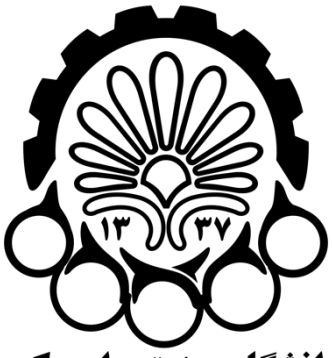
فصل ۱۸:

حسابرسی فناوری های نوین و نوظهور



دانشگاه صنعتی امیرکبیر

(پلی تکنیک تهران)

دوره DBA حسابرسی فناوری اطلاعات

استاد: آقای دکتر حمیدزاده

دانشپذیر: نیما رحیمی

در این فصل، گام‌های اساسی را مورد بحث قرار می‌دهیم که می‌توانید از آن‌ها برای کمک به اجرای حسابرسی فناوری‌هایی که در سایر بخش‌های این کتاب پوشش داده نشده‌اند، استفاده کنید. این گام‌ها به حوزه‌های زیر تقسیم خواهند شد:

- گام‌های مقدماتی
- مدیریت حساب کاربری
- مدیریت مجوزهای دسترسی
- امنیت و کنترل‌های شبکه
- پایش امنیتی و سایر کنترل‌های عمومی

پیش‌زمینه(BACKGROUND)

اگرچه آرزو داشتیم این کتاب می‌توانست هر فناوری ممکنی را که ممکن است در طول حسابرسی‌های خود با آن مواجه شوید پوشش دهد، اما چنین کتابی برای مطالعه بسیار طولانی، برای حمل بسیار سنگین و برای خرید بسیار گران‌قیمت می‌شد. ما تلاش کرده‌ایم برخی از فناوری‌های رایج‌تر را با جزئیات پوشش دهیم؛ با این حال، کثرتِ فناوری‌های موجود (در حالی که موارد جدید به طور مداوم معرفی می‌شوند) بسیار زیاد است و شما بدون شک نیاز به حسابرسی فناوری‌هایی را پیدا خواهید کرد که به طور مشخص در این کتاب پوشش داده نشده است.

خوشبختانه، زمانی که موضوع را کالبدشکافی (تجزیه) کنید، متوجه می‌شوید که صرف‌نظر از آنچه حسابرسی می‌کنید، همان مفاهیم پایه‌ای صدق می‌کنند. حساب‌های کاربری باید ایجاد و مدیریت شوند. شما باید روشی برای احراز هویت (Authentication) آن حساب‌ها داشته باشید و لازم است آنچه را که آن حساب‌ها مجاز به انجامش هستند (Authorization)، مدیریت کنید.

سیستم‌ها باید به صورت امن پیکربندی و پایش شوند. روش‌های اتصال به فناوری باید ایمن‌سازی و مدیریت گردند. اصل مطلب روشن است. اگرچه پیاده‌سازی فنی این کنترل‌ها بر اساس فناوری خاصی که حسابرسی می‌کنید متفاوت خواهد بود، اما همان مفاهیم اساسی به طور کلی اعمال می‌شوند. هدف این فصل، ارائه یک چارچوب به شما

است که بتوانید ظرایف و جزئیات دقیق فناوری خاص و نوظهور مورد بررسی خود را در آن جای‌گذاری کنید.

ضرورت‌های حسابرسی فناوری‌های نوین

همان‌طور که کاوش در فناوری‌های جدید را آغاز می‌کنید، برخی چارچوب‌ها و «بهترین تجارب (Best Practices) «می‌توانند به شما در ساختاردهی به افکارتان کمک کنند.

چارچوب‌های تعمیم‌یافته(Generalized Frameworks)

چارچوب‌های تعمیم‌یافته در جلساتی مفید هستند که در آن‌ها از شما خواسته می‌شود در لحظه (بدون آمادگی قبلی)، سؤالات و ریسک‌های احتمالی مرتبط با یک اپلیکیشن، فناوری یا پروژه را مطرح کنید. حتی ممکن است خود را در وضعیتی بیابید که وارد جلسه‌ای شده، برگه سفیدی برداشته و پیش از شروع جلسه، واژگان«PPTM» ، «STRIDE» و) «PDIO» که در بخش‌های بعدی توضیح داده شده‌اند) را در بالای برگه یادداشت کنید.

سپس، هم‌زمان با بحث درباره سیستم یا پروژه مورد بررسی، می‌توانید سؤالاتی بپرسید و درباره نحوه رسیدگی به هر یک از عناصر این چارچوب‌ها یادداشت‌برداری کنید. در پایان جلسه، اگر در مقابل هر یک از عناصر چارچوب با «جاهای خالی (Blanks) «مواجه شوید، این احتمال وجود دارد که شکافی (Gap) در کنترل‌ها کشف کرده باشید. البته، این نوع فرآیند فکریِ سریع و سرانگشتی (Quick-and-dirty) هرگز نباید جایگزین آزمون‌های دقیق و کامل شود، اما می‌تواند هنگام مشارکت در مباحث اولیه و مشاوره در مورد کنترل‌ها مفید باشد. همچنین همین چارچوب‌ها می‌توانند در زمان تدوین گام‌های دقیق حسابرسی برای فناوری‌ها یا سیستم‌های جدیدِ تحتِ بررسی، سودمند باشند.

چارچوبPPTM

چارچوب **افراد، فرآیندها، ابزارها و معیارها(PPTM)** ، یک چارچوب عالی برای طوفان فکری (Brainstorming) جهت بررسی یک سیستم در سطح کلان (Macro level) است. گام‌های دقیق و فنیِ بازبینی، بر فصول این بخش از کتاب غالب هستند PPTM .به شما

کمک می‌کند تا گام‌های اختصاصی خود را متناسب با شرایط منحصربه‌فردتان، به سرعت و با کارایی بالا استخراج کنید.

افراد(People)

»افراد» در مدلPPTM ، هر جنبه‌ای از سیستم را که با انسان در ارتباط است توصیف می‌کند. برای مثال، اگر فرصت ارائه نظر در طول فرآیند توسعه سیستم را دارید، اطمینان حاصل کنید که افراد مناسب در برنامه‌ریزی، طراحی، پیاده‌سازی یا عملیات پروژه حضور دارند و ذینفعان (Stakeholders) صحیح درگیر شده‌اند. اگر سیستم شامل کاربران نهایی است، اطمینان یابید که کنترل‌هایی پیرامون «تخصیص و لغو دسترسی Provisioning) « and Deprovisioning)وجود دارد و کاربران نهایی در بخش‌هایی که در نهایت با آن‌ها تعامل (Interface) خواهند داشت، مشارکت داده شده‌اند. کمتر چیزی خجالت‌آورتر از این است که زمان و هزینه صرفِ راه‌اندازی سیستمی شود، فقط برای اینکه در نهایت مشخص گردد مدیریت ارشد آن را تایید نمی‌کند یا کاربران نهایی درمی‌یابند که رابط کاربری (Interface) بیش از حد برای استفاده پیچیده است.

