



شرکت آوای فناوری الکترونیک سلامت

پروژه‌ی طراحی، پیاده‌سازی و اجرای زیرساخت تبادل اطلاعات سلامت

عنوان سند: توصیف فنی زیرساخت HIX
شناسه‌ی سند: HIX.TEC.SPC.V.0.3.4
نگارش: V.0.3.4
تاریخ تهیه: ۹۴/۰۵/۱۹

شرح سند:

این سند شامل توصیف زیرساخت HIX به منظور تسهیل انجام تعاملات الکترونیکی بین سامانه‌های نرم‌افزاری در وزارت بهداشت است.

این سند به منظور توصیف فنی زیرساخت HIX جهت ارائه به تولیدکنندگان نرم افزار و به سفارش وزارت بهداشت، درمان و آموزش پزشکی تهیه شده است و هرگونه تکثیر از آن در خارج از محدوده‌ی مورد توافق ممنوع می‌باشد.

تاریخچه

تاریخ	نسخه	شرح	تهیه کننده	تایید کننده	تصویب کننده
۹۴/۰۴/۱۰	۰.۳.۳	ارائه‌ی نسخه‌ی اولیه	تیم فنی	شهرام کیائی	دکتر میرعلی سیدی
۹۴/۰۵/۱۹	۰.۳.۴	اضافه شدن امکانات و سرویس‌های جدید به زیرساخت	تیم فنی	شهرام کیائی	دکتر میرعلی سیدی

فهرست مطالب

۵	۱- مقدمه
۶	۲- انتقال اطلاعات در زیرساخت HIX
۶	۲-۱- مدل مفهومی
۸	۲-۲- توصیف لایه انتقال
۸	۱-۲-۲- متعاملین
۹	۲-۲-۲- نقاط انتهایی ارتباط
۱۲	۳-۲-۲- زیرساخت
۱۴	۳- تحویل مطمئن پیام و امنیت در زیرساخت HIX
۱۴	۱-۳- تحویل مطمئن
۱۸	۲-۳- امنیت
۱۸	۱-۲-۳- امنیت دستیابی به کانال دسترسی
۱۹	۲-۲-۳- امنیت دستیابی به سرویس‌های مقصد
۲۰	۴- فراخوانی کانال دسترسی
۲۰	۱-۴- فراخوانی کانال دسترسی به صورت همگام
۲۰	۱-۱-۴- ارسال پیام به وب سرویس کانال دسترسی
۲۲	۲-۱-۴- بازگشت پاسخ از وب سرویس کانال دسترسی
۲۲	۲-۴- فراخوانی کانال دسترسی بصورت ناهمگام
۲۳	۱-۲-۴- ارسال پیام به یک مقصد
۲۳	۲-۲-۴- ارسال پیام به وب سرویس کانال دسترسی
۲۵	۱-۲-۴- بازگشت پاسخ از وب سرویس کانال دسترسی
۲۵	۳-۲-۴- ارسال پیام به چند مقصد
۲۵	۱-۳-۲-۴- ارسال پیام به وب سرویس کانال دسترسی
۲۷	۲-۳-۲-۴- بازگشت پاسخ از وب سرویس کانال دسترسی
۲۸	۴-۲-۴- تاییدیه رسید پیام به مقصد در ارسال ناهمگام
۲۹	۵-۲-۴- خطاهای زمانی در ارسال ناهمگام

- ۳۰-۴-۲- مدیریت خطا به صورت عمومی ۳۰
- ۳۰-۴-۳-۱- بروز خطا در فراخوانی کانال دسترسی ۳۰
- ۳۲-4-4- امنیت در فراخوانی سرویس ۳۲
- ۳۳-۴-۴-۱- ثبت نام هویت سازمان و نرم افزار مبداء فراخوانی کننده ۳۳
- ۳۴-۴-۴-۲- صدور شناسه و سرفصل امنیتی برای متعاملین سوئیچ ۳۴
- ۳۵-۴-۵- نمونه کد فراخوانی کانال دسترسی زیرساخت HIX به صورت همگام ۳۵
- ۳۵-۴-۵-۱- نمونه کد در NET ۳۵
- ۳۶-۴-۵-۲- نمونه کد در JAVA ۳۶
- ۳۷-۴-۶- سرویس های زیر ساخت ۳۷
- ۳۷-۴-۶-۱- دریافت لیست تفاهم نامه های مربوط به سازمان ۳۷
- ۴۰-۴-۶-۲- آگاهی از وضعیت سرویس در سازمان مقصد ۴۰

۱- مقدمه

در سالیان اخیر روند الکترونیکی شدن تبادل اطلاعات در سازمان‌های دولتی و غیردولتی شتاب بیشتری به خود گرفته است. از این رو وزارت بهداشت، درمان و آموزش پزشکی به‌عنوان متولی سلامت، دارو و درمان اقدام به ایجاد شبکه‌ی نرم‌افزاری نموده و از این طریق قصد دارد سامانه‌های فعال در حوزه‌ی سلامت را با یکدیگر یکپارچه نماید. در این راستا شرکت آوای فناوری اطلاعات سلامت به‌عنوان اپراتور زیرساخت نرم‌افزاری یاد شده در وزارت بهداشت کار خود را در سال ۱۳۹۴ شروع نموده است. با توجه به مدل توزیع شدگی، مدیریتی و نیاز به محرمانه‌ی اطلاعات تبادل موجود در وزارت بهداشت این شرکت طرح توزیع‌شدگی را به‌طور کامل جهت تفکیک ترافیک‌های محلی از ملی و ایجاد فضای امن تبادل اطلاعات داخلی پشتیبانی می‌نماید.

بر این اساس تبادلات اطلاعات بین سامانه‌های نرم‌افزاری در صورتی‌که از مرزهای مدیریتی و سازمانی فراتر رود از طریق زیرساخت HIX انجام می‌شوند. این بدان معناست که سامانه‌های داخلی یک بیمارستان به‌منظور تبادل نیازی به استفاده از HIX ندارند در صورتیکه ارتباط بین دو بیمارستان از طریق HIX صورت می‌پذیرد. بدین ترتیب با وجود زیرساخت ارتباطی و همچنین شرایط نرم‌افزاری موجود این امکان فراهم شده است تا سازمان‌های درگیر در حوزه‌ی سلامت از این زیرساخت جهت رفع نیازمندی‌های اطلاعاتی خود بهره‌لازم را ببرند.

۲- انتقال اطلاعات در زیرساخت HIX

۲-۱- مدل مفهومی

زیرساخت تبادل اطلاعات سلامت (HIX) یک زیرساخت نرم‌افزاری است و به صورت یک میان‌افزا ارتباطی جهت برقراری ارتباط مکانیزه‌ی بین دو سامانه‌ی نرم‌افزاری طراحی و تولید شده است. الگوی تبادل پیام در زیرساخت HIX می‌تواند به صورت‌های ذیل باشد:

- از دیدگاه جهت تبادل:

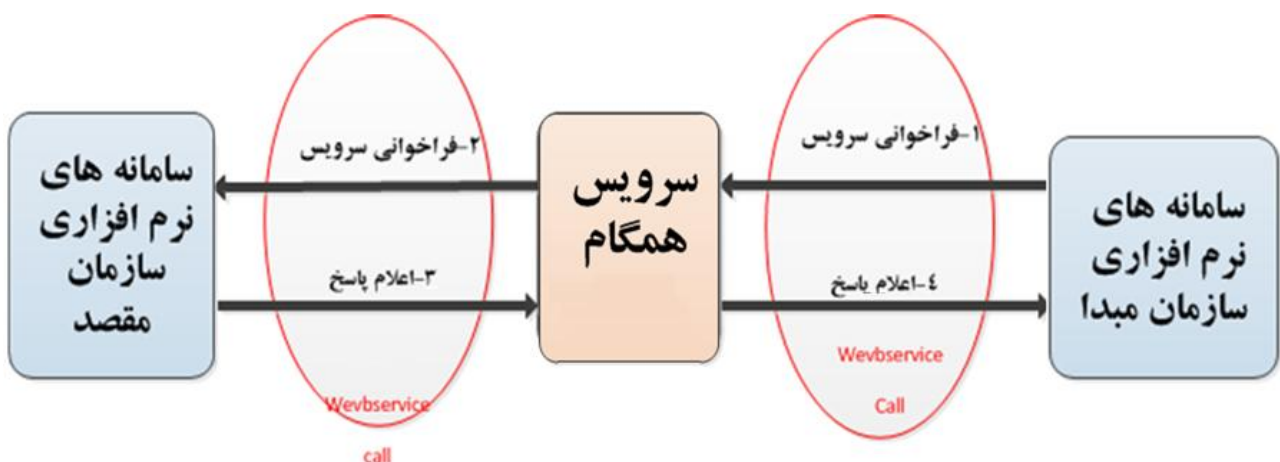
- یک‌طرفه: به این معنا که اطلاعات از یک طرف به طرف دیگر انتقال یافته و سازمان اول منتظر پاسخ از طرف سازمان دوم نیست.

- دوطرفه: به این معنا که اطلاعات به صورت دوطرفه انتقال یافته و طرف اول منتظر پاسخ از سمت طرف دوم است.

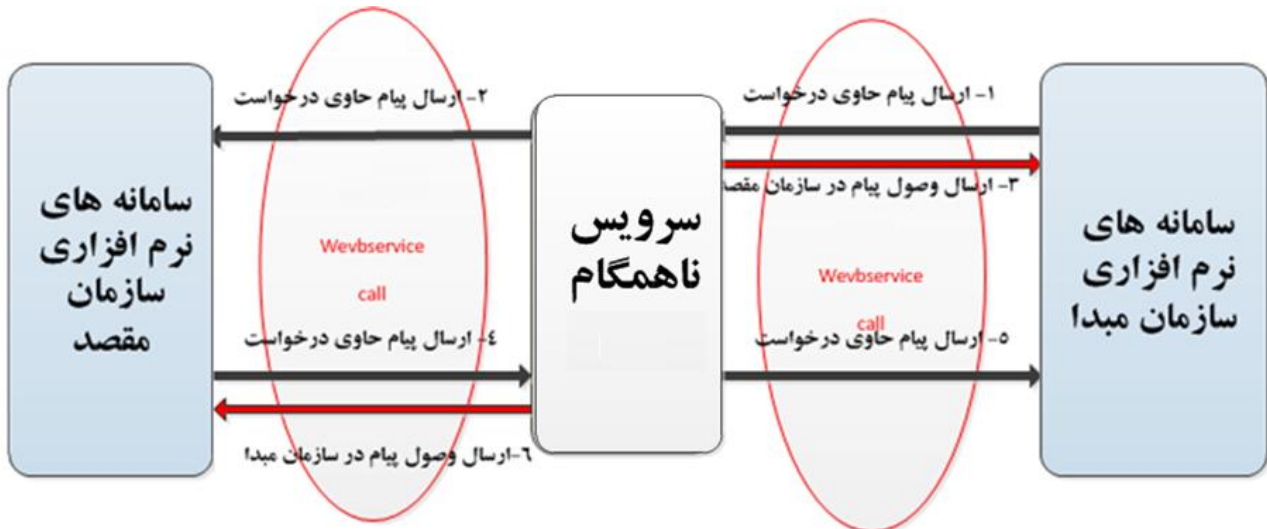
- از دیدگاه رعایت همزمانی تبادل:

- همگام: این ارتباط مانند استفاده از یک وب سرویس معمولی است. سامانه‌ی نرم‌افزاری مبدأ در صورت فراخوانی سرویس تا زمان دریافت پاسخ مسدود می‌ماند. ارتباط همگام در اصل یک ارتباط دوطرفه است به این معنا که فراخوانی سرویس از سرویس‌دهنده نهایی حتماً با بازگشت یک مقدار همراه است. لیکن این مقدار می‌تواند به عنوان پاسخ به درخواست و یا تایید دریافت درخواست تفسیر گردد که در مورد اول تبادل دوطرفه و در مورد دوم تبادل یک طرفه فرض می‌گردد.

- ناهمگام: در این حالت سامانه‌ی ارسال کننده‌ی اطلاعات پس از ارسال درخواست به فعالیت خود ادامه داده و بلوکه نمی‌گردد. در صورتی که ارتباط دو طرفه باشد پاسخ به پیام درخواست به زمان دیگری موکول خواهد شد و در لحظه ارسال درخواست تولید و ارائه نمی‌شود.



شکل شماره ۱: تبادل همگام



شکل شماره ۲: تبادل دو طرفه ناهمگام

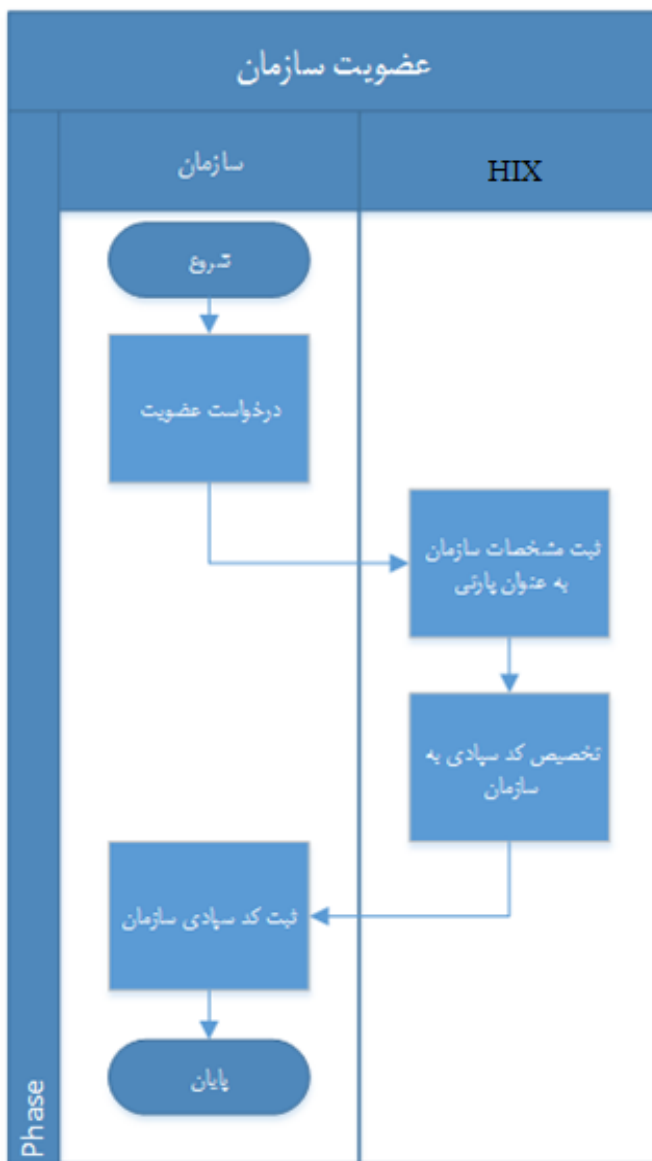
به‌عنوان مثال بیماری را در نظر بگیرید که نسخه‌ای را برای دریافت نزد داروخانه می‌برد، نسخه‌ی فوق توسط داروخانه پیچیده شده و برای سازمان بیمه از طریق زیرساخت HIX ارسال شود گرفتن تأییدیه که نسخه‌ی فوق به دست سازمان بیمه رسیده است، یک ارتباط یک‌طرفه است زیرا هیچ‌گونه پردازشی بر روی نسخه هنوز صورت نگرفته است و از طرفی چنین ارتباطی همگام خوانده می‌شود چون بلافاصله پس از ارسال داروخانه منتظر دریافت تأییدیه‌ی دریافت توسط بیمه است. حال اگر سازمان بیمه تمامی نسخ موجود در یک ماه را جمع‌آوری و سپس تأییدیه‌ای برای داروخانه مبتنی بر نسخ ارسالی از سمت وی ارسال نماید ارتباطی همچنان یک‌طرفه و ناهمگام برقرار نموده است؛ زیرا در این صورت داروخانه به ازای هر نسخه منتظر دریافت تأییدیه نیست و به فعالیت خود ادامه می‌دهد. حال شرایطی را در نظر بگیرید که برای پیچیدن نسخه، داروخانه نیازمند گرفتن تأیید اینترنتی از بیمه جهت تحویل دارو به بیمار است، در چنین شرایطی بایستی داروخانه تا زمان دریافت تأییدیه منتظر بماند و در این حالت یک ارتباط همگام (داروخانه سیستم را معطل دریافت تأییدیه‌ی اینترنتی نموده و کارها را تا زمان دریافت نتیجه‌ی تأییدیه به تعویق می‌اندازد) و همچنین ارتباطی دوطرفه خواهیم داشت زیرا از یک طرف تقاضا برای دریافت تأییدیه توسط داروخانه ارسال می‌گردد و از طرف دیگر بیمه نتیجه‌ی تأییدیه را با انجام پردازشی که بر روی داده‌های ارسالی توسط داروخانه صورت گرفته، اعلام می‌نماید. چنانچه برای تشکیل پرونده برای بیماران خاص داروخانه ملزم به انجام کار باشد، بایستی داده‌ها برای بیمه ارسال گردند و بیمه پس از بررسی‌های لازم بر روی داده‌های ارسالی نتیجه را برای داروخانه مشخص نماید (ارتباط دوطرفه) اما از آنجایی که ارائه‌ی پاسخ چند روز بعد می‌تواند اعلام شود و داروخانه در آن لحظه منتظر دریافت پاسخ نیست ارتباطی ناهمگام خواهیم داشت.

