فصلنامه فرماندهی و کنترل، سال اول، شماره ۲، زمستان ۱۳۹۵

ارائهٔ چارچوبی برای کنترل امنیت و قابلیت اطمینان سامانه‌های نهفتهٔ حیاتی از طریق بازپیکربندی

ابوالقاسم صادقی[۱]، محمدرضا ولوی[۲]، غلامرضا محتشمی[۳]، مرتضی براری[٤]

تاریخ دریافت: ۹۵/۸/۷ تاریخ پذیرش: ۹۵/۹/۲۷

چکیده

سامانه‌های نهفته در سامانه‌های دارای مأموریت‌های حیاتی به‌ویژه در سامانه‌های فرماندهی و کنترل، از اجزای حساس محسوب می‌شوند که عملکردهای مهمی از کل سامانه را انجام می‌دهند. لازمهٔ استمرار فعالیت‌های چنین سامانه هایی مراقبت همزمان از خصوصیات کیفی کلیدی سامانه است تا بتوان استمرار خدمات حیاتی آن را در طول زمان به شکل قابل قبولی تضمین نمود. تاکنون روی مدیریت خصوصیات کلیدی قابلیت اطمینان و امنیت سامانه تحقیقات زیادی به صورت مجزا انجام شده است، اما مدیریت هر یک از این خصوصیات کیفی تمهیدات مجزایی دارد. تاکنون تلاش زیادی برای یکپارچه‌سازی یا حداقل هماهنگ‌سازی آنها با هم انجام نشده است. به همین دلیل، اشکالات مخربی ایجاد شده است که در صورتی که توجه به این دو موضوع به صورت ترکیبی و یکپارچه انجام شود، خطرات و خرابی‌های کمتری برای سامانه‌های حیاتی ایجاد می‌گردد. در این مقاله تلاش شده است چارچوبی یکپارچه برای مدیریت مشترک خصوصیات امنیت و قابلیت اطمینان در سامانه‌های نهفته ارائه شود. ایدهٔ اصلی این چارچوب، استفاده از مدل Willow جهت بازپیکربندی سامانه به‌منظور محافظت از این دو ویژگی کلیدی است. برای این منظور سه نوع کنترل‌کننده در طرح پیشنهادی در نظر گرفته شده و برای هر یک خط‌مشی‌هایی با قواعد نحوی مشخص طراحی و پیاده‌سازی شده است. این خط‌مشی‌ها بر اساس متغیرهایی تعریف شده‌اند که انواع اندازه‌گیری‌های مورد نیاز را در حیطه‌های نرم‌افزاری، سخت‌افزاری، شبکه‌ای و امنیتی انجام می‌دهند، سپس با شبیه‌سازی یک محیط کاربردی متداول نظامی شامل چهار گره ارتباطی با هم‌بندی کامل، اثربخشی این خط‌مشی‌ها و توانایی آنها در پایش و کنترل همزمان امنیت و قابلیت اطمینان نشان داده شده است.

واژگان کلیدی: سامانه‌های نهفته، بازپیکربندی، امنیت، قابلیت اطمینان

۱. دانشجوی دکتری، دانشگاه صنعتی مالک اشتر، sadeghi_ict@mut.ac.ir

۲. دانشیار دانشگاه صنعتی مالک اشتر، valavi@mut.ac.ir

۳. استادیار دانشگاه صنعتی مالک اشتر، barari@mut.ac.ir

٤. استاد دانشگاه فردوسی مشهد، grmohtashami@ferdowsi.um.ac.ir

۱. کلیات

۱-۱. بیان مسئله

سامانه‌های فرماندهی و کنترل بیشتر در شرایط بحران مورد استفاده قرار می‌گیرند. این بحران‌ها شامل مواردی مانند بحران‌های نظامی (نبرد سخت) یا سوانح طبیعی (سیل، زلزله و ...) هستند. هدف از به‌کارگیری این سامانه‌ها در چنین شرایطی، ارائهٔ خدمات حیاتی برای مقابله و مدیریت هر چه بهتر بحران است.

از منظر تهدیدشناسی، دو حوزهٔ امنیت و قابلیت اطمینان با یکدیگر متفاوت هستند. در حوزهٔ امنیت بیشتر تهدیدهای عامدانه با منشأ بیرونی مورد بررسی قرار می‌گیرند، اما در قابلیت اطمینان فقط تهدیدهای ناخواسته درونی مورد توجه می‌باشند که بیشتر ناشی از اشکالات خود سیستم در طراحی و ساخت هستند.

ویژگی خاص سامانه‌های فرماندهی و کنترل این است که در بهره‌برداری از این سامانه‌ها در شرایط بحران، اغلب به جای تأکید بر شناسایی منشأ تهدید علیه سامانه، اصل کارکرد صحیح آن مورد توجه قرار دارد، بنابراین هرگونه خرابی یا تهدید بیرونی که باعث اختلال عملکردهای حیاتی و اصلی سامانه شود، غیرقابل تحمل خواهد بود و در این رابطه توجه چندانی به اینکه منشأ تهدید درون یا بیرون سامانه است یا عامدانه و سهوی بودن منشأ تهدید نمی‌شود. مهم‌ترین هدف، محافظت از عملکردها و خدمات اساسی و حیاتی سامانه است تا بتوان بحران را مدیریت نمود. درست در همین نقطه است که مدیریت و کنترل یکپارچه و ادغام‌شدهٔ امنیت و قابلیت اطمینان در سامانه‌های فرماندهی و کنترل موضوعیت اساسی پیدا می‌کند. طبیعی است که پس از اتمام بحران، منشأ تهدید و آسیب‌پذیری‌ها و اشکالات موجود در سامانه می‌تواند به دقت مورد تحلیل و موشکافی قرار گرفته و اصلاحات مورد نیاز در آن انجام شود.

سامانه‌های فرماندهی و کنترل در بیشتر موارد از سامانه‌های نهفته خاص منظوره‌ای استفاده می‌کنند که عملیات مشخصی را کنترل و هدایت می‌کند. مثال‌هایی از این نوع سامانه‌ها ربات‌های مین‌یاب، ربات‌های آتش‌نشان، سامانه‌های ناوبری رزمی یا تجاری، سامانه‌های هدایت موشک‌ها هستند. در این سامانه‌ها، سامانه‌های نهفتهٔ حیاتی اغلب از اجزای کلیدی و حیاتی هستند که لطمه به عملکرد آنها کل عملکرد سامانه را با مخاطره مواجه می‌کند (Banerjee, 2012)؛ به بیان دیگر در درون هر یک از این سامانه‌های بزرگ، به‌طور معمول، اجزای کلیدی وجود دارند که اشکالات، خرابی‌ها یا دستکاری آنها می‌تواند منجر به صدمات گسترده به عملکرد سامانه شود و این صدمات به نوبهٔ خود آثار وخیم اقتصادی، جانی و امنیتی خواهند داشت. به این اجزا، اجزای کلیدی سامانه‌های فرماندهی و کنترل گفته می‌شود (Goertzel, 2009).