فرآیندها(Processes)

»فرآیندها» درPPTM ، هر جنبه‌ای از سیستم را که با یک سیاست(Policy) ، رویه (Procedure)، متد یا خط‌مشی عملیاتی درگیر است، توصیف می‌کند. تعامل سیستم با سیستم‌های رابط (Interfacing Systems) را بازبینی کرده و انطباق با مدل‌های امنیتی را تایید نمایید (برای مثال، اطمینان حاصل کنید که دیواره‌های آتش (Firewalls) جهت محافظت از سیستم در برابر سیستم‌های خارجی، کاربران، شرکای تجاری و موارد مشابه در جای خود قرار دارند). رویه‌ها و سیاست‌ها باید به گونه‌ای نوشته شوند که نحوه استفاده مورد انتظار از سیستم را پشتیبانی کنند. همچنین مستندات کافی باید برای حمایت از تکنسین‌هایی که نیاز به نگهداری (Maintain) سیستم دارند، وجود داشته باشد.

ابزارها(Tools)

»ابزارها» درPPTM ، هر جنبه‌ای از سیستم را که با یک فناوری یا محصول عینی (Concrete) سر و کار دارد، توصیف می‌کند. اطمینان حاصل کنید که سخت‌افزار و محیط مناسب برای پشتیبانی از سیستم وجود دارد و سیستم با فناوری‌های پیشنهادیِ متناسب با سیاست‌ها و رویه‌های مدنظر شما، تعامل دارد. تایید کنید که سیستم و زیرساخت (Infrastructure)به طور مناسب مورد آزمون و حسابرسی قرار می‌گیرند.

معیارها(Measures)

»معیارها» درPPTM ، هر جنبه‌ای از سیستم را که از نظر مفهومی قابل اندازه‌گیری (Quantifiable)است، توصیف می‌کند؛ مواردی نظیر اهداف تجاری یا عملکرد اپلیکیشن. برای مثال، می‌توانید تایید کنید که سیستم با معیارهای پذیرش (Acceptance Criteria) که به خوبی مستند و سنجیده شده‌اند، مطابقت دارد. اگر هدف سیستم حل یک مشکل تجاریِ قابل اندازه‌گیری است، تایید کنید که در واقع آن مشکل را حل می‌کند. تایید کنید که لاگ‌ها (Logs) معنادار هستند و می‌توانید عملکرد سیستم را اندازه‌گیری کنید.

مدلSTRIDE

واژه اختصاری STRIDE مخفف مفاهیم زیر است:

جعل هویت (Spoofing identity)،

دست‌کاری داده‌ها (Tampering with data)،

انکار (Repudiation)،

افشای اطلاعات (Information disclosure)،

محروم‌سازی از سرویس (Denial of service)

و **ارتقای سطح دسترسی** .(Elevation of privilege)

STRIDE یک روش‌شناسی (Methodology) است که برای شناسایی تهدیدات شناخته شده به کار می‌رود. این مدل، نمونه‌ای از یک مدل ساده شده «تهدید-ریسک» است که به خاطر سپردن و به‌کارگیری آن آسان است. هنگام ارزیابی یک سیستم، می‌توانید از این واژه اختصاری برای تدوین گام‌هایی استفاده کنید که به چگونگی کاهش (تعدیل) هر یک از ریسک‌های زیر می‌پردازند:

جعل هویت(Spoofing Identity)

جعل هویت یک ریسک کلیدی برای سیستم‌هایی است که کاربران بسیاری دارند اما یک «بافتار اجرای واحد (Single Execution Context) «ارائه می‌دهند. به طور خاص، کاربران نباید قادر باشند به جای کاربر دیگری قرار گیرند یا ویژگی‌های (Attributes) کاربر دیگری را تصاحب کنند.

دست‌کاری داده‌ها(Tampering with Data)

داده‌ها باید در مکانی امن و با کنترل‌های دسترسی مناسب ذخیره شوند. سیستم باید داده‌های دریافتی از کاربر را با دقت بررسی کرده و پیش از ذخیره‌سازی یا استفاده، از منطقی (Sane) و معتبر بودن آن‌ها اطمینان حاصل کند (اعتبارسنجی .(در برنامه‌های تحت وب و سایر اپلیکیشن‌هایی که دارای مؤلفه کلاینت (سمت کاربر) هستند، شما باید بررسی‌های اعتبارسنجی خود را در سمت سرور انجام دهید و نه در سمت کلاینت؛ چرا که بررسی‌های اعتبارسنجی در سمت کلاینت ممکن است مورد دست‌کاری قرار گیرند. این موضوع به ویژه برای اپلیکیشن‌های تحت وب اهمیت دارد، جایی که کاربران می‌توانند به طور بالقوه داده‌های تحویل داده شده به خود را تغییر دهند، آن‌ها را بازگردانند و از این طریق اعتبارسنجی سمت کلاینت را دست‌کاری کنند. اپلیکیشن نباید داده‌هایی نظیر نرخ بهره یا دوره‌های زمانی را که تنها از داخل خودِ برنامه قابل دستیابی هستند به کاربر ارسال کند و به وی اجازه دهد که آن داده‌ها را به طور بالقوه دست‌کاری نماید.

انکار(Repudiation)

کاربران ممکن است در صورت کافی نبودنِ حسابرسی (Auditing) یا ثبت سوابق (Recordkeeping)از فعالیت‌هایشان، تراکنش‌ها را به چالش بکشند (انکار کنند). برای مثال، اگر کاربری بگوید: «اما من هیچ پولی به این حساب خارجی انتقال ندادم!» و شما نتوانید فعالیت‌های او را از طریق اپلیکیشن ردیابی کنید، به احتمال بسیار زیاد آن تراکنش باید به عنوان خسارت (ضرر) از حساب خارج شود .(Write off) بنابراین، باید بررسی کنید که آیا سیستم به کنترل‌های «عدم انکار(Nonrepudiation) »، مانند لاگ‌های دسترسی و ردپای حسابرسی (Audit trails) در هر سطح (Tier) نیاز دارد یا خیر. ترجیحاً، سیستم باید صرفاً با سطح دسترسی (Privileges) کاربر و نه چیزی فراتر از آن اجرا شود.

افشای اطلاعات(Information Disclosure)

کاربران به درستی نسبت به ارسال جزئیات خصوصی به یک سیستم محتاط (Wary) هستند. اگر برای یک مهاجم امکان افشای عمومی داده‌ها، به ویژه داده‌های کاربر فراهم باشد — خواه به صورت ناشناس و خواه به عنوان یک کاربر مجاز — بلافاصله اعتماد از بین رفته و دوره‌ای طولانی از خدشه‌دار شدن اعتبار (Reputation loss) رخ خواهد داد. بنابراین، سیستم‌ها باید شامل کنترل‌های قوی جهت جلوگیری از دست‌کاری (Tampering)و سوءاستفاده از شناسه کاربری (User ID) باشند و امنیت داده‌های سیستم را که در پایگاه‌های داده و فایل‌های داده ذخیره شده‌اند، تأمین کنند.

