فصل ۱: مفاهیم قابلیت اتکا (Dependability concepts)

۱-۱ مقدمه (Introduction)

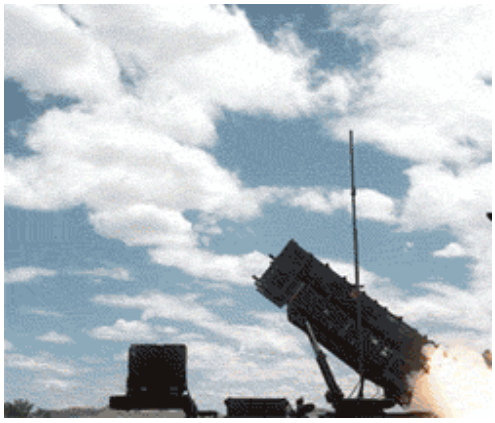
هدف این درس بررسی روش های افزایش تحمل پذیری خطا در سیستم های کامپیوتری و ارزیابی قابلیت اطمینان آنها می باشد. تحمل پذیری خطا به قابلیتی از سیستم اشاره دارد که سیستم را توانا می سازد تا حتی در صورت بروز خطا در آن بتواند آن خطا را پوشش داده و از خرابی سیستم جلوگیری کند. این قابلیت در بسیاری از سیستم ها در زمره مهمترین ملزومات غیرکارکردی سیستم است.

برای آن که اهمیت این موضوع بیشتر روشن شود، به مثال زیر به عنوان یک نمونه از هزاران گزارش مستند توجه کنید.

در ۲۵ فوریه ۱۹۹۱ در خلال جنگ خلیج فارس در الظهران عربستان یک موشک Patriot آمریکا نتوانست موشک Scud عراقی را منهدم کند. موشک Scud به پایگاه نظامی آمریکا برخورد کرد و ۲۸ نفر کشته و ۱۰۰ نفر مجروح شدند. چند نکته در این مورد جالب توجه است: ۱- تسلیحات جنگی جزء در دسته کاربردهای بحرانی از لحاظ ایمنی و سلامت (Safety critical) هستند. ۲- این دسته از موشکهای ضد موشک جزء دسته کاربردهای بیدرنگ (Real-time) نیز هستند. زیرا موشکهای مخاصم که باید ردیابی و منهدم شوند، با سرعتی نزدیک به ۵ ماخ حرکت می کنند. ۳- این موشکها بسیار گران هستند، پس باید بسیار قابل اطمینان باشند.

اما، دلیل این حادثه چه بوده است؟

ساعت داخلی موشک به صورت یک عدد صحیح ۲۴ بیتی پیاده سازی شده بود. این ساعت با ساعت های دیگر سیستم های رادار و ماهواره ای سنکرون عمل می کند. برش (Truncation) اعداد در ثباتهای ۲۴ بیتی خطایی معادل $0.0_{23}1100110011\ldots$ (0_{23} یعنی ۲۳ بار تکرار رقم صفر) یا $0.0_{7}95$ (در مبنای ده) به ازای هر یک دهم ثانیه ایجاد می کند که این خطا بعد از گذشت ۱۰۰ ساعت از روشن بودن سیستم معادل 0.34 ثانیه می شود. یک موشک Scud با سرعت 1,676 متر در ثانیه حرکت می کند، یعنی در این زمان (تاخیر حادث شده به خاطر خطا) نیم کیلومتر طی مسافت خواهد کرد. بنابراین پس از تشخیص موشک مخاصم، سیستم با تاخیر شلیک کرد و نتوانست به درستی موشک را تعقیب و منهدم کند. دلیل این اشتباه این بود که موشک Patriot برای استفاده کوتاه مدت (Short-term) در دهه ۷۰ طراحی شده بود و برای ماموریتهای آماده باش بلند مدت تست نشده بود. به عبارت دیگر، اگر سیستم هر چند ساعت راه اندازی مجدد (Reboot) می شد، این مشکل پیش نمی آمد. هر چند که این اشکال بعدا با به کارگیری ثباتهای ۶۴ بیتی و تغییراتی در کد حل شد!