۱-۲. اهمیت و ضرورت موضوع تحقیق

نمونه‌های متعددی دربارهٔ تاثیرهای مخرب اشکال‌های ناخواسته یا خرابکاری عامدانه در این قبیل سامانه‌ها موجود است. مشکل پیش‌آمده برای سامانهٔ کنترلی سفینهٔ آریان ۵ در اثر خطای نرم‌افزار بخش مکان‌سنج[1] آن که منجر به انفجار سفینه چند ثانیه پس از پرتاب آن شد (Lions, 1996)، مرگ و میرهای متعدد ناشی از خطاهای سامانهٔ کنترل دستگاه رادیوتراپی Therac-25 که منجر به مرگ چندین مریض در اثر تشعشات بسیار

1. IRS: Inertial Reference System

شدید آن شد (Leveson, 1993) و یا مشکل پیش‌آمده در سامانهٔ کنترل هوشمند خودرو Prius شرکت تویوتا که منجر به توقف ناگهانی خودرو در سرعت‌های بالا در بزرگراه‌ها می‌شد (CBS News, 2010)، تنها، نمونه‌هایی از تأثیرهای مخرب اشکال‌های ناخواسته در سامانه‌های نهفته حیاتی بوده‌اند. در کنار آنها برخی خرابکاری‌های عمدی که از زمان انفجار خطوط لولهٔ گاز سیبری در شوروی سابق در ۱۹۸۲ شروع شده (Markoff, 2009) و تا اکنون با شدت بیشتری مانند کرم Stuxnet ادامه دارد، نشان از اهمیت تمهیدات امنیتی در کنار تمهیدات قابلیت اطمینان در این سامانه‌ها دارد. به‌دلیل حیاتی بودن عملکرد این اجزاء، باید بتوان عملکردهای کلیدی آنها را محافظت نمود. این محافظت باید هم در مقابل خرابی‌ها و اشکالات احتمالی ناخواسته سامانه و هم در برابر دستکاری‌ها و خرابکاری‌های عمدی و حمله‌ها انجام شود. به این دلیل، باید مدلی داشت که توانایی پایش، ارزیابی، و محافظت از سطح امنیت و قابلیت اطمینان این سامانه‌ها را به صورت یکپارچه داشته باشد.

۳-۱. پیشینهٔ تحقیق

فعالیت‌هایی که در زمینه‌های مرتبط انجام شده است، اغلب به یکی از دو حوزهٔ امنیت یا قابلیت اطمینان پرداخته‌اند و به ندرت می‌توان تحقیقات و مدل‌هایی را یافت نمود که این دو هدف کیفی را با هم ترکیب نموده باشند. در سامانه‌های نهفتهٔ پیچیده، سامانه را به صورت شبکه‌ای از زیرسامانه‌ها با روابط درونی در نظر می‌گیرند و سپس آن را بر اساس روش‌های ارزیابی و اندازه‌گیری قابلیت اطمینان شبکه‌ای تحلیل می‌کنند (Banerjee, 2012 و Meedeniya, 2011). روش متداول دیگر آن است که سخت‌افزار سامانهٔ نهفته به صورت جداگانه و با روش‌های ارزیابی قابلیت اطمینان سخت‌افزار و نرم‌افزار سامانه با روش‌های قابلیت اطمینان نرم‌افزار (Wolf, 2012) به صورت جداگانه تحلیل و ارزیابی شوند. نظریهٔ قابلیت اطمینان با وجود آنکه برای هدف‌های از پیش تعریف‌شدهٔ خود کامل و جوابگو است، ولی نقص عمدهٔ آن در نظر گرفتن صرف اشکال‌ها و خرابی‌های ناخواستهٔ سامانه است که برای هدف ما، که همان تحلیل و مدیریت یکپارچه خرابی‌های ناخواسته با حملات عامدانه و خرابکاری‌ها در سامانه است ناقص محسوب می‌شود.

تحلیلگران امنیتی برای ارزیابی سطح امنیت سامانه‌ها اغلب از سطح‌بندی‌هایی بهره می‌برند که برخی استانداردها مانند استاندارد معیار مشترک[1] (Common Criteria, 2017) ارائه نموده‌اند. این تحلیل‌ها با آنکه از نظر مقاوم بودن سامانه در مقابل برخی حمله‌ها، برآوردهای خوبی به کاربر می‌دهند، ولی نمی‌توانند به وی اطمینان دهند که آیا سامانهٔ مورد نظر دارای ساختار درونی مقاوم و قابل اطمینانی از نظر استمرار عملکردهای خود می‌باشد یا خیر؟

٤-۱. پرسش تحقیق

نیاز مبرم سامانه‌های حیاتی فرماندهی و کنترل در صحنهٔ بحران، نبرد و ... این است که صحت عملکردهای اساسی آنها تا انتهای عملیات قابل کنترل و مدیریت باشد. این موضوع مستلزم مدیریت همزمان و یکپارچهٔ خصوصیات کیفی حوزه‌های امنیت و قابلیت اطمینان سامانه است. پرسش این است که چگونه می‌توان این

1. CC: Common Criteria

خصوصیات -که در بعضی موارد با هم نامتجانس هستند- را با یک روش یکسان و هماهنگ توصیف و پایش کرد؟، همچنین چگونه می‌توان با مدیریت یکپارچه و توأمان آنها استمرار عملکردهای اساسی سامانه را در شرایط بحران مدیریت و تضمین نمود؟

۱-۵. روش‌شناسی تحقیق

این تحقیق به صورت یک مدل نظری جامع و عمومی ارائه گردیده است، سپس مدل ارائه‌شده در یک وضعیت عملیاتی مفروض فرماندهی و کنترل در محیط MATLAB شبیه‌سازی شده و نتایج آن گزارش شده است. در بخش دوم، ایدهٔ مدل پیشنهادی ارائه می‌گردد. سپس مدل پیشنهادی تشریح می‌شود. اجزای مدل و چگونگی پیاده‌سازی و استفاده از آنها را در انتهای بخش دوم توضیح داده خواهد شد. در بخش سوم، شبیه‌سازی انجام‌شده به همراه نتایج به‌دست آمده گزارش می‌شود. مزایای مدل ارائه‌شده نسبت به سایر مدل‌های مشابه در انتهای بخش سوم، بررسی و ارزیابی می‌شود. در بخش چهارم و نهایی نیز نتیجه‌گیری و جمع‌بندی و پیشنهاد کارهای پژوهشی آتی ارائه می‌گردد.

۱-۶. سازماندهی تحقیق

ساختار این پژوهش در چهار بخش تنظم شده است که بخش اول شامل بیان مسئله، ضرورت و اهمیت، اهداف، پیشینه، پرسش و روش تحقیق است.

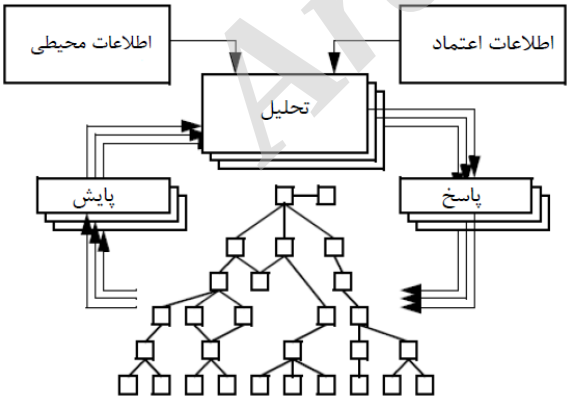
در بخش دوم به مبانی نظری در قالب چارچوب کلی ارزیابی صحنهٔ نبرد اشاره شده و در بخش سوم به یافته‌های تحقیق اعم از مدل‌سازی و شبیه‌سازی شبکهٔ حسگری سایبری، چالش‌ها و راه‌حل‌ها پرداخته می‌شود و در پایان در فصل نتیجه‌گیری به جمع‌بندی و پیشنهاد کارهای آتی اشاره می‌گردد.

۲. ادبیات و مبانی نظری تحقیق

۲-۱. مدل پیشنهادی

برای حل مشکل مطرح شده در بخش پیش، باید مدلی برای طراحی و تولید سامانه‌های نهفته ارائه گردد که قابلیت مدیریت یکپارچه امنیت و پایداری را در خود داشته باشد. به این منظور به‌عنوان پایهٔ ایدهٔ اصلی مدل از معماری Willow (Knight, 2002) در رابطه با سامانه‌های حیاتی بقاپذیر استفاده شده و از آن بر مبنای نیاز کار در مدل پیشنهادی سامانه‌های نهفته به شکل تکمیل شده استفاده می‌شود. از آنجا که هدف اصلی معماری Willow تأمین قابلیت بقا برای سامانه‌های توزیع‌شده بزرگ بوده است، ایدهٔ کلی آن استفاده از تعداد زیادی حلقهٔ کنترلی در آن سامانه توزیع‌شده است که این حلقه‌های کنترلی، ضمن مستقل بودن از یکدیگر با هم تعامل‌پذیر هستند. شکل ۱ نمودار پایهٔ این معماری را نشان می‌دهد.