۲-۲- توصیف لایه انتقال

لایه‌ی انتقال در زیرساخت HIX، انتقال امن و مطمئن اطلاعات را از سامانه‌ی نرم‌افزاری مبدا به سامانه‌ی نرم‌افزاری مقصد برعهده دارد. این لایه دارای مولفه‌های ذیل است که به صورت جداگانه تشریح می‌گردند:

۲-۲-۱- متعاملین

منظور از متعامل، سازمان‌های مبدا و مقصد در تعامل هستند. این سازمان‌ها از طریق سامانه‌های نرم‌افزاری خود اقدام به برقراری ارتباط و تبادل اطلاعات می‌نمایند. تعامل بین سازمان‌ها در قالب ارائه و دریافت سرویس محقق می‌گردد. به منظور انجام تعامل، سازمان متعامل می‌بایست به زیرساخت معرفی گردد؛ به این منظور مشخصات سازمان متعامل در دفتر



ثبت زیرساخت، ثبت گردیده و یک کد کاربری به سازمان تخصیص می‌یابد. براین اساس سازمان می‌تواند تعاملات خود را رصد نموده و مدیریت نماید. با این کد کاربری سازمان می‌تواند سرویس‌های مورد تقاضای خود را از سازمان‌های ارائه‌کننده‌ی سرویس درخواست نموده و همچنین درخواست‌های تقاضای سرویس وارده به خود جهت دریافت سرویس را مدیریت نماید (تایید یا رد نموده و یا بدون پاسخ رها سازد).

سازمان می‌تواند پارامترهای کیفی ارائه‌ی سرویس را به ازای سرویس‌گیرندگان متفاوت، به صورت مجزا تعریف نموده و به این ترتیب مدیریت بهتری بر روی سرویس‌های قابل ارائه خود داشته باشد. به طور مثال سازمان ثبت احوال می‌تواند در زیرساخت، زمان پاسخ‌گویی به درخواست وزارت بهداشت، درمان و آموزش پزشکی را ۲۰ ثانیه و زمان پاسخ‌گویی به درخواست سازمان غذا - دارو را ۴۰ ثانیه تعریف نماید.

علاوه بر این سازمان می‌تواند دسترسی سازمان‌های استفاده‌کننده از سرویس‌های خود را به صورت موقت غیر فعال نموده و یا به صورت کلی ارائه‌ی سرویس را از طریق زیرساخت به شکل یک طرفه قطع نماید.

فرآیند ثبت مشخصات سازمان متعامل در زیرساخت مطابق نمودار روبرو انجام می‌پذیرد.

شکل شماره ۳: فرآیند شناسایی متعاملین

۲-۲-۲- نقاط انتهایی ارتباط

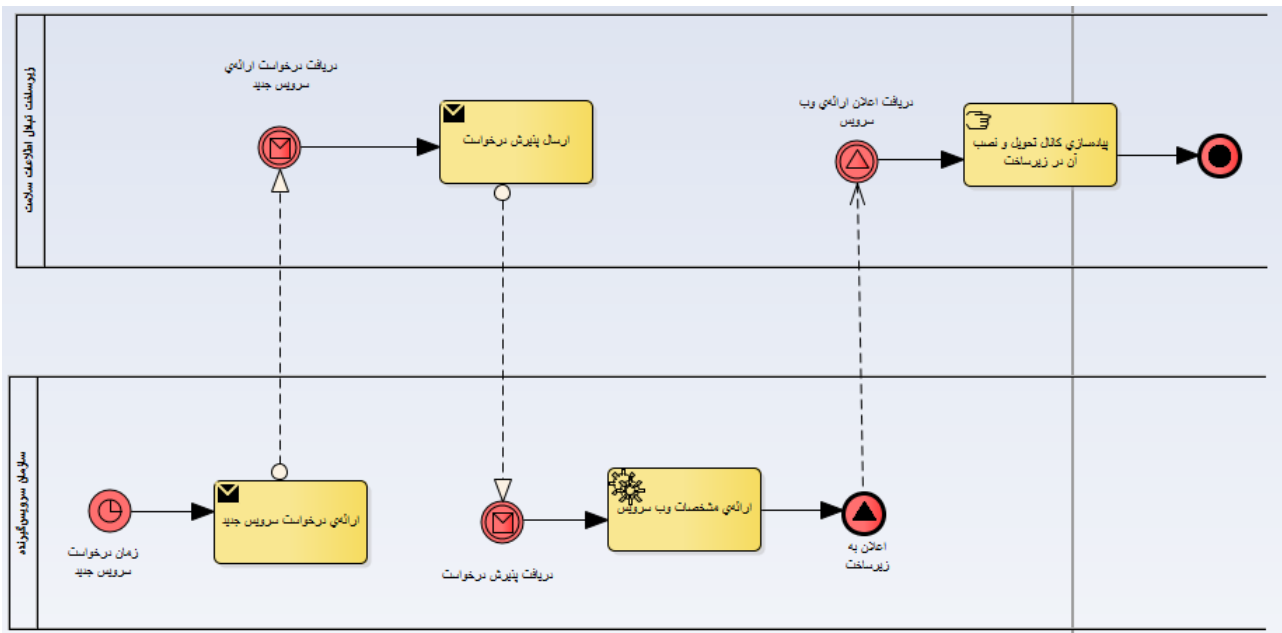
از آنجایی که ارتباط از طریق زیرساخت HIX به صورت ارتباط بین سامانه‌های نرم‌افزاری و به صورت مکانیزه انجام می‌شود، سامانه‌های نرم‌افزاری در سازمان‌های متعامل به عنوان نقاط انتهایی ارتباط شناخته می‌شوند. نقش سامانه‌های نرم‌افزاری براساس نقش سازمان متعامل مربوطه دارای حالت‌های ذیل است:

- سرویس‌دهنده: سامانه، سرویس‌های خود را در اختیار زیرساخت قرار داده و زیرساخت سرویس‌های مذکور را فراخوانی می‌نماید. براین اساس سامانه‌ی سرویس‌دهنده می‌تواند شرایط اتصال زیرساخت به سرویس‌های خود را به زیرساخت دیکته نماید که این شرایط می‌تواند براساس پروتکل‌های ارتباطی و سیاست‌های امنیتی متفاوت باشد. به‌طور مثال جهت اتصال به سامانه‌ی گرفتن تأییدیه اینترنتی در بیمه می‌بایست موارد ذیل رعایت گردد:
 - پیاده‌سازی ماجول سرویس‌گیرنده براساس WSDL سرویس. سرویس براساس پروتکل SOAP و به صورت RPC طراحی گردیده است (سرویس به صورت دوطرفه و همگام ارائه می‌گردد).
 - استفاده از Token سخت‌افزاری به منظور احراز هویت.
 - فراخوانی سرویس براساس URL سرویس
- سرویس‌گیرنده: سامانه، سرویس زیرساخت را فراخوانی می‌نماید. براین اساس زیرساخت، شرایط اتصال به سرویس‌های خود را به سرویس‌گیرنده دیکته می‌نماید. اتصال به زیرساخت تنها از طریق سرویس کانال دسترسی ممکن است. زیرساخت کلیه‌ی خدمات خود را از یک درگاه مشترک ارائه می‌نماید؛ به‌طور مثال به منظور فراخوانی هر سرویس در زیرساخت می‌بایست موارد ذیل رعایت گردد:
 - فراخوانی سرویس کانال دسترسی زیرساخت به صورت REST برای کلیه‌ی سرویس‌ها
 - ارائه‌ی نام سرویس و عملیات به منظور فراخوانی سرویس نهایی توسط زیرساخت
 - ارائه‌ی پارامترهای قابل ارائه به سرویس نهایی به صورت JSON
 - ارائه‌ی شناسه‌ی سخت‌افزاری مربوط به سرور سامانه‌ی نرم‌افزاری مبداء جهت احراز هویت

سازمان قبل از شروع تعامل می‌بایست سامانه(های) نرم‌افزاری خود را به زیرساخت معرفی نماید. معرفی سامانه براساس اینکه می‌خواهد نقش سرویس‌گیرنده و یا سرویس‌دهنده را ایفا نماید، متفاوت است. همچنین تعامل به صورت همگام و ناهمگام نقش تعیین‌کننده‌ای در شیوه‌ی تعریف سامانه در زیرساخت دارد:

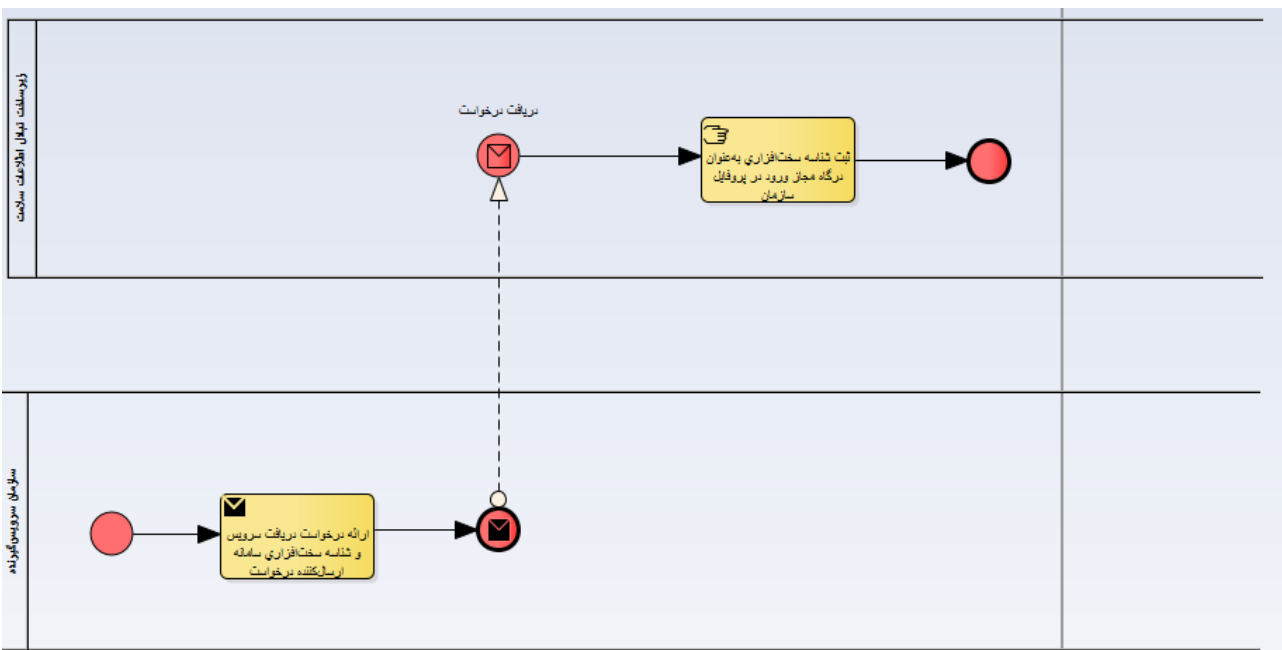
• همگام :

- سرویس‌دهنده: سامانه‌ی نرم‌افزاری می‌بایست سرویس خود را به صورت وب سرویس (SOAP یا REST) ارائه دهد. همچنین سیاست‌های امنیتی اتصال می‌بایست معرفی گردند. در این حالت زیرساخت با تولید یک آداپتور نرم‌افزاری به سرویس متصل می‌شود که به این آداپتور اصطلاحاً "کانال تحویل" اطلاق می‌شود. (فلش با برچسب ۲ یا ۴ از شکل شماره ۱)



شکل شماره ۴: ارائه سرویس به صورت همگام

○ سرویس‌گیرنده: سامانه‌ی نرم‌افزاری می‌بایست توانایی فراخوانی کانال دسترسی زیرساخت که به صورت وب‌سرویس REST آماده‌سازی شده است را داشته باشد. به منظور رعایت سیاست‌های امنیتی سویچ زیرساخت HIX، شناسه‌ی سخت‌افزاری مربوط به سروری را که سامانه‌ی نرم‌افزاری بر روی آن نصب است، می‌بایست به عنوان درگاه ورودی سازمان ارائه شده و در پروفایل مربوط به سازمان در زیرساخت ثبت گردیده باشد. (فلش با برچسب ۱ یا ۳ از شکل شماره ۱)

