

# آزمایشگاه امنیت

## نرم افزار و سیستم عامل



## فهرست

|    |   |
|----|---|
| ۳  | مقدمه.....  |
| ۵  | فصل اول: تهدیدات و آسیب پذیریهای نرم افزارها.....               |
| ۸  | فصل دوم: استانداردهای امنیت نرم افزار.....                      |
| ۱۱ | فصل سوم: آزمایشگاه امنیت نرم افزار و سیستم عامل.....            |
| ۱۴ | فصل چهارم: متدولوژی ایجاد آزمایشگاه نرم افزار و سیستم عامل..... |
| ۲۱ | فصل پنجم: ملاحظات پدافند غیر عامل.....                          |

## مقدمه

گسترش استفاده از فضای تبادل اطلاعات در کشور طی سال‌های گذشته و برقراری ارتباط از طریق وب، موجب افزایش بکارگیری فناوری اطلاعات و وابستگی نهادهای مختلف به این پدیده گردیده است. در کنار این توسعه، نگرانی در مورد امنیت اطلاعات نیز متأثر از فناوری جدید شده است. امروزه بخش قابل توجهی از نگرانی‌های امنیتی در سازمان‌ها در حوزه امنیت شبکه و نرم‌افزار و به‌طور کلی امنیتی فناوری اطلاعات و ارتباطات می‌باشد. گرچه در بعد امنیت شبکه فعالیت‌های زیادی در داخل کشور و همچنین در خارج انجام شده است، ولی دیدگاه امنیتی به تولید نرم‌افزار و استفاده از آن کمتر مورد توجه قرار گرفته است. این دیدگاه منجر به این واقعیت شده است که درصد بالایی از آسیب‌پذیری‌های فضای تبادل اطلاعات در زمینه نرم‌افزار و برنامه کاربردی باشد.

تقابل نظام جمهوری اسلامی ایران با استکبار جهانی، رویکرد استفاده حداکثری از توان داخل کشور برای ایجاد سامانه‌های نرم‌افزاری را اجتناب‌ناپذیر می‌گرداند. در این راستا تمرکز بر حفظ امنیت سامانه‌های نرم‌افزاری و بطور خاص نرم‌افزارهای مورد استفاده در سامانه‌ها و مراکز حیاتی ضروری است. بدیهی است بمنظور نیل به ضریب قابل قبولی از امنیت در این نرم‌افزارها، نیازمند پیش‌بینی مکانیسم‌های مناسبی جهت ارزیابی امنیتی نرم‌افزارهای مورد استفاده در حوزه فناوری اطلاعات و ارتباطات هستیم.



# فصل ۱

## تهدیدات و آسیب پذیری های نرم افزارها

تهدیدات پیش روی برنامه های کاربردی بسیار متنوع هستند، بنابراین برای شناخت بهتر ابتدا باید آن ها را بر طبق استاندارد طبقه بندی کرد. OWASP<sup>۱</sup> تهدیدات و آسیب پذیری ها را با توجه به پارامترهای نوع آسیب پذیری، دوره بروز آن، تأثیرات سوء استفاده از آن، بستر آسیب پذیری، منابع لازم برای حمله، شدت مخاطره و روش های تشخیص آن بررسی کرده و مدل خود را ارائه داده است. تهدیدات مورد اشاره در مدل OWASP در جدول صفحه بعد ارائه گردیده است. کنسرسیوم امنیت برنامه های کاربردی مبتنی بر وب<sup>۲</sup> نیز، تهدیدات را در دسته بندی دیگری، تقریباً مشابه OWASP، ارائه نموده است. مدل STRIDE شرکت مایکروسافت نیز یکی دیگر از استانداردهای شایع و مورد استفاده است که تهدیدات را در ۶ دسته جعل

---

<sup>۱</sup>Open Web Application Security Project

<sup>۲</sup>Web Application Security Consortium(WASC)

هویت، دستکاری داده‌ها، انکارپذیری، افشای اطلاعات، انکار سرویس و تجاوز از حقوق دسترسی طبقه‌بندی کرده است.

