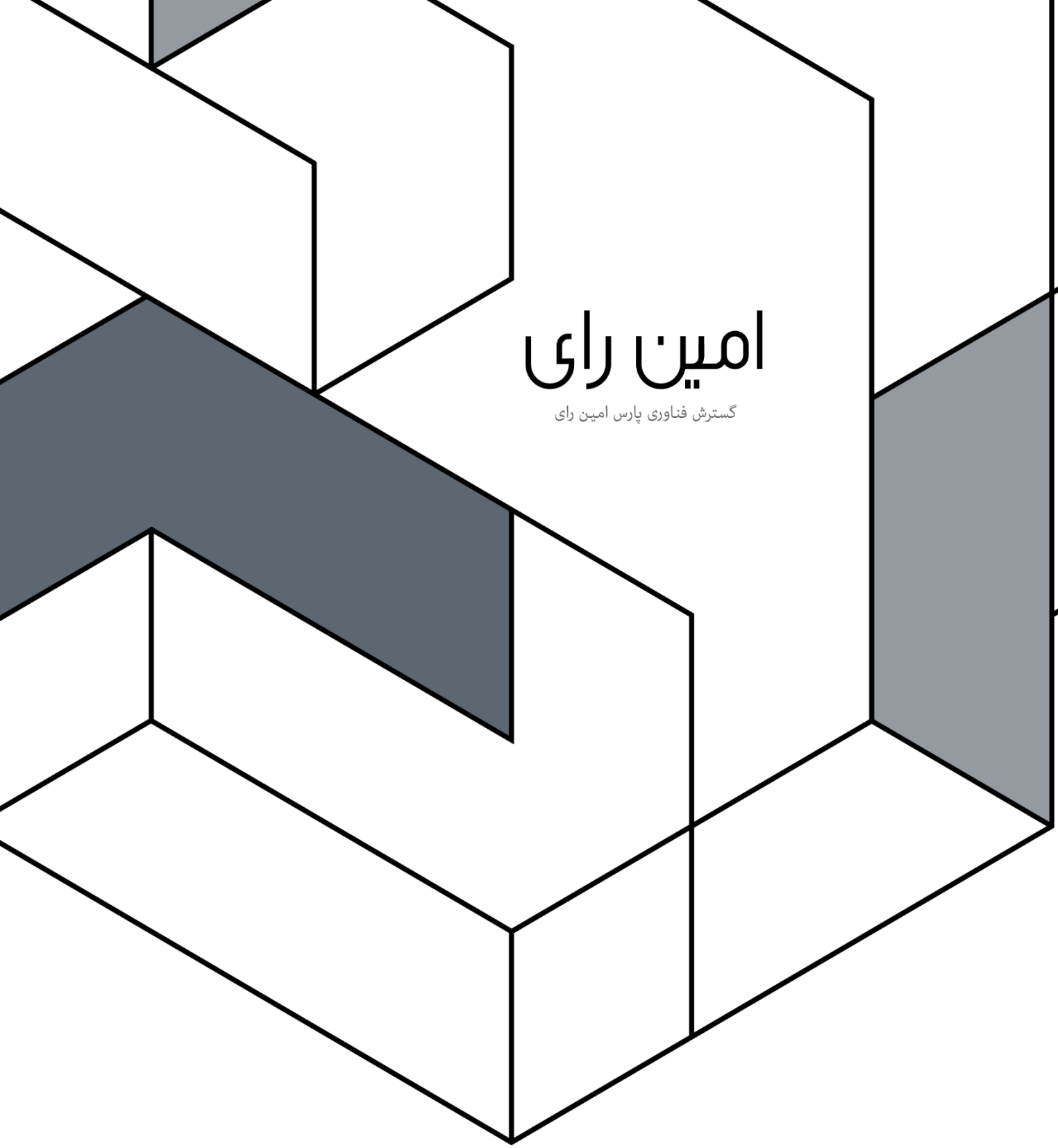


نـام سنـد فهرست نیازمندی‌های امنیتی گردآوری شده برای سیستم سادا	
تهیه کننده: گسترش فناوری پارس امین رای **طبقه بندي:** محرمانه	**تاریخ نشـر:** ۱۴۰۲/۰۶/۱۱
کد سند: AMR_SADA_Security_Requirements_14020611_v1.0	**تاریخ آخرین بازنگري:** ۱۴۰۲/۰۶/۲۵

به‌نام خدا

فهرست مطالب

عنوان **صفحه**

فصل ۱- مقدمه

گردآوری و مستندسازی نیازمندی‌های امنیتی یکی از فعالیت‌های مهمی است که باید در همان فازهای ابتدایی چرخه حیات توسعه امن نرم‌افزارها انجام گیرد. نتیجه این فعالیت به توسعه‌دهندگان در پیاده‌سازی امن نرم‌افزار و همچنین به کارشناسان تست نرم‌افزار در استخراج test caseهای امنیتی و به کارشناسان استقرار و عملیات، در پیکربندی درست و امن نرم‌افزار و محیط عملیاتی آن کمک خواهد کرد.

نیازمندی‌های امنیتی نرم‌افزار را می‌توان از منابع مختلفی مانند الزامات بالادستی سازمان، استانداردها و به‌روش‌های موجود مانند ISO 15408، OWASP ASVS و فعالیت‌های دیگر چرخه حیات توسعه امن نرم‌افزارها همانند مدلسازی تهدیدها و تست‌های امنیتی انجام شده شناسایی کرد.

در سند پیش رو که توسط شرکت «امین رای» تهیه شده است، نیازمندی‌های امنیتی سیستم سادا شناسایی و مستند شده‌اند. این نیازمندی‌ها برمبنای نتایج بدست آمده از مدلسازی تهدیدها که به عنوان یک فعالیت مجزا برای سیستم سادا انجام شده است و گزارش آن در یک فایل با عنوان AMR_SADA_Potential_Threats_14020417_v1.0 در اختیار تیم توسعه سیستم سادا قرار گرفته است، شناسایی شده‌اند.

در ادامه ابتدا در فصل اول، فرایند استخراج نیازمندی‌های امنیتی تشریح شده است، سپس نیازمندی‌های شناسایی شده در فصل دوم تشریح شده‌اند.

فصل ۲- مروری بر فرایند گردآوری نیازمندی‌های امنیتی

برخی بر این باورند که آسیب‌پذیری‌هایی که در نرم‌افزارها شناسایی می‌شوند، ناشی از اشتباهاتی است که توسعه‌دهندگان هنگام کدنویسی نرم‌افزار مرتکب می‌شوند. اما واقعیت آن است که بخش قابل توجهی از این آسیب‌پذیری‌ها به دلیل خطاهایی است که در فازهای طراحی و جمع‌آوری نیازمندی‌ها رخ می‌دهند. عدم شناسایی درست نیازمندی‌های امنیتی یکی از این خطاها است.

به عنوان مثال، رمزنگاری داده‌های محرمانه حین انتقال روی شبکه، به عنوان یکی از نیازمندی‌های امنیتی نرم‌افزار شناسایی و اولویت‌بندی شده باشد تا توسعه‌دهندگان در فاز پیاده‌سازی نرم‌افزار، بر ضرورت انجام آن آگاه و ملزم باشند. از طرف دیگر، این نیازمندی‌ها باید به درستی تبیین شده باشند تا بعدها هیچ نقطه ابهامی در پیاده‌سازی آنها وجود نداشته باشد. به عنوان مثال اگر در تبیین نیازمندی فوق، نوع الگوریتم مورداستفاده برای رمزنگاری داده‌ها، طول کلید مورداستفاده برای رمزنگاری و مد عملیاتی الگوریتم به صورت دقیق مشخص نشده باشد، در فاز پیاده‌سازی آن توسعه‌دهندگان دچار سردرگمی می‌شوند و نمی‌دانند که دقیقا به چه صورت باید آن را پیاده‌سازی کنند. پیامد این اشتباه می‌تواند منجر به آسیب‌پذیر شدن نرم‌افزار شود. بنابراین گردآوری درست و دقیق نیازمندی‌های امنیتی نرم‌افزار یکی از فعالیت‌های مهمی است که در همان فازهای ابتدایی چرخه حیات آن باید انجام بگیرد. در ادامه این بخش، فرایند گردآوری نیازمندی‌های امنیتی نرم‌افزارها آورده شده است.

۱-۲- گردآوری نیازمندی‌های امنیتی

مرحله اول از فرایند گردآوری نیازمندی‌های امنیتی، شناسایی آنها با مراجعه به منابع مختلف است که در این بخش به آن پرداخته شده است. علاوه بر این، انواع نیازمندی‌های امنیتی قابل گردآوری نیز تشریح شده‌اند.

۱-۱-۲- نیازمندی‌های امنیتی کارکردی و غیرکارکردی

نیازمندی‌های امنیتی نرم‌افزار ممکن است رفتار کارکردی سیستم را که منجر به امن شدن آن خواهد شد، توصیف کنند. در این حالت می‌توان گفت، این نیازمندی‌ها، نیازمندی‌های امنیتی کارکردی[۱] سیستم هستند؛ زیرا مشخص می‌کنند که سیستم باید چه کاری انجام دهد. نیازمندی‌های مرتبط با کنترل‌های امنیتی عمومی همانند احرازهویت، کنترل دسترسی و حفاظت از داده‌ها نمونه‌ای از آنها هستند. برخی دیگر از نیازمندی‌های امنیتی، کارکرد مستقیم سیستم نیستند، بلکه مشخص می‌کنند که سیستم چگونه باید باشد. مثلا اینکه سیستم قابل ممیزی باشد و یا همواره در دسترس باشد، یک نیازمندی امنیتی غیرکارکردی[۲] است. هرچند ممکن است این نیازمندی‌ها نیز در برخی از استانداردها یا الزامات بالادستی جزو نیازمندی‌های کارکردی برشمرده شده باشند.

---

[۱] Functional Security Requirement

[۲] Non-functional Security Requirement

۲-۱-۲- منابع استخراج نیازمندی‌های امنیتی

نیازمندی‌های امنیتی را هم می‌توان همانند نیازمندی‌های غیرامنیتی، در طی جلساتی با مشتریان، کاربران و سایر ذینفعان، از طریق شناخت انتظارات و نیازهای آنها، استخراج کرد. اما با توجه به اینکه ممکن است این افراد آگاهی کاملی نسبت به نیازها و انتظارات خود از سیستم در حوزه امنیت آن نداشته باشند، بهتر است تیم توسعه با همکاری کارشناسان امنیت و با بهره‌گیری از به‌روش‌ها، چارچوب‌ها و استانداردهایی که در این حوزه کمک‌کننده هستند، این شناخت را کامل کنند. استاندارد OWASP ASVS و ISO 15408 نمونه‌ای از استانداردهایی هستند که جهت شناسایی نیازمندی‌های امنیتی می‌توانند مورد استفاده قرار گیرند. علاوه بر این، شناسایی تهدیدها و ریسک‌های امنیتی که ممکن است نرم‌افزار با آنها مواجه شود هم می‌تواند در استخراج نیازمندی‌های امنیتی که منجر به کاهش یا رفع این تهدیدها و ریسک‌ها شود، مفید خواهند بود. در ادامه این بخش، توصیف مختصری از هریک از منابع قابل استفاده برای شناسایی نیازمندی‌ها آورده شده‌اند.

۲-۱-۲-۱- مدلسازی تهدیدها

مدلسازی تهدیدها، همانند استخراج و گردآوری نیازمندی‌های امنیتی، یکی از فعالیت‌های مهمی است که در چرخه حیات توسعه امن نرم‌افزارها باید انجام گیرد. در نتیجه انجام این فعالیت، تهدیدهای امنیتی که نرم‌افزار با آنها روبرو است، شناسایی می‌شوند. برای کاهش خطر یا رفع این تهدیدها، می‌بایست راهکارهایی پیاده‌سازی شوند که این راهکارها در قالب نیازمندی‌های امنیتی نرم‌افزار باید تشریح شوند. این فعالیت قبلا برای سیستم سادا انجام شده است و نتیجه آن در یک فایل با عنوان AMR_SADA_Potential_Threats_14020417_v1.0 ارائه شده است.

۲-۱-۲-۲- استاندارد OWASP ASVS

استاندارد OWASP ASVS چارچوبی برای تست و وارسی کنترل‌های امنیتی در نرم‌افزارها است و شامل الزاماتی برای انجام این وارسی است. این الزامات در ۱۴ دسته و سه سطح مختلف، دسته‌بندی شده‌اند. هریک از این الزامات می‌توانند به یک یا چند ویژگی و قابلیت امنیتی که باید در نرم‌افزار پیاده‌سازی شوند، نگاشت شوند. بنابراین به کمک این استاندارد نیز می‌توان برخی از نیازمندی‌های امنیتی را شناسایی کرد.