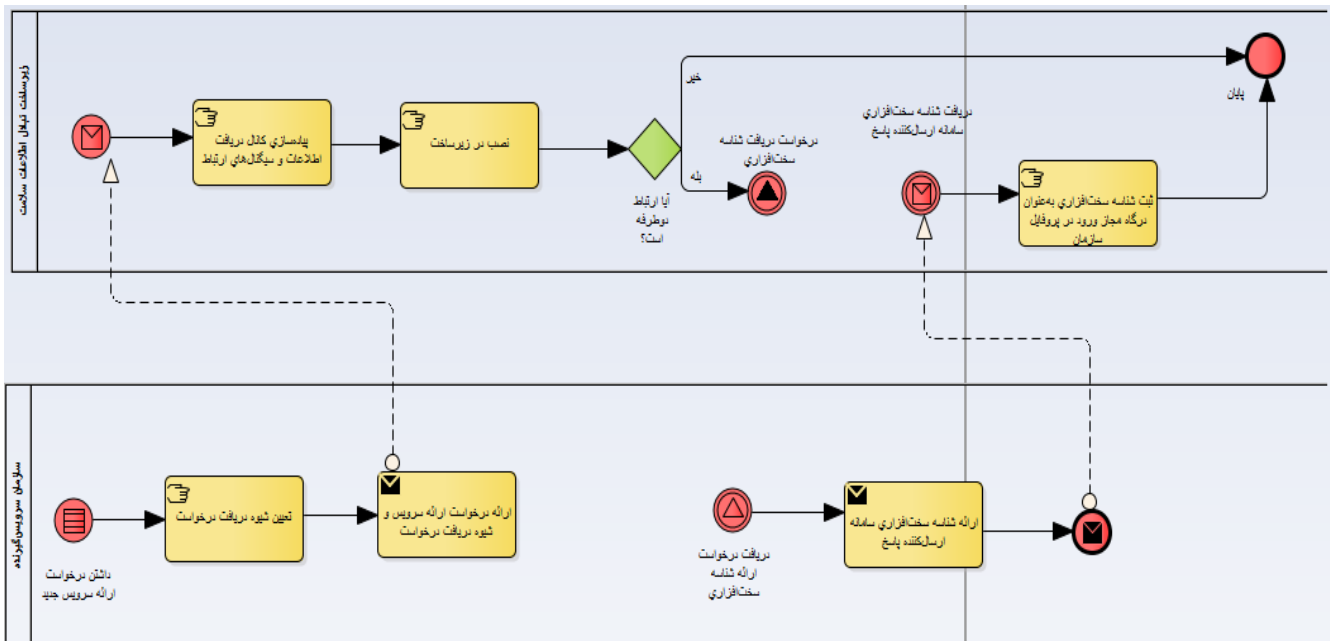


شکل شماره ۵: دریافت سرویس به صورت همگام

• ناهمگام :

- سرویس دهنده: از آنجایی که در سرویس ناهمگام نیازی به تولید و ارائه‌ی پاسخ در لحظه‌ی فراخوانی وجود ندارد، سامانه‌ی نرم‌افزاری سرویس دهنده می‌تواند درخواست را از مسیرهای مختلف دریافت نماید:
 - دریافت از طریق وب سرویس: در این حالت سامانه‌ی نرم‌افزاری می‌بایست سرویس خود را به صورت وب سرویس (SOAP یا REST) ارائه دهد.
 - دریافت از طریق ایمیل: در این حالت سامانه‌ی نرم‌افزاری می‌بایست یک آدرس ایمیل برای دریافت درخواست‌ها به زیرساخت معرفی نماید.
 - دریافت فایل: در این حالت سرویس دهنده می‌بایست یک مسیر فایل را معرفی نموده تا درخواست به صورت فایل توسط زیرساخت در آن محل قراردادده شود.

در کلیه‌ی روش‌های فوق سیاست‌های امنیتی اتصال می‌بایست معرفی گردند. نهایتاً زیرساخت با تولید یک آداپتور نرم‌افزاری امکان اتصال سرویس دهنده به زیرساخت را فراهم می‌آورد، به این آداپتور نیز اصطلاحاً "کانال تحویل" اطلاق می‌گردد. ذکر این نکته ضروری است که شیوه‌ی دریافت درخواست از دید سامانه‌ی مبدا پنهان است (فلش با برچسب ۲ یا ۵ از شکل شماره ۲)

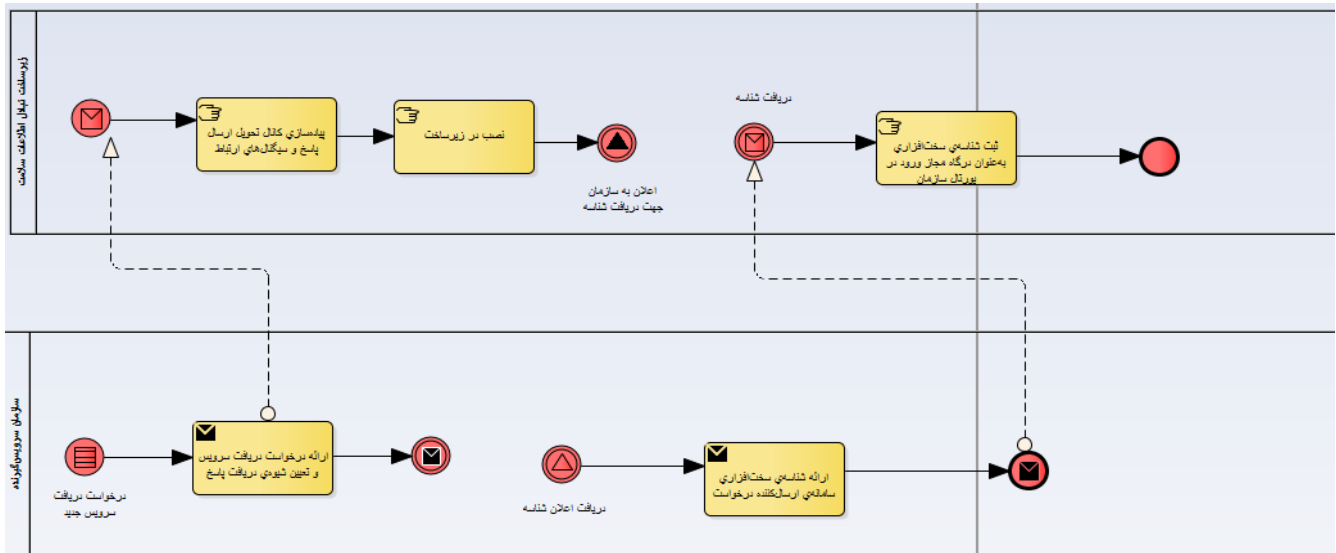


شکل شماره ۶: ارائه سرویس به صورت ناهمگام

- سرویس گیرنده: سامانه‌ی سرویس گیرنده در مدل ناهمگام می‌بایست دارای توانایی‌های ذیل باشد:
 - توانایی فراخوانی سرویس کانال دسترسی زیرساخت که به صورت وب سرویس REST آماده‌سازی شده است: به منظور رعایت سیاست‌های امنیتی سوییچ زیرساخت HIX، شناسه‌ی سخت‌افزاری مربوط به

سروری که سامانه‌ی نرم‌افزاری بر روی آن نصب است می‌بایست به عنوان درگاه ورودی سازمان ارائه شده و در پروفایل مربوط به سازمان در زیرساخت ثبت گردیده باشد (فلش با برچسب ۱ یا ۴ در شکل شماره ۲).

▪ سرویسی ارائه نماید تا زیرساخت به منظور اعلام وصول پیام در مقصد آن را فراخوانی کند (فلش با برچسب ۳ یا ۶ در شکل شماره ۲).



شکل شماره ۷: دریافت سرویس به صورت ناهمگام

۳-۲-۲- زیرساخت

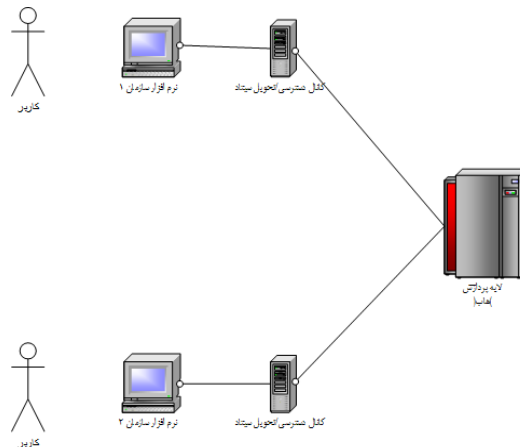
زیرساخت شامل کلیه‌ی مولفه‌هایی است که تبادل را بین دو سامانه ممکن می‌سازد. زیرساخت دارای دو مولفه‌ی مهم است: اولی مولفه‌ای است که سامانه‌ی نرم‌افزاری سرویس‌گیرنده به‌واسطه‌ی آن، آنرا فراخوانی می‌نماید که به این مولفه اصطلاحاً "کانال دسترسی" اطلاق می‌شود و دومی مولفه‌ای است که به‌واسطه‌ی آن سرویس سامانه‌ی نرم‌افزاری سرویس‌دهنده توسط زیرساخت فراخوان می‌شود که این مولفه اصطلاحاً "کانال تحویل" نامیده می‌شود.

- کانال دسترسی: بخشی از زیرساخت HIX است که به سامانه نرم‌افزاری مبداء این امکان را می‌دهد تا اطلاعات خود را به‌منظور انتقال به مقصد تحویل زیرساخت HIX دهد. انتقال اطلاعات به کانال دسترسی برای انتقال همگام و یا ناهمگام به یک شکل و توسط فراخوانی وب سرویس زیرساخت (REST) انجام می‌شود.
- کانال تحویل: بخشی از زیرساخت HIX است که به سامانه‌ی نرم‌افزاری مقصد این امکان را می‌دهد تا مسیر دریافت اطلاعات از زیرساخت را تعیین نماید. مشخصات این کانال براساس تبادل همگام و یا ناهمگام بین مبداء و مقصد می‌تواند متفاوت باشد:
 - ارسال همگام: در صورتی که سامانه‌ی مبداء، سرویس سامانه‌ی مقصد را به صورت همگام نیاز داشته باشد، سامانه‌ی مقصد می‌بایست یک سرویس وب (SOAP یا REST) پیاده‌سازی نموده و به زیرساخت

معرفی نماید. در این صورت متناسب با سرویس ارائه شده، یک کانال دسترسی منحصر به سرویس مذکور در زیرساخت تعریف می‌گردد.

○ ارسال ناهمگام: در صورتی که سامانه‌ی مبدأ، سرویس سامانه‌ی مقصد را به صورت ناهمگام نیاز داشته باشد، سامانه‌ی مقصد می‌تواند کانال‌های دریافت مختلفی را به زیرساخت پیشنهاد دهد. در این صورت متناسب با کانال ارائه شده یک کانال دسترسی منحصر به آن در زیرساخت تعریف می‌گردد. لازم به ذکر است که کانال تحویل در سمت سامانه‌ی مقصد از دید سامانه‌ی مبدأ مخفی است و ارتباط سامانه‌ی مبدأ تنها با زیرساخت و از طریق کانال دسترسی است. کانال ارائه شده می‌تواند به یکی از صورت‌های ذیل باشد:

- Email
- File
- (SOAP/REST) WebService



شکل شماره ۸: کانال‌های دسترسی و تحویل زیرساخت HIX

۳- تحویل مطمئن پیام و امنیت در زیرساخت HIX

۳-۱- تحویل مطمئن

تحویل مطمئن پیام به این معناست که سامانه‌ی نرم‌افزاری مبداء می‌بایست این اطمینان را حاصل کند که پیام ارسالی وی به سامانه‌ی نرم‌افزاری مقصد رسیده باشد و پیام پاسخ در مدت زمان مشخصی به وی بازگردانده شود.

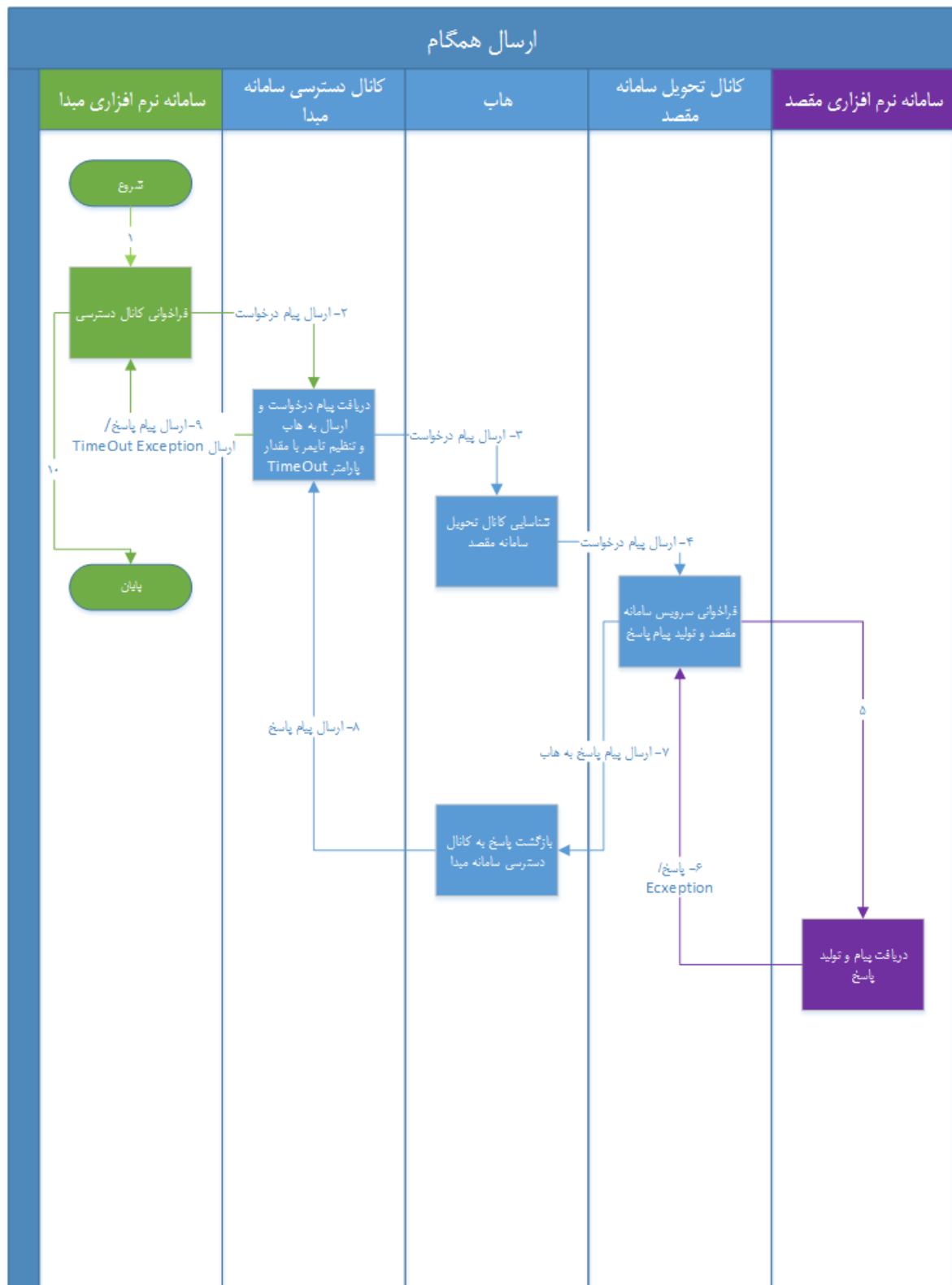
ارتباط همگام

ارتباط همگام در زیرساخت به معنای فراخوانی سرویس وب همگام از سازمان سرویس‌دهنده توسط زیرساخت است. این فراخوانی باعث می‌شود تا زیرساخت تا زمان دریافت پاسخ از سرویس‌دهنده منتظر بماند. از این رو در صورتی که ارتباط بین دو سامانه‌ی نرم‌افزاری از نوع همگام باشد، سامانه‌ی نرم‌افزاری مبداء با فراخوانی کانال دسترسی زیرساخت باید آنقدر منتظر بماند تا پاسخ مناسب از طرف مقابل دریافت گردد. بنابراین در ارتباط همگام خاصیت تحویل مطمئن به صورت تلویحی وجود دارد. با وجود این از آنجایی که زمان دریافت پاسخ در این نوع سرویس توسط سامانه‌ی نرم‌افزاری مبداء می‌تواند تعیین کننده باشد، سامانه‌ی نرم‌افزاری مبداء می‌تواند توسط پارامتر Timeout مقدار انتظار خود را برای دریافت پاسخ به زیرساخت اعلام نماید. براین اساس هنگامی که کانال دسترسی زیرساخت توسط سامانه‌ی نرم‌افزاری مبداء فراخوانی شد، زیرساخت با انتقال پیام از طریق هاب داخلی به کانال تحویل مربوط به سامانه‌ی نرم‌افزاری مقصد، سرویس سامانه‌ی نرم‌افزاری مقصد را فراخوانی می‌کند. همزمان با این عمل، زیرساخت یک تایمر با مقدار ارائه شده توسط سامانه‌ی نرم‌افزاری مبداء (تعیین شده به صورت پارامتر در فراخوانی کانال دسترسی) مقداردهی می‌نماید. هنگامی که دریافت پاسخ از سامانه‌ی نرم‌افزاری مقصد بیش از زمان انتظار تعیین شده طول بکشد، زیرساخت، فراخوانی را ملغی نموده و با ایجاد یک خطای زمان (Timeout Exception) و ارائه‌ی آن به سامانه‌ی نرم‌افزاری مبداء، این سامانه را از حالت انتظار خارج می‌نماید. شکل شماره ۹ فرایند کلی ارتباط همگام از طریق زیرساخت را نمایش می‌دهد.