محروم‌سازی از سرویس(Denial of Service)

طراحان سیستم باید آگاه باشند که سیستم‌های آن‌ها ممکن است در معرض حملات محروم‌سازی از سرویس (DoS) قرار گیرد. از این رو، استفاده از منابع پرهزینه مانند فایل‌های حجیم، محاسبات پیچیده، جست‌وجوهای سنگین یا پرس‌وجوهای (Queries)طولانی باید برای کاربران احراز هویت شده و مجاز محفوظ بماند و نباید در دسترس کاربران ناشناس باشد.

برای سیستم‌هایی که از این مزیت (اجبار به احراز هویت) برخوردار نیستند، تمام ابعاد سیستم باید به گونه‌ای مهندسی شود که کمترین حجم کار ممکن را انجام دهد، از پرس‌وجوهای سریع و حداقلی در پایگاه داده استفاده کند و از ارائه فایل‌های حجیم یا

لینک‌های منحصربه‌فرد برای هر کاربر اجتناب نماید تا از حملات ساده محروم‌سازی از سرویس پیشگیری شود.

ارتقای سطح دسترسی(Elevation of Privilege)

چنانچه سیستمی نقش‌های کاربری و مدیریتی متمایزی ارائه می‌دهد، اطمینان حاصل کنید که کاربر نمی‌تواند نقش خود را به نقشی با امتیازات بالاتر ارتقا دهد. به طور خاص، صرفِ عدم نمایش پیوندهای (Links) مربوط به نقش‌های دارای امتیاز، کافی نیست. در عوض، تمامی اقدامات باید از طریق یک ماتریس مجوزدهی (Authorization Matrix) فیلتر شوند تا اطمینان حاصل گردد که تنها نقش‌های مجاز می‌توانند به عملکردهای حساس (امتیازدار) دسترسی داشته باشند.

مدل PDIO

واژه PDIO از شرکت سیسکو (Cisco Systems)برگرفته شده و مخفف **برنامه‌ریزی، طراحی، پیاده‌سازی و عملیات** (Planning, Design, Implementation, and Operations)است. گاهی لازم است چالش‌های بالقوه را در هر مرحله از یک پروژه در نظر بگیرید. ممکن است این چارچوب را هنگام بررسی یک سیستم جدید و پیش‌بینی چالش‌های آتی مفید بیابید. برای مثال، اگر مدیران سیستم در یک جلسه برنامه‌ریزی یا طراحی برای یک راهکار شبکه‌ای در حال تبادل ایده باشند و مهندس ارشد شبکه در اتاق حضور نداشته باشد، ممکن است مشکلی رخ دهد. اگر از شما به عنوان حسابرس خواسته شود که به پیاده‌سازی یک راهکار جدید نگاه کنید، باید فوراً سؤالاتی در مورد عملیات مستمر (Ongoing Operations)آن راهکار بپرسید. برای جزئیات بیشتر در مورد حسابرسی پروژه‌های شرکتی به فصل ۱۷ مراجعه کنید.

تجارب موفق (Best Practices)

این تجارب می‌توانند به شما در شناسایی سریع نقاط ضعف رایج و کنترل‌های ضعیف کمک کنند.

به‌کارگیری دفاع در عمق(Apply Defense-in-Depth)

رویکردهای لایه‌بندی شده در بلندمدت امنیت بیشتری نسبت به یک توده پیچیده از معماری امنیتی فراهم می‌کنند. برای مثال، می‌توانید از لیست‌های کنترل دسترسی (ACLs)روی تجهیزات شبکه و دیواره آتش (Firewall) استفاده کنید تا تنها ترافیک ضروری اجازه یابد به سیستم برسد. این رویکرد ریسک کلی نفوذ (Compromise) به سیستم را به میزان قابل توجهی کاهش می‌دهد، زیرا دسترسی به سرویس‌ها، پورت‌ها و پروتکل‌هایی را که در غیر این صورت می‌توانستند در معرض مخاطره قرار گیرند، به سرعت حذف می‌کنید.

اصول کلیدی امنیت نرم‌افزار: از مدل امنیتی مثبت تا اصل کمترین سطح دسترسی

در دنیای امنیت سایبری، رویکردهای پیشگیرانه همواره بر روش‌های واکنشی برتری دارند. درک مفاهیمی مانند **مدل امنیتی مثبت**، **شکست امن (Fail-Safe)** و **اصل کمترین سطح دسترسی**، ستون‌های اصلی طراحی یک سیستم نفوذناپذیر را تشکیل می‌دهند. در ادامه، این اصول را با جزئیات بررسی می‌کنیم:

۱ .استفاده از مدل امنیتی مثبت(Positive Security Model)

مدل امنیتی مثبت که به آن **لیست سفید (Whitelist)** نیز گفته می‌شود، بر پایه اصل «اجازه به موارد شناخته‌شده و مسدودسازی سایرین» استوار است.

- **تفاوت با مدل منفی :(Blacklist)** در مدل منفی، همه چیز مجاز است مگر مواردی که به عنوان مخرب شناخته شوند (مانند آنتی‌ویروس‌ها). این روش همیشه یک قدم عقب‌تر از مهاجمان است، زیرا باید دائماً با تهدیدات جدید به‌روز شود.
- **مزیت اصلی** :با استفاده از رویکرد) **Deny by Default**مسدودسازی پیش‌فرض)، سطح حملات (Attack Surface) به شدت کاهش می‌یابد. حتی اگر آسیب‌پذیری ناشناخته‌ای وجود داشته باشد، چون در لیست مجاز نیست، امکان سوءاستفاده از آن به حداقل می‌رسد.

۲ .شکست امن(Fail Safely)

نحوه برخورد سیستم با خطاها و شکست‌های غیرمنتظره، تعیین‌کننده میزان پایداری امنیت آن است. وقتی یک فرآیند با خطا مواجه می‌شود، سیستم نباید به حالتی بازگردد که امنیت را به خطر بیندازد.

- **عدم افشای اطلاعات** :خطاها باید از دید کاربر نهایی مشابه یک عملیات «مسدود شده (Block)» به نظر برسند. نمایش جزئیات فنی خطا (Stack Trace) یا مسیرهای فایل به کاربر، نقشه راهی برای نفوذ در اختیار او قرار می‌دهد.
- **مدیریت لاگ‌ها** :تمام جزئیات خطا باید در محیطی امن و جداگانه (Server Logs) ثبت شوند تا مدیران سیستم بتوانند آن را بررسی کنند، بدون اینکه کاربر عادی از پشت‌صحنه سیستم مطلع شود.

۳ .رعایت اصل کمترین سطح دسترسی(Least Privilege)

این اصل بیان می‌کند که هر حساب کاربری، پردازه یا سیستم، باید تنها به منابع و اطلاعاتی دسترسی داشته باشد که برای انجام وظیفه مشخص خود به آن‌ها نیاز دارد؛ نه بیشتر و نه کمتر.

نوع منبع	محدودیت‌های اعمال شده
دسترسی‌های سیستمی	محدود کردن دسترسی به فایل‌های ریشه (Root) یا تنظیمات حساس.
منابع سخت‌افزاری	تعیین سقف مصرف برای CPU، حافظه (RAM) و پهنای باند شبک
دسترسی فایل	استفاده از سطوح دسترسی Read-Only به جای Full Control در موارد غیرضروری.