۲-۱-۲-۳- استاندارد ISO 15408

استاندارد ISO 15408 که با عنوان Common Criteria هم شناخته می‌شود، استانداردی بین‌المللی برای ارزیابی امنیتی انواع محصولات فناوری اطلاعات است. در این استاندارد، مهمترین مولفه‌های امنیتی که برمبنای آنها می‌توان نیازمندی‌های امنیتی محصول را شناسایی و تشریح کرد، آورده شده است. این نیازمندی‌ها شامل کارکردهای امنیتی ضروری است که می‌بایست در یک محصول فناوری اطلاعات وجود داشته باشد. همچنین نحوه ارزیابی محصول و تضمین آنکه این کارکردها در محصول وجود دارند نیز در استاندارد آورده شده است. بنابراین از این استاندارد نیز می‌توان جهت شناسایی نیازمندی‌های امنیتی نرم‌افزار به عنوان یک محصول فناوری اطلاعات، استفاده کرد.

۴-۲-۱-۲- استخراج موارد سوءاستفاده

گاهی اوقات لازم است نحوه استفاده از نرم‌افزار از دیدگاه یک مهاجم هم بررسی شود. این بررسی منجر به آن خواهد شد که سوءاستفاده‌های احتمالی از کارکردهای نرم‌افزار استخراج شوند. برای پیشگیری و تشخیص این سوءاستفاده‌ها، باید نیازمندی‌هایی در نرم‌افزار دیده شوند که جزو نیازمندی‌های امنیتی آن خواهند شد. البته ناگفته نماند که برخی از موارد سوءاستفاده را می‌توان در فرایند مدلسازی تهدیدها نیز شناسایی کرد.

۲-۲- اولویت‌بندی نیازمندی‌های امنیتی

پس از شناسایی نیازمندی‌های امنیتی، برای اطمینان از پیاده‌سازی آنها توسط اعضای تیم‌ها، این نیازمندی‌ها باید اولویت‌بندی شوند. اولویت‌بندی نیازمندی‌ها را می‌توان برمبنای اهمیت و ضرورت آنها، قابل پیاده‌سازی بودنشان در نرم‌افزار و همچنین میزان تاثیری که بر وضعیت امنیت نرم‌افزار خواهند داشت، انجام داد.

۳-۲- مستندسازی نیازمندی‌ها

نیازمندی‌های امنیتی پس از شناسایی و اولویت‌بندی باید مستند شوند و در اختیار ذینفعان به خصوص توسعه‌دهندگان، کارشناسان تست، کارشناسان استقرار و عملیات نرم‌افزار قرار گیرند. برای مستندسازی نیازمندی‌ها، بهتر است از قالب‌های استانداردی که اعضای تیم‌ها و دیگر ذینفعان با آنها آشنا هستند، استفاده شود.

قالب‌های ارائه شده در استانداردهایی همانند ISO 29148، همانند قالب SRS[۱] که برای مستندسازی نیازمندی‌های کارکردی و غیرکارکردی نرم‌افزارها استفاده می‌شوند و همچنین قالب‌های مستندسازی User Storyها، برای مستندسازی نیازمندی‌های امنیتی نیز می‌توانند مورداستفاده قرار گیرند. البته در صورتی که در تیم‌ها، یک قالب مشخص و توافق‌شده‌ای برای مستندسازی نیازمندی‌ها استفاده می‌شوند، می‌توان از همان قالب برای مستندسازی نیازمندی‌های امنیتی نیز استفاده کرد.

مستقل از آنکه از چه قالبی برای مستندسازی نیازمندی‌های امنیتی استفاده می‌شود، بهتر است هنگام نوشتن این نیازمندی‌ها، به ویژگی‌های اشاره شده در چارچوب SMART توجه شود. این ویژگی‌ها به شرح ذیل هستند:

- **Specific**: نیازمندی تعریف شده باید به صورت دقیق، کامل، بدون ابهام و کاملا شفاف و ساده، آن‌چیزی که موردنیاز است را مشخص کند.
- **Measurable**: نیازمندی تعریف شده باید بعد از پیاده‌سازی، قابل تست و ارزیابی باشد.

[۱] Software Requirements Specification

- **Achievable**: نیازمندی تعیین شده باید قابل دستیابی باشد. نیازمندی که پیاده‌سازی آن بسیار دشوار و یا غیرممکن باشد، ارزش افزوده‌ای برای نرم‌افزار خلق نخواهد کرد.
- **Relevant**: نیازمندی تعیین شده باید مرتبط با نرم‌افزار باشد و منجر به خروجی مثبت برای آن شود. به عبارت بهتر، این نیازمندی با بتواند یک ارزش افزوده برای نرم‌افزار خلق کند.
- **Time-bound**: نیازمندی تعیین شده باید به گونه‌ای باشد که در یک بازه زمانی مشخص، قابل پیاده‌سازی باشد.

فصل ۳- نیازمندی‌های امنیتی گردآوری شده برای سادا

در فصل پیش‌رو فهرست نیازمندی‌های امنیتی سیستم سادا، که در فاز فعلی پروژه، برمبنای نتایج مدلسازی تهدیدها این سیستم، شناسایی شده‌اند، آورده شده است. در فاز بعدی پروژه، دیگر نیازمندی‌های امنیتی که برمبنای استاندارد OWASP ASVS، ISO 15408 و موارد سوءاستفاده از کارکردهای نرم‌افزارها قابل استخراج خواهند بود، به فهرست این بخش افزوده خواهند شد.

قالب مورداستفاده برای مستندسازی این نیازمندی‌ها، یک قالب دلخواه است که شامل بخش‌هایی از قالب SRS و قالب مورداستفاده برای مستندسازی User Storyها در پروژه‌های چابک، است. در ادامه این بخش، سند آورده شده است.

۳-۱- هدف سند

هدف از تدوین این سند، تشریح نیازمندی‌های امنیتی سیستم سادا است که برمبنای خروجی فعالیت مدلسازی تهدید شناسایی شده‌اند و راهکارهای کاهش خطر تهدیدات یا رفع آنها را بیان می‌کنند.

۳-۲- مخاطبین

مخاطبین این سند، تیم توسعه و تیم استقرار و عملیات سیستم سادا هستند.

۳-۳- تشریح نیازمندی‌ها

در این بخش، تشریح نیازمندی‌های امنیتی آورده شده است. تمام نیازمندی‌های تشریح شده در این بخش، جزو نیازمندی‌های امنیتی کارکردی سادا هستند. زیرا منجر به فراهم شدن یک کارکرد امنیتی برای سیستم می‌شوند.

در تشریح هر نیازمندی، ابتدا یک توصیف مختصر از آن آورده شد است. سپس معیارهای پذیرش آن نیازمندی، پس از آن، سرویس یا سرویس‌های امنیتی که توسط هر نیازمندی، فراهم می‌شوند، مشخص شده‌اند. همچنین اولویت پیشنهادی برای نیازمندی نیز از اعداد ۱ تا ۴ مشخص شده‌اند که عدد ۱ بیانگر بالاترین اولویت و عدد ۴ کمترین اولویت را نشان می‌دهند. مسئولیت تیم‌های توسعه و عملیات سادا در قبال هر نیازمندی و در نهایت، توضیحات تکمیلی در مورد علت وجود نیازمندی آورده شده‌اند.

REQ-01: استفاده از کلمات عبور قوی برای کاربران

توصیف نیازمندی: سیستم باید از کلمات عبور قوی برای کاربران استفاده کند.

معیارهای پذیرش (Acceptance Criteria):

- حداقل طول کلمه عبور باید ۱۲ کاراکتر (بعد از ادغام کاراکترهای space پشت سرهم) و حداکثر ۶۴ کاراکتر باشد.

- امکان استفاده از انواع کاراکترهای قابل چاپ برای کلمات عبور وجود داشته باشد.

سرویس‌های امنیتی مربوطه: احراز هویت (Authentication)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید در سرور LDAP و توسط ادمین آن پیکربندی شود.

توضیحات تکمیلی: هرچه کلمه عبور مورداستفاده برای کاربران قویتر باشد، امکان حدس آن توسط مهاجم یا بدست آوردن آن در حملاتی همانند Brute Force سخت‌تر خواهد بود.

REQ-02: بکارگیری مکانیزم قفل حساب کاربری

توصیف نیازمندی: سیستم باید بتواند پس از ۱۰ تلاش ناموفق یک کاربر برای ورود، حساب کاربری مربوطه را به مدت ۱۵ دقیقه قفل کند.

معیارهای پذیرش (Acceptance Criteria):

- حدآستانه تعداد تلاش‌های ناموفق باید قابل تنظیم باشد.
- مدت زمان قفل ماندن حساب کاربری باید قابل تنظیم باشد.
- این نیازمندی باید برای تمام نقش‌ها اعم از کاربران بهره‌بردار، مدیر سادا و مدیران ارشد سادا در نظر گرفته شود.

سرویس‌های امنیتی مربوطه: احراز هویت (Authentication)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی ابتدا باید در سرور LDAP و توسط ادمین آن پیکربندی شود. سپس پیکربندی‌های موردنیاز توسط تیم توسعه در سرویس Keystone نیز در نظر گرفته شوند.

توضیحات تکمیلی: قفل حساب کاربری برای پیشگیری از حملاتی همانند Brute Force برای بدست آوردن حساب کاربری معتبر، الزامی است.

REQ-03: بکارگیری مکانیزم احرازهویت دوفاکتوری

توصیف نیازمندی: سیستم باید بتواند امکان احرازهویت کاربران را از طریق مکانیزم احرازهویت دو فاکتوری[۱] فراهم کند.

معیارهای پذیرش (Acceptance Criteria):

- این نیازمندی برای نقش‌های کاربر مدیر و مدیر ارشد سادا الزامی است.
- این نیازمندی برای نقش کاربر بهره‌بردار به صورت یک گزینه قابل فعالسازی است.
- فاکتور دوم یک کد TOTP[۲] است که از طریق کانال SMS و Email قابل ارسال است.

سرویس‌های امنیتی مربوطه: احراز هویت (Authentication)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی ابتدا باید در سرور LDAP و توسط ادمین آن پیکربندی شود. سپس پیکربندی‌های موردنیاز توسط تیم توسعه در سرویس Keystone نیز در نظر گرفته شوند.

توضیحات تکمیلی: احراز هویت دو فاکتوری برای پیشگیری از حملاتی همانند Brute Force جهت بدست آوردن حساب کاربری معتبر، مفید است.

REQ-04: بکارگیری مکانیزم کپچا

توصیف نیازمندی: سیستم بهتر است پس از ۵ تلاش ناموفق یک کاربر برای ورود، یک کپچا نمایش دهد.

معیارهای پذیرش (Acceptance Criteria):

- کپچا نباید با استفاده از ابزارهای خودکار قابل حل شدن نباشد.
- محتوای آن آنتروپی لازم را داشته باشد و قابل حدس نباشد.
- بعد از مدت کوتاهی (مثلا ۲ دقیقه) منقضی شود.
- پس از هر درخواست کاربر، چه به صورت موفق و چه به صورت ناموفق باید کپچا تغییر کرده و اجازه ارسال درخواست با کپچای قبلی به کاربر داده نشود.
- بعد از ورود موفق، کپچا منقضی گردد.
- زمانی که یک کد کپچای جدید ایجاد شد، کد قبلی منقضی شود.
- طول کپچا نباید ثابت باشد و باید به صورت تصادفی تغییر کند.
- سایز کاراکترهای کپچا باید تصادفی باشد.

---

[۱] TFA (Two-Factor Authentication)

[۲] Time-based One-time Password

- کاراکترهای کپچا بهتر است حالت اعوجاج داشته باشند.
- نباید از کاراکترهای پیچیده استفاده کرد، چون دقت کاربر را کاهش می‌دهند.
- بهتر است کاراکترها با استفاده از تکنیک‌هایی همانند چرخاندن، کشیدن و محو کردن تا حدودی مبهم شوند.
- بهتر است از فونت‌های مختلف استفاده شود.
- بهتر است از یک خط عبوری روی حروف استفاده شود و این خط بهتر است بزرگ باشد.

سرویس‌های امنیتی مربوطه: احراز هویت (Authentication)

اولویت‌بندی: ۲

مسئولیت‌ها: این نیازمندی باید در سرور LDAP و توسط ادمین آن پیکربندی شود.

توضیحات تکمیلی: مکانیزم کپچای قوی برای پیشگیری از حملاتی همانند Brute Force جهت بدست آوردن حساب کاربری معتبر، مفید است.

REQ-05: استفاده از پروتکل TLS برای ارتباطات

توصیف نیازمندی: ارتباط بین سرویس Keystone و سامانه LDAP، ارتباط بین Keystone و وب‌اپلیکیشن حافظ، ارتباط بین Keystone و Swift Proxy، ارتباط بین Keystone و Horizon، ارتباط بین FAM و Swift Proxy، ارتباط بین حافظ و FAM، ارتباط بین FAM و Multi-AV، ارتباط بین حافظ و هیولای محیط مطمئن، ارتباطات سرویس Neutron، ارتباطات سرویس Nova، ارتباطات سرویس Glance، سرویس Placement، ارتباطات با پایگاه‌داده‌ها، ارتباطات RabbitMQ، ارتباط بین کلاینت پروتکل‌های دسترسی از راه دور (VNC، SPIC و RDP) و سرور پراکسی مربوطه، ارتباط بین سرور پراکسی پروتکل‌های دسترسی از راه دور و instanceها، ارتباطات سیستم ELK و ارتباطات سیستم Kafka باید مبتنی بر پروتکل TLS باشد.