براساس آنچه که در شکل توصیف شده، تنها کانال‌های دسترسی و تحویل زیرساخت هستند که در فراخوانی همگام بلوکه می‌شوند و ارتباط بین کانال‌های فوق و هاب زیرساخت مبتنی بر پیام‌دهی Async بوده و مسیر رفت و برگشت پیام و پاسخ یکی نیست.

در اصل کانال دسترسی با منتظر نگاه‌داشتن فراخوانی کننده‌ی سرویس ارتباط همگام را شبیه‌سازی نموده و ارتباط مستقیم و همگامی بین کانال دسترسی و کانال تحویل در زیرساخت وجود ندارد.

براین اساس کانال دسترسی با تنظیم یک تایمر می‌تواند قبل از بازگشت پاسخ پیام از کانال تحویل فراخوانی کننده، کانال دسترسی را رها سازد. در این حالت یک پیام مناسب از نوع Exception به فراخوانی کننده‌ی کانال دسترسی بازگشت داده می‌شود.



شکل ۹: ارتباط همگام از طریق زیرساخت

ارتباط ناهمگام

در صورتی که ارتباط بین دو سامانه‌ی نرم‌افزاری از نوع ناهمگام باشد سامانه‌ی نرم‌افزاری مبداء با فراخوانی کانال دسترسی زیرساخت، یک شماره پیام (به منظور رهگیری‌های آتی) دریافت نموده و از انتظار خارج می‌گردد. لیکن آگاهی از رسید و یا عدم رسید پیام به سامانه‌ی مقصد برای وی مهم است. همچنین حداکثر زمان انتظار برای رسیدن پیام ارسالی به سامانه‌ی مقصد و نیز زمان انتظار سامانه‌ی مبداء برای دریافت پاسخ از سامانه‌ی مقصد می‌تواند دو پارامتر تعیین‌کننده باشد. به منظور پاسخ‌دهی به نیازهای مطرح شده، زیرساخت دارای مفاهیم ذیل می‌باشد:

- پیام تأییدیه رسید (Acknowledge Receipt): این پیام توسط زیرساخت تولید می‌شود و هنگامی که زیرساخت پیام را توسط کانال تحویل به سامانه‌ی مقصد تحویل می‌دهد یک پیام تأییدیه‌ی رسید تولید نموده و بازگشت به پیام ارسالی را به سامانه‌ی نرم‌افزاری مبداء ارسال می‌کند. (سامانه مبداء می‌بایست دارای کانال تحویل برای دریافت پیام مذکور باشد).

در صورتی که به هر دلیل کانال تحویل، در تحویل پیام به سامانه‌ی مقصد دچار مشکل شد یک پیام تأییدیه رسید با محتوی منفی برای سامانه‌ی نرم‌افزاری مبداء ارسال می‌گردد.

- زمان انتظار برای تأییدیه رسید (TimeToAck): پارامتری است که در تفاهم نامه‌ی دریافت سرویس توسط سازمان ارائه‌کننده‌ی سرویس تنظیم شده و نشان‌دهنده‌ی سطح خدمتی است که توسط ارائه‌دهنده‌ی سرویس قابل انجام است.

بر این اساس هنگامی که کانال دسترسی زیرساخت توسط سامانه‌ی نرم‌افزاری مبداء فراخوانی شد، زیرساخت با انتقال پیام از طریق هاب داخلی به کانال تحویل مربوط به سامانه‌ی نرم‌افزاری مقصد، سرویس سامانه‌ی نرم‌افزاری مقصد را فراخوانی می‌کند. همزمان با این عمل زیرساخت یک تایمر (با مقدار تعیین شده در تفاهم نامه بین دو سازمان) مقداردهی می‌نماید. هنگامی که تحویل پیام به سامانه‌ی نرم‌افزاری مقصد بیش از زمان انتظار تعیین شده، طول بکشد زیرساخت فراخوانی را ملغی نموده و با ایجاد یک خطای زمان (TimeOut Exception) و ارائه‌ی آن به سامانه‌ی نرم‌افزاری مبداء، سامانه‌ی نرم‌افزاری مبداء را از عدم رسید پیام به سامانه‌ی مقصد در مدت زمان تعیین شده مطلع می‌سازد.

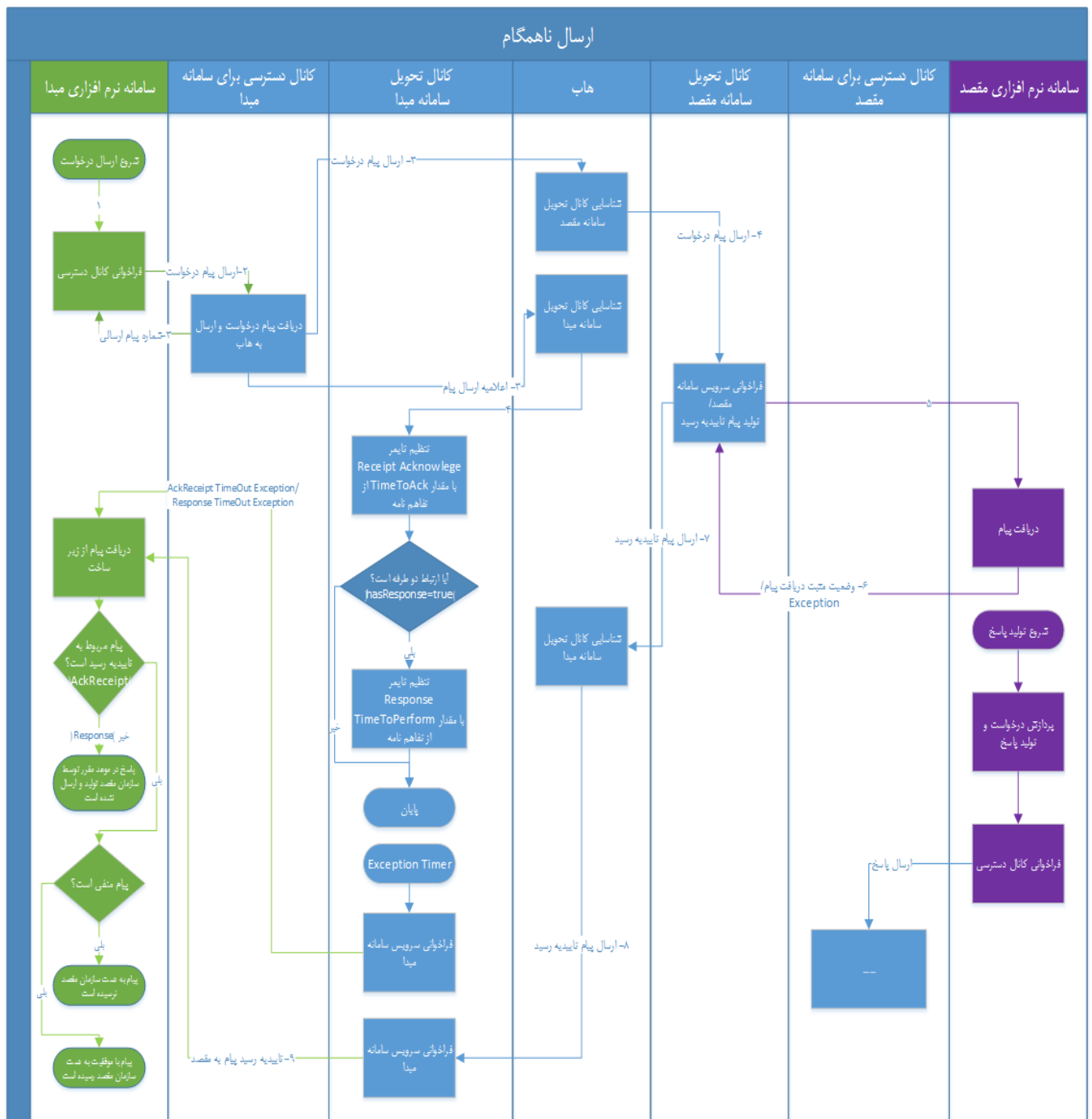
- زمان انتظار برای بررسی درخواست و ارسال پاسخ (TimeToPerform): پارامتری است که در تفاهم نامه‌ی دریافت سرویس توسط سازمان ارائه‌کننده‌ی سرویس تنظیم شده و نشان‌دهنده‌ی سطح خدمتی است که توسط ارائه‌دهنده‌ی سرویس قابل انجام است.

براین اساس هنگامی که کانال دسترسی زیرساخت توسط سامانه‌ی نرم‌افزاری مبداء فراخوانی شد، زیرساخت پیام را طریق هاب داخلی و کانال تحویل مربوطه به سامانه‌ی نرم‌افزاری مقصد تحویل می‌نماید. همزمان با این عمل زیرساخت یک تایمر (با مقدار تعیین شده در تفاهم نامه بین دو سازمان) مقداردهی می‌نماید. هنگامی که دریافت پاسخ از سامانه‌ی نرم‌افزاری مقصد بیش از زمان انتظار تعیین شده طول بکشد، زیرساخت با ایجاد یک خطای زمان (TimeOut Exception) و ارائه‌ی آن به سامانه‌ی نرم‌افزاری مبداء، سامانه‌ی نرم‌افزاری مبداء را از عدم دریافت پاسخ توسط سامانه‌ی مقصد در مدت زمان تعیین شده، مطلع می‌سازد.

در صورتی که تعامل دوطرفه باشد، سامانه‌ی ارسال‌کننده و دریافت‌کننده، تعامل دوطرفه دارند؛ به این معنا که سامانه‌ی ارسال‌کننده می‌بایست مسیری را برای دریافت پاسخ داشته باشد. همچنین سامانه‌ی دریافت‌کننده پیام می‌بایست پاسخ سامانه‌ی

اول را در زمان مقتضی ارسال نماید. از این رو هر یک از سازمان‌های طرف تعامل هم در نقش سرویس‌دهنده و هم در نقش سرویس‌گیرنده هستند.

در صورتی که تعامل یک طرفه باشد سامانه‌ی ارسال‌کننده می‌بایست مسیری را جهت دریافت وضعیت رسید پیام و یا ارسال پیام مربوط به خطای زمان و پاسخ ایجاد نموده تا توسط زیرساخت فراخوانی گردند.



شکل ۱۰: ارسال درخواست به صورت ناهمگام از طریق زیرساخت

* در شکل بالا سناریوی ارسال پاسخ همانند ارسال درخواست طی می‌گردد.

۳-۲- امنیت

مفهوم امنیت در زیرساخت HIX دارای لایه‌های ذیل است:

- امنیت دستیابی به کانال دسترسی برای سازمان مبداء
- کنترل دستیابی به کانال‌های تحویل مربوط به سامانه‌های سازمان مقصد

۳-۲-۱- امنیت دستیابی به کانال دسترسی

پیش شرط برقراری ارتباط با زیرساخت و دریافت سرویس، تعریف سازمان سرویس‌دهنده و سرویس‌گیرنده به عنوان متعامل (Party) در زیرساخت است. در هنگام تعریف سازمان، یک شناسه منحصر به فرد (PartyID) به سازمان اختصاص می‌یابد. سازمان سرویس‌گیرنده از این پس می‌بایست با این شناسه خود را به زیرساخت معرفی نماید (SenderID = PartyID) و همچنین می‌بایست مشخصات سازمان سرویس‌دهنده را نیز توسط شناسه‌ی آن سازمان مشخص نماید (ReceiverID = PartyID).

به منظور ایجاد امنیت و همچنین کنترل ترافیک شبکه، کشور به حوزه‌های جغرافیایی و مدیریتی تقسیم‌بندی شده و به طبع آن زیرساخت نیز به صورت توزیع شده نصب و آماده‌ی ارائه‌ی سرویس شده است. در این توزیع‌شدگی به ازای هر حوزه، یک کانال دسترسی در زیرساخت تعریف شده است و سازمان‌ها براساس موقعیت جغرافیایی خود به یکی از این کانال‌های دسترسی متصل می‌باشند؛ این بدین معنی است که حتی در صورتی که یک سازمان امکان دسترسی شبکه‌ای به کانال دسترسی دیگر را داشته باشد زیرساخت از ارائه‌ی سرویس به سازمان جلوگیری می‌نماید.

دریافت و ارائه‌ی سرویس نیز در زیرساخت تنها از طریق سامانه‌های نرم‌افزاری سازمان‌ها صورت می‌پذیرد و زیرساخت رابط گرافیکی جهت ارائه به کاربران در اختیار ندارد. دریافت سرویس از زیرساخت توسط کانال دسترسی انجام شده که به ازای کلیه‌ی سرویس‌های درخواستی سازمان یکتاست؛ این بدین معنی است که سامانه‌ی نرم‌افزاری سرویس‌گیرنده کلیه‌ی سرویس‌های خود را تنها با فراخوانی یک کانال دسترسی درخواست می‌نماید و نیازی به فراخوانی سرویس‌های مختلف ندارد. به منظور تسهیل در امر شناسایی سامانه‌های نرم‌افزاری مبداء (فراخوانی‌کننده‌ی کانال دسترسی)، تنها شناسه‌ی سخت‌افزاری مربوط به سروری که نرم‌افزارهای سازمان مبداء بر روی آن قرار دارند، می‌بایست به عنوان نقاط انتهایی ارتباط به زیرساخت معرفی گردند.