چرا این اصل حیاتی است؟

اگر یک حساب کاربری با دسترسی محدود هک شود، آسیب وارده به همان بخش محدود می‌شود. اما اگر آن حساب دسترسی ادمین داشته باشد، کل سیستم در معرض سقوط قرار می‌گیرد.

استفاده از مدل امنیتی مثبت

مدل‌های امنیتی مثبت (لیست سفید) تنها به آنچه در لیست است اجازه می‌دهند و هر چیز دیگری را به صورت پیش‌فرض حذف می‌کنند. با این حال، مدل‌های امنیتی منفی (لیست سیاه) به صورت پیش‌فرض به همه چیز اجازه می‌دهند و تنها مواردی را که می‌دانید بد هستند، حذف می‌کنند. این چالشی برای برنامه‌های آنتی‌ویروس است که باید آن‌ها را مدام به‌روزرسانی کنید تا با تعداد حملات (ویروس‌های) جدیدِ ممکن که می‌توانند بر سیستم شما تأثیر بگذارند، همگام بمانید. مشکل این مدل، اگر مجبور به استفاده از آن هستید، این است که مطلقاً باید مدل را به‌روز نگه دارید. با این حال، حتی با به‌روز بودن مدل، ممکن است آسیب‌پذیری‌ای وجود داشته باشد که از آن بی‌خبرید و سطح حمله شما بسیار بزرگ‌تر از زمانی است که از یک مدل امنیتی مثبت استفاده می‌کردید. روش ارجح این است که به صورت پیش‌فرض (دسترسی را) رد کنید و تنها به مواردی اجازه دهید که آگاهانه مجاز می‌دانید.

شکست امن

وقتی یک سیستم با شکست مواجه می‌شود، می‌توان به سه روش با آن برخورد کرد: اجازه، مسدودسازی یا خطا. به طور کلی، خطاهای سیستم باید به همان شیوه‌ی عملیات عدم اجازه (مسدودسازی) از دید کاربر نهایی شکست بخورند. این مهم است، زیرا به این معنی است که کاربر نهایی اطلاعات اضافه‌ای برای استفاده ندارد که ممکن است به او در به خطر انداختن سیستم کمک کند. هر آنچه می‌خواهید ثبت (Log) کنید و هر پیامی را که می‌خواهید در جای دیگری نگه دارید، اما اطلاعات اضافه‌ای به کاربر ندهید که ممکن است از آن‌ها برای به خطر انداختن سیستم شما استفاده کند.

اجرا با کمترین سطح دسترسی

اصل کمترین سطح دسترسی حکم می‌کند که حساب‌ها کمترین میزان دسترسیِ ممکن را برای انجام فعالیت خود داشته باشند. این شامل حقوق کاربران و مجوزهایِ منابع مانند محدودیت‌های پردازنده، ظرفیت حافظه، پهنای باند شبکه و مجوزهای سیستم فایل می‌شود.

پرهیز از امنیت مبتنی بر پنهان‌کاری

مخدوش‌سازی داده‌ها، یا مخفی کردن آن‌ها به جای رمزنگاری، یک مکانیسم امنیتی بسیار ضعیف است. اگر یک انسان بتواند بفهمد چگونه داده‌ها را مخفی کند، چه چیزی مانع فرد دیگر از یادگیری نحوه بازیابی داده‌ها می‌شود؟ برای مثال، در نظر بگیرید که برخی افراد چگونه کلید خانه خود را زیر پادری مخفی می‌کنند. یک مجرم ساده‌ترین راه ممکن برای ورود به خانه را می‌خواهد و مکان‌های رایج مانند زیر پادری، سنگی که به در نزدیک‌تر است و بالای چهارچوب در را برای یافتن کلید بررسی می‌کند. هرگز داده‌های حیاتی را که می‌توان رمزنگاری کرد، مخدوش نکنید (یا بهتر از آن، اصلاً ذخیره نکنید). این به معنای نادیده گرفتن ارزشِ دشوار کردن مکان‌یابی داده‌های حیاتی برای مهاجم نیست. برای مثال، اگر سیستمی دارید که گذرواژه‌ها را ذخیره می‌کند، بهتر است نام سروری که سیستم را در خود جای داده "passwordserver" نگذارید، زیرا این کار باعث می‌شود مهاجم به راحتی بداند که باید آن سیستم را هدف قرار دهد. بهتر بود نام سرور را چیزی مبهم و غیرجذاب بگذارید. با این حال، نباید به آن به عنوان تنها مکانیسم دفاعی خود تکیه کنید و باید امنیت سرور و داده‌های آن را با این فرض تامین کنید که مهاجم در نهایت آن را پیدا خواهد کرد.

ساده نگه داشتن امنیت

مکانیسم‌های امنیتی ساده به راحتی قابل تأیید و پیاده‌سازیِ صحیح هستند. بروس اشنایر، رمزنگار، برای این پیشنهاد مشهور است که سریع‌ترین روش برای شکستن یک الگوریتم رمزنگاری، دور زدن آن است. در صورت امکان، از مکانیسم‌های امنیتیِ بیش از حد پیچیده اجتناب کنید. توسعه‌دهندگان باید از به‌کارگیری نقیض‌های مضاعف و معماری‌های پیچیده، زمانی که یک رویکرد ساده سریع‌تر و آسان‌تر است، خودداری کنند. پیچیدگی را با لایه‌ها اشتباه نگیرید. لایه‌ها خوب هستند؛ پیچیدگی نیست.

تشخیص نفوذ و نگهداری لاگ‌ها

سیستم‌ها باید دارای قابلیت ثبت وقایعِ داخلی باشند که محافظت شده و به راحتی قابل خواندن باشند. لاگ‌ها به شما در عیَب‌یابی مسائل کمک می‌کنند و به همان اندازه مهم هستند که به شما کمک کنند متوجه شوید یک اپلیکیشن چه زمانی یا چگونه ممکن است به خطر افتاده باشد.

هرگز به زیرساخت‌ها و خدمات خارجی اعتماد نکنید

بسیاری از سازمان‌ها از قابلیت‌های پردازشیِ شرکای شخص ثالث استفاده می‌کنند که به احتمال زیاد سیاست‌ها و وضعیت‌های امنیتیِ متفاوتی نسبت به شما دارند. بعید است که بتوانید به طور کامل بر هر شخص ثالثِ خارجی، خواه کاربران خانگی باشند یا تأمین‌کنندگان و شرکای بزرگ، کنترل داشته باشید. بنابراین، اعتماد ضمنی به سیستم‌هایی که به صورت خارجی اداره می‌شوند، خطرناک است.

برقراری پیش‌فرض‌های امن

سیستم‌های شما باید با امن‌ترین تنظیمات پیش‌فرضِ ممکن که همچنان اجازه تداوم فعالیت‌های تجاری را می‌دهند، به دستتان برسند یا به کاربران ارائه شوند. این کار ممکن است مستلزم آموزش کاربران نهایی یا اطلاع‌رسانی باشد، اما نتیجه نهایی، کاهش قابل‌توجه سطح حمله است؛ به‌ویژه زمانی که یک سیستم در میان جمعیت گسترده‌ای توزیع می‌شود.