شکل ۱. نمودار اصلی معماری Willow



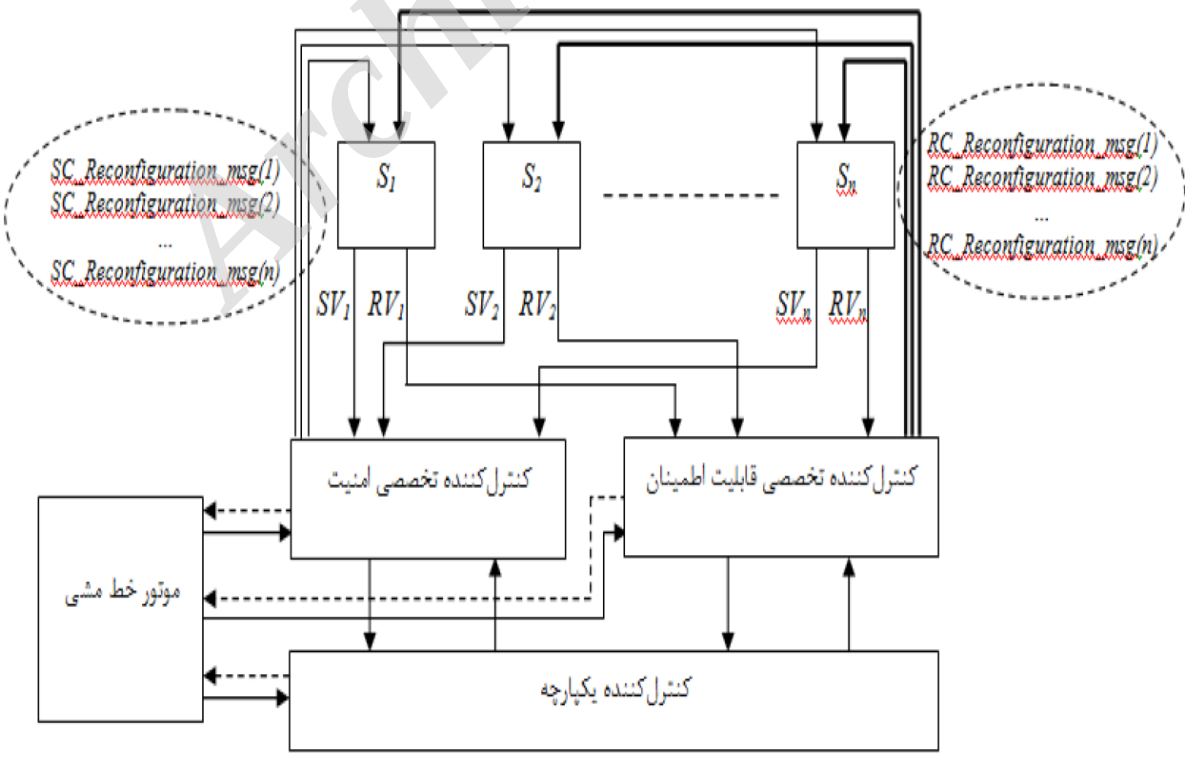
چگونگی عملکرد این معماری به این ترتیب است که در طول دورهٔ عملکرد سامانه حیاتی، افزون بر

وضعیت خود سامانه، وضعیت متغیرهای محیطی نیز پایش می‌شود و نتایج ترکیب‌شدهٔ این پایش به بخش تحلیل جهت بررسی داده می‌شود. سپس در صورت نیاز به انجام کاری مانند تغییر در برخی زیرسامانه‌ها بخش تحلیل به بخش پاسخ، عملیات مربوطه را اعلام می‌کند و بخش پاسخ نیز آن را اجرا می‌نماید. هدف از طرح این بحث، مرور این معماری جهت آمادگی برای توضیح مدل پیشنهادی تحقیق جاری بوده است.

اکنون مدل پیشنهادی در شکل ۲ تشریح می‌گردد. این شکل، شمای یک سامانهٔ نهفتهٔ دارای قابلیت خودکنترلی و بازپیکربندی (هم از نظر امنیتی و هم از نظر پایداری) را نشان می‌دهد.

این سامانه دارای تعدادی زیرسامانه خواهد بود که در شکل ۲ به شکل اجزای S1 تا Sn نمایش داده شده‌اند. برای هر جزء تعدادی ویژگی و کمیت قابل اندازه‌گیری (هم از نظر امنیت و هم از نظر قابلیت اطمینان) مشخص شده است. به این ترتیب هر یک از اجزاء دارای k ویژگی امنیتی S_{t_i} که $1 \le i \le k$ و m ویژگی قابلیت اطمینان R_{t_j} که $1 \le j \le m$ خواهند بود. مجموعه کمیت‌های امنیتی S_{t_i} به صورت یک بُردار $Sv=(S_{t_1},...,S_{t_k})$ و مجموعه کمیت‌های قابلیت اطمینان را به صورت یک بُردار $Rv=(R_{t_1},...,R_{t_m})$ نمایش داده می‌شود. این کمیت‌ها هر یک دارای یک حد آستانه هستند که سطح قابل قبول آنها را نشان می‌دهد و آن با $S_{t_i}(th)$ و $R_{t_j}(th)$ نمایش داده می‌شود.

شکل ۲. مدل پیشنهادی مبتنی بر بازپیکربندی



فلسفهٔ کلی این مدل، پایش مداوم کمیت‌های امنیت و قابلیت اطمینان در تمام اجزاء، تحلیل و ارزیابی آنها و تشخیص وجود مشکلات احتمالی از نظر امنیت و پایداری در هر جزء و در نهایت اعمال بازپیکربندی‌های مورد نیاز جهت اصلاح آن شرایط و مقابله با مشکل است. حدود آستانه و باید با توجه به ماهیت هر کمیت بر اساس شرایط عملی و کاربردی سامانه و به صورت تجربی و با تقریب خطای قابل قبول توسط بهره‌بردار سامانه مشخص شوند. برای نیل به این هدف هر یک از n زیرسامانه‌های Si مقادیر ویژگی‌های امنیت و قابلیت اطمینان خود را به ترتیب به صورت دو بردار Svi و Rvi استخراج و به کنترل‌کننده‌های تخصصی امنیت و قابلیت اطمینان ارسال می‌کنند. کنترل‌کننده‌های تخصصی وظیفه دارند هر یک از کمیت‌های بالا را بررسی و ارزیابی کنند و در صورت عبور هر یک از کمیت‌ها از حد آستانهٔ قابل قبول عکس‌العمل مناسب را نشان دهند.

۲-۲. تشریح مدل پیشنهادی

بخش موتور خط‌مشی وظیفه دارد تمامی تدابیر و اقدامات مورد نیاز برای حالت‌های مختلف انحراف در عملکرد سامانه اعم از نقض‌های امنیتی یا خرابی‌ها و ناپایداری‌های سامانه را بر حسب مقادیر هر یک از کمیت‌ها و یا ترکیب‌های متنوعی از آنها در نظر بگیرد و حالت‌هایی که نیاز به عکس‌العمل دارند را به همراه نوع عکس‌العمل مربوطه مشخص کند. هر سه کنترل‌کنندهٔ موجود در این مدل از تعدادی گزارهٔ شرطی استفاده می‌کنند؛ به این معنی که ورودی‌هایی که به صورت Svi و Rvi دریافت می‌کنند را با شرایط تعیین‌شده و قوانین تعریف شده برای خود می‌سنجند و چنانچه شروط مطرح در برخی قوانین در ورودی‌ها واقع شود اقدامات مربوط به آن قوانین را به صورت فرامین بازپیکربندی اعمال می‌کنند. این گزاره‌های شرطی یا قوانین در واقع همان خط‌مشی‌هایی هستند که توسط موتور خط‌مشی تعیین و به هر یک از کنترل‌کننده‌ها ابلاغ شده‌اند؛ برای نمونه در شکل ۳ چند خط‌مشی مثالی در موتور خط‌مشی ارائه می‌گردد. این خط‌مشی‌ها توسط بهره‌بردار سامانه تعریف و اعمال می‌شوند و در این تحقیق هیچ پیش‌فرض یا توصیه‌ای درباره آنها ارائه نمی‌شود.