۱-۲ مفاهیم (Concepts)

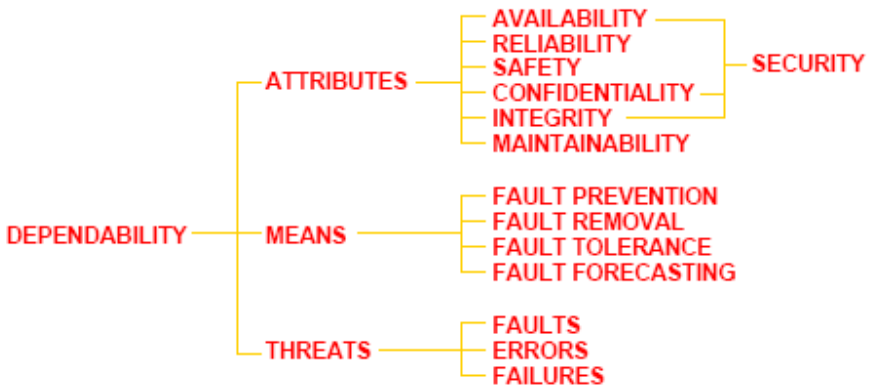
گروه کاری IFIP WG 10.4 (تحمل پذیری خطا و محاسبات قابل اتکا) در تلاشی که از سال ۱۹۸۰ آغاز شده است، سعی کرد تا مفاهیم و واژگان این علم را به صورت استاندارد و بدون ابهام بیان کند. در این درس نیز سعی می شود از همین واژگان استفاده شود.

هر سیستم را با پنج خصیصه زیر ارزیابی می کنند.

- خصوصیات کارکردی(Functionality)
- خصوصیات قابلیت استفاده (Usability)
- کارایی (Performance)
- هزینه (Cost)
- خصوصیات قابلیت اتکا (Dependability)

هدف نهایی تحمل پذیری خطا، ساخت سیستم های قابل اتکا (Dependable) است. قابلیت اتکا یک مفهوم کیفی و عام است. قابلیت اتکا، توانایی یک سیستم برای ارائه و تحویل سرزیس مورد نظر به کاربران است به گونه ای که بتوان به ارائه شدن آن سرویس اطمینان داشت. منظور از سرویس تحویلی توسط سیستم، رفتار سیستم است آن گونه که توسط کاربران دریافت و ادراک می شود.

مفهوم قابلیت اتکا را می توان بر اساس درخت قابلیت اتکا (مطابق شکل زیر) بررسی کرد.



۱-۲-۱ خواص سیستم های قابل اتکا (Dependable systems attributes)

از آنجاییکه قابلیت اتکا یک مفهوم کیفی است، برای ارزیابی مهندسی و دقیق تر سیستم ها چند خاصیت برای سیستم های قابل اتکا تعریف شده است که عموما مفاهیم کمی (عددی و نه کیفی) هستند. در زیر تعریف مختصر آنها ارایه می شود و در فصل های بعدی درس با جزییات بیشتر به هر یک پرداخته خواهد شد.

قابلیت اطمینان (Reliability)، $R(t)$، احتمال شرطی این است که سیستم در بازه زمانی $[t_0,t]$ به درستی کار کند، به شرط این که سیستم در ابتدای بازه (t_0) درست بوده است.

عدم اطمینان (Unreliability)، $Q(t)$، احتمال شرطی این است که سیستم در بازه زمانی $[t_0,t]$ به درستی کار نکرده است، به شرط این که سیستم در ابتدای بازه (t_0) درست بوده است.
با توجه به تعاریف فوق، وقفه های کوتاه و بازه های زمانی حتی کوتاه مدت عدم کارکرد سیستم و سپس تعمیر آن در رابطه با قابلیت اطمینان قابل قبول نمی باشد.

نکته مهم: تحمل پذیری خطا روشی برای افزایش قابلیت اطمینان است. اما توجه شود که یک سیستم تحمل پذیری خطا الزاما بسیار قابل اطمینان نیست.

قابلیت دسترسی (Availability)، $A(t)$، احتمال این است که سیستم در زمان t در حال کار درست و در دسترس باشد و کارکرد خود را انجام دهد.
توجه شود که قابلیت اطمینان در بازه زمانی تعریف می شود ولی قابلیت دسترسی در لحظه زمان تعریف شود.
در ضمن قابلیت تعمیر شدن سیستم در قابلیت اطمینان سیستم تاثیری ندارد اما در قابلیت دسترسی سیستم بسیار مهم است.