لازم به ذکر است که زیرساخت نیازی به شناسایی سامانه‌های نرم‌افزاری سازمان مقصد به این شکل ندارد زیرا به ازای هر یک از سرویس‌های ارائه شده توسط آنها اقدام به شخصی‌سازی کانال تحویل می‌نماید. این شخصی‌سازی می‌تواند حتی در سطح پیاده‌سازی سیاست‌های امنیتی برای هر سازمان و برای هر سرویس به طور جداگانه انجام شود.



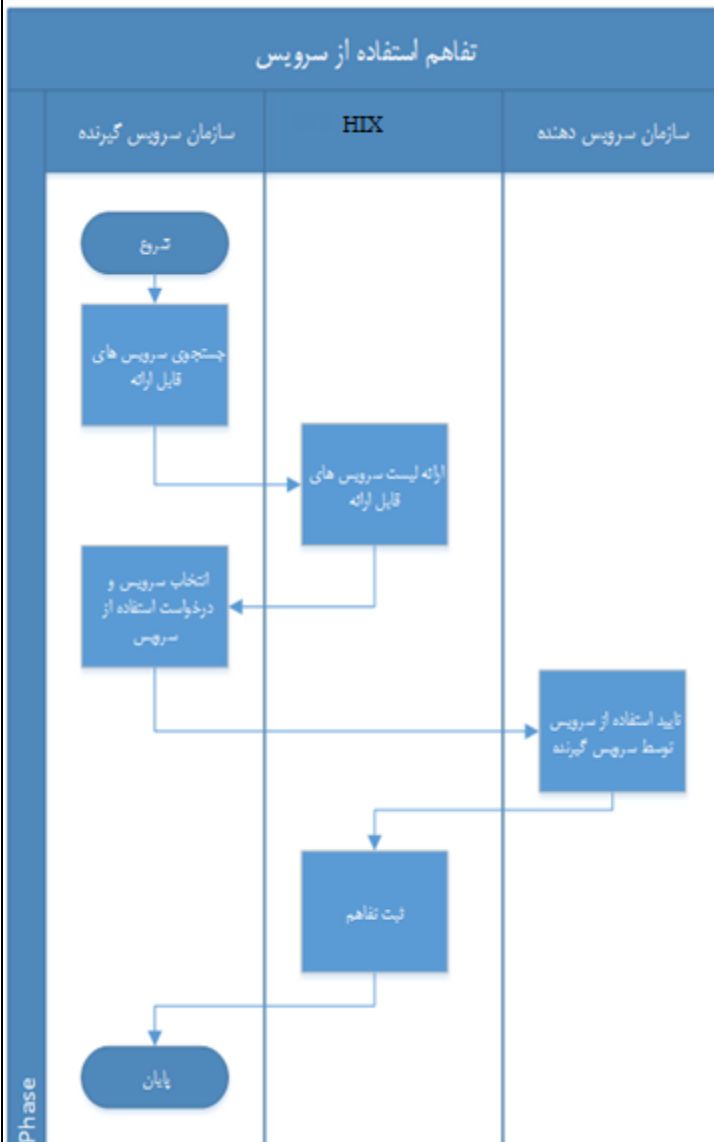
شکل ۱۱ - رابطه بین سرویس‌های ارائه شده توسط سامانه مقصد و کانال تحویل مربوط به زیرساخت

۲-۲-۳- امنیت دستیابی به سرویس‌های مقصد

سازمان‌های سرویس‌دهنده سرویس‌های خود را بر روی زیرساخت منتشر می‌نمایند. آیا این بدان معنی است که زیرساخت توان اتصال به سرویس‌های سرویس‌دهنده را داراست و در صورتی که هر سازمان دیگری به زیرساخت دسترسی داشته باشد لاجرم به سرویس‌های منتشر شده بر روی زیرساخت نیز دسترسی دارد؟

جواب این سؤال خیر است؛ سازمان سرویس‌گیرنده می‌بایست برای دریافت از سازمان سرویس‌دهنده دارای تفاهم‌نامه‌ی ثبت‌شده در زیرساخت باشد. یکی از کاربردهای تفاهم‌نامه در زیرساخت، ایجاد شرایط کنترل دسترسی به سرویس‌های سازمان سرویس‌گیرنده است. از این رو سازمان سرویس‌گیرنده می‌بایست درخواست استفاده از سرویس را به سازمان سرویس‌دهنده ارسال نموده و پس از تأیید سازمان سرویس‌دهنده، تفاهم در زیرساخت ثبت می‌شود و از آن لحظه به بعد سرویس در دسترس سازمان سرویس‌گیرنده قرار خواهد داشت. سازمان سرویس‌دهنده همچنین می‌تواند دسترسی سازمان سرویس‌گیرنده را به صورت موقت قطع نماید.

از این به بعد دو سازمان سرویس‌دهنده و سرویس‌گیرنده می‌توانند لاگ‌های مربوط به تبادل اطلاعاتشان را به ازای هر تفاهم‌نامه در زیرساخت مشاهده نمایند.



شکل شماره ۱۲: فرآیند ثبت تفاهم‌نامه‌ی استفاده از سرویس

۴- فراخوانی کانال دسترسی

سرویس‌های همگام و ناهمگام در سمت زیرساخت HIX توسط یک وب سرویس عمومی پیاده‌سازی شده‌اند. پیاده‌سازی این وب سرویس بصورت REST می‌باشد. وب سرویس مذکور در آدرسی با ساختار زیر قرار خواهد گرفت که در آن - SAPAAD SERVER-URL مربوط به محل نصب سرویس REST زیرساخت می‌باشد:

http://<SAPAAD-SERVER_URL>:8888/dispatch

توجه :

- ۱- فراخوانی وب سرویس مذکور بصورت POST (http POST) می‌باشد.
- ۲- پارامترهای json بایستی در request body اضافه گردند.
- ۳- پاسخ آماده شده برای سرویس پست در قالب فرمت UTF-8 آماده می‌گردد.
- ۴- فقط بخش‌های مربوط به Payload در ورودی متدها بایستی تکمیل گردد. اقلام دیگر به‌عنوان متغیرهای ثابت کنترلی می‌باشند.

۴-۱- فراخوانی کانال دسترسی به‌صورت همگام

کانال دسترسی به‌صورت یک وب سرویس REST در اختیار استفاده‌کنندگان قرار دارد. براین اساس فراخوانی کانال دسترسی به صورت ارسال پیام به وب سرویس و دریافت پاسخ از آن است. همان‌گونه که ذکر شد، در مدل همگام به محض فراخوانی کانال دسترسی، پیام به کانال تحویل ارسال شده و سرویس سازمان مقصد فراخوانی می‌گردد. بر این اساس فراخوانی سرویس دسترسی یا همراه با خطا خواهد بود و یا مقدار بازگشتی از سرویس مقصد در لحظه‌ی فراخوانی کانال دسترسی بازگشت داده می‌شود. فرمت اطلاعات ارسالی و دریافتی به/از وب سرویس کانال دسترسی در ادامه آمده است:

۴-۱-۱- ارسال پیام به وب سرویس کانال دسترسی

به منظور فراخوانی کانال دسترسی به صورت همگام می‌بایست پارامترهای ذیل به وب سرویس کانال دسترسی ارسال گردند:

ردیف	نام پارامتر	نوع	اجباری؟	توضیح
۱	senderId	String	اجباری	کد سازمان مبدا. شناسه‌ای است که به عنوان PartyID در زمان ثبت مشخصات متعامل به آن تخصیص داده می‌شود.
۲	receiverId	String	اجباری	کد سازمان مقصد. شناسه‌ای است که به عنوان PartyID در زمان ثبت مشخصات متعامل به آن تخصیص داده می‌شود.
۳	service	String	اجباری	نام سرویس در سازمان مقصد
۴	action	String	اجباری	نام عملیات مربوط به سرویس در سازمان مقصد
۵	version	String	اجباری	شماره نسخه مربوط به سرویس
۶	replyToConversationId	String	اختیاری	شماره پیامی است که این پیام به‌صورت پاسخ به آن ارسال می‌شود
۷	timeout	String	اختیاری	مقدار زمان تاخیر انتظار برای دریافت سرویس را به ثانیه تعیین می‌نماید.

این مقدار می‌بایست برابر و یا بزرگتر از مقدار تعیین شده در تفاهم‌نامه دریافت سرویس بین سازمان مبداء (سرویس‌گیرنده) و سازمان مقصد (سرویس‌دهنده) باشد. در صورت عدم ارائه‌ی این پارامتر مقدار تعیین شده در تفاهم، توسط زیرساخت در نظر گرفته می‌شود.				
حاوی مقادیری است که به صورت لیستی از پارامترها و یا سند به سرویس سازمان مقصد ارسال می‌گردد. این پارامتر بسته به عملیات سرویس متفاوت مقداردهی شده و شرح جزئیات تکمیل آن در اسناد مربوط به هر سرویس آورده می‌شود.	اختیاری	String	payload	۸
نشان‌دهنده‌ی ساختاری است که محتویات payload براساس آن مقداردهی شده و اعتبار سنجی می‌گردد. این پارامتر بسته به عملیات سرویس متفاوت مقداردهی شده و شرح جزئیات تکمیل آن در اسناد مربوط به هر سرویس آورده می‌شود. در صورتی که سرویس دارای مقدار ورودی نیست این پارامتر می‌بایست با مقدار ذیل تکمیل گردد: ix.dispatch.client.EmptyRequest	اجباری	String	payloadClass	۹
شناسه آشکار	اجباری	String	plainId	۱۰
شناسه امن	اجباری	String	encryptedId	۱۱

تکمیل پارامترها به منظور فراخوانی سرویس به صورت همگام در قالب JSON به صورت زیر است:

```
##request:
{
  "payload": {
    // مقادیر پارامترهای فراخوانی سرویس مقصد
  },
  "payloadClass": "نام کلاس مربوط به مورد کاربردی",
  "senderId": "کد سازمان مبداء",
  "receiverId": "کد سازمان مقصد",
  "service": "نام سرویس در سازمان مقصد",
  "action": "نام عملیات در سرویس سازمان مقصد",
  "version": "نسخه مربوط به سرویس",
  "replyToConversationId": "شماره پیام مبداء",
  "timeout": "مقدار زمان انتظار برای دریافت پاسخ",
  "plainId": "شناسه آشکار",
  "encryptedId": "شناسه امن"
}
```

* اطلاعات شناسه‌های آشکار و امن در بخش امنیت همین سند آمده است.

۲-۱-۴- بازگشت پاسخ از وب سرویس کانال دسترسی

ردیف	نام پارامتر	نوع	اجباری؟	توضیح
۱	type	String	اجباری	نوع پاسخ را تعیین می‌کند که شامل مقادیر ذیل است: <ul style="list-style-type: none"> ORDINARY : در صورتی که فراخوانی سرویس به صورت موفق انجام شده باشد. در این صورت payload حاوی مقادیر بازگشتی است. EXCEPTION : در صورتیکه فراخوانی سرویس به صورت ناموفق انجام شده باشد. در این صورت payload حاوی محتویات خطای رخ داده است.
۲	conversationId	String	اجباری	شماره منحصر به فردی است که زیرساخت به هر فراخوانی نسبت می‌دهد. (شماره پیام)
۳	payload	String	اختیاری	محتویات این پارامتر بر اساس پارامتر type تعیین می‌شود. بر این اساس این پارامتر حاوی مقادیر بازگشتی از فراخوانی سرویس و یا خطای رخ داده است.
۴	payloadClass	String	اجباری	نشان‌دهنده‌ی ساختاری است که محتویات payload براساس آن مقداردهی شده است: <ul style="list-style-type: none"> در صورتی که فراخوانی سرویس با موفقیت انجام شود این پارامتر دارای نام کلاسی خواهد بود که مرتبط با عملیات فراخوانی شده است. نام کلاس مرتبط در مستند مربوط به هر سرویس به صورت جداگانه ارائه می‌گردد. در صورتی که فراخوانی سرویس همراه با خطا باشد این پارامتر دارای مقدار ذیل است: ix.dispatcher.commons.ExceptionPayload

در صورت فراخوانی موفق، مقادیر بازگشتی از وب سرویس کانال دسترسی در قالب JSON به صورت زیر است:

```
##response:
{
  "payload": {
    //مقادیر بازگشتی//
  },
  "payloadClass": "نام کلاس مربوط به مورد کاربری",
  "type": "ORDINARY",
  "conversationId": "شماره پیام"
}
```

۲-۴- فراخوانی کانال دسترسی بصورت ناهمگام

در مدل ناهمگام سامانه‌ی نرم‌افزاری سازمان مبداء در انتظار فراخوانی سرویس سازمان مقصد توسط زیرساخت قرار نمی‌گیرد.

براین اساس پس از فراخوانی موفق زیرساخت یک شماره پیام به سامانه نرم‌افزاری سازمان مبداء بازگشت داده و فراخوانی کانال دسترسی تکمیل می‌گردد. در ادامه‌ی فرآیند، زیرساخت نتیجه‌ی فراخوانی سرویس سازمان مقصد را از طریق کانال تحویل به سامانه‌ی نرم‌افزاری سازمان مبداء اطلاع می‌دهد.

علاوه بر این در مدل ناهمگام این امکان وجود دارد تا پیام به صورت همزمان برای سازمان‌های مختلف ارسال گردد؛ یعنی سازمان مبداء از زیرساخت درخواست می‌نماید تا سازمان‌های مقصد بیش از یک عدد باشند، در این صورت زیرساخت مجاب می‌گردد تا سرویس مقصد در سازمان‌های مختلفی را فراخوانی نموده و پیام دریافتی را به آنها تحویل نماید. در ادامه فراخوانی کانال دسترسی به صورت ناهمگام در دو حالت ذیل توصیف می‌گردد:

- ارسال پیام به یک مقصد
- ارسال پیام به چند مقصد

۴-۲-۱- ارسال پیام به یک مقصد

در این مدل، گیرنده‌ی پیام تنها یک سازمان مقصد است.