### آسیب پذیری ها در مدل OWASP

| توصیف کلان   | آسیب پذیری                                      |
|--|---|
| این آسیب‌پذیری زمانی اتفاق می‌افتد که برنامه کاربردی داده‌های فراهم شده از سمت کاربر را بدون بررسی صحت محتوا و یا encode کردن آن به سمت مرورگر بازگرداند. این آسیب‌پذیری به مهاجم اجازه می‌دهد که اسکریپت خود را اجرا کند.   | XSS   |
| آسیب‌پذیری زمانی اتفاق می‌افتد که داده‌های ارسالی از طرف کاربر به عنوان قسمتی از یک فرمان به سمت یک مفسر (مثل پایگاه داده) فرستاده شود. با استفاده از این آسیب‌پذیری مهاجم می‌تواند فرامین مورد نیاز خود را اجرا کند.  | Injection Flaws                                 |
| ارجاع مستقیم به یک شیء زمانی اتفاق می‌افتد که توسعه دهنده <sup>۱</sup> ارجاع به یک شیء داخلی پیاده‌سازی شده (مثل یک فایل، دایرکتوری و ... ) را از طریق پارامترهای یک URL در دید کاربر قرار دهد. مهاجم با دستکاری این داده‌ها دسترسی غیر مجازی را به شیء‌های مورد نظر خواهد داشت. | Insecure Direct Object Reference                |
| CSRF باعث می‌شود از طرف کاربری که در یک برنامه کاربردی تحت وب log-in کرده است، عمل غیرمجازی بدون آگاهی کاربر و در جهت اهداف مهاجم انجام پذیرد.   | Cross Site Request Forgery(CSRF)                |
| یک برنامه‌ی کاربردی تحت وب ممکن است بطور ناخواسته اطلاعاتی در مورد فایل‌های پیکربندی عملکرد داخلی را بخاطر مشکلات داخلی خود بروز دهد. مهاجم با استفاده از این آسیب‌پذیری اطلاعات حساسی در مورد وضعیت برنامه کاربردی بدست می‌آورند.   | Information Leakage And Improper Error Handling |

<sup>۱</sup> Developer

|  |  |
|--|--|
| <p>از داده‌های مورد استفاده در مورد وضعیت نشست و یا تصدیق هویت دسترسی معمولاً بصورت مناسب حفاظت نمی‌شود. مهاجم با استفاده از این آسیب‌پذیری می‌تواند کلمه‌های عبور، کلیدها و دیگر اطلاعات مهم تصدیق هویت را کشف کرده و با استفاده از آن خودش را به عنوان کاربر مجاز معرفی نماید.</p>               | <p><b>Broken Authentication And Session Management</b></p> |
| <p>برنامه‌های کاربردی اغلب در رمزنگاری جریان‌های اطلاعاتی دچار مشکل می‌شوند. از SSL باید در تمام ارتباطات تصدیق هویت شده استفاده کرد. در غیر اینصورت برنامه‌ی کاربردی اطلاعات حساس مربوط به تصدیق هویت و توکن‌های نشست را لو می‌دهند. علاوه بر اینها اطلاعات حساس دیگر نیز باید رمزنگاری شوند.</p> | <p><b>Insecure Cryptographic Storage</b></p>               |
| <p>برنامه‌های کاربردی در مواقعی که لازم است ترافیک شبکه را رمزنگاری کنند با شکست مواجه می‌شوند و بخاطر این موضوع با استراق سمع ترافیک شبکه می‌توان اطلاعات حساسی را بدست آورد.</p>   | <p><b>Insecure Communication</b></p>                       |
| <p>در اغلب موارد تنها راهکار حفاظتی برای عدم دسترسی به یک URL، نشان ندادن صفحه‌ی مربوطه است. در بعضی موارد یک Attacker با سطح مهارت بالا با استفاده از ترفندهایی می‌تواند به این صفحات دسترسی پیدا کند.</p>  | <p><b>Failure To Restrict URL Access</b></p>               |

## فصل ۲

### استانداردهای امنیت نرم افزار

طبق تحقیقات مؤسسه استاندارد و فناوری، حذف تنها ۵۰ درصد آسیب پذیری ها در حین فرآیند تولید نرم افزار، باعث کاهش ۷۵ درصدی وصله های<sup>۱</sup> نرم افزاری می گردد. بنابراین بمنظور کاهش هزینه آزمون و مطابقت با استانداردهای امنیتی، توجه به آسیب پذیری ها در فرآیند تولید نرم افزار و تولید مستندات امنیتی و سپس تطابق آن با استانداردهای موجود، بهترین و مناسب ترین راه حل برای اطمینان از امنیت نرم افزار است.