استفاده از استانداردهای باز

در صورت امکان، امنیت را بر پایه‌ی استانداردهای باز بنا کنید تا قابلیت جابه‌جایی و همکاری‌پذیری افزایش یابد. استانداردهای باز، استانداردهایی هستند که به صورت عمومی در دسترس بوده و معمولاً از طریق یک فرآیند مشارکتی و باز توسعه یافته و نگهداری می‌شوند. از آنجایی که زیرساخت شما احتمالاً ترکیبی ناهمگون از پلتفرم‌های

مختلف است، استفاده از استانداردهای باز به تضمین سازگاری بین سیستم‌ها همزمان با رشد شما کمک می‌کند. علاوه بر این، استانداردهای باز اغلب شناخته‌شده هستند و توسط همتایان در صنعت امنیت مورد بررسی دقیق قرار می‌گیرند تا از تداوم امنیت آن‌ها اطمینان حاصل شود.

روش‌های آزمون جهت حسابرسی فناوری‌های نوین و سایر تکنولوژی‌ها

گام‌های اولیه

۱ .درباره فناوری مورد بررسی تحقیق کنید تا جزئیات نحوه عملکرد و نقاط کنترلی کلیدی آن را بیاموزید. مراحل حسابرسی را در صورت لزوم به برنامه حسابرسی ارائه‌شده در این فصل اضافه کنید.

هر فناوری ریسک‌ها و ویژگی‌های امنیتی منحصربه‌فرد خود را دارد. اگرچه مراحل تشریح شده در این فصل، پایه و ساختاری را برای شروع در اختیار شما قرار می‌دهند، اما این مراحل باید با جزئیات مربوط به فناوری خاصی که حسابرسی می‌کنید، تکمیل شوند.

چگونگی انجام کار

این بخشی از جذابیت حسابرسی است؛ یعنی یادگیری درباره سیستم‌ها و فناوری‌های جدید. بهترین راهکارها شامل مطالعه **مستندات فنی** و انجام **مصاحبه با مدیر سیستم** خواهد بود. کار را با جستجو در اینترنت برای یافتن منابع بالقوه و پرسیدن از مدیر سیستم درباره منابع پیشنهادی شروع کنید (در واقع، او ممکن است کتابچه‌های راهنمای سیستم را برای امانت دادن به شما داشته باشد.(

زمانی را صرف مطالعه آن منابع کنید و بر دو مورد تمرکز نمایید:

۱ .هدف تجاری که این فناوری به دستیابی به آن کمک می‌کند و اصول اولیه نحوه عملکرد آن.

۲ .ریسک‌های فناوری و نقاط کنترلی کلیدی درونی آن.

نقاط کنترلی کلیدی می‌تواند شامل نحوه ایجاد حساب‌ها، چگونگی اعطای مجوز به منابع و پارامترهای پیکربندی باشد که مواردی نظیر تنظیمات گذرواژه و امنیت سرویس‌های شبکه را تعیین می‌کنند. زمانی که شروع به شکل‌دهی ایده‌ها و پرسش‌های خود کردید، با مدیر سیستم جلسه بگذارید تا درک خود را اعتبارسنجی کرده و پاسخ پرسش‌های باقی‌مانده خود را دریافت کنید.

بخش‌های «چارچوب‌های تعمیم‌یافته» و «بهترین تجارب» را از قسمت‌های پیشین این فصل مرور کنید تا ایده‌های بیشتری برای مراحل آزمون متناسب با فناوری تحت بررسی به دست آورید.

پس از اتمام فعالیت‌های قبلی، هر مرحله آزمونی را که می‌خواهید به لیست موجود در این فصل اضافه کنید، مشخص نمایید.

۲ .اطلاعات پایه‌ای سیستم (مانند شماره نسخه، آخرین سرویس‌پک نصب‌شده، معماری کلی) را برای فناوری مورد بررسی به دست آورید.

این اطلاعات توسط حسابرس جهت کمک به تفسیر نتایج مراحل بعدی حسابرسی استفاده خواهد شد.

چگونگی ؟!

برای کسب این اطلاعات با مدیر سیستم همکاری کنید.

مدیریت حساب‌های کاربری

۳ .رویه‌ها و روال‌های ایجاد حساب را بررسی و ارزیابی کنید تا اطمینان یابید که حساب‌ها تنها در صورت وجود نیاز تجاری مشروع ایجاد می‌شوند. مطمئن شوید که هر حساب به یک کارمند مشخص اختصاص دارد و به راحتی به او قابل ردیابی است. همچنین فرآیندهای مربوط به اطمینان از حذف یا غیرفعال‌سازی به‌موقع حساب‌ها را در صورت قطع همکاری یا تغییر شغل بررسی و ارزیابی کنید.

کنترل‌های مؤثر باید بر ایجاد و حذف حساب حاکم باشند. کنترل‌های نامناسب یا فقدان آن‌ها می‌تواند منجر به دسترسی غیرضروری به منابع سیستم شده و یکپارچگی و در دسترس بودن داده‌ها و فرآیندهای حساس را به خطر اندازد.

اگر مالک یک حساب به وضوح مشخص نباشد، فرآیند تحقیقات جرم‌شناسی (فارنزیک) در خصوص اقدامات نامناسب انجام‌شده توسط آن حساب با مشکل مواجه خواهد شد. اگر چندین نفر از یک حساب استفاده کنند، هیچ‌گونه مسئولیت‌پذیری و پاسخگویی برای اقدامات صورت‌گرفته توسط آن حساب قابل اثبات نخواهد بود.

روش‌های اجرایی حسابرسی دسترسی‌ها، گذرواژه‌ها و سطوح مدیریتی چگونگی (بخش مدیریت حساب‌ها)؟!

با مدیر سیستم مصاحبه کرده و رویه‌های ایجاد حساب را بازبینی کنید. این فرآیند باید شامل شکلی از راستی‌آزمایی باشد که نشان دهد کاربر نیازی مشروع برای دسترسی دارد. نمونه‌ای از حساب‌ها را انتخاب کرده و مستنداتی را بررسی کنید که نشان دهد پیش از ایجاد، به درستی تایید شده‌اند. در روشی جایگزین، می‌توانید نمونه‌ای از حساب‌ها را برداشته و با تحقیق و درکِ وظایف شغلیِ صاحبان حساب، مشروعیت آن‌ها را اعتبارسنجی کنید.

فرآیند حذف حساب‌ها را در زمانی که دیگر به دسترسی نیازی نیست، بازبینی کنید. این فرآیند می‌تواند شامل یک روند خودکار باشد که توسط بخش منابع انسانی (HR) شرکت و با ارائه اطلاعات مربوط به قطع همکاری یا تغییرات شغلی هدایت می‌شود؛ یا می‌تواند شامل بازبینی و تایید دوره‌ای حساب‌های فعال توسط مدیر سیستم و یا سایر مدیران آگاه باشد. نمونه‌ای از حساب‌ها را تهیه کرده و تایید کنید که متعلق به کارکنان فعال هستند و هر کارمند نیازمندیِ تجاریِ مشروعی برای دسترسی دارد.