شکل ۳. چند نمونه خط‌مشی مثالی

Policy Engine:

P1: $if\ S_1 \to S_{t_1} \le S_1 \to S_{t_1}(th) \Rightarrow Shutdown(S_1);$

P2: $if\ S_3 \to S_{t_2} \le S_3 \to S_{t_2}(th) \Rightarrow \mathrm{Re}start(S_2);$

P3: $if\ |Rv_1| \le \frac{\sum_{i=1}^{m} S_1 \to R_{t_i}(th)}{m} \Rightarrow RC\_\mathrm{Re}configuration\_msg(1);$

P4: $if\ \prod_{i=1}^{k} |Sv_i| \le 100 \Rightarrow \mathrm{Re}start\ the\ whole\ system;$

P5: $\forall S_i \in \{S_i \mid 1 \le i \le n\},\ if\ |Sv_i| + |Rv_i| \le 10 \Rightarrow Shutdown(S_i)\ and\ alert\ for\ replacing\ it$

این خط‌مشی‌ها نسبت به هم دارای استقلال زمانی و عملکردی هستند و می‌توانند به موازات هم فعال شوند. فقط در شرایطی که فعال شدن همزمان دو خط‌مشی تداخل ایجاد می‌کند، خود بهره‌بردار سامانه باید ترتیب اولویت اجرای آنها را در سامانه مشخص نماید.

در شکل ۳ خط‌مشی P1 مشخص می‌کند که چنانچه در زیرسامانهٔ S1 کمیت امنیتی از حد آستانهٔ خود پایین‌تر رفت، باید S1 بلافاصله خاموش شود که این خط‌مشی با توجه به محتوی و ماهیت آن باید در کنترل‌کنندهٔ امنیتی اعمال شود. خط‌مشی P2 می‌گوید که اگر در S3، کمیت از حد آستانه خود کمتر شد، S2 شروع مجدد شود. این مسئله می‌تواند به‌دلیل وابستگی‌های احتمالی بین عملکرد و خواص امنیتی S2 و S3 باشد که باز هم به کنترل‌کنندهٔ امنیت مربوط است. خط‌مشی P3 گفته است که اگر اندازهٔ بُردار Rv1 از متوسط مقدار عناصر آن کمتر شد، فرمان RC_Reconfiguration_msg(1) به S1 صادر شود. این خط‌مشی به کنترل‌کنندهٔ قابلیت اطمینان مربوط است. خط‌مشی P4 تعیین کرده که اگر حاصل‌ضرب اندازهٔ تمام بُردارهای Svi از عدد ۱۰۰ کمتر شد، کل سامانه باید شروع مجدد شود که به کنترل‌کنندهٔ امنیت مربوط است. خط‌مشی P5 نیز گفته است که اگر مجموع اندازه‌های بُردارهای Svi و Rvi برای هر زیرسامانهٔ Si کمتر از ۱۰ باشد، آن جزء باید خاموش و جایگزین شود. تشخیص اینکه هر خط‌مشی به کدام کنترل‌کننده مربوط است توسط خود موتور خط‌مشی انجام می‌شود.

۳-۲. تشریح اجزای مدل

۱-۳-۲. معیارها و متغیرهای اندازه‌گیری و پایش

برای اجرای عملی ایدهٔ معرفی‌شده در بخش ۳ باید ابتدا سنجه‌ها یا معیارهای مناسبی برای پایش شدن انتخاب و معرفی نمود. البته هدف اصلی در این تحقیق، نوآوری در زمینهٔ انتخاب و تعیین معیارها نیست، بلکه چگونگی استفاده از آنها در ساختار کنترلی و نظارتی برای مدیریت همزمان امنیت و قابلیت اطمینان مسئلهٔ اصلی است، از این‌رو می‌توان برای انتخاب معیارها به سایر منابع علمی مربوط مراجعه کرد. این معیارها در مراجع مانند (Merseguer, 2012), (Mazurkiewicz, 2016), (Eusgeld, 2008) و (Rubino, 2010) معرفی شده‌اند. در (Mazurkiewicz, 2016) معیارها به دو گروه کلی معیارهای تحلیلی و معیارهای تجربی تفکیک گردیده و برای هر یک تعدادی سنجه معرفی شده است، ولی در مجموع جامع‌ترین و معتبرترین آنها در (Eusgeld, 2008) معرفی و تحلیل شده است که در این تحقیق به همان معیارها استناد می‌شود. بدیهی است که مخاطب در این زمینه آزادی عمل داشته و می‌تواند بر مبنای نیازمندی و تشخیص خود معیارها را کم و زیاد نماید. معیارهایی که در (Eusgeld, 2008) معرفی شده‌اند در گروه‌های زیر دسته‌بندی گردیده‌اند:

(۱) شاخص‌های قابلیت اطمینان،

(۲) شاخص‌های امنیتی،

(۳) شاخص‌های کارایی،

(٤) شاخص‌های هم‌پوشان.

این شاخص‌ها می‌توانند طیف وسیعی از شاخص‌های بسیار ساده مانند حجم ترافیک عبوری شبکه یا تعداد بسته‌های گم شده تا شاخص‌های مشکل و پیچیده مانند

احتمال بروز یک حملهٔ امنیتی خاص در بازهٔ زمانی مشخص در آینده را شامل شوند. در این تحقیق در جدول‌های ۲ و ۳ شاخص‌های در نظر گرفته‌شده برای شبیه‌سازی و ارزیابی عملکرد سیستم معرفی خواهد شد.

۲-۳-۲. کنترل‌کنندهٔ تخصصی قابلیت اطمینان

این کنترل‌کننده وظیفه دارد تمامی اجزاء و زیرسیستم‌ها را از نظر قابلیت اطمینان پایش نماید و در صورت لزوم نسبت به اعمال تغییرات و تصحیحات لازم بر مبنای قوانین مشخص‌شده در موتور خط‌مشی، اقدام کند. سنجه‌های پایش این کنترل‌کننده عبارتند از:

(۱) سلامت/خرابی اجزاء،
(۲) نرخ خرابی،
(۳) تعمیر/جایگزینی اجزاء،
(٤) نرخ تعمیر/جایگزینی،
(۵) میزان مصرف I/O،
(٦) میزان مصرف حافظه،
(۷) متوسط زمان پاسخ،
(۸) قابلیت اطمینان.

این کنترل‌کننده تمام این متغیرها را در مورد هر یک از اجزاء و زیرسامانه‌ها مورد پایش قرار می‌دهد، سپس بر مبنای وضعیت هر جزء و قوانین تعریف‌شده در موتور خط‌مشی، در صورت لزوم اقدام‌های کنترلی مانند جایگزینی را در مورد آن جزء اعمال می‌نماید. افزون بر آن، این کنترل‌کننده بر مبنای مجموع اطلاعات گردآوری شده خود می‌تواند استنتاج‌های سطح بالاتری را نسبت به وضعیت قابلیت اطمینان در کل سامانه به صورت محاسباتی ارائه کند که در صورت نیاز، تصمیم‌ها و اقدام‌های کنترلی در سطح کل سامانه طراحی و پیاده‌سازی شود؛ برای مثال زمانی که در سطح چند جزء سامانه، افت کارایی قابل تحملی ملاحظه شود، ممکن است بر اساس محاسبات و قوانین مربوط به هر جزء، نیازی به تعمیر یا جایگزینی هیچ یک از اجزاء وجود نداشته باشد، اما وقتی در یک تحلیل سطح بالاتر، کل قابلیت اطمینان سامانه در نظر گرفته شود، بر اساس محاسبات و قوانین آن لایه، برخی اقدام‌های اصلاحی ضرورت خواهد یافت.

