فهرست

[فصل 1- مقدمه 5](#_Toc116910185)

[1-1- 3GPP و سیر تکامل آن 6](#_Toc116910186)

[1-2- چگونگی ارتباط ایجاد شده بین نسل پنجم و LTE 9](#_Toc116910187)

[فصل 2- شبکه LTE 10](#_Toc116910188)

[2-1- مقدمه 10](#_Toc116910189)

[2-2- معماری شبکه LTE 12](#_Toc116910190)

[2-3- معرفی اجزا و الگوریتمهای ارتباطی شبکه LTE 16](#_Toc116910191)

[2-3-1- معرفی اجزای شبکه LTE 16](#_Toc116910192)

[2-3-1 رابطهای شبکه LTE 18](#_Toc116910193)

[2-4- طراحی بلوک دیاگرام سامانه 21](#_Toc116910194)

[2-4-1- بلوک دیاگرام LTE همراه با جزئیات 21](#_Toc116910195)

[2-4-2- لایه‎های eNodeB 22](#_Toc116910196)

[2-4-3- لایه‎های بخش داده LTE 24](#_Toc116910197)

[2-4-4- لایه‎های بخش کنترلی LTE 25](#_Toc116910198)

[جمع‎بندی و نتیجه‎گیری 27](#_Toc116910199)

[فصل 3- معماری شبکه های نسل پنجم 29](#_Toc116910200)

[3-1 معرفی 29](#_Toc116910201)

[3-2 گزینه های معماری 5G 29](#_Toc116910202)

[3-3 معماری شبکه هسته 5G 33](#_Toc116910203)

[3-3-1- تابع مدیریت دسترسی و تحرک 37](#_Toc116910204)

[3-3-2- تابع مدیریت تماس 37](#_Toc116910205)

[3-3-3- تابع user plane 38](#_Toc116910206)

[3-3-4- معماری ذخیره سازی داده 38](#_Toc116910207)

[3-3-5- تابع کنترل خط مشی 39](#_Toc116910208)

[3-3-6- تابع دسترسی شبکه 39](#_Toc116910209)

[3-3-7- تابع مخزن شبکه 40](#_Toc116910210)

[3-3-8- انتخاب برش شبکه 40](#_Toc116910211)

[3-3-9- تابع متقابل غیر GPP3 40](#_Toc116910212)

[10-3-3 توابع پشتبان هسته 5G 41](#_Toc116910213)

[3-4- معماری 5G RAN 42](#_Toc116910214)

[3-4-1- رابط NG 46](#_Toc116910215)

[3-4-2- رابط Xn 47](#_Toc116910216)

[3-4-3- رابط E1 48](#_Toc116910217)

[3-4-4- رابط F1 49](#_Toc116910218)

[فصل 4- انواع برنامه های متن باز برای برای پیاده سازی هسته نسل 4 و 5 52](#_Toc116910219)

[4-1 هسته شبکه 52](#_Toc116910220)

[4-1-1 Evolved packet core (EPC) 52](#_Toc116910221)

[2-1-4 هسته 5G 54](#_Toc116910222)

[فصل 5- مقدمه ای بر Open5GS 57](#_Toc116910223)

[5-1- هسته های قابل اجرا در open5GS 57](#_Toc116910224)

[5-1-1- 4G/ 5G NSA Core 57](#_Toc116910225)

[5-1-2- 5G Core 59](#_Toc116910226)

[5-2- نصب Open5GS از منابع 61](#_Toc116910227)

[5-2-1 دریافت MongoDB 62](#_Toc116910228)

[5-2-2- ساخت Open5GS 63](#_Toc116910229)

[5-3- پیکربندی Open5GS 64](#_Toc116910230)

[5-3-1- هسته 5G 64](#_Toc116910231)

[5-3-2- 4G EPC 67](#_Toc116910232)

[5-4- راه اندازی Open5GS 69](#_Toc116910233)

[5-5- ایجاد WebUI Open5GS 74](#_Toc116910234)

[فصل 6- مانیتورینگ شبکه 77](#_Toc116910235)

[6-1- مانیتورینگ شبکه 77](#_Toc116910236)

[1-1-6- مزایای مانیتورینگ شبکه 78](#_Toc116910237)

[2-1-6- مانیتورینگ شبکه چگونه انجام می‌شود 80](#_Toc116910238)

[3-1-6- انواع مانیتورینگ شبکه 82](#_Toc116910239)

[4-1-6- نرم افزار‌های مانیتورینگ شبکه 83](#_Toc116910240)

[6-2- زبیکس 84](#_Toc116910241)

[1-2-6- معماری 84](#_Toc116910242)

[2-2-6- امکانات تخصصی و منحصر به فرد زبیکس 86](#_Toc116910243)

[3-2-6 - متن باز و بدون لایسنس 90](#_Toc116910244)

[4-2-6- برنامه ریزی بهتر برای استفاده از منابع 90](#_Toc116910245)

[5-2-6- روند توسعه و نسخه های زبیکس 90](#_Toc116910246)

[6-3- Log Management و معرفی ابزار های محبوب در این حوزه 91](#_Toc116910247)

[1-3-6- ابزارهای مدیریت لاگ 92](#_Toc116910248)

[6-4- پرومتئوس یا Prometheus 96](#_Toc116910249)

[1-4-6- ساختار پرومتئوس 97](#_Toc116910250)

[2-4-6- کتابخانه های مشتری و صادرکنندگان 98](#_Toc116910251)

[3-4-6- موارد استفاده از پرومتئوس 99](#_Toc116910252)

[4-4-6- پرومتئوس چه وظایفی را نمی تواند انجام دهد 99](#_Toc116910253)

[فصل 7- طراحی شبکه هسته نسل 4 و نسل 5 100](#_Toc116910254)

[7-1- پروتکل های ارتباطی 100](#_Toc116910255)

[7-1-1- پروتکل های ارتباطی هسته نسل چهارم : 100](#_Toc116910256)

[7-1-2- پروتکل های ارتباطی هسته نسل پنجم : 101](#_Toc116910257)

[7-2- جدول شکست IP شبکه 102](#_Toc116910258)

[7-3- شماتیک ارتباطی سرور ها و اجزای شبکه سلولار نسل 4 و نسل 5 108](#_Toc116910259)

[7-4- توانایی تعریف 100 هزار کاربر 109](#_Toc116910260)

[7-5- توانایی دستیابی به نرخ گذردهی تا 100 گیگابیت بر ثانیه 110](#_Toc116910261)

[7-6- دارای یک دیتابیس اصلی و یک دیتابیس ذخیره 110](#_Toc116910262)

[7-7- قابلیت Failover بودن VM ها 110](#_Toc116910263)

[7-8- طراحی سامانه به صورت User Plane مجزا 111](#_Toc116910264)

[7-9- طراحی سامانه به صورت Multi User Plane 112](#_Toc116910265)

[مراجع .......................................................................................................................................................113](#_Toc116910266)

# مقدمه

با رشد فزاینده استفاده از تلفن همراه، افزایش میزان تحرک کاربران، اتصال وسایل ارتباطی به اینترنت و تعریف کاربردهای مختلف برای این وسایل، بستر وسیعی برای نوآوری از نظر پروتکل‎های باند پهن موبایل، الگوریتم­ها، معماری و خدمات ایجاد می­شود. از طرف دیگر، رشد چند صد برابری ترافیک داده، افزایش تعداد ابزارهای متصل به شبکه­های بی‎سیم و ایجاد تنوع در سرویس­های مورد تقاضای کاربران در آینده، پیشرفت­های گسترده­ای در عرصه­ی شبکه‌های مخابرات سیار و نسل­های تلفن همراه صورت داده است. بر این اساس در این پژوهش قصد داریم تا ویژگی‎های مورد نیاز سیستم‎های مخابراتی نسل جدید و نیازمندی‎های آن‎ها را مورد بررسی قرار دهیم.

فناوری موجود 4G LTE پل اصلی تکامل به سوی شبکه های 5G است. همانطور که بعداً در فصل‌های بعدی توضیح داده خواهد شد، نرم‌افزارسازی و مجازی‌سازی سخت‌افزار 4G LTE رویکرد اصلی است که شبکه‌های نسل بعدی را قادر می‌سازد تا تعداد بیشتری از دستگاه‌های مورد انتظار را در خود جای دهند که در واقع شامل IoT، حسگر،دستگاه های ارتباطی ماشین به ماشین (M2M) و همچنین فناوری های تلفن همراه موجود و در حال ظهور می شود.

3GPP بدنه صنعتی اصلی پشت فناوری های 5G است بنابراین در ادامه ابتدا به بررسی 3GPP و سیر تکامل آن خواهیم پرداخت. علاوه بر این انجمن های صنعتی متعدد دیگری نیز در حال توسعه 5G می باشند.O-RAN Alliance، طرح نوآورانه ای است که در سال 2018 برای ایجاد طرحی باز و هوشمند برای شبکه های بی سیم نسل بعدی شکل گرفت و در حال تبدیل شدن به انجمن اصلی متمرکز بر معماری Open RAN است.

شبکه های نسل بعدی از طریق مفاهیمnetwork slicing ،containerization ، تکرار سرویس، مجازی سازی عملکرد شبکه و ابری سازی، عملکرد گسترده و استحکام اتصال را در مدت زمان طولانی تری ارائه خواهند کرد. با این حال، برای درک پیاده‌سازی راه‌حل‌های منبع باز که زیربنای توسعه شبکه‌های 5G هستند، اصول اولیه فناوری غالب 4G LTE باید درک و توضیح داده شود. اصولاً ضروری‌ترین عناصر برای درک، اجزای LTE هستند، مانند کانال‌های دسترسی که بهینه‌سازی دشوار آنها برای عملیاتی کردن و پایدار کردن شبکه تولید بسیار مهم است. درک معماری LTE ضروری است، و بنابراین، ابتدا در فصل بعدی به طور مفصل به همراه تکنیک های دسترسی، فناوری های آنتن، الگوریتم های مسیریابی و ویژگی های امنیتی شبکه های نسل چهارم توضیح داده شده است و سپس در فصل سوم به بررسی معماری 5G پرداخته شده است.

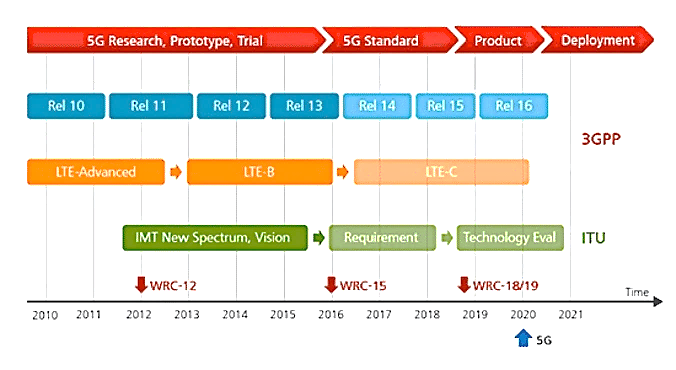
## 3GPP و سیر تکامل آن

شبکه­های ارتباطات سیار تغییر چشمگیری در مصرف داده برای کاربردهای اینترنتی ایجاد کرده است. به همین دلیل، از طرف ارائه‎دهندگان سرویس­ها، که باعث توسعه LTE شدند، نرخ داده بالایی مورد نیاز است. نسل­های قبل­تر از نسل سوم، توان ارائه داده­های زیاد به کاربران را نداشتند، از این رو LTE به عنوان یک فن‎آوری نوظهور برای تحقق خواسته‎های مورد نیاز مطرح شد. در واقع LTE که یک پروژه مهم 3GPP[[1]](#footnote-1) (انجمن جهانی استاندارد مخابرات سلولی) به­حساب می‎آید، برای اولین بار در فن­آوری مخابراتی نسل سوم تعریف شد. این فن­آوری در سال 2004 ارائه شد اما پس از دو سال در کاربردهای واقعی مورد استفاده قرار گرفت. لازم به ذکر است که این نمونه از شبکه، نسخه ششم این پروژه (3GPP) نام داشت. تجهیزات این نسخه مجهز به HSUPA[[2]](#footnote-2) (قابلیت دسترسی به بسته با سرعت بالا)، MBMS[[3]](#footnote-3) (اتصال مخصوص یک نقطه به چند نقطه)، بهبود سیستم مدیریت یکپارچه IMS[[4]](#footnote-4) و ساخت یک شبکه از نوع WLAN[[5]](#footnote-5) است. در سال 2007 نسخه جدیدتری با نام Rel 7 معرفی شد. این نسخه از 3GPP دارای ویژگی­هایی نظیر: بهبود تاخیر و QoS[[6]](#footnote-6) (هر تکنیکی که ترافیک داده را مدیریت می‎کند تا از دست دادن اطلاعات را به حداقل برساند)، VoIP[[7]](#footnote-7) (پروتکل اینترنت جهت ارسال صدا)، EHSPA[[8]](#footnote-8)، سلسله مراتب NFC[[9]](#footnote-9) و تکامل EDGE[[10]](#footnote-10) بود.

یک سال بعد، نسخه 8 که بعدها به نسخه LTE معروف شد، منتشر شد. این نسخه مجهز به OFDMA[[11]](#footnote-11)، به کار بردن فناوری­های چند ورودی و چند خروجی (MIMO) و [[12]](#footnote-12)HSDPA همراه با دو سلول بود. در سال 2009 بود که نسخه 9 با امکانات پشتیبانی از تماس­های صوتی، فعالیت در باندهای فرکانسی اضافی، WiMAX[[13]](#footnote-13)، شبکه LTE، قابلیت همکاری UMTS[[14]](#footnote-14) (پروتکل ارسال متن به­ صورت بسته)، فن‎آوری HSDPA با دو سلول و MIMO، HSUPA[[15]](#footnote-15) در دو سلول و LTE HeNB[[16]](#footnote-16) (مصارف خانگی شبکه) ظاهر شد. نسخه 10 پس از دوسال، با تجهیزاتی نظیر LTE-A و HSDPA چند سلولی و امکان سازگاری با نسخه 8 ارائه شد. امکاناتی نظیر شبکه­های ناهمگن[[17]](#footnote-17) (HetNet)، هماهنگی­های چند نقطه­ای[[18]](#footnote-18) (CoMP)، IDC [[19]](#footnote-19) و سرویس­های خدمات IP پیشرفته نیز در نسخه­ی 11 این پروتکل محقق شد. در سال 2015 بود که نسخه 12 با امکاناتی مانند: عملکرد سلول­های کوچک پیشرفته، تجمیع سیگنال حامل[[20]](#footnote-20) (دو سیگنال حامل در لینک فراسو و 3 حامل برای لینک فروسو، تجمع سیگنال [[21]](#footnote-21)FDD/TDD)، MIMO (مدل‎سازی سه بعدی کانال، شکل‎دهی پرتو و Massice MIMO) پشتیبانی از MTC-UE، ارتباطات D2D[[22]](#footnote-22) و پیشرفت‎های eMBMS در اختیار کاربران قرار گرفت. در نسخه 13، بهبودهایی در حوزه­ی تجمیع سیگنال حامل و قابلیت LAA[[23]](#footnote-23) برای LTE، قابلیت MUST[[24]](#footnote-24) در فرآیند لینک فروسو و بهبود D2D و dual conectivety صورت گرفت.

در سال 2017 امکاناتی مانند افزایش پایداری در سرویس VoLTE[[25]](#footnote-25)، پیشرفت در MBMS و بهبود eMBMS برای LTE در نسخه 14 ارائه شد. نسخه 15 پروتکل 3GPP شامل افزایش نرخ داده به حداکثر مقدار، افزایش پهنای باند، اعمال مرحله پردازش سیگنال و ایجاد امکانات URLLC[[26]](#footnote-26) و MTC[[27]](#footnote-27) است. نسخه 16 و نسخه قبل آن به­روزترین فن‎آوری­های نسل 5 به حساب می­آیند و آخرین پیشرفت‎هایی که پروتکل 3GPP مشخص کرده است را دارا می­باشند. در این نسخه پیشرفت‎های شگرفی در رابطه با تکنولوژی MIMO صورت گرفت. از طرف دیگر بهبودهایی در رابطه با شکل‎دهی پرتو و تجمیع حامل، Dual connectivity و نگهداری توان صورت گرفت. از طرف دیگر این نسخه مجهز به DSS[[28]](#footnote-28) می‎باشد. این نسخه در ماه مارس 2020 منتشر شد. در شکل 1-1 سیر تکامل نسخه­ی 10 به بعد 3GPP نشان داده شده است.

نسل­های متعدد (4G و 5G) از فن‎آوری­های شبکه از قبیل LTE، WiMAX، LTE-A، LTE with VoLTE، AXGP و WiMAX2 استفاده می­کنند. به‌طور اختصاصی مخابرات نسل 4 از انواع فناوری‎های LTE (LTE-A، TD-LTE، AXGP، TD-LTE-A و LTE) به جز WiMAX پشتیبانی می­کند. در این پژوهش جهت اشاره به مجموعه فن­آوری­های تجاری نسل 4 ، به اختصار از LTE استفاده می­شود.



شکل ‏1‑1: سیر تکامل نسخه‎های 3GPP [1]

فناوری موجود 4G LTE پل اصلی تکامل به سوی شبکه های 5G است. همانطور که در ادامه توضیح داده خواهد شد، نرم‌افزارسازی و مجازی‌سازی سخت‌افزار 4G LTE رویکرد اصلی است که شبکه‌های نسل بعدی را قادر می‌سازد تا تعداد بیشتری از دستگاه‌های مورد انتظار را در خود جای دهند که در واقع شامل IoT، حسگر، M2M می‌شود. دستگاه های ارتباطی ماشین به ماشین) و همچنین فناوری های تلفن همراه موجود و در حال ظهور. شبکه های نسل بعدی از طریق مفاهیم network slicing ، containerization ، تکرار سرویس، مجازی سازی عملکرد شبکه و ابری سازی، عملکرد گسترده و استحکام اتصال را در مدت زمان طولانی تری ارائه خواهند کرد. با این حال، برای درک پیاده‌سازی راه‌حل‌های منبع باز که زیربنای توسعه شبکه‌های 5G هستند، اصول اولیه فناوری غالب 4G LTE باید توضیح داده شود. اصولاً ضروری‌ترین عناصر برای درک، اجزای LTE هستند، مانند کانال‌های دسترسی که بهینه‌سازی دشوار آنها برای عملیاتی کردن و پایدار کردن شبکه تولید بسیار مهم است. درک معماری LTE ضروری است، و بنابراین، در فصل بعدی به طور مفصل به همراه تکنیک های دسترسی، فناوری های آنتن، الگوریتم های مسیریابی و ویژگی های امنیتی شبکه های نسل چهارم توضیح داده شده است.

## چگونگی ارتباط ایجاد شده بین نسل پنجم و LTE

همانطور که بیان شد اولین مجموعه کامل از مشخصات سیستم­‌های مبتنی بر نسل 5 مخابرات سلولی در نسخه 15 از استانداردهای پروتکل 3GPP معرفی شد. این نسخه طی دو گام پیاده­سازی شد:

1. Early drop: سرعت‎بخشیدن به دسترسی اولیه امکانات نسل پنجم. این مجموعه، پیکربندی شبکه را تنظیم می­کند که رابط هوایی جدید NR[[29]](#footnote-29) را به EPC از شبکه LTE از نسل چهارم متصل می­کند. در پیکربندی شبکه، LTE برای بخش کنترلی[[30]](#footnote-30) (اطلاعات سیگنالینگ و روشهای جابجایی مانند handover) و NR نیز برای بخش داده[[31]](#footnote-31) (انتقال اطلاعات) استفاده می­شوند. در ژوئن 2018 بود که نسخه اصلی 5G NR standalone (SA) مشخص شد. با SA عملیات اتصال هسته شبکه نسل پنجم (امکان تعریف کاربر و بخش کنترلی برای NR) با شبکه­‌های LTE بدون هیچ وابستگی اضافی صورت گرفت.
2. Last drop: در اواسط سال 2019 امکان همکاری هسته نسل پنجم با شبکه دسترسی رادیویی LTE-RAN[[32]](#footnote-32) صورت گرفت. این امر موجب شد سرویس‎ها بین NR و LTE و استفاده از dual connectivity با متراکم کردن[[33]](#footnote-33) حامل­های NR و LTE تداوم پیدا کند. در یکی از ویژگی­هایی که در نسخه 15 بیان شد، امکان اتصال NR با LTE فراهم شده است که طی آن باهم به­صورت بی‎سیم اتصال پیدا می­کنند.

# شبکه LTE

## مقدمه

در این فصل ابتدا معماری شبکه­های LTE را تشریح می‎کنیم و سپس به معرفی انواع لینک­ها و الگوریتم­های ارتباطی بین هر قسمت این شبکه می­پردازیم. همانطور که اشاره شد، در سال 2009 بود که با معرفی نسخه نهم از پروتکل 3GPP، شبکه LTE عملیاتی شد. این شبکه امکان انتقال خدمات پخش از طریق یک شبکه قابل‎حمل دستی را بدون نیاز به هیچ‎گونه مشخصات اضافی فراهم می‎کند.

هدف نهایی شبکه LTE افزایش ظرفیت سیستم، افزایش مساحت تحت پوشش شبکه، رشد قابل ملاحظه نرخ داده، حفظ حالت چندآنتنه بودن و به حداقل رساندن میزان تأخیر است. LTE هم از FDD و هم از TDD در طیف­های جداگانه پشتیبانی می‎کند و از طرف دیگر، اتصالات را برای شبکه های تلفن همراه هم توسط LTE فراهم می‎آورد [1].

این شبکه در طی فرآیند انتقال بسته­ها در لینک فروسو از OFDM و در حالت لینک فراسو از SC-FDMA[[34]](#footnote-34) استفاده می­کند. در OFDMA به منظور پشتیبانی از کیفیت خدمات، (به عنوان مثال، با کنترل نرخ داده­ هر کاربر) می­توان تعداد متفاوتی از زیر حامل­ها[[35]](#footnote-35) را به کاربران مختلف اختصاص داد.

فن­آوری­هایی که در بهره­گیری از نسل پنجم وجود دارند، نمونه تکامل‎یافته از فناوری­هایی هستند که در نسل قبل معرفی شدند. همان­طور که اشاره شد، در نسل چهارم از فن­آوری LTE به عنوان پلتفرم این نسل از مخابرات بی‎سیم استفاده می­شود. برخی از اجزا و ویژگی­های سامانه LTE که به­عنوان پلتفرم اصلی نسل­های تلفن همراه و شبکه­های مخابراتی شناخته می­شود، عبارتند از:

1. نرخ بیت تا حدود 100 مگابیت بر ثانیه در لینک فروسو
2. تاخیر کمتر از 10 میلی ثانیه
3. قابلیت پشتیبانی از پهنای باند متغیر
4. استفاده از OFDM (OFDMA و SC-FDMA)
5. استفاده از سیستم­های MIMO (چند آنتنه)
6. مدولاسیون­های مرتبه بالا تا 64QAM و AMC
7. HARQ[[36]](#footnote-36)
8. مدهای TDD و FDD
9. مبتنی بر IP بودن

دیگر ویژگی­ها و تجهیزاتی که برای LTE می­توان اشاره کرد، عبارتند از:

1. شبکه­های ناهمگن[[37]](#footnote-37): این شبکه شامل سلول­ها با اندازه­های مختلف از جمله مایکروسل تا فمتوسل می­باشد.
2. شبکه متشکل از سلول­های کوچک که جهت افزایش ظرفیت کاربران شبکه و بهره‎برداری بیشتر از فرکانس تخصیص داده شده مورد استفاده قرار می­گیرد.
3. ارتباط ماشین به ماشین (M2M)[[38]](#footnote-38).
4. استفاده از پهنای باند گسترده­تر در فرکانس­های بالا و تجمیع فرکانس سیگنال­های حامل
5. استفاده از فن‎آوری رادیو شناختگر و استفاده از طیف رادیویی به­صورت فرصت­طلبانه
6. روش­های دسترسی چندگانه غیرمتعامد (NOMA)[[39]](#footnote-39)
7. استفاده از امواج با فرکانس بالای 50 گیگا هرتز
8. استفاده از روش دوطرفه [[40]](#footnote-40)برای ارسال اطلاعات

همان­طور اشاره شد، برای بررسی­های صورت‎گرفته در رابطه با مخابرات بی‎سیم در نسل پنجم، از سامانه LTE استفاده می­شود. این سامانه که به ­عنوان فن‎آوری مورد استفاده در این نسل شناخته می‎شود، دارای ویژگی­های متعددی است. از جمله این ویژگی­ها می­توان به اقسام زیر اشاره کرد:

1. پیچیدگی پایین
2. مجهز به سیستم packet switched
3. کاهش شدید هزینه اجرا و نگهداری
4. ضرورت کارایی طیفی
5. افزایش سرعت سیستم
6. افزایش نرخ اطلاعات و QoS
7. مقاوم در برابر تحرکات زیاد کاربر

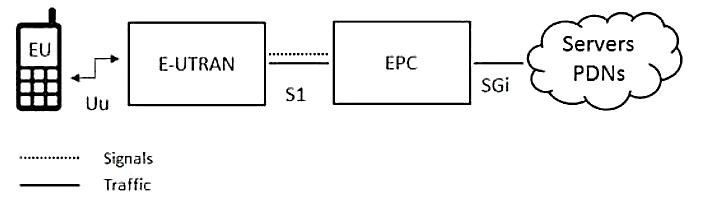
به دلیل وجود چنین ویژگی­هایی در سامانه LTE، این سامانه در سراسر جهان گسترش پیدا کرده است و به­همین دلیل تحقیقات مربوط به فن‎آوری­ها و معماری­های آن در جریان است.

## معماری شبکه LTE

شبکه LTE مبتنی بر سیستم EPS[[41]](#footnote-41) است. EPS شامل شبکه دسترسی رادیویی است که به شبکه­های هسته E-UTRAN و IP معروف است. سیستم سلولی قبلی، عمدتاً متمرکز بر شبکه circuit-switched است. از طرف دیگر، شبکه LTE نیز با متمرکز بودن روی شبکه­ packet-switched پیاده­سازی شد. هدف اصلی ایجاد شبکه LTE، ایجاد ارتباط IP بین شبکه data packet (PDN)[[42]](#footnote-42) و تجهیزات کاربر (UE)[[43]](#footnote-43) بدون ایجاد اختلال برای کاربران متحرک است. هدف اصلی شبکه LTE ، فراهم آوردن اتصالات پروتکل اینترنت (IP) بدون شکاف میان شبکه داده‎های بسته (PDN) و تجهیزات کاربر (UE) بدون ایجاد مزاحمت برای کاربران هنگام تحرک است.

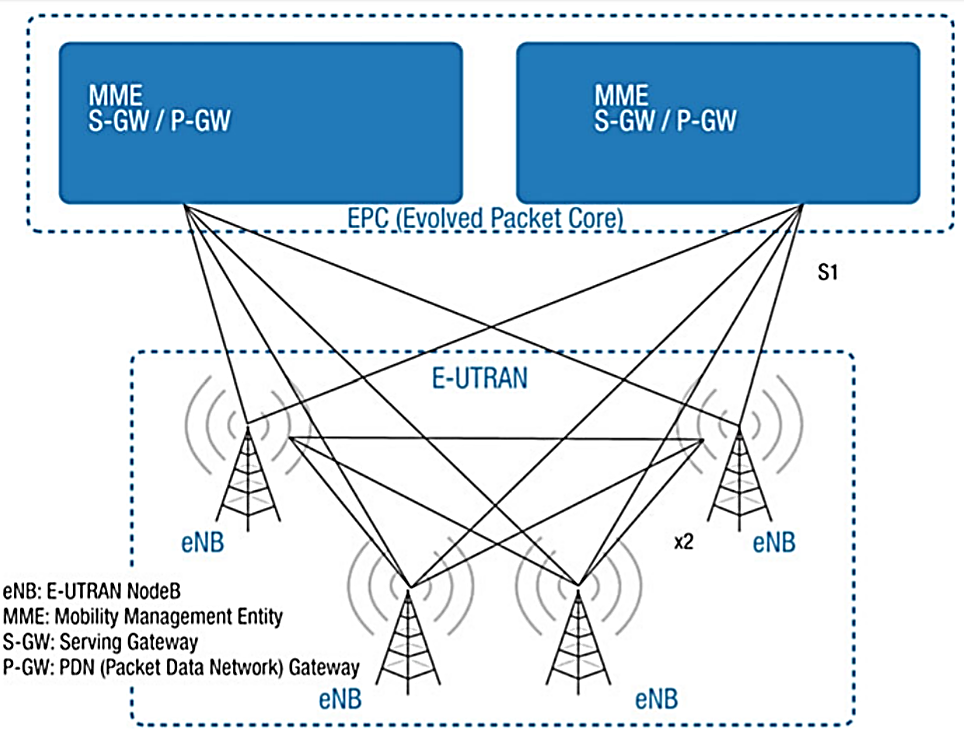
همان­طور که در شکل 2-1 دیده می­شود، به­طورکلی معماری شبکه LTE از سه مؤلفه اصلی زیر تشکیل می­شود:

1. تجهیزات کاربر (UE)
2. شبکه E-UTRAN
3. هسته­ی EPC



شکل ‏2‑1: معماری شبکه‎های LTE [2]

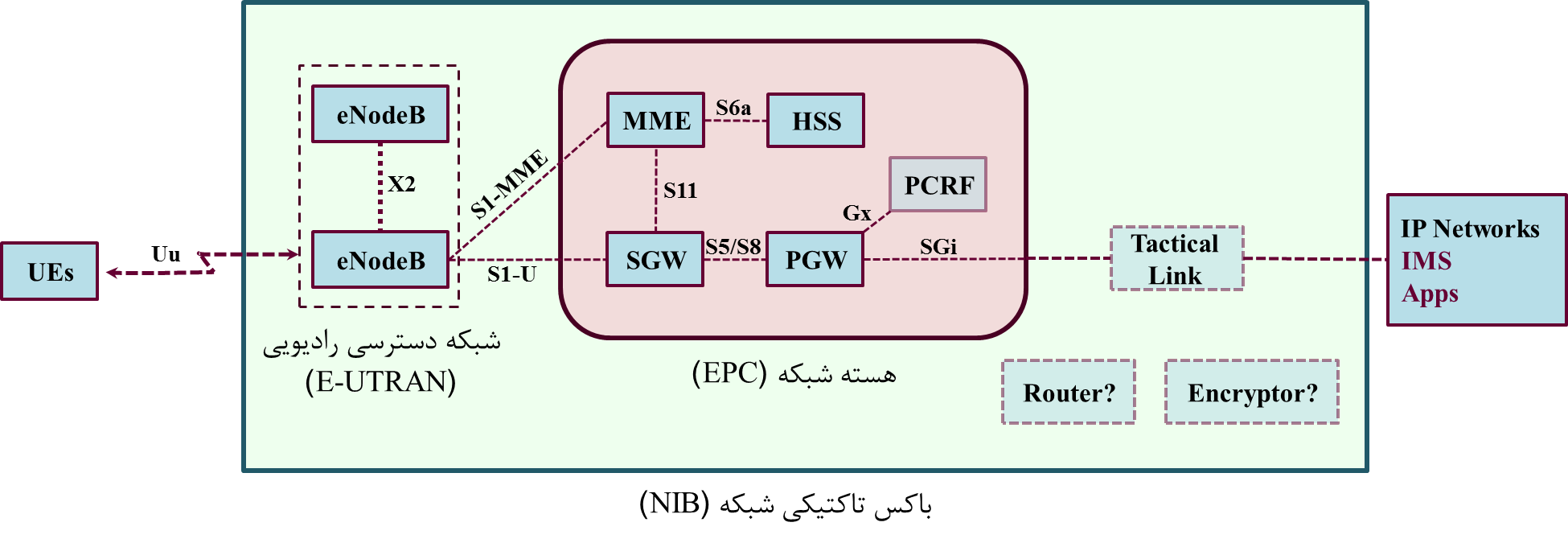
بخش UE اختصاص به تجهيزات متحرک[[44]](#footnote-44) دارد. اين تجهيرات از ماژول­هایی مانند MT[[45]](#footnote-45) که تمام عملکردهای ارتباطی را انجام می دهد، TE[[46]](#footnote-46) که جریان داده دریافتی را کنترل می­کند و در نهایت UICC[[47]](#footnote-47) که وظیفه­ی اجرای اپلیکیشنی که به­ عنوان هویت مشترک شناخته می­شود (USIM)[[48]](#footnote-48) را بر عهده دارد، تشكيل شده‎اند. مؤلفه E-UTRAN نیز به ­عنوان شبکه دسترسی رادیویی تکامل‎یافته زمینی در مقیاس جهانی است که وظیفه­ی آن دسترسی شبکه برای UE است.