معیارهای پذیرش (Acceptance Criteria):

- فقط از نسخه 1.2 یا 1.3 این پروتکل استفاده شود.
- از الگوریتم‌های ضعیف همانند DES، 3DES و RC4 استفاده نشود.
- از مجموعه رمزهای شامل مُد عملیاتی[۱] GCM استفاده شود.
- در صورت معتبر نبودن گواهی‌نامه سرور، ارتباط برقرار نشود.

سرویس‌های امنیتی مربوطه: احراز هویت (Authentication) و رمزنگاری (Encryption)

[۱] Block Cipher Mode

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید تحت نظارت تیم توسعه و توسط عملیات پیاده‌سازی و پیکربندی شود.

توضیحات تکمیلی: استفاده از پروتکل TLS برای پیشگیری از افشا و دستکاری داده‌ها هنگام انتقال آنها و همچنین احرازهویت سرور در این ارتباط، ضروری است.

REQ-06: تنظیم سرآیند امنیتی HSTS در سرآیند پاسخ‌های بازگشتی سرویس Horizon

توصیف نیازمندی: در سرآیند پاسخ‌های بازگشتی سرویس Horizon باید سرآیند امنیتی HTTP Strict Transport Security تنظیم شود.

معیارهای پذیرش (Acceptance Criteria):

- مقدار در نظرگفته شده برای دایرکتیو max-age بهتر است در ابتدا برای تست، به مقدار ۱ روز و بعد به مقدار ۱ سال پیکربندی شود.
- بهتر است دایرکتیو includeSubDomains و preload هم در این سرآیند تنظیم شود.

سرویس‌های امنیتی مربوطه: رمزنگاری (Encryption)

اولویت‌بندی: ۲

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه و در پیکربندی Horizon انجام شود.

توضیحات تکمیلی: سرآیند HSTS از ایجاد ارتباطات ناامن، پس از ایجاد یک ارتباط امن در مرورگرها، پیشگیری می‌کند.

REQ-07: در نظر گرفتن پارامترهای روز و زمان در احرازهویت و دسترسی کاربران به سیستم

توصیف نیازمندی: بهتر است احرازهویت و دسترسی کاربران به سیستم، برمبنای پارامترهای روز و زمان محدود و کنترل شود.

معیارهای پذیرش (Acceptance Criteria):

- این نیازمندی برای نقش‌های کاربر مدیر و مدیر ارشد سادا می‌تواند الزامی باشد.
- این نیازمندی برای نقش کاربر بهره‌بردار می‌تواند به صورت یک گزینه قابل فعالسازی باشد.
- روزها و ساعت‌های معتبر باید قابل تنظیم باشند.

سرویس‌های امنیتی مربوطه: احراز هویت (Authentication)

اولویت‌بندی: ۳

مسئولیت‌ها: این نیازمندی ابتدا باید در سرور سامانه LDAP با تنظیم پالیسی Logon time، پیکربندی شود. سپس پیکربندی‌های موردنیاز توسط تیم توسعه در سرویس Keystone نیز در نظر گرفته شوند.

توضیحات تکمیلی: در نظر گرفتن پارامترهای بیشتر در فرایند احرازهویت و کنترل دسترسی کاربران، امکان سوءاستفاده از سیستم را کاهش می‌دهد. به عنوان مثال، در صورتی که کاربران فقط بتوانند در طول روزهای کاری و ساعت تعیین شده به سیستم وارد شوند و از آن استفاده کنند، در صورت دسترسی مهاجم به حساب‌های کاربری معتبر، حداقل در روزهای غیرکاری و یا ساعت‌های خارج از محدودیت تعیین شده، امکان استفاده از سیستم را نخواهد داشت. این موضوع در شناسایی رفتارهای مخرب هم مفید خواهد بود.

REQ-08: در نظر گرفتن پارامتر آدرس IP در احرازهویت و دسترسی کاربران

توصیف نیازمندی: بهتر است احرازهویت و دسترسی کاربران برمبنای آدرس IP آنها محدود و کنترل شود.

معیارهای پذیرش (Acceptance Criteria):

- این نیازمندی برای نقش‌های کاربر مدیر و مدیر ارشد سادا می‌تواند الزامی باشد.
- این نیازمندی برای نقش کاربر بهره‌بردار می‌تواند به صورت یک گزینه قابل فعالسازی باشد.
- Range آدرس‌های IP معتبر باید مشخص شوند و فقط از این آدرسها امکان دسترسی به سیستم وجود داشته باشد.

سرویس‌های امنیتی مربوطه: احراز هویت (Authentication)

اولویت‌بندی: ۳

مسئولیت‌ها: این نیازمندی ابتدا باید سمت سرور LDAP و توسط مدیر آن پیکربندی شود. سپس پیکربندی‌های موردنیاز توسط تیم توسعه در سرویس Keystone نیز در نظر گرفته شوند.

توضیحات تکمیلی: احرازهویت و کنترل دسترسی کاربران برمبنای آدرس IP آنها، امکان استفاده از سیستم برای آدرس‌های غیرمعتبر وجود نخواهد داشت. یعنی حتی اگر مهاجم به اطلاعات حساب کاربری معتبر دسترسی داشته باشد، از یک آدرس غیرمعتبر امکان دسترسی نخواهد داشت.

REQ-09: استفاده از مکانیزم احرازهویت Keystone برای سرویس‌های OpenStack

توصیف نیازمندی: احرازهویت تمام سرویس‌های OpenStack شامل سرویس Neutron، سرویس Swift، سرویس Glance و سرویس Nova، باید از طریق Keystone انجام شود.

معیارهای پذیرش (Acceptance Criteria):

- مکانیزم احرازهویت هیچ یک از سرویس‌های OpenStack نباید برمبنای استراتژی noauth باشد.
- احرازهویت این سرویس‌ها باید برمبنای service accountهای آنها در Keystone انجام شود.

سرویس‌های امنیتی مربوطه: احراز هویت (Authentication)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه و در پیکربندی هر یک از سرویس‌ها انجام شود.

توضیحات تکمیلی: استفاده از مکانیزم احرازهویت Keystone این اطمینان را فراهم می‌کند که امکان استفاده از این سرویس‌ها، بدون احرازهویت وجود ندارد.

REQ-10: غیرفعال کردن توکن ادمین در Keystone

توصیف نیازمندی: توکن ادمین در پیکربندی سرویس Keystone باید غیرفعال شود.

معیارهای پذیرش (Acceptance Criteria):

- این توکن باید در فایل پیکربندی Keystone غیرفعال شود.
- Middleware مربوط به این توکن نیز باید حذف شود.

سرویس‌های امنیتی مربوطه: احراز هویت (Authentication) و مجازشماری (Authorization)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه و در پیکربندی سرویس Keystone انجام شود.

توضیحات تکمیلی: توکن ادمین که در سرویس Keystone برای بالا آوردن این سرویس استفاده می‌شود، دارایی با ارزشی است که مجوزهای دسترسی ممتازی دارد. بنابراین بهتر است غیرفعال شود.

REQ-11: اجرای سرویس Swift با دسترسی غیر root

توصیف نیازمندی: سرویس Swift باید با مجوز حساب کاربری غیر root (non-root) در سیستم عامل مربوطه اجرا شود.

معیارهای پذیرش (Acceptance Criteria):

- سرویس Swift با دسترسی نام کاربری swift و گروه swift در سیستم عامل بهتر است اجرا شود.

سرویس‌های امنیتی مربوطه: احراز هویت (Authentication) و مجازشماری (Authorization)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه و در پیکربندی سرویس Swift انجام شود.

توضیحات تکمیلی: با توجه به اینکه حساب کاربری root مجوزهای دسترسی ممتاز را در سیستم عامل دارد، بهتر است برای اجرای سرویس Swift از این حساب کاربری استفاده نشود.

REQ-12: تنظیم درست مالکیت و مجوزهای دسترسی فایل‌های پیکربندی

توصیف نیازمندی: مالکیت و مجوزهای دسترسی فایل‌های پیکربندی سرویس‌های Keystone، Neutron، Nova، Swift، Glance و Horizon باید به درستی تنظیم شوند.

معیارهای پذیرش (Acceptance Criteria):

- مالکیت فایلهای پیکربندی Keystone و همچنین دایرکتوری دربرگیرنده آنها، باید به کاربر و گروه keystone تنظیم شود.
- مالکیت فایلهای پیکربندی Neutron و همچنین دایرکتوری دربرگیرنده آنها، باید به کاربر root و گروه neutron تنظیم شود.
- مالکیت فایلهای پیکربندی Nova و همچنین دایرکتوری دربرگیرنده آنها، باید به کاربر root و گروه nova تنظیم شود.
- مالکیت فایل پیکربندی Horizon باید به کاربر root و گروه horizon تنظیم شود.
- مالکیت فایلهای پیکربندی Swift و همچنین دایرکتوری دربرگیرنده آنها، باید به کاربر root و گروه swift تنظیم شود.
- مالکیت فایل پیکربندی FAM باید به کاربر و گروه fam تنظیم شود
- مجوزهای دسترسی فایل‌های پیکربندی Keystone، Neutron، Nova، Glance و Swift حداکثر به 640 یا مجوز کمتر و مجوز دسترسی دایرکتوری دربرگیرنده آن به 750 باید تنظیم شود.
- مجوزهای دسترسی فایل پیکربندی Horizon حداکثر به 640 یا مجوز کمتر باید تنظیم شود.
- مجوزهای دسترسی فایل پیکربندی FAM حداکثر به 640 یا مجوز کمتر تنظیم شود.
- مالکیت فایل‌های مربوط به certificateها و کلیدهای TLS باید به کاربر daemon مربوطه (مثلا کاربر daemon سیستم مدیریت پایگاه داده‌ها و کاربر daemon سرور RabbitMQ) تنظیم شده باشد و مجوزهای دسترسی آنها نیز به مجوز 600 تنظیم شده باشند.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization) و کنترل دسترسی (Access Control)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه و در پیکربندی سرویسها انجام شود.

توضیحات تکمیلی: فایل‌های پیکربندی سرویس‌های OpenStack حاوی پارامترها و اطلاعات مهمی جهت عملکرد درست این سرویس‌ها هستند. بنابراین در صورتی که این فایل‌ها و پارامترها و اطلاعات موجود در آنها، توسط یک کاربر غیرمجاز، به صورت عمدی یا تصادفی تغییر یابند یا حذف شوند، در نتیجه آن، امکان از دسترس خارج شدن آن سرویس و در نتیجه کل سرویس وجود خواهد داشت. بنابراین پیاده‌سازی این نیازمندی اهمیت بسیاری برای سیستم دارد.

REQ-13: تنظیم درست مالکیت و مجوزهای دسترسی دایرکتوری key repository مورداستفاده برای توکن‌های Fernet

توصیف نیازمندی: مالکیت و مجوزهای دسترسی دایرکتوری key repository باید به درستی تنظیم شوند.

معیارهای پذیرش (Acceptance Criteria):

- فقط سرویس Keystone باید بتواند از key repository بخواند و در آن بنویسد؛ بنابراین مالکیت دایرکتوری key repository که fernet-keys نام دارد، باید به کاربر و گروه keystone تنظیم شود.
- مجوز دسترسی دایرکتوری key repository به 750 باید تنظیم شود.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization) و کنترل دسترسی (Access Control)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه و در پیکربندی Keystone انجام شود.

توضیحات تکمیلی: سرویس Keystone از دایرکتوری key repository برای ایجاد توکن‌های Fernet استفاده می‌کند. کلیدهای موجود در این دایرکتوری، برای رمزنگاری و رمزگشایی اطلاعاتی استفاده می‌شوند که payload توکن‌ها را تشکیل می‌دهند. بنابراین پیاده‌سازی این الزام، از هر نوع دسترسی غیرمجاز به این کلیدها که امکان سوءاستفاده از آنها برای تولید توکن‌های معتبر را در پی داشته باشد، پیشگیری خواهد کرد.

REQ-14: تنظیم درست مالکیت و مجوزهای دسترسی دایرکتوری /var/lib/nova شامل اطلاعات instanceها

توصیف نیازمندی: مالکیت و مجوزهای دسترسی دایرکتوری /var/lib/nova باید به درستی تنظیم شوند.

معیارهای پذیرش (Acceptance Criteria):

- مالکیت این دایرکتوری باید به گروه و کاربر nova تنظیم شود.
- کاربر nova باید دسترسی خواندن و نوشتن در دایرکتوری /var/lib/nova/instances داشته باشد.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization) و کنترل دسترسی (Access Control)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه و در پیکربندی‌های مربوط به سرویس Nova انجام شود.