سازمان های مختلفی در زمینه امنیت نرم افزار تحقیق می نمایند، ولی هنوز نمی توان از یک استاندارد معتبر جهانی در حوزه امنیت نرم افزار نام برد. قطعاً اهداف تولید این استانداردها یکسان نبوده و همگی آنان تمامی قسمت های چارچوب مرجع امنیت نرم افزار را پوشش نمی دهند. برخی از این استانداردها عبارتند از:

---

<sup>۱</sup>Patch



🚩 SAMATE<sup>1</sup>:

بمنظور آزمون نفوذ به سیستم کاربرد دارد.

🚩 OWASP

در زمینه امنیت برنامه‌های کاربردی تحت وب متمرکز است.

🚩 CLASP

یک متدولوژی جهت تولید نرم‌افزار بصورت امن است.

🚩 BSI<sup>2</sup>

شامل یکی از بهترین تجربیات موجود در زمینه امنیت نرم‌افزار است که دارای منابعی از قبیل ابزارها، قواعد، راهنماها، اصول و معماری است. نتایج تحقیقات جمع‌آوری شده توسط مؤسسه BSI تمامی فازهای چرخه حیات نرم‌افزار را پوشش می‌دهد.

🚩 CC<sup>3</sup>

معیارهای عمومی، یک استاندارد بین‌المللی برای ارزیابی امنیتی محصولات فناوری اطلاعات است که در آن بطور مفصل نیازمندی‌های امنیتی محصولات فناوری اطلاعات مشخص شده است. CCها در یک ساختار سلسله‌مراتبی، نیازمندی‌های امنیتی را مشخص می‌کند. نیازمندی‌های امنیتی در این استاندارد به ۲ گروه زیر تقسیم می‌شوند:

۱. نیازمندی‌های امنیتی عملکردی<sup>4</sup>

---

<sup>1</sup>Software Assurance Metrics And Tool Evaluation

<sup>2</sup>British Standards Institution

<sup>3</sup>Common Criteria

<sup>4</sup>Functional Security Requirements

## ۲. نیازمندی‌های اطمینان بخشی امنیت<sup>۱</sup>

در نیازمندی‌های امنیتی عملکردی، اهداف امنیتی نرم افزار بیان می‌شود. این نیازمندی‌ها در چندین کلاس به شرح ذیل طبقه‌بندی می‌شوند.

- ممیزی امنیت
- ارتباطات
- حمایت از رمزنگاری
- حفاظت از داده کاربر

...

در بخش نیازمندی‌های اطمینان بخشی امنیت، چگونگی اطمینان از برآورده شدن نیازمندی‌های امنیتی لازم در هفت سطح بیان می‌شود.

---

<sup>۱</sup>Assurance Security Requirements

## فصل ۳

### آزمایشگاه امنیت نرم افزار و سیستم عامل

طراحی و تولید برنامه‌های کاربردی به گونه‌ای نیست که پس از تولید بتوان آزمون کامل و جامعی از امنیت آن در تمامی جنبه‌ها به عمل آورد، اگر هم این کار امکان‌پذیر باشد، هزینه آن به اندازه تولید همان برنامه کاربردی خواهد بود. آزمایشگاه‌های معتبر معدودی، برای آزمون برنامه‌های کاربردی در سطح بین‌المللی وجود دارد که به آزمون امنیتی به عنوان فرآیند بعد از تولید نگاه می‌کند. در صورت ادعای آزمون امنیت برنامه‌های کاربردی توسط این آزمایشگاه‌ها، قطعاً این آزمون محدود به یک آزمون نفوذ و حداکثر بازیابی کد خواهد بود. اغلب مؤسساتی که در زمینه امنیت برنامه‌های کاربردی فعال هستند، خدمات خود را به صورت ارائه مشاوره، فراهم آوردن ابزار و مستندات لازم برای طراحی و کدنویسی امن و در نهایت بازیابی کد و آزمون نفوذ در اختیار مشتریان قرار می‌دهند. در ادامه سه گونه کاملاً متفاوت از آزمایشگاه‌های آزمون امنیت را مورد بررسی قرار می‌دهیم.