شکل ‏2‑2: نمایی از معماری LTE [3]

همان­طور که در شکل 2-2 می­بینید، LTE دارای معماری مسطح است زیرا تنها از ایستگاه پایه گره تکامل یافته (eNodeB)[[49]](#footnote-49) تشکیل شده است، در حالی که هسته مرکزی در هسته­ی آن وجود ندارد. از این رو، این شبکه با UMTS متفاوت است. برای کاهش تأخیر تا 10 میلی ثانیه، تعداد eNodeB به حداقل مقدار خود می­رسد و با معرفی فن­آوری MIMO نرخ داده­های لینک‎های فراسو و فروسو به­ترتیب تا حدود 50 و 100 مگابیت بر ثانیه افزایش خواهد یافت. eNodeB یا گره تکامل یافته NodeB، در سیستم­های E-UTRAN نه تنها به­عنوان ایستگاه پایه (NodeB) موجب محدودکردن رابط رادیویی می‎شود، بلکه به­ عنوان کنترل‎کننده شبکه رادیویی (RNC)[[50]](#footnote-50) نیز مدیریت منابع را بر عهده دارد. این گره جزئی از E-UTRAN است که بر تعامل بین موبایل و EPC نظارت دارد. توجه شود که جایگزینی NodeB با نمونه تکامل‎یافته آن یعنی eNodeB در نسل سوم از نسل‎های مخابرات سیار صورت گرفت. در حال حاضر نمونه­ی eNodeB با کنترل‎کننده شبکه رادیویی ادغام شده است. eNB به عنوان واسط بین UE عمل می­کند که می­تواند همزمان به چندین سلول کمک کند. eNB به وسیله­ی رابط S1-U به SGW[[51]](#footnote-51) و همینطور توسط رابط S1-MME به MME[[52]](#footnote-52) متصل می­شود. از طرف دیگر جهت اتصال با eNodeB دیگر از رابط X2 که وظیفه­ی تحویل بسته­های اطلاعات را طی handover بر عهده دارد، استفاده می­شود.

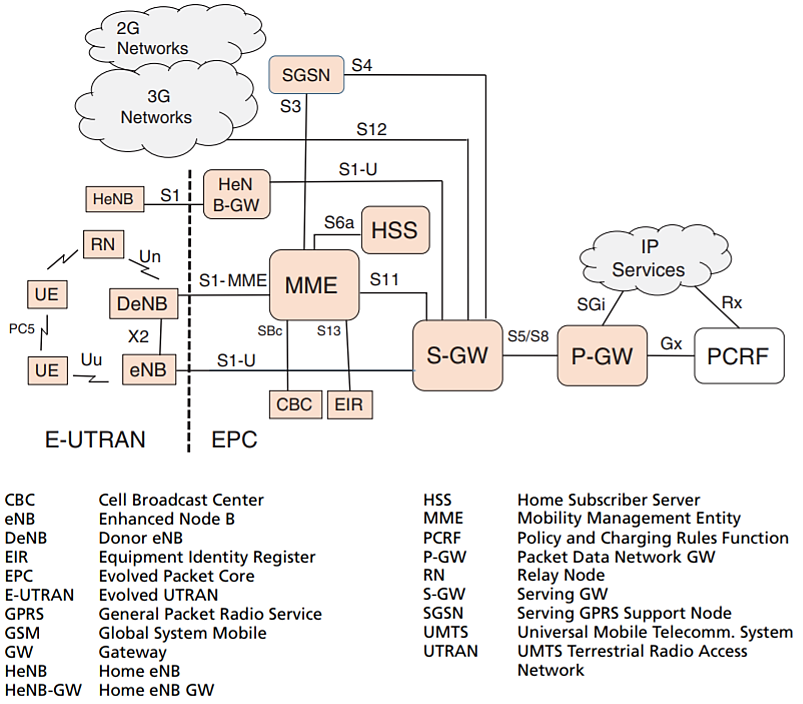
اما علاوه بر معماری مرسوم شبکه LTE که مشاهده کردیم، معماری این شبکه در حالت تاکتیکی نیز مورد توجه می‎باشد که در شکل ‏2‑3 می‎توان آن را به صورت کامل مشاهده کرد.



شکل ‏2‑3: معماری تاکتیکی شبکه LTE [4]

همانطور که مشاهده می‎شود فرق اساسی این معماری استفاده آن است که از لینک تاکتیکی برای اتصال به EPC و یا اتصال از EPC به دنیای اینترنت در آن استفاده می‎شود. همچنین استفاده از تکنیک‎های رمزنگاری و. روتر نیز از جمله قابلیت‎هاییست که می‎تواند در لینک‎های تاکتیکی استفاده شود. البته ممکن است که مؤلفه‎هایی همانند روتر و لینک تاکتیکی در سناریوهای تاکتیکی با eNodeB تجمیع شوند. تفاوت دیگر در سناریوهای تاکتیکی آن است که eNodeBها در این حالت باید بتوانند بدون شبکه back-end کار کنند بدین معنی که بسیاری از فعالیت‎های EPC را نیز باید انجام دهند.

## معرفی اجزا و الگوریتم­های ارتباطی شبکه LTE



شکل ‏2‑4: نمونه‎ای از اجزا و ارتباطات در شبکه LTE [5]

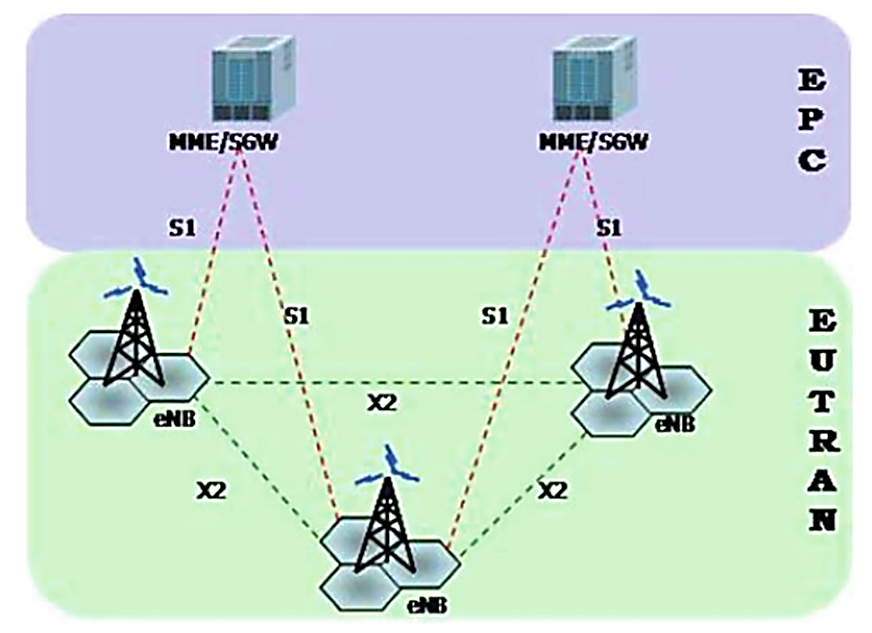
در شکل 2-3 کلیه اجزا و رابط­های شبکه LTE نشان داده شده است و می‎توان با توجه به این شکل رابط­ها و اجزای این شبکه را تشریح کرد.

### معرفی اجزای شبکه LTE

همانطور که در شکل 2-3 و شکل ‏2‑5 دیده می­شود، در بخش E-UTRAN اجزایی نظیر DeNB، eNB، UE و HeNB وجود دارند که در ادامه به تشریح آن­ها می­پردازیم:

DeNB[[53]](#footnote-53) یک eNB پیشرفته است که علاوه بر پشتیبانی از UE خود، می­تواند از RNها نیز پشتیبانی کند. رفتار DeNB نسبت به RN مانندMME ، eNB و SGW است. HeNB نیز یک eNB بهینه برای استفاده در سلول­های کوچکتر است که می­تواند در خانه­ها یا اماکن عمومی به­کار برده شود. بسته به میزان قدرت، ظرفیت و پوشش آن، این سلول‎های کوچک به نام‎های فمتوسل، پیکوسل یا میکروسل شناخته می­شوند.

بخش EPC مسئولیت کنترل کلی شبکه و اجزای آن را برای تسهیل ارسال اطلاعات و خدمات صوتی به کاربران خود بر عهده دارد. EPC از واحدهای زیر تشکیل شده است:



شکل ‏2‑5: نمایی از معماری E-UTRAN ]3[

SGW جهت ایجاد ارتباط بین E-UTRAN و PGW[[54]](#footnote-54) و همچنین هدایت و تحویل بسته‌های اطلاعاتی از UE و PDN مورد استفاده قرار می‎گیرد. این بخش جهت مدیریت تحرک بخش داده و همچنین حفظ مسیر اطلاعاتبین eNBها و PGW با ارسال و مسیریابی بسته­های اطلاعاتی کاربر مورد استفاده قرار می­گیرد.

PGW ارتباط بین UE و شبکه­های (PDN) خارجی را برقرار می‎کند و عملیات ورود و خروج اطلاعات را برای داده­های UE صورت می­دهد. عملیات اختصاص آدرس IP به UE نیز توسط PGW صورت می­گیرد. اجرای سیاست‎گذاری (بر اساس قوانین موجود در PCRF[[55]](#footnote-55)) در بررسی کامل و فیلترکردن بسته­ها توسط این بخش صورت می­گیرد. کیفیت خدمات (QoS) هم توسط PGW مدیریت می­شود.

MME به­عنوان گره کنترل اصلی برای شبکه دسترسی LTE SAE، وظیفه­های متعددی دارد که از جمله آن می‎توان به موارد زیر اشاره کرد: ردیابی UE که در حالت ارسال و دریافت داده به سر نمی‎برند، انتخاب SGW برای یک UE، تعامل با HSS[[56]](#footnote-56) برای تایید اعتبار کاربر در پیوست[[57]](#footnote-57) و پیاده‎سازی محدودیت‌های رومینگ. به­طور کلی این بخش وظیفه­ی مدیریت اتصال و فعالیت­های مربوط به مدیریت حامل[[58]](#footnote-58) را بر عهده دارد. MME ممکن است با یک مرکز پخش سلول (CBC)[[59]](#footnote-59)، (یک سرور توزیع اطلاعات که برای پخش پیام به کلیه کاربران متصل به eNBها مورد استفاده قرار می‎گیرد) در ارتباط باشد.

HSS نیز یک بانک اطلاعاتی مرکزی است که شامل داده­های به اشتراک‎گذاشته‎شده­ کاربران می‎باشد. همچنین HSS وظایفی مانند مدیریت تحرک، تأیید اعتبار و دسترسی کاربر، مجوز خدمات، مشخصات سرویس و امنیت کاربر را بر عهده دارد. به علاوه HSS داده­های مربوط به PDN‎هایی که UE ممکن است در آن عضو شود را حفظ می­کند.

[[60]](#footnote-60)HeNB-GW برای جمع­آوری و تمرکز ترافیک گروه بزرگی از سلول­های کوچک (HeNBs) در S-GW و ایجاد ارتباطات یکپارچه برای کاربران هنگام حرکت در داخل و خارج از سلول‎های HeNB و eNB استفاده می شود. در این حالت امنیت یک مسئله­ی اصلی است زیرا ممکن است برخی از این سلول‎ها جزئی از یک شبکه قابل اعتماد نباشند.

### رابط­های شبکه LTE

در این بخش به بررسی رابط­هایی که در شبکه LTE بین اجزای تعریف شده در بخش قبلی وجود دارند، می­پردازیم. رابط­های شبکه LTE به چهار بخش تقسیم می­شوند که در ادامه هر بخش را تشریح می­کنیم.

#### ارتباطات هوایی

رابط بین UE و eNB در اصطلاح LTE Uu نامیده می‎شود. این رابط که به ­صورت بی­سیم است، بیت­ها را از طریق سیگنال­های الکترونیکی کد می‎کند و سپس در لایه فیزیکی ارسال می‎شود. Un، رابط بین RN و DeNB است. از طرف دیگر PC5 یک رابط هوایی نسبتاً جدید بین دو UE است.

در نسخه 12 که توسط 3GPP معرفی شد، برای ارائه خدمات مبتنی بر مجاورت[[61]](#footnote-61) (ProSe) از این رابط استفاده شد. به­دلیل وجود این سرویس، UEها می­توانند مستقیماً و بدون کمک eNB ارتباط برقرار کنند. این حالت در شرایطی خاص مانند بلایای طبیعی یا هجوم هکرها و اختلالگرها به شبکه ممکن است مفید واقع شود.

#### رابط E-UTRAN

این رابط­ها ممکن است تحت دو دسته مورد بحث قرار گیرند. یک دسته شامل رابط­هایی است که برای اتصال دستگاه­های انتهایی استفاده می­شود. دسته دیگر شامل رابط­هایی است که برای اتصال eNBها به اجزای EPC به کار می‎روند. رابط X2 نیز برای اتصال‎های انتهایی استفاده می‎شود.

پروتکل stack در این رابط همانند رابط S-MME رفتار می­کند، با این تفاوت که پروتکل لایه application در X2 به­صورت X2-AP است. به­ طورکلی، رابط S1 برای اتصال eNBها به اجزای مختلف شبکه هسته استفاده می­شود و بسته به مؤلفه‎ای که eNB به آن متصل است، S1 یک شناسه اضافی دریافت می‎کند و در ادامه S1 قرار می­گیرد. برای مثال، S1-MME عملیات ارائه پروتکل­های سیگنالینگ بین eNodeB و MME را صورت می‌دهد. این رابط شامل یک پروتکل کنترل جریان انتقال[[62]](#footnote-62) (SCTP) از طریق IP است و چندین UE را از طریق یک واحد SCTP پشتیبانی می­کند. همچنین تضمین اطلاعات تحویلی را نیز فراهم می­کند.

رابط S1-U نیز به ­عنوان رابط خارجی بخش داده بین eNodeB و SGW قلمداد می‎شود.

#### رابط EPC

همانطور که در ادامه مورد بررسی قرار می­گیرد، رابط­های این گروه برای تبادل سیگنال و اطلاعات کاربر در بین اجزای یک شبکه اصلی استفاده می­شوند:

S5 برای اتصال SGW و PGW برای ایجاد دستورالعمل­های مدیریت tunnel استفاده می شود. آن‎ها در اصل همان رابط‎ها هستند با این تفاوت که S5 در شبکه LTE داخلی است ، S8 یک واسط کار اینترنتی است که هنگام رومینگ در بین شبکه­های مختلف مورد استفاده قرار می گیرد. به عبارت دیگر، S8 ارتباطاتی بین S-GW در شبکه بازدید شده و PGW در شبکه میزبان است.

S6a رابط بین MME و HSS است و برای انتقال داده‎های مشترک و تأیید هویت استفاده می‌شود. S9 نیز بین PCRF شبکه بازدید شده و PCRF شبکه مبدا جهت انتقال اطلاعات هویتی و اشتراکی مورد استفاده قرار می­گیرد تا هویت کاربر بررسی شود. رابط S10 بین MMEها قرار می‎گیرد و بیشتر در هنگام انتقال برای تبادل اطلاعات سیگنالینگ در بین آن‎ها استفاده می­شود.

 S11 برای اتصال MME به SGW به­ کار برده می­شود (یک MME می­تواند چندین SGW را پشتیبانی کند). این رابط برای مدیریت استقرار کانال‎های حامل در شبکه اصلی استفاده می‌شود. S13 رابطی است که MME را به EIR متصل می­کند تا اطلاعاتی را که در بررسی هویت UE استفاده می شود، تبادل کند. پروتکلی به نام Diameter برای این منظور در لایه application وجود دارد.

 Gx واسط بین PGW و PCRF است که برای انتقال قوانین charging از PCRF به PGW استفاده می‎شود. SBc برای ارسال هشدار تحویل پیام و توابع کنترلی بین CBC و MME مورد استفاده قرار می­گیرد. از طرف دیگر SBc-AP در این رابط از SCTP برای انتقال پیام­های خود استفاده می­کند.

#### رابط شبکه­های غیر LTE

رابط­های این گروه برای تسهیل ارتباطات بین اجزای EPC و مؤلفه­های متعلق به شبکه­های خارجی مورد استفاده قرار می­گیرند. رابط S3 برای ایجاد تعامل با شبکه های مبتنی بر GPRS نسل دوم و GSM نسل سوم استفاده می­شود. از طریق این رابط بین MME و SGSN، یک کاربر LTE می­تواند با یک کاربر غیر LTE در یک شبکه سازگار با پروتکل 3GPP، انتقال اطلاعات را صورت داد.

S4 مشابه S3 است، به جز اینکه بین SGSN و SGW قرار می­گیرد و تبادل اطلاعات سیگنالینگ و کنترلی را برای یک کاربر LTE و یک کاربر غیر LTE فراهم می کند. SGi برای اتصال PGW به یک شبکه داده استفاده می­شود که می‎تواند داخل یا خارج از شبکه LTE باشد. Rx نیز برای اتصال PCRF به یک شبکه داده استفاده می‎شود. S12 نیز بین P-GW و یک شبکه نسل سوم برای انتقال ترافیک کاربر هنگام ایجاد plane tunnel برای کاربر مستقیم استفاده می­شود. لازم به ذکر است که پیکربندی این رابط بستگی به اپراتور دارد.

## طراحی بلوک دیاگرام سامانه

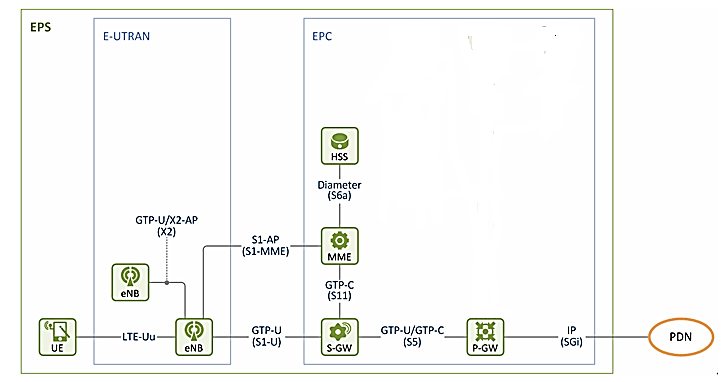
در این قسمت سعی بر آن است که بلوک دیاگرام سامانه LTE به طور کامل با جزئیات نمایش و مورد بررسی قرار بگیرد. این ساختار شامل واحدها و لایه‎هایی می‎باشد که هر کدام وظایف خاصی دارند که در این بخش ارائه خواهند شد.

### بلوک دیاگرام LTE همراه با جزئیات

همانطور که ذکر شد در این فصل از گزارش، ساختار شبکه LTE شرح داده می‎شود. شبکه LTE که EPS نیز نامیده می‎شود، یک شبکه تمام IP می‎باشد و به دو بخش تقسیم‎بندی می‎شود. قسمت LTE که با فن‎آوری‎های شبکه دسترسی رادیویی (E-UTRAN) در ارتباط است و قسمت EPC که با فن‎آوری‎های مربوط به هسته شبکه در ارتباط است. به علاوه منظور از تمام IP بودن نیز آن است که تمامی داده‎ها در EPS مبتنی بر پروتکل IP ارسال می‎شوند.

لازم به ذکر است که به منظور فراهم‎شدن خدمات LTE برای یک کاربر در شبکه LTE، یک مدل مرجع شبکه LTE (NRM)[[63]](#footnote-63) شامل سه حوزه مختلف می‎باشد که عبارتند از: حوزه BSS/OSS[[64]](#footnote-64) برای زیرحامل‎ها، حوزه Application برای ارائه خدمات و حوزه شبکه انتقال IP جهت ارسال بسته‎های IP.

در شکل ‏2‑6 می‎توان مدل مرجع شبکه LTE را مشاهده کرد. این ساختار شامل واحدهای LTE (UE و eNB) و واحدهای EPC (SGW، PGW، MME و HSS) می‎باشد. PDN نیز یک قسمت از حوزه IP بیرونی یا داخلی اپراتور می‎باشد که UE قصد برقراری ارتباط با آن را دارد و خدماتی نظیر اینترنت و IMS را برای UE فراهم می‎کند. در ادامه کاربردها و جزئیات هر یک از واحدهای LTE و EPC ارائه می‎شود.

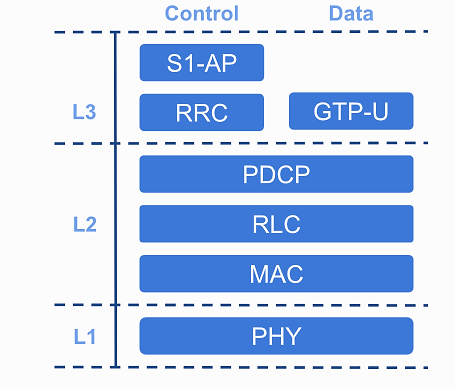


شکل ‏2‑6: مدل مرجع شبکه LTE [6]

### لایه‎های eNodeB

این قسمت از EUTRAN به واسطه رابط رادیویی با UE ارتباط برقرار می‎کند و وظیفه مدیریت منابع رادیویی از جمله اختصاص منابع پویا، کنترل پذیرش رادیویی[[65]](#footnote-65)، کنترل تحرک ارتباطات، کنترل حامل رادیویی[[66]](#footnote-66) و ICIC[[67]](#footnote-67) را بر عهده دارد.

کاربردهای eNodeB شامل سه لایه 1، 2 و 3 می‎شود (شکل ‏2‑7). در قسمت پایین شکل، لایه فیزیکی همه اطلاعات را از لایه MAC به واسطه رابط هوایی حمل می‎کند. در واقع این قسمت مسئول سازگاری لینک و کنترل توان می‎باشد. لایه MAC نیز داده میان یک یا چند کانال منطقی را بین بلوک‎های انتقالی که از لایه فیزیکی عبور می‎کنند، تقسیم می‎کند. به عبارتی لایه MAC وظیفه برنامه‎ریزی ارسال‎های فراسو و فروسو UEها را به وسیله سیگنالینگ کنترلی، ارسال مجدد و تصحیح خطا (HARQ) و مدیریت حق تقدم میان کانال‎های منطقی انجام می‎دهد.



شکل ‏2‑7: ساختار پایه eNodeB [7]

لایه RLC نیز می‎تواند در سه حالت زیر فعالیت کند که عبارتند از: [[68]](#footnote-68)TM، UM[[69]](#footnote-69) و [[70]](#footnote-70)AM. این لایه چندین کانال منطقی را برای هر UE مدیریت می‎کند. در حالت TM، داده از طریق RLC عبور داده می‎شود؛ در حالت UM عملیات الحاق[[71]](#footnote-71) و بخش‎بندی[[72]](#footnote-72) واحدهای داده انجام می‎شود. حالت AM نیز وظیفه ارسال مجدد داده‎هایی که ارسال ناموفق داشته‎اند را بر عهده دارد.

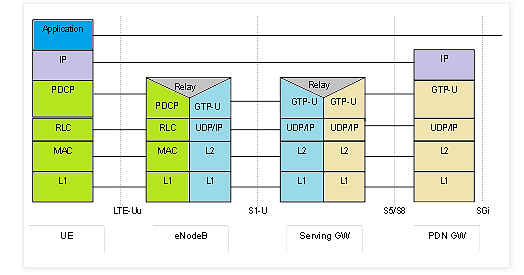
لایه PDCP نیز وظیفه رمزگذاری بخش داده و کنترلی، حفاظت صحت ترافیک بخش کنترلی و به ترتیب تحویل متوالی ترافیک بخش داده و کنترلی به لایه‎های RRC و GTP-U را بر عهده دارد. ضمنا PDCP در صورت امکان می‎تواند فشرده‎سازی قسمت ابتدایی[[73]](#footnote-73) داده IP را انجام دهد.

لایه RRC نیز تبادلات بخش کنترلی میان eNodeB و UE را مدیریت می‎کند. در واقع این لایه بلوک‎های SIB[[74]](#footnote-74)را که توسط eNodeB ارسال می‎شود، تولید می‎کند و همچنین برقراری، حفظ و انقطاع ارتباط RRC با UEها را کنترل می‎کند.

بالای لایه RRC، لایه S1AP[[75]](#footnote-75) وجود دارد که ارتباطات کنترلی میان eNodeB و EPC را مدیریت می‎کند. در واقع S1AP به MME در شبکه هسته متصل می‎شود. به علاوه پیام‎هایی که از MME به UEها ارسال می‎شوند به وسیله S1AP به لایه RRC ارسال می‎گردد. توجه شود که پیام‎ها از UE به MME نیز قبل از عبور از S1AP و رسیدن به MME در eNodeB RRCاستخراج می‎شوند.

لایه GTP-U[[76]](#footnote-76) نیز ارتباط بخش داده میان eNodeB و EPC را فراهم می‎کند. در واقع این لایه به S-GW در شبکه هسته متصل می‎شود. ترافیک IP بخش داده بعد از تونل‎شدن[[77]](#footnote-77) به EPC در P-GW استخراج و سپس به اینترنت ارسال می‎شوند.

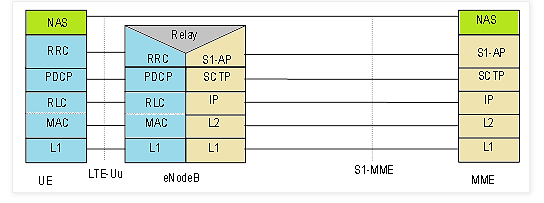
### لایه‎های بخش داده LTE



شکل ‏2‑8: معماری پروتکل بخش کاربر (داده) LTE[8]

بخش کاربر برای پشتیبانی و تحقق گذردهی بالا طراحی شده است. همانطور که در شکل ‏2‑8 نشان داده شده است از لایه PDCP کاربر (UE) به سمت پایین، تمامی لایه‎ها به صورت متناظر به لایه‎های eNodeB نگاشت پیدا می‎کنند. توجه شود که این لایه‎ها حاوی ارسال بسته داده و همچنین امنیت آن‎ها در لینک رادیویی می‎باشند. بین UE و eNodeB نیز رابط هوایی LTE-Uu وجود دارد. به علاوه میان ENodeB و SGW و همچنین SGW و PGW، GTP-U و لایه انتقال (UDP) به عنوان پروتکل اصلی استفاده می‎شوند. لایه application در شکل نیز به لایه application واحد PDN متصل می‎باشد.

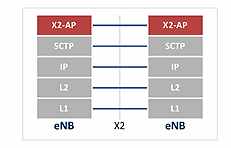
### لایه‎های بخش کنترلی LTE



شکل ‏2‑9: معماری پروتکل بخش کنترلی LTE [8]

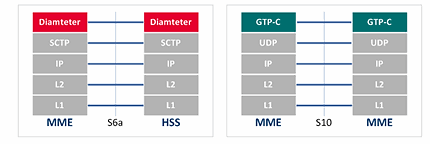
در شکل ‏2‑9 معماری بخش کنترلی LTE را مشاهده می‎کنید. این بخش جهت مدیریت تحرک و ایستایی کاربران در LTE معرفی شده است. در واقع پروتکل‎های این بخش مسئولیت ارسال انواع سیگنال‎هایی که در رومینگ، برقراری و اتمام ارسال و رهگیری کاربر استفاده می‎شوند را داراست. لازم به ذکر است که بالاترین لایه در UE، لایه NAS می‎باشد. توجه شود که NAS فقط با MME NAS و به وسیله eNodeB ارتباط برقرار می‎کند؛ به طوری که eNodeB تنها اطلاعات NAS را عبور می‎دهد و توجهی به آن‎ها ندارد. این واحد وظیفه مدیریت تحرک و فعالیت‎های مربوط به مدیریت حامل را بر عهده دارد.

لازم به ذکر است که از لایه RRC به سمت پایین، همه لایه‎ها عینا به لایه‎های eNodeB نگاشت می‎شوند. مسئولیت این لایه‎ها نیز برقراری، اصلاح و قطع رابط هوایی میان UE و eNodeB و همچنین حفظ امنیت رابط هوایی می‎باشد. توجه شود که میان eNodeB و MME رابط هوایی S1-MME وجود دارد که روی پروتکل لایه application (S1AP) کار می‎کند.