لیست حساب‌ها را بازبینی کنید و مطمئن شوید که انتساب هر حساب به یک کارمند واحد به راحتی امکان‌پذیر است. هرگونه حسابی را که به نظر می‌رسد اشتراکی باشد (مانند حساب‌های مهمان یا حساب‌های اپلیکیشن) مورد پرسش قرار دهید. اگر چنین حساب‌هایی مورد نیاز هستند، تعیین کنید که استفاده از آن‌ها چگونه کنترل شده و مسئولیت‌پذیری در قبال آن‌ها چگونه حفظ می‌شود.

۴ .تایید کنید که کنترل‌های مناسب برای گذرواژه و احراز هویت برقرار هستند. همچنین، تعیین کنید که آیا گذرواژه‌های پیش‌فرض حساب‌ها تغییر یافته‌اند یا خیر .مناسب بودنِ گذرواژه و سایر کنترل‌های احراز هویت، به حساسیت داده‌ها و فرآیندهای مدیریت‌شده در سیستمِ تحتِ بررسی بستگی دارد. کنترل‌های بیش از حد ضعیف، سیستم را در معرض نفوذ قرار می‌دهند و گذرواژه‌های بیش از حد سخت‌گیرانه می‌توانند سربار غیرضروری بر استفاده از سیستم تحمیل کنند.

بسیاری از سیستم‌ها، به‌ویژه سیستم‌های خریداری شده، دارای حساب‌های پیش‌فرض با گذرواژه‌های پیش‌فرضِ شناخته‌شده هستند. بسیاری از این حساب‌های پیش‌فرض برای مدیریت سیستم استفاده می‌شوند و بنابراین دارای دسترسی‌های سطح بالا هستند. اگر آن گذرواژه‌های پیش‌فرض تغییر نکنند، دسترسی کاربران غیرمجاز به اپلیکیشن آسان خواهد بود.

چگونگی

تعیین کنید تنظیمات گذرواژه در فناوری مورد بررسی، در کجا کنترل می‌شوند. با کمک مدیر سیستم، آن تنظیمات را بازبینی کرده و با سیاست‌های شرکت مقایسه کنید. کنترل‌هایی مانند طول عمر گذرواژه، طول، پیچیدگی، تاریخچه، زمان اتمام جلسه (Timeout)و سیاست‌های قفل شدن حساب (Lockout) را مد نظر قرار دهید.

نیاز به فرم‌های قوی‌ترِ احراز هویت (مانند احراز هویت دومرحله‌ای) را از طریق گفتگو با مدیر سیستم و کاربرانِ اصلی فناوریِ مورد بررسی، کاوش کنید.

با کمک مدیر سیستم و از طریق بازبینی مستندات سیستم و جستجوی اینترنتی، تعیین کنید که آیا حساب‌ها و گذرواژه‌های پیش‌فرض وجود دارند یا خیر. اگر وجود دارند، یکی از ساده‌ترین راه‌ها برای تعیین اینکه آیا تغییر کرده‌اند یا نه، تلاش برای ورود (Logon) با استفاده از حساب‌ها و گذرواژه‌های پیش‌فرض است (هرچند احتمالاً بهتر است از مدیر اپلیکیشن بخواهید که این کار را انجام دهد.(

۵ .اطمینان حاصل کنید که دسترسی مدیر سیستم (Administrator) به سیستم به طور مناسب کنترل می‌شود .یک حساب کاربری و یا عملکرد مدیریتی باید وجود داشته باشد تا به مدیریت کاربران، داده‌ها و فرآیندهای درون سیستم مورد بررسی کمک کند. این

حساب یا قابلیت باید به شدت کنترل شود تا از سوءاستفاده و اختلال در خدمات‌رسانی به سایر کاربران جلوگیری شود.

چگونگی

از طریق بازبینی مستندات و مصاحبه با مدیر سیستم، نحوه عملکردِ بخشِ مدیریت سیستم در آن فناوری را تعیین کنید. لیستی از تمام کارکنانی که سطح دسترسی مدیریتی به آن‌ها اعطا شده تهیه کرده و مناسب بودن هر یک را بررسی کنید.

مدیریت مجوزها

۶. مکانیزم مجوزدهی (Authorization) سیستم را بازبینی کنید تا درک کنید چگونه به کاربران دسترسی به منابع حساس (مانند تراکنش‌ها و داده‌ها) اعطا می‌شود. مجوزهای منابع حساس و همچنین فرآیندهای اعطای دسترسی به آن منابع را بررسی نمایید.

به کارکنان باید تنها همان میزان دسترسی به سیستم داده شود که برای انجام وظایف شغلی‌شان ضروری است. اگر منابع حیاتی به درستی محافظت نشوند (یعنی دسترسی‌های غیرضروری و بیش از حد فراهم شود)، این امر می‌تواند منجر به افشای نامناسب یا تغییر داده‌های حساس یا ایجاد اختلال در سیستم گردد.

چگونگی

از طریق بازبینی مستندات فنی و مصاحبه با مدیر سیستم و کاربران کلیدی فناوری، منابعی (مانند فایل‌ها، درایوهای اشتراکی، تراکنش‌ها) را که در سیستم از همه حیاتی‌تر هستند، شناسایی کنید. تعیین کنید که آیا این منابع به طور مناسب ایمن شده‌اند (دسترسی تنها به افرادی که نیاز دارند محدود شده است) و آیا فرآیندهای مناسب برای اعطا و ابطال دسترسی به آن منابع وجود دارد یا خیر.

۷. ارزیابی استفاده از رمزنگاری

نیاز به رمزنگاری اغلب توسط سیاست‌ها، مقررات، میزان حساسیت شبکه یا حساسیت داده‌ها تعیین می‌شود. در صورت امکان، باید از تکنیک‌های رمزنگاری برای گذرواژه‌ها و سایر داده‌های محرمانه‌ای که در سطح شبکه ارسال می‌شوند، استفاده کرد. این کار مانع

از آن می‌شود که سایر افراد حاضر در شبکه اقدام به «شنود (Sniffing)» و ربودن این اطلاعات کنند. برای داده‌های حساس مانند گذرواژه‌ها، رمزنگاری باید در حالت سکون (در فضای ذخیره‌سازی) نیز در نظر گرفته شود. این موضوع به‌ویژه برای داده‌هایی که خارج از محیط شرکت شما ذخیره می‌شوند، بسیار حائز اهمیت است.

چگونگی (در حوزه رمزنگاری)

سیستم را به همراه مدیر سیستم بازبینی کنید تا وجود رمزنگاری در موارد مقتضی را مورد ارزیابی قرار دهید.

امنیت شبکه و کنترل‌های آن

۸. تعیین کنید چه سرویس‌های شبکه‌ای روی سیستم فعال هستند و ضرورت آن‌ها را با مدیر سیستم تایید نمایید. برای سرویس‌های ضروری، رویه‌ها و روال‌های مربوط به ارزیابی آسیب‌پذیری‌های مرتبط با آن سرویس‌ها و به‌روزرسانی (Patching) آن‌ها را بررسی و ارزیابی کنید.

