پنجمین کنفرانس انجمن رمز ایران
انجمن رمز ایران
Iranian Society of Cryptology
۲۴ الی ۲۶ مهر ماه ۱۳۸۶ دانشگاه علم و صنعت ایران

وارسی کنترل دسترسی در ترکیب وب‌سرویس‌ها به کمک مدل حفاظتی Take-Grant

زهرا درخشنده
دانشگاه اصفهان
derakhshandeh@eng.ui.ac.ir

بهروز ترک لادانی
دانشگاه اصفهان
ladani@eng.ui.ac.ir

ناصر نعمت‌بخش
دانشگاه اصفهان
nemat@eng.ui.ac.ir

چکیده: ترکیب وب‌سرویس‌ها یکی از نیازمندی‌های کلیدی جهت پاسخگویی به نیاز کاربرانی است که درخواست‌های آنها به‌وسیله وب-سرویس‌های منفردِ موجود، برآورده نمی‌شود. تحلیل نیازمندی‌های غیرعملکردی ترکیب، به‌منظور تضمینِ برآوردن سیاست‌های وب‌سرویس‌های شرکت‌کننده و عدم نقض یک سیاست توسط سیاست‌های دیگران، یکی از نیازهای پایه‌ای است که برای اطمینان از عملکرد درست و امنیت وب-سرویس ترکیبی الزامی است. در این مقاله با بهره‌مندی از مدل حفاظتی *Take-Grant(TG)* به ارائهٔ روشی برای وارسی کنترل دسترسی در وب-سرویس‌های ترکیبی پرداخته‌ایم. در روش ارائه شده، توصیف مدل امنیتی وب‌سرویس ترکیبی و بررسی وجود یا عدم وجود تغییر یا نادیده‌گرفتن سیاست‌های امنیتی وب‌سرویس‌های شریک توسط سیستم ترکیبی، امکان‌پذیر خواهد بود. به‌علاوه نحوهٔ توصیف مدل امنیتی وب‌سرویس ترکیبی و وارسی سیاست‌های امنیتی آن در قالب یک مثال کاربردی نشان داده شده است.

واژه های کلیدی: ترکیب وب‌سرویس‌ها، وارسی، مدل حفاظتی *Take-Grant*، کنترل دسترسی، *WSTG*

۱- مقدمه

با ظهور اینترنت و افزایش روبه رشد استفاده از آن در زمینهٔ کاربردهای شخصی، تجاری و غیره اندیشهٔ ایجاد مجموعه‌های قابل گسترشی از نرم‌افزارهای توزیع‌شده در سطح اینترنت برای بهره‌مندی متقاضیان آنها در هر موقعیت مکانی، مقدمهٔ پیدایش و توسعهٔ وب‌سرویس‌ها[1] شد. وب‌سرویس‌ها ماژول‌های نرم‌افزاری هستند که با استانداردهای باز[2] و از طریق وب، قابل دسترسی بوده و مجموعه-ای از عملکردها[3] جهت انجام کسب و کار و یا هرگونه استفادهٔ دیگری را فراهم می‌آورند[۱]. در بسیاری از موارد، درخواست کاربر به‌وسیلهٔ وب‌سرویس‌های اتمیک موجود برآورده نمی‌شود. نه تنها عاقلانه نیست که به ازای هر درخواستِ خاص، به طراحی یک وب‌سرویس پیچیده‌تر برای برآورده کردن آن نیاز بپردازیم، بلکه این امر در حالت کلی غیرممکن بوده و مقرون به‌صرفه نمی‌باشد. درحالی‌که چنین درخواستی را می‌توان به‌وسیلهٔ ایجاد ترکیبی از وب‌سرویس‌های موجود برآورده نمود[۲]. به فرآیندِ گردهم‌آوری وب‌سرویس‌های اتمیک، به‌منظور ایجاد یک مجموعهٔ یکپارچه و هماهنگِ ترکیبی برای برآوردن اهداف وسیع‌تر و پیچیده‌تر از آنچه توسط تک‌تک وب‌سرویس‌های اتمیک بدست می‌آید، ترکیب وب-سرویس‌ها[4] گوییم. ترکیب وب‌سرویس‌ها ما را وارد حیطهٔ وسیع-تری می‌کند که علاوه بر دربرگرفتن کلیهٔ نیازمندی‌های موجود در حیطهٔ وب‌سرویس‌ها، به‌واسطهٔ گردهم‌آیی و نیاز به یکپارچگی بین آنها، نیازمندی‌های بیشتری را نیز طلب می‌کند. نیازمندی‌های یک وب‌سرویس ترکیبی به دو دستهٔ عملکردی و غیرعملکردی[5] قابل

[1] Web Services
[2] Open
[3] Functional
[4] Web service composition
[5] Non-functional

تقسیم است. نیازمندی‌های عملکردی ناظر به بیان مقتضیات روند کاری سیستم حاصل از ترکیب است اما نیازمندی‌های غیرعملکردی خصوصیت‌های کیفیتی آن را دربر داشته و درگذر زمان و با تغییر علایق سازمان‌های وابسته، دچار تغییر می‌شوند. از جمله نیازمندی‌های غیرعملکردی می‌توان به قابلیت اطمینان، توسعه‌پذیری و امنیت اشاره نمود. برآوردن نیاز غیرعملکردیِ امنیت، در کنار نیازمندی‌های عملکردی از ضرورت‌های اجتناب‌ناپذیر در امر ترکیب وب‌سرویس‌ها است[۳].