شکل ‏2‑10: ارتباط بخش کنترلی میان دو eNB [9]

در شکل ‏2‑10 می‎توان ارتباط بخش کنترلی میان دو eNodeB را مشاهده کرد. همینطور که مشخص است برای برقراری این ارتباط از پروتکل X2-AP استفاده شده است. این پروتکل تحرکات UE و فعالیت‎های مربوط به SON را در EUTRAN پشتیبانی می‎کند. به منظور پشتیبانی از تحرکات UE، پروتکل X2-AP اعمالی از قبیل ارسال داده، انتقال وضعیت SN[[78]](#footnote-78) و محتوای UE را انجام می‎دهد. در فعالیت‎های مربوط به SON[[79]](#footnote-79) نیز، eNodeBها اطلاعات مربوط به وضعیت منابع، اطلاعات میزان ترافیک و همچنین اطلاعات مربوط به بروزرسانی پیکربندی eNodeB را تبادل می‎کنند و بدین ترتیب پارامترهای تحرک را به وسیله پروتکل X2-AP تنظیم می‎کنند.



شکل ‏2‑11: ارتباط بخش کنترلی دو MME و همچنین MME و HSS [9]

در شکل ‏2‑11 نیز مشخص است که ارتباط بخش کنترلی میان دو MME به وسیله پروتکل GTP-C اتفاق می‎افتد که وظیفه پشتیبانی تبادل اطلاعات کنترلی برای ایجاد، اصلاح و اتمام تونل‎های GTP را بر عهده دارد. در واقع این پروتکل، تونل‎های ارسال داده را در حالت handover در شبکه LTE ایجاد می‎کند. همچنین پروتکل Diameter، وظیفه تبادل اطلاعات مربوط به تأیید و شناسایی زیرحامل‎ها میان HSS و MME را بر عهده دارد.

## جمع‎بندی و نتیجه‎گیری

در ای فصل تمامی جزییات سامانه، بلوک دیاگرام، معماری و زیرسیستم­های سامانه مورد بررسی و اشاره قرار گرفت. در این فصل تمامی ارتباط­هایی که در سامانه وجود دارد به همراه رابط‎های موجود بین زیرسیستم­ها تشریح شدند. سپس بلوک‎دیاگرام سامانه LTE را طرح کردیم. در واقع معماری این سیستم ارائه شد و همچنین لایه‎ها و پروتکل‎های واحدهای اصلی بخش داده و کنترلی LTE از جمله eNodeB، HSS، PGW، SGW و MME مورد بررسی قرار گرفت.

# معماری شبکه های نسل پنجم

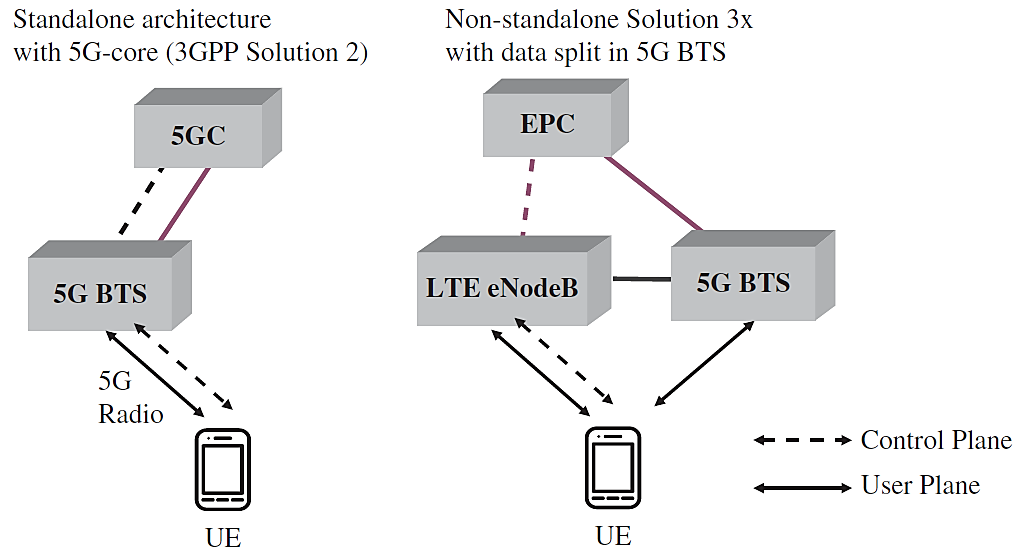
## معرفی

معماری شبکه 5G تحت تأثیر عوامل مختلفی قرار گرفت. توانایی پیاده سازی های مبتنی بر فضای ابری، آمادگی برای مقابله با نرخ داده های بزرگتر و تأخیرهای کمتر نسبت به نسل های قبلی، امکان سرویس های جدید و همچنین نیاز به همکاری با Long-Term Evolution (LTE)، به ویژه در اولین فاز. همه این عوامل بر معماری 5G تأثیر گذاشته است.علاوه بر توسعه فن‌آوری رادیو , یک شبکه هسته‌ 5G جدید نیز تعریف شده‌است که به عناصر خدمات جدید از نظر خدمات محلی و جهانی و همچنین مفهوم جدید کیفیت خدمات مبتنی بر جریان و پشتیبانی از network slicing و بسیاری از ویژگی‌های دیگر اجازه می‌دهد که اجرای کارآمدتری را در مقایسه با هسته LTE (EPC[[80]](#footnote-80)) انجام دهد. این فصل مروری بر معماری‌های مختلف 5G ارائه می‌کند و سپس با جزئیات بیشتر معماری و رابط‌های شبکه دسترسی رادیویی (RAN) و همچنین شبکه 5G Core (5GC) را با عناصر و عملکردهای کلیدی پوشش می‌دهد.

## گزینه های معماری 5G

در ابتدا هشت گزینه معماری در مباحث 3GPP ]11[ وجود داشت که تمام ترکیب‌های ممکن و ترکیبات شبکه اصلی در نظر گرفته شده بود، اما با پیشرفت کار، دو گزینه اصلی معماری نهایی شدند. اولین رویکرد در طرح نهایی، رویکرد همراه با LTE به عنوان لنگر اتصال و استفاده از هسته LTE موجود، با رادیو 5G به عنوان سلول ثانویه بود که در اصطلاح بعنوان معماری گزینه سوم در مباحث 3GPP شناخته می شود. رویکرد بعدی که تکمیل شد، رادیو 5G مستقل با هسته جدید 5G بود که در اطلاح بعنوان معماری گزینه دوم در مباحث 3GPP شناخته می شود که در شکل 3-1 نشان داده شده است. گزینه های نشان داده شده در شکل 3-1 را می توان به صورت زیر توصیف کرد (به ترتیب نهایی شدن در 3GPP):

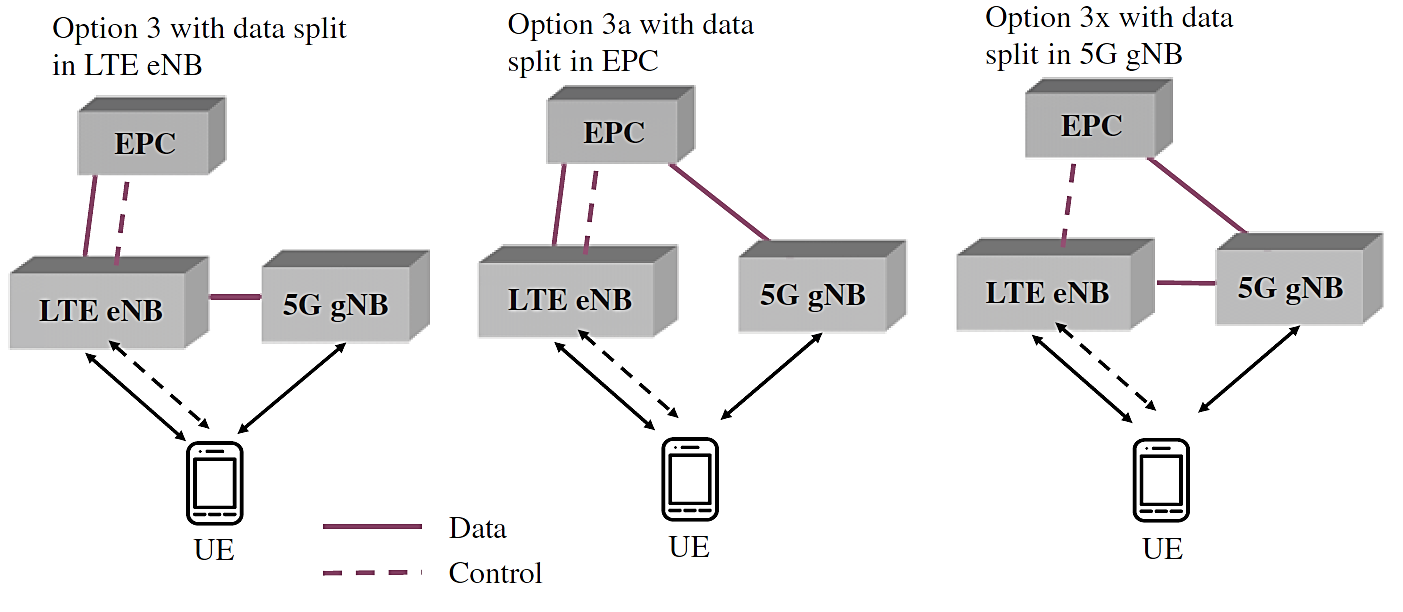
* گزینه 3 به گزینه معماری اشاره دارد که از LTE به عنوان لنگری برای اتصال و همچنین اتصال با EPC موجود استفاده می کند. در این معماری5G فقط روی user plane در قسمت رادیویی است که سپس در اتصال دوگانه با LTE استفاده می شود. برای مسیریابی داده های UP[[81]](#footnote-81)، انتخاب های بیشتری وجود دارد که انواع مختلفی از معماری گزینه 3 ایجاد می کند که در ادامه توضیح داده خواهد شد. از نقطه نظر خدمات، تنها پهنای باند موبایل[[82]](#footnote-82) (MBB) پیشرفته در دسترس است.
* گزینه 2 به معماری با رادیو 5G متصل به شبکه هسته 5G اشاره دارد. این معماری همه عملکردهای جدید ارائه شده توسط هسته 5G، مانند معماری مبتنی بر سرویس[[83]](#footnote-83) (SBA) یا network slicing را همانطور که در ادامه این فصل توضیح داده شده است فعال می کند. 3GPP 15 ارائه شده شامل MBB پیشرفته و همچنین ارتباطات با تأخیر کم بسیار قابل اعتماد (URLLC) است، همچنین خدمات اساسی مانند صدا از طریق 5G را نیز پوشش می دهد. در این معماری نیازی به کار کردن با اتصال دوگانه LTE-5G نیست، و بنابراین برای راه اندازی مشابه LTE است (اگرچه اتصال دوگانه 5G-5G در مرحله بعدی فعال شد). گزینه 2 کار متقابل با LTE (handovers/انتخاب مجدد به/از LTE) را که بعداً در این فصل توضیح داده شده است را امکان پذیر می کند.



شکل 3-1: گزینه های اصلی معماری برای عرضه اولیه 5G ]10[.

بسته به چگونگی تقسیم user plane بین LTE و 5G، مسیریابی داده می تواند به روش های مختلفی انجام شود که در شکل 3-2 نشان داده شده است. انتخاب های مختلف معماری گزینه 3 به شرح زیر است:

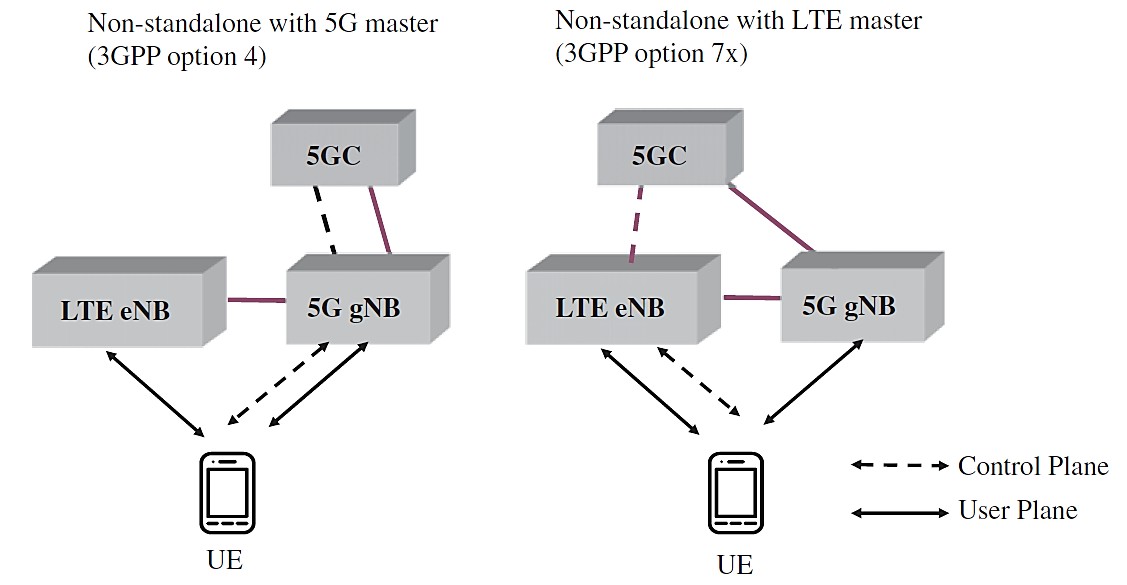
* گزینه 3 با تقسیم داده در LTE eNB . داده‌های user plane دریافت‌شده از EPC توسط LTE eNB بین رادیو LTE و 5G GNB (داده‌های رابط X2) تقسیم می‌شود. این اجازه می دهد تا یک سرویس واحد (با تقسیم حامل) از هر دو gNB و eNB منتقل شود و به ترتیب از هر دو طرف در uplink دریافت شود.
* گزینه 3A با تقسیم داده ها در EPC . این معماری امکان تجمیع نرخ داده را برای یک سرویس واحد فراهم نمی کند، اما یک سرویس واحد توسط یک LTE eNB یا یک 5G gNB سرویس می شود. در این مورد، رابط X2 برای انتقال user-plane استفاده نمی شود. از این معماری برای برخی از سرویس‌های با نرخ داده پایین مانند VoLTE، می‌تواند استفاده کرد در حالی که پهنای باند تلفن همراه از یکی از معماری های دیگر استفاده می‌کند که امکان تجمیع نرخ داده را فراهم می‌کند.
* گزینه 3X با تقسیم داده در 5G gNB . داده های دریافتی از EPC توسط 5G gNB بین رادیو 5G و LTE eNB (داده های رابط X2) تقسیم می شود. این معماری معمولاً مورد توجه است زیرا سرمایه‌گذاری در سمت LTE را به حداقل می‌رساند زیرا معمولاً LTE eNB اساسا برای رسیدگی به چنین نرخ‌های داده بالایی که شامل بخش انتقال backhaul می باشند طراحی نشده است.همچنین، اگر بیشتر بسته ها در هر صورت از طریق 5G بروند، تأخیر کاهش می یابد (شکل 5-2).



شکل 3-2: تفاوت گزینه های تقسیم داده برای خانواده گزینه 3 ]10[.

گزینه های معماری دیگری نیز وجود دارد که در شکل 3-3 نشان داده شده است و در ژوئن 2019 به عنوان بخشی از "late drop" در نسخه 15 3GPP نهایی شدند و گزینه های اتصال دوگانه اضافی را به شرح زیر بررسی می کنند.

* گزینه 4، با 5G gNB بعنوان لنگر اتصال. اتصال LTE ثانویه در اتصال با صفحه کنترل است که از طریق 5G gNB به سمت شبکه اصلی 5G می رود.این برای مواردی مناسب است که پوشش کافی برای اتصال 5G وجود داشته باشد، یا ایجاد یک شبکه متراکم کافی سپس استفاده از یک باند فرکانس پایین برای یک حامل 5G (حداقل برای یکی از حامل های 5G). این گزینه از نیاز به اتصال LTE eNB برای شبکه اصلی 5G جلوگیری می کند.
* گزینه 7، با LTE eNB که بعنوان لنگر اتصال است. اکنون اتصال 5G ثانویه در اتصال با user-plane است که از طریق LTE eNB به سمت شبکه اصلی 5G می رود. در این معماری فرض بر این است که LTE eNB به‌روزرسانی شده است تا امکان اتصال به شبکه هسته 5G را فراهم کند.این برای مواردی مناسب است که انتظار می‌رود پوشش LTE بسیار بهتر از پوشش 5G موجود باشد، به عنوان مثال، هنگام استفاده از 5G در باندهای mmW با شعاع سلول نسبتاً کوچک(شکل 3-3).



شکل 3-3 : گزینه های بیشتر معماری 3GPP بخشی از نسخه "Late drop" 3GPP 15 ]10[.

از نقطه نظر معماری، در نسخه 15 3GPP امکان اتصال دوگانه 5G-5G بین دو 5G gNB نیز فراهم شده است. در این مورد هم استفاده از باندهای mmW همراه با 5G در باندهای پایین تر و هم از نظر در نظر گرفتن 5G با باند پایین به عنوان باند پوشش در اتصال دوگانه با باند 2.5 گیگاهرتز یا 3.5 گیگاهرتز به عنوان باند ظرفیت ابراز مورد توجه قرار گرفته است. اتصال دوگانه 5G-5G در نسخه 15 فقط به نمونه های همگام سازی شده محدود شده است و قرار است نمونه ناهمزمان در فاز دوم 5G در نسخه مورد بررسی قرار گیرد.

## معماری شبکه هسته 5G

در سال‌های اخیر، بسیاری از ارائه‌دهندگان خدمات ارتباطی[[84]](#footnote-84) (CSP) EPC خود را مجازی‌سازی کرده‌اند و استقرار را از یک سری تجهیزات فیزیکی اختصاصی (سرورهای تعبیه شده خاص برای هر عملکرد)، به توابع شبکه مجازی‌سازی شده میزبانی شده بر روی سرورهای همه منظوره در فضای ابری تغییر داده‌اند. این شبکه‌های هسته مجازی‌سازی شده به CSPها اجازه می‌دهند تا به سرویس‌های جدید، مانند اینترنت اشیاء سلولار (IoT) یا ارائه خدمات هسته ویژه سازمانی، یا تجدید یا گسترش شبکه‌های قدیمی EPC گسترش یابند. در عمل، این مجازی سازی EPC در استقرار و استفاده از تجهیزات ساخته شده توسط سازنده های مختلف به خوبی کار نمی کند، زیرا در وهله اول در مشخصات EPC تعبیه نشده بود.

بنابراین، پشتیبانی از مجازی‌سازی عملکرد شبکه و شبکه‌سازی تعریف‌شده نرم‌افزاری یکی از اولین الزامات معماری برای شبکه هسته 5G بود. علاوه بر این، به همین اندازه مهم است که شبکه هسته 5G به قابلیت پیاده سازی در فضای ابری را نیز داشته باشد. برنامه های طراحی شده برای فضای ابری همچنین باید از اصول طراحی مقیاس بندی مانند تقسیم control panel و user panel و همچنین جداسازی منابع محاسباتی و منابع ذخیره سازی پیروی کنند، قابلیت هایی که در معماری EPC تعبیه نشده اند.

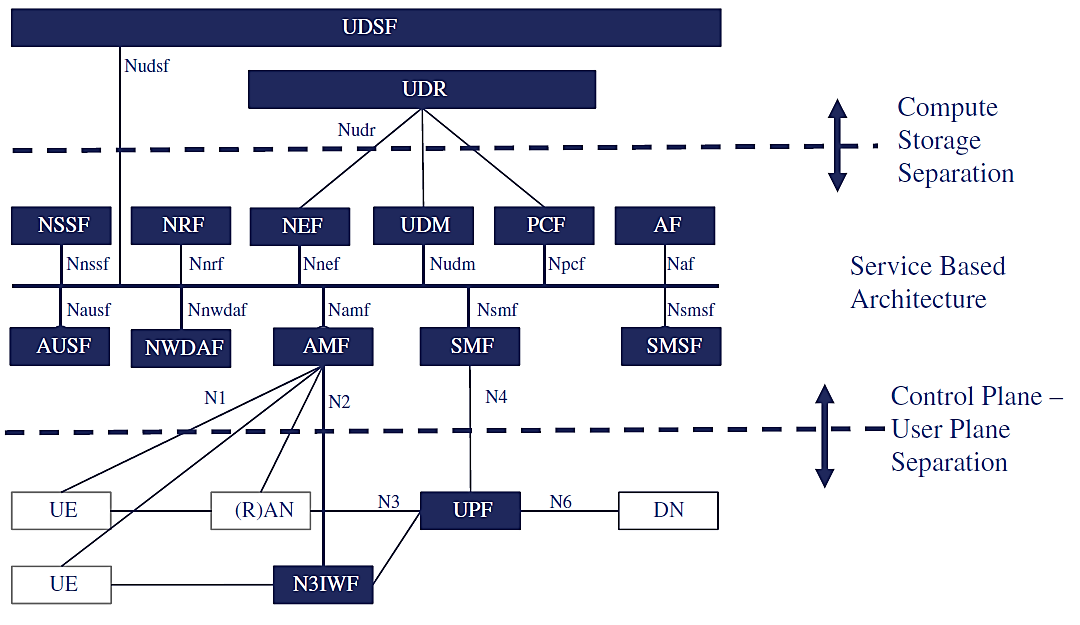
جداسازی user-plane و control-palne، امکان استفاده از معماری‌های ابری لبه توزیع‌شده[[85]](#footnote-85) را برای برنامه‌های کاربردی با پهنای باند فشرده و تاخیرات بحرانی که نیاز به توزیع مستقل توابع user-plane از توابع control-plane دارند، فراهم می‌کند. به عنوان مثال، برای برخی موارد استفاده با تأخیر کم، قرار دادن منابع user palne نزدیک به نقطه دسترسی که کاربر/دستگاه متصل است، برای کاهش تأخیر انتها به انتها و بار شبکه انتقال مهم است. شبکه هسته 5G بیشتر کاربران را قادر می سازد تا به طور همزمان به لبه و ابر مرکزی متصل شوند و بین آنها جابجا شوند.

شبکه هسته باید از توابع شبکه بدون حالت[[86]](#footnote-86) پشتیبانی کند که در آن منبع محاسباتی از منبع ذخیره سازی جدا شده است.جداسازی محاسبات و ذخیره سازی، مقیاس پذیری خطی نامحدود و انعطاف پذیری شدید را امکان پذیر می کند. در معماری 5G، مدیریت داده یکپارچه[[87]](#footnote-87) (UDM) میزبان داده‌های مشترک، داده‌های session، خط‌مشی‌ها، داده‌های عملیاتی، بارگزاری ها و داده‌های بررسی خواهد بود. همچنین یک اکوسیستم باز[[88]](#footnote-88) را برای نمایش داده ها و تجزیه و تحلیل آن ها فعال می کند.

تفاوت عمده دیگر در مقایسه با EPC این است که پشتیبانی چنددسترسی[[89]](#footnote-89) از همان ابتدا ایجاد شده است، که امکان استفاده کارآمد از منابع مختلف شبکه ثابت و تلفن همراه برای دسترسی به خدمات ارزش افزوده مستقل، حداکثر نرخ داده، افزایش قابلیت اطمینان و بهبود تجربه کاربر را فراهم می کند.

قابلیت ‌جدید ایجاد انتها به انتهای network slicing ؛ اپراتورهای عمومی و خصوصی را قادر می‌سازد تا با تطبیق پویای شبکه با نیازهای خاص خود، نیازهای بخش های خاص مختلف بازارهای هدف را برطرف کنند. برش های شبکه را می توان از نظر اتصال و تخصیص منابع از یکدیگر جدا کرد و "شبکه های خدمات خصوصی مجازی[[90]](#footnote-90)" را ایجاد کرد. اینها را می‌توان به روش‌های خاصی پیکربندی کرد تا مثلاً به تأخیر کم یا قابلیت اطمینان بالا یا هر دو دست یابد، و بخش‌های جدید شبکه را می‌توان به راحتی از طریق اتوماسیون معرفی کرد، بنابراین مدل‌های تجاری جدیدی را که با شبکه‌های فیزیکی اختصاصی امکان‌پذیر نبودند، امکان‌پذیر می‌سازد.

معماری ساده شده شبکه هسته 5G در شکل 3-4 نشان داده شده است.



شکل 3-4 : معماری شبکه هسته 5G (ساده شده) ]10[.

معماری نقطه به نقطه که در 4G استفاده شده است هنوز هم برای برخی از رابط های 5G اعمال می شود. در مدل نقطه به نقطه، تابعهای مختلف شبکه از طریق واسط های استاندارد شده[[91]](#footnote-91) متصل می شوند که امکان استفاده از شبکه های تولید شده توسط چندین سازنده را فراهم می کند. این هم از نظر مفهومی و هم از نظر عملیاتی به خوبی درک می شود و برای چندین دهه به اپراتورهای تلفن همراه خدمت کرده است. در معماری سیستم 5G، رابط‌های بین تجهیزات کاربر (UE) و هسته (N1)، شبکه دسترسی و هسته (N2 و N3)، و صفحه کنترل و کاربر (N4) همچنان از معماری نقطه به نقطه استفاده خواهد کرد.

با این حال، اکنون که با انتقال به زیرساخت ابری و نیاز به چابکی خدمات بیشتر، مدل نقطه به نقطه دیگر بهترین گزینه برای تمام رابط‌های شبکه هسته نیست.برای [[92]](#footnote-92)CSP هایی که 5G را به عنوان فرصتی برای تغییرات دگرگون کننده هم از نظر عملکرد و هم از نظر هزینه بر بیت می بینند، یک رابط مبتنی بر سرویس به جای رابط های نقطه به نقطه جذاب تر به نظر می رسد.

چالش معماری نقطه به نقطه این است که شامل بسیاری از واسط های منحصر به فرد یا شبه منحصر به فرد بین عناصر عملکردی است که هر کدام به چندین عنصر مجاور متصل هستند. این درهمتنیدگی اتصالات وابستگی هایی بین توابع ایجاد می کند و تغییر معماری مستقر را دشوار می کند. اگر یک تابع جدید معرفی شود، یا یک تابع موجود گسترش یا ارتقا یابد، اپراتورها باید چندین تابع مجاور را دوباره پیکربندی کنند و پیکربندی جدید را قبل از شروع به کار آزمایش کنند. این امر آستانه نمونه های تجاری را برای آزمایش و استقرار خدمات جدید افزایش می دهد.

در واقع، سرویس کاربر نهایی به شبکه گره خورده است و در نتیجه بازار قابل شناسایی اپراتور به طور مصنوعی محدود می شود. این معماری در جایی که مجموعه خدمات به خوبی تعریف شده و نسبتاً ثابت است (صدا، پهنای باند و غیره) قابل قبول است، اما در عصر 5G ، که اپراتورها انتظار دارند خدمات متنوعی را ارائه دهند باید بتوانند با تقاضاهایی که به سرعت در حال تغییر می باشند سازگار شوند یا نیازمندی های خاص صنعت، معماری پویاتر و چابک تر مورد نیاز است.

[[93]](#footnote-93)SBA سرویس کاربر نهایی را از زیرساخت شبکه و پلتفرم اصلی جدا می کند و با انجام این کار، چابکی عملکردی و خدماتی را امکان پذیر می کند.با توجه به اینکه SBA بر روی یک پایه اختصاصی ابری کار می کند، برای اپراتور بسیار ساده تر است که توابع شبکه مجازی (VNFs[[94]](#footnote-94)) را از مسیر پردازش شبکه اضافه، حذف یا تغییر دهد (چابکی عملکردی) و مسیرهای سرویس جدید را در صورت تقاضا ایجاد کند (چابکی خدمات). 3GPP، SBA را برای کاربرد در control plane هسته 5G انتخاب کرده است (قسمت میانی شکل3-4. رابط های مبتنی بر سرویس همگی بر اساس HTTP هستند و در شکل 3-4 با نشانه های اختصاصی Nxxx نشان داده شده اند.

### تابع مدیریت دسترسی و تحرک[[95]](#footnote-95)

عملگر مدیریت دسترسی و تحرک (AMF) بخشی از پردازش control panel در سمت هسته است.AMF سیگنال‌های Non-Access Stratum (NAS) را از UE از طریق شبکه دسترسی دریافت می‌کند و همچنین برای سیگنالینگ control panel بین شبکه دسترسی و شبکه هسته به gNB متصل می‌شود.عملکردهای کلیدی به شرح زیر است:

* درگاه ارتباطی (رادیویی) دسترسی به سیگنالینگ صفحه کنترل شبکه (N2).
* درگاه ارتباطی سیگنالینگ NAS (N1).
* احراز هویت دسترسی و مجوز دسترسی را ارائه می دهد.
* رمزگذاری NAS و محافظت از یکپارچگی را فراهم می کند.
* مدیریت رجیستری، مدیریت اتصال، مدیریت دسترسی، و مدیریت تحرک را فعال می کند.
* انتقال و پروکسی را برای پیام های مدیریت تماس[[96]](#footnote-96) (SM) بین UE و تابع مدیریت تماس[[97]](#footnote-97) (SMF) را فراهم می کند.
* پیام های SMS را بین UE و تابع سرویس پیام کوتاه (SMSF) حمل می کند.
* از سرویس موقعیت پشتیبانی می کند (به عنوان مثال مسیریابی پیام ها بین UE و تابع مدیریت موقعیت[[98]](#footnote-98) (LMF) و همچنین بین RAN و LMF).

### تابع مدیریت تماس

تابع SMF از مدیریت تماس مانند استقرار، اصلاح و انتشار تماس مراقبت می کند، از جمله نگهداری تونل بین تابع صفحه کاربر[[99]](#footnote-99) (UPF) و گره شبکه دسترسی، به عنوان مثال، gNB. وظایف این تابع شامل موارد زیر است:

* تخصیص و مدیریت آدرس IP UE را ارائه می دهد.
* انتخاب و کنترل UPF
* رابط تابع policy
* محموعه داده های بارگذاری شده در UPF را کنترل و هماهنگ می کند.
* مود Session و Service Continuity (SSC) تماس واحد داده پروتکل[[100]](#footnote-100) (PDU) را تعیین می کند.