۳-۳-۲. کنترل‌کنندهٔ تخصصی امنیت

این کنترل‌کننده از نظر اهداف و ماهیت آن، درست مانند کنترل‌کنندهٔ تخصصی قابلیت اطمینان می‌باشد، ولی از نظر محتوایی معیارها و سنجه‌های مربوط به حوزهٔ امنیت اطلاعات و سامانه‌ها را پوشش می‌دهد. برخی از شاخص‌های مناسب برای پایش و کنترل توسط این کنترل‌کنندهٔ تخصصی در جدول ۱ بر اساس دسته‌بندی‌های تخصصی آنها آمده است.

جدول ۱. شاخص‌های امنیتی برای پایش و کنترل

گروه شاخص	نام شاخص
رمزنگاری	پیچیدگی
	طول کلید
	مدل Dolev-Yao
امنیت شبکه	نشانی‌های IP مبدأ و مقصد
	پورت‌های شبکهٔ مورد حمله قرار گرفته
	ترتیب پورت‌های مورد حمله
	زمان متوسط بین حمله‌ها
	تعداد بدافزارهای کشف شده
	تعداد حملات کشف شده
	شاخص‌های FP و FN[1]
	شاخص‌های PPV و NPV[2]
	نوع حمله‌های کشف‌شده
	تعداد آسیب‌پذیری‌های موجود
امنیت نرم‌افزار	نرم‌افزارهای مورد حمله قرار گرفته
	زمان متوسط بین حمله‌ها
	تعداد حمله‌های کشف شده به نرم‌افزارها
	نوع حملات کشف شده به نرم‌افزارها
	تعداد آسیب‌پذیری‌های موجود
مشترک	ALE[3]
	ROSI[4]
	VaR[1]

٤-۳-۲. کنترل‌کننده یکپارچه

این کنترل‌کننده می‌تواند به‌عنوان مکمل کنترل‌کننده‌های تخصصی و با ترکیب اطلاعات به دست آمده از هر دو کنترل‌کنندهٔ امنیت و قابلیت اطمینان، تحلیل‌هایی با استفاده از ترکیب اطلاعات هر دو کنترل‌کننده و به صورت هوشمندتر و سطح بالاتر ارائه نماید.

مراجعی مانند (G¨artner, 2003)، (Hoepman, 2003)، (Meadows, 1996) و (Pfitzmann, 2004) نشان داده‌اند که نوعی ترکیب یا هماهنگی یا وابستگی میان معیارهای امنیت با معیارهای قابلیت اطمینان یا قابلیت اعتماد وجود دارد. این کنترل‌کننده به دنبال آن است که با ترکیب پایش و کنترل این دو حوزه، امکان مدیریت بهتر، دقیق‌تر و مؤثرتر این دو حوزه را به صورت همزمان در سامانه ایجاد نماید.

٥-۳-۲. موتور خط‌مشی

برای آنکه مدیر سامانه بتواند بر اساس نیازها و الزام‌های خاص خود، سامانه را تنظیم و مدیریت نماید باید بتواند قیود و کنترل‌های مد نظر خود را به سامانه وارد نماید. همان‌گونه که در بخش ۳ مقاله مشخص شده است، این کار از طریق تعریف قوانینی انجام می‌شود که به موتور خط‌مشی سامانه داده می‌شود. کنترل‌کننده‌های مختلف، بر اساس قوانین یادشده در موتور خط‌مشی، تمام تراکنش‌ها و فعل و انفعالات سامانه را کنترل نموده و در صورت لزوم اقدام‌های مدیریتی و کنترلی را اعمال می‌کنند. چگونگی تعریف قوانین بر اساس قواعد نحوی زیر انجام می‌شود:

- $p \Rightarrow q$: به طور کلی تمام قوانین به صورت گزاره‌های شرطی توصیف می‌شوند که با رابطه‌های $p \Rightarrow q$ توصیف می‌شوند؛ به این مفهوم که اگر شرط p برقرار باشد باید اقدام q انجام شود.
- $S_i \rightarrow S_{t_i}$: ویژگی امنیتی S_{t_i} از زیرسیستم S_i
- $S_i \rightarrow R_{t_i}$: ویژگی قابلیت اطمینان R_{t_i} از زیرسیستم S_i
- $S_{t_i}(th)$ و $R_{t_i}(th)$ مقادیر آستانهٔ قابل قبول برای هر یک از ویژگی‌های امنیت و قابلیت اطمینان
- فرض بر این است که تمامی ویژگی‌های امنیت و قابلیت اطمینان، قابل اندازه‌گیری کمی و عددی هستند و تمام قوانین موتور خط‌مشی نیز بر اساس مقایسه‌های عددی نوشته می‌شوند. بدیهی است که در صورت وجود برخی شاخص‌های کیفی، می‌توان با روش‌هایی مانند آنچه در (Vijayaraghavan, 2010) آمده است، آنها را تبدیل به مقادیر معادل کمّی نمود.

٦-۳-۲. کنترل سازگاری قوانین موتور خط‌مشی

در مورد تمام سیستم‌هایی که فعالیت‌های کنترلی را بر طبق قوانینی که کاربر برای آن تعریف می‌کند –مانند دیوار آتش یا سوئیچ شبکه– انجام می‌دهند همواره این پرسش و دغدغه وجود دارد که چگونه می‌توان اطمینان داشت که قوانینی که کاربر تعریف می‌کند با یکدیگر تناقض نداشته باشند. این موتور خط‌مشی نیز همین مسئله روبه‌رو بوده و نیاز است روش مناسبی برای این موضوع پیش‌بینی شود. کارهایی مانند (Gouda, 2004) و (Buttyán, 2009) در زمینه سازگاری قوانین کاربر در دیوار آتش شبکه روش ارائه نموده‌اند، حتی به صورت عمومی‌تر، با توجه به استفاده از رویکرد توصیف فرمال برای توصیف قوانین موتور خط‌مشی، می‌توان از قضایای معروف سازگاری موجود در منطق مرتبهٔ اول

که در (Halbeisen, 2011) آمده است، برای کنترل و اثبات سازگاری یا عدم سازگاری قوانین تعریف‌شده توسط کاربر استفاده مناسب نمود.

۷-۳-۲. چگونگی عملکرد سیستم

عملکرد سیستم به این ترتیب است که تمام کنترل‌کننده‌ها به صورت مداوم در حال دریافت ورودی‌ها از حسگرهای مختلف و انجام محاسبه‌های موردنیاز برای تطبیق آنها با قوانین تعریف‌شده در موتور خط‌مشی هستند. به ازای هر تغییر در حسگرها یا هر تغییر در قوانین موتور خط‌مشی، این محاسبه‌ها یک بار از ابتدا توسط کنترل‌کننده‌های مرتبط انجام می‌شود. شبه‌کُد این فرایند در شکل ٤ آمده است.