BPEL زبان رایج توصیف فرآیند ترکیب وب‌سرویس‌ها است که تنها امکان توصیف بخش عملکردی فرآیند ترکیبی را به ما می‌دهد و در بیان خصوصیات غیرعملکردی فرآیند ترکیبی ناتوان است. برای گنجاندن مفاهیم امنیتی در فرآیندهای ناشی از *BPEL*، راه‌هایی پیشنهاد شده اما به جرأت می‌توان گفت به‌طورکلی در زمینهٔ امنیت ترکیب وب‌سرویس‌ها تاکنون اقدام کافی و قانع‌کننده‌ای صورت نپذیرفته و این مسئله یکی از موارد ابهام در زمینهٔ بکارگیری *BPEL* است[۴،۷]. در این زمینه اقدامات اندکی در[۱۶،۷] مشاهده می‌شود که همچنان جای کار دارد. مسئلهٔ دیگر موضوع وارسی است. وارسی یک طراحی، به این معنا است که آیا آن طراحی، خصوصیت‌هایی را که مد نظرمان است، برآورده می‌کند یا خیر. نیاز به تحلیل و وارسی ترکیب وب‌سرویس‌ها زمانی به میان می‌آید که می‌خواهیم نتیجهٔ بدست‌آمده، اهداف و خصوصیت‌های مورد نظر ما از این ترکیب را برآورده کند. *BPEL* به عنوان رایج‌ترین زبان مورد استفاده در زمینهٔ ترکیب وب‌سرویس‌ها، به دلیل نداشتن یک شالودهٔ صوری[1]، امکانات لازم برای وارسی را در بر ندارد و از این‌رو با تبدیل آن به زبان‌های صوری، امکان تحلیل و وارسی آن را مهیا می‌سازند. تاکنون تحقیقاتی در زمینهٔ وارسی فرآیندهای ترکیب انجام پذیرفته است[۱۷،۱۸،۱۹،۲۰] اما این تحقیقات تنها نیازمندی‌های عملکردی ترکیب را مدنظر قرار داده‌اند. وارسی نیازهای غیرعملکردی ترکیب وب‌سرویس‌ها نیازمند توجه بیشتری است. به‌نظر می‌رسد [۱۵] از معدود مواردی است که نیازمندی‌های غیرعملکردی در ترکیب وب‌سرویس‌ها را درنظر گرفته است. در آن با استفاده از زبان *Event Calculus* به وارسی امنیت ترکیب پرداخته و امکان تشخیص نمونه‌هایی از برخوردهای[2] موجود در بین سیاست‌های شرکا را فراهم نموده است.

مسئله مهم در زمینهٔ امنیت ترکیب وب‌سرویس‌ها، این است که ممکن است وب‌سرویس‌های شرکت‌کننده در ترکیب از امنیت قابل قبولی برخوردار بوده و سیاست‌های امنیتی مشخصی را دنبال کنند، اما ترکیبی از آنها، تضمین‌کنندهٔ ارضای سیاست‌های تک تک وب‌سرویس‌ها نباشد. چه‌بسا مواردی وجود دارد که وب‌سرویس‌های شریک دارای سیاست‌های متناقضی باشند. ازاین‌رو به امکاناتی برای وارسی فرآیند ترکیب نیازمندیم.

با درنظرگرفتن وجود این نیاز در ترکیب وب‌سرویس‌ها با استفاده از *BPEL*، در این مقاله به بیان روشی برای وارسی صوری امنیت در زمینه کنترل دسترسی، در ترکیب وب‌سرویس‌ها پرداخته‌ایم. نتایج این وارسی برای اطمینان از درستی فرآیند ترکیب قابل استفاده خواهد بود. برای مدل‌سازی نیازمندی‌های کنترل دسترسی از مدل حفاظتی *Take-Grant (TG)*[3] استفاده می‌نماییم. این مدل یکی از مدل‌های مورد استفاده در زمینهٔ کنترل دسترسی است که برای بررسی چگونگی انتقال حقوق و اطلاعات در سیستم به‌کار می‌رود. در واقع ما تلاش می‌نماییم که از فرآیند *BPEL*، یک مدل حفاظتی صوری و تحلیل‌پذیر استخراج کنیم. سپس از امکانات تحلیلی مدل *TG* برای تحلیل مدلِ به‌دست آمده از فرآیند *BPEL* و وارسی کنترل دسترسی و انتقال حقوق و اطلاعات بهره‌مند می‌شویم.

در ادامهٔ مقاله، ابتدا در بخش۲ خلاصه‌ای از مفاهیم پایه در مدل *TG* را از نظر می‌گذرانیم. در بخش۳ ترکیب وب‌سرویس‌ها را با توصیف زبان *BPEL* و سپس به‌طور دقیق‌تر مسائل و معایب حاکم بر این زبان و نیازمندی‌های آن را مورد بررسی قرارمی‌دهیم. در بخش۴ به توصیف قوانین کنترل دسترسی در فرآیند ترکیب وب‌سرویس‌ها پرداخته و مدل حفاظتی خود برای مدل سازی کنترل دسترسی در ترکیب وب‌سرویس‌ها تحت عنوان *WSTG*[4] و طریقهٔ استخراج این مدل از فرآیند ترکیب و نحوهٔ وارسی آن را به تفصیل بیان می‌کنیم. در بخش۵ با بررسی یک مثال موردی، نمونه‌ای از آنچه در بخش‌های قبل عنوان نمودیم را تبیین می‌نماییم.نهایتاً در بخش۶ به نتیجه‌گیری و بیان کارهای در دست اقدام خود می‌پردازیم.

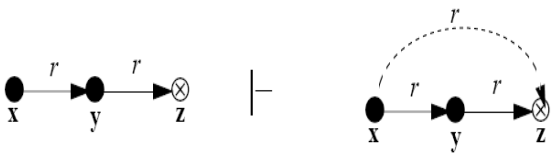
۲- مدل حفاظتی *Take-Grant*

مدل حفاظتی *TG* یک مدل صوری کنترل دسترسی است که

[1] Formal
[2] Conflict
[3] Take-Grant protection model
[4] Web Service Take Grant protection model