۴-۲-۲- ارسال پیام به وب سرویس کانال دسترسی

ردیف	نام پارامتر	نوع	اجباری؟	توضیح
۱	senderId	String	اجباری	کد سازمان مبداء
۲	senderService	String	اجباری	مشخصات سرویسی است که سامانه‌ی نرم‌افزاری جهت دریافت نتیجه ارسال به زیرساخت معرفی می‌نماید
۳	senderVersion	String	اجباری	
۴	receiverId	String	اجباری	کد سازمان مقصد
۵	service	String	اجباری	نام سرویس در سازمان مقصد
۶	action	String	اجباری	نام عملیات مربوط به سرویس در سازمان مقصد
۷	version	String	اجباری	شماره نسخه مربوط به سرویس
۸	replyToConversationId	String	اختیاری	شماره پیامی است که این پیام به صورت پاسخ به آن ارسال می‌شود
۹	hasResponse	Boolean	اجباری	تعامل یک طرفه و یا دو طرفه را به زیرساخت اعلام می‌نماید.
۱۰	timeToAck	Number	اختیاری	مقدار زمان تاخیر انتظار برای آگاهی از وضعیت ارسال پیام به سازمان مقصد را به ثانیه تعیین می‌نماید. این مقدار می‌بایست برابر و یا بزرگتر از مقدار تعیین شده در تفاهم دریافت سرویس بین سازمان مبداء (سرویس گیرنده) و سازمان مقصد (سرویس دهنده) باشد. در صورت عدم ارائه این پارامتر، مقدار تعیین شده در تفاهم نامه توسط زیرساخت برای فراخوانی در نظر گرفته می‌شود.
۱۱	timeToPerform	Number	اختیاری	در تعاملات دو طرفه مقدار زمان تاخیر انتظار برای دریافت پیام از سازمان مقصد در پاسخ به پیام ارسالی را به ثانیه تعیین می‌نماید. این مقدار می‌بایست برابر و یا بزرگتر از مقدار تعیین شده در تفاهم نامه دریافت سرویس بین سازمان مبداء (سرویس گیرنده) و

ردیف	نام پارامتر	نوع	اجباری؟	توضیح
				سازمان مقصد (سرویس دهنده) باشد. در صورت عدم ارائه‌ی این پارامتر مقدار تعیین شده در تفاهم نامه توسط زیرساخت برای فراخوانی در نظر گرفته می‌شود.
۱۲	payload	String	اختیاری	حاوی مقادیری است که به صورت لیستی از پارامترها و یا سند به سرویس سازمان مقصد ارسال می‌گردد. این پارامتر بسته به عملیات سرویس متفاوت مقداردهی شده و شرح جزئیات تکمیل آن در اسناد مربوط به هر سرویس آورده می‌شود.
۱۳	payloadClass	String	اجباری	نشان‌دهنده‌ی ساختاری است که محتویات payload براساس آن مقداردهی شده و اعتبارسنجی می‌گردد. این پارامتر بسته به عملیات سرویس متفاوت مقداردهی شده و شرح جزئیات تکمیل آن در اسناد مربوط به هر سرویس آورده می‌شود. در صورتی که سرویس دارای مقدار ورودی نیست این پارامتر می‌بایست با مقدار ذیل تکمیل گردد: ix.dispatch.client.EmptyRequest
۱۴	plainId	String	اجباری	شناسه آشکار
۱۵	encryptedId	String	اجباری	شناسه امن

فراخوانی سرویس کانال دسترسی به صورت ناهمگام در قالب JSON به صورت زیر است:

```
##request:
{
  "payload" : {
    // مقادیر پارامترهای فراخوانی سرویس مقصد
  },
  "payloadClass" : "نام کلاس مربوط به مورد کاربردی",
  "senderId" : "کد سازمان مبدا",
  "senderService" : "نام سرویس سازمان مبدا",
  "senderVersion" : "نسخه مربوط به سرویس سازمان مبدا",
  "receiverId" : "کد سازمان مقصد",
  "service" : "نام سرویس در سازمان مقصد",
  "action" : "نام عملیات در سرویس سازمان مقصد",
  "version" : "نسخه مربوط به سرویس سازمان مقصد",
  "replyToConversationId" : "شماره پیام مبدا",
  "hasResponse" : true/false,
  "timeToAck" : "مدت زمان انتظار برای اعلام وضعیت پیام ارسال شده",
  "timeToPerform" : "مدت زمان انتظار برای دریافت پیام پاسخ در تعاملات دو طرفه",
  "plainId" : "شناسه آشکار",
  "encryptedId" : "شناسه امن",
}
```


۴-۲-۲-۱- بازگشت پاسخ از وب سرویس کانال دسترسی

ردیف	نام پارامتر	نوع	اجباری؟	توضیح
۱	type	String	اجباری	نوع پاسخ را تعیین می‌کند که شامل مقادیر ذیل است: <ul style="list-style-type: none"> ORDINARY : در صورتی که فراخوانی سرویس به صورت موفق انجام شده باشد. در این صورت payload حاوی مقادیر بازگشتی است. EXCEPTION : در صورتیکه فراخوانی سرویس به صورت ناموفق انجام شده باشد. در این صورت payload حاوی محتویات خطای رخ داده است.
۲	conversationId	String	اجباری	شماره‌ی منحصر به فردی است که زیرساخت به هر فراخوانی نسبت می‌دهد.
۳	payload	String	اجباری	محتوی مقدار ثابت received
۴	payloadClass	String	اجباری	محتوی مقدار ثابت java.lang.String

سند پاسخ آدرس پست در قالب JSON به صورت زیر است:

##response:

```
{
  "payload": "received",
  "payloadClass": "java.lang.String",
  "type": "ORDINARY",
  "conversationId": "شماره پیام ارسالی"
}
```

۴-۲-۳-۱- ارسال پیام به چند مقصد

در این مدل، گیرنده‌ی پیام بیش از یک سازمان مقصد می‌باشد. فراخوانی کننده‌ی کانال دسترسی در این مدل قصد دارد تا یک پیام را به چندین سازمان مقصد ارسال کند. براین اساس پیام را یک بار تحویل کانال دسترسی داده و از آن درخواست می‌کند تا پیام را به چندین سازمان مقصد ارسال نماید. کانال دسترسی برای هر مقصد یک پیام مجزا تولید نموده و آن‌ها را به صورت مستقل ارسال می‌نماید. لازم به ذکر است که از این روش برای ارسال پاسخ در پیام‌های دوطرفه نمی‌توان استفاده نمود و تنها در مورد پیام‌های درخواست کاربرد دارد.

۴-۲-۳-۱- ارسال پیام به وب سرویس کانال دسترسی

ردیف	نام پارامتر	نوع	اجباری؟	توضیح
۱	senderId	String	اجباری	کد سازمان مبدا.
۲	senderService	String	اجباری	مشخصات سرویسی است که سامانه‌ی نرم‌افزاری جهت دریافت نتیجه‌ی ارسال به زیرساخت معرفی می‌نماید.
۳	senderVersion	String	اجباری	

ردیف	نام پارامتر	نوع	اجباری؟	توضیح
۴	receiverIds	ListString	اجباری	کد سازمان‌های مقصد به صورت لیست.
۵	service	String	اجباری	نام سرویس در سازمان مقصد
۶	action	String	اجباری	نام عملیات مربوط به سرویس در سازمان مقصد
۷	version	String	اجباری	شماره نسخه مربوط به سرویس
۸	hasResponse	Boolean	اجباری	تعامل یک طرفه و یا دوطرفه را به زیرساخت اعلام می‌نماید.
۹	timeToAck	Number	اختیاری	مقدار زمان تاخیر انتظار برای آگاهی از وضعیت ارسال پیام به سازمان مقصد را به ثانیه تعیین می‌نماید. این مقدار می‌بایست برابر و یا بزرگتر از مقدار تعیین شده در تفاهم‌نامه‌ی دریافت سرویس بین سازمان مبداء (سرویس گیرنده) و سازمان مقصد (سرویس دهنده) باشد. در صورت عدم ارائه این پارامتر، مقدار تعیین شده در تفاهم‌نامه توسط زیرساخت برای فراخوانی در نظر گرفته می‌شود.
۱۰	timeToPerform	Number	اختیاری	در تعاملات دو طرفه مقدار زمان تاخیر انتظار برای دریافت پیام از سازمان مقصد در پاسخ به پیام ارسالی را به ثانیه تعیین می‌نماید. این مقدار می‌بایست برابر و یا بزرگتر از مقدار تعیین شده در تفاهم‌نامه دریافت سرویس بین سازمان مبداء (سرویس گیرنده) و سازمان مقصد (سرویس دهنده) باشد. در صورت عدم ارائه این پارامتر، مقدار تعیین شده در تفاهم توسط زیرساخت برای فراخوانی در نظر گرفته می‌شود.
۱۱	payload	String	اختیاری	حاوی مقادیری است که به صورت لیستی از پارامترها و یا سند به سرویس سازمان مقصد ارسال می‌گردد. این پارامتر بسته به عملیات سرویس متفاوت مقداردهی شده و شرح جزئیات تکمیل آن در اسناد مربوط به هر سرویس آورده می‌شود.
۱۲	payloadClass	String	اجباری	نشان‌دهنده‌ی ساختاری است که محتویات payload براساس آن مقداردهی شده و اعتبار سنجی می‌گردد. این پارامتر بسته به عملیات سرویس متفاوت مقداردهی شده و شرح جزئیات تکمیل آن در اسناد مربوط به هر سرویس آورده می‌شود. در صورتی که سرویس دارای مقدار ورودی نیست این پارامتر می‌بایست با مقدار ذیل تکمیل گردد: ix.dispatch.client.EmptyRequest
۱۳	plainId	String	اجباری	شناسه آشکار
۱۴	encryptedId	String	اجباری	شناسه امن

فراخوانی سرویس کانال دسترسی به صورت ناهمگام در قالب JSON بصورت زیر است:

```
##request:
```

```
{
```

```
  "payload" : {
```

```
    // مقادیر پارامترهای فراخوانی سرویس مقصد
```

```

},
"payloadClass": "نام کلاس مربوط به مورد کاربردی",
"senderId": "کد سازمان مبداء",
"senderService": "نام سرویس سازمان مبداء",
"senderVersion": "نسخه مربوط به سرویس سازمان مبداء",
"receiverIds": ["کد سازمان مقصد ۱", "کد سازمان مقصد ۲", ...],
"service": "نام سرویس در سازمان مقصد",
"action": "نام عملیات در سرویس سازمان مقصد",
"version": "نسخه مربوط به سرویس سازمان مقصد",
"hasResponse": true/false,
"timeToAck": "مدت زمان انتظار برای اعلام وضعیت پیام ارسال شده",
"timeToPerform": "مدت زمان انتظار برای دریافت پیام پاسخ در تعاملات دو طرفه",
"plainId": "شناسه آشکار",
"encryptedId": "شناسه امن",
}
    
```

۲-۳-۴- بازگشت پاسخ از وب سرویس کانال دسترسی

ردیف	نام پارامتر	نوع	اجباری؟	توضیح
۱	type	String	اجباری	محتوی مقدار ثابت MULTI_RECIPIENT. از آنجایی که به ازای هر سازمان مقصد یک پیام توسط زیرساخت تولید می‌گردد، مشخصات هر پیام به همراه کد سازمان مقصد مربوطه در پارامتر response ارائه می‌گردد.
۲	conversationId	String	اجباری	به دلیل چند پیمای مشخصات هر پیام جداگانه در پارامتر payload ارائه می‌گردد و این پارامتر دارای مقدار null است.
۳	payload	String	اجباری	محتوی یک پارامتر به نام conversationsDetails که شامل لیستی از objectها به شکل پاسخ کامل در فراخوانی‌های با یک مقصد به همراه کد سازمان مقصد مرتبط با پیام است.
	conversationsDetails	List	اجباری	محتوی زوج مرتب receiverId و response به ازای هر سازمان ارائه شده در پارامتر receiverIds به هنگام فراخوانی کانال دسترسی است.
	receiverId	String	اجباری	کد سازمان مقصد
	response	String	اجباری	دقیقاً مانند پاسخ کامل در فراخوانی با یک مقصد می‌باشد
۴	payloadClass	String	اجباری	محتوی مقدار ثابت ix.dispatch.client.multirecipient.MultiRecipientResponse

سند پاسخ آدرس پست در قالب JSON به صورت زیر است:

```

##response:
{
    
```

```

"payload" : {
  "conversationsDetails" : [ {
    "receiverId" : "کد سازمان مقصد",
    "response" : {
      "payload" : "received",
      "payloadClass" : "java.lang.String",
      "type" : "ORDINARY",
      "conversationId" : "شماره پیام ارسالی"
    }
  } ]
},
"payloadClass" : "ix.dispatch.client.multirecipient.MultiRecipientResponse",
"type" : "MULTI_RECIPIENT",
"conversationId" : null
}
    
```

* نکته : در هنگام فراخوانی کانال دسترسی، زیرساخت به ازای هر سازمان مقصد یک پیام تولید نموده و نتیجه‌ی ارسال و یا عدم ارسال پیام را به صورت لیست گزارش می‌نماید. از این رو سامانه‌ی سازمان مبداء می‌بایست امکان دریافت گزارش ارسال به صورت لیست را دارا بوده و شماره‌های دریافتی پیام را به ازای هر سازمان مقصد دریافت نموده و به منظور پیگیری‌های آتی نگهداری نماید.

۴-۲-۴- تاییدیه رسید پیام به مقصد در ارسال ناهمگام

زمانی که پیام توسط زیرساخت به مقصد ارسال می‌شود سرویس مقصد فراخوانی شده و سعی در تحویل پیام به سازمان مقصد می‌کند. در این فرآیند حالت‌های ذیل اتفاق می‌افتد:

- پیام به دست سرویس سازمان مقصد می‌رسد: در این صورت زیرساخت با ارسال یک پیام، به ارسال کننده‌ی پیام اطلاع می‌دهد که پیام شما به مقصد تحویل شد. فرمت پیام مذکور که به آن **Positive Acknowledge Receipt** اطلاق می‌شود مانند ذیل است:

```

## response
{
  "reason" : null,
  "status" : true,
  "type" : "RECEIPT",
  "ackToConversationId" : "شماره پیام ارسال شده"
}
    
```

- پیام به دست مخاطب نمی‌رسد: در این صورت زیرساخت با ارسال یک پیام، به ارسال کننده‌ی پیام اطلاع می‌دهد که پیام شما به مقصد تحویل نشد. فرمت پیام مذکور که به آن **Negative Acknowledge Receipt** اطلاق می‌شود، به شکل کلی زیر است:

```

## response
{
  "reason" : {
    "exceptionCode" : "exceptionCode",
    "exceptionMessage" : "exceptionMessage",
    "exceptionStackTrace" : " exceptionStackTrace ",
  },
}
    
```

```
"status" : false,  
"type" : "RECEIPT",  
"ackToConversationId" : " شماره پیام ارسال شده "  
}
```

* نکته : فرآیند اعلام Positive Acknowledge و Negative Acknowledge در ارسال پیام به یک سازمان و یا چند سازمان به دلیل استقلال پیام ها از یکدیگر از دیدگاه زیرساخت و یا سامانه‌ی نرم‌افزاری مبداء هیچ تفاوتی ندارد.

۵-۲-۴ - خطاهای زمانی در ارسال ناهمگام

همان‌طور که پیشتر گفته شد در ارسال ناهمگام دو زمانبندی وجود دارد:

- اولی مربوط به حداکثر انتظار فرستنده پیام برای تحویل پیام به مقصد می‌باشد. این پارامتر که در تفاهم‌نامه‌ی بین سازمان سرویس‌گیرنده و سازمان سرویس‌دهنده تعیین می‌شود زیرساخت را متعهد می‌کند تا زمان‌بندی ارسال را تعیین کرده و پس از زمان سپری شده، فرستنده‌ی پیام را از وقوع این رخداد مطلع نماید.

```
## response  
{  
  "reason" : {  
    "exceptionCode" : "timeout.occurred",  
    "exceptionMessage" : "the timeout has occurred",  
    "exceptionStackTrace" : null,  
  },  
  "status" : false,  
  "type" : "RECEIPT",  
  "ackToConversationId" : " شماره پیام ارسال شده "  
}
```

* در صورتی که زیرساخت قبل از سررسیدن زمان تایمر، پیام Ack (مثبت/منفی) صادر نماید تایمر غیر فعال می‌شود.