## 🚩 آزمایشگاه‌های CCTL<sup>۱</sup>

در آزمایشگاه‌های CCTL ارزیابی براساس معیارهای عمومی انجام می‌گیرد و بسیاری از نیازمندی‌های تعریف شده در معیارهای عمومی سطح پایین تر از آن هستند که در یک برنامه کاربردی بتوان از آن‌ها استفاده کرد. معیارهای عمومی در سطح نرم‌افزار برای نرم‌افزارهای سیستمی و کارگزار<sup>۲</sup> مناسب می‌باشند.

## 🚩 آزمایشگاه Ounce

آزمایشگاه Ounce با تعریف خاص خود از فرآیند اطمینان‌بخشی از امنیت نرم‌افزار و تهیه چارچوب امنیتی خود برای ارزیابی امنیتی برنامه‌های کاربردی، ادعای توانایی درگیر شدن در هر نقطه از فرآیند تولید نرم‌افزار، برای ارزیابی و اطمینان از امنیت نرم‌افزار را دارد. هرچند شواهد کافی برای این امر وجود ندارد. قطعاً این آزمایشگاه بیشتر به عنوان یک مشاور برای آموزش به تیم تولید نرم‌افزار عمل می‌کند و در صورتی که به ارزیابی امنیتی مقرون به صرفه یک برنامه کاربردی پردازد، این ارزیابی محدود به ارزیابی آسیب‌پذیری‌های آن خواهد بود، و نه یک فرایند جامع اطمینان از امنیت نرم‌افزار.

## 🚩 آزمایشگاه Bug Huntress

آزمایشگاه Bug Huntress نیز با انجام انواع مختلف آزمون که آزمون امنیت یکی از آن‌ها است سعی در اطمینان‌بخشی ذینفعان از کیفیت نرم‌افزار را دارد. آزمون‌هایی که این آزمایشگاه بر روی نرم‌افزارها انجام می‌دهد، شامل طیف وسیعی است که هر کدام جنبه‌ای از پارامترهای لازم برای کیفیت نرم‌افزار را ارزیابی می‌کند.

---

<sup>۱</sup>Common Criteria Testing Laboratory

<sup>۲</sup>Server

با توجه به گزاره‌های فوق به این نتیجه می‌رسیم که آزمون امنیت معنی و مفهوم خاص خود را داشته و آزمون برای برنامه‌های کاربردی که در فرآیند تولید آن‌ها شرکت نداریم، تنها محدود به آزمون آسیب‌پذیری‌های آن می‌باشد.

## وظایف آزمایشگاه امنیت نرم افزار و سیستم عامل

بطور کلی وظایف آزمایشگاه امنیت نرم افزار و سیستم عامل به شرح ذیل تعریف می‌گردد:

- + ارزیابی امنیتی انواع نرم افزارها با سکوها<sup>۱</sup> متفاوت
- + بررسی امنیتی انواع کد منبع با بسترهای متفاوت توسعه
- + ارائه راه حل های امنیتی پس از تشخیص آسیب پذیری‌ها در نرم‌افزار یا کد منبع
- + ارائه متدولوژی ارزیابی امنیتی در سطح کد منبع و یا نرم‌افزار
- + ارزیابی امنیتی انواع پایگاه داده مانند SQL Server و Oracle و...
- + ارزیابی امنیتی انواع کارگزارهای وب مانند آپاچی و IIS
- + ارزیابی امنیتی انواع سیستم عامل‌ها

---

<sup>۱</sup>Platform

## فصل ۴

### **متدولوژی ایجاد آزمایشگاه نرم افزار و سیستم عامل**

مدیریت سازوکارهای مختلف یک آزمایشگاه از زمان عقد قرارداد تا انجام آزمون و تحویل نتایج و نهایتاً صدور گواهی، جنبه‌های مختلفی دارد و نیازمند یک متدولوژی جامع است تا در آن تمامی تعاملات بین اجزاء مختلف آزمایشگاه و مشتری به‌طور شفاف تعریف شده باشد.