هر زمان که دسترسی از راه دور مجاز باشد (یعنی هر زمان که یک سرویس شبکه فعال شود)، یک بردار حمله بالقوه جدید ایجاد می‌شود که خطر ورود غیرمجاز به سیستم را افزایش می‌دهد. بنابراین، سرویس‌های شبکه تنها زمانی باید فعال شوند که نیاز تجاری مشروع برای آن‌ها وجود داشته باشد.

حفره‌های امنیتی جدید به تناوب برای اکثر پلتفرم‌های فنی کشف و گزارش می‌شوند. اگر مدیر سیستم از این هشدارها آگاه نباشد و وصله‌های امنیتی را نصب نکند، ممکن است حفره‌های امنیتی شناخته‌شده‌ای در سیستم وجود داشته باشند که مسیری را برای نفوذ به سیستم فراهم کنند.

چگونگی

از طریق بازبینی مستندات فنی و مصاحبه با مدیر سیستم، تعیین کنید چه سرویس‌هایی فعال هستند. پس از به دست آوردن لیست سرویس‌های فعال، موارد لیست را با مدیر سیستم بررسی کنید تا نیاز به هر سرویس را درک نمایید. برای هر سرویسی که مورد نیاز نیست، مدیر سیستم را به غیرفعال کردن آن ترغیب کنید.

فرآیند مورد استفاده برای مطلع ماندن از آسیب‌پذیری‌های جدیدِ سرویس‌های فعال و نحوه دریافت و اعمال وصله‌ها برای رفع آن آسیب‌پذیری‌ها را درک و ارزیابی کنید. اطلاعات مربوط به این فرآیند را می‌توان از طریق مصاحبه و بازبینی مستندات جمع‌آوری کرد. بر اساس تحقیق و مصاحبه‌های خود، ممکن است تشخیص دهید که برخی سرویس‌ها باید به شیوه‌ای خاص پیکربندی شوند تا به صورت ایمن فعال باشند. در صورت لزوم، پیکربندی سرویس‌های فعال را بازبینی کنید.

۹ .در صورت امکان، یک ابزار پویش (اسکن) آسیب‌پذیری شبکه را برای بررسی آسیب‌پذیری‌های فعلی در محیط اجرا کنید.

این کار، نمایی لحظه‌ای (Snapshot) از سطح امنیت فعلی سیستم را از منظر سرویس‌های شبکه ارائه می‌دهد. دنیای آسیب‌پذیری‌های شبکه همواره در حال تغییر است و ایجاد یک برنامه حسابرسی ایستا که تصویری به‌روز از آسیب‌پذیری‌های قابل بررسی ارائه دهد، غیرواقع‌بینانه است. بنابراین، ابزار پویشی که به طور مکرر به‌روزرسانی می‌شود، واقع‌بینانه‌ترین مکانیزم برای درک وضعیت امنیتی فعلی ماشین (سیستم) است. علاوه بر این، اگر مدیر سیستم فرآیندی برای نصب وصله‌های امنیتی در نظر گرفته باشد، این پویش، کارآمدی آن فرآیند (یا اینکه آیا واقعاً در حال اجراست یا خیر) را اعتبارسنجی می‌کند.

چگونگی

از طریق تحقیق فنی و مصاحبه با مدیر سیستم، تعیین کنید که آیا ابزار پویش آسیب‌پذیری شبکه‌ای وجود دارد که برای فناوری مورد بررسی مناسب باشد یا خیر. در صورت وجود، اجرای آن ابزار را با مدیر سیستم هماهنگ کرده و نتایج را بازبینی کنید.

نکته :با وجود اینکه بسیاری از این ابزارها به گونه‌ای طراحی شده‌اند که مخل فعالیت سیستم نباشند و نیازی به دسترسی به سیستم ندارند، همیشه باید پرسنل فناوری اطلاعات مربوطه (مانند مدیر سیستم، تیم شبکه و امنیت اطلاعات) را از تصمیم خود برای اجرای ابزار مطلع کنید، تاییدیه آن‌ها را بگیرید و زمانی را برای اجرای ابزار با آن‌ها هماهنگ نمایید. ابزارهای پویش می‌توانند به شکلی غیرمنتظره با سیستم تعامل داشته باشند و باعث اختلال شوند، بنابراین آگاهی دیگران از فعالیت‌های شما اهمیت زیادی دارد.

این ابزارها معمولاً باید در حالت «ایمن» (غیرمخرب) اجرا شوند، به طوری که تلاشی برای سوءاستفاده (Exploit) از آسیب‌پذیری‌های کشف‌شده انجام ندهند. در موارد نادری ممکن است بخواهید برای به دست آوردن نتایج دقیق‌تر، یک اکسپلویت واقعی را اجرا کنید؛ اما این کار تنها باید با جلب موافقت و هماهنگی کامل مالک سیستم و مدیر آن انجام شود.

پایش امنیتی و سایر کنترل‌های عمومی در حسابرسی سیستم‌ها

۱۰. اطمینان حاصل کنید که سیستم دارای لاگ‌های حسابرسی (Audit Logs) است که مطابق سیاست‌های سازمان شما ثبت می‌شوند.

لاگ‌های حسابرسی در پس‌آیند یک رخداد، شواهد لازم را فراهم کرده و به عیب‌یابی مسائل عملیاتی و امنیتی کمک می‌کنند.

چگونگی از طریق بازبینی مستندات فنی و مصاحبه با مدیر سیستم، قابلیت‌های ثبت وقایع (Logging) سیستم را تعیین کرده و فعال بودن آن لاگ‌ها را بررسی کنید. همچنین، امنیت و مدت‌زمان نگهداری (Retention) لاگ‌های حسابرسی را ارزیابی نمایید.

۱۱. رویه‌ها و روال‌های مدیر سیستم را برای پایش و حفظ وضعیت امنیتی سیستم بررسی و ارزیابی کنید.

اگر مدیر سیستم فرآیندهایی برای انجام پایش امنیتی نداشته باشد، ممکن است حفره‌های امنیتی ایجاد شوند و حوادث امنیتی بدون آگاهی او رخ دهند.

چگونگی

با مدیر سیستم مصاحبه کرده و هرگونه مستندات مرتبط را بررسی کنید تا درکی از شیوه‌های پایش امنیتی به دست آورید. این کار می‌تواند برای مثال شامل پویش‌های روتین و رفع آسیب‌پذیری‌های شناخته‌شده و یا ارسال هشدارها و بررسی آن‌ها در زمان انجام فعالیت‌های کلیدی در سیستم باشد. وجود سطحی از پایش اهمیت دارد، اما سطح پایشِ مورد نیاز باید با میزان حساسیت سیستم و ریسک ذاتی محیط متناسب باشد.