* در صورتی که این تایمر قبل از ارسال تأییدیه‌های رسید (مثبت یا منفی) فعال گردد، پیام‌های مذکور Mask شده و به اطلاع فرستنده‌ی پیام نمی‌رسند.

- دومی مربوط به حداکثر زمان انتظار فرستنده‌ی پیام برای دریافت پاسخ از مقصد است. این پارامتر در تفاهم‌نامه‌ی بین سازمان سرویس‌گیرنده و سازمان سرویس‌دهنده تعیین می‌شود. در صورتی که فرستنده، پارامتر `hasResponse = true` را در هنگام ارسال پیام به کانال دسترسی تنظیم نماید، زیرساخت را متعهد می‌کند تا زمان‌بندی دریافت پاسخ را تعیین کرده و پس از زمان سپری شده فرستنده‌ی پیام را از وقوع این رخداد مطلع نماید.

```
## response  
{  
  "reason" : {
```

```
"exceptionCode" : "timeout.occurred",
"exceptionMessage" : "the timeout has occurred",
"exceptionStackTrace" : null,
},
"status" : false,
"type" : "PERFORMED",
"ackToConversationId" : "شماره پیام ارسال شده"
}
```

* از دید زیرساخت هر پیامی که توسط گیرنده‌ی پیام به سمت فرستنده ارسال می‌شود و فیلد replyToConversationId آن به شماره پیامی اشاره کند که تایمر پاسخ برای آن فعال شده است، در صورتی که پیام قبل از سررسیدن زمان تایمر دریافت شود، زیرساخت با فرض اینکه پاسخ پیام ارسالی دریافت شده، تایمر را غیرفعال می‌کند.

* در صورتی که این تایمر فعال گردد از ارسال پاسخ توسط گیرنده به فرستنده جلوگیری نمی‌شود.

۴-۳- مدیریت خطا به صورت عمومی

از آنجایی که زیرساخت HIX رابط بین سرویس گیرنده و سرویس دهنده است دو دسته خطا ممکن است در ارتباط رخ دهد:

- خطاهای ایجاد شده توسط کانال دسترسی : این خطاها مربوط به زیرساخت بوده و به سرویس دهنده‌ی نهایی مربوط نمی‌گردند. خطاهایی که توسط کانال دسترسی تولید می‌شوند و عموماً دارای طبقه بندی ذیل هستند:
 - خطای نقص در پارامترهای فراخوانی سرویس کانال دسترسی
 - خطای امنیتی دسترسی به زیرساخت و یا سرویس نهایی
 - خطای عدم برپایی سرویس دهنده نهایی
 - خطای زمانی
 - خطای داخلی
- خطاهای ایجاد شده توسط سرویس دهنده‌ی نهایی : این خطاها مربوط به زیرساخت نبوده و به سرویس دهنده‌ی نهایی (سرویس سازمان مقصد) مربوط می‌گردند. در این حالت زیرساخت در فراخوانی سرویس مقصد دچار خطا می‌شود. در این سند خطاهای زیرساخت تشریح شده و به منظور آگاهی از خطاهای تولید شده توسط سرویس دهنده‌ی نهایی می‌بایست به اسناد مربوط به سرویس مذکور مراجعه گردد.

۴-۳-۱- بروز خطا در فراخوانی کانال دسترسی

در صورت بروز خطا در فراخوانی سرویس کانال دسترسی، پیام زیر تولید شده و بازگشت داده می‌شود:

```
## response
{
  "payload" : {
    "exceptionCode" : "exceptionCode",
    "exceptionMessage" : "exceptionMessage",
```

```
"exceptionStackTrace" : " exceptionStackTrace ",
},
"payloadClass" : "ix.dispatcher.commons.ExceptionPayload",
"type" : "EXCEPTION",
"conversationId" : "شماره پیام"
}
```

در صورتی که زیرساخت به صورت ارسال پیام به چند سازمان فراخوانی گردد، ممکن است برخی از پیام ها دچار خطا بوده و مابقی بدون اشکال باشند؛ از این رو خطا داخل payload مربوط به هر سازمان مقصد اعلام می گردد. به طور مثال پاسخ زیر در فراخوانی دو مقصد ارائه گردیده است. ارسال پیام به سازمان ۱ دچار خطا شده و ارسال پیام به سازمان ۲ بدون مشکل انجام شده است:

```
##response:
{
  "payload" : {
    "conversationsDetails" : [ {
      "receiverId" : "کد سازمان مقصد ۱",
      "response" : {
        "payload" : {
          "exceptionCode" : "exceptionCode",
          "exceptionMessage" : "exceptionMessage",
          "exceptionStackTrace" : " exceptionStackTrace ",
        },
        "payloadClass" : "ix.dispatcher.commons.ExceptionPayload",
        "type" : "EXCEPTION",
        "conversationId" : "شماره پیام ارسالی به سازمان ۱"
      }
    },
    {
      "receiverId" : "کد سازمان مقصد ۲",
      "response" : {
        "payload" : "received",
        "payloadClass" : "java.lang.String",
        "type" : "ORDINARY",
        "conversationId" : "شماره پیام ارسالی به سازمان ۲"
      }
    }
  ]
},
"payloadClass" : "ix.dispatch.client.multirecipient.MultiRecipientResponse",
"type" : "MULTI_RECIPIENT",
"conversationId" : null
}
```

همانگونه که مشخص شد، در صورت بروز خطا، Payload با یک Object از نوع ExceptionPayload تکمیل شده و ارائه می گردد. exceptionCode یا کد خطا یک رشته خواهد بود و براساس جدول ذیل تفسیر می گردد.

ردیف	کد خطا (exceptionCode)	تفسیر خطا (تفسیر متن exceptionMessage)
۱	required.fields.not.found	یکی از فیلدهای اجباری پیام درخواست ارسال نشده است.
۲	system.error	ایراد سیستمی
۳	agreement.not.found	سازمان سرویس گیرنده تفاهم نامه‌ی فعالی با سازمان سرویس دهنده بر روی سرویس و عملیات درخواستی ندارد.
۴	sender.not.corresponding.dispatcher	سازمان سرویس گیرنده از کانال دسترسی مناسب متصل نشده است
۵	micro.server.unreachable	کانال تحویل سازمان سرویس دهنده موقتاً در دسترس نیست. بعداً تلاش نمایید
۶	security.pass.gen.invalid	ترکیب فیلدهای 'encryptedId, plainId, senderId' نامعتبر است
۷	agreement.time.violated	از زمان تفاهم نامه گذشته است
۸	agreement.max.request.violated	تعداد درخواستها از تعداد مشخص شده در تفاهم نامه بیشتر است
۹	agreement.temporary.deactivate	تفاهم نامه موقتاً توسط سازمان سرویس دهنده غیرفعال شده است
۱۰	timeout.occurred	خطای زمانی رخ داده است.
۱۱	receiverId.xor.receiverIds	شما می‌توانید به صورت همزمان تنها ارسال به یک مقصد یا ارسال به چند مقصد را انجام دهید.
۱۲	this.action.only.one.receiverId	این عملیات تنها می‌تواند در حالت ارسال پیام به یک مقصد استفاده شود.

لازم به ذکر است که جدول فوق مربوط به ایراداتی است که بعد از فراخوانی موفق کانال دسترسی است و توسط کانال دسترسی صادر می‌گردد. ایرادات مربوط به فراخوانی که مربوط به کدهای خطای پروتکل HTTP می‌شوند، در این جدول نیامده است و فرض می‌گردد که کدهای مذکور واضح و قابل درک هستند.

۴-۴- امنیت در فراخوانی سرویس

همانگونه که در بخش‌های قبل ذکر شد سازمان‌ها می‌بایست به عنوان متعامل در زیرساخت معرفی گردیده و نقاط انتهایی ارتباط نیز به صورت یک شناسه‌ی سخت‌افزاری برای سامانه‌های نرم‌افزاری مبداء، به زیرساخت ارائه گردد. از این رو می‌بایست در فرآیند ثبت اطلاعات متعاملین، شناسه‌ی سخت‌افزاری ذکر شده توسط سازمان به زیرساخت HIX ارائه شده تا در زیرساخت ثبت و تنظیم گردد. از این رو مراحل طی‌شونده برای فراخوانی سرویس از زیرساخت HIX دارای مراحل ذیل است:

- ثبت مشخصات سازمان به عنوان Party در زیرساخت
- صدور شناسه‌ی سخت‌افزاری مربوط به سخت‌افزاری که سامانه(های) نرم‌افزاری مبداء بر روی آن نصب شده است.

- پیاده‌سازی کتابخانه (Partner Security API (PS API
- یکپارچه‌سازی PS API توسط سازمان
- نوع پروتکل‌ها

کتابخانه PS API شامل فایل‌های زیر می‌باشد که در اختیار تأمین‌کننده قرار می‌گیرد.

ردیف	شرح فایل	نام فایل
۱	فایل مربوط به دستگاه‌های ۳۲ بیتی برای استفاده در Windows و برنامه‌های غیر از Java مانند Delphi ، C ، و ...	BPJWSSec32.dll
۲	فایل مربوط به دستگاه‌های ۶۴ بیتی برای استفاده در Windows و برنامه‌های غیر از Java مانند Delphi ، C ، و ...	BPJWSSec64.dll
۳	فایل مربوط به دستگاه‌های ۳۲ بیتی برای استفاده در Windows و برنامه‌های Java	JniBPJWSSec32.dll
۴	فایل مربوط به دستگاه‌های ۶۴ بیتی برای استفاده در Windows و برنامه‌های Java	JniBPJWSSec64.dll
۵	فایل مربوط به دستگاه‌های ۶۴ بیتی برای استفاده در Linux و برنامه‌های غیر از Java	libBPJWSSec64.so
۶	فایل مربوط به دستگاه‌های ۶۴ بیتی برای استفاده در Linux و برنامه‌های Java	libJniBPJWSSec64.so
۷	فایل جاوا برای کار با فایل‌های ردیف ۳ ، ۴ و ۶ این فایل بایستی در برنامه کلاینت و در package زیر قرار گیرد bjp.secdep.secutil فایل‌های dll نیز بایستی در مسیر پروژه و یا کنار jar فایل اجرایی برنامه قرار گیرند ** اجرای برنامه‌های جاوا و به‌کارگیری این کتابخانه در نسخه جاری برای JDK 1.7+ امکان پذیر می‌باشد	JniBPJWSSec.java
۸	فایل مربوط به برنامه‌هایی که بر روی IIS سرور استقرار می‌یابند. به همراه این کتابخانه یک پوشه حاوی ۴ dll دیگر قرار دارد که این پوشه بایستی در کنار این کتابخانه قرار بگیرد.	BPJWSSecAnyCpu.dll

اولین گام برای راه‌اندازی، انتخاب کتابخانه PS می‌باشد که سازمان بایستی ضمن انتخاب درست فایل‌های مورد نیاز خود بسته به مشخصات پلت فرم سرویس‌دهنده‌ی خود، فرایند ساخت نسخه و استقرار آن را به‌درستی انجام دهد.
در این مدل تعامل PS API وضعیت Client نسبت به زیرساخت HIX را دارد. وضعیت Client به معنی اینکه نرم‌افزار تأمین‌کننده وب سرویس سوئیچ را فراخوانی نموده و بایستی الزامات امنیتی سوئیچ و استفاده از کتابخانه PS را مطابق با این دستورالعمل لحاظ نماید.

۱-۴-۴- ثبت نام هویت سازمان و نرم‌افزار مبداء فراخوانی کننده

سازمان درخواست‌کننده‌ی اتصال باید درخواست اتصال به سامانه را طی تشریفات تفاهم‌نامه تحویل دهد. درخواست مذکور شامل اطلاعات هویتی و مختصات فنی سازمان می‌باشد که براساس آن سرفصل هویتی در سامانه‌ی مدیریت سوئیچ تعریف می‌گردد.

سازمان درخواست‌کننده‌ی اتصال، بعد از انتخاب سرویس‌دهنده‌ی خود، برای ثبت آن، شناسه‌ی واحد آن را در درخواست خود

درج و یا به کارشناسان واحد امنیت ابلاغ نماید. برای استخراج شناسه‌ی فوق که منطبق با سخت‌افزار سرویس‌دهنده است بایستی از کتابخانه PS استفاده گردد. فرایند به‌کارگیری کتابخانه شامل دو مرحله کلی می‌باشد:

الف (مرحله‌ی راه‌اندازی، یکپارچه‌سازی و استقرار (Initialization Step)

ب (مرحله‌ی فراخوانی وب‌سرویس‌های سوئیچ زیرساخت HIX (Usage Step)

در مرحله‌ی راه‌اندازی، تأمین‌کننده بایستی براساس جدول فایل‌ها و منطبق با پلت فرم سرویس‌دهنده، کتابخانه‌های مطابق با نرم‌افزار خود را انتخاب و فرایند یکپارچه‌سازی نرم‌افزار با کتابخانه‌های PS را اجرا نماید.

لازم به ذکر است که در زمان به‌کارگیری کتابخانه‌ی PS API برای فراخوانی وب سرویس سوئیچ، نقش برنامه‌ی تأمین‌کننده، Client بوده و نمونه کدهای مربوط به آن قابل بهره‌برداری می‌باشد.

۲-۴-۴- صدور شناسه و سرفصل امنیتی برای متعاملین سوئیچ

هویت دیجیتال صادره برای مخاطبین سوئیچ در سه سطح امنیتی توکن نرم، توکن سخت و گذرواژه‌ی پویا ارائه می‌گردد و بنا به بضاعت، زیرساخت‌ها، الزامات، سیاست‌ها و مختصات فنی سازمان‌ها متفاوت و منعطف می‌باشد.

بعد از تأیید درخواست اتصال سازمان توسط سوئیچ، اعتبارنامه اتصال به وب‌سرویس در اختیار سازمان مربوطه قرار خواهد داد. گام‌های یکپارچه‌سازی با کتابخانه توابع:

۱. قرار دادن dll یا so فایل مربوطه در برنامه‌ی تأمین‌کننده منطبق با پلت فرم برنامه‌ی مذکور با بارگذاری این کتابخانه توابع زیر در دسترس خواهد بود.

```
public class JniBPJWSSec
```

```
{  
    public native String getVer();  
    public native String getID();  
    public native String passGen2(String rand1,String param1);  
    public native boolean validate2(String _lock,String _id,String pgen2,String rand1,String param1);  
}
```

۲. استقرار برنامه‌ی نمونه بر روی سرویس‌دهنده‌ی نرم‌افزار تأمین‌کننده

۳. فراخوانی متد `getID` و به‌دست آوردن شناسه‌ی واحد سرویس‌دهنده براساس سرفصل‌های امنیتی

۴. ارسال شناسه به واحد امنیت

۵. اخذ نام کاربری یا شناسه‌ی سازمان متعامل (`SenderId`)

وب سرویس سوئیچ دارای سه پارامتر خاص برای امنیت می‌باشد:

`SenderId`: نام کاربری که در بخش قبل نحوه‌ی بدست آوردن آن بیان گردید.

`PlainId`: شناسه‌ی آشکار که به‌عنوان یک عدد تصادفی در هر بار فراخوانی یا تراکنش با وب سرویس سوئیچ توسط نرم‌افزار تأمین‌کننده بایستی تولید شود. این عدد تصادفی الزاماً باید بصورت تصاعدی باشد چرا که در غیر این صورت وب سرویس سوئیچ با خطای عدم معتبر بودن شناسه‌ی آشکار پاسخ خواهد داد.