ایمنی و سلامت (Safety)، S(t)، احتمال این است که سیستم در بازه زمانی $[t_0, t]$ یا بدرستی کار خود را انجام دهد (یعنی خراب نشده باشد) و یا اگر خراب شود به گونه ای این اتفاق رخ دهد که تاثیر مخرب بر محیط زیست، سلامت انسان نداشته باشد (اصطلاحا به صورت خوش خیم (Fail-safe) خراب شود).

قابلیت نگهداشت (Maintainability)، M(t)، احتمال این است که یک سیستم خراب بتواند تا زمان t ترمیم شود و آماده کارکرد مجدد شود.

آزمون پذیری (Testability) بیانگر میزان سهولت و در دسترس بودن راهکارهایی برای آزمودن سیستم و یا بازبینی (Verification) سیستم می باشد.

۱-۲-۲ تهدیدات (Threats)

خطا (Fault)، اشتباه (Error)، و خرابی (Failure) مفاهیمی هستند که معمولا مترادف هم و به جای هم استفاده می شوند. اما با توجه به کار استاندارد سازی این مفاهیم در گروه کاری IFIP WG 10.4 کاربرد و تعریف هر کدام را در این درس به طور مشخص بیان می کنیم.

در حقیقت خطا، اشتباه، و خرابی به صورت یک زنجیرعلت و معلولی تعریف می شوند. هر کاستی و نقص (عموما فیزیکی) مانند قطع شدن یک سیم، سوختن یک ترانزیستور یا دیود و ... یک خطا (Fault) تلقی می شود. اشتباه (Error) به علت بروز خطا ممکن است رخ دهد و در حقیقت اگر خطا حالت و وضعیت یک مولفه از سیستم را تغییر دهد، اشتباه (Error) روی داده است. حال اگر بروز این اشتباه موجب شود که سیستم نتواند کارکرد صحیح خود را ارائه دهد، اشتباه منجر به خرابی (Failure) شده است.

مثال زیر یکی از بهترین مثالهایی است که می تواند ارتباط این مفاهیم را روشن کند. ایجاد یک سوراخ در لاستیک چرخ اتومبیل یک خطا است. حال اگر این سوراخ منجر به پنچر شدن لاستیک شود، اشتباه (Error) بروز کرده است. تا زمانی که این لاستیک پنچر شده، کارکرد درست اتومبیل را تحت تاثیر قرار ندهد، اشتباه به خرابی تبدیل نشده است. به عنوان مثال اگر لاستیک پنچر شده، لاستیک چرخ زاپاس اتومبیل باشد. اما اگر اینگونه نباشد و یا یکی دیگر از چرخ های اتومبیل پنچر شود، اشتباه منجر به خرابی شده است.

پس می توان گفت که رابطه علت و معلولی زیر برای این مفاهیم صادق است.

به عبارت دیگر خطا در حوزه و دنیای فیزیکی (Physical universe) روی می دهد، اشتباه (Error) در حوزه و دنیای اطلاعات (Information universe) و خرابی در حوزه و دنیای خارجی (External universe) روی می دهد.

درنگ یا نهفتگی خطا (Fault latency) به فاصله زمانی بین بروز خطا و ظهور اشتباه حاصل از آن خطا اطلاق می شود.

درنگ یا نهفتگی اشتباه (Error latency) به فاصله زمانی بین بروز اشتباه و خرابی سیستم حاصل از آن اشتباه اطلاق می شود.

۱-۲-۳ روش ها (Means)

در این بخش روش ها و مکانیزم های مورد استفاده برای افزایش قابلیت اتکا سیستم معرفی می شود.

الف- پیشگیری از خطا (Fault prevention): به روشهایی اطلاق می شود که سعی دارند از بروز خطا پیشگیری کنند و اکثر آنها مبتنی بر استفاده از روشهای کنترل کیفیت بر ساخت سیستم های سخت افزاری و نرم افزاری می باشند. برخی از این روشها عبارتند از: برنامه سازی ساخت یافته و شی گرا، استفاده از متدولوژی در توسعه نرم افزار، Shielding، استفاده از دیواره های آتش (Firewall) و ...