شکل ٤. شبه کُد تطبیق تغییرات در حسگرها و کنترل عملیات کنترل‌کننده‌ها بر اساس قوانین موتور خط مشی

```
R={set of policy engine rules (R_j)}
Security Controller:
{
  for each change in Sv_i
    for each R_j ∈ R
      if (Sv_i <-Δ-> (R_j → p)) ⇒ do (R_j → q)
}
Reliability Controller:
{
  for each change in Rv_i
    for each R_j ∈ R
      if (Rv_i <-Δ-> (R_j → p)) ⇒ do (R_j → q)
}
Integrated Controller:
{
 for each change in Sv_i or Rv_i
    for each R_j ∈ R
      if ( (Sv_i, Rv_i) <-Δ-> (R_j → p)) ⇒ do (R_j → q)
}
```

در این شبه‌کُد، هر R_j یکی از قوانین موتور خط‌مشی است که به صورت گزاره‌های مقدم و تالی و با نماد $R_j:P_j \Rightarrow q_j$ نوشته می‌شود. اگر خواسته شود از هر R_j به گزاره‌های مقدم و تالی آن اشاره گردد، از نماد $R_j \rightarrow p$ و $R_j \rightarrow q$ استفاده می‌شود. رابطه $\overset{\Delta}{\leftrightarrow}$ نشان می‌دهد که تغییر ایجاد شده در ورودی‌های Sv_i یا Rv_i در بخش شرط از قانون R_j یعنی $R_j \rightarrow p$ صدق می‌کند یا خیر. اگر چنین باشد آن گاه تصمیم به اجرای $R_j \rightarrow q$ خواهد گرفت.

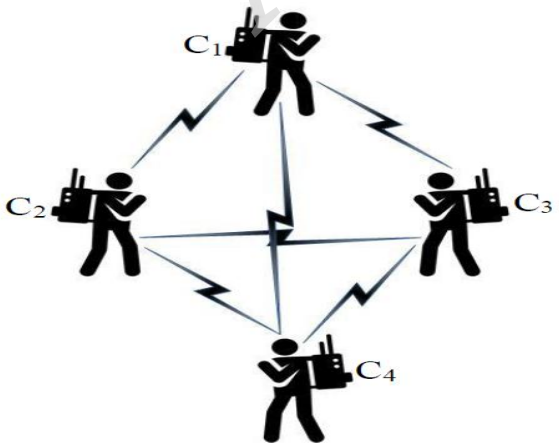
۳. یافته‌های تحقیق

برای اینکه بتوان اطمینان حاصل نمود که این مدل قابلیت لازم را برای مدیریت امنیت و قابلیت اطمینان در سیستم‌های نهفته حیاتی دارد، یک چارچوب شبیه‌سازی برای آن پیشنهاد می‌گردد. سپس با خط‌مشی‌های پیشنهادی و کنترل آنها، چگونگی ارتقای توأمان امنیت و قابلیت اطمینان نشان داده می‌شود.

۱-۳. طراحی سیستم شبیه‌سازی

چارچوب پیشنهادی شبیه‌سازی با توجه به کاربردهای معمول فرماندهی و کنترل، یک شبکهٔ چهار گره از سامانه‌های ارتباطی است که با یکدیگر دارای ارتباط مستقیم هستند و در نتیجه، یک هم‌بندی شبکهٔ کامل می‌دهد. شکل ۵ این مثال را نشان می‌دهد.

شکل ۵. شبکه مورد مطالعه



شاخص‌هایی که برای کنترل و مدیریت امنیت و قابلیت اطمینان در این سامانه‌ها در نظر گرفته شده در جدول‌های ۲ و ۳ آمده است.

جدول ۲. شاخص‌های امنیتی شبیه‌سازی

عنوان شاخص	شرح شاخص	نماد
ترافیک ورودی	میزان ترافیک ورودی به سامانه	TR_{in}
ترافیک خروجی	میزان ترافیک خروجی از سامانه	TR_{out}
پورت‌ها	پورت‌های نرم‌افزاری هدف در سامانه	Pr_i
سرویس‌ها	خدمات فراخوانی شده در سامانه	Sr_x
حافظه	تعداد دفعات دسترسی به حافظه	RAC
گزارش حمله	گزارشات حمله‌ها از IDS	Alert
فایل مشکوک	گزارش فایل‌های مشکوک از ضدبدافزار	Fi_{susp}

جدول ۳. شاخص‌های قابلیت اطمینان شبیه‌سازی

عنوان شاخص	شرح شاخص	نماد
زمان ping	زمان پاسخ به درخواست ping	T_{ping}
مصرف انرژی	میزان مصرف توان	Pw
ترافیک ورودی	میزان ترافیک ورودی به سامانه	TR_{in}
ترافیک خروجی	میزان ترافیک خروجی از سامانه	TR_{out}
میزان کارکرد CPU	تعداد متوسط اجرای دستورات در CPU	INS_{CPU}

۲-۳. اجرای شبیه‌سازی

برای انجام شبیه‌سازی لازم است در سیستم مورد نظر در شکل ۵، تعدادی خطمشی برای هر یک از کنترل‌کننده‌ها پیشنهاد نمود. این خطمشی‌ها با توجه به انعطاف بالایی که در مدل پیشنهادی وجود دارد، باید در هر کاربردی متناسب با تهدیدها و دغدغه‌های آن مسئلهٔ خاص طراحی شوند. برای بررسی عملکرد سیستم تعدادی خطمشی به‌نسبت ساده یا کمی پیچیده در نظر گرفته شده است. بدیهی است می‌توان با رعایت قواعد نحوی ارائه‌شده خطمشی‌هایی با پیچیدگی بسیار بیشتر را برای هر یک از کنترل‌کننده‌ها در نظر گرفت. خطمشی‌های در نظر گرفته شده برای این نمونه در شکل ٦ نشان داده شده‌اند. در این شکل خطمشی‌های مربوط به کنترل‌کنندهٔ قابلیت اطمینان، کنترل‌کنندهٔ امنیت و کنترل‌کنندهٔ یکپارچه از یکدیگر تفکیک شده‌اند.

برای انجام شبیه‌سازی گزینه‌های مختلفی بررسی شده است. محیط‌های مختلف شبیه‌سازی مانند MATLAB، NS2 و OPNET می‌توانند گزینه‌های مناسبی برای اجرای شبیه‌سازی باشند. هر سه گزینه امکانات مناسب و مورد نیاز برای این شبیه‌سازی را دارا هستند. به دلیل آشنایی مخاطبان و فراگیر بودن بیشتر، محیط MATLAB برای این منظور انتخاب شده است.

در محیط MATLAB، هر گره ارتباطی را به صورت یک شیء مجزا با نام Ci و شاخص‌های معرفی‌شده در جدول‌های ۲ و ۳ به صورت دو بُردار جداگانه تعریف شده است.

شکل ٦. خط مشی‌های مورد استفاده در مطالعهٔ موردی

Policy Engine:

- **Security Controller Policies:**

P1: $if\ C_i \rightarrow TR_{in} \geq 30Mbps \Rightarrow block\ (C_i \rightarrow TR_{in})\ for\ 10\,\text{seconds}\ ;$

P2: $if\ C_i \rightarrow TR_{in} \geq 10Mbps\ \&\&\ C_{j \neq i} \rightarrow Fi_{susp} = 1 \Rightarrow block\ (C_j \rightarrow TR_{out})\ ;$

P3: $if\ \left|C_i \rightarrow PR\right| > \max\_number\_of\_ports \Rightarrow restart\ (C_i)$; $C_i \rightarrow PR = \{C_i \rightarrow \Pr_k \mid \forall k = 1 \ldots 2^{16}\}$

P4: $if\ C_i \rightarrow RAC > 1000 \Rightarrow alert\ \ to\ user\ \ for\ overloading\ C_i\ ;$

P5: $if\ C_i \rightarrow Fi_{susp} = TRUE \Rightarrow shutdown\ \&\ replace\ (C_i)\ ;$

P6: $if \sum_i C_i \rightarrow TR_{in} \neq \sum_i C_i \rightarrow TR_{out} \Rightarrow Alert\ (potential\ \ cyber\ \ threat\ to\ C_i)\ ;$