### تابع user plane

پردازشUP توسط UPF پوشش داده می شود.ممکن است بیش از یک UPF در شبکه وجود داشته باشد، به عنوان مثال، یک UPF ترافیک محلی را مدیریت می کند و UPF دیگر تمام ترافیک های دیگر را مدیریت می کند.سرویس‌هایی که نیاز به تأخیر بسیار کم دارند (مانند موارد مربوط به URLLC و غیره) برای مدیریت محلی UPF مناسب هستند زیرا می‌توان از تأخیر اضافی بواسطه فاصله انتقال طولانی اجتناب کرد. عملکردهای کلیدی UPF به شرح زیر است:

* اجرای همه UPF ها روی بسته ها : ارسال، مسیریابی، علامت گذاری، کیفیت خدمات (QoS)، بازرسی و غیره.
* اجرا خط‌مشی‌های دریافتی از تابع کنترل policy (PCF) (از طریق SMF).
* ارسال گزارش مصرف ترافیک برای بارگذاری به SMF .
* به عنوان نقطه لنگر برای mobility بین فناوری دسترسی درون/بین رادیویی (RAT) عمل می کند.
* اتصال به شبکه های خارجی (N6) یا سایر UPF ها .

### معماری ذخیره سازی داده

مخزن یکپارچه داده[[101]](#footnote-101) (UDR) یک پایگاه داده مشترک برای انواع ساختارهای داده استاندارد شده از جمله داده های مشترکین، داده های خط مشی، داده های ساختار یافته برای نمایش دادن و داده های برنامه است. سه تابع مختلف شبکه می‌توانند به UDR برای خواندن، به‌روزرسانی، حذف و اشتراک در اعلان‌های تغییرات داده دسترسی داشته باشند.

UDM منطق برنامه را از UDR پنهان می کند.به عنوان مثال، UDM از تولید اعتبارنامه های احراز هویت، مدیریت هویت (هم کاربر و هم اشتراک)، ذخیره سازی اطلاعات در مورد بخش های خدماتی مانند AMF و SMF و پشتیبانی تحویل پیامک مراقبت می کند.علاوه بر داده‌های ساختاریافته استاندارد شده، معماری 5G هر تابع شبکه را قادر می‌سازد تا بدون ساختار مشخصی در تابع ذخیره‌سازی داده‌های بدون ساختار[[102]](#footnote-102) (UDSF) ذخیره کند.این قابلیت یک تابع شبکه را قادر می سازد تا تمام داده های مربوط به UE را در UDSF ذخیره کند و بنابراین تابع شبکه می تواند بدون حالت باشد. در نسخه 15، این قابلیت برای AMF تعریف شده است.

### تابع کنترل خط مشی[[103]](#footnote-103)

PCF رفتار شبکه را با اعمال یک چارچوب سیاست یکپارچه کنترل می کند. این تابع با قابلیت های اصلی زیر فعال می شود:

* ایجاد SM و مجموعه سیاست های مدیریت دسترسی.
* ارائه و حذف سیاست ها و تصمیمات مدیریت بارگذاری.
* ارائه و حذف تصمیمات مدیریت دسترسی و تحرک.
* ارایه خط‌مشی‌های انتخاب دسترسی UE و انتخاب تماس PDU به UE از طریق AMF.
* امکان استفاده از اطلاعات اشتراک مربوط به کنترل خط مشی و اطلاعات خاص برنامه ذخیره شده در UDR .
* امکان دریافت گزارش محدودیت مصرف از تابع بارگذاری.
* تعامل با NEF (تابع دسترسی شبکه)، به عنوان مثال، برای دریافت مجوز هدایت ترافیک تحت تأثیر عملکرد برنامه.

### تابع دسترسی شبکه[[104]](#footnote-104)

* قابلیت ها و رویدادهای هسته 5G را در دسترس برنامه های خارجی قرار می دهد. از UDR به عنوان منبع داده خود استفاده می کند.
* به اشخاص ثالث اجازه می دهد تا به طور ایمن اطلاعات را در اختیار هسته 5G قرار دهند (مانند رفتار مورد انتظار UE).
* پنهان کردن اطلاعات حساس شبکه نسبت به تجهیزات بیرونی را کنترل می‌کند.

### تابع مخزن شبکه[[105]](#footnote-105)

* نمایه تابع شبکه (NF) نمونه های NF موجود و خدمات پشتیبانی شده آنها را حفظ می کند و کشف و انتخاب NF را فراهم می کند.
* معیارهای انتخاب می تواند شامل مکان (تاخیر)، بار، نام شبکه داده (برنامه)، نوع شبکه دسترسی، برش و غیره باشد.
* در مقایسه با انتخاب نام دامنه سیستم[[106]](#footnote-106) (DNS) خط مشی ها و قابلیت های پویا تری را ارائه می دهد.

### انتخاب برش شبکه[[107]](#footnote-107)

* AMF یا مجموعه AMF کاندید مورد استفاده برای خدمت به UE را تعیین می کند.
* برش های شبکه ای را که UE می تواند متصل شود را انتخاب می کند.
* از نگاشت شناسه های خاص برش شبکه مراقبت می کند.

### تابع متقابل غیر GPP[[108]](#footnote-108)3

Non-3GPP InterWorking Function (N3IWF) از اتصال UE ها از طریق دسترسی غیر 3GPP (دسترسی غیرقابل اعتماد Wi-Fi در نسخه 15) پشتیبانی می کند. این با قابلیت های اصلی زیر فعال می شود:

* پشتیبانی از ایجاد تونل IPsec با UE به منظور احراز هویت و مجوز دسترسی به هسته 5G
* ارتباط امنیتی IPsec را برای پشتیبانی از ترافیک تماس PDU ایجاد می کند.
* وظایف مربوط به ترافیک user plane را انجام می دهد، مانند انتقال uplink و downlink بسته های UP بین UE و UPF و علامت گذاری بسته ها
* مدیریت ترافیک control plane مانند انتقال سیگنالینگ NAS بین UE و AMF و عبور سیگنال از SMF مربوط به تماس PDU و QoS

### توابع پشتبان هسته 5G[[109]](#footnote-109)

تابع سرویس پیام کوتاه[[110]](#footnote-110) (SMSF) :

توابع لازم را برای فعال کردن سرویس پیام کوتاه بین 5G UE و عملکرد SMS قدیمی در شبکه را فراهم می کند.

تابع سرور احراز هویت[[111]](#footnote-111) (AUSF) :

سندیت UE را تأیید می کند و ملزومات کلیدی مرتبط را برای آن فراهم می کند.

پروکسی محافظ لبه امنیتی[[112]](#footnote-112) (SEPP) :

فیلتر کردن پیام و نظارت بر رابط های سطح کنترل بین PLM را ارائه می دهد و توپولوژی شبکه را از سایر تجهیزات پنهان می کند زیرا می تواند به عنوان یک نقطه تماس عمل کند.

تابع تجزیه و تحلیل داده های شبکه[[113]](#footnote-113) (NWDAF) :

تجزیه و تحلیل داده های برش های مشخص شبکه را به PCF و NSSF ارائه می دهد.

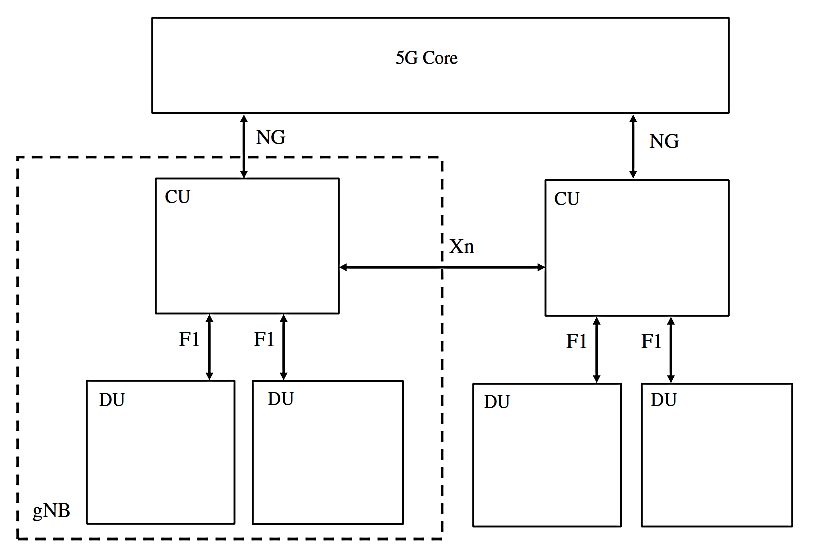
تابع مدیریت مکان[[114]](#footnote-114) (LMF):

اندازه گیری های موقعیت را به طور مساوی از UE و شبکه دسترسی رادیویی به دست می آورد.

## معماری 5G RAN

معماری دسترسی رادیویی G5 از دو عنصر اساسی به شرح زیر تشکیل شده است:

* واحد مرکزی[[115]](#footnote-115) (CU)، که برای رسیدگی به لایه های نسبتاً بالاتر روی لایه فیزیکی در نظر گرفته شده است. عملیاتی کردن فیزیکی CU می تواند سخت افزار اختصاصی یا یک نوع پیاده سازی رادیویی ابری باشد. همانطور که در شکل 3-5 نشان داده شده است، CU از یک طرف به واحد توزیع شده ([[116]](#footnote-116)DU) و از طرف دیگر به شبکه هسته 5G (هسته نسل بعدی[[117]](#footnote-117) (NGC) متصل است.
* DU که همراه با آنتن ها و واحد RF در محل سلول قرار دارد. انتظار می رود DU به ویژه پردازش های حساس به زمانی را انجام دهد که تاخیر زیادی را برای پردازش دور از سایت سلول تحمل نمی کنند.



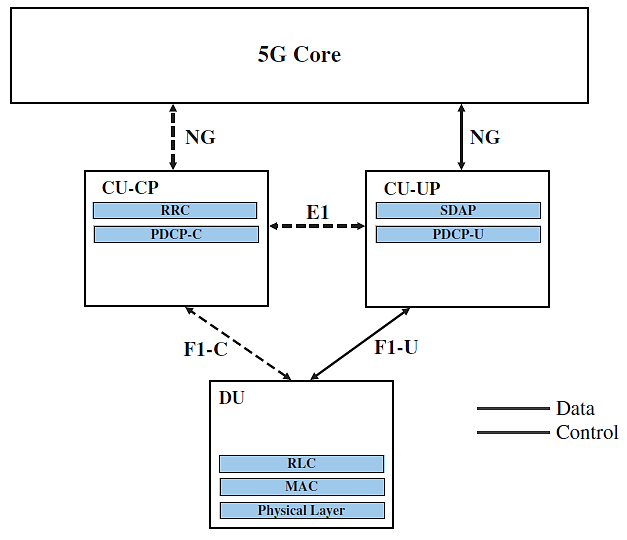
شکل 3-5: معماری کلی شبکه دسترسی رادیویی G5 ]10[.

در ملاحظات 3GPP، رویکرد وجود فقط RF در سایت (با آنتن) به دلیل الزامات انتقال بسیار بالای ارسال نمونه های IQ مطلوب در نظر گرفته نشده است. اگرچه چنین رویکردی در برخی از بازارها با LTE مورد استفاده قرار گرفت، تعداد زیاد آنتن‌ها و همچنین پهنای باند بزرگ‌تر در نظر گرفته شده باعث می‌شود نرخ داده‌های مورد نیاز آنقدر زیاد باشد که اغلب در استقرار شبکه‌های مقیاس بزرگ از این رویکرد استفاده نشود.

رابط های زیر به عنوان بخشی از معماری دسترسی رادیویی[[118]](#footnote-118) 5G تعریف شده اند:

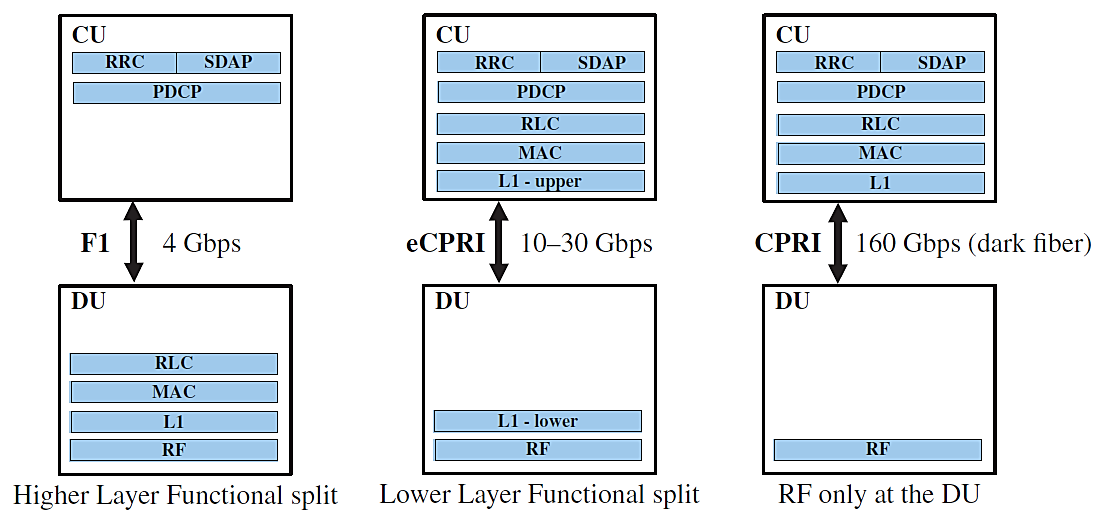
* رابط F1، CU را به یک یا چند DU متصل می کند. انتظار می رود که یک CU واحد بتواند چندین DU را مدیریت کند.
* رابط Xn، CUهای مختلف را به هم متصل می کند. هنگام در نظر گرفتن معماری استقرار اولیه با هسته LTE (EPC)، رابط بین LTE eNB و 5G gNB، رابط X2 نامیده می شود.
* رابط E1 جداسازی پردازش control panel و user panel ؛ CU را به بخش های CU-CP و CU-UP تسهیل می کند، همانطور که در شکل 3-6 نشان داده شده است.رابط E1 از نسخه های ژوئن 2018 به بعد در دسترس است.
* رابط NG به سمت هسته 5G (5GC). به بخش صفحه کاربر (NG-U) و قسمت صفحه کنترل (NG-C) تقسیم می شود.در معماری مرجع کلی در 3GPP، NG-U را رابط N3 و NG-C را رابط N2 می نامند.

تقسیم عملکردی که در شکل 3-6 نشان داده شده است، تقسیم عملکردی لایه بالاتر نیز نامیده می شود. این تنها تقسیم عملکردی است که در 3GPP در نسخه 15 تعریف شده است.این تقسیم عملکردی همچنین برای مواردی مناسب است که backhaul دارای محدودیت‌هایی در زمان تاخیر است زیرا ارسال مجدد به صورت محلی در DU کنترل می‌شود.این همچنین به CU اجازه می دهد تا نسبتاً دورتر از زمانی که ارسال مجدد در سمت DU کنترل می شد قرار گیرد.



شکل 3-6 : معماری شبکه دسترسی 5G با جداسازی user palne و control plane برای CU با تقسیم لایه بالاتر]10[.

رویکردهای تقسیم عملکردی لایه پایین تا کنون در 3GPP پوشش داده نشده است. جایگزین های مختلف به هر حال در طول انتشار 3GPP 15 مورد مطالعه قرار گرفتند.هدف این رویکرد عملکرد بیشتر در CU برای تسهیل مدیریت مرکزی و داشتن عملکرد کمتر در سایت و واحدهای RF می باشد.به هر حال وجود برخی از عملکردهای L1 در RF و آنتن‌های موجود در سایت برای جلوگیری از ارسال نمونه‌های I/Q ضروری به نظر می‌رسند، زیرا این امر منجر به الزامات گسترده در اتصال backhaul می‌شود. این در شکل 3-7 نشان داده شده است، که نشان می دهد ضریب 40 تفاوت در نرخ داده های تخمین زده شده هنگام مقایسه بیت های ارسالی پس از لایه پروتکل همگرایی داده های بسته (PDCP) و هنگام ارسال نمونه های I/Q برای فقط RF فقط در سایت DU وجود دارد. انتقال MAC/RLC به CU به طور طبیعی منجر به نرخ های داده تا حدودی بالاتر می شود، زیرا اکنون انتقال مجدد به CU منتقل شده است و بنابراین برخی از بسته ها چندین بار از طریق رابط F1 علاوه بر سیگنال کنترل اضافی که ضروری می شود، ارسال می شوند.



شکل 3-7 : تاثیر اتصالات backhaul با آپشن ها تقسیم عملکردی متفاوت ]10[.

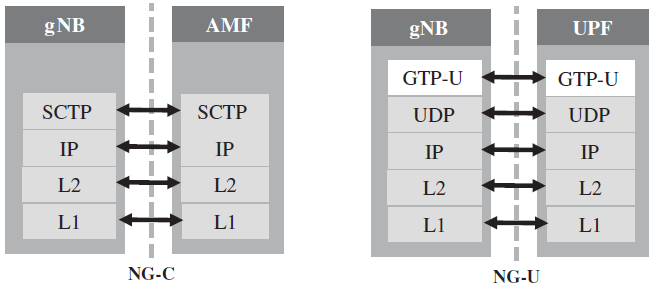
یک تقسیم لایه پایین تر با CPRI بهبود یافته (eCPRI) که در شکل 3-7 نشان داده شده است در سازمان O-RAN Alliance [12] خارج از 3GPP معرفی شده است، و رابط نیز بعنوان رابط رادیویی عمومی مشترک[[119]](#footnote-119) (CPRI) تحت عنوان eCPRI تعریف شده است که گزینه هایی غیر از فیبر تاریک مانند اترنت را تسهیل و ممکن می کند.3GPP در نسخه 15 یا 16 چنین تقسیمات عملکردی لایه پایین تر در نسخه های بعدی را بررسی نمی کند پاسخ نمی دهد و حداقل در نسخه 16 برنامه ریزی نشده است که از چنین تقسیماتی استفاده شود. بنابراین، از این منظر، مشخصات F1 در نسخه 15 تنها تقسیم عملکردی لایه بالاتر را پوشش می دهد. اصطلاح رابط F2 که به رابط eCPRI اشاره دارد، خارج از 3GPP برای اشاره به رابط بین DU (با RF و L1 پایین) و CU (L1 بالایی و سایر عملکردهای لایه بالاتر) استفاده شده است، اما رابط eCPRI/F2 توسط مشخصات 3GPP پوشش داده نمی شود.

### رابط NG

رابط NG gNB (5G-RAN) را به هسته 5G متصل می کند. این مبتنی بر استفاده از انتقال IP مبتنی بر GTP-U و UDP است.مشابه رابط S1 بین LTE و هسته EPC، رابط NG نیز به عنوان یک رابط چند فروشنده باز تعریف شده است.رابط NG به روش دسترسی-آگنوستیک[[120]](#footnote-120) و سازگار با سایر پلتفرم ها طراحی شده است تا همگرایی به سمت استفاده از هسته 5G را برای سایر فناوری های دسترسی نیز تسهیل کند.این نیازمند توانایی دستگاه های نهایی برای استفاده از سیگنالینگ NAS تعریف شده 3GPP برای راه اندازی/رجیتسری اتصال است.

پشته های پروتکل رابط NG برای صفحات کنترل و کاربر در شکل 3-8 نشان داده شده است.بخش صفحه کنترل مبتنی بر استفاده از پروتکل کنترل جریان انتقال[[121]](#footnote-121) (SCTP) در بالای اتصال IP است.در حالی که بخش user plane داده های کاربر را منتقل می کند، عملکردهای کلیدی بخش control plane رابط NG به شرح زیر است:

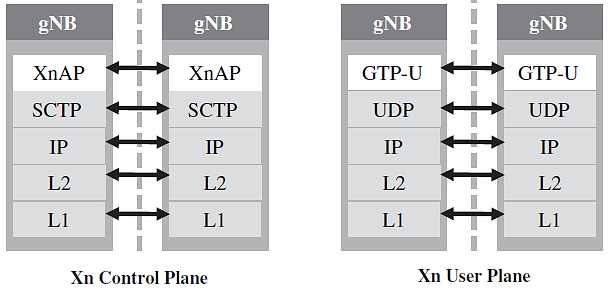
* مدیریت رابط NG
* شرایط UE و مدیریت تحرک
* انتقال پیام های NAS
* Paging
* مدیریت تماس PDU



شکل 3-8: پشته پروتکل صفحه کنترل و کاربر رابط NG ]10[.

### رابط Xn

رابط Xn، gNB ها را به gNB دیگر (یا به eNB) متصل می کند.اصطلاح Xn-Interface همیشه هنگام کار با هسته NG (5G-core) استفاده می شود.هنگامی که گزینه معماری 3 با هسته 4G EPC استفاده می شود، آنگاه رابط کاربری X2-Interface نامیده می شود. پشته پروتکل رابط Xn در شکل 3-9 نشان داده شده است.مانند رابط X2 در شبکه های LTE، رابط Xn برای فعال کردن عملکرد به عنوان یک رابط multivendor تعریف شده است.شکل 3-9 تقسیم پروتکل Xn را نشان می دهد.از نقطه نظر ابعاد، رابط Xn با رابط X2 مبتنی بر LTE متفاوت است زیرا داده های کاربر از طریق Xn/X2 به طور پیوسته تر از LTE هدایت می شوند.در LTE، فقط برای ارسال موقت بسته استفاده می شد.



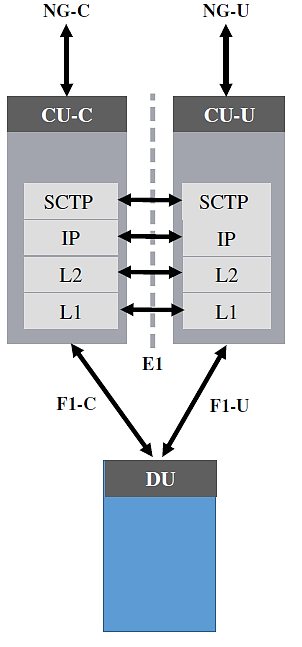
شکل 3-9 : پروتکل های صفحه کنترل و کاربر Xn ]10[.

برای سمت user plane، عملکرد کلیدی انتقال داده و کنترل جریان داده است، در حالی که در سمت control plane، عملکردهای کلیدی رابط Xn به شرح زیر است:

* مدیریت رابط Xn
* مدیریت تحرک UE، از جمله انتقال محتوی و صفحه بندی RAN
* اتصال دوگانه

### رابط E1

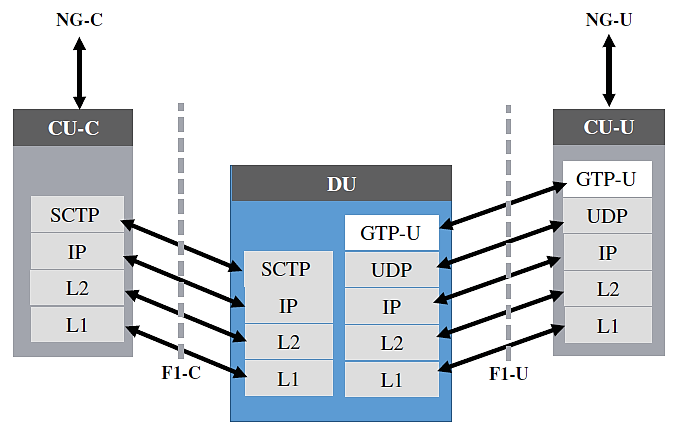
رابط E1 صفحه کاربر و بخش های صفحه کنترل تابع CU را به هم متصل می کند.رابط E1 در نسخه ژوئن 2018 به همراه گزینه 5G مستقل و هسته 5G اضافه شد.پشته پروتکل رابط E1 در شکل 3-10 نشان داده شده است.



شکل 3-10 : پشته پروتکل رابط E1 ]10[

### رابط F1

رابط F1، CU و DU را به هم متصل می کند. در نسخه 15، تقسیم عملکردی پشتیبانی شده، همانطور که قبلا در این فصل ارائه شد، این تقسیم عملکردی لایه بالاتر است.پشته پروتکل رابط F1 در شکل 3-11 نشان داده شده است.



شکل 3-11: پشته های پروتکل رابط F1 ]10[

# انواع برنامه های متن باز برای برای پیاده سازی هسته نسل 4 و 5

## هسته شبکه

در قسمت لیستی از نرم افزارهای open source که میتوان برای اجرای هسته شبکه های LTE و 5G استفاده کرد لیست شده است]15[.

### Evolved packet core (EPC)

این قسمت پیاده‌سازی های باز EPC لیست شده است.

**OAI CN**

مشارک کنندگان اصلی :‌ OAI Alliance

License : OAI Public License

وضعیت : فعال

OAI-CN با زبان‌های برنامه نویسی C و C++ نوشته شده است. QoS پویا با ایجاد، اصلاح و حذف حامل‌های اختصاصی متعدد، و به‌روزرسانی QoS مبتنی بر policy نیز ویژگی‌هایی هستند که توسط OAI-CN MME پیاده‌سازی شده‌اند. عملیات الگوی جریان ترافیک[[122]](#footnote-122) (TFT) مانند تشخیص خطا، قوانین فیلتر و فیلترهای IP نیز ارائه شده است. همچنین از شاخص های انباشتگی implicit (به عنوان مثال، شکست درخواست خدمات) وexplicit (به عنوان مثال، منبع حامل و دستورات حذف) به همراه نام نقطه دسترسی چندگانه[[123]](#footnote-123) (APN)، paging، و روش‌های بازیابی پشتیبانی می‌شوند.

**SrsEPC :**

مشارکت کنندگان اصلی : SRS

لایسنس : GNU AGPLv3

وضعیت : فعال

پیاده سازی EPC موجود در مجموعه نرم افزار srsRAN، یعنی srsEPC، به زبان C++ نوشته شده است. با سیستم عامل های لینوکس اوبونتو و فدورا سازگار است.HSS از پیکربندی پارامترهای مربوط به UE در قالب یک فایل CSV متنی ساده پشتیبانی می کند. احراز هویت UE توسط الگوریتم های احراز هویت XOR و Milenage پشتیبانی می شود. srsEPC شناسه کلاس QoS (QCI)و پیکربندی IP پویا یا استاتیک را برای هر کاربر فعال می کند.

**OMCE**

مشارکت کنندگان اصلی : ONF, Intel, Deutsche Telekom, Sprint, AT&T

لایسنس: Apache V2.0

وضعیت : فعال

این یک پیاده‌سازی منبع باز با کارایی بالا از LTE Release 13 EPC است که توسط ONF همراه با اپراتورهای مخابراتی و شرکای صنعتی مانند Intel، Deutsche Telekom، Sprint و AT&T توسعه یافته است.OMEC با استفاده از معماری NFV برای حفظ مقیاس پذیری در سناریوهای مقیاس بزرگ مانند برنامه های 5G و IoT ساخته شده است. ویژگی های اتصال،صدور صورتحساب و شارژ را برای کابران ارائه می دهد. OMEC را می توان به عنوان یک EPC مستقل یا یکپارچه در چارچوب های بزرگتر یعنی همگرای چند دسترسی و هسته[[124]](#footnote-124) (COMAC) استفاده کرد.

### هسته 5G

این بخش پیاده سازی های باز هسته 5G را فهرست می کند]15[.

**Open5GS:**

مشارکت کنندگان اصلی: Open5GS

لایسنس: GNU AGPLv3

وضعیت : فعال

Open5GS یک پیاده‌سازی منبع باز نسخه 16 4G/5G مطابق با EPC NSA و یک هسته 5G SA ارایه می دهد. اجزای شبکه 4G/5G NSA و عملکردهای شبکه 5G SA به زبان C نوشته شده و تحت مجوز AGPLv3 توزیع شده است. این پیاده سازی با انواع توزیع های لینوکس مانند دبیان، اوبونتو، فدورا و CentOS و همچنین FreeBSD و macOS سازگار است. برخلاف سایر EPCها، Open5GS از ارسال تماس های صوتی و پیام های متنی از طریق شبکه LTE به جای تکیه بر شبکه های سوئیچینگ مدار سنتی[[125]](#footnote-125) پشتیبانی می کند. این امر به ترتیب با استفاده از سرویس های سومی مانند Kamailio و Osmocom ممکن می شود که میتوان به تکنولوژی صدا از طریق LTE (VoLTE) و راه حل های SG-SMS (SMSoSGs) به دست می آید. Open5GS شامل ماژول‌های PCRF/PCF است که از طریق آن اپراتورها می‌توانند سیاست‌های شبکه را بصورت real-time مشخص کنند، از جمله اولویت‌بندی نوع خاصی از ترافیک. قابلیت‌های هسته 5G SA، مانند slicing، در حال حاضر در دست توسعه هستند.Open5GS با انواع نرم افزارهای 4G و 5G و تجهیزات رادیویی فیزیکی استفاده می شود. لیستی از تجهیزات سازگار آزمایش شده در https://open5gs.org/open5gs/docs/hardware/01-genodebs موجود است.

**Free5GC**

مشارکت اصلی : free5GC

لایسنس :‌ Apache v2.0

وضعیت : فعال

free5GC به زبان برنامه نویسی Go نوشته شده است و با ماشین هایی که سیستم عامل لینوکس اوبونتو را اجرا می کنند سازگار است. این پیاده سازی که در ابتدا بر اساس NextEPC ( Open5GSفعلی) بود، از مدیریت دسترسی کاربر، تحرک و تماس ها (AMF و SMF) و کشف خدمات ارائه شده توسط سایر توابع شبکه (تابع مخزن شبکه (NRF)) پشتیبانی می کند. همچنین شامل توابع شبکه برای انتخاب برش های شبکه برای تخصیص به UE ها (عملکرد انتخاب قطعه شبکه (NSSF))، مدیریت، ذخیره و بازیابی داده های کاربر (مدیریت یکپارچه داده (UDM) و مخزن یکپارچه داده (UDR))، برای انجام احراز هویت UEs در شبکه (عملکرد سرور احراز هویت (AUSF)) می باشد. توابعی برای عملیات، مدیریت و مدیریت شبکه هسته (عملیات، مدیریت و نگهداری (OAM))، و انجام هماهنگی شبکه، در میان سایر موارد، نیز گنجانده شده است.به عنوان مثال، این پیاده سازی امکان پشتیبانی از موارد استفاده MEC را فراهم می کند.

# مقدمه ای بر Open5GS

Open5GS شامل مجموعه‌ای از اجزای نرم‌افزاری و توابع شبکه است که عملکردهای اصلی 4G/5G NSA و 5G SA را پیاده‌سازی می‌کنند]13[.