توضیحات تکمیلی: دایرکتوری /var/lib/nova شامل جزئیات اطلاعات instanceهای در حال اجرا روی یک هاست است که باید محافظت شود.

REQ-15: بکارگیری ابزارهای FIM برای پایش مداوم دسترسی‌ها و تغییرات فایل‌ها و دایرکتوری‌های حساس

توصیف نیازمندی: بهتر است دسترسی به فایل‌های پیکربندی و دایرکتوری‌های حساس، به کمک یک ابزار FIM[۱] به صورت مداوم پایش و تغییرات آنها رصد شود.

معیارهای پذیرش (Acceptance Criteria):

- این نیازمندی برای تمام فایل‌های پیکربندی سرویس‌های OpenStack و مولفه FAM، فایل‌های مربوط به certificateها و کلیدهای TLS، دایرکتوری key repository و دایرکتوری /var/lib/nova انجام شود.

سرویس‌های امنیتی مربوطه: کنترل دسترسی (Access Control)

اولویت‌بندی: ۲

مسئولیت‌ها: این الزام باید توسط تیم عملیات و تحت نظارت تیم توسعه پیاده‌سازی شود. درواقع تیم توسعه باید فهرست دایرکتوری‌ها و فایل‌های حساس را مشخص کنند و تیم عملیات در ابزار FIM درج کنند.

توضیحات تکمیلی: ابزارهایی همانند OSSEC، Wazuh و Samhain نمونه ابزارهای FIM هستند که به کمک آنها می‌توان هر نوع تغییر غیرمجاز در فایل‌ها و دایرکتوری‌های حساس را شناسایی کرد.

[۱] File Integrity Monitoring

REQ-16: پیکربندی چارچوب کنترل دسترسی MAC برای حفاظت از فایل‌ها و دایرکتوری‌های حساس

توصیف نیازمندی: بهتر است برای افزایش سطح کنترل دسترسی به فایل‌ها و دایرکتوری‌های حساس، از چارچوب‌های کنترل دسترسی MAC[1] استفاده شود.

معیارهای پذیرش (Acceptance Criteria):

- در صورت استفاده از سیستم عامل‌های مبتنی RedHat برای سرور، باید قابلیت SELinux و پالیسی‌های امنیتی موردنیاز در آن، پیکربندی شوند.
- در صورت استفاده از سیستم عامل‌های مبتنی Debian برای سرور، باید قابلیت AppArmor و پروفایل‌های مودنیاز در آن، پیکربندی شوند.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization) و کنترل دسترسی (Access Control)

اولویت‌بندی: ۳

مسئولیت‌ها: این الزام باید توسط تیم عملیات و تحت نظارت تیم توسعه پیاده‌سازی شود.

توضیحات تکمیلی: در چارچوب‌های MAC، کنترل دسترسی به objectها (یعنی همان منابع)، برمبنای برچسب‌هایی انجام می‌شود که به این objectها و subjectها (همانند پراسس‌ها یا کاربران) اختصاص داده می‌شود. درواقع این برچسب‌ها، میزان حساسیت اطلاعاتی را که objectها دربرمی‌گیرند، مشخص می‌کنند. هر subject تنها می‌تواند به objectهایی دست یابد که برچسب حساسیت آنها کمتر از برچسب اعطا شده به کاربر باشد. این موضوع به کمک پالیسی‌ها یا پروفایل‌ها مشخص و توسط سیستم عامل کنترل می‌شود. بنابراین به کمک چارچوب کنترل دسترسی MAC می‌توان از فعالیت‌های مخرب کاربران سیستم عامل، حتی کاربر root، پیشگیری کرد. درواقع اگر فعالیت‌های کاربران منطبق بر پالیسی‌ها یا پروفایل‌های مشخص شده نباشد، اجازه انجام آن حتی برای کاربر root هم فراهم نخواهد بود.

REQ-17: رمزنگاری secretهای موجود در فایل‌های پیکربندی

توصیف نیازمندی: بهتر است credentialها یا همان secretهای موجود در فایل‌های پیکربندی مولفه‌های مختلف سیستم به صورت رمزشده در این فایل‌ها ذخیره شوند.

معیارهای پذیرش (Acceptance Criteria):

[1] Mandatory Access Control

- کلمه عبور پایگاه داده‌های سرویس Keystone، Nova، Glance، Placement و FAM باید به صورت رمزشده در فایل‌های پیکربندی آنها ذخیره شود.
- کلمه عبور service accountهای اختصاص داده شده به سرویس‌های Neutron، Nova، Placement، Glance، Swift و FAM در Keystone باید به صورت رمزشده در فایل‌های پیکربندی مربوطه ذخیره شود.
- کلمه عبور اختصاص داده شده به حساب کاربری Keystone در سیستم LDAP، باید به صورت رمزشده در فایل پیکربندی Keystone ذخیره شود.
- کلید اصلی رمزنگاری objectها (تحت عنوان encryption_root_secret) باید به صورت رمزشده در فایل پیکربندی میان‌افزار keymaster قرار گیرد.
- کلمه عبور اختصاص داده شده به حساب کاربری openstack در RabbitMQ باید به صورت رمزشده در فایل پیکربندی سرویس‌های Neutron و Nova ذخیره شود.
- کلمه عبور metadata proxy مربوط به سرویس Nova به صورت رمزشده در فایل پیکربندی مربوطه ذخیره شود.
- کلمه عبور کاربر بهره‌بردار در فایل پیکربندی Rclone بهتر است به صورت رمزشده در این فایل ذخیره شود.
- کلمه عبور اختصاص داده شده به حساب کاربری مولفه FAM در Mutli-AV، باید به صورت رمزشده در فایل پیکربندی این مولفه ذخیره شود.
- رمزنگاری secretها باید به کمک یک الگوریتم رمزنگاری قوی همانند AES و ChaCha20 با تولید کلید ۲۵۶ بیت انجام شود و از الگوریتم‌های ضعیفی همانند DES و RC4 استفاده نشود.
- مد عملیاتی الگوریتم رمزنگاری نیز باید امن باشد. از مد CBC، CTR و یا GCM استفاده شود و از مد ECB استفاده نشود.
- کلیدهای مورداستفاده برای رمزنگاری secretها خود باید به کمک الگوریتم‌های قوی همانند PBKDF2، Bcrypt، Scrypt و Argon2 تولید شوند. استفاده از SHA-256 بدون استفاده از salt برای تولید کلید رمزنگاری به هیچ وجه توصیه نمی‌شود.

سرویس‌های امنیتی مربوطه: رمزنگاری (Encryption)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی ابتدا باید توسط تیم توسعه و سپس تیم عملیات سادا پیاده‌سازی شود. نکته ای که وجود دارد آن است که تیم توسعه نباید از secretهای محیط عملیاتی مطلع و به آنها دسترسی داشته باشند. درواقع درج secretهای رمزشده در فایلها توسط تیم عملیات باید انجام گیرد.

توضیحات تکمیلی: رمزنگاری credentialهای موجود در فایل‌های پیکربندی از افشای این اطلاعات و سوءاستفاده‌های بعدی از آنها، در صورت دسترسی مهاجم به این فایل‌ها، جلوگیری خواهد کرد.

REQ-18: ذخیره‌سازی و مدیریت secretها در سیستم متمرکز مدیریت secretها

توصیف نیازمندی: بهتر است secretهایی که در نیازمندی قبلی به آنها اشاره شد، در یک سیستم مدیریت متمرکز secretها ذخیره شوند.

معیارهای پذیرش (Acceptance Criteria):

- تمام کلمات عبور و کلیدهایی که در نیازمندی قبلی به آنها اشاره شد، بهتر است در سیستم مدیریت متمرکز secretها ذخیره شوند.

سرویس‌های امنیتی مربوطه: رمزنگاری (Encryption)

اولویت‌بندی: ۲

مسئولیت‌ها: این نیازمندی باید توسط عملیات سادا و تحت نظارت تیم توسعه انجام گیرد.

توضیحات تکمیلی: در صورت ذخیره‌سازی credentialها در یک سیستم متمرکز مدیریتی، علاوه بر اینکه نیاز به ذخیره‌سازی این اطلاعات حساس را در فایل‌های پیکربندی از بین می‌برد، بلکه امکان رمزنگاری آنها و همچنین کنترل دسترسی به آنها را نیز فراهم می‌کند. ضمنا امکان تعریف پالیسی‌های مختلف برای مدیریت credentialها را نیز فراهم می‌کند. به عنوان مثال، اینکه این credentialها چه طولی داشته باشند، از هر چند وقت یکبار تولید مجدد شوند و چه زمان منقضی شوند، در این سیستم قابل تعریف خواهد بود.

در حال حاضر، ابزارهای مختلفی اعم از Hashicorp Vault، KMIP، PSONO و دیگر ابزارها برای مدیریت متمرکز کلیدها و کلمات عبور وجود دارند. اما OpenStack خود یک سرویس تحت عنوان Barbican دارد که امکان مدیریت متمرکز secretها را برای سایر سرویس‌های OpenStack فراهم می‌کند. استفاده از سیستم Barbican مزایایی همانند کنترل دسترسی به secretها برمبنای پالیسی‌های تعریف شده، اعمال quota، امکان تنظیم لیست ACL به ازای هر secret، گروه‌بندی secretها و ردیابی استفاده‌کنندگان از secretها را فراهم می‌کند. البته ناگفته نماند که این سرویس می‌تواند با سیستم‌های خارجی همانند Hashicorp Vault، KMIP و دیگر سیستم‌هایی که پلاگین آنها در OpenStack موجود است، نیز ادغام شود. برای این منظور کافی است که در backend، APIهای ارائه شده این سیستم‌ها را فراخوانی کند. البته امکان ذخیره‌سازی secretها در

پایگاه‌داده خود سرویس Barbican نیز وجود دارد، اما استفاده از سیستم‌های خارجی امکان ذخیره‌سازی secretها را در یک سیستم کاملا منفک فراهم می‌کند.

REQ-19: تنظیم حداقل مجوزهای دسترسی برای حساب کاربری تعریف شده برای Keystone در سامانه LDAP

توصیف نیازمندی: حساب کاربری تعریف شده برای Keystone در سیستم LDAP باید مجوزهای دسترسی محدودی در این سیستم داشته باشد.

معیارهای پذیرش (Acceptance Criteria):

- این حساب کاربری هرگز نباید امکان ایجاد تغییر در سیستم LDAP را داشته باشد.

سرویس‌های امنیتی مربوطه: کنترل دسترسی (Access Control)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید سمت سامانه LDAP توسط ادمین آن پیکربندی شود.

توضیحات تکمیلی: در نظر گرفتن حداقل دسترسی‌ها برای حساب کاربری Keystone، از سوءاستفاده از این حساب کاربری برای افزایش سطح دسترسی و دسترسی غیرمجاز به منابع و اطلاعات دیگر، پیشگیری خواهد کرد.

REQ-20: پیکربندی درست پارامترهای مکانیزم Key rotation، برای کلیدهای مورداستفاده در تولید توکن‌های Fernet

توصیف نیازمندی: پارامترهای مورداستفاده در مکانیزم key rotation (چرخش کلیدها) برای کلیدهای مورداستفاده در تولید توکن‌های Fernet، باید به درستی پیکربندی شوند.

معیارهای پذیرش (Acceptance Criteria):

- پارامترهای مربوط به حداکثر مدت زمان اعتبار توکن‌ها، حداکثر مدت زمان چرخش کلیدها و حداکثر تعداد کلیدهای فعال باید طوری پیکربندی شوند که کلیدهای موردنیاز برای رمزگشایی توکن‌های معتبر، در هر لحظه در key repository موجود باشند. در عین حال، فقط کلیدهای موردنیاز، در این دایرکتوری وجود داشته باشند.

سرویس‌های امنیتی مربوطه: در دسترس‌پذیری (Availability)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه و در پیکربندی Keystone انجام شود.

توضیحات تکمیلی: مکانیزم key rotation برای تنزل کلیدهای primary مورداستفاده برای رمزنگاری توکن‌های Fernet به کلیدهای secondary و تولید کلیدهای جدید انجام می‌شود. با اجرای این مکانیزم، اگر مهاجم توانسته باشد به کلید primary دست یابد، هرچند امکان رمزگشایی توکن‌های قبلی برایش وجود دارد، اما نمی‌تواند با استفاده از آن، توکن جدید و معتبر تولید کند. از طرف دیگر، پارامترهای این مکانیزم باید طوری پیکربندی شوند که هیچ توکن معتبری که غیرقابل رمزگشایی باشد، وجود نداشته باشد.

REQ-21: محدود کردن حداکثر اندازه body در درخواست‌های ارسالی به Keystone

توصیف نیازمندی: حداکثر اندازه body درخواست‌های ارسالی به سرویس Keystone باید محدود شود.

معیارهای پذیرش (Acceptance Criteria):

- این پارامتر یا باید به مقدار پیش‌فرض آن که ۱۱۴۶۸۸ بایت است تنظیم شود و یا یک مقدار مناسب دیگر

سرویس‌های امنیتی مربوطه: در دسترس‌پذیری (Availability)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه در پیکربندی سرویس Keystone انجام شود، اما مقدار آن می‌تواند برمبنای نظر کارفرما تنظیم شود.

