازی بستگی دارند. الزامات ذخیره سازی در مقیاس پتابایت، تابع مقیاس ترابایتی را در بر می گیرند، و به زودی ذخایر ذخیره سازی مجاز می تواند مقیاس petabyte را از بین ببرد. علاوه بر مزایای دیگر، مدل محاسبات ابری چنین برنامه های ذخیره سازی در مقیاس بزرگ را بیشتر امکان پذیر می کند.

نکته:

برخی از منابع اینترنتی برای اطلاعات سایت‌هایی هستند که افراد حرفه‌ای و همتاها در آن مشارکت دارند. اگرچه تعداد زیادی سایت این چنینی وجود دارد اما چندین سایت هستند که تنها برای رایانش ابری هستند. در زیر تعدادی از آنها را بیان کرده‌ایم:

- گروه‌های گوگل: گوگل، گوریل اینترنتی ۹۰۰ پوند اینترنت، یک ابزار بزرگ و غنی را برای همکاری میان گروهی از افراد با منافع مشترک فراهم کرده است. بزرگترین مشکل گروه گوگل تعداد زیادی از گروه ها است! در بسیاری از این گروه ها رهبران بسیار مهمی در آن زمینه عضویت دارند.
- اتحاد ابر امنیتی[۷۵]: به نظر می رسد که این سازمان تغییری در شرایط سازماندهی اعتباربخشی خود به خود انجام داده است.
- *LinkedIn*: این سایت شبکه حرفه ای با حدود ۱۰۰ میلیون عضو حرفه ای در بیش از ۲۰۰ کشور است. این یک ابزار شبکه بسیار موثر برای پیدا کردن مشتریان بالقوه‌ی یک محصول، ارائه‌دهنده‌ها و خبره‌ها در آن زمینه است. گروه های LinkedIn در زمینه های امنیت و رایانش ابری بسیار فعال هستند و طیف گسترده ای از بحث های در حال انجام در مورد موضوعات متعدد فنی، بازار و دیگر مسائل مرتبط است.

منابع

۱. Swanson M., Guttman B. NIST SP ۸۰۰-۱۴, “Generally Accepted Principals and Practices for Securing Information Technology Systems,” National Institute of Standards and Technology, Technology Administration; ۱۹۹۶.

۲. Ibid.

۳. Caralli R., Stevens J., Wallen C., Wilson W. *Sustaining Operational Resiliency: A Process Improvement Approach to Security Management*. CMU Networked Systems Survivability Program; ۲۰۰۶.

[۷۵] *The Cloud Security Alliance*