## هسته های قابل اجرا در open5GS

### 4G/ 5G NSA Core

هسته 4G/5G NSA ؛ open5GS همانطور که در شکل 4-1 نشان داده شده است شامل اجزای زیر می‌باشد:

•MME - Mobility Management Entity

• HSS - Home Subscriber Server

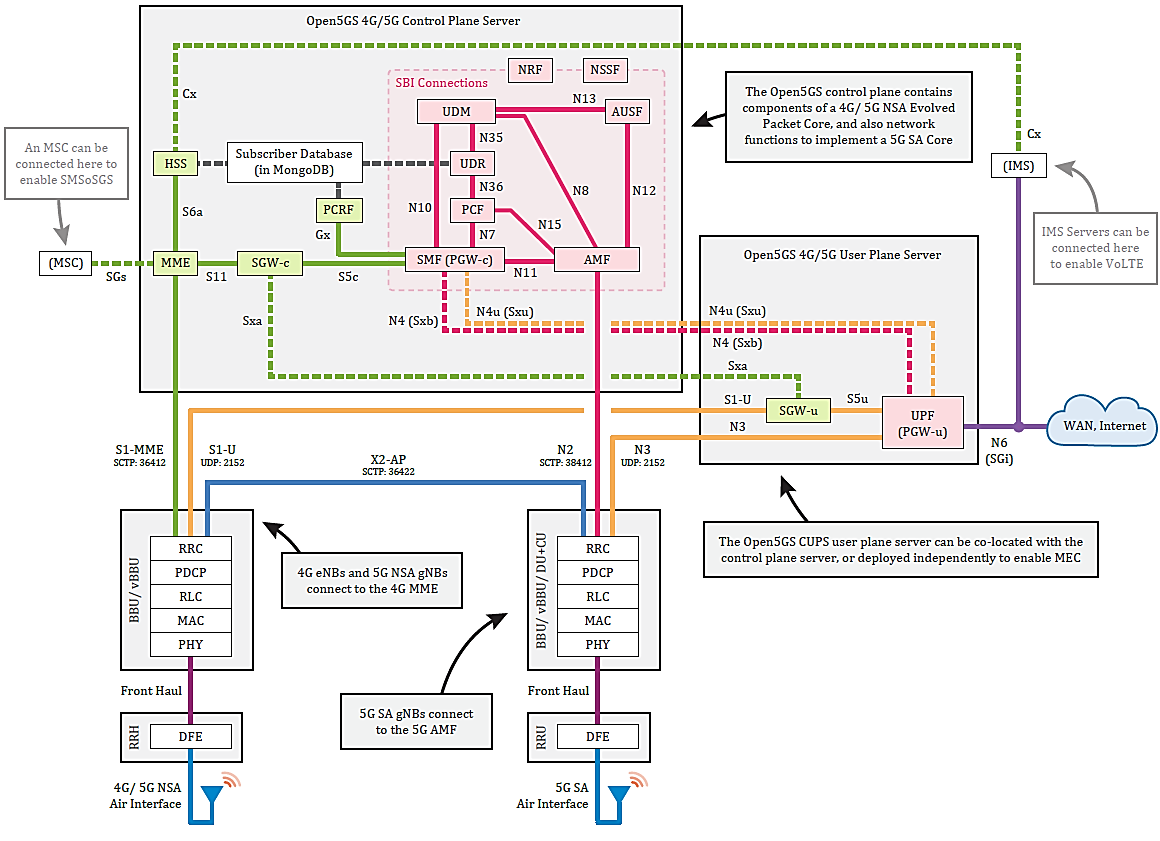
• PCRF - Policy and Charging Rules Function

• SGWC - Serving Gateway Control Plane

• SGWU - Serving Gateway User Plane

• PGWC/SMF - Packet Gateway Control Plane / (component contained in Open5GS SMF)

• PGWU/UPF - Packet Gateway User Plane / (component contained in Open5GS UPF)



شکل 4-1 : اجزا و توابع open5GS ]13[.

هسته دارای دو صفحه اصلی است: صفحه کنترل و صفحه کاربر. اینها به صورت فیزیکی در Open5GS از هم جدا شده اند همچنین CUPS[[126]](#footnote-126) (جداسازی صفحه کنترل/ کاربر) پیاده سازی شده است]13[.

MME، هاب اصلی control plane هسته است.در وهله اول تماس ها، تحرک، صفحه بندی و حامل ها را مدیریت می کند. به HSS، که بردارهای احراز هویت سیم کارت را تولید می کند و مشخصات مشترکین را نگه می دارد، و همچنین به SGWC و PGWC/SMF که سطوح کنترلی سرورهای gateway هستند متصل می شود. جز نهایی صفحه کنترل PCRF است که بین PGWC/SMF و HSS قرار می گیرد و بارگذاری را مدیریت می کند و سیاست های مشترکین را اجرا می کند.

User plane بسته های داده کاربر را بین eNB/NSA gNB (ایستگاه های پایه 5G NSA) و WAN بیرونی حمل می‌کند. دو جزء اصلی صفحه کاربر SGWU و PGWU/UPF هستند. هر یک از اینها به همتایان خود در صفحه کنترل متصل می شوند. gNBs eNB/ NSA به SGWU که به PGWU/UPF وصل می شود و به WAN متصل می شوند. با تفکیک فیزیکی صفحه های کنترل و کاربر به این صورت، می توانید چندین سرور user panel پیاده سازی کنید (مثلاً جایی با اتصال اینترنت پرسرعت)، در حالی که عملکرد کنترل را متمرکز نگه دارید.

همه این اجزای Open5GS دارای فایل های پیکربندی هستند.هر فایل پیکربندی شامل آدرس‌های پیوند IP مؤلفه / نام‌های رابط محلی و آدرس‌های IP / نام‌های DNS سایر مؤلفه‌هایی است که باید به آنها متصل شود. که در انتهای قسمت بعد آدرس IP واحدهای مختلف نشان داده شده است.

### 5G Core

هسته Open5GS 5G SA شامل توابع زیر است]13[:

•AMF - Access and Mobility Management Function

• SMF - Session Management Function

• UPF - User Plane Function

• AUSF - Authentication Server Function

• NRF - NF Repository Function

• UDM - Unified Data Management

• UDR - Unified Data Repository

• PCF - Policy and Charging Function

• NSSF - Network Slice Selection Function

• BSF - Binding Support Function

هسته 5G SA به روشی متفاوت از هسته 4G کار می کند - از معماری مبتنی بر خدمات (SBI) استفاده می کند. توابع control plane برای رجیستری با NRF پیکربندی شده اند و NRF سپس به آنها کمک می کند تا سایر توابع هسته را کشف کنند. اجرای سایر توابع: AMF مدیریت اتصال و تحرک را بر عهده دارد. زیر مجموعه ای از آنچه که 4G MME وظیفه دارد. gNB ها (ایستگاه های پایه 5G) به AMF متصل می شوند. UDM، AUSF و UDR عملیات مشابه 4G HSS را انجام می دهند و بردارهای احراز هویت سیم کارت را تولید می کنند و مشخصات مشترکین را نگه می دارند. مدیریت تماس ها همه توسط SMF انجام می شود (که قبلاً مسئولیت 4G MME/SGWC/PGWC بود). NSSF راهی برای انتخاب برش شبکه فراهم می کند. در نهایت PCF وجود دارد که برای بارگذاری و اجرای سیاست های مشترکین استفاده می شود.

User plane هسته 5G SA بسیار ساده تر است، زیرا فقط یک تابع واحد دارد.UPF بسته های داده کاربر را بین gNB و WAN خارجی حمل می کند. به SMF نیز متصل می شود.

به استثنای SMF و UPF، همه فایل‌های پیکربندی برای توابع هسته 5G SA فقط حاوی آدرس‌های پیوند IP تابع/ نام‌های رابط محلی و آدرس IP/نام DNS، NRF هستند.

تنظیمات پیش‌فرض همه اجزای Open5GS را به‌طور کامل برای استفاده در یک رایانه پیکربندی می‌کنند.آنها تنظیم شده اند تا با استفاده از فضای آدرس loopback محلی (127.0.0.X) با یکدیگر ارتباط برقرار کنند. آدرس های پیش فرض برای هر یک از رابط های bind برای این مؤلفه ها و توابع به شرح زیر است:

MongoDB = 127.0.0.1 (subscriber data) - http://localhost:3000

MME-s1ap = 127.0.0.2 :36412 for S1-MME

MME-gtpc = 127.0.0.2 :2123 for S11

MME-frDi = 127.0.0.2 :3868 for S6a

SGWC-gtpc = 127.0.0.3 :2123 for S11

SGWC-pfcp = 127.0.0.3 :8805 for Sxa

SMF-gtpc = 127.0.0.4 :2123 for S5c, N11

SMF-gtpu = 127.0.0.4 :2152 for N4u (Sxu)

SMF-pfcp = 127.0.0.4 :8805 for N4 (Sxb)

SMF-frDi = 127.0.0.4 :3868 for Gx auth

SMF-sbi = 127.0.0.4 :7777 for 5G SBI (N7,N10,N11)

AMF-ngap = 127.0.0.5 :38412 for N2

AMF-sbi = 127.0.0.5 :7777 for 5G SBI (N8,N12,N11)

SGWU-pfcp = 127.0.0.6 :8805 for Sxa

SGWU-gtpu = 127.0.0.6 :2152 for S1-U, S5u

UPF-pfcp = 127.0.0.7 :8805 for N4 (Sxb)

UPF-gtpu = 127.0.0.7 :2152 for S5u, N3, N4u (Sxu)

HSS-frDi = 127.0.0.8 :3868 for S6a, Cx

PCRF-frDi = 127.0.0.9 :3868 for Gx

NRF-sbi = 127.0.0.10:7777 for 5G SBI

AUSF-sbi = 127.0.0.11:7777 for 5G SBI

UDM-sbi = 127.0.0.12:7777 for 5G SBI

PCF-sbi = 127.0.0.13:7777 for 5G SBI

NSSF-sbi = 127.0.0.14:7777 for 5G SBI

BSF-sbi = 127.0.0.15:7777 for 5G SBI

UDR-sbi = 127.0.0.20:7777 for 5G SBI

## نصب Open5GS از منابع

در این قسمت نحوه کامپایل و نصب کد منبع open5GS را در توزیع لینوکس مبتنی بر Debian/Ubuntu توضیح داده شده است ]14[.

(توجه: در Ubuntu 16.04 (xenial) و نسخه های قبلی و Debian 9 (stretch) و نسخه های قبلی پشتیبانی نمی شوند.)

### دریافت MongoDB

MongoDB را با مدیریت بسته نصب کنید. به عنوان پایگاه داده برای NRF/PCF/UDR و PCRF/HSS استفاده می شود.

$ sudo apt update

$ sudo apt install mongodb

$ sudo systemctl start mongodb (if '/usr/bin/mongod' is not running)

$ sudo systemctl enable mongodb (ensure to automatically start it on system boot)

راه اندازی دستگاه TUN

دستگاه TUN را با نام رابط ogstun ایجاد کنید.

$ sudo sh -c "cat << EOF > /etc/systemd/network/99-open5gs.netdev

[NetDev]

Name=ogstun

Kind=tun

EOF"

$ sudo ip tuntap add name ogstun mode tun

$ sudo ip addr add 10.45.0.1/16 dev ogstun

$ sudo ip addr add 2001:db8:cafe::1/48 dev ogstun

$ sudo ip link set ogstun up

$ sudo systemctl enable systemd-networkd

$ sudo systemctl restart systemd-networkd

$ sudo sh -c "cat << EOF > /etc/systemd/network/99-open5gs.network

[Match]

Name=ogstun

[Network]

Address=10.45.0.1/16

Address=2001:db8:cafe::1/48

EOF"

$ sudo systemctl restart systemd-networkd

### ساخت Open5GS

ابتدا پیش نیاز را برای ساختن کد منبع نصب کنید.

sudo apt install python3-pip python3-setuptools python3-wheel ninja-build build-essential flex bison git libsctp-dev libgnutls28-dev libgcrypt-dev libssl-dev libidn11-dev libmongoc-dev libbson-dev libyaml-dev libnghttp2-dev libmicrohttpd-dev libcurl4-gnutls-dev libnghttp2-dev libtins-dev libtalloc-dev meson

کلون کردن open5GS از گیت هاب:

$ git clone https://github.com/open5gs/open5gs

کامپایل کردن با meson :

$ cd open5gs

$ meson build --prefix=`pwd`/install

$ ninja -C build

بررسی کنید که آیا کامپایل درست است یا خیر.

$ ./build/tests/attach/attach ## EPC Only

$ ./build/tests/registration/registration ## 5G Core Only

تمام برنامه های تست ها را مانند زیر اجرا کنید.

cd build

meson test -v

سپس مراحل نصب را تکمیل کنید.

cd build

ninja install

cd ../

## پیکربندی Open5GS

### هسته 5G

برای تنظیم آدرس IP NGAP، شناسه NSSAI،PLMN، TAC کانقیگ فایل install/etc/open5gs/amf.yaml را تغییر دهید.

$ diff -u /etc/open5gs/amf.yaml.old /etc/open5gs/amf.yaml

--- amf.yaml 2020-09-05 20:52:28.652234967 -0400

+++ amf.yaml.new 2020-09-05 20:55:07.453114885 -0400

@@ -165,23 +165,23 @@

- addr: 127.0.0.5

port: 7777

ngap:

- - addr: 127.0.0.5

+ - addr: 10.10.0.5

guami:

- plmn\_id:

- mcc: 999

- mnc: 70

+ mcc: 001

+ mnc: 01

amf\_id:

region: 2

set: 1

tai:

- plmn\_id:

- mcc: 999

- mnc: 70

- tac: 1

+ mcc: 001

+ mnc: 01

+ tac: 2

plmn\_support:

- plmn\_id:

- mcc: 999

- mnc: 70

+ mcc: 001

+ mnc: 01

s\_nssai:

- sst: 1

security:

برای تنظیم آدرس IP GTP-U و PFCP کانفیگ فایل install/etc/open5gs/upf.yaml را تغییر دهید.

$ diff -u /etc/open5gs/upf.yaml.old /etc/open5gs/upf.yaml

--- upf.yaml 2020-09-05 20:52:28.652234967 -0400

+++ upf.yaml.new 2020-09-05 20:52:55.279052142 -0400

@@ -137,9 +137,7 @@

pfcp:

- addr: 127.0.0.7

gtpu:

- - addr:

- - 127.0.0.7

- - ::1

+ - addr: 10.11.0.7

subnet:

- addr: 10.45.0.1/16

- addr: 2001:db8:cafe::1/48

### 4G EPC

برای تنظیم کردن IP آدرس S1AP؛ PLMN ID و TAC کافیگ فایل install/etc/open5gs/mme.yaml را تغییر دهید.

$ diff -u /etc/open5gs/mme.yaml.old /etc/open5gs/mme.yaml

--- mme.yaml 2020-09-05 20:52:28.648235143 -0400

+++ mme.yaml.new 2020-09-05 20:56:05.434484208 -0400

@@ -204,20 +204,20 @@

mme:

freeDiameter: /home/acetcom/Documents/git/open5gs/install/etc/freeDiameter/mme.conf

s1ap:

- addr: 127.0.0.2

+ addr: 10.10.0.2

gtpc:

addr: 127.0.0.2

gummei:

plmn\_id:

- mcc: 999

- mnc: 70

+ mcc: 001

+ mnc: 01

mme\_gid: 2

mme\_code: 1

tai:

plmn\_id:

- mcc: 999

- mnc: 70

- tac: 1

+ mcc: 001

+ mnc: 01

+ tac: 2

security:

integrity\_order : [ EIA1, EIA2, EIA0 ]

ciphering\_order : [ EEA0, EEA1, EEA2 ]

برای تنظیم IP آدرس GTP-U کانفیگ فایل install/etc/open5gs/sgwu.yaml را تغییر دهید.

$ diff -u /etc/open5gs/sgwu.yaml.old /etc/open5gs/sgwu.yaml

--- sgwu.yaml 2020-09-05 20:50:39.393022566 -0400

+++ sgwu.yaml.new 2020-09-05 20:51:06.667838823 -0400

@@ -51,7 +51,7 @@

#

sgwu:

gtpu:

- addr: 127.0.0.6

+ addr: 10.11.0.6

pfcp:

addr: 127.0.0.6

اگر فایل های پیکربندی را در حین اجرا شدن دیمون های Open5GS تغییر داده اید، باید آنها را مجدداً راه اندازی کنید.

## راه اندازی Open5GS

برای راه اندازی توابع open5gs ابتدا به مسیر پوشه open5gs رفته و سپس از کامندهای زیر برای راه اندازی استفاده می کنیم.

راه اندازی MME

$ ./install/bin/open5gs-mmed

Open5GS daemon v2.1.0

08/21 22:53:47.328: [app] INFO: Configuration: '/home/acetcom/Documents/git/open5gs/install/etc/open5gs

/mme.yaml' (../src/main.c:54)

08/21 22:53:47.328: [app] INFO: File Logging: '/home/acetcom/Documents/git/open5gs/install/var/log/open5gs

/mme.log' (../src/main.c:57)

08/21 22:53:47.365: [app] INFO: MME initialize...done (../src/mme/app-init.c:33)

08/21 22:53:47.365: [gtp] INFO: gtp\_server() [127.0.0.2]:2123 (../lib/gtp/path.c:32)

08/21 22:53:47.365: [gtp] INFO: gtp\_connect() [127.0.0.3]:2123 (../lib/gtp/path.c:59)

08/21 22:53:47.366: [mme] INFO: s1ap\_server() [127.0.0.2]:36412 (../src/mme/s1ap-sctp.c:57)

راه اندازی SGWC

$ ./install/bin/open5gs-sgwcd

Open5GS daemon v2.1.0

08/21 22:54:43.059: [app] INFO: Configuration: '/home/acetcom/Documents/git/open5gs/install/etc/open5gs

/sgwc.yaml' (../src/main.c:54)

08/21 22:54:43.059: [app] INFO: File Logging: '/home/acetcom/Documents/git/open5gs/install/var/log/open5gs

/sgwc.log' (../src/main.c:57)

08/21 22:54:43.065: [app] INFO: SGW-C initialize...done (../src/sgwc/app.c:31)

08/21 22:54:43.066: [gtp] INFO: gtp\_server() [127.0.0.3]:2123 (../lib/gtp/path.c:32)

08/21 22:54:43.066: [pfcp] INFO: pfcp\_server() [127.0.0.3]:8805 (../lib/pfcp/path.c:32)

08/21 22:54:43.066: [pfcp] INFO: ogs\_pfcp\_connect() [127.0.0.6]:8805 (../lib/pfcp/path.c:60)

راه اندازی SMF

$ ./install/bin/open5gs-smfd

Open5GS daemon v2.1.0

08/21 22:54:56.000: [app] INFO: Configuration: '/home/acetcom/Documents/git/open5gs/install/etc/open5gs

/smf.yaml' (../src/main.c:54)

08/21 22:54:56.000: [app] INFO: File Logging: '/home/acetcom/Documents/git/open5gs/install/var/log/open5gs

/smf.log' (../src/main.c:57)

08/21 22:54:56.050: [gtp] INFO: gtp\_server() [127.0.0.4]:2123 (../lib/gtp/path.c:32)

08/21 22:54:56.050: [app] INFO: SMF initialize...done (../src/smf/app.c:31)

08/21 22:54:56.050: [gtp] INFO: gtp\_server() [::1]:2123 (../lib/gtp/path.c:32)

08/21 22:54:56.050: [pfcp] INFO: pfcp\_server() [127.0.0.4]:8805 (../lib/pfcp/path.c:32)

08/21 22:54:56.050: [pfcp] INFO: pfcp\_server() [::1]:8805 (../lib/pfcp/path.c:32)

08/21 22:54:56.050: [pfcp] INFO: ogs\_pfcp\_connect() [127.0.0.7]:8805 (../lib/pfcp/path.c:60)

08/21 22:54:56.051: [sbi] INFO: sbi\_server() [127.0.0.4]:7777 (../lib/sbi/server.c:298)

راه اندازی amf

$ ./install/bin/open5gs-amfd

Open5GS daemon v2.1.0

08/21 22:55:14.015: [app] INFO: Configuration: '/home/acetcom/Documents/git/open5gs/install/etc/open5gs

/amf.yaml' (../src/main.c:54)

08/21 22:55:14.015: [app] INFO: File Logging: '/home/acetcom/Documents/git/open5gs/install/var/log/open5gs

/amf.log' (../src/main.c:57)

08/21 22:55:14.039: [app] INFO: AMF initialize...done (../src/amf/app.c:33)

08/21 22:55:14.040: [sbi] INFO: sbi\_server() [127.0.0.5]:7777 (../lib/sbi/server.c:298)

08/21 22:55:14.040: [amf] INFO: ngap\_server() [127.0.0.5]:38412 (../src/amf/ngap-sctp.c:56)

راه اندازی sgwu

$ ./install/bin/open5gs-sgwud

Open5GS daemon v2.1.0

08/21 22:54:10.357: [app] INFO: Configuration: '/home/acetcom/Documents/git/open5gs/install/etc/open5gs

/sgwu.yaml' (../src/main.c:54)

08/21 22:54:10.357: [app] INFO: File Logging: '/home/acetcom/Documents/git/open5gs/install/var/log/open5gs

/sgwu.log' (../src/main.c:57)

08/21 22:54:10.360: [pfcp] INFO: pfcp\_server() [127.0.0.6]:8805 (../lib/pfcp/path.c:32)

08/21 22:54:10.360: [app] INFO: SGW-U initialize...done (../src/sgwu/app.c:31)

08/21 22:54:10.361: [gtp] INFO: gtp\_server() [127.0.0.6]:2152 (../lib/gtp/path.c:32)

راه اندازی upf

$ ./install/bin/open5gs-upfd

Open5GS daemon v2.1.0

08/21 22:54:21.596: [app] INFO: Configuration: '/home/acetcom/Documents/git/open5gs/install/etc/open5gs

/upf.yaml' (../src/main.c:54)

08/21 22:54:21.596: [app] INFO: File Logging: '/home/acetcom/Documents/git/open5gs/install/var/log/open5gs

/upf.log' (../src/main.c:57)

08/21 22:54:21.601: [pfcp] INFO: pfcp\_server() [127.0.0.7]:8805 (../lib/pfcp/path.c:32)

08/21 22:54:21.601: [app] INFO: UPF initialize...done (../src/upf/app.c:31)

Building Open5GS from Sources | Open5GS https://open5gs.org/open5gs/docs/guide/02-building-open5gs-from-sources/

5 of 10 6/29/22, 10:25

08/21 22:54:21.601: [gtp] INFO: gtp\_server() [127.0.0.7]:2152 (../lib/gtp/path.c:32)

08/21 22:54:21.601: [gtp] INFO: gtp\_server() [::1]:2152 (../lib/gtp/path.c:32)

راه اندازی hss

$ ./install/bin/open5gs-hssd

Open5GS daemon v2.1.0

08/21 22:57:17.450: [app] INFO: Configuration: '/home/acetcom/Documents/git/open5gs/install/etc/open5gs

/hss.yaml' (../src/main.c:54)

08/21 22:57:17.450: [app] INFO: File Logging: '/home/acetcom/Documents/git/open5gs/install/var/log/open5gs

/hss.log' (../src/main.c:57)

08/21 22:57:17.451: [dbi] INFO: MongoDB URI: 'mongodb://localhost/open5gs' (../lib/dbi/ogs-mongoc.c:99)

08/21 22:57:17.519: [app] INFO: HSS initialize...done (../src/hss/app-init.c:31)

راه اندازی pcrf

$ ./install/bin/open5gs-pcrfd

Open5GS daemon v2.1.0

08/21 22:57:45.894: [app] INFO: Configuration: '/home/acetcom/Documents/git/open5gs/install/etc/open5gs

/pcrf.yaml' (../src/main.c:54)

08/21 22:57:45.894: [app] INFO: File Logging: '/home/acetcom/Documents/git/open5gs/install/var/log/open5gs

/pcrf.log' (../src/main.c:57)

08/21 22:57:45.896: [dbi] INFO: MongoDB URI: 'mongodb://localhost/open5gs' (../lib/dbi/ogs-mongoc.c:99)

08/21 22:57:45.997: [app] INFO: PCRF initialize...done (../src/pcrf/app-init.c:31)

راه اندازی nrf

$ ./install/bin/open5gs-nrfd

Open5GS daemon v2.1.0

08/21 22:56:35.472: [app] INFO: Configuration: '/home/acetcom/Documents/git/open5gs/install/etc/open5gs

/nrf.yaml' (../src/main.c:54)

08/21 22:56:35.472: [app] INFO: File Logging: '/home/acetcom/Documents/git/open5gs/install/var/log/open5gs

/nrf.log' (../src/main.c:57)

08/21 22:56:35.472: [app] INFO: NRF initialize...done (../src/nrf/app.c:31)

08/21 22:56:35.473: [sbi] INFO: sbi\_server() [127.0.0.10]:7777 (../lib/sbi/server.c:298)

08/21 22:56:35.473: [sbi] INFO: sbi\_server() [::1]:7777 (../lib/sbi/server.c:298

راه اندازی ausf

$ ./install/bin/open5gs-ausfd

Open5GS daemon v2.1.0

08/21 22:55:41.899: [app] INFO: Configuration: '/home/acetcom/Documents/git/open5gs/install/etc/open5gs

/ausf.yaml' (../src/main.c:54)

08/21 22:55:41.899: [app] INFO: File Logging: '/home/acetcom/Documents/git/open5gs/install/var/log/open5gs

/ausf.log' (../src/main.c:57)

08/21 22:55:41.900: [app] INFO: AUSF initialize...done (../src/ausf/app.c:31)

08/21 22:55:41.900: [sbi] INFO: sbi\_server() [127.0.0.11]:7777 (../lib/sbi/server.c:298)

راه اندازی udm

$ ./install/bin/open5gs-udmd

Open5GS daemon v2.1.0

08/21 22:56:02.154: [app] INFO: Configuration: '/home/acetcom/Documents/git/open5gs/install/etc/open5gs

/udm.yaml' (../src/main.c:54)

08/21 22:56:02.154: [app] INFO: File Logging: '/home/acetcom/Documents/git/open5gs/install/var/log/open5gs

/udm.log' (../src/main.c:57)

08/21 22:56:02.155: [app] INFO: UDM initialize...done (../src/udm/app.c:31)

08/21 22:56:02.155: [sbi] INFO: sbi\_server() [127.0.0.12]:7777 (../lib/sbi/server.c:298)

راه اندازی pcf

$ ./install/bin/open5gs-pcfd

Open5GS daemon v2.1.0

08/21 22:56:02.154: [app] INFO: Configuration: '/home/acetcom/Documents/git/open5gs/install/etc/open5gs

/pcf.yaml' (../src/main.c:54)

08/21 22:56:02.154: [app] INFO: File Logging: '/home/acetcom/Documents/git/open5gs/install/var/log/open5gs

/pcf.log' (../src/main.c:57)

08/21 22:56:02.155: [app] INFO: PCF initialize...done (../src/pcf/app.c:31)

08/21 22:56:02.155: [sbi] INFO: sbi\_server() [127.0.0.12]:7777 (../lib/sbi/server.c:298)

راه اندازی nssf

$ ./install/bin/open5gs-nssfd

Open5GS daemon v2.1.0

08/21 22:56:02.154: [app] INFO: Configuration: '/home/acetcom/Documents/git/open5gs/install/etc/open5gs

/nssfd.yaml' (../src/main.c:54)

08/21 22:56:02.154: [app] INFO: File Logging: '/home/acetcom/Documents/git/open5gs/install/var/log/open5gs

/nssfd.log' (../src/main.c:57)

Building Open5GS from Sources | Open5GS https://open5gs.org/open5gs/docs/guide/02-building-open5gs-from-sources/

08/21 22:56:02.155: [app] INFO: NSSF initialize...done (../src/nssfd/app.c:31)

08/21 22:56:02.155: [sbi] INFO: sbi\_server() [127.0.0.12]:7777 (../lib/sbi/server.c:298)

راه اندازی bsf

$ ./install/bin/open5gs-bsfd

Open5GS daemon v2.1.0

08/21 22:56:02.154: [app] INFO: Configuration: '/home/acetcom/Documents/git/open5gs/install/etc/open5gs

/bsf.yaml' (../src/main.c:54)

08/21 22:56:02.154: [app] INFO: File Logging: '/home/acetcom/Documents/git/open5gs/install/var/log/open5gs

/bsf.log' (../src/main.c:57)

08/21 22:56:02.155: [app] INFO: BSF initialize...done (../src/bsf/app.c:31)

08/21 22:56:02.155: [sbi] INFO: sbi\_server() [127.0.0.12]:7777 (../lib/sbi/server.c:298)

راه اندازی udr

$ ./install/bin/open5gs-udrd

Open5GS daemon v2.1.0

08/21 22:56:15.810: [app] INFO: Configuration: '/home/acetcom/Documents/git/open5gs/install/etc/open5gs

/udr.yaml' (../src/main.c:54)

08/21 22:56:15.810: [app] INFO: File Logging: '/home/acetcom/Documents/git/open5gs/install/var/log/open5gs

/udr.log' (../src/main.c:57)

08/21 22:56:15.813: [dbi] INFO: MongoDB URI: 'mongodb://localhost/open5gs' (../lib/dbi/ogs-mongoc.c:99)

08/21 22:56:15.813: [app] INFO: UDR initialize...done (../src/udr/app.c:31)

08/21 22:56:15.813: [sbi] INFO: sbi\_server() [127.0.0.13]:7777 (../lib/sbi/server.c:298)

چندین گزینه خط فرمان برای کامندهای راه اندازی ارائه شده است که به شرح زیر می باشد.

$ ./install/bin/open5gs-amfd -h

Usage: ./install/bin/open5gs-amfd [options]

Options:

-c filename : set configuration file

-l filename : set logging file

-e level : set global log-level (default:info)

-m domain : set log-domain (e.g. mme:sgw:gtp)

-d : print lots of debugging information

-t : print tracing information for developer

-D : start as a daemon

-v : show version number and exit

-h : show this message and exit

همچنین می توانید باینری ها را در /usr/bin کپی کنید تا بتوانید آنها را از هر محلی در سیستم اجرا کنید.

/cp open5gs\* /usr/bin

برای راحتی کار، می توانید با استفاده از دستور زیر، تمام توابع را بصورت یکجا اجرا کنید.

$ ./build/tests/app/5gc ## 5G Core Only with ./build/configs/sample.yaml

$ ./build/tests/app/epc -c ./build/configs/srslte.yaml ## EPC Only with ./build/configs/srslte.yaml

$ ./build/tests/app/app ## Both 5G Core and EPC with ./build/configs/sample.yaml

## ایجاد WebUI Open5GS

Node.js برای ساخت WebUI Open5GS مورد نیاز است]14[.

sudo apt install curl

curl -fsSL https://deb.nodesource.com/setup\_14.x | sudo E bash

sudo apt install nodejs

برای اجرای WebUI، پیش نیازهای مورد نیاز را نصب کنید

cd webui

npm ci --no-optional

WebUI به صورت یک اسکریپت npm اجرا می شود.

npm run dev

فراخوانی سرور را می توان با تنظیم متغیر محیطی HOSTNAME یا PORT به صورت زیر تغییر داد.