انتقال حقوق و اطلاعات را بین نهادها در یک سیستم مدل می‌کند. این مدل برای اولین بار توسط *Jones* و دیگران در[۵] مطرح گردید و از آن برای حل مسئله ایمنی[1] استفاده شد. در این مدل اجزای سیستم به‌وسیله یک گراف محدود مدل می‌شود. در این گراف رأسها، نهادهای سیستم و یالها حقوق رئوس نسبت به یکدیگر را مدل می‌کنند. یالها دارای برچسب‌هایی هستند که بیانگر حقوق رأس مبدا نسبت به رأس مقصدند. نهادها می‌توانند فعال و یا غیرفعال باشند. نهادهای فعال (فاعل‌ها) با ● و نهادهای غیرفعال (مفعول) با ○ نشان داده می‌شوند. رأسی که ممکن است هر دو نوع نهاد را نمایش دهد با ⊗ نشان داده می‌شود. مجموعهٔ حقوق اولیهٔ تعریف‌شده در این مدل، با مجموعه $R=\{t, g, r, w\}$ تعریف می‌شود که t و g و r و w به ترتیب حقوق *take* ، *grant* ، *read* و *write* می‌باشند[۹،۸]. برای مدل‌کردن انتقال حقوق از مجموعه‌ای از قواعد به‌نام قوانین حقوقی[2] استفاده می‌شود. این قوانین عبارتند از: *take* ، *grant* ، *create* و *remove*. و برای مدل‌کردن انتقال اطلاعات نیز از مجموعه‌ای از قواعد به‌نام قوانین عملی و غیررسمی[3] استفاده می‌شودکه عبارتند از : *post* ، *pass* ، *spy* و *find* [۸،۱۰]. برخی از قوانین این مدل که در ادامه از آنها استفاده می‌کنیم به شرح زیر هستند:

۱. *spy* : فرض کنید x، y، z سه رأس متمایز در گراف حفاظتی G_0 باشند و x و y فاعل در نظر گرفته شوند. اگر یک یال از x به y با برچسب α موجود باشد به‌طوری‌که $r\in\alpha$ و یک یال از y به z با برچسب β وجود داشته باشد به‌طوری که $r\in\beta$، این قانون گراف G_0 را با افزودن یک یال ضمنی جدید از x به z با برچسب r به گراف حفاظتی G_1 تبدیل می‌کند (شکل ۱).



شکل ۱. قانون *spy*

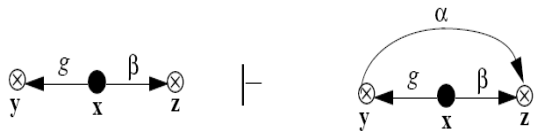
۲. *pass* : فرض کنید x، y، z سه رأس متمایز در گراف حفاظتی G_0 باشند و y فاعل در نظر گرفته شود. اگر یک یال از y به x با برچسب α موجود باشد به طوری که $w\in\alpha$ و یک یال از y به z با برچسب β وجود داشته‌باشد به طوری که $r\in\beta$ ،

[1] Safety
[2] De jure rules
[3] De facto rules

قانون *pass* گراف G_0 را با افزودن یک یال ضمنی جدید از x به z با برچسب r به گراف حفاظتی G_1 تبدیل می‌کند (شکل ۲).

شکل ۲. قانون *pass*

۳. *grant* : فرض کنید x، y، z سه راس متمایز در گراف حفاظتی G_0 باشند و x یک عامل در نظر گرفته شود. اگر یک یال از x به y با برچسب γ موجود باشد به‌طوری‌که $g\in\gamma$ و یک یال از x به z با برچسب β وجود داشته باشد به‌طوری که $\alpha\subseteq\beta$، قانون *grant* گراف G_0 را با افزودن یک یال جدید از y به z با برچسب α به گراف حفاظتی G_1 تبدیل می‌کند(شکل ۳).



شکل ۳. قانون *grant*

۳- ترکیب وب‌سرویس‌ها

در این بخش ابتدا به معرفی مختصر زبان توصیف ترکیب وب‌سرویس‌ها (*BPEL*) پرداخته، سپس نقاط ضعف امنیتی آن را بررسی می‌کنیم.

۳-۱ مروری بر زبان *BPEL*

BPEL[4] یک زبان مبتنی بر *XML* است که از ترکیب سرویس مبتنی بر فرآیند[5] حمایت می‌کند. این زبان توسط *OASIS*[6] استاندارد شده و هم‌اکنون به عنوان زبان ترکیب وب‌سرویس‌ها مورد استفاده قرار می‌گیرد[۳]. *BPEL* یک فرآیند کسب و کار را مدل می‌کند که شامل مجموعه‌ای از فعالیت‌هاست[7] [۲،۳،۱۱، ۶] و از طریق واسط *WSDL*[8] با سرویس‌ها ارتباط برقرار می‌کند[۱۲]. هستهٔ مفهومی فرآیند *BPEL* پیامهایی است که بین فرآیند و سرویس‌های شریک در آن رد و بدل می‌شود[۲]. *BPEL* چند گروه مولفه دارد که مهم‌ترین آنها در جدول ۱ آمده است [۱۳].

[4] Business Process Execution Language
[5] Process-oriented service composition
[6] Organization for the Advancement of Structured Information Standards, www.oasis-open.org.
[7] Activities
[8] Web Service Description Language

جدول ۱. تعدادی از مولفه‌های *BPEL*

نوع مولفه	*Tag* مورد استفاده
فراخوانی‌ها و احضارها	<*invoke*> ,<*reply*>,<*receive*>
دستکاری و تغییر داده‌ها	<*assign*> , <*copy*>,...
اجرای ترتیبی و موازی	<*sequence*> , <*flow*>

۳-۲- مسائل امنیتی ترکیب وب‌سرویس‌ها با *BPEL*

همان‌طور که قبلاً ذکر شد در استفاده از *BPEL*، مسائلی به‌چشم می‌خورد که به‌اندازه کافی مورد توجه قرارنگرفته‌اند. از جمله: *BPEL* یک زبان توصیف ترکیب است که تنها امکان توصیف بخش عملکردی فرآیند ترکیبی را داشته و امکاناتی را برای بیان خصوصیات امنیتی ترکیب که یک ویژگی غیرعملکردی است فراهم‌نکرده است[۴،۷،۱۴]. یکی از مشکلات امنیتی در *BPEL* این است که فرآیند ترکیبی الزاماً تضمینی برای برآورده شدن خط مشی‌های امنیتی وب‌سرویس‌های ترکیب شده ایجاد نمی‌کند. هر فرآیند ترکیبی قبل از ایجاد ترکیب، از سیاست‌های امنیتی شریکان خود آگاه می‌گردد و در حین ایجاد و اجرای ترکیب ملزم به رعایت این قیود است. اما اینکه آیا فرآیند ترکیبی می‌تواند به تمام قیود شرکای مختلف در ضمن ترکیب پایبند بماند یا نه مورد بررسی قرار نگرفته و فقدان توانایی بیان سیاست‌های امنیتی در *BPEL*، این مسئله را بیشتر دچار بحران می‌نماید[۵،۷]. در بخش بعد طرز بهره‌مندی از فعالیت‌های *BPEL* برای وارسی کنترل دسترسی را مورد بررسی قرار داده، این عمل را بدون درنظرگرفتن هیچ ملحقاتی به آن انجام می‌دهیم.