P7: $if \dfrac{\sum_x C_i \rightarrow Sr_x}{\sum_y C_i \rightarrow \Pr_y} > 3 \Rightarrow C_i \rightarrow Alert\ ;$

- **Reliability Controller Policies:**

P8: $if\ C_i \rightarrow T_{ping} > 10ms \Rightarrow restart(C_i);\ ;$

P9: $if\ C_i \rightarrow Pw > 10\,WPH \Rightarrow replace\ \ (C_i)\ ;$

P10: $if\ C_i \rightarrow INS_{CPU} \geq 10MFLOPS \Rightarrow restart\ \ (C_i)\ ;$

P11: $if \dfrac{C_i \rightarrow TR_{in} + C_i \rightarrow TR_{out}}{C_i \rightarrow INS_{CPU}} > some\_threshhold \Rightarrow restart\ (C_i)\ ;$

- **Integrated Controller Policies:**

P12: $if \dfrac{C_i \rightarrow INS_{CPU}}{C_i \rightarrow RAC} < some\_threshhold \Rightarrow C_i \rightarrow Alert\ (Suspicious\ Behavior)\ ;$

P13: $if\ C_i \rightarrow Fi_{susp} = TRUE \Rightarrow shutdown\ (C_i)\ ;$

P14: $if\ \left\|C_i \rightarrow Sv\right\| + \left\|C_i \rightarrow Rv\right\| \leq some\_threshhold \Rightarrow shutdown\ \&replace\ (C_i)\ ;$

شکل ۷ شبه کُد پیاده‌شده در محیط MATLAB را نشان می‌دهد.

شکل ۷. شبه کد شبیه‌سازی

```
for each event E in Ci
  for each rule Pi
      if (E in Pi→p)  do Pi→q
```

خط‌مشی‌های P1 تا P14 با توجه به سهولت فهم و عددی بودن تمام مقادیر مورد مقایسه، نیاز به توضیح چندانی ندارند و به‌راحتی قابل تبدیل به کُد نرم‌افزاری در محیط MATLAB بوده‌اند، از این‌رو برخی از آزمون‌هایی که در اثر اعمال این قوانین به شبکه در طول شبیه‌سازی انجام شده، ارائه می‌گردد؛ برای مثال افزایش ترافیک ورودی C1 به حدود ۵۰ Mbps باعث شد تا بر مبنای قانون P1 به مدت ۱۰ ثانیه تمام ورودی‌های این جزء بسته شوند، در عین حال این میزان برای فعال‌سازی شرط مندرج در خط‌مشی P11 کافی نبود. برای آنکه شرط قانون P11 برآورده شده و در نتیجه، این قانون، عملیاتی شود، با توجه به تعداد دستورات قابل پردازش جزء C1 که حدود ۱ MFLOPS فرض شده و عدد آستانه ۱۰۰، باید ترافیک به حدود ۲۰۰ Mbps افزایش می‌یافت که در این حالت، شرط قانون P11 برقرار و در نتیجه دستور شروع مجدد C1 صادر شد. سایر ورودی‌ها و اجزای بردارهای Sv و Rv نیز به همین

ترتیب، قابل آزمون و ارزیابی هستند. یک کاربرد مهم این شبیه‌سازی‌ها، تکرار زیاد آنها در شرایط مختلف به‌منظور تنظیم دقیق اعداد آستانه برای قوانینی مانندP2، P11 و P14 با هدف کنترل و مدیریت بهینهٔ سیستم و کاهش حداکثری هشدارهای اشتباه می‌باشد.

۳-۳. مزایای مدل پیشنهادی

در مورد ویژگی‌های مدل پیشنهادی می‌توان به موارد زیر اشاره کرد.

ایدهٔ موجود در معماری Willow را که برای سامانه‌های توزیع‌شده بزرگ مطرح شده و برای سامانه‌های منفرد طراحی نشده را به سامانه‌های فرماندهی و کنترل خورانده است. این مسئله باعث می‌شود تا سامانهٔ حیاتی موردنظر – که عملکرد صحیح و امن آن بسیار مهم و تعیین‌کننده است- با تداوم و مقاومت بیشتری به عملکرد خود در شرایط بحرانی ادامه دهد.

خط‌مشی‌های صوری می‌توانند به صورت استاندارد با استفاده از یک زبان توصیف رسمی استاندارد مانند Z، VDM، ... با قواعد مشخص تعریف و کنترل شوند.

از تحقیقات وسیعی که در حوزهٔ بازپیکربندی سامانه‌ها انجام شده است، می‌توان برای غنای بیشتر مدل استفاده نمود.

یکپارچه‌سازی تحلیل امنیت و قابلیت اطمینان و مدیریت آنها بر اساس بازپیکربندی کل سامانه یا اجزای آن، ویژگی اساسی این مدل محسوب می‌شود. این موضوع به شدت در حوزهٔ فرماندهی و کنترل اهمیت دارد تا بتوان حداکثر محافظت را از عملکردهای حیاتی و اصلی سامانه در شرایط بحران انجام داد.

تفکیک انجام‌شده در ساختار کنترلی سامانه، مزایایی را ایجاد نموده است که عبارتند از:

کنترل‌کننده‌های تخصصی امنیت و قابلیت اطمینان، می‌توانند بررسی‌های تخصصی دقیق‌تر و عمیق‌تری را روی وضعیت هر یک از زیرسامانه‌ها از نگاه خود انجام دهند.

برخی عکس‌العمل‌های حیاتی و فوری از نظر امنیت و قابلیت اطمینان در برخی حالات نیاز خواهند بود که می‌توانند به صورت یک مجموعه خط‌مشی ویژه و با اولویت بالا در کنترل‌کننده‌های تخصصی تعریف شوند و عکس‌العمل حیاتی را در حداقل زمان انجام دهند. برای این کار می‌توان برخی از ویژگی‌های امنیت و قابلیت اطمینان اجزاء را به‌عنوان ویژگی‌های حیاتی در نظر گرفت –که البته برای هر زیرسامانه متفاوت خواهد بود- و آنها را با علامت و مشخص نمود و سپس برای آنها در موتور خط‌مشی، قوانین ویژه و با اولویت بالا تعریف کرد.

با وجود این، تفکیک کنترل‌های اختصاصی در کنترل‌کننده‌های تخصصی انجام می‌شود و کنترل‌کننده یکپارچه می‌تواند به صورت همزمان و همروند کنترل‌های هوشمندتر و دقیق‌تری را به شکل بی‌درنگ انجام دهد و خرابی‌ها یا حمله‌های پیچیده‌تر و پیشرفته‌تر را در زمان کوتاه‌تر و با دقت بالاتری تشخیص دهد و کنترل کند، همچنین می‌توان از تعارض‌های احتمالی موجود در داخل هر کنترل‌کننده تخصصی یا بین دو کنترل‌کنندهٔ تخصصی، پیشگیری نمود و مانع از بروز وضعیت‌های بغرنج از قبیل شرط مسابقه یا قفل مرگ در کل سامانه شود. این تعارض‌ها با توجه به بیان قوانین با زبان‌های توصیف رسمی با

استفاده از تحقیقاتی مانند (Bernard, 2010, Davy, 2008) و (Thanasegaran, 2009) قابل مدیریت و پیشگیری یا درمان هستند.

۴. نتیجه‌گیری

مشکل مهم سامانه‌های فرماندهی و کنترل مانند سامانه‌های مدیریت بحران یا سامانه‌های نظامی، این است که این سامانه‌ها هم از نظر قابلیت اطمینان نیازمند ویژگی‌های قدرتمند و متعالی هستند و هم از نظر امنیتی و حمله‌های سایبری در معرض تهدیدها و سوءاستفاده‌های متنوعی قرار دارند، از این‌رو وجود چارچوبی که بتواند تهدیدها و مخاطرات هر دو حوزه را به صورت توأمان و یکپارچه مدیریت کند، می‌تواند در راستای محافظت از عملکرد کلی سامانه بسیار مفید و مؤثر باشد. در این مقاله به معرفی یک چارچوب پایه برای چگونگی مدیریت ترکیب‌شده و یکپارچهٔ امنیت و قابلیت اطمینان در این سامانه‌ها بر مبنای بازپیکربندی پرداخته شد. برای تکمیل و ارتقای این مدل می‌توان چند تحقیق مکمل انجام داد.