توضیحات تکمیلی: در صورتی که حداکثر اندازه body درخواست‌های ارسالی به Keystone مشخص نشده باشد، امکان ارسال درخواست‌های با حجم بالا و در نتیجه از دسترس خارج شدن سرویس وجود خواهد داشت.

REQ-22: بررسی صحت و اصالت imageها قبل از بوت شدن آنها

توصیف نیازمندی:image های مورداستفاده برای ایجاد instanceها بهتر است قبل از بوت شدن، ابتدا مورد صحت‌سنجی و اصالت‌سنجی قرار گیرند.

معیارهای پذیرش (Acceptance Criteria):

- برای بررسی صحت و اصالت imageها قبل از بوت شدن آنها، باید از قابلیت image signature verification سرویس Glance استفاده شود.
- پارامترهایی که استفاده از این قابلیت را در سرویس‌های دیگر همانند Nova فراهم می‌کنند، باید در فایل پیکربندی آنها فعال شوند.

سرویس‌های امنیتی مربوطه: رمزنگاری (Encryption)

اولویت‌بندی: ۲

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه در پیکربندی سرویس‌های Glance و Nova انجام شود.

توضیحات تکمیلی: در حال حاضر OpenStack با تولید یک checksum از نوع MD5 برای imageهای آپلود شده، امکان تشخیص imageهای دستکاری شده را هنگام بوت شدن آنها دارد. اما در صورتی که دفعات بعدی از همان image استفاده شود، این بررسی دیگر انجام نخواهد شد. بنابراین در صورت دستکاری imageها، کاربر از آن مطلع نخواهد شد. ضمن اینکه این روش، تنها دستکاری imageها را تشخیص می‌دهد و نمی‌تواند اصالت آن را تایید کند. سرویس Glance با فراهم کردن قابلیت image signature verification برای سایر سرویس‌ها، امکان بررسی صحت و اصالت imageها را فراهم می‌کند.

REQ-23: عدم افشای اطلاعات اضافی در پاسخ‌های بازگشتی Keystone

توصیف نیازمندی: در پاسخ‌های بازگشتی سرویس Keystone نباید اطلاعات اضافی وجود داشته باشد.

معیارهای پذیرش (Acceptance Criteria):

- مقدار پارامتر insecure_debug در فایل پیکربندی Keystone باید به مقدار false تنظیم شود.

سرویس‌های امنیتی مربوطه: مدیریت خطاها (Error Handling)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه و در پیکربندی سرویس Keystone انجام شود.

توضیحات تکمیلی: در صورتی که مقدار پارامتر insecure_debug در فایل پیکربندی Keystone به مقدار true تنظیم شده باشد، اطلاعاتی که در پاسخ‌های بازگشتی سرویس وجود خواهد داشت، بیش از حد نرمال خواهد بود. به عنوان مثال ممکن است جزئیات بیشتری در مورد شکست در عملیات احرازهویت را برای کاربران افشا کند.

REQ-24: عدم افشای اطلاعات محل ذخیره‌سازی imageها

توصیف نیازمندی: اطلاعات محل ذخیره‌سازی imageها در سرور نباید در اطلاعات بازگشتی سرویس Glance برای کاربران افشا شود.

معیارهای پذیرش (Acceptance Criteria):

- مقدار پارامتر show_image_direct_url در پیکربندی Glance باید به مقدار false تنظیم شود.

سرویس‌های امنیتی مربوطه: مدیریت خطاها (Error Handling)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه و در پیکربندی سرویس Glance انجام شود.

توضیحات تکمیلی: در صورتی که پارامتر show_image_direct_url به مقدار true تنظیم شود، محل ذخیره‌سازی imageها در پاسخ بازگشتی سرویس Glance دیده می‌شود که ممکن است توسط مهاجمین مورد سوءاستفاده قرار گیرد.

REQ-25: در نظر گرفتن حداقل دسترسی برای حساب کاربری تعریف شده برای سرویس‌ها در پایگاه‌داده‌های مربوطه

توصیف نیازمندی: حساب کاربری در نظر گرفته شده برای سرویس‌ها باید حداقل دسترسی‌های موردنیاز در سیستم مدیریت پایگاه داده‌هایشان را داشته باشد.

معیارهای پذیرش (Acceptance Criteria):

- دسترسی در نظر گرفته شده برای حساب کاربری سرویس‌ها نباید شامل سطح دسترسی کامل در سیستم مدیریت پایگاه داده باشد.
- سرویس‌ها باید فقط به پایگاه داده خود دسترسی داشته باشند.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات سادا و تحت نظارت تیم توسعه انجام شود.

توضیحات تکمیلی: در نظر گرفتن حداقل دسترسی برای سرویس‌ها در سیستم مدیریت پایگاه داده‌هایشان، از دستکاری عمدی یا تصادفی در داده‌های موجود در پایگاه داده‌ها و همچنین ایجاد پیکربندی نادرست در آنها پیشگیری می‌کند.

REQ-26: فعال کردن audit logging پایگاه داده‌های مورداستفاده در مولفه‌های مختلف

توصیف نیازمندی: قابلیت audit logging در سیستم مدیریت پایگاه داده‌های سرویس‌های OpenStack، مولفه‌های FAM و Multi-AV و سیستم ELK باید فعال شود.

معیارهای پذیرش (Acceptance Criteria):

- پلاگین‌ها و extensionهای مربوط به audit logging در سیستم مدیریت پایگاه داده‌های سرویس‌ها (MySQL) نصب و پیکربندی شده باشد.
- قابلیت audit logging سیستم ELK در پیکربندی آن باید فعال شود.
- در سیستم پایگاه داده مولفه‌های FAM و Multi-AV نیز بهتر است قابلیت audit logging فعال شود.

سرویس‌های امنیتی مربوطه: رویدادنگاری (Logging)

اولویت‌بندی: ۲

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات سادا و تحت نظارت تیم توسعه انجام شود.

توضیحات تکمیلی: فعال بودن قابلیت audit logging سیستم مدیریت پایگاه داده‌ها، امکان ردیابی فعالیت‌های کاربران پایگاه داده‌ها اعم از به‌روزرسانی جداول، اجرای کوئری‌ها، اعطای دسترسی‌ها و غیره، برای تشخیص فعالیت‌های مخرب احتمالی آنها فراهم می‌کند.

REQ-27: استفاده از مکانیزم TLS Client Authentication برای سرویس‌ها

توصیف نیازمندی: بهتر است احرازهویت و مجازشماری سرویس‌های OpenStack در Keystone، در سیستم مدیریت پایگاه داده‌هایشان و همچنین سرویس RabbitMQ، با استفاده از مکانیزم TLS client authentication انجام شود.

معیارهای پذیرش (Acceptance Criteria):

- از گواهی‌های X.509 در کلاینت‌ها استفاده شود.

سرویس‌های امنیتی مربوطه: احراز هویت (Authentication) و رمزنگاری (Encryption)

اولویت‌بندی: ۳

مسئولیت‌ها: این نیازمندی توسط تیم توسعه انجام شود.

توضیحات تکمیلی: احرازهویت سرویس‌ها برمبنای مکانیزم TLS client authentication، یک فاکتور در دیگر علاوه بر نام کاربری و کلمه عبور، در فرایند احرازهویت فراهم می‌کند و ریسک دسترسی غیرمجاز را در صورت لو رفتن نام کاربری و کلمه عبور سرویس‌ها در Keystone، پایگاه داده‌هایشان و RabbtiMQ، کاهش می‌دهد.

REQ-28: بررسی صحت و اصالت فایل‌های کاربران قبل از انتقال به هیولای محیط مطمئن

توصیف نیازمندی: صحت و اصالت محتواها یا همان فایل‌های کاربران قبل از انتقال به هیولای محیط مطمئن، باید مورد بررسی قرار گیرد.

معیارهای پذیرش (Acceptance Criteria):

- برای بررسی صحت و اصالت محتواها، بهتر است سمت مبدا آن یعنی محیط نا مطمئن، یک امضای دیجیتال برای فایل‌ها تولید و همراه فایل‌ها به مقصد یعنی محیط مطمئن ارسال شود.
- سمت محیط مطمئن قبل از انتقال فایل‌ها به هیولا، امضای دیجیتال فایل‌ها باید مورد بررسی قرار گیرد و فایل‌هایی که صحت آنها مخدوش شده است، به هیولا منتقل نشوند و یک لاگ هم برای آن ثبت شود.
- برای تولید امضای دیجیتال از الگوریتم‌های قوی همانند DSA و ECDSA و EdDSA می‌توان استفاده کرد.
- کلیدهای خصوصی و عمومی مورد استفاده برای تولید امضای دیجیتال هم باید به کمک الگوریتم‌های قوی همانند PBKDF2، Bcrypt، Scrypt و Argon2 تولید شوند. استفاده از SHA-256 بدون استفاده از salt برای تولید کلید رمزنگاری به هیچ وجه توصیه نمی‌شود.

سرویس‌های امنیتی مربوطه: رمزنگاری (Encryption) و احرازهویت (Authentication)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه سادا پیاده‌سازی شود.

توضیحات تکمیلی: استفاده از امضای دیجیتال، امکان بررسی صحت، اصالت محتواها و همچنین ویژگی انکارناپذیری را برای آنها فراهم می‌کند. برای این منظور، با استفاده از یک کلید خصوصی در محیط نامطمئن، امضای دیجیتال، تولید و سمت محیط مطمئن، بررسی می‌شود.

REQ-29: عدم نمایش فایل‌های با تگ آلوده در فهرست فایل‌های قابل تایید یا رد مدیر سادا

توصیف نیازمندی: فایل‌هایی که توسط Multi-AV به عنوان فایل‌های آلوده شناسایی شده‌اند، نباید در فهرست فایل‌های قابل تایید یا رد توسط مدیر سادا در سامانه حافظ برای انتقال به محیط مطمئن، نمایش داده شوند.

معیارهای پذیرش (Acceptance Criteria):

- در فهرست نمایش داده شده به کاربر مدیر سادا در ابزار انتقال فایل در سامانه حافظ، نباید فایل‌هایی که تگ آلوده دارند، دیده شوند.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه سادا پیاده‌سازی شود.

توضیحات تکمیلی: عدم نمایش فایل‌های با تگ آلوده در فهرست مدیر سادا، می‌تواند از امکان تایید انتقال آنها به محیط مطمئن، به صورت عمدی یا سهوی ممانعت کند.

REQ-30: عدم امکان تایید درخواست انتقال فایل‌های با تگ آلوده به محیط مطمئن توسط مدیر سادا

توصیف نیازمندی: امکان انتقال فایل‌هایی که توسط Multi-AV به عنوان فایل‌های آلوده شناسایی شده‌اند، نباید برای مدیر سادا قابل تایید باشد.

معیارهای پذیرش (Acceptance Criteria):

- حتی اگر مدیر سادا به صورت عمدی یا سهوی، دکمه تایید انتقال فایل‌های آلوده را در ابزار ممیزی انتقال فایل در سامانه حافظ بزند، سیستم باید مانع از انتقال آنها شود و یک لاگ هم برای آن ثبت کند.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه سادا پیاده‌سازی شود.

توضیحات تکمیلی: در صورتی که سیستم مانع از انتقال فایل‌های با تگ آلوده به محیط مطمئن شود، هر نوع خطای انسانی در تایید درخواست انتقال فایل‌های آلوده، پوشش داده خواهد شد.

REQ-31: به‌روزرسانی منظم الگوهای تشخیص بدافزار در Multi-AV

توصیف نیازمندی: الگوهای تشخیص بدافزار درMulti-AV باید همواره به‌روز باشند.

معیارهای پذیرش (Acceptance Criteria):

- سیستم Multi-AV بهتر است قابلیت به‌روزرسانی منظم و خودکار الگوهای تشخیص بدافزار را فراهم کرده باشد.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۱

مسئولیت‌ها: این قابلیت ابتدا باید این قابلیت در مولفه Multi-AV توسط تیم پیمانکار آن پیاده‌سازی شده باشد. سپس تیم توسعه و عملیات هم آن را به درستی پیکربندی نمایند.

توضیحات تکمیلی: به‌روزرسانی منظم الگوهای تشخیص بدافزار درMulti-AV، احتمال تشخیص فایل‌های آلوده جدید را فراهم خواهد کرد.

REQ-32: مجاز کردن پورت‌های موردنیاز پروتکل‌های دسترسی از راه دور به instance، در فایروال نود

توصیف نیازمندی: پورت‌های موردنیاز پروتکل‌های دسترسی از راه دور به instance باید در قالب قوانینی به فایروال نود مربوطه اضافه شوند.