۴-وارسی کنترل دسترسی در ترکیب وب‌سرویس‌ها

۴-۱-مدل کنترل دسترسی *WSTG*

از آنجا که *BPEL* فاقد یک شالوده صوری می‌باشد، با تبدیل آن به مدل *TG*, امکان وارسی و استفاده از روشهای تحلیل بدست آمده در این مدل، برای وب‌سرویس ترکیبی نیز فراهم می‌آید. ما برای مدل‌کردن سیاست‌های کنترل دسترسی در وب‌سرویس‌ها از مدل حفاظتی *TG* استفاده می‌کنیم. مدل توسعه یافته برای این منظور را از این پس مدل حفاظتی *Web Service Take-Grant(WSTG)* می‌نامیم. *WSTG* را به صورت زیر تعریف می‌کنیم:

اگر S مجموعهٔ فاعلها، O مجموعهٔ اشیاء و R مجموعهٔ حقوق ممکن باشد، گراف *WSTG* مجموعه‌ای از سه‌تایی‌های (V,L,R) می‌باشد به‌طوریکه: $V \subseteq S \cup O$ مجموعهٔ رئوس، L مجموعهٔ یالها است که $L \subseteq V \times V \times R$ و $R=\{r,w,x,o,t,g\}$ ، که درآن حقوق r,w,t,g همچون گذشته و o , x به‌صورت زیر تعریف می‌شوند:

- ***X*** *(execute)* : حق اجرای یک فاعل نسبت به یک مفعول (مانند اجرای یک عملیات در وب‌سرویس). در برخورد با حق اجرای یک فاعل نسبت به یک وب‌سرویس، بسته به نقش آن فاعل نسبت به آن وب‌سرویس و نوع اجرای مدنظر آن، این حق به یکی از حقوق *read* یا *write* تفسیر و سپس همانند حقوق r و w با آنها برخورد می‌شود. در ادامه بیشتر با چگونگی این امر آشنا می‌شویم.
- ***O****(ownership)* : حق مالکیت یک فاعل نسبت به یک مفعول. در *WSTG*، O حق مالکیت یک وب‌سرویس نسبت به عملیاتش می‌باشد که می‌تواند طی شرایطی به مولفه‌های دیگر سیستم داده شود. در اینجا به‌منظور حفظ سادگی، هر وب‌سرویس را قادر به ارائهٔ یک نوع خدمت فرض می‌کنیم. هرچند گسترشِ آن به‌حالت چندمنظوره، آسیبی به صحت موضوع مورد بحث وارد نخواهد کرد.

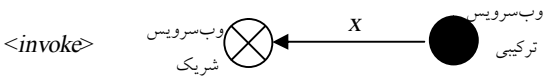
در*WSTG*، وب‌سرویس‌ها توسط رئوس گراف مدل می‌شوند و حقوق هر وب‌سرویس نسبت به منابع یا وب‌سرویس‌های دیگر، توسط یالهای برچسب‌دار مشخص می‌شود. لذا در حالت کلی سیاست‌های هر وب‌سرویس نسبت به منابع مورد استفاده و روند ارتباطات و عملکرد آن نسبت به وب‌سرویس‌های دیگر توسط یک گراف *WSTG* با حداقل یک رأس مشخص‌می‌گردد. نقض کنترل دسترسی در یک گراف، حاکی از مجموعه شرایطی است که باعث انتقال غیرمجاز و مغایر با سیاست‌ها و حقوق دسترسیِ پایهٔ آن وب‌سرویس(ها) می‌شود. در ادامه نحوهٔ استخراج مدل *WSTG* از روی فرآیند ترکیبی *BPEL* وسپس نحوهٔ وارسی کنترل دسترسی در این مدل را تشریح می‌کنیم.

۴-۲- استخراج مدل *WSTG* از *BPEL*

BPEL از طریق یک مجموعه دستورات و ساختارها به ایجاد ارتباط بین وب‌سرویس‌ها پرداخته و نتایج این ارتباط را در نهایت در اختیار کاربر درخواست‌کننده قرار می‌دهد. در زمینهٔ کنترل دسترسی فعالیت‌هایی که فرآیند *BPEL* را با محیط خارج از خود درگیر می‌نماید، مورد نظرند؛ چرا که انتقال حقوق و اطلاعات از این طریق میسر می‌گردد. فعالیت‌های احضار وب‌سرویس‌ها (*invoke*)، دریافت درخواست (*receive*) و پاسخ‌گویی به درخواست

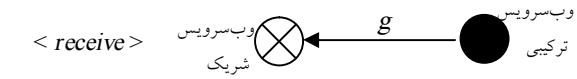
(*reply*)، پل‌های ارتباطی فرآیند ترکیبی با ارکان دیگر هستند.

- *Invoke* : وب‌سرویس ترکیبی با احضار وب‌سرویس‌های شریک در واقع حق اجرای آنها را در دست دارد.این امر با یک یال از فرآیند ترکیبی به سمت وب‌سرویس احضار شده که با x برچسب خورده است، مشخص می‌شود. همان‌طورکه در بخش ۴-۱ مطرح شد حق اجرای یک رأس توسط فرآیند ترکیب به این معنی است که اطلاعاتی در اثر اجرای آن وب-سرویس به فرآیند ترکیبی منتقل می‌شود و یا اطلاعاتی از فرآیند ترکیبی به وب‌سرویس مورد نظر برای پردازش ارسال می‌گردد (شکل۴). این حالت‌ها به ترتیب حق خواندن وب‌سرویس و حق نوشتن در وب‌سرویس توسط ترکیب را دربردارد. به زبان دیگر حق x، بسته به عملیاتی که فرآیند ترکیبی از وب‌سرویس انتظار دارد و خدماتی که وب‌سرویس به ترکیب ارائه می‌کند، به یکی از حقوق r یا w یا هر دوی آنها تبدیل می‌شود.