اگر پایش امنیتی انجام می‌شود، فراوانی (فرکانس) پایش و کیفیت انجام آن را ارزیابی کنید. به دنبال شواهدی باشید که نشان دهد ابزارهای پایش امنیتی واقعاً مورد استفاده قرار می‌گیرند. نتایج اخیر را بازبینی کنید و تعیین کنید که آیا موارد استثنا مورد بررسی و رفع قرار گرفته‌اند یا خیر. از نتایج سایر بخش‌های حسابرسی در انجام این ارزیابی بهره بگیرید؛ برای مثال، اگر در حوزه‌ای که ظاهراً پایش می‌شده، مسائل قابل توجهی پیدا کردید، این موضوع می‌تواند به تردید درباره اثربخشی آن پایش منجر شود.

۱۲. تأیید کنید که سیاست‌ها و رویه‌هایی برای شناسایی زمان انتشار وصله‌ها و همچنین ارزیابی و اعمال وصله‌های کاربردی در نظر گرفته شده است. اطمینان حاصل کنید که تمامی وصله‌های تأیید شده، مطابق با سیاست سازمان شما نصب شده‌اند.

اکثر تأمین‌کنندگان (Vendors) برای محصولات خود برنامه‌های زمانی منظمی جهت انتشار وصله‌ها دارند. اگر این وصله‌ها نصب نشوند، ممکن است آسیب‌پذیری‌های امنیتی شناخته‌شده یا مشکلات عملکردی بحرانی در سیستم باقی بماند. توجه داشته باشید که پیش‌تر در بخش «امنیت شبکه و کنترل‌های آن»، موضوع به‌روزرسانی سرویس‌های شبکه را بررسی کردیم؛ این مرحله به تمامی وصله‌های دیگری اشاره دارد که ممکن است برای خودِ محصول اصلی (هسته سیستم) منتشر شود.

چگونگی

با مدیر سیستم مصاحبه کنید تا مشخص شود چه کسی توصیه‌نامه‌های امنیتی (Advisories)تأمین‌کنندگان را بازبینی می‌کند، چه گام‌هایی برای آماده‌سازی وصله‌ها برداشته می‌شود و وصله‌ها پیش از اعمال در محیط عملیاتی(Production) ، چه مدت آزمایش می‌شوند. درخواست کنید تا یادداشت‌های مربوط به چرخه قبلی به‌روزرسانی را بازبینی کنید. وصله‌های در دسترس را با وصله‌های اعمال شده در سیستم مقایسه نمایید. با مدیر سیستم درباره اقداماتی که برای کاهش ریسک‌های احتمالی در صورت عدم نصب به‌موقع وصله‌ها انجام شده است، گفتگو کنید.

۱۳. مراحل مندرج در فصل ۵ را تا جایی که به سیستم مورد حسابرسی شما مربوط می‌شود، اجرا کنید.

علاوه بر حسابرسی امنیت منطقی سیستم، باید اطمینان حاصل کنید که کنترل‌های فیزیکی و عملیاتی مناسب برای تأمین حفاظت و پایداری (Availability) سیستم برقرار است.

چگونگی

به مراحل ذکر شده در فصل ۵ رجوع کرده و مواردی را که با سیستم تحت حسابرسی مرتبط هستند، اجرا نمایید. به عنوان نمونه، موضوعات زیر احتمالاً با کار شما مرتبط خواهند بود:

- امنیت فیزیکی(Physical security)
- کنترل‌های محیطی(Environmental controls)
- برنامه‌ریزی ظرفیت(Capacity planning)
- مدیریت تغییرات(Change management)
- پایش سیستم(System monitoring)
- فرآیندهای پشتیبان‌گیری(Backup processes)
- برنامه‌ریزی بازیابی از فاجعه(Disaster recovery planning)

چک‌لیست‌های مرجع(MASTER CHECKLISTS)

جداول زیر خلاصه‌ای از مراحل ذکر شده در اینجا برای حسابرسی فناوری‌های نوین و نوظهور هستند.

چک‌لیست حسابرسی مراحل اولیه

۱. **تحقیق درباره فناوری مورد بررسی** جهت یادگیری جزئیات نحوه عملکرد و نقاط کنترلی کلیدی آن. مراحل حسابرسی را در صورت لزوم به برنامه حسابرسی ارائه‌شده در این فصل اضافه کنید.

۲ .کسب اطلاعات پایه سیستم)مانند شماره نسخه، آخرین سرویس‌پک نصب‌شده، معماری کلی) برای فناوری مورد بررسی.

چک‌لیست حسابرسی مدیریت حساب‌های کاربری

۳ .بررسی و ارزیابی رویه‌های ایجاد حساب و اطمینان از اینکه حساب‌ها تنها در صورت وجود نیاز تجاری مشروع ایجاد می‌شوند. اطمینان حاصل کنید که هر حساب با یک کارمند مشخص مرتبط است و به راحتی به او قابل ردیابی می‌باشد. همچنین فرآیندهای مربوط به اطمینان از حذف یا غیرفعال‌سازی به‌موقع حساب‌ها را در صورت قطع همکاری یا تغییر شغل، بررسی و ارزیابی کنید.

۴ .تایید برقراری کنترل‌های مناسب گذرواژه و احراز هویت .همچنین، تعیین کنید که آیا گذرواژه‌های پیش‌فرض حساب‌ها تغییر یافته‌اند یا خیر.

۵ .اطمینان از کنترل مناسب دسترسی مدیر سیستم (Administrator) به سیستم.

چک‌لیست حسابرسی مدیریت مجوزها

۶ .بررسی مکانیزم مجوزدهی (Authorization) سیستم برای درک چگونگی اعطای دسترسی به کاربران برای منابع حساس (مانند تراکنش‌ها و داده‌ها). مجوزهای منابع حساس و همچنین فرآیندهای اعطای دسترسی به آن منابع را بازبینی کنید.

۷ .ارزیابی استفاده از رمزنگاری.(Encryption)

چک‌لیست حسابرسی امنیت و کنترل‌های شبکه

۸ .تعیین سرویس‌های شبکه فعال روی سیستم و تایید ضرورت آن‌ها با مدیر سیستم. برای سرویس‌های ضروری، رویه‌ها و روال‌های مربوط به ارزیابی آسیب‌پذیری‌های مرتبط با آن سرویس‌ها و به‌روزرسانی (Patching) آن‌ها را بررسی و ارزیابی کنید.

۹. **در صورت امکان، اجرای یک ابزار پویش (اسکن) آسیب‌پذیری شبکه** برای بررسی آسیب‌پذیری‌های فعلی در محیط.

چک‌لیست حسابرسی پایش امنیتی و سایر کنترل‌های عمومی

۱۰. **اطمینان از ثبت لاگ‌های حسابرسی (Audit Logs)** مطابق با سیاست‌های سازمان شما.

۱۱. **بررسی و ارزیابی رویه‌های مدیر سیستم** برای پایش و حفظ وضعیت امنیتی سیستم.

۱۲. **تایید وجود سیاست‌ها و رویه‌ها برای شناسایی زمان انتشار وصله‌ها** و همچنین ارزیابی و اعمال وصله‌های کاربردی. اطمینان حاصل کنید که تمامی وصله‌های تایید شده، مطابق با سیاست شما نصب شده‌اند.

۱۳. **اجرای مراحل مندرج در فصل ۵** تا جایی که به سیستم مورد حسابرسی شما مربوط می‌شود.