معیارهای پذیرش (Acceptance Criteria):

- برای دسترسی از راه دور از طریق پروتکل VNC، پورت‌های ۶۰۸۰، ۶۰۸۱، ۶۰۸۲، ۵۹۰۰ و ۵۹۱۰ باید به فایروال اضافه شود.
- برای دسترسی از راه دور از طریق پروتکل SPICE، پورت ۶۰۸۲ باید به فایروال اضافه شود.
- برای دسترسی از راه دور از طریق پروتکل RDP، پورت ۶۰۸۳ باید به فایروال اضافه شود.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه و در پیکربندی سرویس Nova انجام شود.

توضیحات تکمیلی: مجاز کردن پورت‌های ضروری در فایروال و بستن سایر پورت‌ها، از دسترسی غیرمجاز از طریق پورت‌های غیرضروری جلوگیری کرد.

REQ-33: محدود کردن دامنه آدرس‌های IP مجاز برای دسترسی از راه دور به instanceها

توصیف نیازمندی: بهتر است دامنه آدرس‌های مجاز دسترسی از راه دور به instanceها مشخص و در قالب قوانینی به فایروال اضافه شوند.

معیارهای پذیرش (Acceptance Criteria):

- دامنه آدرس‌های IP کاربران شناسایی و در فایروال اضافه شود.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۳

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات سادا و با راهنمایی تیم توسعه انجام شود.

توضیحات تکمیلی: محدود کردن دامنه آدرس‌های IP مجاز به دسترسی از راه دور به instanceها، می‌تواند از دسترسی غیرمجاز به آنها پیشگیری کند.

REQ-34: حفاظت از اطلاعات مربوط به نتیجه اسکن فایل‌ها توسط سیستم FAM

توصیف نیازمندی: باید نتیجه اسکن فایل‌ها که در پایگاه داده FAM ذخیره می‌شوند، مورد حفاظت قرار گیرند.

معیارهای پذیرش (Acceptance Criteria):

- برای حفاظت از اطلاعات مربوط به نتیجه اسکن فایل‌ها بهتر است این اطلاعات به صورت رمزشده در پایگاه داده ذخیره شوند.
- همچنین این اطلاعات برای بقیه حساب کاربری به غیر از FAM، به صورت read-only باشند.

سرویس‌های امنیتی مربوطه: رمزنگاری (Encryption)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه سادا انجام شود.

توضیحات تکمیلی: با توجه به اینکه اطلاعات مربوط به نتیجه اسکن فایل‌ها حساس هستند و برمبنای این اطلاعات، برای انتقال یا عدم امکان انتقال فایل‌ها به محیط مطمئن، تصمیم‌گیری می‌شود، باید از دستکاری آنها پیشگیری شود.

REQ-35: ثبت رویدادهای امنیتی در لاگ‌ها

توصیف نیازمندی: رویدادهای امنیتی که در مولفه‌های مختلف سادا رخ می‌دهند، باید در لاگ‌ها ثبت شوند.

معیارهای پذیرش (Acceptance Criteria):

- انواع رویدادهای امنیتی به شرح ذیل باید در لاگ ثبت شوند:
  - موفقیت و شکست در فرایند احرازهویت
  - شکست در فرایند مجازشماری
  - شکست در اعتبارسنجی داده‌ها

- آپلود فایل‌ها
- اسکن فایل‌ها
- عبور از حدآستانه تعریف شده برای نرخ فرخوانی APIها (rate limiting)
- تغییر نقش‌ها و سطح دسترسی‌ها
- تعریف و حذف کاربران
- خواندن، تغییر و حذف فایل‌های پیکربندی
- عملکردهای حساس مانند ارائه درخواست انتقال فایل، تغذیه فایل، تایید و رد درخواست انتقال فایل
- خاموش شدن، روشن شدن و ریستارت سرویس‌ها و instanceها
- متوقف شدن تولید لاگ توسط مولفه‌ها

سرویس‌های امنیتی مربوطه: رویدادنگاری (Security Logging)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه سادا و تیم عملیات پیاده‌سازی شود.

توضیحات تکمیلی: وجود لاگ کافی از رویدادهای امنیتی و فعالیت‌های کاربران سادا در بخش‌های مختلف، امکان پایش و ممیزی این فعالیت‌ها و تشخیص رفتارهای غیرعادی را فراهم خواهد کرد.

REQ-36: ارسال لاگ تمام مولفه‌ها به سرور ELK

توصیف نیازمندی: لاگ تمام مولفه‌های سادا باید به سرور ELK ارسال شوند و در آنجا مدیریت شوند.

معیارهای پذیرش (Acceptance Criteria):

- لاگ تمام مولفه‌ها از جمله سرویس‌های OpenStack، حافظ، FAM، Multi-AV، سیستم مدیریت پایگاه‌داده‌ها و همچنین instanceها باید به ELK ارسال شود.

سرویس‌های امنیتی مربوطه: رویدادنگاری (Security Logging) و پایش (Monitoring)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات سادا و تحت نظارت تیم توسعه پیاده‌سازی شود.

توضیحات تکمیلی: با مدیریت متمرکز لاگ‌ها در سیستم ELK، امکان پایش و تحلیل این لاگ‌ها، کنترل دسترسی به آنها و حفاظت از آنها فراهم خواهد شد.

REQ-37: پایش مداوم لاگ‌های مولفه‌ها

توصیف نیازمندی: داده‌های لاگ ذخیره شده در ELK باید به طور مداوم پایش شوند.

معیارهای پذیرش (Acceptance Criteria):

- برای پایش مداوم داده‌های لاگ، لازم است معیارهای موردنیاز برمبنای این داده‌ها تعریف شده باشند.
- وضعیت مصرف منابع توسط مولفه‌ها، تعداد درخواست‌های ارسالی به مولفه‌ها، زمان لاگین به مولفه‌ها، تعداد دفعات لاگین ناموفق، حجم ترافیک نمونه‌ای از معیارها هستند.

سرویس‌های امنیتی مربوطه: پایش (Monitoring)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات و تحت نظارت تیم توسعه انجام شود.

توضیحات تکمیلی: پایش مداوم لاگ‌های مولفه‌های مختلف سادا، امکان تشخیص و پاسخگویی سریع به حوادث را فراهم می‌کنند.

REQ-38: تعریف نقش‌ها و مجوزهای دسترسی‌ها در Elasticsearch

توصیف نیازمندی: در سیستم Elasticsearch باید نقش‌های مختلفی با حداقل مجوزهای دسترسی‌های موردنیاز برای کاربران و گروه‌های آنها تعریف شوند.

معیارهای پذیرش (Acceptance Criteria):

- حساب کاربری elastic که شامل دسترسی‌های superuser است، بهتر است تا حد امکان استفاده نشود.
- به ازای کاربران و گروه‌های مختلف، بهتر است نقش‌های مناسب آنها با حداقل مجوزهای دسترسی موردنیاز در نظر گرفته شود.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization) و کنترل دسترسی (Access Control)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات سادا و تحت نظارت تیم توسعه پیاده‌سازی شود.

توضیحات تکمیلی: با توجه به اینکه سیستم ELK حاوی لاگ مولفه‌های سادا است، بنابراین باید دسترسی به آنها به درستی کنترل شود و برای این منظور باید نقش‌های و حداقل مجوزهای دسترسی مورنیاز آنها تعریف شوند.

REQ-39: فعالسازی IP Filtering در Elasticsearch

توصیف نیازمندی: بهتر است قابلیت IP Filtering در Elasticsearch فعال شده و فهرست یا دامنه آدرس‌های IPای که امکان دسترسی به کلاستر را دارند، مشخص و محدود شوند.

معیارهای پذیرش (Acceptance Criteria):

- فهرست یا دامنه آدرس‌های IP مجاز و غیرمجاز باید مشخص و در لیست های مربوطه در فایل پیکربندی ELK قرار گیرند.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۲

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات سادا پیاده‌سازی شود.

توضیحات تکمیلی: محدود کردن IPهایی که می‌توانند به کلاسترهای ELK وصل شوند، می‌تواند تهدید دسترسی‌های غیرمجاز را کاهش دهد.

REQ-40: ارزیابی و رفع آسیب‌پذیری‌های مولفه‌های مختلف سادا

توصیف نیازمندی: آسیب‌پذیری‌های احتمالی موجود در مولفه‌های مختلف سادا باید شناسایی و رفع شوند.

معیارهای پذیرش (Acceptance Criteria):

- بهتر است فهرست کاملی از مولفه‌های مورداستفاده برای توسعه و استقرار سادا به همراه اطلاعات زیر برای هر یک از آنها، تهیه شود:
  - محل استفاده از آن مولفه
  - نسخه مورداستفاده
  - نحوه توسعه (داخلی یا شخص ثالث)
  - لایسنس
  - آخرین وضعیت پشتیبانی

- فهرست آسیب‌پذیری‌های شناسایی شده و وصله‌ها یا آپدیت‌های ارائه شده برای رفع آنها

- این فهرست باید به صورت مداوم به‌روز شود.
- برای رفع آسیب‌پذیری‌های مولفه‌های مختلف، باید وصله‌ها یا آپدیت‌های موردنیاز نصب شوند.

سرویس‌های امنیتی مربوطه: بیشتر سرویس‌ها

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه سادا پیاده‌سازی شود و در ادامه توسط تیم عملیات مدیریت و پایش شود.

توضیحات تکمیلی: شناسایی و رفع آسیب‌پذیری‌های مولفه‌های مختلف استفاده شده در سادا، امکان بهره‌برداری از آنها توسط مهاجمین برای اجرای حملات را کاهش خواهد داد.

REQ-41: بررسی صحت آپدیت‌ها و وصله‌های دریافتی برای مولفه‌های سادا

توصیف نیازمندی: بهتر است هنگام دریافت آپدیت‌ها و وصله‌های مولفه‌های مختلف سادا، صحت آنها قبل از نصب مورد بررسی قرار گیرند.

معیارهای پذیرش (Acceptance Criteria):

- برای آپدیت‌ها و وصله‌های مولفه‌های سادا باید یک checksum تولید شود که بعدا قابل بررسی باشد.
- هنگام نصب آپدیت‌ها و وصله‌ها، این checsumها باید بررسی شوند.
- برای تولید checksumها بهتر است از الگوریتم SHA256 به جای MD5 استفاده شود.

سرویس‌های امنیتی مربوطه: صحت (Integrity)

اولویت‌بندی: ۲

مسئولیت‌ها: تولید checksumها باید توسط تیم توسعه و بررسی آن هنگام نصب، توسط تیم عملیات و تحت نظارت تیم توسعه انجام شود. ضمنا بررسی صحت آپدیت‌ها و وصله‌های مولفه‌های مختلف OpenStack و Multi-AV سمت تیم توسعه باید انجام شود.

توضیحات تکمیلی: بررسی صحت وصله‌ها و آپدیت‌های مولفه‌های مختلف، این اطمینان را فراهم می‌کند که این آرتیفکت‌ها دستکاری نشده‌اند.

REQ-42: مدیریت ریسک آسیب‌پذیری‌های غیرقابل رفع مولفه‌ها

توصیف نیازمندی: برای مولفه‌هایی که وصله‌های مربوط به آسیب‌پذیری‌های شناسایی شده روی آنها قابل نصب نیستند، به عبارت دیگر، آن آسیب‌پذیری‌ها غیر قابل رفع هستند، باید ریسک آنها مدیریت شود.

معیارهای پذیرش (Acceptance Criteria):

- در مورد مدیریت ریسک آسیب‌پذیری‌های غیرقابل رفع، یا باید ریسک آنها پذیرفته شود و یا یک راه‌حل جبرانی برای ریسک آنها در نظرگرفته شود.

سرویس‌های امنیتی مربوطه: بیشتر سرویس‌ها

اولویت‌بندی: ۳

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات سادا و با نظارت تیم توسعه انجام گیرد.

توضیحات تکمیلی: مدیریت ریسک آسیب‌پذیری‌های غیرقابل رفع، این اطمینان را فراهم می‌کند که به وضعیت تمام آسیب‌پذیری‌ها رسیدگی شده است.

REQ-43: اعتبارسنجی فایل‌ها قبل از انتقال به هیولا

توصیف نیازمندی: بهتر است فرمت، نوع و signature فایل‌ها قبل از انتقال از USB به هیولای محیط نامطمئن و همچنین قبل از انتقال از محیط نامطمئن به هیولای محیط مطمئن، در سامانه حافظ، مورد اعتبارسنجی قرار گیرند.

معیارهای پذیرش (Acceptance Criteria):

- هرچند فهرست انواع (extensionها) و فرمت‌های مجاز برای فایل‌ها، باید توسط کارفرما تعیین و در سامانه حافظ، کنترل شود، اما بهتر است از انتقال فایل‌های خطرناک شامل EXE، BIN، MSI، PIF، MSP، COM، GADGET، SCR، HTA، CPL، MSC، JAR، BAT، CMD، VB، VBS، JS، JSE، WS، WSF، WSC، WSH، PS1، PS1XML، PS2، PS2XML، PSC1، PSC2، SCF، LNK، INF، REG، CGI، ZIP و RAR پیشگیری شود.
- Signature فایل‌ها باید بررسی شده و منطبق نوع و فرمت مشخص شده برای فایل‌ها باشد.
- حداکثر اندازه عنوان فایل باید مشخص و کنترل شود.
- حداکثر اندازه فایل هم باید مشخص و کنترل شود.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه پیاده‌سازی شود.