یک آزمایشگاه امنیت نرم افزار و سیستم عامل نیاز به دو بخش طراحی و پیاده سازی دارد. بخش طراحی خود شامل دو بخش فنی و مدیریتی است.

## طراحی آزمایشگاه امنیت نرم افزار و سیستم عامل

متدولوژی طراحی آزمایشگاه امنیت نرم افزار و سیستم عامل شامل دو بخش فنی و مدیریتی می باشد. این دو بخش براساس استانداردهای ISO/IEC ۱۷۰۲۵:۲۰۰۵ و استاندارد مدیریت پروژه PMBOK<sup>۱</sup> شکل گرفته است.

بخش فنی در متدولوژی آزمایشگاه امنیت نرم افزار و سیستم عامل، فرآیند کلی انتخاب موارد آزمون، رویه‌ها، ابزار و قالب خروجی‌های آزمون و سایر مواردی را که برای انجام آزمون مورد نیاز است، مشخص می کند. این بخش شامل موارد زیر است:

✚ شناخت نرم افزار

✚ انتخاب رویه‌های آزمون

✚ انجام آزمون و تهیه گزارش نهایی

رویه‌های تست، شامل روش و دستورالعمل چگونگی انجام تست می باشد. براساس شناختی که از نرم افزار بعمل می آید و توافقی که با کارفرما انجام می شود، می توان تمامی آزمون ها یا بخشی از آن ها را به انجام رساند. در بخش مدیریتی آزمایشگاه کلیه سازوکارهای لازم برای ایجاد و مدیریت آزمایشگاه تعریف می شوند. بنابراین در این بخش موارد زیر باید مدنظر قرار گیرد:

✚ لیست تجهیزات مورد نیاز

✚ توافقی نامه عدم افشای اطلاعات

✚ نقش ها، مسئولیت ها و فعالیت های اصلی در آزمایشگاه

---

<sup>۱</sup>Project Management Body of Knowledge

✚ ساختار سازمانی نقش‌های درگیر در آزمایشگاه

✚ مدیریت منابع انسانی

✚ مدیریت مخاطرات

✚ مدیریت زمان

✚ مدیریت کیفیت

انجام آزمون امنیتی برای ارزیابی مخاطرات امنیتی محصولات نرم‌افزاری مبتنی بر دانش آزمون‌گر بوده و تهیه یک رویه دقیق برای انجام آزمون در این حوزه غیر ممکن است. بنابراین نمی‌توان به ساختار یک آزمایشگاه به عنوان یک سیستم نقش‌گرا که در آن تمامی روال‌ها مشخص گردیده و فقط نقش، جایگاه و مسئولیت سازمانی افراد در آن حائز اهمیت است، نگاه کرد.

مناسب‌ترین رویکرد برای مدیریت یک آزمایشگاه، نگاه وظیفه‌گرا و ماتریسی به آن می‌باشد که در آن می‌توان به هر سامانه که برای آزمون به آزمایشگاه وارد می‌شود، به عنوان یک پروژه جدید نگاه کرد. با توجه به این موضوع بخش مدیریت آزمایشگاه به صورت پروژه محور شکل گرفته و بر این اساس کلیه سازوکارهای لازم برای مدیریت یک پروژه آزمون امنیت نرم‌افزار در آن اندیشیده می‌شود.

همانطور که گفته شد برای طراحی مدیریتی آزمایشگاه استانداردهای PMBOK و ISO ۹۰۰۱ مورد توجه قرار گرفته است. بخشی از پیکره دانش مدیریت پروژه PMBOK که در اغلب پروژه‌ها مورد استفاده قرار می‌گیرد در قالب یک راهنما با عنوان PMBOK Guide ارائه شده است. این راهنما دانش مدیریت پروژه را در ۹ حوزه اصلی دانشی تقسیم می‌نماید. به عبارت دیگر فرآیند



اصلی مدیریت پروژه به ۹ حوزه دانش تقسیم شده و هر حوزه نیز به نوبه خود به چندین فرآیند تقسیم می‌گردد. این حوزه‌ها به شرح ذیل می‌باشد:

+ مدیریت یکپارچه سازی پروژه

+ مدیریت دامنه پروژه

+ مدیریت زمان پروژه

+ مدیریت هزینه پروژه

+ مدیریت کیفیت پروژه

+ مدیریت ارتباطات پروژه

+ مدیریت ریسک پروژه

+ مدیریت روند<sup>۱</sup> پروژه

+ مدیریت منابع انسانی

فرآیندهای حوزه‌های دانشی مختلف به صورت ذیل تعریف شده اند.