$ HOSTNAME=192.168.0.11 npm run dev

$ PORT=7777 npm run dev

ثبت اطلاعات مشترکین

به http://127.0.0.1:3000 متصل شوید و با حساب admin وارد شوید.

Username : admin

Password : 1423

برای افزودن اطلاعات مشترکین، می توانید عملیات WebUI را به ترتیب زیر انجام دهید:

به منوی Subscriber بروید.

برای افزودن یک مشترک جدید روی دکمه + کلیک کنید.

IMSI، زمینه امنیتی (K، OPc، AMF) و APN مشترک را پر کنید.

روی دکمه SAVE کلیک کنید

توجه: مشترکین اضافه شده با این ابزار بلافاصله در Open5GS HSS/UDR بدون نیاز به راه اندازی مجدد دیمون ثبت می شوند. با این حال، اگر از WebUI برای تغییر نمایه مشترک استفاده می‌کنید، باید دیمون Open5GS AMF/MME را مجدداً راه‌اندازی کنید تا تغییرات اعمال شوند.

مسیریابی NAT + IP برای اتصال UE به اینترنت U

انتقال IPv4/IPv6 را فعال کنید:

$ sudo sysctl -w net.ipv4.ip\_forward=1

$ sudo sysctl -w net.ipv6.conf.all.forwarding=1

قوانین NAT را اضافه کنید

$ sudo iptables -t nat -A POSTROUTING -s 10.45.0.0/16 ! -o ogstun -j MASQUERADE

$ sudo ip6tables -t nat -A POSTROUTING -s 2001:db8:cafe::/48 ! -o ogstun -j MASQUERADE

# مانیتورینگ شبکه

## مانیتورینگ شبکه

پایش یا مانیتورینگ شبکه (Network Monitoring) یعنی نظارت مستمر بر یک شبکه کامپیوتری جهت شناسایی هرگونه ایراد یا نقص احتمالی و اطمینان از این‌که شبکه درست به کار خود ادامه ‌دهد. مثلا با کمک مانیورینگ شبکه می‌توان وضعیت اجزای شبکه از جمله روترها، سوئیچ‌ها، فایروال‌ها، سرورها و ماشین‌های مجازی را تحت نظر گرفت. اگر عملکرد هر یک از این اجزاء کند یا مختل شود، نرم‌افزار مانیتورینگ شبکه موضوع را به مدیر شبکه اطلاع می‌دهد و در نتیجه مانع از کار افتادن شبکه می‌شود]18[.

یکی از وظایف اصلی که نرم افزارهای مانیتورینگ شبکه به عهده دارند، پایش پهنای باند شبکه و ترافیک‌های جاری بین دستگاه‌‌های موجود می باشد. به بیانی واضح تر، همه نرم افزارهای مانیتورینگ بررسی می‌کنند که آیا یک دستگاه در شبکه بخوبی کار می‌کند یا خیر.

این ابزار پیامی به سمت دستگاه موردنظر ارسال می‌کنه و در صورت مطلوب (OK) بودن وضعیت دستگاه، پیغام پینگ را مبنی بر سلامت دستگاه دریافت می‌کنه. اعلام هشدار، تشخیص مشکلات و پیش بینی وقایع از دیگر وظایف اصلی مانیتورینگ شبکه در سطح عملکردی می باشد.

مانیتورینگ شبکه روش‌های مختلفی دارد که یکی از کارآمدترین آن‌ها استفاده از سیستم یا نرم افزار مانیتورینگ شبکه است.

یکی از وجوه مهم مانیتورینگ شبکه این است که باید کنشگر[[127]](#footnote-127) باشد، نه واکنشگر[[128]](#footnote-128). یعنی فعالانه در جستجوی مشکلات و تنگناهای احتمالی شبکه باشد؛ نه این‌که منتظر بماند تا مشکلی پیش آید و بعد بخواهد آن را شناسایی کند. با این رویکرد می‌توان مشکلات را در همان مرحله اول شناسایی کرد. مانیتورینگ کنشگرانه می‌تواند از قطعی یا معیوب شدن شبکه جلوگیری کند.

### مزایای مانیتورینگ شبکه

شاید بتوان گفت مهم‌ترین مزیت مانیتورینگ شبکه این است که شما را از وضعیت جاری شبکه مطلع می‌کند تا بتوانید بر مبنای آن شبکه‌ را تنظیم یا مشکلات آن را حل کنید. با در نظر داشتن این مزیت کلی می‌توان برخی از مزایای مانیتورینگ شبکه را چنین برشمرد:

کاهش هزینه‌ها

بدون مانیتورینگ شبکه، عملکرد و بازده اجزا و بخش‌های مختلف شبکه از چشم‌تان پنهان می‌ماند. اگر نواقص و ایرادهای شبکه را دیر تشخیص‌ دهید، احتمال قطعی شبکه و خدمات‌ آن افزایش می‌یابد. با کمک سیستم مانیتورینگ شبکه درمی‌یابید که کدام‌یک از عناصر شبکه زیادی یا کدام‌یک کمتر از حد معمول کار می‌کنند. با آگاهی از چنین مواردی می‌توانید جلوی هزینه‌های بی‌جا را گرفته یا تجهیزاتی را که باید تعویض شوند شناسایی شود.

افزایش بازده شبکه

مانیتورینگ شبکه یکی از بهترین راه‌ها برای نظارت بر بازده شبکه و اجزای مختلف آن است. با مانیتورینگ کنشگر (proactive) می‌توان مشکلات را زودتر و پیش از آن‌که بر سازوکارهای شبکه اثر نامطلوب زیادی به جای بگذارند، شناسایی کرد. نرم‌افزارهای مانیتورینگ شبکه شاخص‌های متعددی همچون مصرف پهنای باند، بسته‌های معیوب، تاخیر و… را بی‌درنگ اندازه‌ می‌گیرند. ارائه گزارش‌های دقیق در این خصوص، به گروه مدیریت شبکه کمک می‌کند تا راحت‌تر به علت بروز مشکل پی ببرند.

کمک به ارتقای امنیت

وظیفه اصلی نرم‌افزارهای مانیتورینگ شبکه، تامین امنیت نیست اما کاری که انجام می‌دهند می‌تواند به بهبود امنیت شبکه‌ کمک ‌کند. برای مثال، اگر مانیتورینگ ترافیک شبکه گزارش دهد که ترافیک شبکه‌ غیرعادی و ناگهانی افزایش یافته یا دستگاه ناآشنایی به شبکه‌ متصل شده است، می‌توان آن را نشانه‌ای دال بر حملات سایبری تلقی کرد. ضمنا اگر شبکه درست مدیریت شود، کاربران را از دسترسی اتفاقی به وب‌سایت‌های آلوده و مضر برای شبکه بازمی‌دارد.

تسریع عیب‌یابی

با کمک نرم‌افزارهای مانیتورینگ شبکه، زمان موردنیاز برای عیب‌یابی شبکه کاهش چشمگیری می‌یابد. نرم‌افزار مانیتورینگ درباره بازده عادی شبکه‌ داده‌های شفافی ارائه می‌دهد که نواقص را مشخص می‌کند و شما را در ریشه‌یابی مشکل و علت بروز آن بسیار یاری می‌دهد. با مانیتورینگ شبکه، مشکلات احتمالی در اجزای شبکه (مثل سرور) به محض وقوع، به مدیر گزارش می‌‌شود و او سریع‌تر می‌تواند علت بروز مشکل را پیدا کند. با استفاده از مانیتورینگ شبکه به‌راحتی می‌توان گفت که چه زمانی باید جزء خاصی از شبکه را تعویض یا تنظیم کرد.

**مانیتورینگ سخت افزار های شبکه**

یک شبکه شامل بسیاری از دستگاه های سخت افزاری است، مانند دستگاه هایی که برای مسیریابی و سوئیچینگ، ذخیره سازی، اتصال، سرورهای برنامه و غیره استفاده می شوند. سخت افزار ستون فقرات کل زیرساخت IT را تشکیل می دهد. اگر یک سخت افزار حیاتی برای عملیات روزمره شبکه خراب شود، این امر منجر به خرابی کل شبکه می شود. به عنوان مثال، منبع تغذیه معیوب در سوئیچ هسته ای یا گرم شدن بیش از حد روتر لبه ای می تواند باعث اختلال در شبکه شود. برای اطمینان از عملکرد روان شبکه، نظارت بر سلامت و عملکرد دستگاه های سخت افزاری در شبکه مهم است.

برای درک جزئیات مربوط به سلامت سخت افزار، معیارهای مختلفی وجود دارد که باید بررسی شوند. در ادامه به چند معیار مهم و اینکه چرا باید آنها را کنترل کرد اشاره می شود:

CPU: وظایف دستگاه توسط پردازنده آن انجام می شود. اگر بهره گیری از پردازنده به حداکثر مقدار خود برسد، عملکرد دستگاه می تواند دچار اختلال شود.

دما: هنگام انجام فعالیت ها، استفاده از پردازنده دستگاه نیز می تواند افزایش یابد. این به نوبه خود می تواند دما را افزایش دهد. افزایش دما می تواند باعث سوء عملکرد دستگاه شود و در نتیجه شبکه را دچار اختلال کند.

سرعت و وضعیت فن: دما و عملکرد فن با هم مرتبط هستند. نظارت بر سرعت فن به اطمینان از عملکرد مناسب فن کمک می کند و حتی تعادل را در خنک سازی حفظ می کند و دمای دستگاه را در حد مطلوب خود نگه می دارد.

وضعیت منبع تغذیه: منبع تغذیه معیوب یا افزایش جریان در دستگاه باعث سوء عملکرد آن شده و در نهایت منجر به خرابی می شود. نظارت با توجه به هشدارها بر اساس آستانه ها به مدیر کمک می کند مشکلات احتمالی را پیدا کند.

**نظارت بر دیسک:**

داده ها یا اطلاعات یکی از مهم ترین منابع یک سازمان است. سازمان ها برای برنامه ریزی تجاری و همچنین عملکرد صحیح به داده نیاز دارند. داده های مورد نیاز یک سازمان نیز باید برای استفاده از سوابق یا برای استفاده های بعدی ذخیره شود. در شرکت ها، داده ها در آرایه های ذخیره سازی که دارای چندین دیسک هستند، جمع آوری و ذخیره می شوند. هر مسئله ای که بر روی دیسک ها یا آرایه های ذخیره سازی که داده های تجاری را ذخیره می کند، بوجود می آید، می تواند پیامدهای جدی برای تداوم تجارت داشته باشد.

نظارت بر دیسک شامل مدیریت مناسب فضای دیسک برای استفاده موثر از فضا، نظارت بر عملکرد دیسک برای کشف خطاها، آمار پرونده های بزرگ، فضای خالی و تغییر در میزان استفاده از فضای دیسک، عملکرد ورودی و خروجی و غیره است.

نظارت به مدیران اجازه می دهد تا از قبل برای ارتقا سیستم و همچنین فضا، تشخیص مشکلات مربوط به ذخیره سازی و کاهش خرابی در صورت بروز مشکل برنامه ریزی کنند.

### مانیتورینگ شبکه چگونه انجام می‌شود

مانیتورینگ شبکه برای آزمودن دسترسی‌پذیری (availability) و بازده (performance) شبکه از تکنیک‌های مختلفی بهره می‌برد. داده‌هایی که نرم‌افزار مانیتورینگ شبکه برای بررسی و تحلیل وضعیت شبکه نیاز دارد، به‌وسیله این تکنیک‌ها گردآوری می‌شود. برخی از رایج‌ترین تکنیک‌های جمع‌آوری داده برای نرم‌افزارهای مانیتورینگ شبکه عبارتند از]19[:

**Ping:**

پینگ از ابتدایی‌ترین تکنیک‌هایی است که نرم‌افزار مانیتورینگ برای ارزیابی هاست‌های درون یک شبکه از آن بهره می‌برد. در این شیوه، سیستم‌ مانیتورینگ، به دستگاهی که باید پایش شود سیگنال می‌فرستند و اطلاعات مدنظرش را بر همین اساس دریافت و ثبت می‌کند (مثلا بررسی می‌کند ‌که آیا سیگنال ارسال شده، در مقصد دریافت شد یا نه؛ یا چه مدت طول کشید تا سیگنال به مقصد برسد و آیا سیگنالی از دست رفت یا نه). سیستم مانیتورینگ شبکه سپس با تحلیل این داده‌ها مواردی همچون فعال یا غیرفعال بودن هاست، میزان بازده آن، زمان لازم برای انتقال سیگنال، و از دست رفتن بسته‌های داده هنگام ارتباط با هاست و… را درمی‌یابد.

**SNMP:**

پروتکل مدیریت آسان شبکه یا SNMP تجهیزات متصل به شبکه را از طریق نرم‌افزار مانیتورینگ بررسی می‌کند. برای این منظور، همه تجهیزات باید نرم‌افزار مانیتورینگ را نصب شده داشته باشند تا بتوانند عملکرد خود را به مدیر شبکه گزارش دهند. مدیر شبکه این اطلاعات را در پایگاه‌داده‌ای گردآوری و آن‌ را تحلیل می‌کند تا خطاهای احتمالی را کشف کند. پروتکل SNMP پرکاربردترین پروتکل در سیستم‌های مدرن مدیریت شبکه است.

**Syslog:**

هرگاه اتفاقی رخ دهد که بر عملکرد یکی از تجهیزات تحت شبکه تاثیر بگذارد، سیستم مدیریت خودکار Syslog به مسئولان مربوطه پیغام‌ می‌فرستد. تکنسین‌های شبکه می‌توانند تجهیزات را طوری تنظیم کنند که در صورت بروز خطا، قطعی‌های غیرمنتظره، نقص در پیکربندی و… موضوع را از طریق Syslog به مدیر شبکه اطلاع دهند. این پیغام‌ها اغلب حاوی اطلاعاتی هستند که برای سیستم‌های مدیریت شبکه و سیستم‌های امنیتی مفیدند.

**Script:**

اگر در شبکه‌ای به‌رغم استفاده از نرم‌افزار مانیتورینگ شبکه، هنوز شکاف‌هایی باقی مانده بود، اسکریپت‌ها می‌توانند شکاف‌های کوچک را پر کنند. اسکریپت‌ها برنامه‌های ساده‌ای هستند که اطلاعات ابتدایی جمع‌آوری می‌کنند و به شبکه دستور می‌دهند تا تحت شرایط مشخص، کار خاصی انجام دهد. ریست‌ شدن دستگاهی خاص طبق زمان‌بندی مشخص و پیکربندی مکرر و هر روزه یک کامپیوتر از رایج‌ترین مثال‌هایی است که می‌توان با کمک اسکریپت‌ها انجام داد. البته اسکریپت‌ها می‌توانند برای سیستم مانیتورینگ شبکه نیز داده جمع‌آوری کنند.

### انواع مانیتورینگ شبکه

مانیتورینگ شبکه به یکی از دو شیوه زیر یا ترکیبی از هر دو صورت می‌پذیرد]19[:

* Agent monitoring یا مانیتورینگ ایجنت‌محور
* Agentless monitoring یا مانیتورینگ بدون ایجنت

**Agent monitoring یا مانیتورینگ ایجنت‌محور :**

ایجنت (Agent) را می‌توان «عامل نرم‌افزاری» معنا کرد. در حوزه مانیتورینگ شبکه، ایجنت، نرم‌افزار یا اپلیکیشن کوچکی است که روی تجهیزات متصل به شبکه نصب می‌شود تا آن تجهیزات بتوانند وضعیت خود را به نرم‌افزار مانیتورینگ شبکه اطلاع دهند. یکی از مهم‌ترین مزایای ایجنت این است که می‌تواند درباره تجهیزاتی که پایش می‌شوند داده‌های دقیق‌تری گرد آورد و درنتیجه کیفیت مانیتورینگ‌‌، هشدارها و گزارش‌ها را ارتقا دهد.

در روش مانیتورینگ با ایجنت (Agent monitoring) هر دستگاه متصل به شبکه برای این‌که پایش شود، باید ایجنت را نصب شده داشته باشد. داده‌های گردآوری شده توسط ایجنت، به ایستگاه مانیتورینگ ارسال می‌شود. بهترین ایجنت، ایجنتی است که داده‌های موردنیاز نرم‌افزار مانیتورینگ را برای آن بفرستد و نه هر داده‌‌ای را. در این‌صورت ایجنت اصطلاحا سبک خواهدبود (یعنی منابع زیادی مصرف نمی‌کند). ایجنت سبک می‌تواند درباره شاخص‌های جزئی نیز داده گرد آورد و به ارتقای پایش‌ها، گزارش‌ها و هشدارها و عمیق‌تر شدن تحلیل‌ها و عیب‌یابی‌های شبکه کمک کند.

اما بعضی از ایجنت‌ها بسیار سنگین هستند (یعنی زیاد از منابع شبکه استفاده می‌کنند). ایجنت‌های سنگین به سرورهایی که باید پایش‌ شوند، فشار وارد می‌آورند. ایجنتی که سیاست‌های کاریش را خودش (و نه ایستگاه مانیتورینگ) مدیریت می‌کند، ممکن است بار کاری مضاعفی روی دوش سرورهای کلاینت بگذارد و بازده کلی سرورها و سرویس‌های مربوطه را کاهش دهد. بسیاری از نرم‌افزارهای مدیریت و مانیتورینگ شبکه مثل Zabbix، IBM Tivoli، HP Openview، BMC، Patrol و CA Unicenter از مدل ایجنت بهره می‌برند. گاهی سنگینی بیش از حد ایجنت سبب می‌شود کارکرد عکس بیابد؛ یعنی این شیوه مانیتورینگ به‌جای آن‌که به بهبود بازده شبکه کمک کند، خود موجب مصرف زیادی منابع شبکه و کاهش بازده سرورها شود. پس در صورت انتخاب مانیتورینگ ایجنت‌محور باید از ایجنت‌های سبک بهره برد که می‌توانند درباره شاخص‌های جزئی‌تر نیز داده‌های عمیقی جمع‌آوری کنند اما به سرورها فشار نمی‌آورند.

**Agentless monitoring یا مانیتورینگ بدون ایجنت**

نرم‌افزارهای مانیتورینگ بدون ایجنت (Agentless monitoring)، روی تجهیزات شبکه، عامل نرم‌افزاری (ایجنت) خاصی نصب نمی‌کنند و به‌جای آن از رابط‌های صنعتی استاندارد بهره می‌برند. همین موضوع سبب می‌شود تا نرم‌افزارهای مانیتورینگ بدون ایجنت (Agentless) نسبت به نرم‌افزارهای ایجنت‌محور، سبک‌تر باشند. ضمنا تنظیم و به‌کارگیری سیستم‌های بدون ایجنت، آسان‌تر از سیستم‌های ایجنت‌محور است. اما در عوض، نرم‌افزارهای مانیتورینگ بدون ایجنت، بیشتر بر شاخص‌های مهم و کلیدی تمرکز دارند و داده‌های‌شان به‌اندازه سیستم‌های ایجنت‌محور، جزء‌نگر و عمیق نیست.

مانیتورینگ بدون ایجنت به دو روش انجام می‌شود: یا با کمک API راه دور؛ یا با تحلیل مستقیم و بی‌واسطه بسته‌های شبکه که بین اجزا جریان دارند.

### نرم افزار‌های مانیتورینگ شبکه

نرم افزار مانیتورینگ شبکه یک ابزار مانیتورینگ شبکه است که محیطی را برای ما فراهم می‌کند تا به صورت کلی تمام شبکه خود همراه تجهیزات و نحوه استفاده از پهنای باند را مشاهده کنیم. در این نرم افزار می‌توان دستگاه‌ها را برای پایش اضافه و یا حذف کرد همچنین برای پورت‌ها محدودیت پهنای باند تعریف کرد، یکسری فعالیت از پیش تعریف شده ایجاد کرد تا در هنگام بروز خطا اجرا شوند و بسیاری موارد دیگر که برای مدیران آی‌تی بسیار مفید است.

در زیر برخی نرم افزار های مانیتورینگ شبکه نام برده شده است:

* [zabbix](https://www.zabbix.com/)
* Graylog
* Prometheus
* [solarwinds](https://www.solarwinds.com/)
* [PRTG](https://www.paessler.com/prtg)
* [ManageEngine](https://www.manageengine.com/)
* و …

هر کدام از این نرم افزار‌ها مزیت‌های مخصوص به خود را دارند که در ادامه به بررسی سه مورد اول که میتواند در مدیریت مانیتورینگ شبکه 5G بسیار پرکاربرد باشد خواهیم پرداخت.

## زبیکس

زبیکس (Zabbix) یک نرم افزار متن باز برای پایش شبکه‌ها و برنامه‌هاست که با به کارگیری پارامترهای متعدد تمامی اطلاعات مربوط به این شبکه‌ها و برنامه‌ها را برای شما جمع آوری می‌کند. این سیستم می‌تواند در هر لحظه به طور همزمان به پایش هزاران سرور و تجهیزات شبکه بپردازد. در ادامه در مورد سیستم مانیتورینگ Zabbix و ویژگی‌ها و امکانات آن صحبت خواهیم کرد]16[.

کمپانی Zabbix در سال ۲۰۰۵ و توسط الکسی ولادیشو (Alexei Vladishev) تأسیس گردید. او ایده راه حل مانیتورینگ خود را که پیشتر به صورت نرم افزاری درون سازمانی در سال ۱۹۹۸ ارائه داده بود، به عنوان جزوی از اهداف راهبری این شرکت دنبال نمود و طولی نکشید که توانست در این زمینه به موفقیت‌های چشمگیری دست پیدا کند. اولین نسخه آزمایشی این نرم افزار در سال ۲۰۰۱ منتشر شد و نخستین بار به طور رسمی در سال ۲۰۰۴ از آن بهره برداری گردید.

Zabbix تا کنون بیش از ۱۳ نسخه بروزرسانی شده را به بازار عرضه کرده است که آخرین مورد آن به تاریخ ۲۰ می ۲۰۲۱ تحت عنوان [ZABBIX5.4](https://sadidafarin.ir/articles/what-is-zabbix-5-4)   بازمی‌گردد.

زبیکس یک نرم افزار قدرتمند در زمینه مانیتورینگ و جمع اوری اطلاعات در شبکه میباشد. با استفاده از این نرم افزار میتوانید بصورت Real-Time بیش از 10 هزار سرور، ماشین مجازی و دیگر سخت افزار های شبکه ای را هم زمان مانیتور کنید. Zabbix در کنار جمع اوری اطلاعات، با استفاده از یک رابط کاربری مناسب انواع Graph ها و Map ها را در اختیار شما قرار میدهد تا به بهترین شکل تمامی سخت افزار های مورد نظر را زیر نظر داشته باشید. کارایی بسیار قدرتمند این ابزار در جمع اوری اطلاعات و انالیز، آن را برای سازمان های بزرگ مناسب میسازد. مانیتورینگ توضیع شده نیز با استفاده ازZabbix Proxies امکانپذیر میباشد.

### معماری

سرور زبیکس، هسته مرکزی در نرم افزار زبیکس بوده و وظیفه ی مانیتورینگ، تبادل اطلاعات با Proxy Zabbix ها وAgent ها، محاسبه ی حدود آستانه و ارسال هشدارها را در کنار نقش Repository مرکزی اطلاعات بر عهده دارد.

سرور زبیکس، یک Repository مرکزی است که تمامی تنظیمات، آمارها و داده های قابل استفاده در آن ذخیره می شوند. ماهیت زبیکس به این صورت است که به محض وقوع مشکل برای هر یک از سیستم های مانیتور شده، هشداری را به صورت خودکار برای Administrator ها ارسال می نماید.

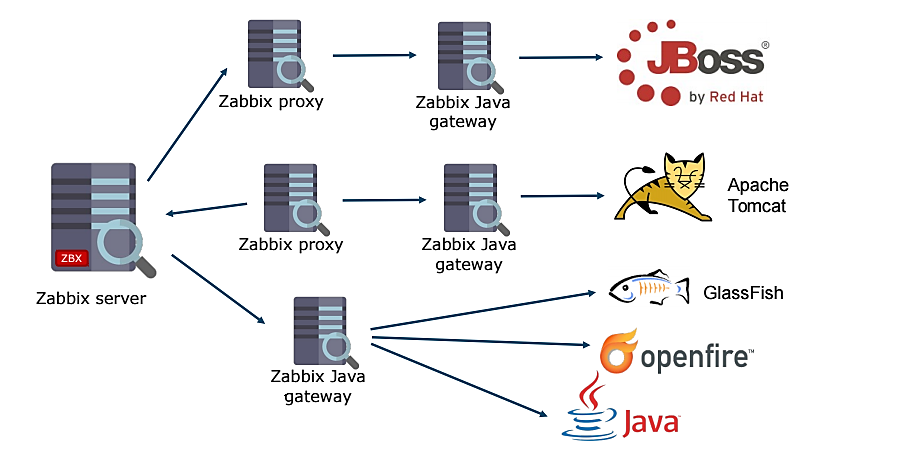
**معرفی zabbix agent ها**

ابزاری است جهت مانیتورینگ پارامترهایمختلف مانند: درایوهای ذخیره سازی، حافظه، وضعیت پردازنده،شبکه، فایل سیستم ها و بسیاری موارد دیگر.

Zabbix Agent اطلاعات قابل استفاده را جمع آوری کرده و آنها را جهت پردازش های بعدی به سرور زبیکس ارسال می نماید. Zabbix Agent به دلیل استفاده از Call System هرسیستم عامل از بهره وری بالایی برخوردار است.

**معرفی zabbix proxy**

یک Zabbix Proxy داده ها را از دستگاه ها به جای سرور زبیکس جمع آوری می کند.در این حالت Zabbix Proxy می تواند مقداری از باری که بر روی سرور زبیکس قرار دارد را کم نموده و کارایی مناسب تری را فرآهم آورد. همین طور استفاده از Zabbix Proxy مناسب ترین راه جهت پیاده سازی مانیتورینگ توزیع شده در عین مدیریت متمرکز است.



شکل 6-1: نمایی از رویکرد ارتباط سرور زبیکس با اجزای مختلف شبکه]16[.

این نرم افزار با یک رابط کاربری تحت وب ارائه شده است که این محیط برای ورود با اعتبار سنجی امن شده است. در این محیط میتوانید کاربرانی با سطوح دسترسی مختلف ایجاد نمایید. جمع اوری اطلاعات با استفاده از نصب Agent در سیستم میزبان انجام میشود، البته متد Agent-Less نیز در این نرم افزار فراهم بوده و میتوان بدون Agent نیز سرور و سخت افزار ها را مانیتور کرد. Zabbix میتواند بصورت اتوماتیک تمامی سخت افزار های شبکه را شناسایی کند. با استفاده از امکان Network Discovery میتواند بصورت اتوماتیک تمامی سخت افزار های درون شبکه را شناسایی و در صورت تمایل ان را مانیتور نمایید.

زبیکس از مای اس کیوال، پست گرس کیوال، اس کیوال لایت، Oracle و دی بی۲ برای ذخیره داده ها پشتیبانی میکند. برنامه نویسی سمت سرور از زبان C بهره میبرد و برنامه نویسی سمت کاربری آن از زبان PHP استفاده می کند.زبیکس همراه با ذخیره سازی اطلاعات ویژگی های مانند گراف ها ، نقشه ها و … را با هدف آنالیز و هشدار را در اختیار مدیران شبکه میگذارد، که در صورت رد شدن محدودیت هایی که در شبکه اعمال شده مدیر شبکه را از طریق ایمیل ، SMS یا تلفن آگاه سازد.

### امکانات تخصصی و منحصر به فرد زبیکس

این نرم افزار به عنوان یکی از بهترین سیستم های مانیتورینگ در جهان شناخته می‌شود و از امکانات و ویژگی‌های تخصصی و منحصر به فردی برخوردار بوده که آن را در مقابل سایر رقیبان خود در این زمینه سرآمد ساخته‌ است. در ادامه به معرفی این ویژگی‌ها و امکانات می‌پردازیم.

**مانیتورینگ فعال**

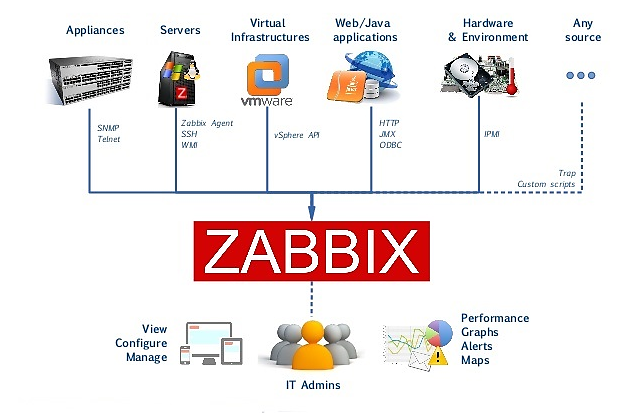
سیستم مانیتورینگ Zabbix به شما کمک می‌کند که هزینه‌های عملیاتی و Downtime سرویس‌های تجاریی خود را کاهش دهید. زبیکس با استفاده از ایمیل یا اس ام اس یا راه‌های دیگر هر رویدادی را در سیستم به اطلاع شما می‌رساند. همچنین شما می‌توانید سرویس‌های هشدار مدنظر خود را بر روی زبیکس پیاده سازی کنید و به این ترتیب کیفیت سیستم خود را ارتقا بخشید.

**مانیتورینگ در سطح سازمانی**

طراحی الگوریتم‌های هوشمند Zabbix به گونه‌ای است که امکان پایش سازمانی شبکه‌های گسترده با هزاران دستگاه متصل را برای کاربران خود فراهم می‌آورد. در حقیقت سرعت عمل این نرم افزار در حد بیش از ۳ میلیون بررسی در دقیقه تخمین زده می‌شود. این سیستم‌ها می‌توانند روزانه چندین گیگابایت داده را جمع آوری کنند. این موضوع کارایی این نرم افزار را به عنوان یکی از شناخته شده ترین سیستم‌های پایش در رتبه بندی بسیار بالایی قرار می‌دهد.