توضیحات تکمیلی: اعتبارسنجی فایل‌ها، خطر آلوده شدن فضای ذخیره‌سازی هیولا را کاهش خواهد داد.

REQ-44: غیرفعال کردن قابلیت Autorun در سیستم عامل سرور تغذیه فایل

توصیف نیازمندی: بهتر است در سیستم‌عامل سرور حافظ، قابلیت اجرای دستورات Autorun غیرفعال شود.

معیارهای پذیرش (Acceptance Criteria):

- این قابلیت باید در تنظیمات GPO سیستم عامل ویندوز انجام شده باشد.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات سادا پیاده‌سازی شود.

توضیحات تکمیلی: غیرفعال کردن قابلیت AutoRun در سیستم‌عامل، می‌تواند از اجرای خودکار برنامه‌های موجود در USB که ممکن است حاوی بدافزار باشند، ممانعت کند.

REQ-45: فعال کردن مکانیزم احرازهویت بایومتریک برای دسترسی به سرور تغذیه فایل

توصیف نیازمندی: بهتر است در صورت امکان مکانیزم احرازهویت بایومتریک در سرور تغذیه فایل فعال شود.

معیارهای پذیرش (Acceptance Criteria):

- برای فعال کردن مکانیزم احرازهویت بایومتریک در سرور، می‌توان فاکتور fingerprint را به عنوان فاکتور دوم در احرازهویت استفاده کرد.

سرویس‌های امنیتی مربوطه: احراز هویت (Authentication)

اولویت‌بندی: ۴

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات سادا انجام شود.

توضیحات تکمیلی: با توجه به فراهم بودن امکان اتصال USB به سرور تغذیه فایل، فعال بودن مکانیزم احرازهویت بایومتریک، خطر تهدید دسترسی غیرمجاز به این سرور را کاهش خواهد داد.

REQ-46: در نظر گرفتن حداقل دسترسی‌های موردنیاز برای نقش‌های کاربری

توصیف نیازمندی: برای نقش‌های کاربری مختلف در سیستم سادا، باید حداقل دسترسی‌های موردنیاز آنها در نظر گرفته شود.

معیارهای پذیرش (Acceptance Criteria):

- بهتر است نقش کاربری مدیر سادا به محتوای داده‌های کاربران پروژه‌های تحت مدیریتش دسترسی نداشته باشد.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۲

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات و با نظارت تیم توسعه پیکربندی شود.

توضیحات تکمیلی: در نظر گرفتن حداقل دسترسی‌های موردنیاز برای نقش‌های مختلف، به خصوص نقش مدیر سادا، از دستکاری محتواهای کاربران و حذف آنها به صورت عمد و غیرعمد پیشگیری خواهد کرد.

REQ-47: شناسایی و حذف دارایی‌های اطلاعاتی بلااستفاده

توصیف نیازمندی: فهرست دارایی‌های اطلاعاتی بلااستفاده در محیط عملیاتی باید شناسایی و حذف شوند.

معیارهای پذیرش (Acceptance Criteria):

- Instanceهای یتیم که متعلق به هیچ کاربر، پروژه یا دامنه ای نیستند باید شناسایی و خاموش شوند و منابع منابع پردازشی و ذخیره سازی آنها بازیابی شوند.
- حساب‌های کاربری قدیمی که استفاده نمی‌شوند و غیرفعال هستند باید شناسایی و حذف یا آرشیو شوند.
- قوانین فایروالی بلااستفاده، باید حذف گردند.
- کلیدها و گواهی‌های TLS بلااستفاده باید حذف شوند.
- دستگاه‌های سخت‌افزاری بلااستفاده بهتر است از فهرست دستگاه‌های پشتیبانی شده توسط شبیه‌ساز QEMU[۱] حذف شوند و فقط دستگاه‌های موردنیاز باقی بماند.

سرویس‌های امنیتی مربوطه: بیشتر سرویس‌ها

اولویت‌بندی: ۳

[۱] Quick Emulator

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات و تحت نظارت تیم توسعه پیاده‌سازی شود.

توضیحات تکمیلی: شناسایی و حذف دارایی‌های اطلاعاتی بلااستفاده، منجر به کاهش سطح حمله سیستم خواهد شد.

REQ-48: امنسازی کوکی‌ها

توصیف نیازمندی: در صورت استفاده از کوکی‌ها در داشبورد Horizon برای نگهداری توکن‌های دسترسی، باید پیکربندی امن آنها انجام شود.

معیارهای پذیرش (Acceptance Criteria):

- فلگ HttpOnly و ویژگی Secure کوکی‌ها باید به مقدار true تنظیم شود.

سرویس‌های امنیتی مربوطه: رمزنگاری (Encryption)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه پیاده‌سازی شود.

توضیحات تکمیلی: تنظیم مقدار true برای فلگ HttpOnly کوکی‌ها از دسترسی کدهای جاوااسکریپت به مقدار این کوکی‌ها و در نتیجه دزدیده شدن این مقادیر، در صورت وقوع حملات XSS ممانعت خواهد کرد. تنظیم مقدار true برای فلگ Secure هم موجب می‌شود، آن کوکی فقط از طریق پروتکل HTTPS قابل انتقال خواهد بود.

REQ-49: غیرفعالسازی قابلیت به اشتراک‌گذاری حافظه اجرایی instanceها در هایپروایزر

توصیف نیازمندی: بهتر است قابلیت به اشتراک‌گذاری حافظه برای محیط‌هایی که tenantها از سطح trust یکسانی برخوردار نیستند، غیرفعال شود.

معیارهای پذیرش (Acceptance Criteria):

- در صورت استفاده از هایپروازیر KVM بهتر است قابلیت KSM و در صورت استفاده از XEN بهتر است قابلیت TPS غیرفعال شود.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه و عملیات پیاده‌سازی شود.

توضیحات تکمیلی:

هایپروایزرها برای مدیریت بهتر حافظه اجرایی ماشین‌های مجازی از تکنیک‌های بهینه‌سازی حافظه استفاده می‌کنند که در آنها صفحات حافظه اجرایی برای ماشین‌هایی که نیاز به ذخیره‌سازی داده‌های یکسانی در حافظه دارند، با استفاده از مکانیزم CoW[۱] به اشتراک گذاشته می‌شود.

اما نکته‌ای که وجود دارد آن است که برای محیط‌های multi-tenantای که همه tenantها از سطح trust یکسانی برخوردار نیستند، بهتر است از این تکنیک‌ها استفاده نشود. زیرا در این حالت ممکن است مهاجم از طریق ماشین مجازی خود، با استفاده از حملات کانال جانبی[۲] و آنالیز زمانهای دسترسی به حافظه اجرایی، اقدام به جمع‌آوری اطلاعات در مورد سایر ماشین‌ها نماید. مثلا آنکه نرم‌افزارهای در حال اجرا و نسخه آنها را روی ماشین‌های مجازی که حافظه آنها به اشتراک گذاشته شده است، شناسایی کند. قابلیت‌های KSM[۳] و TPS[۴] به ترتیب در هایپروایزرهای KVM و XEN نسبت به این حملات، آسیب‌پذیر هستند.

REQ-50: غیرفعالسازی قابلیت دسترسی مستقیم به سخت‌افزارهای ماشین میزبان در هایپروایزر

توصیف نیازمندی: بهتر است قابلیت دسترسی مستقیم ماشین‌های مجازی به سخت‌افزار ماشین میزبان یا همان PCI passthrough در هایپروایزر غیرفعال گردد.

معیارهای پذیرش (Acceptance Criteria):

- در صورتی که دسترسی مستقیم instanceها به سخت‌افزار جزو نیازمندی‌های سیستم باشد، برای پیشگیری از دسترسی مستقیم instanceها به حافظه اجرایی (DMA[۵]) ماشین میزبان، هایپروایزر باید طوری پیکربندی شود که از قابلیت IOMMU[۶] که توسط سخت‌افزارها ارائه می‌شود، استفاده کرده باشد. قابلیت IOMMU در پردازنده‌های Intel با عنوان VT-d و در پردازنده‌های AMD با عنوان AMD-Vi شناخته می‌شوند.
- همچنین از تکنولوژی TPM[۷] برای تشخیص تغییرات firmware دستگاه‌ها باید استفاده شود و یا firmwareها مجددا نصب شوند.

---

[۱] Copy-on-Write

[۲] Side Channel Attack

[۳] Kernel Samepage Merging

[۴] Transparent Page Sharing

[۵] Direc Memory Access

[۶] Input-Output Memory Management Unit

[۷] Trusted Platform Module

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه و عملیات پیاده‌سازی شود.

توضیحات تکمیلی:

بسیاری از هایپروایزرها قابلیتی به نام PCI passthrough دارند که امکان دسترسی و مدیریت مستقیم سخت‌افزار ماشین میزبان را برای ماشین‌های مجازی فراهم می‌کنند. این قابلیت برای instanceهایی که نیازمند استفاده از سخت‌افزارهایی همانند کارت‌های گرافیکی یا GPUها برای تسریع محاسبات خود هستند، مفید خواهد بود. اما نکته‌ای که وجود دارد آن است که این سخت‌افزارها دسترسی مستقیمی به حافظه دارند، در حالیکه قاعدتا instanceها نباید بتوانند به طور مستقیم به حافظه سیستم میزبان دسترسی مستقیم داشته باشند؛ زیرا از این طریق می‌توانند تمام پراسس‌های در حال اجرای میزبان و همچنین instanceها دیگر را رصد کنند. قابلیت IOMMU در سخت‌افزارها مانع از این تهدید خواهد شد.

علاوه بر این، دسترسی مستقیم ماشینهای مجازی به سخت‌افزارها، ممکن است امکان طریق دستکاری firmware یا بخشی از دستگاه جانبی را فراهم کند. با توجه به اینکه این دستگاه توسط سیستم عامل میزبان و دیگر instanceها نیز استفاده می‌شوند، امکان انتشار یک بدافزار در کل سیستم فراهم خواهد شد که در نتیجه آن ممکن است instanceها بتوانند کدهایی را اجرا کنند که خارج از حوزه امنیتی آنها خواهد بود. حتی تهدیدهای دیگری همانند دسترسی به شبکه مدیریتی ماشینها نیز ممکن است فراهم شود. برای کاهش خطر این تهدید، راهکارهایی همانند نصب مجدد firmware و همچنین تکنولوژی TPM برای تشخیص دستکاری غیرمجاز firmware می‌تواند مورداستفاده قرار گیرد.

REQ-51: استفاده از قابلیت‌های compiler hardening هنگام کامپایل QEMU

توصیف نیازمندی: بهتر است هنگام کامپایل شبیه‌ساز QEMU، از گزینه‌های ارائه شده کامپایلرها برای افزایش امنیت باینری‌های تولید شده استفاده شود.

معیارهای پذیرش (Acceptance Criteria):

- در صورت ایجاد هر نوع تغییر در پیکربندی شبیه‌ساز QEMU و نیاز به کامپایل مجدد آن، بهتر است از گزینه‌های ارائه شده کامپایلرها برای compiler hardening استفاده شود.
- برای اطمینان از استفاده از گزینه‌های compiler hardening برای باینری‌های موجود برای شبیه‌ساز QEMU بهتر است این موضوع به کمک ابزارهایی همانند checksec.sh بررسی شود.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۳

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات و با نظارت تیم توسعه انجام شود..

توضیحات تکمیلی:

بسیاری از هایپروایزرها از QEMU برای شبیه‌سازی دستگاه‌های سخت‌افزاری موردنیاز instanceها استفاده می‌کنند. بنابراین این شبیه‌ساز نقش مهمی در عملکرد ماشین‌های مجازی ایفا می‌کند به همین دلیل امن بودن آن در برابر حملاتی همچون Buffer Overflow بسیار اهمیت دارد. برای رسیدن به این هدف، می‌توان از گزینه‌های ارائه شده کامپایلرها همانند RERLO[۱]، Stack canaries، NX[۲]، PIE[۳] و ASLR[۴] برای امنسازی باینری‌های این شبیه‌ساز هنگام کامپایل آن استفاده کرد.

REQ-52: پیکربندی چارچوب کنترل دسترسی MAC برای حفاظت از ماشین‌های مجازی

توصیف نیازمندی: بهتر است برای افزایش سطح کنترل دسترسی ماشین‌های مجازی (instanceها) به منابع موردنیازشان از چارچوب‌های کنترل دسترسی MAC استفاده شود.

معیارهای پذیرش (Acceptance Criteria):

- برای پیاده‌سازی این نیازمندی می‌توان از تکنولوژی sVirt در سیستم عامل‌های مبتنی RedHat استفاده کرد.
- برای سیستم عامل‌های مبتنی Debian هم باید قابلیت AppArmor را پیکربندی کرد.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۳

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات انجام شود..