+ فرآیندهای آغازین: تشخیص، تدوین و ارائه مراحل و فعالیت‌های لازم

برای شروع پروژه

+ فرآیندهای برنامه‌ریزی: تبیین و تعیین اهداف و انتخاب راهکار بهینه برای

کسب نتایج موفقیت‌آمیز و ایفای کامل تعهدات

+ فرآیندهای اجرایی: مجموعه عملیات هماهنگی بین کلیه ارکان اجرایی

پروژه مطابق برنامه

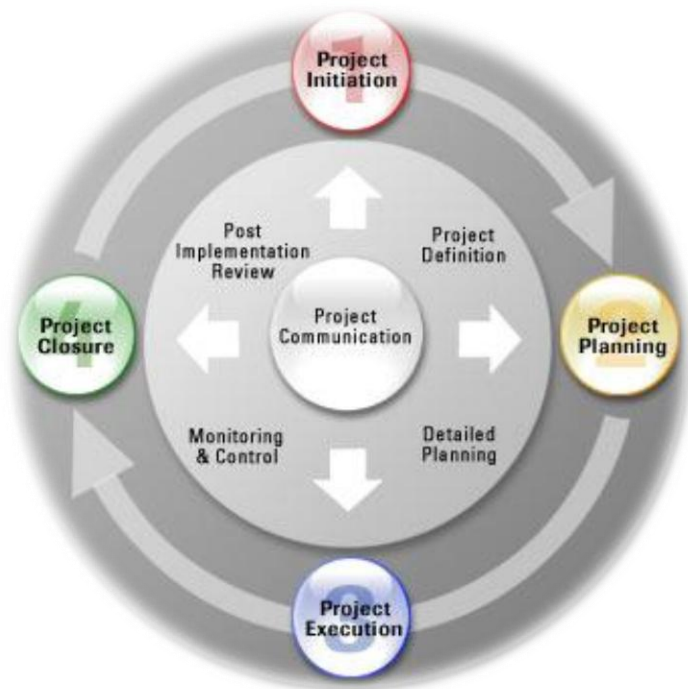
---

<sup>۱</sup>Procurement

✚ فرآیندهای کنترلی: مجموعه فعالیت‌های کسب اطمینان از دستیابی به اهداف پروژه

✚ فرآیندهای اختتامی: مجموعه فعالیت‌های مربوط به اتمام عملیات پروژه، ثبت سوابق و تجربیات و بازنگری کلان بر روند اجرای پروژه

دسته‌بندی پنج‌گانه برای حوزه‌های موجود در PMBOK



## پیاده سازی آزمایشگاه امنیت نرم افزار و سیستم عامل

۱۵۰ NIST Handbook یک استاندارد مدیریتی مناسب برای پیاده سازی آزمایشگاه امنیت می باشد که از ISO۹۰۰۱ نشأت گرفته است. بومی سازی این استاندارد جهت استفاده پس از راه اندازی آزمایشگاه امنیت نرم افزار و سیستم عامل، ضروری بنظر می رسد.

الزامات استاندارد مذکور و نحوه برآورده سازی آنها

| نحوه پیاده سازی  | الزامات   |
|--|---|
| تدوین خط مشی، تدوین نظام نامه ۱۵۰ NIST، تدوین ساختار سازمانی، تدوین شرح وظایف، تدوین روش اجرایی تامین منابع انسانی | سازماندهی سامانه مدیریت                         |
| تدوین روش اجرایی کنترل اسناد و مدارک   | کنترل مدارک                                     |
| تدوین روش اجرایی امکان سنجی و بازنگری قراردادها و درخواست ها   | بازنگری درخواست ها و قراردادها                  |
| تدوین روش اجرایی انتخاب و ارزیابی و کنترل پیمانکاران   | پیمانکاران آزمون و کالیبراسیون                  |
| تدوین روش اجرایی خرید خدمات  | خرید خدمات و تامین کنندگان                      |
| تدوین روش اجرایی ارتباط با مشتریان   | خدمات مشتری                                     |
| تدوین روش اجرایی کنترل خدمات نامنطبق آزمون و کالیبراسیون   | کنترل عدم انطباق های عملیات آزمون و کالیبراسیون |
| تدوین روش اجرایی بهبود   | بهبود   |
| تدوین روش اجرایی ممیزی های داخلی   | ممیزی های داخلی                                 |