ابتدا اینکه همهٔ زیرسامانه‌ها دارای ارزش و اولویت یکسانی نیستند و باید نوعی سطح‌بندی و تفکیک بین آنها ایجاد شود. دوم اینکه به پیشنهادی برای انتخاب زبان مناسب توصیف رسمی برای نوشتن خط‌مشی‌ها و اعمال آنها در کنترل‌کننده‌ها نیاز است. در ادامه، چگونگی تشخیص اینکه یک کمیت از حد مجاز خود عبور کرده یا خیر و چگونگی محاسبه و استخراج کمیت‌ها، مسئله‌ای ویژه و تخصصی است که نیاز به یک معماری مجزا و هوشمند دارد و به‌احتمال زیاد از ترکیب روش‌های مختلفی مانند تشخیص هجمه، ضدبدافزار و بلوک دیاگرام قابلیت اطمینان و ... می‌توان برای ایجاد آن استفاده نمود. این معماری خود می‌تواند بر حسب روش خواندن و ارائهٔ داده‌های برداری به کنترل‌کننده‌های تخصصی به دو گروه کلی مبتنی بر رویداد و مبتنی بر زمان تقسیم شود.

ارائهٔ روشی هوشمند و خودکار برای کنترل سازگاری و عدم تعارض قوانین موتور خط‌مشی با یکدیگر نیز می‌تواند بهبود مؤثری در مدل ارائه‌شده ایجاد کند.

فهرست منابع

1. Banerjee A, et al (2012), "Ensuring Safety, Security, and Sustainability of Mission-Critical Cyber–Physical Systems", ***Proceedings of the IEEE***, Vol. 100, No. 1, pp. 283-299.
2. Bernard, S, Matwin S, Felty, A (2010), "Strategies for Reducing Risks of Inconsistencies in Access Control Policies", ***IEEE Int. Conf***, on Availability, Reliability, and Security (ARES).
3. Brotby, K, and Hinson, G (2013), ***Pragmatic Security Metrics: Applying Metametrics to Information Security***, New York, CRC Press.
4. Buttyán, L, Gábor, P, Thong, T (2009), "Consistency Verification of Stateful Firewalls is not harder than the Stateless Case", ***Infocommunications Journal***, Vol. 64, No. 1, pp. 2-8.
5. CBS News (2010), ***http://www.cbsnews.com/news/toyota-unintended-acceleration-has-killed-89***
6. Common Criteria Standard (2017), Version 3.1, Release 5, available at: http://www.commoncriteriaportal.org/
7. Davy, S, Brendan, J, Strassner, J (2008), "Using an Information Model and Associated Ontology for Selection of Policies for Conflict Analysis", ***IEEE Workshop on Policies for Distributed Systems and Networks***.
8. Eusgeld, I, Freiling, F, and Ralf, H. eds (2008), Dependability Metrics: GI-Dagstuhl Research Seminar, Dagstuhl Castle, Germany, October 5-November 1, 2005, ***Advanced Lectures***, Vol. 4909. Springer.
9. G¨artner, F, Butty´an, L, Kursawe, K (2003), Dependable Systems: Podsy Workshop Report - From Fault Tolerance to Security and Back, ***IEEE Distributed Systems Online***, Vol. 4, No. 9.
10. Gates, C, et al (2015), "24 14491–Socio-Technical Security Metrics", ***Socio-Technical Security Metrics***, pp. 1-21.
11. Goertzel, K (2009), "Software Survivability: where Safety and Security Converge", ***The Journal of Defense Software Engineering***, Vol. 22, No. 6, pp. 15-19.
12. Gouda, G, X-YA Liu. (2004), "Firewall design: Consistency, Completeness, and Compactness", Proceedings", ***24th IEEE International Conference on Distributed Computing Systems***.
13. Halbeisen, L (2011), "First Order Logic in Nutshell", in: Combinatorial Set Theory: with a Gentle Introduction to Forcing, ***Springer Science & Business Media***, Springer, Berlin, pp 27-39.
14. Hoepman, H (2003), Security, Fault-Tolerance and their Verification for Ambient Systems. In: Gritzalis, D., di Vimercati, S.D.C., Samarati, P., Katsikas, S. (eds.), Security and Privacy in the Age of Uncertainty, ***IFIP TC11 18th Int. Conf. on Information Security (SEC2003)***, Athens, Greece, Kluwer Academic Publishers, pp. 441–446.
15. Knight, J, et al (2002), "The Willow Architecture: Comprehensive Survivability for large-scale Distributed Applications", ***Proc. of IEEE/IFIP Int. Conf. on Dependable Systems and Networks***, USA.
16. Leveson, N, Turner, C (1993), "An Investigation of the Therac-25 Accidents", ***IEEE Computer***, Vol. 26, No. 7, pp. 18-41.
17. Lions, J. (1996), Ariane 5 flight 501 failure, Available at: ***http://www.di.unito.it/~damiani/ariane5rep.html***.
18. Markoff, J (2009), Siberian Pipeline Sabotage, available at: http***://en.wikipedia.org/wiki/Siberian_pipeline_sabotage***
19. Mazurkiewicz, J (2016), Dependability Metrics for Network Systems—Analytical and Experimental Analysis, In: ***Proceedings of the 17th International Conference on Dependability and Complex Systems DepCoS-RELCOMEX***, pp. 343-354.
20. Meadows, C (1996), Applying the Dependability Paradigm to Computer Security. In: ***Proceedings of the 1995 New Security Paradigms Workshop***, IEEE Computer Society Press, Los Alamitos, pp. 75-79.
21. Meedeniya, I, et al. (2011), "Reliability-driven Deployment Optimization for Embedded Systems", ***Journal of Systems and Software***, Vol. 84, No. 5, pp. 835-846.
22. Merseguer, J, & Bernardi, S (2012), Dependability Analysis of DES based on MARTE and UML State Machines Models, ***Discrete Event Dynamic Systems***, 22 (2), 163-178.
23. Pfitzmann, A (2004), Why Safety and Security should and Will Merge, ***Proceedings of SAFECOMP 2004***. LNCS, Springer, Heidelberg, pp 1–2.
24. Rubino, G (2010), "Evaluating Dependability Metrics of Critical Systems: Monte Carlo Techniques for Rare Event Analysis", In: ***Tutorial in the Third International Conference on Dependability***.
25. Sanders, H (2014), "Quantitative Security Metrics: Unattainable Holy Grail or a Vital Breakthrough within Our Reach?", ***IEEE Security & Privacy***, Vol. 12, No. 2, pp. 67-69.
26. Thanasegaran, S, et al. (2009), "A Topological Approach to Detect Conflicts in Firewall Policies", ***IEEE International Symposium on Parallel & Distributed Processing (IPDPS)***.
27. Vijayaraghavan, V, Sanjoy P, Rajarathnam N. (2010), "iMeasure Security (iMS): A Framework for Quantitative Assessment of Security Measures and its Impacts", ***Information Security Journal: A Global Perspective***, Vol. 19, No. 4, pp. 213-225.
28. Wolf, W (2012), "Computers as Components: Principles of Embedded Computing System Design", ***Elsevier***, pp. 260-264.
29. Yasasin, E, Guido, S (2015), ***Requirements for IT Security Metrics–An Argumentation Theory Based Approach", Twenty-Third European Conference on Information Systems (ECIS)***, Münster, Germany.