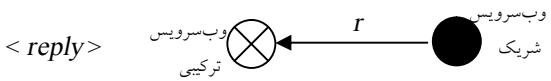
شکل ۴: تبدیل *invoke* به مدل *WSTG* معادل

- *receive* : عملیات انتظار وب‌سرویس ترکیبی برای دریافت درخواست است. یک فرآیند ترکیبی از طریق اجرای این دستور در واقع وکالت انجام ارتباطات خود با وب‌سرویس‌های شریک را به‌طور ضمنی به کاربر درخواست‌کننده می‌بخشد. به زبان دیگر وب‌سرویس ترکیبی با اجرای دستور مذکور، آمادگی خود را برای بخشیدن حق اجرای وب‌سرویس‌های شریک به درخواست‌کننده اعلام می‌کند. این عملیات به‌وسیلهٔ حق g *(grant)* مشخص می‌شود که حاکی از بخشیدن حقوق وب-سرویس ترکیبی، به درخواست‌کنندهٔ آن است (شکل۵).



شکل ۵: تبدیل *receive* به مدل *WSTG* معادل

reply : وب‌سرویس ترکیبی نتیجهٔ عملیات درخواست‌شده توسط درخواست‌کننده را به وی ارسال می‌کند که معادل حق خواندن (r) درخواست‌کننده نسبت به وب‌سرویس ترکیبی می‌باشد (شکل ۶). در بخش بعد طرز وارسی مدل بدست آمده را بیان می‌کنیم.



شکل۶: تبدیل *reply* به مدل *WSTG* معادل

۴-۳- وارسی کنترل دسترسی در مدل *WSTG*

پس از بیان چگونگی استخراج و تبدیل فعالیت‌های مربوط به کنترل دسترسی در *BPEL* به *WSTG* در بخش۴-۲، اکنون زمان آن رسیده است که روند کلی فرآیند وارسی را بیان نماییم.

ابتدا سیاست‌های امنیتی تک تک وب‌سرویس‌های شرکت-کننده در ترکیب را مبتنی بر مدل*WSTG* بیان می‌نماییم. سپس با بررسی چگونگی تعامل وب‌سرویس ترکیبی با وب‌سرویس‌های دیگر، به استخراج قوانین کنترل دسترسی ناشی از ترکیب پرداخته و مدل *WSTG* حاصل از ترکیب را بدست می‌آوریم. با استفاده از این مدل و با بهره‌مندی از روش‌های گوناگون تحلیلی بدست آمده در مدل *TG* از قبیل [۸،۹،۱۰] به وارسی و تحلیل کنترل دسترسی در ترکیب وب‌سرویس‌ها و میزان پایبندی آن به سیاست‌های شرکا می‌پردازیم.

وارسی کنترل دسترسی در مدل *TG* به صورت کلی با بکارگیری مسندهایی برای بررسی چگونگی انتقال حقوق و اطلاعات تعریف شده انجام می‌شود. برخی از این مسندات و نحوه استفاده از آنها به‌طور خلاصه در ادامه ذکر شده اند:

- $can.know(x,y,G_0)$: این مسند دارای نتیجهٔ درست است اگر در گراف حفاظتی G_0، رأس x بتواند اطلاعات را از رأس y پس از اعمال یک سری قانون غیررسمی بدست‌آورد. در واقع اگر یک سری گراف حفاظتی G_0 ,..., G_n موجود باشد به‌طوریکه G_n توسط اعمال قوانین از G_0 مشتق شده باشد و یک یال از x به y در G_n موجود باشد، مسند مقدار درستی خواهد داشت.
- $can.share(r,x,y,G0)$: همانند مسند *can.know* است با این تفاوت که تنها مبتنی بر اعمال قوانین حقوقی و نه قوانین غیررسمی (بخش۲) بررسی می‌شود.
- $can.steal(r,x,y,G_0)$: این مسند در صورتی درست است که رأس x بدون هیچ اقدام و کمکی از طرف رأس y (به عنوان مالک حق r در گراف G_0) این حق را بدست آورد. یعنی y شروع کنندهٔ هیچ قانونی در این فرآیند نباشد

و می‌گوییم x این حق را دزدیده است.

توضیح کامل این مسندها در [۸،۹،۱۰] آمده که مجال بیان آن در اینجا نمی‌گنجد. بدین وسیله، از امکانات گسترده‌ای برای تحلیل فرآیند ترکیب وب‌سرویس‌ها در زمینه‌های متفاوت بهره‌مند می‌شویم. مثلاً برای مشخص‌کردن اینکه آیا یک وب‌سرویس می‌تواند از طریق تعاملات فرآیند ترکیبی از اطلاعات وب‌سرویس دیگر مطلع شود یا خیر، می‌توانیم از مسند *can.know* استفاده کنیم. در بخش بعد نمونه‌ای از کاربرد مسند *can.know* را برای مشخص کردن اینکه آیا یک وب‌سرویس می‌تواند از طریق تعاملات فرآیند ترکیبی از اطلاعات وب‌سرویس دیگر مطلع شود یا خیر، نشان می‌دهیم.