(برای تولید شناسه آشکار می توان از فرمت ترکیبی تاریخ و زمان به شکل yyyyMMddHHmmssfff در زبان های برنامه نویسی استفاده نمود که دارای خصوصیات ذکر شده بالا نیز می باشد).

EncryptedId: شناسه ای امن که توسط متد passGen2 با پارامترهای شناسه ای آشکار و نام کاربری تولید می گردد.
String passGen2(String rand1,String param1);

rand1: شناسه آشکار

param1: نام کاربری (PartyID)

۴-۵- نمونه کد فراخوانی کانال دسترسی زیرساخت HIX به صورت همگام

۴-۵-۱- نمونه کد در .NET

نمونه کد برای فراخوانی متدهای REST در .NET :

```
using BoxBPJWSSec;
.
.
.

AutoLoad3264.path = AppDomain.CurrentDomain.BaseDirectory;

AutoLoad3264.Register("LZ4mm");
AutoLoad3264.Register("LZ4cc");

//Call A restful Service by Post Method
//Change sapaad-server with real sapaad service provider url

string senderId = "sender_id";
string receiverId = "receiver_id";
string url = "http://<SAPAAD_ADDRESS>:8888/dispatch";
string plainId = DateTime.Now.ToString("yyyyMMddHHmmssfff");
String encryptedId = LZ4Codec.passGen2(plainId, senderId);

//this values should be received from Sapaad and You get them
//in your web application and save them to reuse in calling our web services
var reqData = new
{
    payload = new { PAYLOAD_CLASS },
    //fill payload variable with with parmeters that you want to send to target
    payloadClass = "payload_class_name",
    senderId= senderId,
    receiverId= receiverId,
    service= "service_name", //fill with destination service name
    action= "action_name", //fill with destination action name
    version= "version", //fill with destination service version number
    timeout= "time_out", //time for waiting for get service
    plainId= plainId,
    encryptedId= encryptedId
};

//Make Request
WebRequest request = WebRequest.Create(url);
request.Method = "POST";
```

```
request.ContentType = "application/json";

//serialize request in JSON by JavaScriptSerializer class and get the string value
JavaScriptSerializer serializer = new JavaScriptSerializer();
string s = serializer.Serialize(reqData);

//Convert the string to utf-8 byte[]
byte[] byteArray = System.Text.UTF8Encoding.Unicode.GetBytes(s);
request.ContentLength = byteArray.Length;

//send Request and Get Response
Stream reqStream = request.GetRequestStream();
reqStream.Write(byteArray, 0, byteArray.Length);
reqStream.Close();

WebResponse response = request.GetResponse();
Stream resStream = response.GetResponseStream();
StreamReader reader = new StreamReader(resStream);
// the response is in Json format which includes payload result or Exception
string resString = reader.ReadToEnd();
reader.Close();
```

۲-۵-۴ - نمونه کد در JAVA

نمونه کد برای فراخوانی متدهای REST در JAVA:

```
public void callPost(){

    JniBPJWSSec _j1=new JniBPJWSSec();

    public static String SERVER="http://sapaad_address:8888/dispatch";
    public static String payloadClass="payload_class_name";
    public static String senderId="sender_id";
    public static String receiverId="receiver_Id";
    public static String serviceName="service_name";
    public static String actionName="action_name";
    public static String version="version";
    public static String timeout="time_out";

    String plainId = new SimpleDateFormat("yyyyMMddHHmmssS").format(new Date());
    String encryptId=_j1.passGen2(plainId.toString(), senderId);

    RestTemplate rest = new RestTemplate();
    Map<String, Object> params = new HashMap<String, Object>();
    Map<String, Object> payloads = new HashMap<String, Object>();

    payloads.put("var_name", <VAR_VALUE>);
    // fill variable map value for send as payload to target service

    params.put("payload", payloads);
    params.put("payloadClass", payloadClass);
    params.put("senderId", senderId);
    params.put("receiverId", receiverId);
    params.put("service", serviceName);
    params.put("action", actionName);
    params.put("version", version);
    params.put("timeout", timeout);
    params.put("plainId", plainId);
    params.put("encryptedId", encryptId);
```

```

Map response = rest.postForObject(SERVER, params, Map.class);
System.out.println(response.toString());
}
    
```

۴-۶- سرویس‌های زیر ساخت

به منظور تسهیل تعامل سامانه‌های نرم افزاری بایکدیگر، در این نسخه چند سرویس ارزشمند جهت ارائه به سامانه‌های نرم‌افزاری متعامل ارائه گردیده است.

۴-۶-۱ دریافت لیست تفاهم‌نامه‌های مربوط به سازمان

براساس این سرویس سازمان‌ها می‌توانند توسط سامانه‌های نرم‌افزاری خود لیست تفاهم‌نامه‌های مربوط به خود در حالت سرویس‌دهنده و سرویس‌گیرنده را دریافت نمایند.

ردیف	نام پارامتر	نوع	اجباری؟	توضیح
۱	senderId	String	اجباری	کد سازمان
۲	receiverId	String	اجباری	مقدار ثابت ix-almighty
۳	service	String	اجباری	مقدار ثابت repo-central
۴	action	String	اجباری	مقدار ثابت loadPartyAgreements
۵	version	String	اجباری	مقدار ثابت ۱.۰.۰
۶	timeout	Number	اختیاری	حداکثر زمان انتظار برای دریافت سرویس
۷	payload			حاوی یک پارامتر به نام partyId که در آن می‌بایست کد سازمان تکمیل گردد. این سرویس تفاهم‌نامه‌های دریافت سرویس برای این party را گزارش می‌کند.
۷	payloadClass	String	اجباری	مقدار ثابت ix.microserver.repo.messages._ agreement.LoadPartyAgreementsRequest
۸	plainId	String	اجباری	شناسه آشکار
۹	encryptedId	String	اجباری	شناسه امن

فراخوانی سرویس کانال دسترسی به صورت همگام در قالب json بصورت زیر است:

```

##request:
{
  "payload": {
    "partyId": "کد سازمان مبداء"
  },
  "payloadClass": "ix.common.message.HeartbeatRequest",
  "senderId": "کد سازمان مبداء",
  "receiverId": "ix-almighty",
  "service": "repo-central",
  "action": "loadPartyAgreements",
  "version": "۱.۰.۰",
}
    
```

```

"مدت زمان انتظار برای دریافت پاسخ" : "timeout",
"شناسه آشکار" : "plainId",
"شناسه امن" : "encryptedId"
}
    
```

مقدار بازگشتی از فراخوانی سرویس کانال دسترسی به صورت جدول ذیل ارائه می گردد:

ردیف	نام پارامتر	نوع	اجباری؟	توضیح
۱	payload		اجباری	Object حاوی تفاهم نامه ها
	agreements		اجباری	محتوی لیستی از تفاهم های سازمان به صورت object
	number	String	اجباری	شماره تفاهم در زیر ساخت
	referToNumber	String	اجباری	شماره تفاهم قبلی که تفاهم جدید در ادامه آن است
	producer		اجباری	Object حاوی اطلاعات سازمان سرویس دهنده
	mcId		اجباری	Object حاوی اطلاعات سرویس و سازمان مربوطه
	partyId	String	اجباری	کد سازمان سرویس دهنده
	Service	String	اجباری	نام سرویس مورد تفاهم
	Version	String	اجباری	نسخه ی سرویس
	action	String	اجباری	نام عملیاتی از سرویس که مورد تفاهم است
	producerPartyDisplayName	String	اجباری	نام نمایشی سازمان ارائه دهنده ی سرویس
	consumerPartyId	String	اجباری	کد سازمان سرویس گیرنده
	consumerPartyDisplayName	String	اجباری	نام نمایشی سازمان سرویس گیرنده
	startDate	String	اجباری	تاریخ شروع تفاهم
	endDate	String	اختیاری	تاریخ پایان تفاهم
	timeout	Number	اجباری	مقدار زمان تاخیر تفاهم شده در دریافت سرویس در عملیات همگام. در سرویس های ناهمگام مقدار آن صفر است
	timeToAck	Number	اجباری	مقدار زمان تاخیر تفاهم شده در دریافت درخواست در عملیات ناهمگام. در عملیات همگام مقدار آن صفر است
	timeToPerform	Number	اجباری	مقدار زمان تاخیر تفاهم شده در ارسال پاسخ در عملیات ناهمگام. در عملیات همگام مقدار آن صفر است
	temporaryDeactivate	Boolean	اجباری	نشان دهنده ی این است که آیا این تفاهم موقتاً غیر فعالی است؟
	active	Boolean	اجباری	نشان دهنده ی فعال بودن یا نبودن تفاهم
	timeUnit	Number	اجباری	واحد زمانی قابلیت استفاده از سرویس شامل: <ul style="list-style-type: none"> • DAYS : به معنای روز • ...
	Duration	Number	اجباری	طول مدت زمان قابلیت استفاده از سرویس براساس واحد timeUnit
۲	type	String	اختیاری	مقدار ثابت ORDINARY
۳	payloadClass	String	اجباری	مقدار ثابت

ردیف	نام پارامتر	نوع	اجباری؟	توضیح
				ix.microserver.repo.messages.agreement._LoadPartyAgreementsResponse
۴	conversationId	String	اختیاری	Null

مقادیر بازگشتی از سرویس کانال دسترسی به صورت همگام در قالب json به صورت زیر است:

##response

```
{
  "payload" : {
    "agreements" : [
      {
        "number": "شماره تفاهم نامه ۱",
        "referToNumber": null,
        "producerPartyDisplayName": "سازمان من",
        "consumerPartyDisplayName": "سازمان ۱",
        "producer": {
          "mcId": {
            "partyId": "کد سازمان من",
            "service": "نام سرویس",
            "version": "شماره نسخه سرویس"
          }
        },
        "action": "نام عملیات مربوط به سرویس",
      },
      {
        "consumerPartyId": "کد سازمان ۱",
        "startDate": "تاریخ شروع",
        "endDate": null,
        "timeOut": 30,
        "timeToAck": 0,
        "timeToPerform": 0,
        "active": true,
        "timeUnit": "DAYS",
        "duration": 100,
        "temporaryDeactivate": false
      },
      {
        "number": "شماره تفاهم نامه ۲",
        "referToNumber": null,
        "producerPartyDisplayName": "سازمان ۲",
        "consumerPartyDisplayName": "سازمان من",
        "producer": {
          "mcId": {
            "partyId": "کد سازمان ۲",
            "service": "نام سرویس"
          }
        }
      }
    ]
  }
}
```

"version": "شماره نسخه سرویس"

},

"action": "نام عملیات مربوط به سرویس",

},

"consumerPartyId": "کد سازمان من",

"startDate": "تاریخ شروع",

"endDate": null,

"timeOut": 0,

"timeToAck": 30,

"timeToPerform": 180,

"active": true,

"timeUnit": "DAYS",

"duration": 200,

"temporaryDeactivate": false

}

]

},

"payloadClass" : "ix.microserver.repo.messages.agreement.LoadPartyAgreementsResponse",

"type" : "ORDINARY",

"conversationId" : null

}

در خروجی بالا سازمان من دارای دو تفاهم نامه است:

- در تفاهم‌نامه‌ی اول سرویس‌دهنده است؛ سازمان من در این تفاهم به مدت ۱۰۰ روز به سازمان ۱ سرویس ارائه می‌کند و سرویس فوق به صورت همگام می‌باشد.
- در تفاهم‌نامه‌ی دوم سرویس‌گیرنده است. سازمان من در این تفاهم به مدت ۲۰۰ روز از سازمان ۲ سرویس دریافت می‌کند و این سرویس به صورت ناهمگام است.

۲-۶-۴- آگاهی از وضعیت سرویس در سازمان مقصد

سازمان های سرویس‌گیرنده می‌توانند توسط سامانه‌های نرم‌افزاری خود به واسطه‌ی این سرویس از وضعیت سرویس سازمان مقصد مطلع گردند.

ردیف	نام پارامتر	نوع	اجباری؟	توضیح
۱	senderId	String	اجباری	کد سازمان مبدا
۲	receiverId	String	اجباری	کد سازمان مقصد
۳	service	String	اجباری	نام سرویس در سازمان مقصد
۴	action	String	اجباری	نام عملیات مربوط به سرویس در سازمان مقصد
۵	version	String	اجباری	شماره نسخه مربوط به سرویس
۶	timeout	Number	اختیاری	حداکثر زمان انتظار برای دریافت سرویس
۷	payloadClass	String	اجباری	مقدار ثابت ix.common.message.HeartbeatRequest

ردیف	نام پارامتر	نوع	اجباری؟	توضیح
۸	plainId	String	اجباری	شناسه آشکار
۹	encryptedId	String	اجباری	شناسه امن

فراخوانی سرویس کانال دسترسی به صورت همگام در قالب json به صورت زیر است:

##request:

```
{
  "payload" : null,
  "payloadClass" : "ix.common.message.HeartbeatRequest",
  "senderId" : "کد سازمان مبدا",
  "receiverId" : "کد سازمان مقصد",
  "service" : "نام سرویس در سازمان مقصد",
  "action" : "نام عملیات در سرویس سازمان مقصد",
  "version" : "نسخه مربوط به سرویس سازمان مقصد",
  "timeout" : "مدت زمان انتظار برای دریافت پاسخ",
  "plainId" : "شناسه آشکار",
  "encryptedId" : "شناسه امن"
}
```

مقدار بازگشتی از فراخوانی سرویس کانال دسترسی به صورت جدول ذیل ارائه می گردد:

ردیف	نام پارامتر	نوع	اجباری؟	توضیح
۱	payload			
	status	Boolean	اجباری	دارای مقادیر ذیل: <ul style="list-style-type: none"> • true : به معنای این است که سرویس دهنده‌هایی فعال است. • false : به معنای این است که کانال تحویل زیرساخت فعال است لیکن سرویس دهنده‌ی نهایی غیرفعال است.
	microServerId			مشخصات سرویس نهایی
	partyId	String	اجباری	کد سازمان مقصد
	service	String	اجباری	نام سرویس
	version	String	اجباری	شماره نسخه‌ی مربوط به سرویس
۲	type	String	اختیاری	مقدار ثابت ORDINARY
۳	payloadClass	String	اجباری	مقدار ثابت ix.common.message.HeartbeatRequest
۴	conversationId	String	اختیاری	null

مقادیر بازگشتی از سرویس کانال دسترسی به صورت همگام در قالب json به صورت زیر است:

##response

```
{
  "payload" : {
```

```
"status" : true/false,
"microServerId" : {
  "partyId" : "کد سازمان مقصد",
  "service" : "نام سرویس",
  "version" : "شماره نسخه سرویس"
}
},
"payloadClass" : "ix.common.message.HeartbeatResponse",
"type" : "ORDINARY",
"conversationId" : null
}
```

* در صورت بروز خطای زمانی (timeout) مفهوم این است که کانال تحویل سازمان مقصد غیرفعال است. به منظور آگاهی از سایر خطاهای احتمالی به بخش توصیف خطاهای فراخوانی در همین سند مراجعه شود.