**امکان مانیتور کردن هر چیزی**

یکی از امکانات منحصر به فرد نرم افزار Zabbix این است که شما می‌توانید از آن برای پایش هر سیستم یا شبکه‌ای استفاده کنید. به عبارتی محدودیتی از نظر پشتیبانی برای این ابزار مانیتورینگ وجود ندارد. این سیستم بر روی پلتفرم‌های ویندوز، یونیکس و لینوکس قابل نصب بوده و به علت قابلیت پردازش مستقل نیاز به محیط‌هایی چون dot NET یا جاوا نخواهد داشت. در زیر تعدادی از موارد کاربرد زبیکس را برمی‌شماریم:



شکل 6-2: نمایی از مانیتورینگ سخت افزار و نرم افزارهای مختلف توسط زبیکس]16[.

مانیتورینگ VMware :

Zabbix با برخورداری از قابلیت «جستجوی عمیق» این امکان را برای کاربران متعدد خود فراهم آورده است که به آسانی آمار و داده‌های مرتبط با سیستم‌های مجازی VMware چون VMware vCenter یا vSphere را تحت پایش قرار دهند.

مانیتورینگ دیتابیس:

مانیتورینگ دیتابیس یکی دیگر از خصوصیات منحصر به فرد برنامه Zabbix به شمار می‌آید. همه شرکت‌ها اطلاعات مهم خود از قبیل اطلاعات مربوط به امور مالی، امور مشتریان و اطلاعات کارمندان خود را در دیتابیس‌های ویژه‌ ذخیره و نگهداری می‌کنند. بنابراین این دیتابیس‌ها جزئی لاینفک در سیستم فناوری اطلاعات شرکت‌ها به شمار می‌آید. Zabbix از طیف گسترده‌ای از دیتابیس‌ها از جمله MySQL، Oracle، PostgreSQL و SQL Server پشتیبانی می‌کند و قادر است جزئیات این دیتابیس‌ها را برای شما مانیتور کند.

مانیتورینگ تجهیزات شبکه:

پروتکل SNMP جزئی جدانشدنی از تجهیزات شبکه به شمار می‌آید. Zabbix با داشتن امکان پشتیبانی از این پروتکل می‌تواند به پایش داده‌های مرتبط با تجهیزات شبکه، سیستم‌های برق، سیستم‌های سرمایشی و هرگونه سیستم دیگری که از این پروتکل استفاده می‌کند بپردازد.

مانیتورینگ بدون عامل نرم افزاری:

در اکثر موارد Zabbix با استفاده از Agent (عامل نرم افزاری) مانیتورینگ بر روی سیستم‌ها را انجام می‌دهد. اما گاهی اوقات ممکن است برخی سیستم‌ها از Agent پشتیبانی نکنند و بنابراین امکان پایش آن‌ها از این راه وجود نداشته باشد. با این حال Zabbix برای این مورد نیز راه حلی اندیشیده است و می‌تواند بدون نیاز به Agent عمل مانیتورینگ را بر روی سرویس‌هایی چون Web Server، FTP، SSH، SMTP، POP و IMAP انجام دهد.

سفارشی سازی:

گاهی اوقات برخی داده‌ها برای کسانی که آن‌ها را مدیریت می‌کنند، از اهمیت بالاتری برخوردار بوده و در نتیجه ضرورت پایش آن‌ها بیشتر مطرح خواهد بود. اگر با چنین داده‌هایی سر و کار دارید، Zabbix انتخاب مناسبی برای شما خواهد بود. شما می‌توانید به کمک این برنامه پایش، مانیتورینگ سیستم‌های امور مالی یا سیستم‌های دیگر خود را به صورت سفارشی و بدون محدودیت انجام دهید.

مانیتورینگ سرورهای جاوا:

زبیکس با پشتیبانی از سرورهای جاوا می‌تواند نرم افزارهای مبتنی بر این پلتفرم‌ را مورد پایش قرار دهد. تکنولوژی جدیدی که Zabbix از نسخه دوم به بعد آن را به سرویس خود اضافه کرده است Zabbix Java Gateway نام دارد و کاربران را قادر می‌سازد که سرورهای جاوا را تحت مانیتورینگ قرار دهند.

مانیتورینگ وب سرویس ها:

اگرچه مانیتور کردن دیتابیس‌ها اطلاعات زیادی را به ما ارائه می‌دهد؛ اما این کافی نیست و گاهی اوقات ممکن است علی رغم پایش دیتابیس‌ها سرویس مورد نظر به درستی کار نکند. یک عامل مهم که کارایی این سرویس‌ها را به طور دقیق‌تری تضمین می‌کند، وضعیت وب سرویس‌های آن‌ها در شبکه اطلاعات است. Zabbix با پشتیبانی از مانیتورینگ داخلی وب می‌تواند به سادگی دسترسی افراد، مدت زمان پاسخگویی و سرعت دانلود، وضعیت فروشگاه و یا سیستم پشتیبانی آنلاین را در خصوص شرکت‌های مختلف مورد پایش قرار دهد.

مانیتورینگ سخت افزاری:

Zabbix علاوه بر پشتیبانی از نرم افزارهای مختلف، قابلیت پایش سخت افزارها را نیز داراست. به عبارت دیگر این سیستم مانیتورینگ می‌تواند بدون هیچ گونه واسطه‌ای به طور مستقیم به استخراج اطلاعات سخت افزارهایی که دارای قابلیت IPMI هستند بپردازد.

به این ترتیب شما می‌توانید به طور همزمان اطلاعات مربوط به وضعیت هارد دیسک، سی پی یو، درجه حرارت، عملکرد فن‌ها و دیگر بخش‌های سخت افزار سیستم‌های خود را دریافت کنید. این قابلیت حتی در زمانی که سیستم خاموش است نیز عمل می‌کند. به علاوه از طریق این ویژگی، امکان کنترل از راه دور و تحت شبکه سیستم عامل و خاموش یا روشن کردن آن نیز وجود خواهد داشت.

مانیتورینگ سیستم های ابری :

از جمله دیگر قابلیت‌های سیستم مانیتورینگ Zabbix امکان پایش وضعیت سیستم‌های ابری است. به این ترتیب می‌توانید به اطلاعات ذخیره شده در شبکه کلاود نیز دسترسی پیدا کنید.

مانیتورینگ اینترنت اشیاء :

شبکه‌های IOT (اینترنت اشیاء) از چندین و چند سنسور گوناگون تشکیل شده‌اند که به صورت لحظه‌ای اطلاعات محیط پیرامون خود را دریافت می‌دارند. Zabbix می‌تواند اطلاعات شبکه‌های اینترنت اشیاء را مورد پایش قرار داده و این اطلاعات را برای شما جمع آوری کند.

### متن باز و بدون لایسنس

متن باز بودن و عدم نیاز به لایسنس از جمله ویژگی‌های منحصر به فرد نرم افزار مانیتورینگ Zabbix به شمار می‌آیند. شما می‌توانید هر تعداد دستگاه را با استفاده از این نرم افزار به طور کاملاً رایگان تحت پایش قرار دهید. همچنین متن باز بودن زبیکس این قابلیت را به شما می‌دهد که خودتان کدهای این نرم افزار را مطابق با نیازهایتان تغییر دهید و به طور سفارشی به مانیتورینگ سیستم‌ها و شبکه‌ها بپردازید.

### برنامه ریزی بهتر برای استفاده از منابع

نرم افزار مانیتورینگ Zabbix امکان تحلیل اطلاعات و منابع موجود را به شما اعطا خواهد کرد. این ویژگی از اهمیت بسزایی برخوردار است؛ چراکه تهیه تجهیزات جدید ممکن است هفته‌ها به طول بیانجامد و در این مدت بخش آی تی سازمان شما باید برنامه ریزی دقیقی از چگونگی استفاده از منابع موجود داشته باشد.

Zabbix امکان پایش و رصد کردن این داده‌ها را به شما می‌دهد. به این ترتیب می‌توانید از وقوع حوادث غیرمترقبه‌ای چون استفاده بیش از حد از انرژی و اینترنت و یا اتمام فضای دیسک جلوگیری کنید. همچنین می‌توانید از میزان هدر رفت‌های سیستم آگاه شوید و از این داده‌ها جهت برنامه ریزی‌های بهتر در آینده استفاده کنید.

### روند توسعه و نسخه های زبیکس

از سال ۲۰۰۱ تا کنون، هر نسخه از برنامه Zabbix با فاصله حدود شش ماه از نسخه قبلی منتشر شده است. تیم فنی شرکت Zabbix در نظر دارد که به طور مرتب هر شش ماه یکبار یک نسخه جدید و پایدار از این نرم افزار را در اختیار کاربران خود قرار دهد. جدیدا در آخرین به روز رسانی خود نسخه ۵.۴ را رونمایی کرد. همچنین این شرکت دارای آینده نگری و برنامه‌‎ریزی مخصوصی است که شما میتوانید مسیری از به روز رسانی نسخه زبیکس 5 تا نسخه 7 آن را مشاهده کنید:

* [Zabbix 5.0 LTS](https://sadidafarin.ir/articles/آخرین-نسخه-زبیکس-zabbix-5-0-lts-منتشر-شد)
* [Zabbix 5.2](https://sadidafarin.ir/articles/بررسی-نسخه-جدید-زبیکس،-زبیکس-۵-۲)
* [Zabbix 5.4](https://sadidafarin.ir/articles/what-is-zabbix-5-4)
* Zabbix 6.0 LTS
* Zabbix 6.2
* Zabbix 6.4
* Zabbix 7.0 LTS

## Log Management و معرفی ابزار های محبوب در این حوزه

مدیریت لاگ، عبارت است از جمع‌آوری، ذخیره، پردازش، ترکیب و تحلیل داده‌ها از برنامه‌های کاربردی متفاوت به منظور بهینه‌سازی عملکرد سیستم، شناسایی مشکلات فنی، مدیریت بهتر منابع، تقویت امنیت و بهبود انطباق]20[.

لاگ یک فایل تولید شده توسط کامپیوتر است که فعالیت های درون سیستم عامل یا برنامه های نرم افزاری را ثبت می کند. فایل لاگ، به طور خودکار هر گونه اطلاعات طراحی شده توسط مدیران سیستم از جمله پیام ها، گزارش های خطا، درخواست های فایل و انتقال فایل را مستند سازی می کند، این فعالیت همچنین دارای زمان ثبت است که به متخصصان فناوری اطلاعات و توسعه دهندگان کمک می کند تا بفهمند چه اتفاقی و در چه زمانی رخ داده است.

فعالیت هایی که در مدیریت لاگ انجام می شود را می توان به دسته های زیر تقسیم بندی کرد:

جمع آوری

جمع آوری اطلاعات از سیستم عامل، برنامه ها، سرورها، کاربران، نقاط پایانی یا هر منبع مرتبط دیگری در سازمان.

مانیتورینگ

ردیابی رویدادها و فعالیت ها و همچنین زمان وقوع آنها.

تحلیل

مجموعه لاگ ها را از سرور گزارش مرور می‌کند تا به طور فعال باگ‌ها، تهدیدات امنیتی یا سایر مسائل را شناسایی کند.

نگهداری

تعیین می‌کند داده‌های گزارش چه مدت باید در فایل گزارش نگهداری شوند.

indexing و جستجو

به سازمان کمک می‌کند تا داده‌ها را در همه لاگ ها، فیلتر، مرتب‌سازی، تجزیه و تحلیل و یا جستجو کند.

گزارش

گزارش‌دهی از گزارش حسابرسی (Audit Log) را به‌عنوان ارزیابی عملیاتی، تخصیص منابع، امنیت یا انطباق با مقررات، خودکار می‌کند.

### **ابزارهای مدیریت لاگ**

در این بخش، می‌خواهیم دو راه‌حل محبوب متن باز را که برای ساده‌سازی فرآیند مدیریت گزارش‌ها استفاده می‌ شوند معرفی و با هم مقایسه کنیم: Graylog در مقابل ELK (Elasticsearch + Logstash + Kibana).

#### **ابزار ELK**



شکل 6-3: سرویس های مورد استفاده در ابزار ELK ]20[.

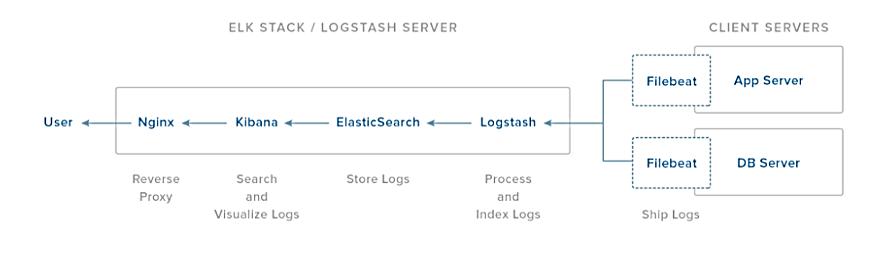
[ELK](https://www.elastic.co/what-is/elk-stack) ترکیبی از سه سرویس جداگانه است که همه آن ها متن باز هستند و توسط یک تیم توسعه یافته اند.

Elasticsearch یک موتور جستجوی بسیار قدرتمند و مقیاس پذیر است که می تواند حجم زیادی از داده را ذخیره کند و به عنوان یک خوشه مورد استفاده قرار دهد.

Logstash ابزاری برای واکشی (Fetch) داده ها از/به یک مکان خاص است. دارای تنوع گسترده ای از پلاگین ها و یک جامعه کاربری بزرگ است.

Kibana یک رابط کاربری گرافیکی است که به شما امکان جستجو، تجزیه و تحلیل و مصورسازی مقادیر زیادی از داده های پیچیده از پایگاه داده Elasticsearch را می دهد. فرآیند استقرار، بیش از 5 دقیقه طول نمی کشد.

در بیشتر موارد، پشته ELK از Filebeat استفاده می کند. هدف این ابزار، تحویل لاگ ها به یک سرور خاص است. پس از اینک لاگ ها با Filebeat تحویل داده شدند و با Logstash پردازش شدند، در کلاستر ElasticSearch قرار می گیرند. از آنجا، لاگ ها برای مصورسازی توسط کیبانا، گرفته می شوند. تمامی مراحل ذکر شده در طرح زیر توضیح داده شده است.

****

شکل 6-4:روند جمع آوری لاگ ها توسط ابزار ELK ]20[

#### مزایای Logstash

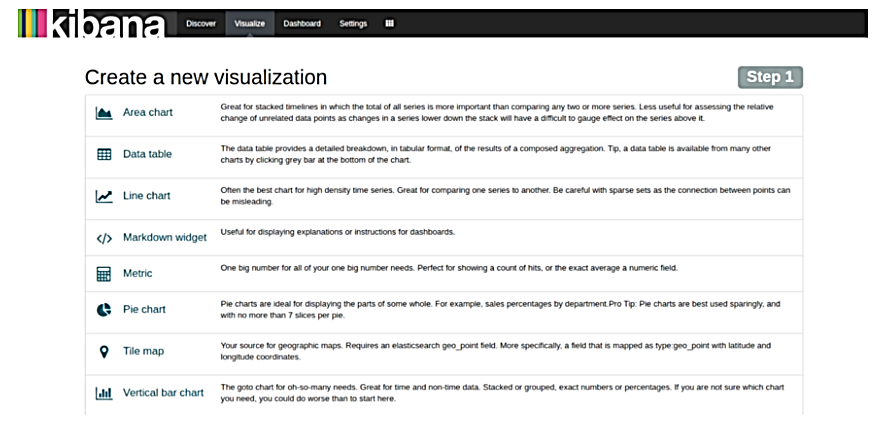
دلیل محبوبیت Logstash این است که می توان آن را به راحتی با سایر محصولات الاستیک ادغام کرد. علاوه بر این، یک دسته کامل از پلاگین ها وجود دارد که این ابزار را بسیار انعطاف پذیر می کند. همچنین با اسناد جامعی همراه است که همه چیز لازم را برای پیکربندی و استفاده از Logstash در تقریباً هر سناریویی دارد. مراحل نصب آن نیز بسیار ساده است.

**معایب Logstash**

نقطه ضعف Logstash همیشه عملکرد و مصرف منابع بوده است. اگرچه عملکرد آن در حال حاضر بسیار بهتر از چند سال پیش است، اما در مقایسه با راه حل های دیگر هنوز بسیار کند است. سرورهای Logstash در مقایسه با سرورهای Elasticsearch، برای استقرار ترافیک بالا، مشکل دارند. و آخرین اما نه کم‌اهمیت تزین مزیت Logstash، این است که فاقد رابط کاربری گرافیکی در نسخه خارج از جعبه است، بنابراین باید با تغییر فایل‌های پیکربندی، آن را به صورت دستی پیکربندی کنید.

**سرویس Kibana**

صفحه زیر مکانی است که می توانید نوع نمایش داده را انتخاب کنید.

****

شکل 6-5: نمایی از نرم افزار kibani ]20[.

پس از انتخاب نوع، از شما پرسیده می شود که از کدام فهرست می خواهید اطلاعات دریافت کنید. انتخاب شاخص بر اساس شاخص‌هایی است که در حال حاضر در خوشه ElasticSearch شما موجود است.

علاوه بر ذخیره داده ها و تجسم آنها، ما از کیبانا برای ارائه اطلاعات دقیق و آماری در مورد نحوه عملکرد برنامه های کاربردی مشتریان در محیط تولید، به آن ها، استفاده می کنیم.

به طور کلی، ELK یک ترکیب همه کاره است. پشته را می توان به عنوان یک برنامه کاربردی مستقل استفاده کرد یا با برنامه های موجود برای دریافت به روزترین داده ها یکپارچه شد.

#### ابزار Graylog

[Graylog](https://www.graylog.org/) یک ابزار قدرتمند برای مدیریت گزارش‌ها است که گزینه‌های زیادی برای تجزیه و تحلیل گزارش‌های ورودی از سرورهای مختلف در اختیار شما قرار می‌دهد. روش کار Graylog تقریباً شبیه به ELK است. علاوه بر سرور Graylog که از برنامه کاربردی و سرور رابط وب تشکیل شده است، برای اینکه بتوانید کل پشته را به طور کامل قابل اجرا کنید، باید MongoDB و Elasticsearch را نیز داشته باشید.

بنابراین موارد زیر، مواردی هستند که برای کار با Graylog به آن ها نیاز دارید:

* Elasticsearch
* MongoDB
* Graylog main server
* Graylog web interface

فرآیند پیکربندی شامل 2 مرحله است:

* ایجاد یک پیکربندی برچسب گذاری شده Graylog برای استفاده توسط مشتریان در سمت سرور.
* ارسال پیکربندی به مشتریان

به عنوان مثال، هنگامی که یک مشتری را سِت می کنید، فایل پیکربندی آن را با یک برچسب "production" علامت گذاری می کنید. پس از آن، تنظیمات دقیقی را روی سرور برای مشتریان، با تگ "production" ایجاد می کنید. اکنون به محض اعمال آن، همه تغییرات به طور خودکار از سرور دریافت می شود. هیچ تغییر دیگری لازم نیست در سمت مشتری انجام شود.

**مزایای Graylog**

رابط کاربری گرافیکی دوستانه ای دارد و از طیف گسترده ای از فرمت های داده پشتیبانی می کند. گزینه های خوبی برای کنترل احراز هویت و تعیین مجوزهای کاربر، در اختیار شما قرار می دهد. همچنین می توانید آن را برای دریافت هشدار، از طریق ایمیل پیکربندی کنید. Graylog از Good-ol REST API استفاده می کند.

**معایب Graylog**

Graylog نمی تواند از فایل های syslog بخواند، بنابراین شما باید پیام های خود را مستقیماً به Graylog ارسال کنید.

#### Graylog یا ELK کدام بهتر است؟

هر دو ابزار از نظر مجموعه ای اساسی از ویژگی ها، بسیار مشابه هستند. با این حال، به عنوان طرفداران Elasticsearch، ما همچنان Graylog را به ELK ترجیح می دهیم، زیرا دارای رابط کاربری گرافیکی دوستانه است و به شما امکان می دهد مجوزها (permissions) را مدیریت کنید. همچنین کیبانا هیچ قابلیتی برای ایجاد کاربر ندارد. با این حال، این مشکل با تنظیم احراز هویت اولیه HTTP در Nginx حل شده است که یک سرور در حال اجرا بر روی یک پورت تصادفی است که درخواست ها را به رابط وب Kibana ارسال می کند.

## پرومتئوس یا Prometheus

پرومتئوس (Prometheus) یکی از [انواع ابزارهای مانیتورینگ سرور](https://zaminhost.com/blog/مانیتورینگ-سرور/) است که به صورت منبع باز بوده و داده ها را بر اساس خط زمانی [[129]](#footnote-129)آن ها جمع آوری می کند]17[.

Prometheus در ابتدا در [Soundcloud](https://soundcloud.com/) توسعه داده شد اما اکنون توسط بنیاد محاسبات بومی ابری (CNCF) پشتیبانی می شود. این ابزار مانیتورینگ در دهه گذشته به سرعت به شهرت رسید، زیرا ترکیبی از ویژگی های خاص آن که از فضای ابری هم پشتیبانی می کنند، آن را به مجموعه مانتیورینگی ایده آل برای برنامه های امروزی تبدیل کرده است.

بخشی از محبوبیت این ابزار مربوط به قابلیت همکاری یا ادغام با سایر پلتفرم‌ های دیگر است که می ‌تواند داده ‌ها را در قالب ‌های راحتتری نمایش دهد. [ادغام Prometheus با Grafana](https://zaminhost.com/blog/ادغام-گرفانا-و-پرومتئوس/" \t "_blank) از جمله اموری است که معمولا انجام می شود. [Grafana](https://zaminhost.com/blog/سرور-گرافانا/" \t "_blank) یک ابزار گرافیکی است که معیارهای Prometheus را نشان می دهد و کار با آن را آسان تر می کند.

Prometheus داده ها یا رویدادها را در لحظه ذخیره می کند. این رویدادها می توانند هر چیز مرتبط با برنامه شما باشند، مانند میزان مصرف حافظه، استفاده از شبکه یا درخواست های ورودی. به واحد داده های جمع آوری شده **“**متریک**”** می گویند. هر معیار، یک نام و مجموعه ای از برچسب ها (Tag) دارد که می توانند برای فیلتر کردن معیارهای موجود در پایگاه داده استفاده شوند.

این معیارها، همیشه براساس یکی از این چهار نوع ابزار اصلی هستند:

-Counter: مقدار یا ارزشی که به طور پیوسته افزایش پیدا می کند و هرگز کم یا مجدد تنظیم نمی شود.

-Gauge: مقدار یا ارزشی که می تواند در هر زمانی به هر جهتی تغییر کند.

-Histogram: نمونه برداری از مقادیر یا ارزش های متعدد که مجموع تمام مقادیر ذخیره شده و همچنین تعداد رویدادهای ثبت شده را ارائه می دهد.

-Summary: همانند هیستوگرام عمل می کند اما از معیارهای قابل تنظیم برای نظارت بر کل دوره های زمانی پشتیبانی می کند.

### **ساختار پرومتئوس**

Prometheus از اجزای متعددی تشکیل شده است که هر یک از آن ها عملکرد خاصی را ارائه می کنند که برای سیستم مانیتورینگ سرور و هشدار، این ابزار حیاتی هستند. این اجزا عبارتند از:

سرور پرومتئوس:

به عنوان مغز سیستم شناخته می شود. سرور، معیارها یا داده ها را در طول خط زمانی جمع آوری و آن ها را ذخیره می کند. فرآیند جمع آوری داده ها به عنوان Scraping نامیده می شود. این داده ها را می توانید با استفاده از تگ ها فیلتر کنید و فقط معیارهای مشخصی از آن ها را بررسی کنید.

به عنوان مثال می خواهید تعداد درخواست های HTTP برای ورود به صفحات مختلف وب را بررسی کنید. یکی از روش های انجام این کار، ایجاد معیارهای متعدد است که هر کدام به یک مسیر مختلف مربوط می شوند مانند:

* http\_requests\_login\_total
* http\_requests\_logout\_total
* http\_requests\_adduser\_total
* http\_requests\_comment\_total

این درخواست ها مربوط به صفحه ورود، صفحه خروج، صفحه addUser و صفحه نظرات است.

اگر در شرایط خاصی نیاز دارید تا تمام درخواست های HTTP را در کوئری های وب سرور خود بررسی کنید، کار کمی دشوار می شود. در این حالت باید تمام مسیرهای برنامه وب تان را بشناسید و تمام معیارهای آن را جستجو کنید. اما با برچسب ها می توانید یک معیار واحد تنظیم کنید، مثلا http\_requests\_total را به همراه یک تگ “Path” به آن اضافه کنید:

http\_requests\_total{path="/login"{

این کار به شما کمک می کند که کل درخواست ها را صرفا با حذف تگ “path” از کوئری ها و یا افزودن آن به درخواست های مربوط به یک صفحه خاص را در قالبی که بالا توضیح دادیم جستجو کنید.

سرور Prometheus به طور فعال داده ها را از برنامه هایی که نظارت می کند در فواصل زمانی خاص جمع آوری یا Scrapes می کند. سرور پرومتئوس با انجام تمام کارهای سنگین از بار کاری کلاینت ها کم می کند، تنها کاری که کلاینت ها باید انجام دهند این است که معیارهای خود را به گونه ای نشان دهند که سرور Prometheus بتواند به آن ها دسترسی داشته باشد.

### **کتابخانه های مشتری و صادرکنندگان**[[130]](#footnote-130)

همانطور که قبلتر هم اشاره کردیم، کلاینت ها باید معیارهای خود را برای جمع آوری توسط Prometheus در دسترس بگذارند اما باید در قالبی باشند که پرومتئوس داده ها را درک کند. Prometheus داده ها را فقط در قالب خط زمانی درک می کند. با این حال، معیارهای قالب بندی شده به صورت خط زمانی از اپلیکیشن ها بیرون نمی آیند. معمولا ابزار دقیقی که این نوع معیار را تولید می کند باید به صورت دستی اضافه شود.

در مواردی که کد منبع را کنترل می کنید، می توانید با چند خط کد و با استفاده از کتابخانه های کلاینت در پرومتئوس، معیارهای مورد نظر خود را اضافه کنید. کتابخانه کلاینت، معیارهای مورد نظر شما را در قالبی ارسال می کند که برای Prometheus قابل درک باشند. اکثر کتابخانه ها همچنین معیارهای خاصی مانند میزان استفاده از CPU و آمار موارد زائد جمع آوری شده و کتابخانه مورد استفاده را هم ارائه می دهند.

اما در مواردی که کنترلی بر روی کد منبع ندارید، امکان کد زدن هم وجود ندارد. برای مثال، می خواهید اطلاعات متریکی درمورد هسته لینوکس داشته باشید. اگرچه خود هسته احتمالاً مکانیزمی برای خروجی اطلاعات متریکی دارد، اما بعید است که این داده ها در قالبی باشند که توسط Prometheus قابل درک و جمع آوری باشند. در این حالت نمی توانیم از کتابخانه استفاده کنیم زیرا به کد منبع هسته لینوکس دسترسی نداریم. در این حالت از صادرکنندگان (Exporters) استفاده می کنیم.

صادرکنندگان نرم افزارهایی هستند که می‌توانید آن ‌ها را در کنار برنامه‌ هایی که به معیارهای شان علاقه دارید، مستقر کنید. صادرکنندگان قطعات ضروری در ابزار مانیتورینگ پرومتئوس هستند. هر برنامه ای که به عنوان مشتری در پرومتئوس عمل می کند، یک صادر کننده را در هسته خود دارد.

یک صادر کننده متشکل از ویژگی‌ های نرم ‌افزاری است که داده ‌های اندازه ‌گیری را تولید و یک سرور HTTP که معیارهای تولید شده را از طریق یک نقطه پایانی مشخص در دسترس قرار می ‌دهد می باشد. معیارها بر اساس فرمت خاصی که سرور Prometheus می تواند بخواند و جمع آوری کند در معرض نمایش قرار می گیرند.

### **موارد استفاده از پرومتئوس**

پرومتئوس اخیرا به عنوان یکی از ابزارهای مانیتورینگ سرور شهرت پیدا کرده است. استفاده آسان، ادغام شدن با دیگر ابزار و برنامه ها و تطبیق پذیری با دستورات، باعث محبوبیت آن در دنیای مانیتورینگ شده است.

این ابزار با اینترنت اشیاء (IOT)، مانیتورینگ سرور ابری و هر مورد دیگری به خوبی سازگاری دارد. البته یکی از ضعف های پرومتئوس عدم توانایی در جمع آوری اطلاعات پروژه های کوتاه مدت است. هر چند Pushgateway برای رسیدگی به جمع آوری این داده ها ایجاد شده است اما خود پرومتئوس به تنهایی قادر به انجام این کار نیست.

Pushgateway به عنوان نوعی حافظه پنهان متریکی برای پروژه های کوتاه مدت عمل می کند. جدا از این نکته، باید بگوییم پرومتئوس می تواند اکثر نیازهای شما را برای مانیتورینگ سرورتان برآورده کند و با توجه به منبع باز بودنش، می توانید در صورت نیاز آن را بسته به نیازهای خود شخصی سازی کنید.

### **پرومتئوس چه وظایفی را نمی تواند انجام دهد**

با توجه به همه صحبت های قبل، تقریبا می توان گفت Prometheus یک بازیکن همه فن حریف است، اما در این میان نقش هایی هم وجود دارند که قادر به انجام شان نیست. این ابزار طبق اصول مهندسی طراحی شده تا به طور دقیق تمام داده ها را جمع آوری کند. اما در کل Prometheus تضمین نمی کند که تمام داده های جمع آوری شده 100% دقیق باشند. البته این مورد برای پروژه های با حجم بالاست. اگر آمار بسیار حساسی را در سرور خود رصد می کنید پس بهتر است ابزارهای دیگر مانیتورینگ را هم بررسی کنید.

# طراحی شبکه هسته نسل 4 و نسل 5

طراحی ساختار ارتباطی بین تجهیزات شبکه یکی از مهمترین عوامل در بهره گیری از حداکثر توان ابزار و تجهیزات و نیز افزایش کیفیت خدمات می باشد. در این زمینه یکی از اساسی ترین و حساس ترین موارد، طرح ریزی و ارایه نقشه ارتباطی بین تجهیزات می باشد.

طرح ریزی ارتباطی بین تجهیزات شبکه باید به صورتی باشد که ملزومات زیر را پشتیبانی و مد نظر قرار دهد :

* شناسایی تمامی درگاه های ارتباطی
* مدیریت IP Pool برای استفاده بهینه
* پیش بینی ارتباط بین تجهیزات و ارایه مسیر بهینه ارتباطی ( جلوگیری از کابل کشی اضافی)
* بهره گیری از Vlan بندی برای تجهیزات حساس شبکه
* دسته بندی مناسب و منطقی برای تجهیزات مستقل از هم جهت ایجاد امکان بروز رسانی
* پیشگیری از شکستن IP Pool ها به اندازه های خیلی کوچک ( در روند رشد شرکت این IP ها تلف می شوند یا مدیریت IP سخت می شود).