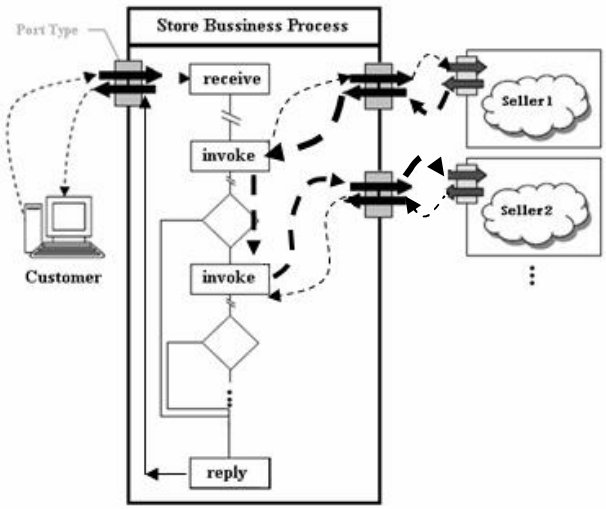
۵- بررسی موردی

تعدادی فروشنده را در نظر بگیرید که اجناسی برای فروش به مشتریان دارند. فرض کنید فروشندگان دارای محصولاتی با کیفیت‌های متفاوت هستند؛ به‌گونه‌ای که فروشندهٔ۱ دارای اجناس با کیفیت بالاتر و فروشندهٔ۲ و۳ و ... به‌ترتیب از همان اجناس با کیفیت محصول پایین‌تری برخوردارند. حال فرآیندی همانند فرآیند مذاکره[1] را در نظر بگیرید. مشتری ابتدا به فروشندهٔ۱ مراجعه می‌کند. درصورتی‌که ملاک‌های مشتری با توجه به پارامترهایی که فروشندهٔ۱ ارائه می‌کند، برآورده‌نشد، به سراغ فروشندهٔ۲ می‌رود و این عمل را تا زمانی‌که جنس مورد نظرش را با قیمت مناسب بخرد ادامه می‌دهد. از آنجا که مشتری برای فروشندهٔ۱ مورد اعتماد است اطلاعات خود را اعم از نوع محصول، مشخصات کیفی، قیمت و ... در اختیار مشتری قرار می‌دهد و اگر مشتری بر اساس شرایط موجود در پاسخ فروشندهٔ۱، راضی نشد می‌تواند مبنی بر این اطلاعات درخواست مناسبی مطابق خواسته‌هایش به فروشندهٔ۲ ارائه دهد. به‌عنوان مثال مشتری ممکن است شرط کند که درصورتی‌که قیمت کالای مذکور از قیمت پیشنهادی توسط فروشندهٔ۱ که مبتنی بر فلان کیفیت و سیاست قیمت‌گذاری عنوان شده بود، کمتر باشد حاضر به خرید جنس مذکور از فروشندهٔ۲ است. مشاهده می‌شود که به‌طور ضمنی و در حین حرکتِ به‌سمت جلوی مشتری در فرآیند مذکور، اطلاعاتی همچون اطلاعات کیفی و طریقهٔ سیاست قیمت‌گذاری و حدود قیمت هر کالا در نزد فروشندگان سطح بالاتر، به فروشندگان با رده‌های کیفی پایین‌تر منتقل می‌شود. این درحالی است که اگر فروشندهٔ۲ درخواست خریدی را صریحاً به فروشندهٔ۱ بفرستد، فروشندهٔ۱ به دلیل رقابت موجود و تلاش برای بهره‌مندی از بهترین سیاست تجاری ممکن، وی را معتمد ندانسته و از پذیرش درخواست فروشندهٔ۲ و پاسخ‌دهی به آن سرباز می‌زند. از این‌رو اطلاعات زیادی در همین زمینه به‌طور ناخواسته و به‌واسطهٔ مشتری برای فروشندهٔ۲ فاش می‌شود. درحالی‌که فروشندهٔ۱ از این مسئله بی‌اطلاع و از هویت مشتری خود مطمئن است. فروشندهٔ۲ می‌تواند پس از آگاهی از حدود قیمت محصول و اطلاعاتی از این قبیل از سطح بالاتر، سیاست خود را به گونه‌ای تغییر دهد که بتواند به سود بیشتری دست یابد. رقابتی که اصولاً در حراجها به چشم می‌خورد و آگاهی از اطلاعات شرکا می‌تواند در موفقیت فروشندگان موثر باشد. اکنون این مثال را به‌وسیلهٔ وب‌سرویس‌ها بیان می‌کنیم:

هر فروشنده یک وب‌سرویس است و فرآیند مذاکره برای برآوردن درخواست مشتری توسط یک وب‌سرویس ترکیبی پیاده می‌شود. وب‌سرویس ترکیبی همان روند حرکت مشتری (که پیش از این به تفصیل عنوان شد) را دنبال می‌کند. فرآیند *BPEL* ناشی از این ترکیب به‌طور خلاصه و تنها با تاکید بر عناصر مهم در زمینهٔ کنترل دسترسی آن، در یک فلوچارت ترسیم‌شده که در شکل۷ قابل مشاهده است. وب‌سرویس ترکیبی پس از دریافت درخواست از مشتری، وب‌سرویس فروشندهٔ۱ را با ارسال درخواست مشتری به آن احضار می‌کند. پس از ارسال پاسخ از فروشندهٔ۱، مجموعه عملیاتی که شامل رد و بدل اطلاعات با مشتری و تصمیم‌گیری مشتری در مورد پذیرش یا مراجعه به وب‌سرویس بعدی است، انجام می‌شود که به‌دلیل حفظ وضوح شکل به‌طور کامل ترسیم نشده است. درصورتی‌که قیمت و شرایط فروشندهٔ۱ برای مشتری قابل قبول باشد، عمل *Reply* انجام و فرآیند پایان می‌پذیرد. اگرنه از پاسخ فروشندهٔ۱ مجموعه‌ای از اطلاعات کیفی و طریقهٔ سیاست قیمت‌گذاری و به عنوان مثال قیمت α برای کالای مورد نظر بدست می‌آید. در بهترین حالتِ ممکن، مشتری و به تبع آن وب‌سرویس ترکیبی از فروشندهٔ۲ می‌پرسد که آیا با شرایطی همچون قیمت زیر α ریال و غیره می‌تواند درخواست را برآورده

[1] negotiation

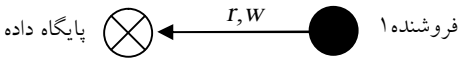
سازد. پیکان مقطع تیره در شکل ۷ حرکت داده‌های مذکور از فروشندهٔ۱ به سمت فروشندهٔ۲ را نشان می‌دهد. به این ترتیب