ب- تحمل پذیری خطا (Fault tolerance): به روشهایی اطلاق می شود که سعی دارند حتی با بروز خطا، از گسترش آن و خراب شدن سیستم جلوگیری کنند. این روش ها می توانند مبتنی بر کشف اشتباه (Error detection) و ترمیم آن (Error recovery) باشند. گونه دیگر این روش ها بر اساس پوشش دادن خطا (Fault masking) عمل می کنند که سعی دارد با افزودن افزونگی به سیستم بدون نیاز به کشف خطا، اثر آن را خنثی نماید. روش های این دسته را در فصول بعد به طور مشروح بررسی خواهد شد.

پ- برداشت خطا (Fault removal): روش های برداشت خطا هم در زمان توسعه و هم در زمان به کارگیری و بهره برداری سیستم انجام می شود. در زمان توسعه سیستم در این روش، سه مرحله بازبینی و ممیزی (Verification)، تشخیص و عیب شناسی (Diagnosis) و تصحیح (Correction) انجام می شود. در مرحله بازبینی و ممیزی بررسی می شود که آیا سیستم خصوصیات و کارکرد لازم را دارد. اگر کارکرد لازم را نداشت، مرحله تشخیص و عیب شناسی برای یافتن علت مشکل انجام می شود و سپس خطای کشف شده اصلاح می شود. در زمان بهره برداری سیستم نیز روش های نگهداری سیستم سعی در حذق خطا های گزارش شده دارد و در ضمن همچنان سیستم مورد ارزیابی قرار می گیرد تا خطاهای احتمالی که هنوز ظاهر نشده اند، نیز کشف و برطرف شوند.

ت- پیش بینی خطا (Fault forecasting): در این روش ها سعی می شود رفتار سیستم در مقابل رخ دادن خطاهای مختلف ارزیابی شود. ارزیابی سیستم می تواند به صورت کیفی (qualitative) و یا کمی (Quantitative) باشد. روش های احتمالی مانند زنجیره مارکف (Markov chain) و یا دیاگرام بلوکی قابلیت اطمینان (RBD) برای ارزیابی استفاده می شود. از نتایج پیش بینی خطا می توان نقاط ضعف سیستم را ارزیابی کرد و سپس به کمک روش های دیگر با آن مقابله کرد.

۱-۳ کاربردهای سیستم های قابل اتکا (Dependable applications)

گونه های مختلف کاربردهای سیستم های قابل اتکا را می توان به چند دسته تقسیم کرد:

کاربردهای با عمر زیاد: کاربردهایی نظیر ماهواره ها و سفینه های فضایی از این دسته هستند که باید در یک بازه عملیاتی طولانی (مثلا ۱۰ سال) با احتمال بالایی کار کنند. قابلیت اطمینان متعارف این کاربردها برای بازه ۱۰ سال برابر 0.95 است.

محاسبات و پردازش های بحرانی: به کاربردهایی اطلاق می شود که کارکرد درست آنها برای محیط زیست و سلامت انسانها حیاتی هستند. به عنوان مثال سیستم های کنترل صنعتی مانند مراکز نیروی هسته ای، سیستم های کنترل پرواز در هواپیما از این دسته هستند. قابلیت اطمینان متعارف این کاربردها برای بازه ۳ ساعت برابر 0.9_7 (0.9999999) است.

کاربردهای نیازمند تاخیر در نگهداشت (Maintenance Postponement): کاربردهایی هستند که تعمیر کردن آنها بسیار پر هزینه و یا نا متعارف و یا غیرممکن است. سیستم های فضانوردی، مخابراتی و یا تجهیزات انتقال و تقویت رادیویی و تلویزیونی که در مناطق صعب العبور قرار دارند، از این دسته هستند. در این گونه از کاربردها سعی می شود با روشهای تحمل پذیری خطا، خطاها موجب خرابی سیستم نشود تا در فرصت مناسب بتوان تعمیر و بازسازی انجام داد.

کاربردهای با دسترسی بالا (High availability): به کاربردهایی نظیر سیستم های بانکداری، سیستم های مخابراتی (و تلفنی) و سیستم های اطلاع رسانی اطلاق می شود که باید بتواند سرویس درخواستی را با احتمال بالایی پاسخ دهند. بنابراین وقفه های کوتاه در سرویس در این گونه از کاربردها قابل تحمل است به شرطی که استفاده کنندگان از سرویس متوجه گسستگی سرویس نشود.