از این رو می بایست قبل از شروع هر گونه طراحی سخت افزاری و نرم افزای، نوع و نحوه ارتباطات بین سایتی باید به صورت کاملا دقیق مشخص شده باشد.

## پروتکل های ارتباطی

در زیر لیستی کلی از درگاه های ارتباطی مطرح برای یک شبکه سلولار مطرح می باشد.

هر شبکه مخابراتی از چندین نوع پروتکل ارتباطی و تجهیزات مختلف تشکیل شده است، از این رو برای جلوگیری از تداخل بین سیستمی و شفافیت مسیر های ارتباطی از جدول IP شبکه بهره گرفته می شود.

در هر شبکه سلولار چندین تجهیز و پروتکل ارتباطی مهم و حیاتی وجود دارد. در زیر مختصری از هر یک از مسیر های ارتباطی توضیح داده می شود:

### پروتکل های ارتباطی هسته نسل چهارم :

S1U : برای ایجاد ارتباط بین eNodB و SGW می باشد. هر eNodB یک IP واحد دارد.

SGI : برای ارتباط PGQ با شبکه IP بیرون همانند اینترنت می باشد. به صورتی که IP کاربر ( مبدا) و با IP ( مقصد) از این مسیر خارج می شود. لازم به ذکر است که IP های اختصاص داده شده به کاربران از همین رنج IP ها خواهد بود ( بغیر از IP های Static و Range IP ها ).

S1app : برای تبادل اطلاعات کنترلی بین eNodB و MME در سمت هسته، از این لینک ارتباطی استفاده می شود. هر eNodB یک IP واحد دارد.

RAN : خود eNodeB ها داری یک IP می باشند که توسط Next Hub شناسایی می شود. در داخل هر eNodeB دو قسمت وجود دارد که هر کدام IP های خاص خود را برای ایجاد ارتباط با MME و SGW خواهند داشت ( این IP ها همان IP های S1U و S1app می باشند).

OM : برای مباحث کنترلی و مدیریت سرور ها یک رنج IP در نظر گرفته می شود.

### پروتکل های ارتباطی هسته نسل پنجم :

N3 : برای تبادل اطلاعات بین gNodeB و UPF بکار گرفته می شود.

N6 : تونل و مسیر و رنج IP مورد استفاده UE برای اتصال به دنیای خارج از شبکه داخلی مانند اینترنت می باشد. این IP بین UPF و دنیای بیرون است.

N2 : برای ایجاد ارتباط کنترلی بین gNodeB و AMF سمت هسته نسل پنجم مورد استفاده قرار می گیرد.

RAN : یک IP برای هر gNodeB باید اختصاص یابد.

OM : برای کنترل و مدیریت سرور های یک رنج IP در نظر گرفته می شود.

برای تجهیزات و اطلاعات حساس :

DB/Logs : برای این که سرور دیتابیس و لاگ های سیستم از مهمترین اطلاعات در یک هسته می باشند، از این رو سرور مربوط به دیتابیس و لاگ های سیتمی به صورت مجزا از سرور های هسته طراحی می شود. به عبارتی هسته برای دسترسی به اطلاعات دیتابیس باید از مسیر رنج IP های مشخص شده استعلام نماید.

Master/BCKUP : یک سرور جهت کنترل و مدیریت و ارایه دسترسی های خاص به سایر سرور ها در نظر گرفته می شود. همچنین بکاپ دو ورژن از VM ها به صورت دایمی و خودکار در این سرور ذخیره می گردد. این رنج IP برای ارایه مسیر دسترسی این سرور به سایر سرور ها می باشد.

## جدول شکست IP شبکه

در زیر جدول IP شبکه کلی به ازای هسته نسل 4 و نسل 5 قابل مشاهده می باشد. با توجه به این که برای امنیت بهتر و طراحی اصولی نیاز به ساختار و برنامه ریزی است، در این حوزه نیز ابتدا جدول IP مشخص شده و از روی آن تجهیزات IP دهی می شوند.



جدول 7-1: رنج IP تجهیزات و قسمت های مختلف شبکه سلولار

رنج IP ها به ترتیب برای هسته نسل چهارم و پنجم و در درگاه مربوط به SGI و N6 به ازای سرور های مجزا در جدول زیر قابل مشاهده می باشد.



جدول 7-2: شکست ریز IP برای پروتکل N6 و SGi

رنج IP ها به ترتیب برای هسته نسل چهارم و پنجم و در درگاه مربوط به S1U و N3 به ازای سرور های مجزا در جدول زیر قابل مشاهده می باشد.



جدول 7-3: شکست ریز IP برای پروتکل N3 و S1U

رنج IP ها به ترتیب برای هسته نسل چهارم و پنجم و در درگاه مربوط به S1app و N2 به ازای سرور های مجزا در جدول زیر قابل مشاهده می باشد.



جدول 7-4: شکست ریز IP برای پروتکل N2 و S1app

از نظر سخت افزاری، خروجی هر یک از سرور ها باید به یک سویچ واحد متصل گردد تا بتوانند با سایر سرور ها ارتباط بر قرار کنند. از این رو جدولی برای این که کابل از کدام درگاه سرور و از چه مدلی خارج و به کدام درگاه سویچ متصل می گردد در جدول زیر برای هسته نسل چهارم و پنجم طراحی و ارایه شده است. طبیعی است در زمان اجرا این جداول معیار اجرا خواهند بود.

لازم به ذکر است که برای هر هسته 2 سویچ در نظر گرفته می شود تا در صورت اختلال در هر یک از سویچ ها، سویچ دیگر در مدار فعالیت نماید.



جدول 7-6: نحوه اتصال سخت افزاری سرور های هسته نسل 4 به سویچ اول و نوع کانکتور مورد نظر



جدول 7-7: نحوه اتصال سخت افزاری سرور های هسته نسل 5 به سویچ اول و نوع کانکتور مورد نظر



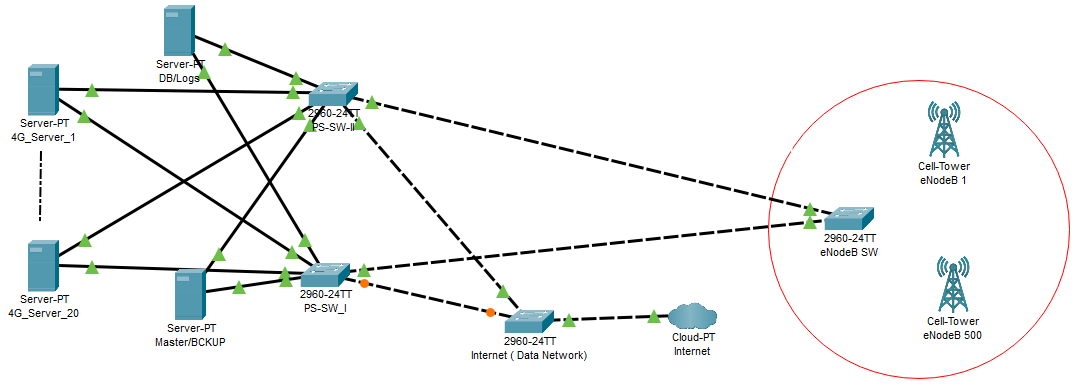
جدول 7-8: نحوه اتصال سخت افزاری سرور های هسته نسل 4 به سویچ دوم و نوع کانکتور مورد نظر



جدول 7-9: نحوه اتصال سخت افزاری سرور های هسته نسل 5 به سویچ دوم و نوع کانکتور مورد نظر

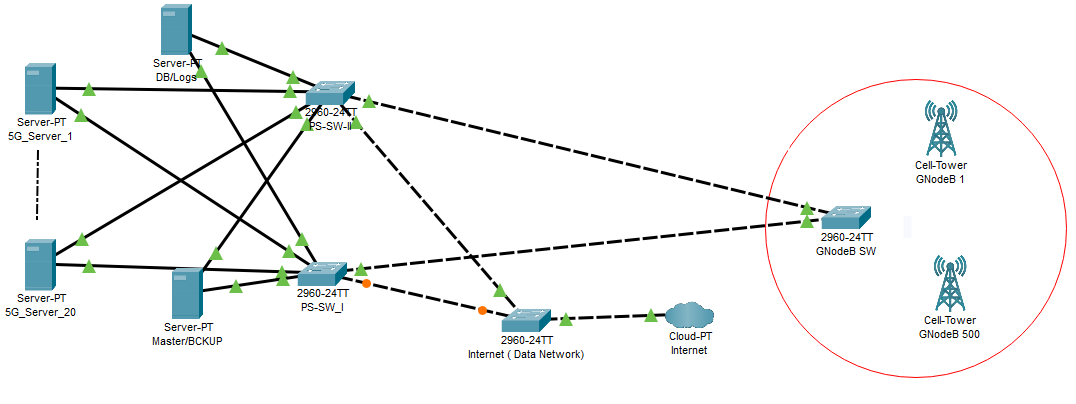
## شماتیک ارتباطی سرور ها و اجزای شبکه سلولار نسل 4 و نسل 5

در شکل زیر ساختار سرور های پردازشگر هسته سلولار نسل 4 مشاهده می شود که توسظ سویچ 2960 گیگ ( سویچ بالایی دست چپ) به همدیگر وصل شده اند، در همچنین با سویچ دیگریه ( سویچ پایین سمت چپ) تمامی سرور ها به شبکه بیرونی دسترسی دارند. دسترسی به شبکه بیرونی به منظور بهره گیری از اینترنت یا دیتا می باشد که باید پروت شبکه ها ظرفیت بالا انتخاب شده باشند ( 10 گیگ فیبر نوری). ساختار سرور های هسته با این طراحی آماده ایجاد ارتباط بین EnodeB ها به شبکه بوده و از طریق سویچ شبکه سمت راست تصویر اطلاعات درخواستی و ارسالی به سایت ها مدیریت و متصل می گردد. سویچ روتر متصل به شبکه اینترنت باید قابلیت پشتیبانی از Netflow را داشته باشد و در اصتلاح رایج به آن Evolved Packet Gateway (EPG) گفته می شود.



شکل 7-1: ساختار شبکه ارتباطی هسته نسل 4

تمامی موارد قید شده برای شکل 1-7 که برای هسته نسل 4 می باشد، برای هسته نسل 5 نیز صادق می باشد. تنها تفاوت مهم بین این دو شبکه نرخ گذردهی بالا می باشد. از این رو اگر در شبکه ارتباطی هسته نسل 4 می شد از درگاه های 1 گیگابیت استفاده نمود، در هسته شبکه سلولار نسل 5 تمامی پروت ها و سویچ ها باید 10G فیبر نوری باشند.



شکل 7-2: ساختار ارتباطی هسته نسل 5

## توانایی تعریف 100 هزار کاربر

اوصولا تعریف کاربر در سمت دیتابیس و HSS مطرح می باشد که به واسطه ایجاد ارتباط و قابلیت خواندن و نوشتن توسط HSS در آن می توان به تعداد کاربران اضافه یا کم نمود. دیتابیس بکارگیری شده در این پروژه از نوع noSQL و MongoDB می باشد. این دیتابیس تا 10 میلیون کاربر را می تواند پشتیبانی نماید. عمده موضوع مطر در مرحله بعدی پروژه سرعت query زدن و دریافت پاسخ می باشد. این امر زمانی خود را نشان خواهد داد که به هر دلیلی شبکه دچار مشکل شده و 1 یا 10 میلیون کاربر از شبکه بیرون شده باشند و حالا می خواهند وارد شبکه شوند. در این حالت سرعت پاسخ گویی دیتابیس امر مهمی می باشد که مانع از Time Out شدن کاربران شود.



شکل 7-2: ساختار ارتباطی هسته نسل 5

## توانایی دستیابی به نرخ گذردهی تا 100 گیگابیت بر ثانیه

با توجه به مشکل نرخ گذردهی در کرنل لیونکس که عمدتا در حالت ایدآل از 1 گیگابیت و در عمل از 500 الی 600 مگابیت بالاتر نمی رود، بنابراین هر VM نمی تواند بیش از 500 مگابیت نرخ گذردهی داشته باشد. از این رو با فرض ایجاد 10 VM در هر سرور و اختصاص آنها به UPF در مجموع می توان از هر سرور نزدیک به 5 گیگابیت بر ثانیه نرخ کشید. از طرفی همانگونه که در پیشنهاد نامه آمده است برای دست یابی به نرخ گذردهی 100 گیگابیت 20 عدد سرور با مشخصات قید شده لازم می باشد. همچنین 2 سرور یکی برای پشتیبان گیری از اطلاعات حساس و دیگری برای دیتابیس و لاگ سیستم ها لازم است که بنابراین برای هریک از هسته های نسل 4 و نسل 5 به صورت جداگانه 22 سرور مورد نیاز می باشد. در مرحله بعدی و با رفتن به سمت DPDK این سامانه توانایی اجرای 1 ترابیت ظرفیت با همین تعداد سرور را فراهم می نماید.

## دارای یک دیتابیس اصلی و یک دیتابیس ذخیره

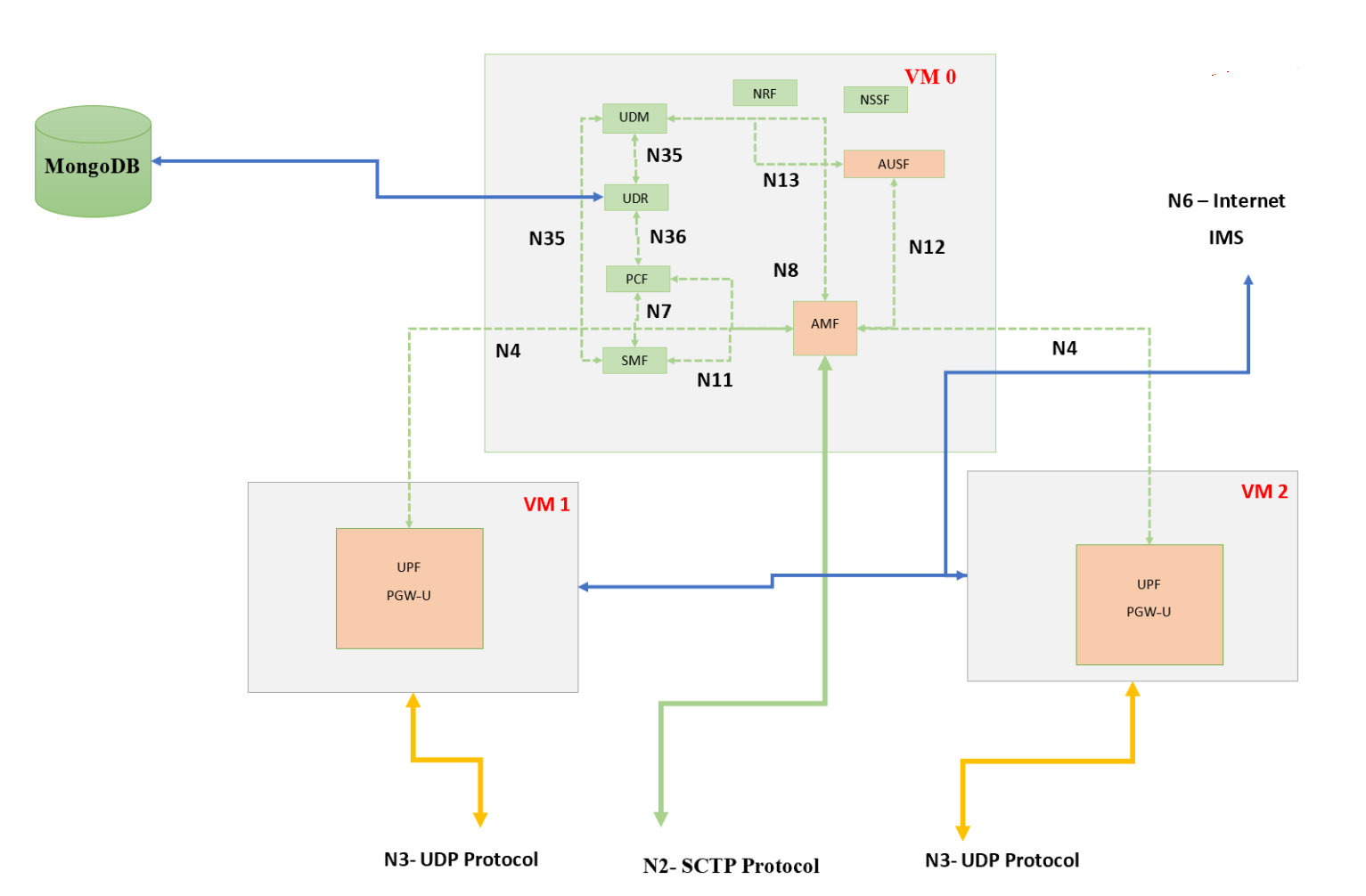
خمانگونه که در شکل های 7-1 و 7-2 قابل مشاهده می باشد، دو سرور جدا برای پشتیبان گیری و دیتابیس و لاگ ها در نظر گرفته شده است. دیتابیس در سرور مربوط به خود به صورت Master/Slave و در دو VM جداگانه اجرایی خواهد شد. از سمت دیگر یک نسخه پشتیبان به صورت اتوماتیک وار در سرور مربوط به پشتیبان گیری ذخیره خواهد گردید.

## قابلیت Failover بودن VM ها

برای ایجاد قابلیت Fialover می توان از سیگنالینگ heart bite استفاده نمود. به این صورت که سرور متناوبا برپا بودن و کارایی هر یک از VM ها را پایش می نماید و در صورت خرابی و یا عدم پاسخگویی در بازه زمانی قید شده به سیگنال heart bite به سرعت VM پشتیبان را بالا می آورد. و آن VM را خاموش می نماید. این تکنیک به صورت موقت کارایی بالایی دارد ولی در مرحله بعدی قرارداد به سمت کوبرنتیز خواهیم رفت که اختیار و سرعت عمل بالایی را فراهم می نماید همچنین مانع از مصرف منابع برای این منظور می شود و در صورت نیاز از فضای آزاد با هوشمندی خودش بلوک های مربوطه را اجرایی می نماید.

## طراحی سامانه به صورت User Plane مجزا

کد استفاده شده در هسته به صورت جدا از هم طراحی شده و نوشته شده است به صورتی که بلوک های Control Plane می توانند در یک VM مجزا و بلوک های User Plane به صورت جدا گانه در یک VM یا سخت افزار مجزا نصب و راه اندازی شوند. این مزیت اجازه می دهد که در صورت نیاز به استقرار User Plane در نزدیکی سایت یا همان eNodeB برای دستیابی به سرعت Ping سریعتر قرار گیرد. در این حالت عمده تبادل اطلاعات مورد نیاز شبکه به صورت محلی انجام می پذیرد و نیازی به افزایش زیرساخت ارتباطی نیست. همچنین برای دسترسی به اینترنت از پروتکل N6 بهره گرفته می شود که به صورت مستقیم مطابع شکل 7-3 به خود UPF یا همان User Plane متصل می گردد. برای ارتباطات درون شبکه ای مانند تماس تصویری یا انتقال دیتا حجم بالا بین دو کاربر که در یک UPF می باشند، این تبادل بدون اشغال شبکه ارتباطی زیرساختی می باشد.



شکل 7-2: نحوه اجرای Multi User Plane

## طراحی سامانه به صورت Multi User Plane

همانگونه که در شکل بالا قابل مشاهده می باشد، کد بکارگرفته شده به دلیل قابلیت جدا سازی User Plane و Control Plane قابلیت اتصال چندین UPF یا همان User Plane را دارا می باشد که در مستندات بعدی و بعد از راه اندازی آنها، نحوه پیکربندی و تغییرات کد مورد نظر بیان و مستندات و کد های آن در اختیار قرار داده خواهد شد.

مراجع

[1] Shurman, Mohammad, et al. "Performance Enhancement in 5G Cellular Networks Using Priorities in Network Slicing." 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT). IEEE, 2019.

[2] Abejide, Adebayo Emmanuel. Performance Analysis of an LTE-4G Network Running Multimedia Applications. Diss. Eastern Mediterranean University (EMU)-Doğu Akdeniz Üniversitesi (DAÜ), 2014.

[3] Jamal, Tasleem, et al. "LTE Network: Performance Analysis Based on Operating Frequency." Recent Trends in Communication, Computing, and Electronics. Springer, Singapore, 2019. 125-135.

[4] Satake, Hiroshi, et al. "Government reference architecture extensions for application to base stations." MILCOM 2013-2013 IEEE Military Communications Conference. IEEE, 2013.

[5] Ulema, Mehmet. Fundamentals of Public Safety Networks and Critical Communications Systems: Technologies, Deployment, and Management. John Wiley & Sons, 2019.

[6] <https://www.netmanias.com/en/post/techdocs/5904/lte-network> architecture/lte-network-architecture-basic

[7] <https://docs.srslte.com/en/latest/srsenb/source/1_enb_intro.html>.

[8] <https://lteunlimited.blogspot.com/2015/11/lte-protocol-architecture.html>.

[9]<https://www.netmanias.com/en/post/techdocs/5904/lte-network-architecture/lte-network-architecture-basic>

[10] Holma, Harri, Antti Toskala, and Takehiro Nakamura, eds. 5G technology: 3GPP new radio.john wiley & Sons.2020.

[11] 3GPP Tdoc RP-161266 (2016).5G architecture options – full set, Deutsche Telecom. June 2016.

[12] O-RAN Alliance. <https://www.o-ran.org>

[13] <https://open5gs.org/open5gs/docs/guide/01-quickstart/>

[14] <https://open5gs.org/open5gs/docs/guide/02-building-open5gs-from-sources/>

[15] <http://open5gs.info/core/>

[16] https://sadidafarin.ir/zabbix-network-monitoring-software

[17]https://zaminhost.com/blog/%D9%BE%D8%B1%D9%88%D9%85%D8%AA%D8%A6%D9%88%D8%B3-prometheus/

[18]https://avashnet.com/whats-network-monitoring/

[19] <https://www.danapardaz.net/blog/network-monitoring-everything-need-to-know/>

[20] https://virgool.io/@aminbagheri021/%D9%85%D8%B9%D8%B1%D9%81%DB%8C-%D8%A7%D8%A8%D8%B2%D8%A7%D8%B1-%D9%87%D8%A7%DB%8C-%D9%85%D8%AF%DB%8C%D8%B1%DB%8C%D8%AA-log-lnfu6tfqchtf



1. Third-generation partnership project [↑](#footnote-ref-1)
2. High speed packet access [↑](#footnote-ref-2)
3. Multimedia broadcast multicast service [↑](#footnote-ref-3)
4. Integrated measurement system [↑](#footnote-ref-4)
5. Wireless local area network [↑](#footnote-ref-5)
6. Quality of service [↑](#footnote-ref-6)
7. Voice Over Internet Protocol [↑](#footnote-ref-7)
8. # Evolved High Speed Packet Access

   [↑](#footnote-ref-8)
9. Near-field communication [↑](#footnote-ref-9)
10. Enhanced GPRS [↑](#footnote-ref-10)
11. Orthogonal frequency-division multiple access [↑](#footnote-ref-11)
12. High Speed Downlink Packet Access [↑](#footnote-ref-12)
13. Worldwide Interoperability for Microwave Access [↑](#footnote-ref-13)
14. Universal Mobile Telecommunications Service [↑](#footnote-ref-14)
15. High Speed Uplink Packet Access [↑](#footnote-ref-15)
16. Home eNodeB [↑](#footnote-ref-16)
17. Heterogeneous networks [↑](#footnote-ref-17)
18. Coordinated Multipoint [↑](#footnote-ref-18)
19. In device Coexistence [↑](#footnote-ref-19)
20. Carrier aggregation [↑](#footnote-ref-20)
21. Frequency Division Multiplexing/Time Division Duplexing [↑](#footnote-ref-21)
22. Device to device [↑](#footnote-ref-22)
23. Licensed Assisted Access [↑](#footnote-ref-23)
24. Multi-user Supposition Transmissions [↑](#footnote-ref-24)
25. Voice Over LTE [↑](#footnote-ref-25)
26. Ultra-Reliable Low-Latency Communication [↑](#footnote-ref-26)
27. Mobile Terminated Call [↑](#footnote-ref-27)
28. dynamic spectrum sharing [↑](#footnote-ref-28)
29. New radio [↑](#footnote-ref-29)
30. control plane [↑](#footnote-ref-30)
31. user (data) plane [↑](#footnote-ref-31)
32. LTE radio access network [↑](#footnote-ref-32)
33. aggregation [↑](#footnote-ref-33)
34. Single-carrier frequency-division multiple access [↑](#footnote-ref-34)
35. Sub-carriers [↑](#footnote-ref-35)
36. Hybrid Aoutomatic Repeat Request [↑](#footnote-ref-36)
37. Heterogeneous networks [↑](#footnote-ref-37)
38. Machine to Machine [↑](#footnote-ref-38)
39. Not orthonormal multiple access [↑](#footnote-ref-39)
40. Full duplex [↑](#footnote-ref-40)
41. Evolved packet system [↑](#footnote-ref-41)
42. Packet Data Network [↑](#footnote-ref-42)
43. User equipment [↑](#footnote-ref-43)
44. Mobile Equipment [↑](#footnote-ref-44)
45. Mobile Terminal [↑](#footnote-ref-45)
46. Terminal Equipment [↑](#footnote-ref-46)
47. Universal Integrated Circuit Card [↑](#footnote-ref-47)
48. Universal Subscriber Identity Module [↑](#footnote-ref-48)
49. Evolved node base station [↑](#footnote-ref-49)
50. Radio network controller [↑](#footnote-ref-50)
51. Serving Gateway [↑](#footnote-ref-51)
52. Mobility management entity [↑](#footnote-ref-52)
53. Donor eNB [↑](#footnote-ref-53)
54. PDN Gateway [↑](#footnote-ref-54)
55. Policy Control and Charging Rules Function [↑](#footnote-ref-55)
56. Home Subscriber Server [↑](#footnote-ref-56)
57. attachment [↑](#footnote-ref-57)
58. Bearer management [↑](#footnote-ref-58)
59. Cell Broadcast Centre [↑](#footnote-ref-59)
60. HeNB Gateway [↑](#footnote-ref-60)
61. Proximity-based Services [↑](#footnote-ref-61)
62. Stream Control Transmission Protocol [↑](#footnote-ref-62)
63. Network Reference Model [↑](#footnote-ref-63)
64. Operations Support Systems / Business Support Systems [↑](#footnote-ref-64)
65. Radio admission control [↑](#footnote-ref-65)
66. Radio bearer control [↑](#footnote-ref-66)
67. Inter cell interference coordination [↑](#footnote-ref-67)
68. Transparent Mode [↑](#footnote-ref-68)
69. Unacknowledged Mode [↑](#footnote-ref-69)
70. Acknowledged Mode [↑](#footnote-ref-70)
71. concatenation [↑](#footnote-ref-71)
72. segmentation [↑](#footnote-ref-72)
73. Header [↑](#footnote-ref-73)
74. System Information Blocks [↑](#footnote-ref-74)
75. S1 Application Protocol [↑](#footnote-ref-75)
76. GPRS Tunnelling Protocol User Plane [↑](#footnote-ref-76)
77. Tunneling [↑](#footnote-ref-77)
78. Sequence Number [↑](#footnote-ref-78)
79. self-organizing network [↑](#footnote-ref-79)
80. Evolved Packet Core [↑](#footnote-ref-80)
81. user plane [↑](#footnote-ref-81)
82. Mobile Broadband [↑](#footnote-ref-82)
83. Service-Based Architecture [↑](#footnote-ref-83)
84. communication service providers [↑](#footnote-ref-84)
85. distributed edge cloud [↑](#footnote-ref-85)
86. stateless [↑](#footnote-ref-86)
87. Unified Data Management [↑](#footnote-ref-87)
88. Open ecosystem [↑](#footnote-ref-88)
89. multi-access support [↑](#footnote-ref-89)
90. virtual private service networks [↑](#footnote-ref-90)
91. standardized interfaces [↑](#footnote-ref-91)
92. communication service providers [↑](#footnote-ref-92)
93. Service-Based Architecture [↑](#footnote-ref-93)
94. virtual network functions [↑](#footnote-ref-94)
95. Access and MobilityManagement Function [↑](#footnote-ref-95)
96. Session management [↑](#footnote-ref-96)
97. Session Management Function [↑](#footnote-ref-97)
98. Location Management Function [↑](#footnote-ref-98)
99. User Plane Function [↑](#footnote-ref-99)
100. protocol data unit [↑](#footnote-ref-100)
101. Unified Data Repository [↑](#footnote-ref-101)
102. Unstructured Data Storage Function [↑](#footnote-ref-102)
103. Policy Control Function [↑](#footnote-ref-103)
104. Network Exposure Function [↑](#footnote-ref-104)
105. Network Repository Function [↑](#footnote-ref-105)
106. Domain Name System [↑](#footnote-ref-106)
107. Network Slice Selection [↑](#footnote-ref-107)
108. Non-3GPP Interworking Function [↑](#footnote-ref-108)
109. Auxiliary 5G Core Functions [↑](#footnote-ref-109)
110. Short Message Service Function [↑](#footnote-ref-110)
111. Authentication Server Function [↑](#footnote-ref-111)
112. Security Edge Protection Proxy [↑](#footnote-ref-112)
113. Network Data Analytics Function [↑](#footnote-ref-113)
114. Location Management Function [↑](#footnote-ref-114)
115. Central Unit [↑](#footnote-ref-115)
116. Distributed Unit [↑](#footnote-ref-116)
117. Next Generation Core [↑](#footnote-ref-117)
118. radio access [↑](#footnote-ref-118)
119. Common Public Radio Interface [↑](#footnote-ref-119)
120. access-agnostic [↑](#footnote-ref-120)
121. Stream Control Transmission Protocol [↑](#footnote-ref-121)
122. Traffic Flow Template [↑](#footnote-ref-122)
123. Multi-Access Point Name [↑](#footnote-ref-123)
124. Converged Multi-Access and Core [↑](#footnote-ref-124)
125. Traditional circuit switching [↑](#footnote-ref-125)
126. Control/user plane separation [↑](#footnote-ref-126)
127. proactive [↑](#footnote-ref-127)
128. reactive [↑](#footnote-ref-128)
129. Time series [↑](#footnote-ref-129)
130. Client Libraries and Exporters [↑](#footnote-ref-130)