توضیحات تکمیلی: به کمک چارچوب کنترل دسترسی MAC، می‌توان از تهدیدهایی همانند دسترسی غیرمجاز ماشین‌های مجازی به هایپروازیرها و در نتیجه دسترسی غیرمجاز به سایر ماشین‌های مجازی و همچنین منابع نرم‌افزاری و سخت‌افزاری ماشین میزبان همانند سیستم‌عامل، دستگاه‌ها و شبکه ماشین‌ها، پیشگیری کرد.

---

[۱] RELocation Read-Only

[۲] Never eXecute

[۳] Position Independent Executable

[۴] Address Space Layout Randomization

REQ-53: بکارگیری کلمه عبور برای bootloader ماشین‌مجازی

توصیف نیازمندی: بهتر است برای bootloader در ماشین‌های مجازی یا همان instanceها یک کلمه عبور تعیین شود.

معیارهای پذیرش (Acceptance Criteria):

- باید در پیکربندی grub یک کلمه عبور برای آن در نظر گرفته شود. علاوه بر این امکان تعیین کلمه عبور برای BIOS هم وجود دارد.

سرویس‌های امنیتی مربوطه: احرازهویت (Authentication) و مجازشماری (Authorization)

اولویت‌بندی: ۲

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات و توسعه انجام شود..

توضیحات تکمیلی: استفاده از کلمه عبور برای bootloader ماشین‌های مجازی، امکان دسترسی غیرمجاز مدیر سیستم را به آن ماشین‌ها کاهش خواهد داد.

REQ-54: پشتیبان‌گیری منظم از دایرکتوری /var/lib/nova

توصیف نیازمندی: بهتر است از دایرکتوری /var/lib/nova به طور منظم، نسخه پشتیبان تهیه شود.

معیارهای پذیرش (Acceptance Criteria):

- درصورت تهیه نسخه پشتیبان کامل از ماشین‌های مجازی، باید دایرکتوری /var/lib/nova نیز مورد پشتیبان‌گیری قرار گیرد.
- در صورتی که قرار نیست نسخه پشتیبان کامل از ماشین‌های مجازی تهیه شود، می‌توان دایرکتوری /var/lib/nova/instances را exclude کرد اما مابقی باید در پشتیبان وجود داشته باشد.

سرویس‌های امنیتی مربوطه: دسترس‌پذیری (Availability)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات انجام شود..

توضیحات تکمیلی: این نیازمندی، با توجه به اینکه دایرکتوری /var/lib/nova حاوی اطلاعات مربوط به instanceها و متادیتاهای آنها است، مهم است.

REQ-55: حذف حساب کاربری پیش‌فرض سرویس RabbitMQ

توصیف نیازمندی: بهتر است حساب کاربری پیش‌فرض سرویس RabbitMQ که guest نام دارد، در صورتی که هنوز وجود دارد، حذف شود.

معیارهای پذیرش (Acceptance Criteria):

- این حساب کاربری باید در سرور RabbitMQ حذف شود.
- به ازای تمام سرویس‌هایی که قرار است از سرویس RabbitMQ استفاده کنند، باید یک حساب کاربری مجزا با دسترسی‌های مشخص و محدود وجود داشته باشند.

سرویس‌های امنیتی مربوطه: احرازهویت (Authentication)، مجازشماری (Authorization)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه و عملیات انجام شود.

توضیحات تکمیلی: احرازهویت و مجازشماری سرویس‌ها از طریق حساب‌های کاربری مجزایی که برایشان در نظر گرفته می‌شود، امکان ردیابی و نظارت دقیقتر فعالیت‌های آنها را فراهم می‌کند.

REQ-56: پیشگیری از اجرای حملات ARP Spoofing توسط ماشین‌های مجازی

توصیف نیازمندی: در صورت وجود معماری flat برای شبکه ماشین‌های مجازی، باید از حملات ARP Spoofing پیشگیری شود.

معیارهای پذیرش (Acceptance Criteria):

- برای پیشگیری از حملات ARP Spoofing باید قابلیت مربوطه در پیکربندی پلاگین سوئیچ مجازی (همانند Open vSwitch) فعال شود.

سرویس‌های امنیتی مربوطه: احرازهویت (Authentication)، مجازشماری (Authorization)

اولویت‌بندی: ۲

مسئولیت‌ها: این نیازمندی باید توسط تیم توسعه و عملیات انجام شود.

توضیحات تکمیلی: در صورتی که معماری شبکه سیستم، به صورت flat باشد، پیاده‌سازی این نیازمندی، خطر تهدید ARP Spoofing را کاهش خواهد داد.

REQ-57: الزام quotaهای پیش‌فرض برای پروژه‌ها

توصیف نیازمندی: بهتر است تعداد قابل دسترس شبکه برای پروژه‌ها یا همان quotaها، مستقل از محدودیت‌هایی که به ازای هر پروژه تعریف می‌شود، به طور کلی برای تمام پروژه‌ها مشخص شده باشند.

معیارهای پذیرش (Acceptance Criteria):

- بهتر است تمام quotaها شامل حداکثر تعداد شبکه‌ها و زیرشبکه‌های قابل تعریف برای tenantها، تعداد پورت‌های قابل تخصیص، تعداد security groupهای قابل تعریف و تعداد قوانین قابل تعریف در آنها مشخص و محدود شود.
- بهتر است به ازای هیچ کدام از این پارامترها مقدار 1- یعنی نامحدود در نظر گرفته نشود.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization) و دسترس‌پذیری (Availability)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط عملیات و با نظارت تیم توسعه در در فایل پیکربندی سرویس neutron مشخص شود.

توضیحات تکمیلی: پیاده‌سازی این نیازمندی، این اطمینان را فراهم می‌کند که در نهایت، منابع قابل استفاده محدود خواهند شد؛ حتی اگر مدیر سادا، به ازای پروژه‌های تحت نظارت خود آنها را تعریف نکرده باشد.

REQ-58: پیشگیری از تداخل قوانین تعریف شده در security groupها

توصیف نیازمندی: در صورت تعریف security groupها در سرویس Neutron، باید این قابلیت ذاتی در سرویس Nova غیرفعال شود و تمام درخواست‌های security group به سمت سرویس Neutron پراکسی شود.

معیارهای پذیرش (Acceptance Criteria):

- پارامترهای firewall_driver و security_group_api در پیکربندی سرویس Nova باید به درستی تنظیم شوند.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات و با نظارت تیم توسعه در پیکربندی سرویس Nova انجام شود.

توضیحات تکمیلی: این نیازمندی از هر گونه تداخلی که ممکن است به دلیل تعریف security groupها هم در سرویس Neutron و هم Nova به وجود آید، ممانعت می‌کند.

REQ-59: محدود کردن اتصالات سرویس‌های داخلی و حساس

توصیف نیازمندی: آدرس IP سرویس‌های داخلی و حساس بهتر است در شبکه مدیریتی و محدود سادا پیکربندی شوند.

معیارهای پذیرش (Acceptance Criteria):

- آدرس‌های IP و پورت‌های سرورهای پایگاه داده‌ها و RabbitMQ و admin API endpointهای سرویس‌ها باید مشخص و در فایل‌های پیکربندی مربوطه مشخص و در فایروال محدود شوند.

سرویس‌های امنیتی مربوطه: مجازشماری (Authorization)

اولویت‌بندی: ۱

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات و با نظارت تیم توسعه در پیکربندی سرورهای مربوطه باید انجام شود.

توضیحات تکمیلی: با محدود کردن اتصالات سرورهای حساس مثل پایگاه داده‌ها و RabbitMQ این اتصالات در شبکه مدیریتی و محدود شده سادا برقرار خواهند شد.

REQ-60: احرازهویت کلاینت‌ها در Kafka

توصیف نیازمندی: بهتر است هویت کلاینت‌های سرویس Kafka قبل از اتصال به آن، مشخص و احراز شود.

معیارهای پذیرش (Acceptance Criteria):

- در حال حاضر Kafka از مکانیزم‌های احرازهویت مبتنی بر چارچوب SASL[۱] شامل SASL/PLAIN، SASL/GSSAPI، SASL/SCRAM-SHA-256 و SASL/OAUTHBEARER، پشتیبانی می‌کند و باید احرازهویت کلاینت‌ها به کمک یکی از این مکانیزم‌ها انجام شود.

دسته‌بندی: احراز هویت (Authentication)

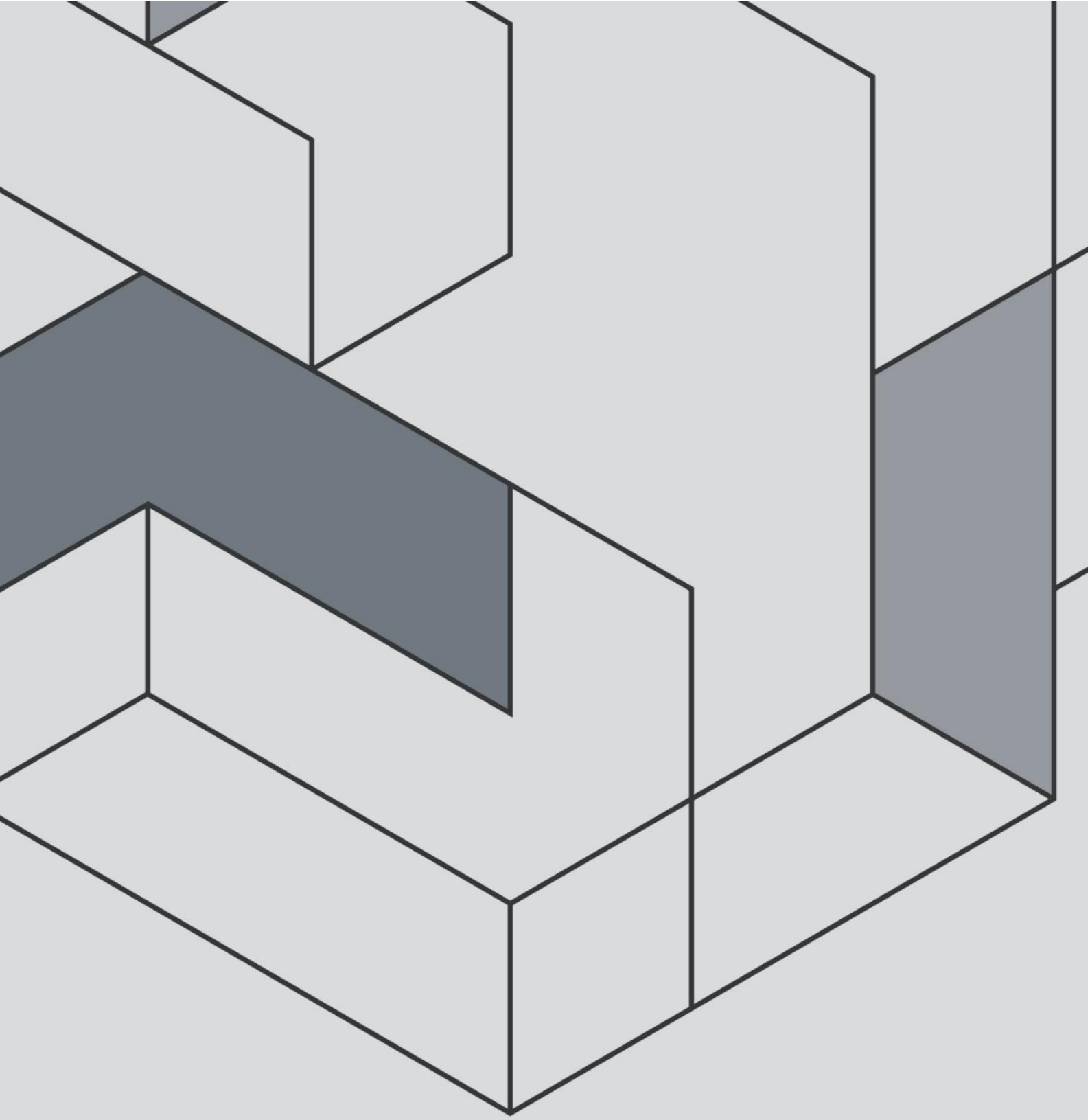
اولویت‌بندی: ۲

مسئولیت‌ها: این نیازمندی باید توسط تیم عملیات سادا انجام شود.

توضیحات تکمیلی: با توجه به اینکه داده‌های رد و بدل شده بین مولفه‌های مختلف Kafka در سیستم سادا شامل لاگ‌های مولفه‌ها و سرویس‌های مختلف این سیستم است، هر نوع دسترسی غیرمجاز به آنها منجر به

[۱] Simple Authentication and Security Layer

افشای اطلاعات موجود در لاگ‌ها خواهد شد. با توجه به اینکه این سرویس، به طور پیش‌فرض بدون مکانیزم احرازهویت، نصب می‌شود، باید مکانیزم موردنظر توسط ادمین آن پیکربندی شود.



www.aminraay.com
info@aminraay.com
+98 21 44899372
+98 24 33411694