## فصل ۵

### ملاحظات پدافند غیر عامل

کلیه فعالیت هایی که در هر کشور انجام می گیرد و یا خدماتی که ارائه می گردد از نظر اهمیت به سه سطح ذیل دسته بندی می شوند:

✚ سطح حیاتی: خدمات و فعالیت هایی هستند که دارای گستره ملی بوده و وجود و استمرار آنها برای کشور حیاتی است و آسیب یا وقفه در آنها بوسیله دشمن باعث اختلال کلی در اداره امور کشور می گردد.

✚ سطح حساس: خدمات و فعالیت هایی هستند که دارای گستره منطقه ای بوده و وجود و استمرار آنها برای مناطقی از کشور ضروری است و آسیب و یا ایجاد وقفه در آنها بوسیله دشمن باعث بروز اختلال در بخشی از کشور می گردد.

✚ سطح مهم: خدمات و فعالیت‌هایی هستند که دارای گستره فعالیت محلی بوده و وجود و استمرار آن‌ها برای بخشی از کشور دارای اهمیت است و آسیب و یا ایجاد وقفه در آن‌ها بوسیله دشمن باعث بروز اختلال در بخشی از کشور می‌گردد.

براین اساس امنیت سامانه‌های ارائه دهنده این خدمات ارزشی معادل سطح اهمیت فعالیت خود پیدا می‌کند و لازم است تا ملاحظات لازم در تولید و ارزیابی آنها لحاظ گردد. در ادامه به ذکر چند رهنمود کلی در این خصوص خواهیم پرداخت.

✚ ملاحظات پدافند غیرعامل در شاخه‌های مختلف نرم‌افزاری می‌بایست بعنوان یکی از ورودی‌های اصلی در طراحی و تولید نرم‌افزارهای بومی مورد توجه قرار گیرد.

✚ آزمایشگاه امنیت نرم‌افزار و سیستم عامل بعنوان واحد ارزیابی، نظارت فرآیند تولید و تأیید امنیت می‌بایست در داخل کشور ایجاد گردد.

✚ لازم است براساس سه سطح حیاتی، حساس و مهم، سطوح ارزیابی امنیتی و نظارت بر چرخه تولید تعریف گردد و رهنمودها منطبق با این سطوح احصاء و آزمون‌ها انجام گیرد.

○ سطح ۱: این سطح مربوط به سامانه‌هایی است که به سرویس‌ها و فعالیت‌های حیاتی کشور مرتبط، می‌باشد.

○ سطح ۲: این سطح مربوط به سامانه‌هایی است که به سرویس‌ها و فعالیت‌های حساس کشور مرتبط می‌باشد.

○ سطح ۳: این سطح مربوط به سامانه‌هایی است که به سرویس‌ها و فعالیت‌های مهم کشور مرتبط می‌باشد.

❖ رویه های آزمون امنیت نرم افزار، پایگاه داده، سیستم عامل و کارگزار وب می بایست بصورت جداگانه در آزمایشگاه امنیت نرم افزار و سیستم عامل مشخص و نوع آزمون های امنیتی مبتنی بر کلاس نیازمندی های امنیتی تعیین شود.

انواع آزمون های امنیتی به صورت ذیل دسته بندی می گردد:

- ✓ آزمون های هویت شناسی و کنترل دسترسی
- ✓ آزمون های رمزنگاری
- ✓ آزمون های انکارناپذیری
- ✓ آزمون های اعتبارسنجی داده ها و بررسی خطاها
- ✓ آزمون های رویدادنگاری
- ✓ آزمون های پیکربندی نرم افزار
- ✓ آزمون های مدیریت نرم افزار
- ✓ آزمون های جامعیت