شکل۷: فرآیند *BPEL* مربوط به مثال موردی

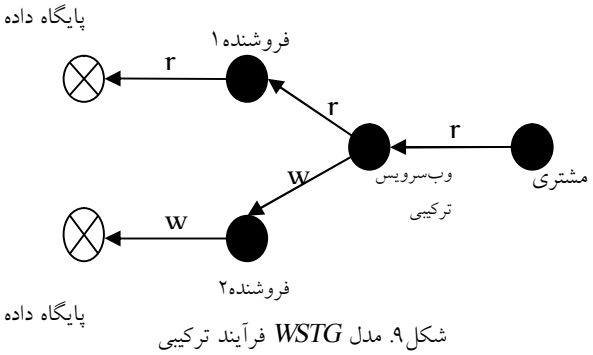
کمترین داده‌ای که فروشندهٔ۲ بدست می‌آورد این است که وب-سرویس با ردهٔ بالاتر حدود قیمتی بالای α ریال را بیان نموده است که البته این حداقل اطلاعات است. با فاش شدن چنین اطلاعاتی، فروشندهٔ۲ و به دنبال آن فروشندگان بعدی در صورت ادامهٔ فرآیند به اطلاعات خاصی دست می‌یابند که فروشندهٔ۱ با آن موافق نیست. اما عملکرد وب‌سرویس ترکیبی به‌ظاهر هیچ مشکلی نداشته و این ترکیب به‌خوبی کار می‌کند. اکنون الگوریتم پیشنهادی خود برای وارسی کنترل دسترسی را در این مثال به‌کار می‌بریم:

ابتدا سیاست‌های تک‌تک وب‌سرویس‌های شریک در ترکیب را با *WSTG*، مدل می‌کنیم. هر وب‌سرویس فروشنده، دارای پایگاه‌داده-ای است که فقط خود آن وب‌سرویس و نه دیگر شرکا می‌تواند اطلاعات آن را خوانده یا تغییر دهد ازاین‌رو مبتنی بر مدل *WSTG*، این سیاست با حق r و w وب‌سرویس فروشنده نسبت به پایگاه-داده‌اش مدل می‌شود(شکل۸). از طرفی مسند G_0(فروشگاه۱,فروشگاهi, *can. know*) (که در آن i=2,...) باید دارای مقدار نادرستی باشد. حال فرآیند *BPEL* را با *WSTG* مدل می‌کنیم: درخواست مشتری برای خرید کالا معادل حق اجرای فرآیند ترکیبی توسط وی می-باشد. این حق بسته به نوع اجرا به یکی از حقوق انتقال اطلاعات (r یا w) تبدیل می‌شود. در این مثال در واقع مشتری به خواندن قیمت و اطلاعات کیفی کالا می‌پردازد (r). به همین ترتیب حق اجرای فرآیند ترکیبی، نسبت به وب‌سرویس‌های شرکت‌کننده در ترکیب بسته به نوع ارتباط و رد و بدل داده‌اش با آنها بدست‌می‌آید. از آنجا



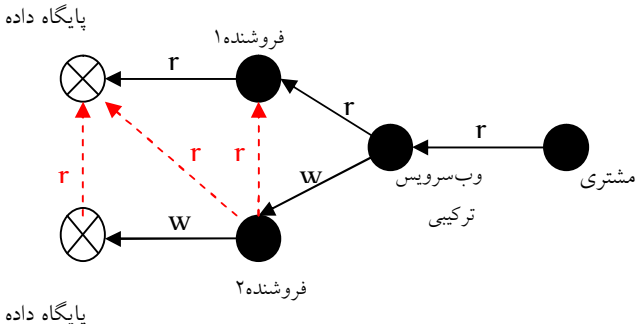
شکل۸: مدل *WSTG* نمونه سیاست‌های وب‌سرویس فروشنده۱

که حالتی مد نظر ماست که وب‌سرویس ترکیبی اطلاعات را از وب‌سرویس فروشندهٔ۱ دریافت کرده و چون درخواست مشتری را برآورده نمی‌کند، آنها را به فروشندهٔ۲ می‌فرستد، لذا حق اجرای فرآیند ترکیب نسبت به وب‌سرویس فروشندهٔ۱ به r و نسبت به وب‌سرویس فروشندهٔ۲ به w تبدیل می‌گردد(شکل۹).



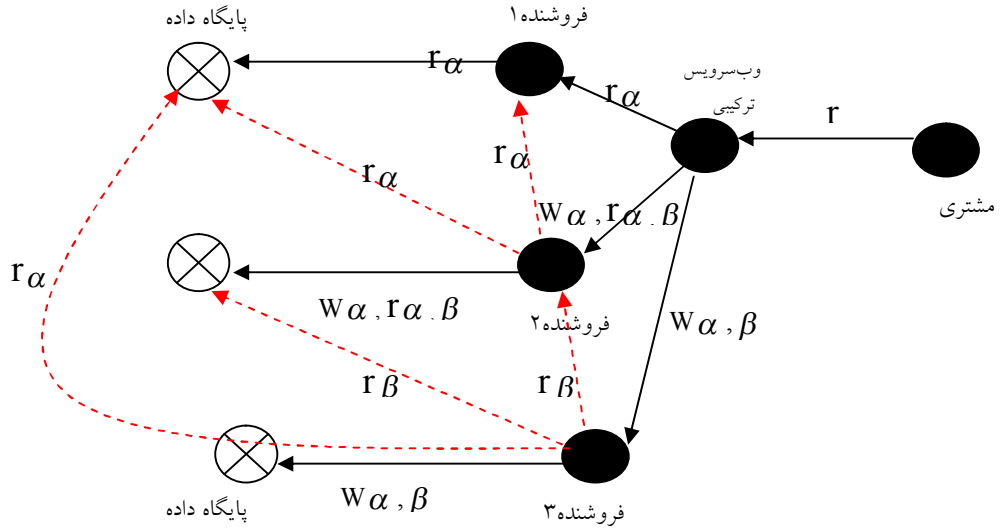
شکل۹. مدل *WSTG* فرآیند ترکیبی

حال با استفاده از قوانین تحلیلی بدست‌آمده در مطالعات مربوط به-مدل*TG* چگونگی انتقال اطلاعات را بین نهادهای سیستم بررسی می‌نماییم. یکی از گراف‌هایی که پس از اعمال قوانین مذکور بدست می‌آید در شکل۱۰ نشان داده شده است. این گراف پس از اعمال قوانین *pass*، *spy*، *pass* (به ترتیب) حاصل شده است.



شکل۱۰: گراف فرآیند پس از اعمال چند قانون

مشاهده می‌شود که بعد از اعلام قیمت کالا(α ریال) توسط فروشندهٔ۱ و عدم رضایت مشتری، درخواست کالا با شرط قیمت زیر α ریال، به فروشندهٔ۲ فرستاده می‌شود. توسط این روند نهایتاً فروشندهٔ۲ حق خواندن اطلاعات پایگاه دادهٔ فروشندهٔ را بدست می-



شکل ۱۱: گراف فرآیند برای سه فروشنده

آورد و مسند(G_0 , فروشگاه۱, فروشگاه۲)*can. know* دارای مقدار درستی می‌شود که این امر خلاف سیاست‌های امنیتی و کنترل دسترسی در وب‌سرویس فروشندهٔ۱ است (شکل۸). همین روند را برای سه وب-سرویس فروشنده و درحالی‌که فروشندهٔ۱ اطلاعات αو فروشندهٔ۲ اطلاعات βرا می‌دهد در شکل ۱۱ نشان داده‌ایم.

۶- نتیجه‌گیری و کارهای آتی

در این مقاله با استفاده از مدل حفاظتی *WSTG* برای مدل کردن فرآیند ترکیبی حاصل از ترکیب وب‌سرویس‌ها مبتنی بر*BPEL* ، روشی برای برخورد با مشکل فقدان زیربنای منطقی در این زبان ارائه نمودیم، به-علاوه توسط این مدل روشی برای تحلیل روند انتقال اطلاعات و مسئلهٔ کنترل دسترسی که در زمینهٔ ترکیب وب‌سرویس‌ها توجه زیادی به آن نشده بود بیان کردیم. هم اکنون، در حال بررسی گسترش مدل به *WSTG* زماندار هستیم تا بدین وسیله فعالیت‌های*sequence* و *flow* را که در فرآیندهای مبتنی بر *BPEL* برای بیان ترتیب و زمان اجرای فعالیت‌ها نسبت به یکدیگر به‌کار می‌رود را تحت پوشش قرار دهیم. از طرف دیگر قصد داریم با گسترش مدل به یک *WSTG* قیددار، امکان پارامتردهی به حقوق مختلف رئوس نسبت به یکدیگر را فراهم آوریم.لازم به ذکر است مدل پیشنهادی در این مقاله امکان سازگاری مناسبی برای وارسی کنترل دسترسی در دیگر زبان‌های ترکیب وب-سرویس‌ها را نیز دارد.

مراجع

[1] H. Kerger, Fulfilling the web services promise, communication of the ACM, June 2003/ Vol.46, No.6.
[2] F.Curbera, R.Khalaf, N.Mukhi, S.Tai, and S.Weerawarana, The next step in Web Services, communications of the ACM October 2003/Vol. 46, No.10.
[3] A.Ploskonos, An Introduction into Web Service Composition, Fundamentals of Service-Oriented Engineering, 2005.
[4] A.Charfi, M.Mezini, Middleware Services for Web Service Compositions, Chiba, Japan, ACM .WWW 2005, May ,2005.
[5] A. Jones, R. Lipton, and L. Snyder, A Linear Time Algorithm for Deciding Security, Proc 17 th Annual Symp. on the Foundations of Computer Science (Oct. 1976), 33-41.
[6] R.Hull, J. Su, Tools for Design of Composite Web Services, SIGMOD 2004, Paris, France. 2004 ACM. June13-18, 2004.
[7] A.Charfi, M.Mezini, Using Aspects for Security Engineering of Web Service Compositions, German Research Foundation (DFG) as part of the PhD Program "Enabling Technologies for Electronic Commerce".2005.
[8] M. Bishop, "Conspiracy and information flow in the Take-Grant Protection Model", Journal of Computer Security, vol 4(4), 1996, pp 331-360.
[9] M. Bishop, Theft of Information in the Take-Grant Protection Model, Journal of Computer Security (1994/1995).
[10] M. Bishop, The transfer of information and authority in a protection systems, 1979 ACM0-89791-009-5/79/1200/0045.
[11] S. Weerawarana, F. Curbera, Business Process with BPEL4WS: Understanding BPEL4WS.
http:// www-128.ibm.com/developer works/library/ws-bpelcol1/ 2002.
[12] R. Hull, J. Su, Tools for Composite Web Services: A Short Overview, SIGMOD Record, Vol. 34, No. 2, June 2005.
[13] N. Milanovic, M. Malek , current solutions for Web Service Composition, IEEE Internet Computing November 2004.
[14] N. Milanovic, M. Malek, Verifying correctness of Web Service Composition, Proceedings of the 11th Infofest, Budva, Montenegro 2004.
[15] M. Rouached, O. Perrin, C. Godart, Securing Web Service Compositions: Formalizing Authorization policies using Event Calculus, 2006.
[16] A. Charfi, M Mezini, Middleware Services for Web Service Compositions ,WWW 2005, Chiba, Japan.ACM, May, 2005.
[17] G. Diaz , Juan-Jos´e Pardo, Mar´ıa-Emilia Cambronero , Verification of Web Services with Timed Automata , Electronic Notes in Theoretical Computer Science 157 (2006) 19–34.
[18] Mariya Koshkina , Franck van Breugel, Verification of Business Processes for Web Services, York University, Department of Computer Science, October 2003.
[19] L.G. Meredith, S. Bjorg. Contracts and Types. Communications of the ACM, 46, No. 10, pp 41-47, October 2003.
[20] H. Schlingloff, A. Martens, K. Schmidt, Modeling and Model Checking Web Services, Electronic Notes in Theoretical Computer Science 126 (2005) 3–26.